TRƯỜNG ĐẠI HỌC KINH TẾ

**KHOA THỐNG KÊ – TIN HỌC**

–––––––––––––––––––––––––––––––

s

**BÁO CÁO THỰC TẬP NHẬN THỨC**

**HỆ THỐNG QUẢN LÝ**

**THÔNG TIN SINH VIÊN**

**Nhóm: 03**

**Sinh viên thực hiện:**

1. Trưởng nhóm: Chế Thị Nhã Quyên Lớp: 45K14
2. Thành viên: Trần Đỗ Hòa Lớp: 45K14

**Giảng viên hướng dẫn:** TS.Hoàng Thị Thanh Hà

***Đà Nẵng, 6/2021.***

# DANH SÁCH THÀNH VIÊN THAM GIA

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Họ và tên** | **Lớp** | **Nhiệm vụ** |
| 1 | Chế Thị Nhã Quyên | 45K14 | Trưởng nhóm |
| 2 | Trần Đỗ Hòa | 45K14 | Thành viên |

# LỜI CẢM ƠN

Trước tiên chúng em xin gửi đến các quý Thầy Cô khoa Thống kê – Tin học bộ môn Hệ thống thông tin quản lý đã tạo điều kiện cho chúng em học hỏi và trải nghiệm kỳ thực tập nhận thức lời cảm ơn chân thành.

Đặc biệt, chúng em xin gửi đến cô Hoàng Thị Thanh Hà – người đã tận tình hướng dẫn, giúp đỡ chúng em hoàn thành đề tài báo cáo thực tập nhận thức lần này lời cảm ơn sâu sắc nhất.

Với điều kiện thời gian cũng như kinh nghiệm còn hạn chế của một sinh viên năm hai nên bài báo cáo này không thể tránh được những thiếu sót. Chúng em rất mong nhận được ý kiến góp ý của thầy cô để có thêm được nhiều kinh nghiệm hơn và hoàn thành tốt hơn bài báo cáo thực tập nghề nghiệp và thực tập tốt nghiệp trong những năm tới.

Chúng em xin chân thành cảm ơn!

# LỜI CAM ĐOAN

Chúng em xin cam đoan đề tài: ***“Hệ thống quản lý thông tin sinh viên”*** là kết quả nghiên cứu và thực hiện dưới sự hướng dẫn của giáo viên hướng dẫn: TS. Hoàng Thị Thanh Hà. Ngoài ra không có bất cứ sự sao chép nào của người khác.

Đề tài, nội dung báo cáo thực tập nhận thức là sản phẩm mà chúng em đã nghiên cứu và thực hiện trong quá trình tự học tại nhà từ các video tham khảo mà giáo viên hướng dẫn đã đưa ra. Kết quả trình bày trong báo cáo là hoàn toàn trung thực, chúng em xin chịu hoàn toàn trách nhiệm, kỷ luật của bộ môn và nhà trường đề ra nếu như có vấn đề xảy ra.

# MỤC LỤC

[DANH SÁCH THÀNH VIÊN THAM GIA i](#_Toc76112950)

[LỜI CẢM ƠN i](#_Toc76112951)

[LỜI CAM ĐOAN ii](#_Toc76112952)

[MỤC LỤC iii](#_Toc76112953)

[DANH MỤC HÌNH ẢNH v](#_Toc76112954)

[DANH MỤC BẢNG BIỂU vii](#_Toc76112955)

[DANH MỤC CÁC TỪ VIẾT TẮT viii](#_Toc76112956)

[PHÂN CHIA CÔNG VIỆC ix](#_Toc76112957)

[LỜI MỞ ĐẦU xii](#_Toc76112958)

[CHƯƠNG 1. MÔ TẢ BÀI TOÁN 1](#_Toc76112959)

[1.1. Lý do chọn đề tài 1](#_Toc76112960)

[1.2. Tóm tắt bài toán 1](#_Toc76112961)

[1.2.1. Tóm tắt hoạt động của hệ thống 1](#_Toc76112962)

[1.2.2. Thông tin sử dụng trong hệ thống 1](#_Toc76112963)

[1.3. Mô tả tổng quan về hệ thống 1](#_Toc76112964)

[1.3.1. Chức năng của hệ thống 1](#_Toc76112965)

[1.3.2. Phần mềm hỗ trợ thiết lập hệ thống 2](#_Toc76112966)

[1.3.3. Cách thức hoạt động của hệ thống 2](#_Toc76112967)

[CHƯƠNG 2. PHÂN TÍCH HỆ THỐNG, THIẾT KẾ CƠ SỞ DỮ LIỆU 3](#_Toc76112968)

[2.1. Phân tích hệ thống 3](#_Toc76112969)

[2.1.1. Sơ đồ chức năng của hệ thống 3](#_Toc76112970)

[2.1.2. Lưu đồ công việc của hệ thống 4](#_Toc76112971)

[2.2. Thiết kế mô hình cơ sở dữ liệu 7](#_Toc76112972)

[2.2.1. Các thực thể có trong hệ thống 7](#_Toc76112973)

[2.2.2. Thuộc tính và khóa của các thực thể 7](#_Toc76112974)

[2.2.3. Mô hình Thực thể - Mối quan hệ (E-R) 8](#_Toc76112975)

[2.2.4. Các lược đồ quan hệ 8](#_Toc76112976)

[2.2.5. Mô tả thuộc tính các quan hệ 9](#_Toc76112977)

[2.2.6. Mô hình CSDL (Database Diagram) 11](#_Toc76112978)

[CHƯƠNG 3. TRIỂN KHAI HỆ THỐNG 12](#_Toc76112979)

[3.1. Form đăng nhập hệ thống 12](#_Toc76112980)

[3.2. Giao diện form chính của hệ thống 13](#_Toc76112981)

[3.2.1. Quản lý sinh viên 13](#_Toc76112982)

[3.2.2. Quản lý khoa 21](#_Toc76112983)

[3.2.3. Quản lý lớp sinh hoạt 24](#_Toc76112984)

[3.2.4. Quản lý môn học 24](#_Toc76112985)

[3.3. Form thống kê số lượng sinh viên 25](#_Toc76112986)

[3.4. Lưu bảng thống kê số lượng sinh viên 28](#_Toc76112987)

[KẾT QUẢ 30](#_Toc76112988)

[a. Kết quả đạt được 30](#_Toc76112989)

[b. Ưu điểm của hệ thống 30](#_Toc76112990)

[c. Nhược điểm của hệ thống 30](#_Toc76112991)

[KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 31](#_Toc76112992)

[TÀI LIỆU THAM KHẢO 32](#_Toc76112993)

[PHỤ LỤC 33](#_Toc76112994)

# DANH MỤC HÌNH ẢNH

[Hình 2. 1: Sơ đồ chức năng 3](#_Toc75966628)

[Hình 2. 2: Lưu đồ đăng nhập hệ thống 4](#_Toc75966629)

[Hình 2. 3: Lưu đồ thêm thông tin sinh viên 4](#_Toc75966630)

[Hình 2. 4: Lưu đồ sửa thông tin sinh viên 5](#_Toc75966631)

[Hình 2. 5: Xóa thông tin sinh viên 5](#_Toc75966632)

[Hình 2. 6: Lưu đồ tìm kiếm thông tin sinh viên 6](#_Toc75966633)

[Hình 2. 7: Lưu đồ thống kê số lượng sinh viên theo mã lớp 6](#_Toc75966634)

[Hình 2. 8: Lưu đồ thoát khỏi hệ thống 7](#_Toc75966635)

[Hình 2. 9: Mô hình E-R 8](#_Toc75966636)

[Hình 2. 10: Sơ đồ CSDL 11](#_Toc75966637)

[Hình 3. 1: Form đăng nhập hệ thống 12](#_Toc76239077)

[Hình 3. 2: Form đăng nhập sai 12](#_Toc76239078)

[Hình 3. 3: Form giao diện chính 13](#_Toc76239079)

[Hình 3. 4: Form Sinh viên 14](#_Toc76239080)

[Hình 3. 5: Form báo lỗi không nhập đầy đủ thông tin 15](#_Toc76239081)

[Hình 3. 6: Form báo lỗi nhập số điện thoại không hợp lệ 15](#_Toc76239082)

[Hình 3. 7: Form báo lỗi nhập mã sinh viên không hợp lệ 16](#_Toc76239083)

[Hình 3. 8: Form báo lỗi mã sinh viên đã tồn tại 16](#_Toc76239084)

[Hình 3. 9: Form thông báo thêm thành công 17](#_Toc76239085)

[Hình 3. 10: Form hiển thị sinh viên vừa thêm mới 17](#_Toc76239086)

[Hình 3. 11: Form chọn sinh viên muốn sửa 18](#_Toc76239087)

[Hình 3. 12: Form báo lỗi sửa số điện thoại không hợp lệ 19](#_Toc76239088)

[Hình 3. 13: Form thông báo sửa thành công 19](#_Toc76239089)

[Hình 3. 14: Form thông báo xóa thành công 20](#_Toc76239090)

[Hình 3. 15: Form tìm kiếm sinh viên theo mã sinh viên 21](#_Toc76239091)

[Hình 3. 16: Form Khoa 22](#_Toc76239092)

[Hình 3. 17: Form báo lỗi không thể xóa 23](#_Toc76239093)

[Hình 3. 18: Form thông báo xóa thành công 23](#_Toc76239094)

[Hình 3. 19: Form Lớp sinh hoạt 24](#_Toc76239095)

[Hình 3. 20: Form Môn học 25](#_Toc76239096)

[Hình 3. 21: Form thống kê theo lớp 26](#_Toc76239097)

[Hình 3. 22: Form thống kê theo khoa 26](#_Toc76239098)

[Hình 3. 23: Form thống kê theo phường/xã 27](#_Toc76239099)

[Hình 3. 24: Form thống kê theo quận/huyện 27](#_Toc76239100)

[Hình 3. 25: Form thống kê theo tỉnh/thành phố 28](#_Toc76239101)

[Hình 3. 26: Lưu bảng thống kê thành file Word 28](#_Toc76239102)

[Hình 3. 27: Lưu bảng thống kê thành file Excel 29](#_Toc76239103)

[Hình 3. 28: Form thoát khỏi hệ thống 29](#_Toc76239104)

# DANH MỤC BẢNG BIỂU

[Bảng 2. 1: Bảng SINHVIEN 9](#_Toc76226492)

[Bảng 2. 2: Bảng KHOA 10](#_Toc76226493)

[Bảng 2. 3: Bảng LOPSH 10](#_Toc76226494)

[Bảng 2. 4: Bảng MONHOC 10](#_Toc76226495)

# DANH MỤC CÁC TỪ VIẾT TẮT

|  |  |  |
| --- | --- | --- |
| **Thứ tự** | **Kí hiệu chữ viết tắt** | **Chữ viết đầy đủ** |
| 1 | CNTT | Công nghệ thông tin |
| 2 | CSDL | Cơ sở dữ liệu |
| 3 | NQL | Người quản lý |

# PHÂN CHIA CÔNG VIỆC

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **STT** | **Họ tên** | **Lớp** | **Nội dung công việc** | **Thời gian bắt đầu** | **Thời gian kết thúc** | **Kết quả** |
| 1 | Chế Thị Nhã Quyên  Trần Đỗ Hòa | 45K14 | -Tham khảo các tài liệu  -Lên ý tưởng hệ thống  -Thiết kế các bảng, các thuộc tính | 11/6/2021 | 11/6/2021 | Hoàn thành |
| 2 | Chế Thị Nhã Quyên | 45K14 | Thiết lập CSDL: tạo các bảng, các thuộc tính, các kiểu dữ liệu từng thuộc tính, liên kết giữa các bảng, ràng buộc khóa chính, khóa ngoại | 11/6/2021 | 11/6/2021 | Hoàn thành |
| 3 | Trần Đỗ Hòa | 45K14 | Thiết kế giao diện Form | 11/6/2021 | 12/6/2021 | Hoàn thành |
| 4 | Chế Thị Nhã Quyên | 45K14 | Chuẩn bị dữ liệu và nhập dữ liệu vào cơ sở dữ liệu | 12/6/2021 | 12/6/2021 | Hoàn thành |
| 5 | Trần Đỗ Hòa | 45K14 | Thiết lập liên kết giữa các Form | 12/6/2021 | 13/6/2021 | Hoàn thành |
| 6 | Chế Thị Nhã Quyên | 45K14 | Viết báo cáo Chương 1: Mô tả bài toán | 12/6/2021 | 13/6/2021 | Hoàn thành |
| 7 | Trần Đỗ Hòa | 45K14 | Lập trình C# cho form đăng nhập | 13/6/2021 | 13/6/2021 | Hoàn thành |
| 8 | Chế Thị Nhã Quyên | 45K14 | Vẽ lưu đồ công việc của hệ thống | 14/6/2021 | 14/6/2021 | Hoàn thành |
| 9 | Trần Đỗ Hòa | 45K14 | Vẽ sơ đồ chức năng của hệ thống | 14/6/2021 | 14/6/2021 | Hoàn thành |
| 10 | Chế Thị Nhã Quyên | 45K14 | Lập trình chức năng thêm và xóa cho bảng Sinh viên | 14/6/2021 | 15/6/2021 | Hoàn thành |
| 11 | Trần Đỗ Hòa | 45K14 | Lập trình chức năng sửa, khởi tạo và tìm kiếm cho bảng Sinh viên | 14/6/2021 | 15/6/2021 | Hoàn thành |
| 12 | Chế Thị Nhã Quyên | 45K14 | Viết và hoàn thiện báo cáo chương 2 | 15/6/2021 | 16/6/2021 | Hoàn thành |
| 13 | Trần Đỗ Hòa | 45K14 | Lập trình các chức năng cho các form còn lại | 15/6/2021 | 16/6/2021 | Hoàn thành |
| 14 | Chế Thị Nhã Quyên | 45K14 | Tạo Stored Procedure trên SQL và lập trình thống kê sinh viên theo lớp, khoa, phường/xã | 20/6/2021 | 21/6/2021 | Hoàn thành |
| 15 | Trần Đỗ Hòa | 45K14 | Lập trình thống kê sinh viên theo quận/huyện, tỉnh/thành phố | 20/6/2021 | 21/6/2021 | Hoàn thành |
| 16 | Chế Thị Nhã Quyên | 45K14 | Viết báo cáo chương 3 và phần ưu, nhược điểm | 26/6/2021 | 28/6/2021 | Hoàn thành |
| 17 | Trần Đỗ Hòa | 45K14 | Viết báo cáo phần kết quả đạt được | 30/6/2021 | 1/7/2021 | Hoàn thành |
| 18 | Chế Thị Nhã Quyên | 45K14 | Hoàn chỉnh bảng phân chia công việc và hoàn thiện báo cáo | 30/6/2021 | 2/7/2021 | Hoàn thành |
| 19 | Chế Thị Nhã Quyên  Trần Đỗ Hòa | 45K14 | Chỉnh sửa và hoàn thiện form | 1/7/2021 | 1/7/2021 | Hoàn thành |
| 20 | Chế Thị Nhã Quyên | 45K14 | Làm slide báo cáo | 3/7/2021 | 4/7/2021 | Hoàn thành |
| 21 | Trần Đỗ Hòa | 45K14 | Quay video demo hệ thống | 4/7/2021 | 4/7/2021 | Hoàn thành |

# LỜI MỞ ĐẦU

1. **Mục tiêu nghiên cứu của đề tài**

Yêu cầu xây dựng một hệ thống quản lý thông tin sinh viên của trường Đại học Kinh tế - Đại học Đà nẵng nhằm hỗ trợ cho công tác quản lý sinh viên, khoa, lớp, môn học với các chức năng cơ bản: đọc, ghi, sửa, xóa, tìm kiếm thông tin và thống kê.

1. **Nhiệm vụ của đề tài**

Nghiên cứu, phân tích và thiết kế hệ thống quản lý thông tin sinh viên để công việc có hiệu quả, tiết kiệm được thời gian cho người quản lý.

1. **Kết cấu của đề tài**

Đề tài được tổ chức gồm phần mở đầu, 4 chương nội dung và phần kết luận và hướng phát triển.

* Mở đầu
* Chương 1: Mô tả bài toán
* Chương 2: Phân tích hệ thống, thiết kế hệ thống và thực thi cơ sở dữ liệu
* Chương 3: Triển khai hệ thống
* Chương 4: Kết quả
* Kết luận và hướng phát triển

# 

# MÔ TẢ BÀI TOÁN

## Lý do chọn đề tài

Hiện nay, tổng số sinh viên đang theo học tại Trường Đại học Kinh tế - Đại học Đà Nẵng là trên 10.000 người. Với số lượng sinh viên ngày càng đông, nhu cầu thông tin của sinh viên phải cập nhật (thêm, xóa) và tìm kiếm thường xuyên gây khó khăn trong việc quản lý sinh viên. Vì vậy, nó đòi hỏi phải có một hệ thống giúp cho việc quản lý một cách nhanh chóng.

Hiểu được nhu cầu này, chúng em đã xây dựng nên *“Hệ thống quản lý thông tin sinh viên”* này hy vọng sẽ giúp cho công tác quản lý sinh viên diễn ra dễ dàng, đơn giản và tiết kiệm thời gian hơn.

## Tóm tắt bài toán

### Tóm tắt hoạt động của hệ thống

Mô tả môi trường: bộ phận CNTT của trường sẽ thực hiện quản lý thông tin của sinh viên theo khoa, với mỗi khoa sẽ chia thành nhiều lớp sinh hoạt, mỗi lớp có một danh sách sinh viên trực thuộc, mỗi môn học sẽ do một khoa quản lý.

Việc quản lý thông tin sinh viên như sau: mỗi sinh viên sẽ được cấp cho một mã sinh viên và dùng nó làm khóa chính để quản lý thông tin liên quan đến thông tin cá nhân, khoa, lớp sinh hoạt, môn học. *Lưu ý: Mã sinh viên này không thay đổi trong suốt quá trình học tập và quản lý thông tin của sinh viên*.

### Thông tin sử dụng trong hệ thống

Hệ thống bao gồm các thông tin chi tiết về thông tin cá nhân, khoa, lớp sinh hoạt, môn học.

## Mô tả tổng quan về hệ thống

### Chức năng của hệ thống

Hệ thống bao gồm việc quản lý khoa, quản lý lớp sinh hoạt, quản lý môn học, quản lý sinh viên, quản lý môn học và được tích hợp các chức năng liên quan đến quản lý thông tin sinh viên, có khả năng thực hiện/thao tác với các chức năng: xem, thêm, sửa, xóa, tìm kiếm, sao lưu các dữ liệu khi thay đổi thông tin, bảo toàn trọn vẹn các thông tin một cách tối ưu nhất.

### Phần mềm hỗ trợ thiết lập hệ thống

Được triển khai trong môi trường:

* Visual Studio 2013 với ngôn ngữ lập trình C#
* Hệ quản trị CSDL SQL Server.
* Ngoài ra, trong quá trình thực tập để xây dựng hệ thống có sử dụng phần mềm TeamViewer và GitHub Desktop.

### Cách thức hoạt động của hệ thống

Bộ phận CNTT – NQL sẽ đăng nhập vào hệ thống bằng tài khoản và mật khẩu của mình. Hệ thống sẽ kiểm tra tên tài khoản và mật khẩu đã chính xác hay chưa: nếu đúng thì truy cập vào giao diện chính, nếu không đúng – nhập lại tài khoản và mật khẩu hoặc kết thúc sử dụng. Sau khi bước vào giao diện chính, NQL có quyền sử dụng thông tin và thao tác/thực hiện các chức năng xem, thêm mới, sửa, xóa, tìm kiếm, thống kê các thông tin liên quan đến sinh viên giúp cho công tác quản lý sinh viên được thực thi nhanh gọn, dễ dàng và chính xác trong thời gian ngắn.

Những người không phải là NQL không được cung cấp tài khoản sẽ không có quyền truy cập vào hệ thống, vì vậy tạo nên tính bảo mật một cách tốt nhất.

# PHÂN TÍCH HỆ THỐNG, THIẾT KẾ CƠ SỞ DỮ LIỆU

## Phân tích hệ thống

### Sơ đồ chức năng của hệ thống

Sơ đồ chức năng dưới đây là toàn bộ mô phỏng khái quát và chi tiết về chức năng quản lý thông tin sinh viên của hệ thống để biết được nó có thể làm những gì. Biểu đồ được phân mức từ trên xuống:

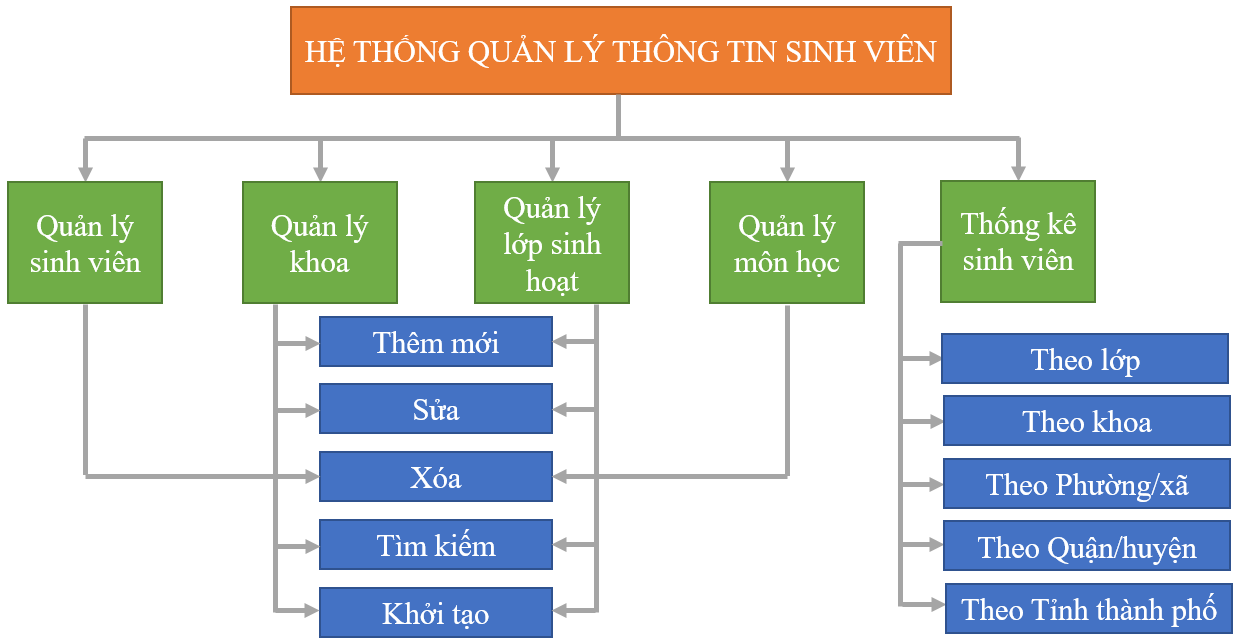
Chức năng 1: Thêm thông tin sinh viên, lớp sinh hoạt, khoa, môn học

Chức năng 2: Sửa thông tin sinh viên, lớp sinh hoạt, khoa, môn học

Chức năng 3: Xóa thông tin sinh viên, lớp sinh hoạt, khoa, môn học

Chức năng 4: Tìm kiếm thông tin sinh viên, lớp sinh hoạt, khoa, môn học

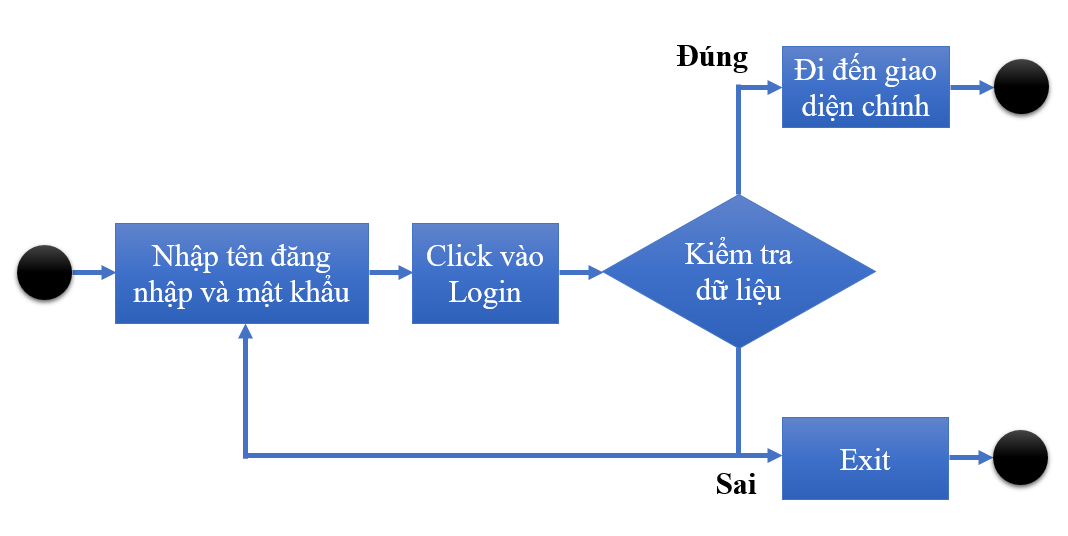
Chức năng 5: Thống kê thông tin sinh viên, lớp sinh hoạt, khoa, môn học



Hình 2. 1: Sơ đồ chức năng

### Lưu đồ công việc của hệ thống

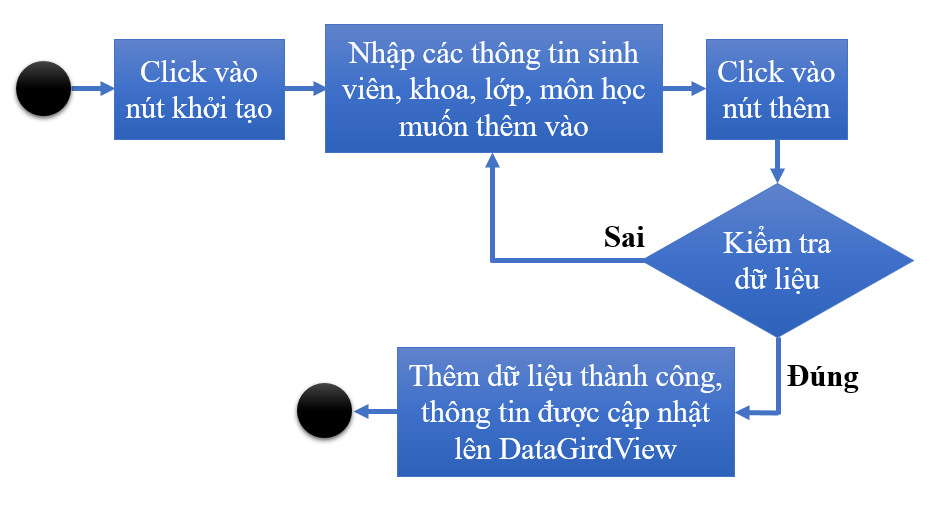
#### Lưu đồ đăng nhập



Hình 2. 2: Lưu đồ đăng nhập hệ thống

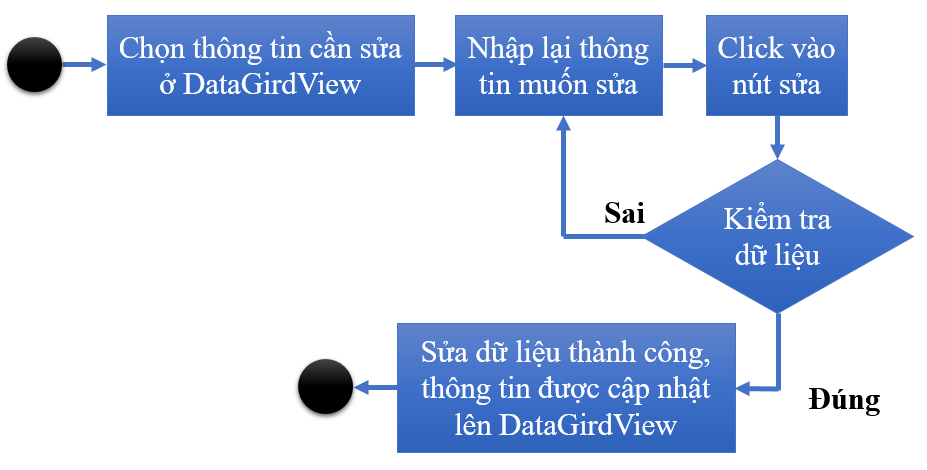
#### Lưu đồ quản lý

* Thêm thông tin sinh viên (Thêm khoa, lớp, môn học tương tự)



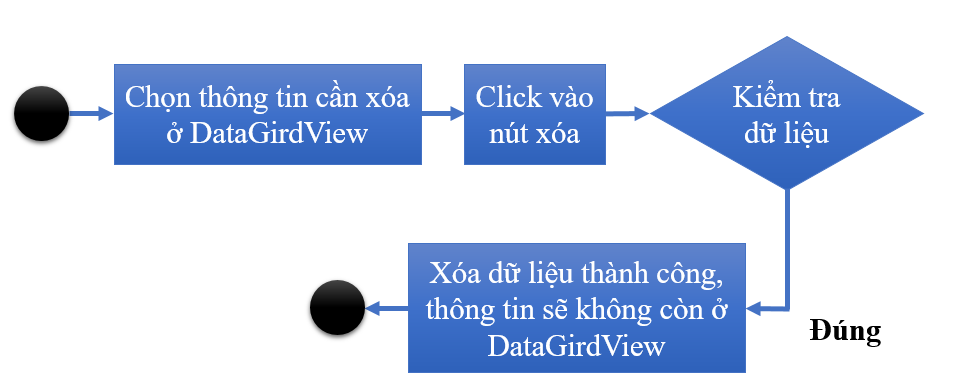
Hình 2. 3: Lưu đồ thêm thông tin sinh viên

* Sửa thông tin sinh viên (Sửa khoa, lớp, môn học tương tự)



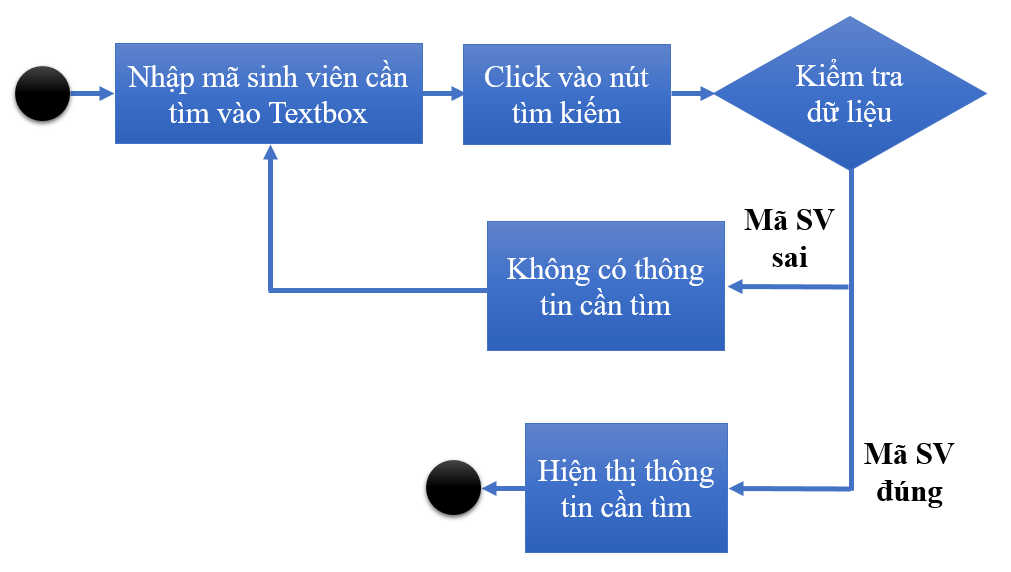
Hình 2. 4: Lưu đồ sửa thông tin sinh viên

* Xóa thông tin sinh viên (Xóa khoa, lớp, môn học tương tự)



Hình 2. 5: Xóa thông tin sinh viên

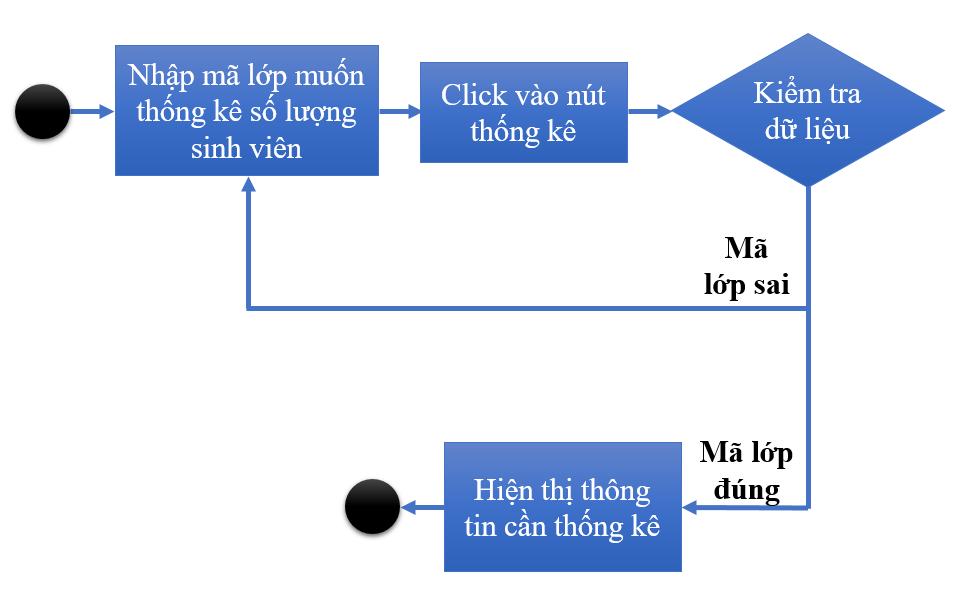
* Tìm kiếm thông tin sinh viên (Tìm kiếm khoa, lớp, môn học tương tự)



Hình 2. 6: Lưu đồ tìm kiếm thông tin sinh viên

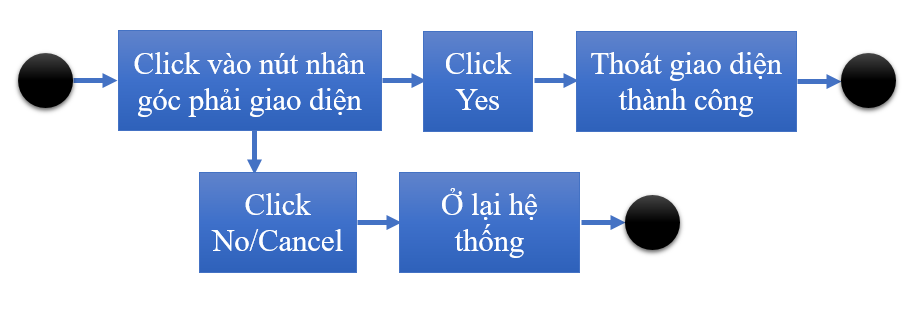
#### Lưu đồ báo cáo thống kê

Dưới dây là lưu đồ thống kê số lượng sinh viên theo lớp (Thống kê số lượng sinh viên theo khoa, địa chỉ phường/xã, quận/huyện, Tỉnh thành phố tương tự)



Hình 2. 7: Lưu đồ thống kê số lượng sinh viên theo mã lớp

#### Lưu đồ thoát khỏi hệ thống



Hình 2. 8: Lưu đồ thoát khỏi hệ thống

## Thiết kế mô hình cơ sở dữ liệu

### Các thực thể có trong hệ thống

Cơ sở dữ liệu quản lý thông tin sinh viên bao gồm các thực thể như sau:

* Tài khoản (TAIKHOAN)
* Khoa (KHOA)
* Lớp sinh hoạt (LOPSH)
* Sinh viên (SINHVIEN)
* Môn học (MONHOC)

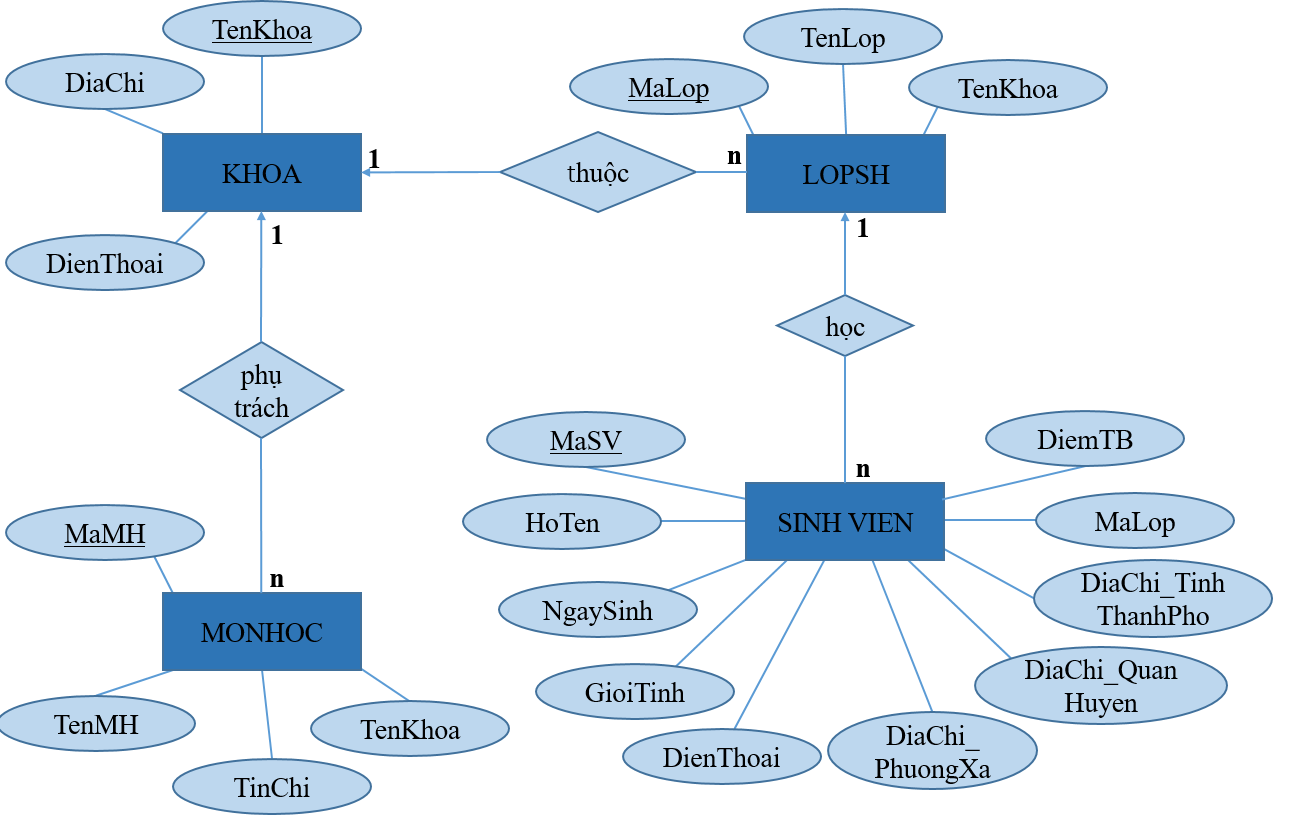
### Thuộc tính và khóa của các thực thể

Các thực thể của hệ thống được thiết kế như sau:

* KHOA gồm các thuộc tính: tên khoa, địa chỉ, điện thoại (**TenKhoa**, DiaChi, DienThoai)
* LOPSH gồm các thuộc tính: mã lớp, tên lớp, tên khoa (**MaLop**, TenLop, *TenKhoa*)
* SINHVIEN gồm các thuộc tính: mã sinh viên, họ tên, ngày sinh, giới tính, địa chỉ theo Phường/Xã, Quận/Huyên, Tỉnh thành, điện thoại, mã lớp (**MaSV**, HoTen, NgaySinh, GioiTinh, DiaChi\_PhuongXa, DiaChi\_QuanHuyen, DiaChi\_TinhThanhPho, DienThoai, *MaLop*)
* MON HOC gồm các thuộc tính: mã môn học, tên môn học, số tín chỉ, tên khoa (**MaMH**, TenMH, TinChi, *TenKhoa*)

Trong các bảng trên, khóa chính được **in đậm**, khóa ngoại được *in nghiên:* tạo khóa ngoại cho bảng LOPSH, tham chiếu tới bảng KHOA dựa trên cột *TenKhoa*; tạo khóa ngoại cho bảng SINHVIEN, tham chiếu tới bảng LOPSH dựa trên cột *MaLop*; tạo khóa ngoại cho bảng MONHOC, tham chiếu tới bảng KHOA dựa trên cột *TenKhoa*.

### Mô hình Thực thể - Mối quan hệ (E-R)



Hình 2. 9: Mô hình E-R

### Các lược đồ quan hệ

* SINHVIEN (**MaSV**, HoTen, NgaySinh, GioiTinh, DiaChi\_PhuongXa, DiaChi\_QuanHuyen, DiaChi\_TinhThanhPho, DienThoai, *MaLop*)

Mỗi sinh viên có một mã số sinh viên để phân biệt với các sinh viên khác, mã sinh viên không trùng lắp nhau. Mỗi sinh viên có một họ tên, một ngày sinh, một giới tính, một địa chỉ, một số điện thoại và thuộc một lớp duy nhất.

* KHOA (**TenKhoa**, DiaChi, DienThoai)

Mỗi khoa có một tên khoa để phân biệt với các khoa khác, tên khoa không trùng lắp nhau. Mỗi khoa có địa chỉ và số điện thoại khoa.

* LOPSH (**MaLop**, TenLop, *TenKhoa*)

Mỗi lớp có một mã lớp để phân biệt với các lớp khác, tên lớp không trùng lắp nhau. Mỗi lớp sinh hoạt sẽ thuộc một khoa.

* MONHOC (**MaMH**, TenMH, TinChi, *TenKhoa*)

Mỗi môn học có một mã môn học để phân biệt với các môn học khác, tên môn học không trùng lắp nhau. Mỗi môn học do mỗi khoa phụ trách và có số tín chỉ quy định.

### Mô tả thuộc tính các quan hệ

* SINHVIEN (**MaSV**, HoTen, NgaySinh, GioiTinh, DiaChi\_PhuongXa, DiaChi\_QuanHuyen, DiaChi\_TinhThanhPho, DienThoai, *MaLop*)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Kích thước | Diễn giải và Ràng buộc |
| 1 | MaSV | char | 12 | Mã số sinh viên, **Primary key** |
| 2 | HoTen | nvarchar | 50 | Họ và tên |
| 3 | NgaySinh | date |  | Ngày sinh |
| 4 | GioiTinh | nvarchar | 10 | Giới Tính |
| 5 | DiaChi\_PhuongXa | nvarchar | 50 | Địa chỉ Phường/xã |
| 6 | DiaChi\_QuanHuyen | nvarchar | 50 | Địa chỉ Quận/huyện |
| 7 | DiaChi\_TinhThanhPho | nvarchar | 50 | Địa chỉ Tỉnh thành phố |
| 8 | DienThoai | char | 10 | Số điện thoại |
| 9 | MaLop | char | 7 | Mã lớp sinh hoạt |

Bảng 2. 1: Bảng SINHVIEN

* KHOA (**TenKhoa**, DiaChi, DienThoai)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Kích thước | Diễn giải và Ràng buộc |
| 1 | TenKhoa | nvarchar | 50 | Tên khoa, **Primary key** |
| 2 | DiaChi | nvarchar | 150 | Địa chỉ khoa |
| 3 | DienThoai | char | 11 | Số điện thoại khoa |

Bảng 2. 2: Bảng KHOA

* LOPSH (**MaLop**, TenLop, *TenKhoa*)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Kích thước | Diễn giải và Ràng buộc |
| 1 | MaLop | Char | 7 | Mã lớp, **Primary key** |
| 2 | TenLop | nvarchar | 50 | Tên lớp sinh hoạt |
| 3 | TenKhoa | nvarchar | 50 | Tên khoa |

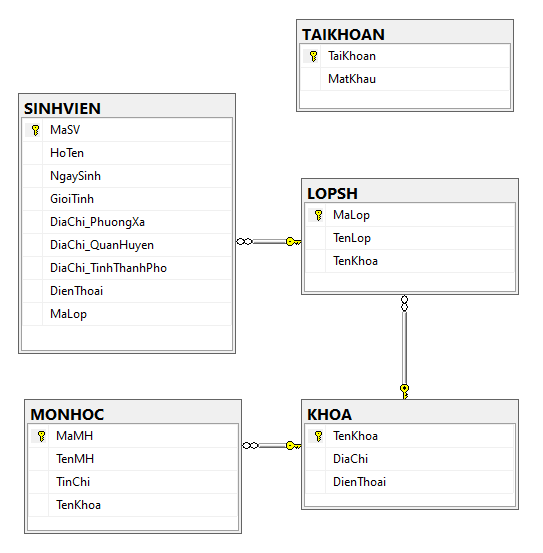
Bảng 2. 3: Bảng LOPSH

* MONHOC (MaMH, TenMH, TinChi, TenKhoa)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Thuộc tính | Kiểu dữ liệu | Kích thước | Diễn giải và Ràng buộc |
| 1 | MaMH | char | 7 | Mã môn học,  **Primary key** |
| 2 | TenMH | nvarchar | 100 | Tên môn học |
| 3 | TinChi | int |  | Số tín chỉ |
| 4 | TenKhoa | nvarchar | 50 | Tên khoa phụ trách |

Bảng 2. 4: Bảng MONHOC

### Mô hình CSDL (Database Diagram)



Hình 2. 10: Sơ đồ CSDL

# TRIỂN KHAI HỆ THỐNG

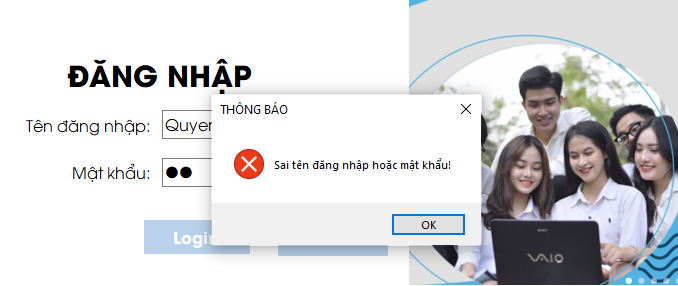
## Form đăng nhập hệ thống

Form đăng nhập của hệ thống:



Hình 3. 1: Form đăng nhập hệ thống

NQL sử dụng tên đăng và mật khẩu được cấp để đăng nhập khi muốn sử dụng các chức năng có trong hệ thống. Nếu nhập tài khoản đúng thì đi đến giao diện chính của hệ thống. Nếu sai hệ thống sẽ thông báo “*Sai tên đăng nhập hoặc mật khẩu*”, trường hợp này có thể nhập lại cho chính xác hoặc nhấp chuột vào nút “*Exit*” để thoát khỏi hệ thống.



Hình 3. 2: Form đăng nhập sai

## Giao diện form chính của hệ thống

Sau khi đăng nhập thành công, hệ thống sẽ đưa đến một giao diện chính như hình bên dưới:

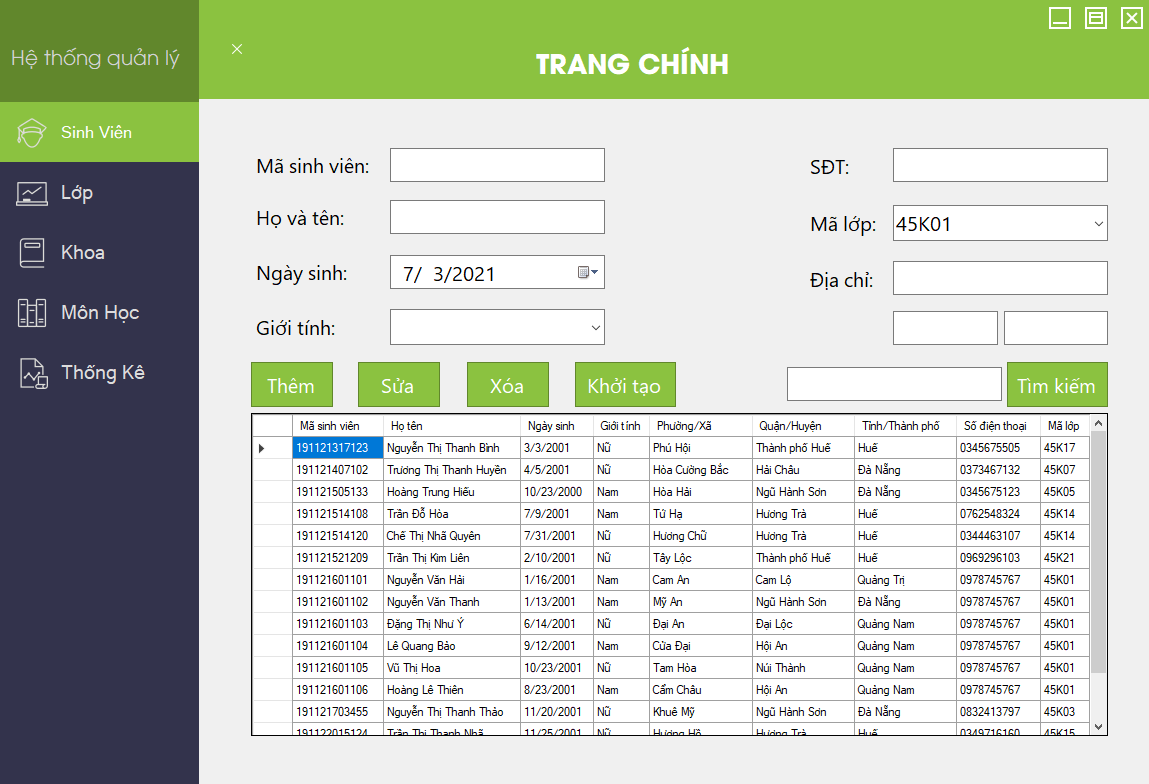


Hình 3. 3: Form giao diện chính

Tại đây, NQL sẽ lựa chọn các danh mục mà mình muốn thực hiện các chức năng trên form trong mỗi danh mục bằng cách bấm vào các button hiển thị trên form. Bao gồm danh mục quản lý sinh viên, quản lý khoa, quản lý lớp sinh hoạt, quản lý môn học. Khi NQL nhấp chuột vào các button, hệ thống sẽ tiến đến form đã được liên kết trước đó với tương ứng mỗi button. Ngoài các danh mục quản lý, hệ thống còn có thêm mục báo cáo thống kê số lượng sinh viên theo các dữ liệu khác nhau.

### Quản lý sinh viên

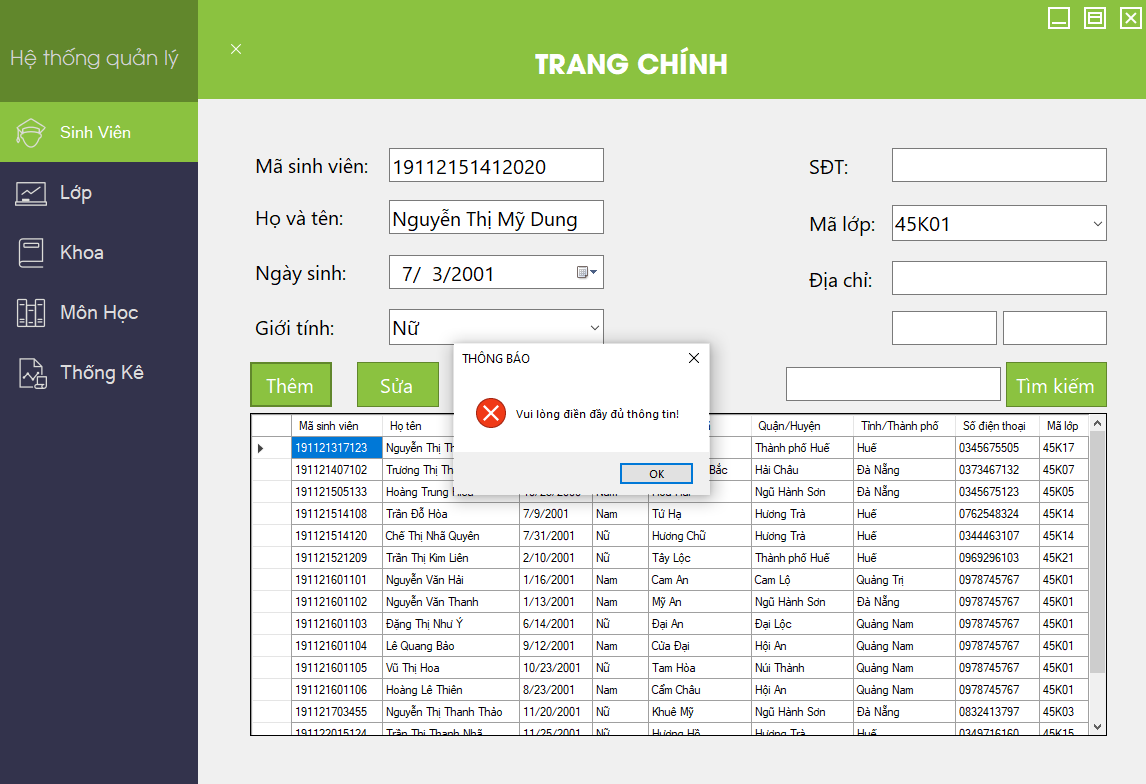
NQL nhấp chuột vào nút button **Sinh viên**, hệ thống sẽ xuất hiện các ô trống gồm Mã sinh viên, Họ và tên, Ngày sinh, Giới tính, Phường/Xã, Quận/Huyện, Tỉnh/Thành phố, Số điện thoại và Mã lớp để hỗ trợ cho việc thực hiện các chức năng thêm, sửa, xóa, khởi tạo và tìm kiếm cùng với bảng DataGirdView bên dưới để hiển thị thông tin.



Hình 3. 4: Form Sinh viên

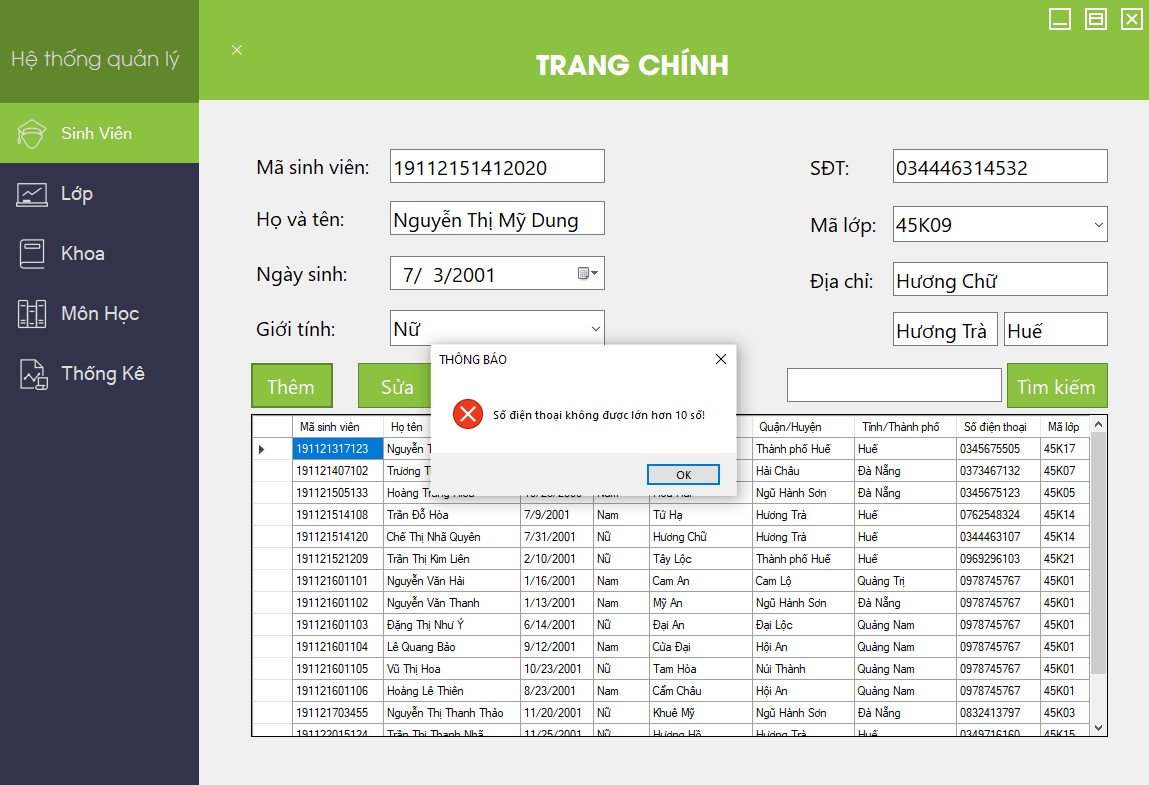
Khi muốn thêm sinh viên, NQL cần nhập thông tin sinh viên cần thêm vào các ô trống, sau đó nhấp chuột vào nút **Thêm**.

* Nếu không nhập đầy đủ thông tin ở các ô trống thì hệ thống sẽ thông báo lỗi “Vui lòng điền đầy đủ thông tin!”.



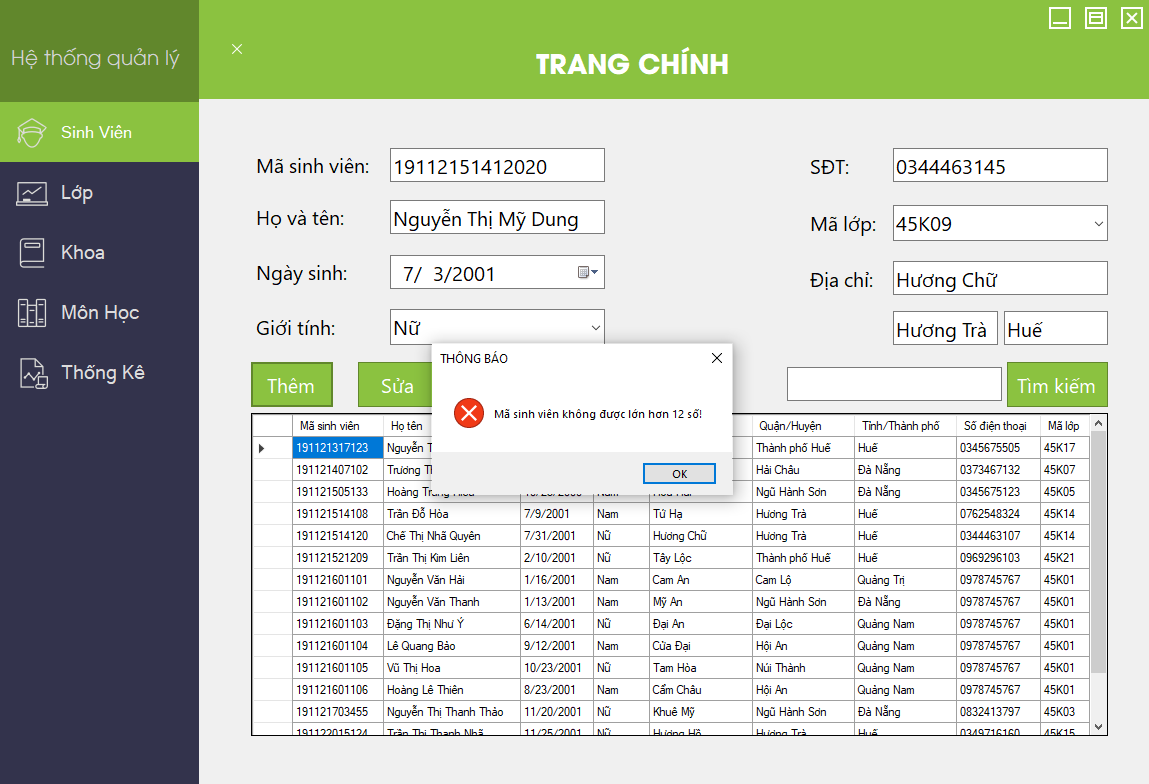
Hình 3. 5: Form báo lỗi không nhập đầy đủ thông tin

Nếu nhập thông tin số điện thoại không hợp lệ thì hệ thống sẽ báo lỗi “Số điện thoại không được lớn hơn 10 số!”

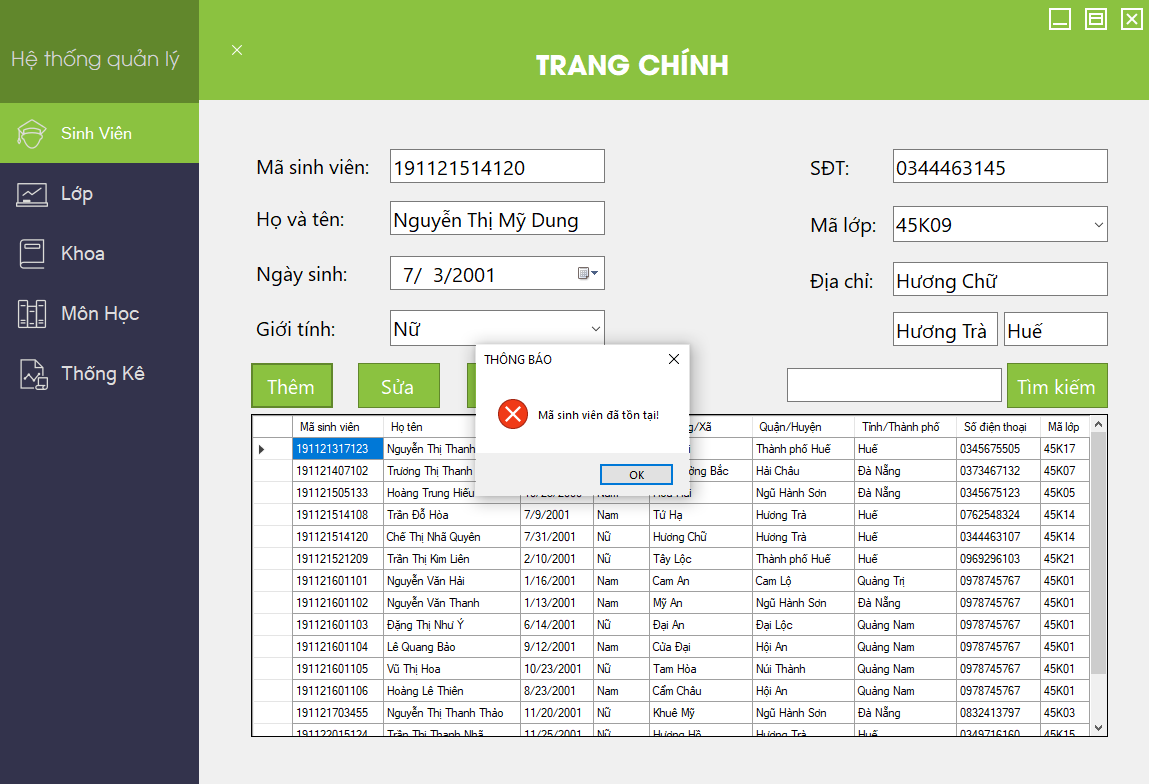


Hình 3. 6: Form báo lỗi nhập số điện thoại không hợp lệ

* Nếu nhập thông tin mã sinh viên không hợp lệ thì hệ thống sẽ báo lỗi “Mã sinh viên không được lớn hơn 12 số!” hoặc “Mã sinh viên đã tồn tại!”,

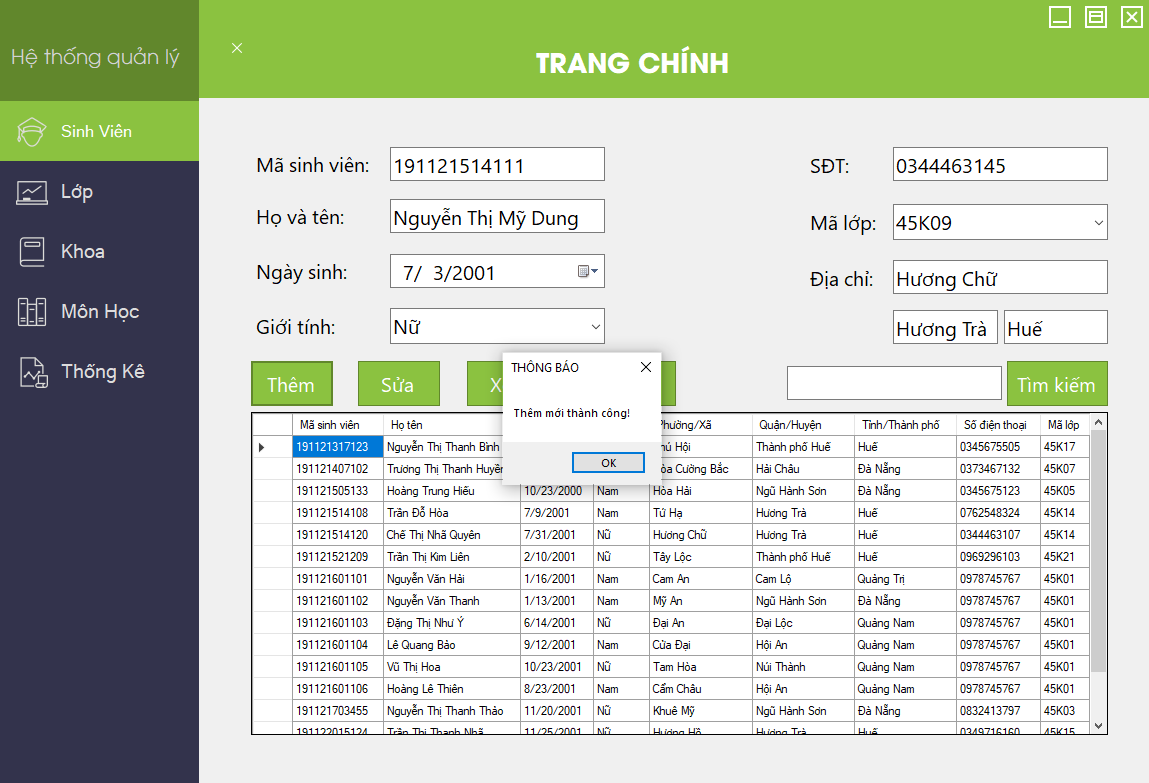


Hình 3. 7: Form báo lỗi nhập mã sinh viên không hợp lệ

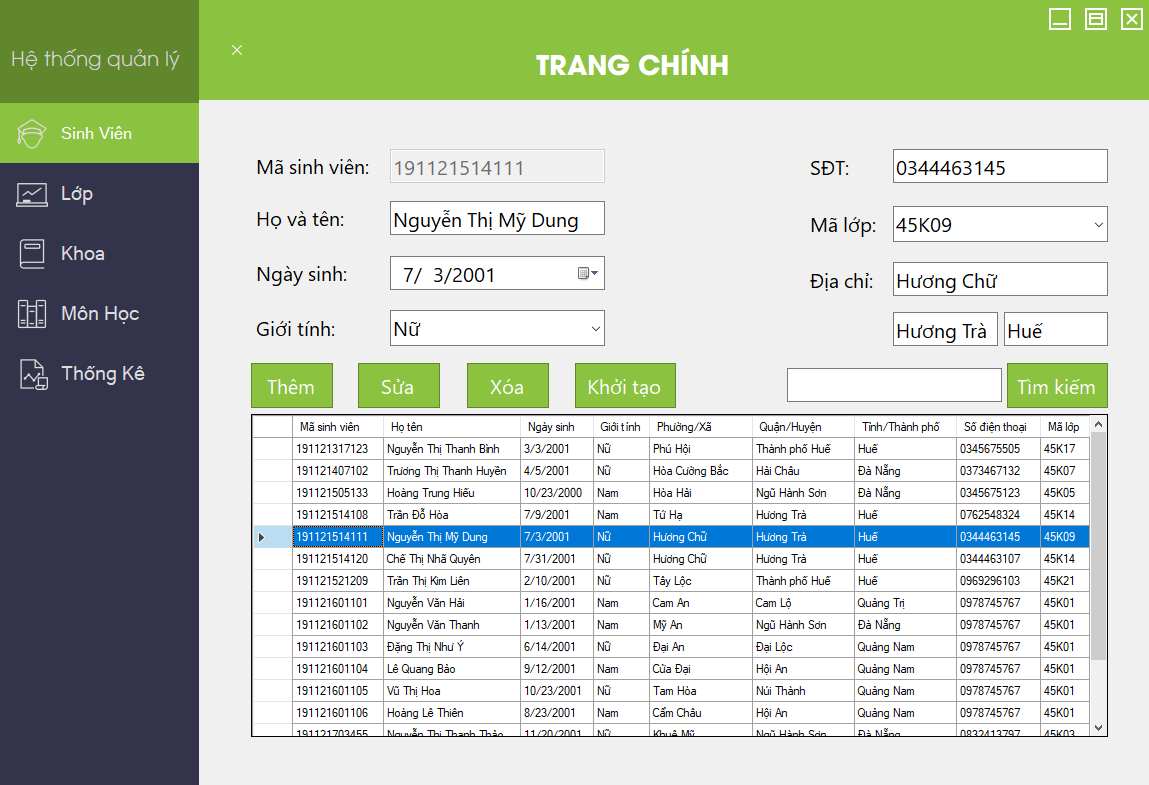


Hình 3. 8: Form báo lỗi mã sinh viên đã tồn tại

* Nếu nhập đầy đủ và hợp lệ thì hệ thống sẽ thông báo “Thêm mới thành công”. Nhấn “OK” thì sinh viên vừa thêm được cập nhật ở bảng DataGirdView.

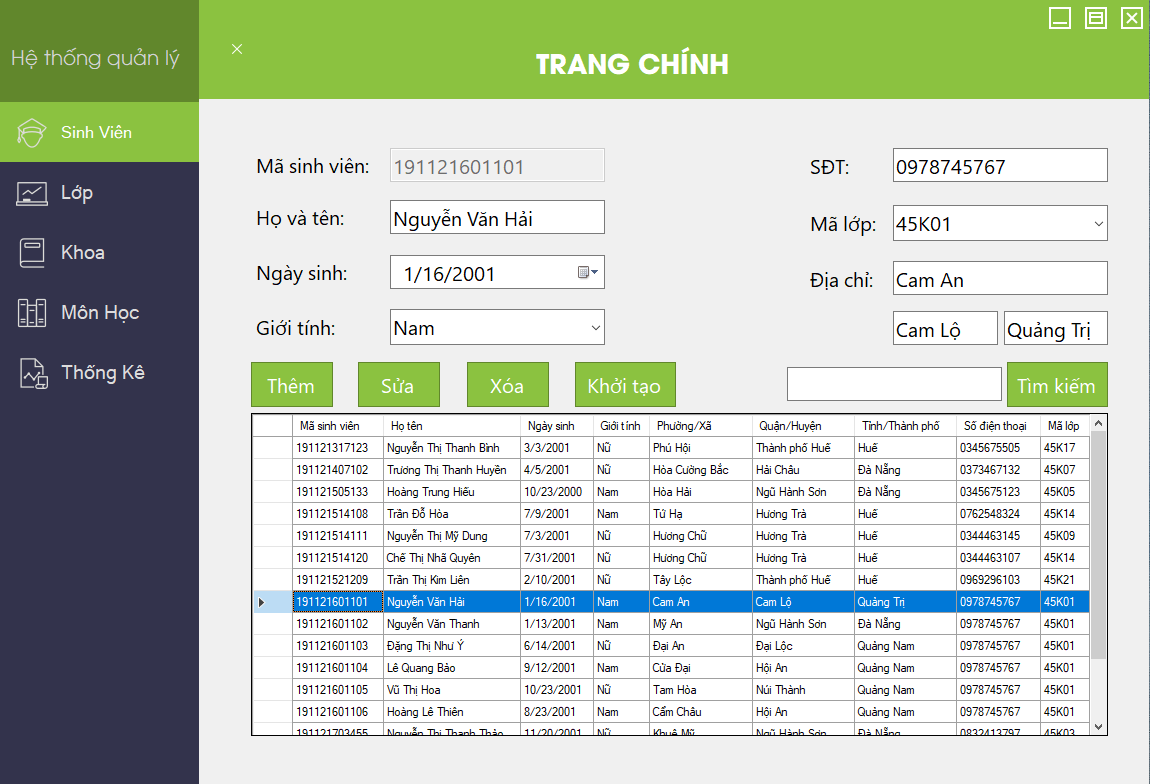


Hình 3. 9: Form thông báo thêm thành công



Hình 3. 10: Form hiển thị sinh viên vừa thêm mới

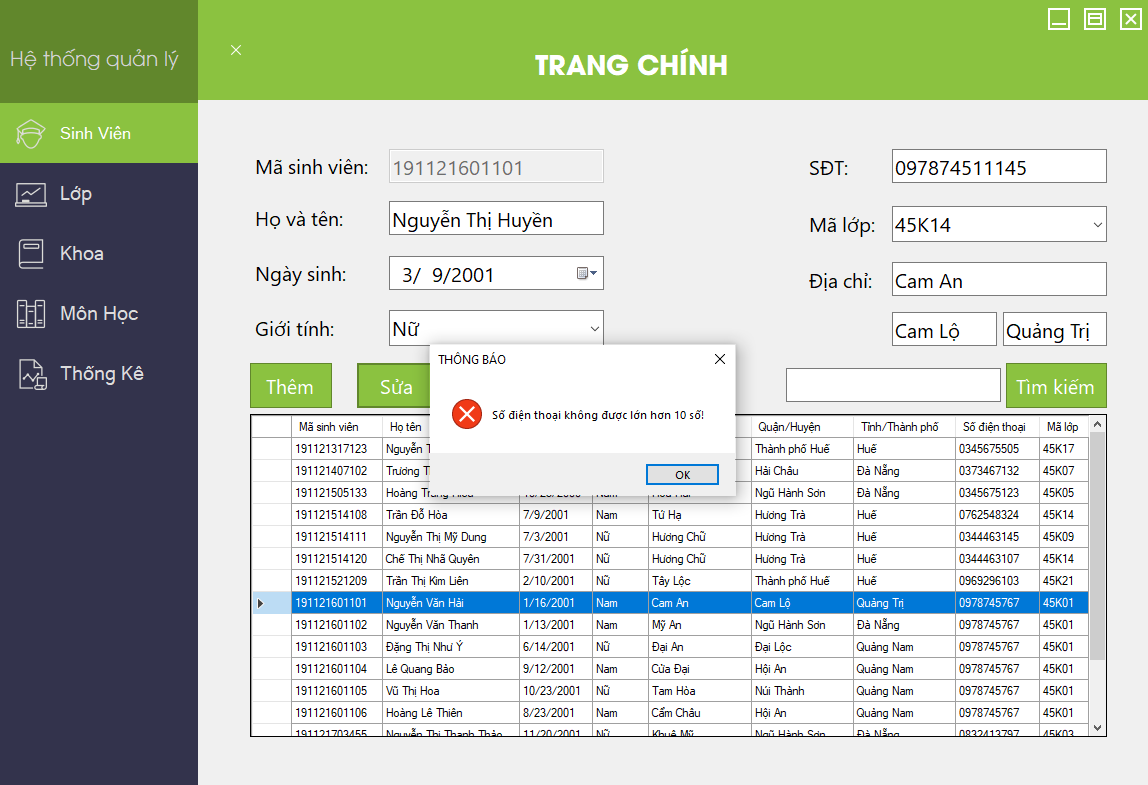
Khi muốn sửa thông tin sinh viên, NQL nhấp chọn sinh viên cần sửa ở bảng DataGirdView. Lúc này, thông tin của sinh viên sẽ hiển thị ở các ô trống.



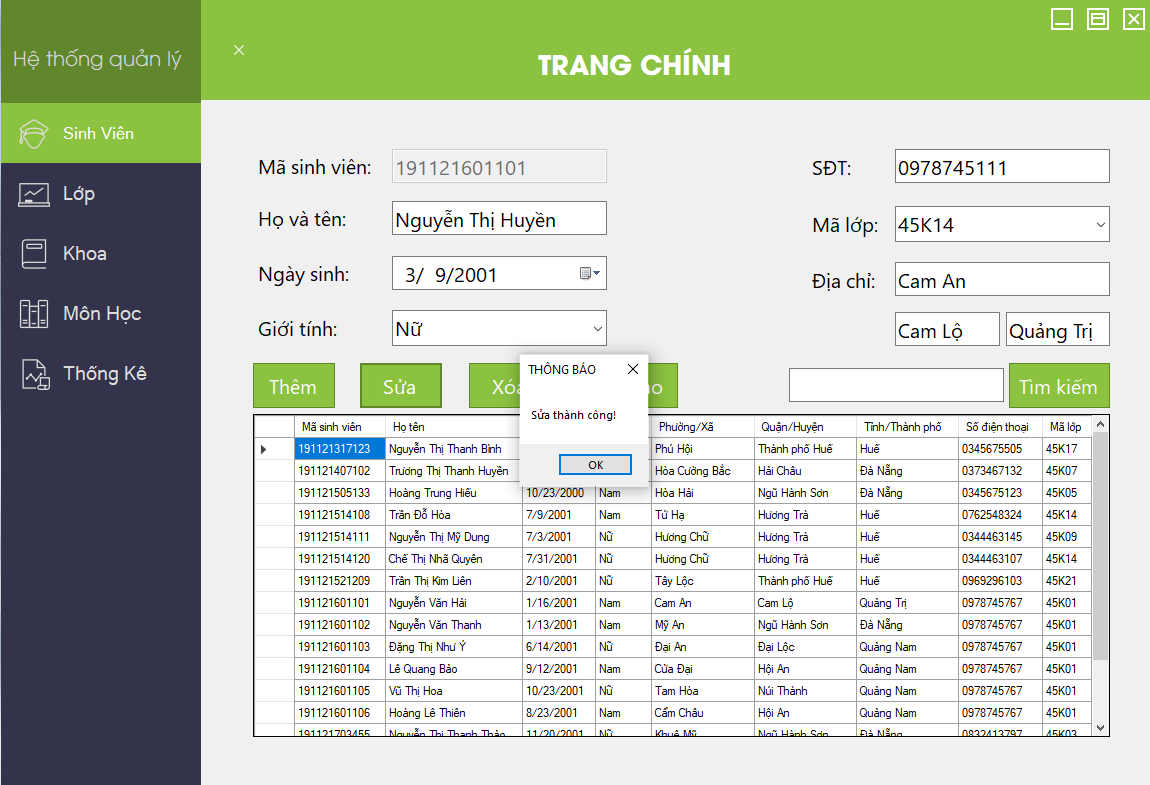
Hình 3. 11: Form chọn sinh viên muốn sửa

NQL tiến hành việc sửa thông tin bằng cách nhập lại thông tin vào các ô tương ứng rồi nhấp chuột vào Sửa.

* Trường hợp sửa thông tin số điện thoại không hợp lệ thì hệ thống sẽ báo lỗi “Số điện thoại không được lớn hơn 10 số!” khi đó bạn phải nhập đúng mới sửa được thông tin đó.
* Ngược lại, trường hợp nhập đầy đủ và hợp lệ hệ thống sẽ thông báo “Sửa thành công”. Thông tin sinh viên được sửa sẽ được tự động cập nhật ngay ở dưới bảng DataGirdView.

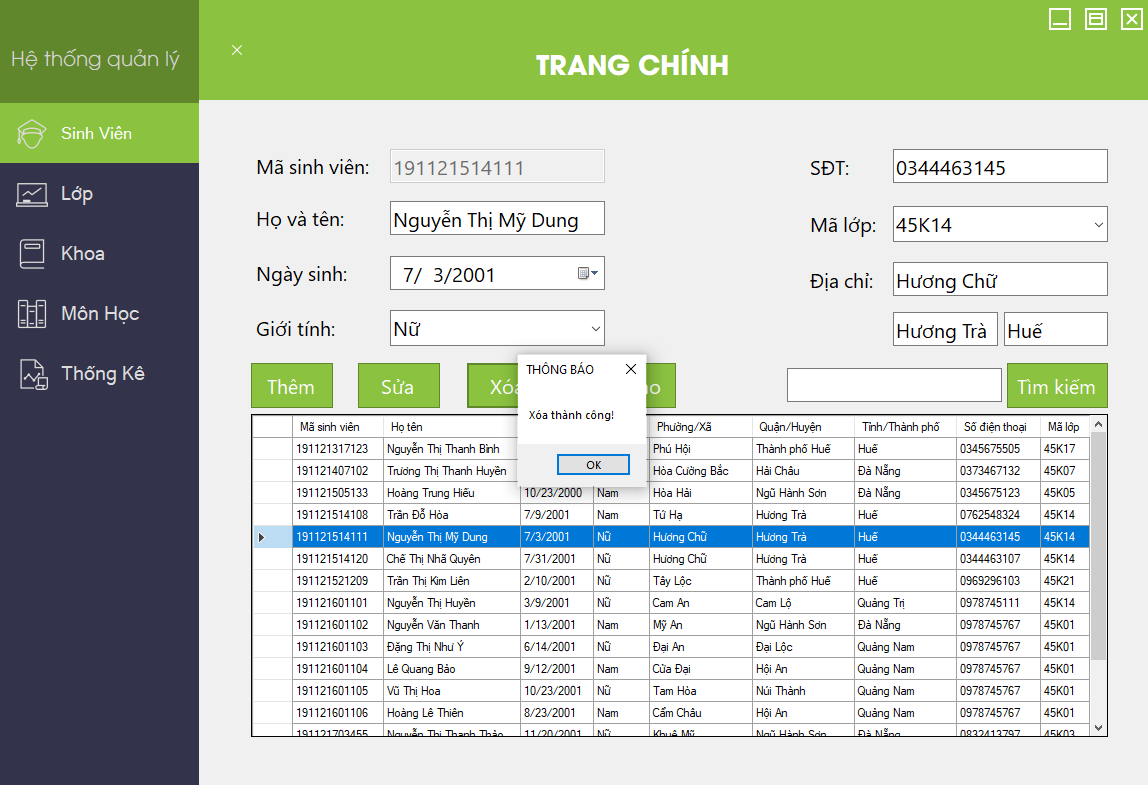


Hình 3. 12: Form báo lỗi sửa số điện thoại không hợp lệ



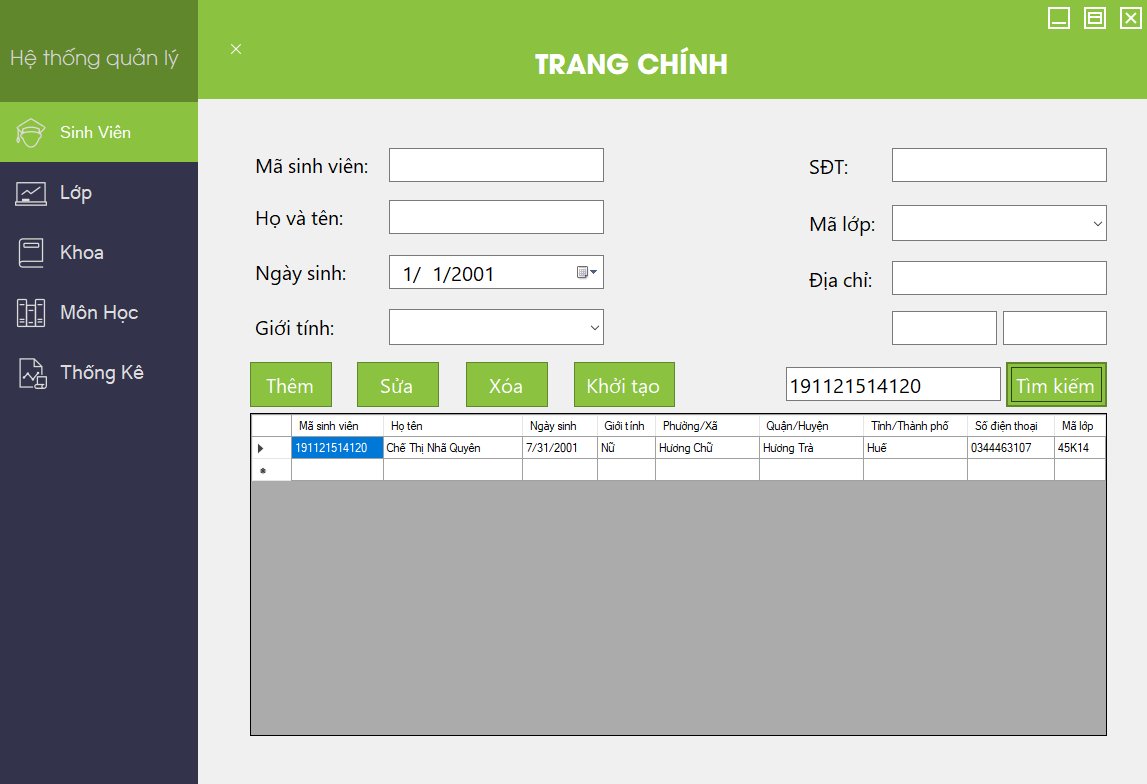
Hình 3. 13: Form thông báo sửa thành công

Khi sinh viên thôi học, cần phải xóa sinh viên đó, NQL nhấp chọn sinh viên cần xóa ở bảng DataGirdView rồi nhấp chuột vào nút **Xóa**. Lúc này hệ thống hiện lên thông báo “Xóa thành công”. Sinh viên đã được xóa sẽ không còn hiện ở bảng DataGirdView.



Hình 3. 14: Form thông báo xóa thành công

Khi muốn tìm kiếm một sinh viên nào đó, NQL sẽ nhập mã sinh viên vào ô trống, sau đó nhấp chuột vào nút **Tìm kiếm**.

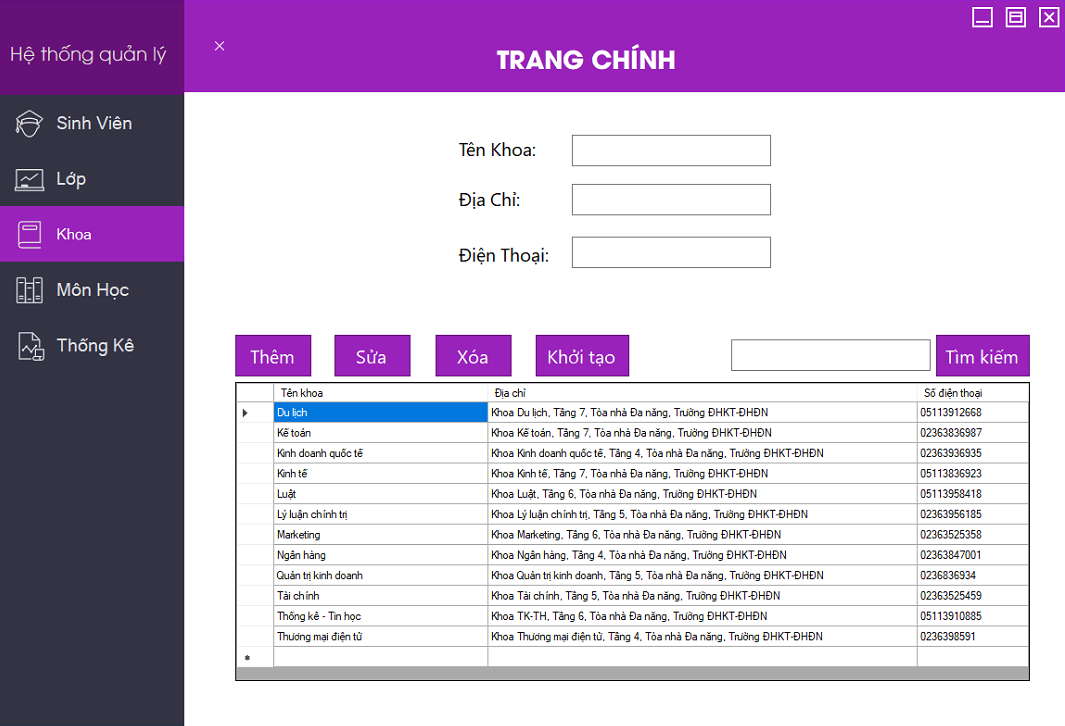


Hình 3. 15: Form tìm kiếm sinh viên theo mã sinh viên

Ở bảng DataGirdView sẽ hiển thị thông tin sinh viên cần tìm. Trường hợp nhập sai mã sinh viên, bảng sẽ không xuất hiện thông tin sinh viên nào.

### Quản lý khoa

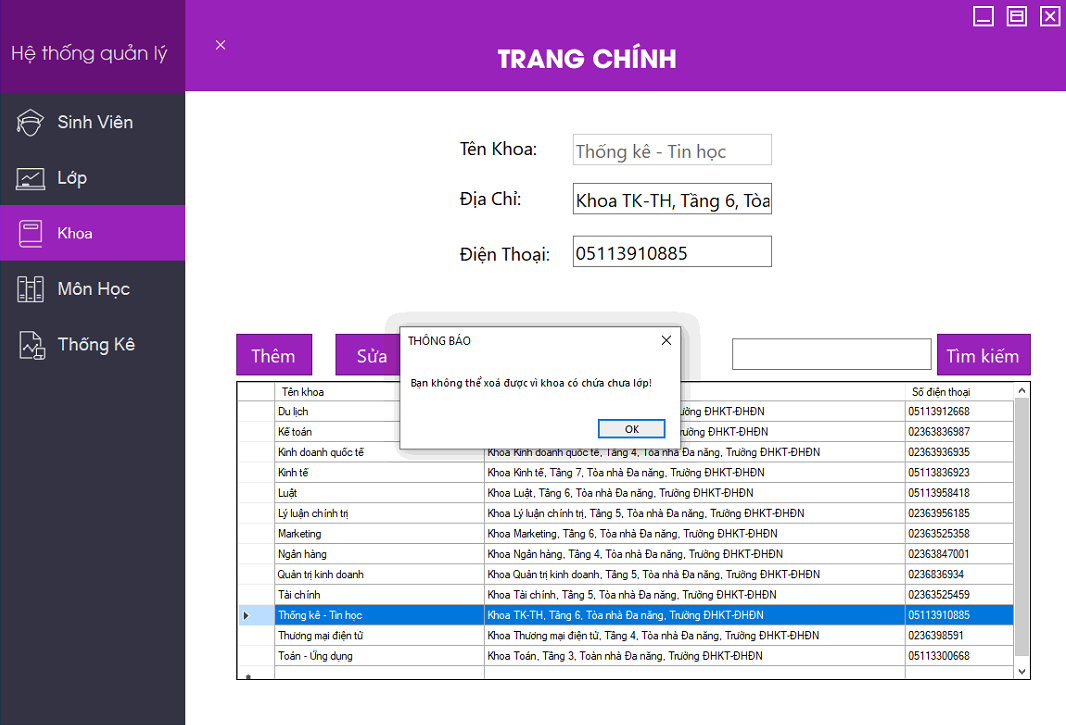
NQL nhấp chuột vào nút button **Khoa**, hệ thống sẽ xuất hiện các ô trống gồm Tên khoa, Số điện thoại khoa và Địa chỉ khoa. Tương tự như quản lý sinh viên, quản lý khoa cũng có các chức năng thêm, sửa, xóa, khởi tạo, tìm kiếm khoa và bảng DataGirdView bên dưới để hiển thị thông tin.



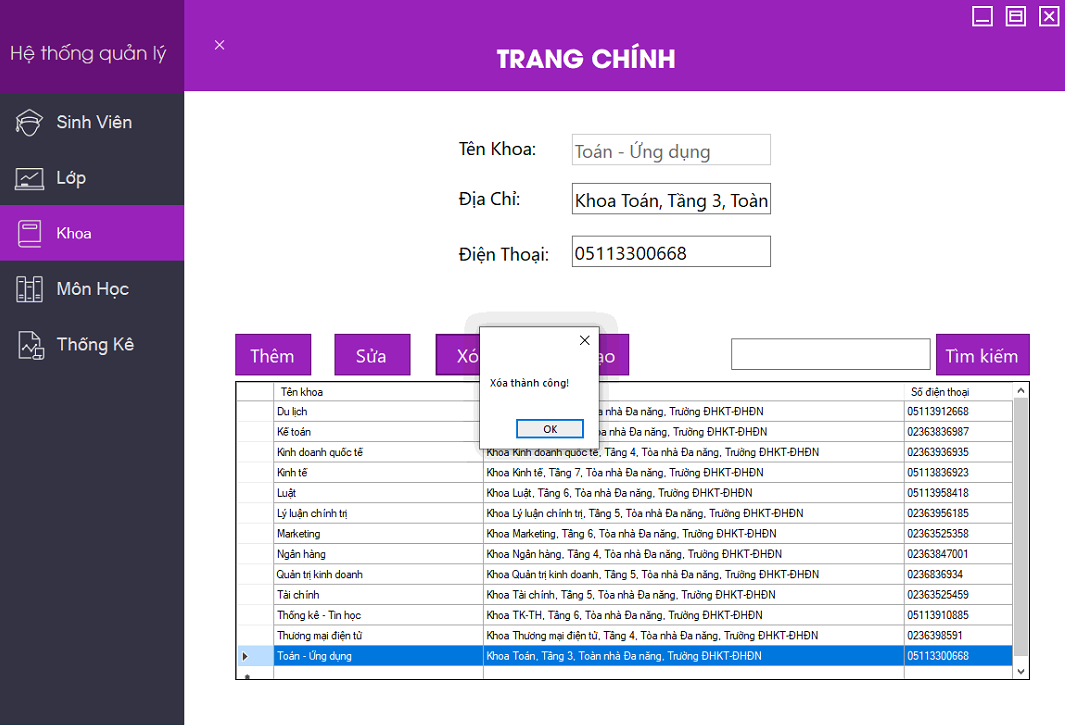
Hình 3. 16: Form Khoa

NQL có thể tìm kiếm số điện thoại hay địa chỉ của một khoa nào đó theo tên khoa. Cũng có thể sửa số điện thoại hoặc địa chỉ nếu khoa đó thay đổi số hoặc chuyển vị trí văn phòng khoa. Và cũng có thể thêm một khoa mới, cũng như xóa một khoa nào đó. Lưu ý đối với trường hợp xóa:

* Nếu khoa có chứa lớp, hệ thống sẽ thông báo lỗi “Không thể xóa vì khoa có chứa lớp!”
* Nếu khoa không chứa lớp, hệ thống sẽ thông báo “Xóa thành công”



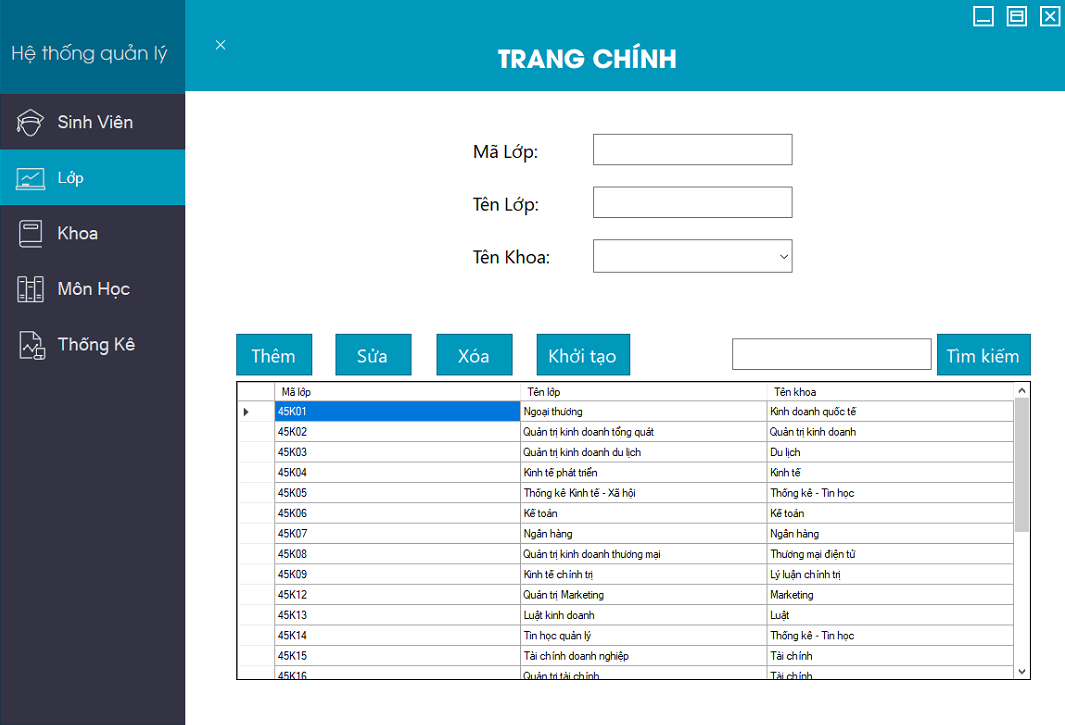
Hình 3. 17: Form báo lỗi không thể xóa



Hình 3. 18: Form thông báo xóa thành công

### Quản lý lớp sinh hoạt

Việc quản lý lớp sinh hoạt cũng tương tự như quản lý sinh viên và khoa. Quản lý lớp sinh hoạt cũng có đầy đủ các chức năng thêm, sửa, xóa, khởi tạo và tìm kiếm thông tin.



Hình 3. 19: Form Lớp sinh hoạt

Lưu ý đối với trường hợp xóa:

* Nếu lớp có chứa sinh viên, hệ thống sẽ thông báo lỗi “Không thể xóa vì lớp có chứa sinh viên!”
* Nếu lớp không chứa sinh viên, hệ thống sẽ thông báo “Xóa thành công”

### Quản lý môn học

Tương tự việc quản lý sinh viên, khoa và lớp sinh hoạt. Quản lý môn học cũng có đầy đủ các chức năng thêm, sửa, xóa, khởi tạo và tìm kiếm thông tin.



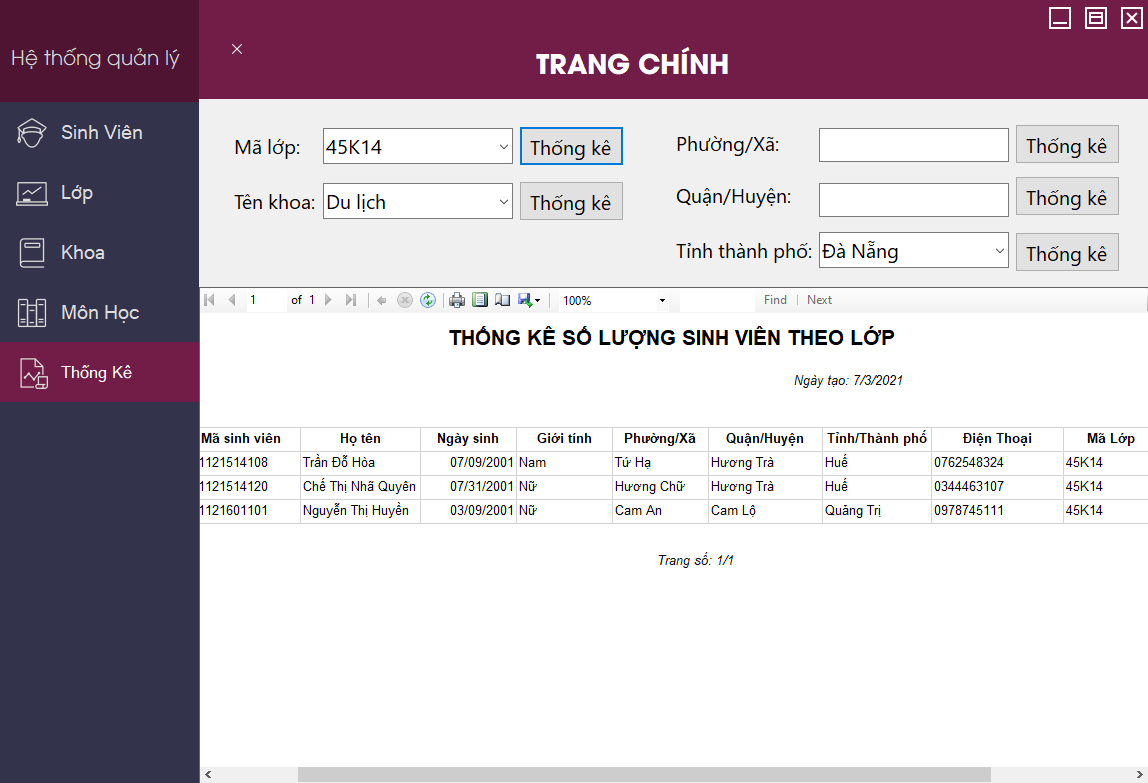
Hình 3. 20: Form Môn học

## Form thống kê số lượng sinh viên

Với số lượng sinh viên của một trường Đại học rất đông nên muốn biết có bao nhiêu sinh viên trong một lớp, khoa, … khá mất thời gian và không mang lại độ chính xác cao nếu làm thủ công. Vì vậy, hệ thống này có thể giúp cải thiện tiến độ công việc một các nhanh chóng và chính xác.

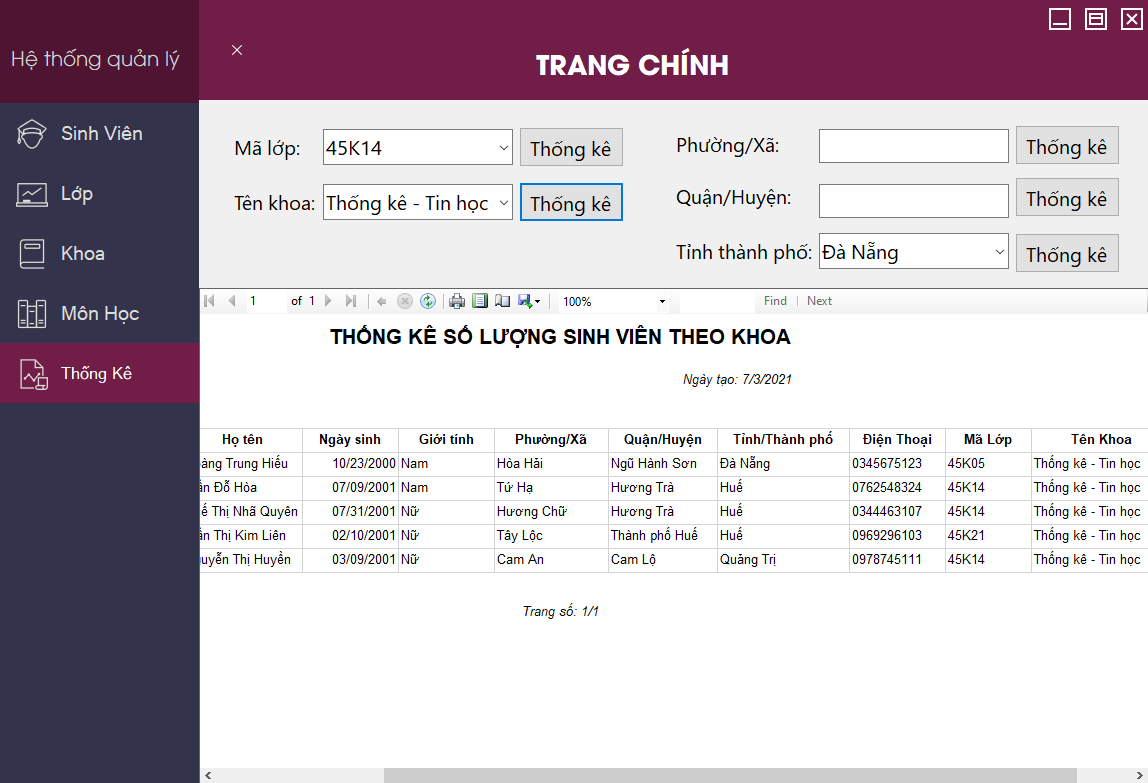
Hệ thống bao gồm thống kê theo lớp, theo khoa, theo từng địa phương (như Phường/Xã, Quận/Huyện, Tỉnh/Thành phố). Tùy vào các mục đích khác nhau mà NQL chọn loại thống kê tương ứng với mục đích đó.

Ví dụ, NQL muốn biết có bao nhiêu sinh viên đang theo học lớp 45K14 có bao nhiêu sinh viên thì NQL tiến hành chọn Mã lớp ở Combobox nút **Thống kê**, thông tin muốn thông kê được hiển thị ở bảng Report View.

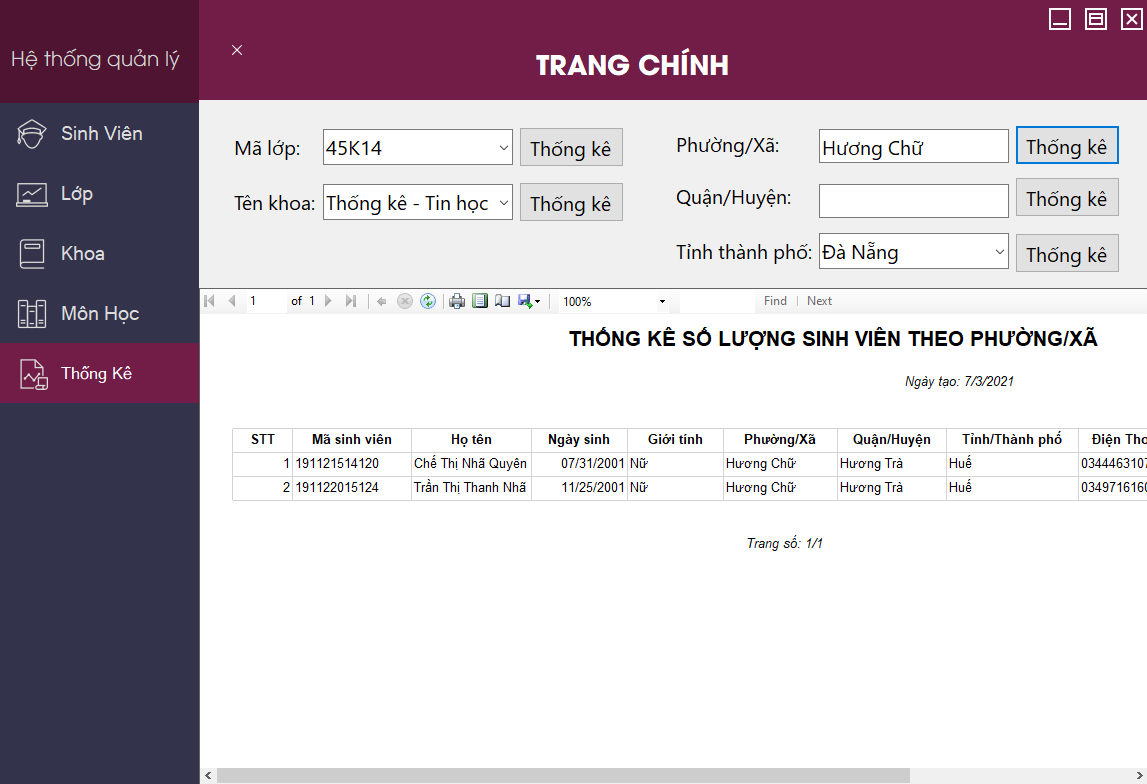


Hình 3. 21: Form thống kê theo lớp

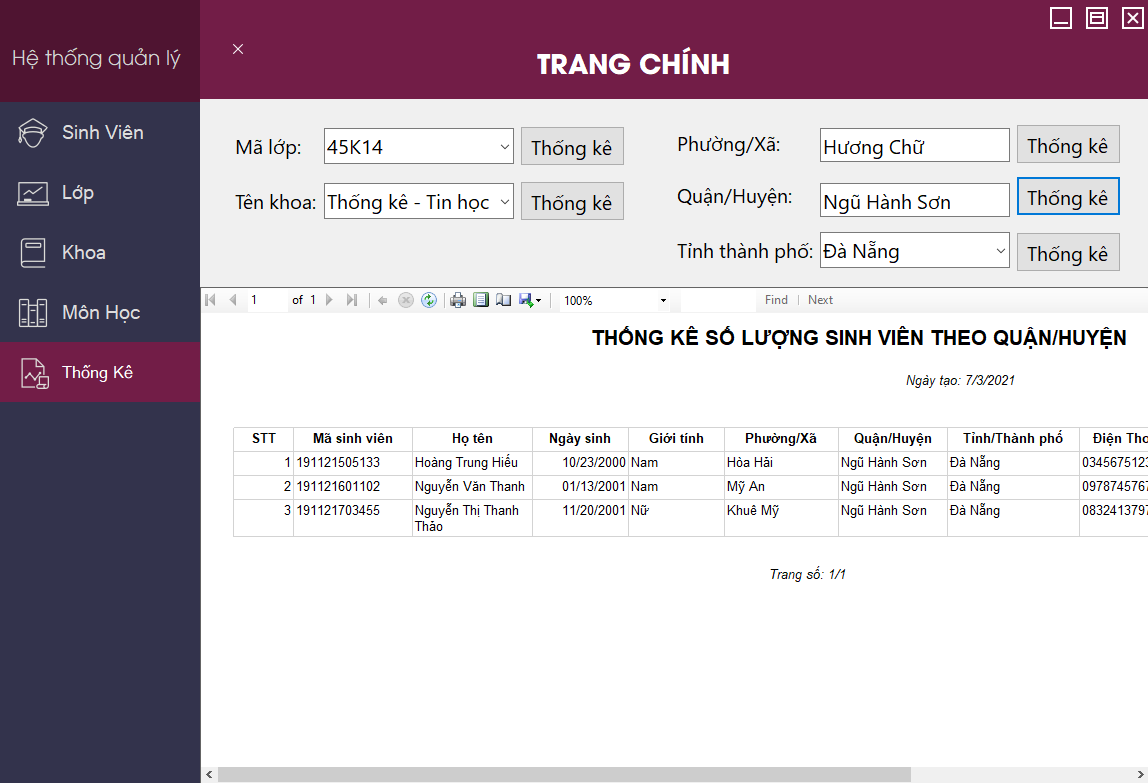
Tương tự đối với việc thống kê theo khoa, theo từng địa phương.



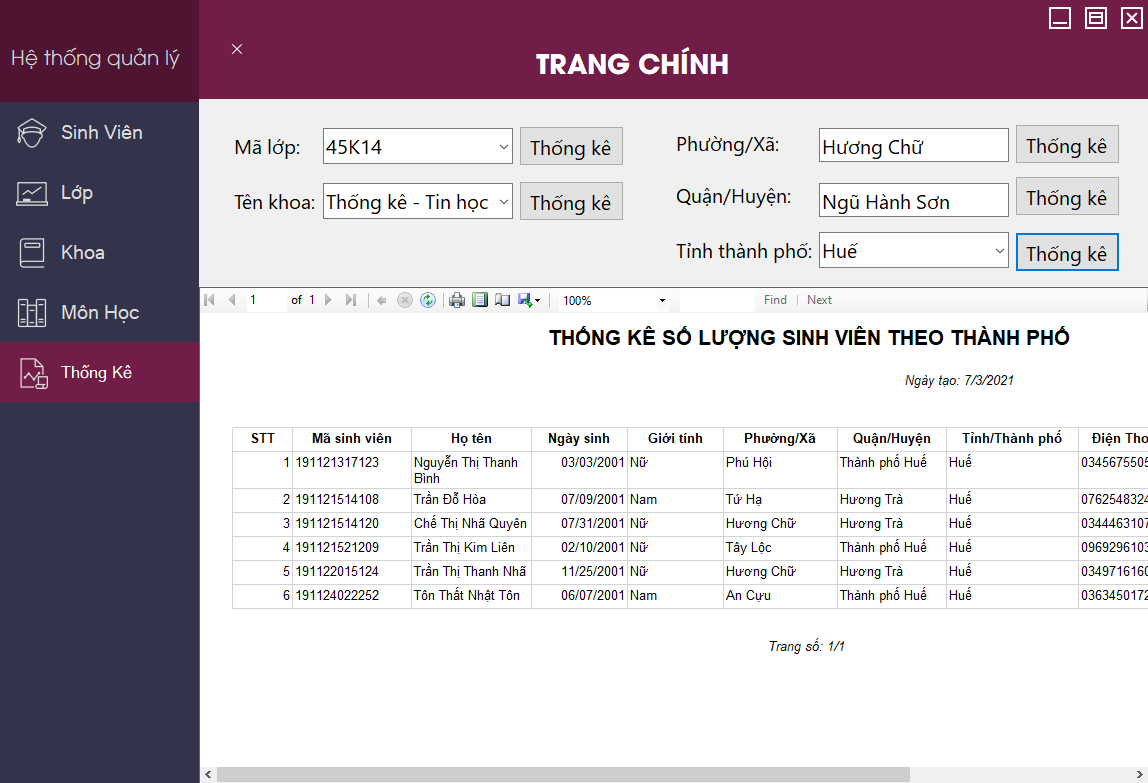
Hình 3. 22: Form thống kê theo khoa



Hình 3. 23: Form thống kê theo phường/xã



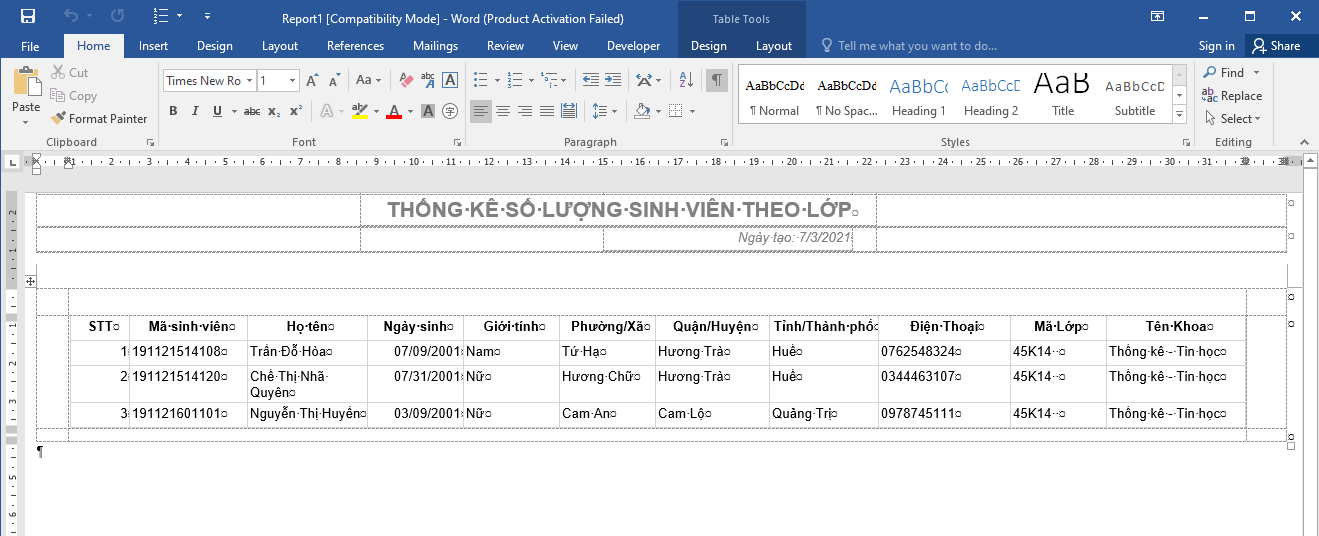
Hình 3. 24: Form thống kê theo quận/huyện



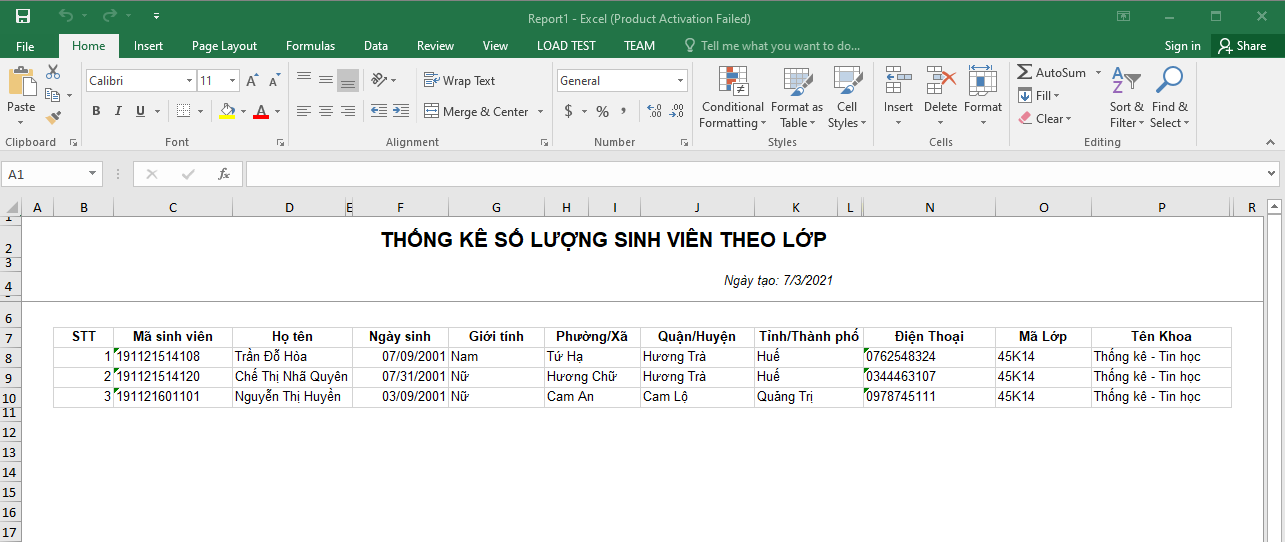
Hình 3. 25: Form thống kê theo tỉnh/thành phố

## Lưu bảng thống kê số lượng sinh viên

Việc lập danh sách cũng mất khá nhiều thời gian. Vì thế, để thuận tiện hơn sau khi thống kê NQL có thể lưu bảng thống kê đó thành file Word, Excel, PDF. Để lưu bảng, ta chỉ cần nhấp chuột vào “*biểu tượng Save*” ở bảng Report View và chọn hình thức mà mình muốn lưu.



Hình 3. 26: Lưu bảng thống kê thành file Word

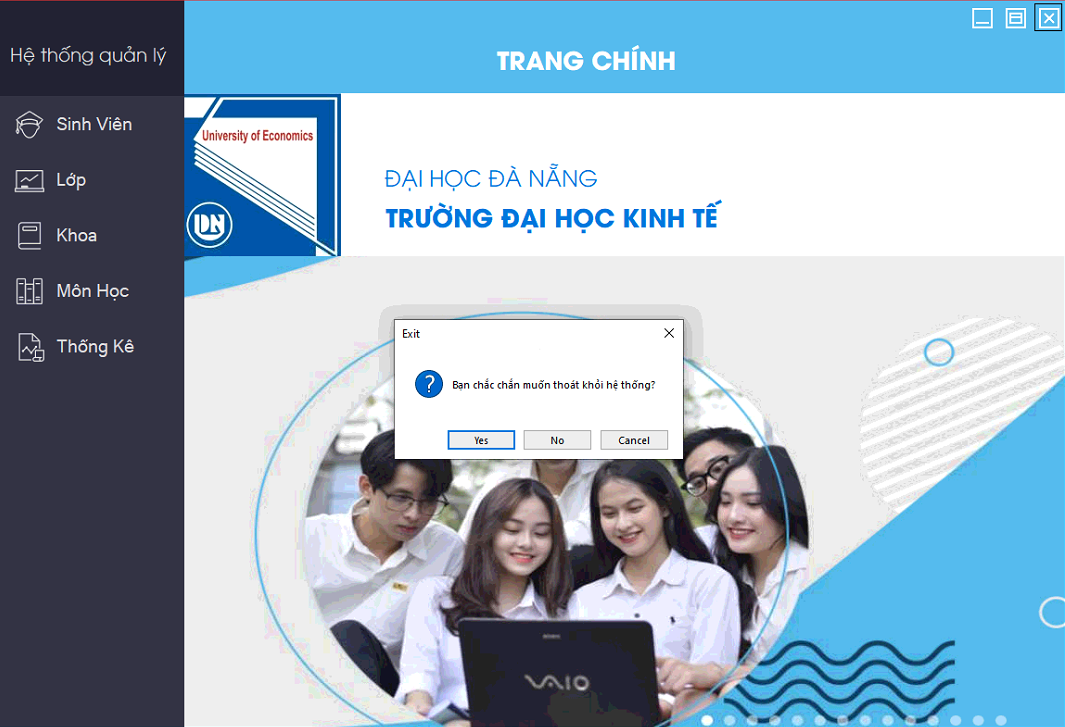


Hình 3. 27: Lưu bảng thống kê thành file Excel

## Thoát khỏi hệ thống

Nếu muốn thoát, nhấp chuột đến dấu nhân góc phải giao diện, hệ thống sẽ hiển thị thông báo “Bạn chắc chắn muốn thoát khỏi hệ thống?”.

* Chọn No/Cancel: ở lại giao diện.
* Chọn Yes: thoát hệ thống.



Hình 3. 28: Form thoát khỏi hệ thống

# KẾT QUẢ

## Kết quả đạt được

* Phân tích và thiết kế được hệ thống quản lý thông tin sinh viên với dữ liệu và các chức năng đơn giản.
* Vẽ được các lưu đồ công việc để mô tả hệ thống, mô hình Thực thể - Mối quan hệ (E-R), lược đồ quan hệ thiết kế được cơ sở dữ liệu ở mức khái niệm.
* Thiết kế và thực thi CSDL, bảng, ràng buộc dữ liệu, thiết lập được quan hệ giữa các bảng dữ liệu, các ràng buộc toàn vẹn dữ liệu trong CSDL (khóa chính, khóa ngoại) và nhập dữ liệu tương ứng. Hoàn thành được cơ sở dữ liệu đơn giản.
* Thiết kế được các StoredProcedure trong MS SQL Server.
* Thiết kế được các form và thực thi được code liên kết giữa các bảng để chạy thành công chương trình. Hoàn thiện hệ thống với các chức năng đơn giản.
* Biết tìm hiểu và sửa lỗi gặp phải trong quá trình xây dựng hệ thống.
* Hoàn thiện báo cáo theo yêu cầu nội dung đã đề ra.
* Rèn luyện kỹ năng làm việc nhóm, tương tác nhóm được cải thiện hơn.
* Tiếp xúc với các ứng dụng làm việc online như: Google meet, Teamviewer, Facebook, GitHub Desktop.

## Ưu điểm của hệ thống

* Hệ thống dễ sử dụng, thao tác đơn giản, có thể thực hiện các chức năng đơn giản.
* Là công cụ hỗ trợ đắc lực cho NQL, giúp tiết kiệm thời gian đáng kể.
* Thống kê được số lượng sinh viên tùy mục đích. Xuất được các thống kê ra file theo nhu cầu sử dụng 🡪 Lưu trữ và xử lý dữ liệu với độ chính xác cao.
* Người quản lý có thể sử dụng hệ thống để thao tác bất kỳ chức năng nào.

## Nhược điểm của hệ thống

* Vì khả năng còn hạn hẹp, hệ thống còn đơn giản và thiếu sót như độ bảo mật, tính thẩm mĩ, …
* Tính chuyên nghiệp chưa cao.

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Đề tài đã thực hiện được mục tiêu đề ra là “Yêu cầu một hệ thống quản lý thông tin sinh viên của trường Đại học Kinh tế - Đại học Đà nẵng nhằm hỗ trợ cho công tác quản lý sinh viên, khoa, lớp, môn học với các chức năng cơ bản: đọc, ghi, sửa, xóa, tìm kiếm thông tin và thống kê.” Tuy nhiên, để đưa vào sử dụng thực tiễn, hệ thống cần được bổ sung một số yêu cầu như:

* Nâng cao tính bảo mật cho hệ thống.
* Cải thiện tốc độ cũng như độ mượt của hệ thống.
* Phát triển thêm các chức năng mới phù hợp giúp cho việc quản lý.
* Cải thiện tính thẩm mĩ mang vẻ hiện đại hơn.

# TÀI LIỆU THAM KHẢO

Các trang web giáo viên hướng dẫn đăng tải về:

* Lập trình Winform cơ bản:

<https://www.howkteam.vn/course/lap-trinh-winform-co-ban-27?fbclid=IwAR2g-4nIsEUMEJTM3MBzA6MmQKKpSGvPhSS-qYBs46-3i_Kov1BI8MWLuaI>

* Lập trình CSDL Sql Server với Windows Form:

<https://www.youtube.com/playlist?list=PLMI-61XivrDr3W22JAyK-H7kfq0MueNUg>

Các video bài học trong kế hoạch thực tập:

* Giới thiệu tóm tắt cách thao tác với Github và Github Desktop:

<https://www.youtube.com/watch?v=QKPlvlFtP3I>

* Làm việc với Windows Form (mức cơ bản):

<https://www.youtube.com/watch?v=LZfcpH8e7HA>

* Cách tạo MDI form:

<https://www.youtube.com/watch?v=G1uVYIZVIcI>

* [Windows form] Làm việc với cơ sở dữ liệu:

<https://www.youtube.com/watch?v=ozDE7Ti2G80>

Trang web khác:

<https://www.youtube.com/watch?v=FRBZU9Dg4IA>

<https://www.youtube.com/watch?v=Q6dR_fpD-8w>

<https://timoday.edu.vn/phan-5-tao-bao-cao-voi-c/?fbclid=IwAR1-Fe8rchW7qZol-VCGkEiZAjdFNucIGl7jM7KxJZOW6N3IBAdDN1GyXOk>

# PHỤ LỤC

1. **Code kết nối dữ liệu**

private void button1\_Click(object sender, EventArgs e)

{

if (this.textBox1.TextLength == 0 || this.textBox2.TextLength == 0)

{

MessageBox.Show("Bạn vui lòng điền đầy đủ thông tin!", "THÔNG BÁO", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

else

{

SqlConnection con = new SqlConnection(sCon);

try

{

con.Open();

string tk = textBox1.Text;

string mk = textBox2.Text;

string sql = "select \*from TAIKHOAN where TaiKhoan ='" + tk + "' and MatKhau='" + mk + "'";

SqlCommand cmd = new SqlCommand(sql, con);

SqlDataReader dta = cmd.ExecuteReader();

if (dta.Read() == true)

{

MainMenu f = new MainMenu();

this.Hide();

f.ShowDialog();

this.Show();

}

else

{

MessageBox.Show("Sai tên đăng nhập hoặc mật khẩu!", "THÔNG BÁO", MessageBoxButtons.OK, MessageBoxIcon.Error);

textBox2.Text = "";

}

}

catch (Exception ex)

{

MessageBox.Show("Xảy ra lỗi trong quá trình kết nối DB");

}

}

}

private void button2\_Click(object sender, EventArgs e)

{

Application.Exit();

}

1. **Code form giao diện chính**

* **Code từ form giao diện để mở các form khác**

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

{

OpenChildForm(new Forms.SinhVien(), sender);

}

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

{

OpenChildForm(new Forms.Lop(), sender);

}

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

{

OpenChildForm(new Forms.Khoa(), sender);

}

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

{

OpenChildForm(new Forms.MonHoc(), sender);

}

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

{

OpenChildForm(new Forms.ThongKe(), sender);

}

* **Code thoát hệ thống**

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

{

if (MessageBox.Show("Bạn chắc chắn muốn thoát khỏi hệ thống?","Exit", MessageBoxButtons.YesNoCancel, MessageBoxIcon.Question) == DialogResult.Yes)

{

Close();

System.Windows.Forms.Application.Exit();

}

}

1. **Code đổ dữ liệu về DataGirdView ở form sinh viên**

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

{

SqlConnection con = new SqlConnection(sCon);

try

{

con.Open();

}

catch (Exception ex)

{

MessageBox.Show("Xảy ra lỗi trong quá trình kết nối DB!1");

}

string sQuery = "Select \* from SINHVIEN";

SqlDataAdapter adapter = new SqlDataAdapter(sQuery, con);

DataSet ds = new DataSet();

adapter.Fill(ds, "SinhVien");

con.Close();

}

1. **Code hiển thị dữ liệu ở DataGirdView lên các Textbox ở form sinh viên**

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

{

textBox1.Text = dataGridView1.Rows[e.RowIndex].Cells["MaSV"].Value.ToString();

textBox2.Text = dataGridView1.Rows[e.RowIndex].Cells["HoTen"].Value.ToString();

textBox3.Text = dataGridView1.Rows[e.RowIndex].Cells["DienThoai"].Value.ToString();

textBox4.Text = dataGridView1.Rows[e.RowIndex].Cells["DiaChi\_PhuongXa"].Value.ToString();

textBox5.Text = dataGridView1.Rows[e.RowIndex].Cells["DiaChi\_QuanHuyen"].Value.ToString();

textBox6.Text = dataGridView1.Rows[e.RowIndex].Cells["DiaChi\_TinhThanhPho"].Value.ToString();

comboBox3.Text = dataGridView1.Rows[e.RowIndex].Cells["MaLop"].Value.ToString();

comboBox1.Text = dataGridView1.Rows[e.RowIndex].Cells["GioiTinh"].Value.ToString();

dateTimePicker1.Value = Convert.ToDateTime(dataGridView1.Rows[e.RowIndex].Cells["NgaySinh"].Value);

textBox1.Enabled = false;

}

1. **Code thực hiện chức năng thêm ở form sinh viên**

private void button1\_Click(object sender, EventArgs e)

{

//Bước 1

if (this.textBox1.TextLength == 0 || this.textBox2.TextLength == 0 || this.textBox3.TextLength == 0 || this.textBox4.TextLength == 0 | this.textBox5.TextLength == 0

|| this.textBox6.TextLength == 0 || this.dateTimePicker1.Value == null || this.comboBox1.Text == null || this.comboBox3.Text == null

)

{

MessageBox.Show("Vui lòng điền đầy đủ thông tin!", "THÔNG BÁO", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

else

{

SqlConnection con = new SqlConnection(sCon);

try

{

con.Open();

}

catch (Exception ex)

{

MessageBox.Show("Xảy ra lỗi trong quá trình kết nối DB!");

}

//Bước 2

// chuan bi du lieu

// kiem tra tinh hop le

// gan du lieu vao bien

string sMaSV = textBox1.Text;

string sHoTen = textBox2.Text;

string sDienThoai = textBox3.Text;

string sDiaChi\_PhuongXa = textBox4.Text;

string sDiaChi\_QuanHuyen = textBox5.Text;

string sDiaChi\_TinhThanhPho = textBox6.Text;

string sGioiTinh = comboBox1.Text;

string sMaLop = comboBox3.Text;

string sNgaySinh = dateTimePicker1.Value.ToString("yyy-MM-dd");

string sQuery1 = " insert into SINHVIEN values(@MaSV,@HoTen,@NgaySinh,@GioiTinh,@DiaChi\_PhuongXa,@DiaChi\_QuanHuyen,@DiaChi\_TinhThanhPho,@DienThoai,@MaLop) ";

SqlCommand cmd = new SqlCommand(sQuery1, con);

cmd.Parameters.AddWithValue("@MaSV", sMaSV);

cmd.Parameters.AddWithValue("@HoTen", sHoTen);

cmd.Parameters.AddWithValue("@NgaySinh", sNgaySinh);

cmd.Parameters.AddWithValue("@GioiTinh", sGioiTinh);

cmd.Parameters.AddWithValue("@DiaChi\_PhuongXa", sDiaChi\_PhuongXa);

cmd.Parameters.AddWithValue("@DiaChi\_QuanHuyen", sDiaChi\_QuanHuyen);

cmd.Parameters.AddWithValue("@DiaChi\_TinhThanhPho", sDiaChi\_TinhThanhPho);

cmd.Parameters.AddWithValue("@DienThoai", sDienThoai);

cmd.Parameters.AddWithValue("@MaLop", sMaLop);

try

{

cmd.ExecuteNonQuery();

MessageBox.Show("Thêm mới thành công!", "THÔNG BÁO", MessageBoxButtons.OK);

}

catch (Exception ex)

{

if (textBox3.TextLength > 10)

MessageBox.Show("Số điện thoại không được lớn hơn 10 số!", "THÔNG BÁO", MessageBoxButtons.OK, MessageBoxIcon.Error);

else

{

if (textBox1.TextLength > 12)

MessageBox.Show("Mã sinh viên không được lớn hơn 12 số!", "THÔNG BÁO", MessageBoxButtons.OK, MessageBoxIcon.Error);

else

MessageBox.Show("Mã sinh viên đã tồn tại!", "THÔNG BÁO", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

string sQuery = "Select \* from SINHVIEN";

SqlDataAdapter adapter = new SqlDataAdapter(sQuery, con);

DataSet ds = new DataSet();

adapter.Fill(ds, "SinhVien");

dataGridView1.DataSource = ds.Tables["SinhVien"];

con.Close();

}

}

1. **Code thực hiện chức năng sửa ở form sinh viên**

private void button2\_Click(object sender, EventArgs e)

{

SqlConnection con = new SqlConnection(sCon);

try

{

con.Open();

}

catch (Exception ex)

{

MessageBox.Show("Xảy ra lỗi trong quá trình kết nối DB3!");

}

string sMaSV = textBox1.Text;

string sHoTen = textBox2.Text;

string sDienThoai = textBox3.Text;

string sDiaChi\_PhuongXa = textBox4.Text;

string sDiaChi\_QuanHuyen = textBox5.Text;

string sDiaChi\_TinhThanhPho = textBox6.Text;

string sGioiTinh = comboBox1.Text;

string sMaLop = comboBox3.Text;

string sNgaySinh = dateTimePicker1.Value.ToString("yyy-MM-dd");

string sQuery2 = "Update SINHVIEN Set HoTen = @HoTen, NgaySinh = @NgaySinh, GioiTinh = @GioiTinh, DiaChi\_PhuongXa = @DiaChi\_PhuongXa, DiaChi\_QuanHuyen = @DiaChi\_QuanHuyen,DiaChi\_TinhThanhPho=@DiaChi\_TinhThanhPho, DienThoai=@DienThoai, MaLop=@MaLop Where MaSV = @MaSV";

SqlCommand cmd = new SqlCommand(sQuery2, con);

cmd.Parameters.AddWithValue("@MaSV", sMaSV);

cmd.Parameters.AddWithValue("@HoTen", sHoTen);

cmd.Parameters.AddWithValue("@NgaySinh", sNgaySinh);

cmd.Parameters.AddWithValue("@GioiTinh", sGioiTinh);

cmd.Parameters.AddWithValue("@DiaChi\_PhuongXa", sDiaChi\_PhuongXa);

cmd.Parameters.AddWithValue("@DiaChi\_QuanHuyen", sDiaChi\_QuanHuyen);

cmd.Parameters.AddWithValue("@DiaChi\_TinhThanhPho", sDiaChi\_TinhThanhPho);

cmd.Parameters.AddWithValue("@DienThoai", sDienThoai);

cmd.Parameters.AddWithValue("@MaLop", sMaLop);

try

{

cmd.ExecuteNonQuery();

MessageBox.Show("Sửa thành công!", "THÔNG BÁO", MessageBoxButtons.OK);

}

catch (Exception ex)

{

if (textBox3.TextLength > 10)

MessageBox.Show("Số điện thoại không được lớn hơn 10 số!", "THÔNG BÁO", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

string sQuery = "Select \* from SINHVIEN";

SqlDataAdapter adapter = new SqlDataAdapter(sQuery, con);

DataSet ds = new DataSet();

adapter.Fill(ds, "SinhVien");

dataGridView1.DataSource = ds.Tables["SinhVien"];

con.Close();

}

1. **Code thực hiện chức năng xóa ở form sinh viên**

private void button3\_Click(object sender, EventArgs e)

{

SqlConnection con = new SqlConnection(sCon);

try

{

con.Open();

}

catch (Exception ex)

{

MessageBox.Show("Xảy ra lỗi trong quá trình kết nối DB!");

}

string sMaSV = textBox1.Text;

string sQuery2 = "delete SINHVIEN where MaSV=@MaSV";

SqlCommand cmd = new SqlCommand(sQuery2, con);

cmd.Parameters.AddWithValue("@MaSV", sMaSV);

try

{

cmd.ExecuteNonQuery();

MessageBox.Show("Xóa thành công!", "THÔNG BÁO", MessageBoxButtons.OK);

}

catch (Exception ex)

{

MessageBox.Show("Xóa thất bại!", "THÔNG BÁO", MessageBoxButtons.OK,MessageBoxIcon.Error);

}

string sQuery = "Select \* from SINHVIEN";

SqlDataAdapter adapter = new SqlDataAdapter(sQuery, con);

DataSet ds = new DataSet();

adapter.Fill(ds, "SinhVien");

dataGridView1.DataSource = ds.Tables["SinhVien"];

con.Close();

}

1. **Code chức năng khởi tạo ở form sinh viên**

private void button4\_Click(object sender, EventArgs e)

{

textBox1.Text = "";

textBox2.Text = "";

textBox3.Text = "";

textBox4.Text = "";

textBox5.Text = "";

textBox6.Text = "";

comboBox3.Text = "";

comboBox1.Text = "";

dateTimePicker1.Value = new DateTime(2001, 1, 1);

textBox1.Enabled = true;

SqlConnection con = new SqlConnection(sCon);

string sQuery = "Select \* from SINHVIEN";

SqlDataAdapter adapter = new SqlDataAdapter(sQuery, con);

DataSet ds = new DataSet();

adapter.Fill(ds, "SinhVien");

dataGridView1.DataSource = ds.Tables["SinhVien"];

con.Close();

}

1. **Code thực hiện chức năng tìm kiếm ở form sinh viên**

private void button5\_Click(object sender, EventArgs e)

{

SqlConnection con = new SqlConnection(sCon);

try

{

con.Open();

}

catch (Exception ex)

{

MessageBox.Show("Xảy ra lỗi trong quá trình kết nối DB5!");

}

string sMaSV = textBox7.Text;

string sQuery = "select \*from SINHVIEN where MaSV=@MaSV";

SqlDataAdapter adapter = new SqlDataAdapter(sQuery, con);

SqlCommand cmd = new SqlCommand(sQuery, con);

cmd.Parameters.AddWithValue("@MaSV", sMaSV);

SqlDataReader dr = cmd.ExecuteReader();

DataTable dt = new DataTable();

dt.Load(dr);

dataGridView1.DataSource = dt;

}

1. **Code tạo báo cáo thống kê theo lớp**

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

{

SqlConnection con = new SqlConnection();

con.ConnectionString = Properties.Settings.Default.QLHSSVConnectionString;

SqlCommand cmd = new SqlCommand();

cmd.CommandText = "THONGKE";

cmd.CommandType = CommandType.StoredProcedure;

cmd.Connection = con;

cmd.Parameters.Add(new SqlParameter("@MaLop", txtML.Text));

DataSet ds = new DataSet();

SqlDataAdapter da = new SqlDataAdapter(cmd);

da.Fill(ds);

reportViewer1.ProcessingMode = ProcessingMode.Local;

reportViewer1.LocalReport.ReportPath = "Report1.rdlc";

ReportDataSource rds = new ReportDataSource();

rds.Name = "DataSet1";

rds.Value = ds.Tables[0];

reportViewer1.LocalReport.DataSources.Clear();

reportViewer1.LocalReport.DataSources.Add(rds);

reportViewer1.RefreshReport();

}

1. **Ghi chú**

* Code các chức năng cho form khoa, lớp, môn học tương tự form sinh viên (mục 4 🡪 10).
* Code thống kê theo khoa, phường/xã, quận/huyện, tỉnh/thành phố tương tự code thống kê theo lớp (mục 11).