**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM**

**KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**

**NGÀNH CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO ĐỒ ÁN CUỐI KỲ**

**MÔN HỌC: HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU**

**QUẢN LÝ TUYỂN SINH**

**SVTH1: LÊ QUỐC NGUYÊN VƯƠNG - 17145253**

**SVTH2: VÕ THỊ KIỀU DIỄM - 17110109**

**GVPT: ThS. HOÀNG LONG**

**Tp. Hồ Chí Minh, tháng 12 năm 2019**

**MỤC LỤC**

Danh mục hình 1

Danh mục bảng 3

LỜI CẢM ƠN 4

CHƯƠNG 1: TỔNG QUAN ĐỒ ÁN 5

1.1. Giới thiệu phần mềm quản lý tuyển sinh 5

1.2. Lý do, mục đích, mục tiêu chọn đề tài 5

1.2.1. Lý do, mục đích chọn đề tài 5

1.2.2. Mục tiêu chọn đề tài 5

1.3. Lựa chọn ngôn ngữ lập trình 6

CHƯƠNG 2: NỘI DUNG ĐỒ ÁN 7

2.1. Phân tích yêu cầu 7

2.1.1. Yêu cầu về chức năng 7

2.1.2. Yêu cầu phi chức năng 10

2.1.3. Yêu cầu khác 11

2.2. Thiết kế chức năng 11

2.2.1. Thiết kế kiến trúc 11

2.2.2. Thiết kế chi tiết 11

2.2.2.1. Thiết kế chi tiết SQL 11

2.2.2.2. Thiết kế chi tiết đối tượng. 13

2.2.3. Thiết kế giao diện 14

2.2.4. Thiết kế cơ sở dữ liệu 15

2.2.4.1. Stored 15

2.2.4.2. Trigger 20

2.2.4.3. Function 22

2.2.4.4. Transaction 22

2.2.4.5. Index 24

Chương 3: HIỆN THỰC HÓA HỆ THỐNG 27

3.1. Class đối tượng DTO. 27

3.2. Class chức năng DAO. 27

3.3. PresentationLayer 29

CHƯƠNG 4: KẾT LUẬN 36

4.1. Tổng kết 36

4.2. Ưu điểm 36

4.3. Nhược điểm 36

4.4. Hướng phát triển 36

BẢNG PHÂN CÔNG CÔNG VIỆC 38

# Danh mục hình

Hình 1: Mô hình ERD 7

Hình 2: Database Datagram 11

Hình 3: Giao diện chính 15

Hình 4: CHECKLOGIN 15

Hình 5: LoadTable 16

Hình 6: InsertAccount 16

Hình 7: Edit data 17

Hình 8: DeleteAccount 18

Hình 9: Reset Pass 18

Hình 10: Search data 19

Hình 11: Trigger Insert 20

Hình 12: Trigger Edit 21

Hình 13: Function Loaddata 22

Hình 14: Check Điểm 22

Hình 15: Transaction 23

Hình 16: Transaction1 24

Hình 17: Index 24

Hình 18: Select index 25

Hình 19: Index HoKhau 25

Hình 20: Index Nganh 25

Hình 21: declare index 26

Hình 22: Index Nganh1 26

Hình 23: Index NGANH 26

Hình 24: Select NGANH 26

Hình 25: From Loading 30

Hình 26: Form Login 30

Hình 27: Form Quản lý 31

Hình 28: Form Quản Lý Ngành 31

Hình 29: Form quản lý đối tượng 32

Hình 30: Form quản lý khu vực 33

Hình 31: Form Thông tin tài khoản 33

Hình 32: Quản lý sinh viên 34

Hình 33: Tra cứu điểm 34

# 

# Danh mục bảng

Bảng 1: Chức năng 10

Bảng 2: Thiết kế SQL 13

Bảng 3: Thiết kế đối tượng 14

Bảng 4: DTO 27

Bảng 5: DAO 29

Bảng 6: Phân công 38

# LỜI CẢM ƠN

Nhóm thực hiện xin chân thành cảm ơn sâu sắc về sự hướng dẫn tận tình của thầy Hoàng Long, cả về chuyên môn lẫn định hướng cho nhóm trong quá trình học tập cũng như quá trình hoàn thành đồ án. Nhờ sự chỉ dẫn của thầy mà nhóm có được cơ sở lý thuyết vững vàng cũng như là khả năng áp dụng các thuật toán đã học trong đề tài. Với điều kiện thời gian cũng như kiến thức chuyên môn còn hạn chế nên nhóm chưa thể tối ưu hết các thuật toán sử dụng trong phần mềm quản lý tuyển sinh cũng như không thể tránh khỏi các thiếu sót trong quá trình thực hiện đồ án. Vì vậy nhóm thực hiện rất mong nhận được sự chỉ bảo và đóng góp ý kiến của thầy để đồ án của nhóm có thể hoàn thiện và phát triển hơn.

.

# CHƯƠNG 1: TỔNG QUAN ĐỒ ÁN

## Giới thiệu phần mềm quản lý tuyển sinh

Ngày nay, công nghệ thông tin ngày càng phát triển và được phát triển vào nhiều lĩnh vực và mọi ngành ngành nghề. Vì vậy để phục vụ cho công việc, ngành giáo dục và đào tạo cũng là một ngành cần thiết phải áp dụng công nghệ vào lĩnh vực quản lý tuyển sinh. Xuất phát từ những yêu cầu thực tế về công tác tuyển sinh có nhiều thành phần và thông tin cần quản lý rất lớn cũng như để tiết kiệm thời gian nên rất cần đến sự trợ giúp của phần mềm quản lý.

## Lý do, mục đích, mục tiêu chọn đề tài

### Lý do, mục đích chọn đề tài

Sau khi học môn lập trình hướng đối tượng, cấu trúc dữ liệu và lập trình hướng đối tượng, nhóm được tiếp xúc nhiều hơn với các khối kiến thức cơ sở, điều đó giúp nhóm có thể xây dựng các chương trình phần mềm quản lý trên máy tính, áp dụng các kiểu dữ liệu trong thực tế. Thông qua đó, nhóm vận dụng kiến thức về stored, trigger, view, scrip, index... để xây dựng **phần mềm quản lý tuyển sinh đại học**. Với mục đích xây dựng phần mềm với tính tiện dụng, gần gũi, dễ quản lý, dễ sử dụng và tính chính xác cao, phù hợp với mọi người dùng và bám sát kiến thức đã học. Và quan trọng là hiểu rõ hơn về các program sử dụng, tìm hiểu thêm về thiết kế giao diện trong phần mềm, rèn luyện kỹ năng lập trình, làm tiền đề cho các môn học sau này và ứng dụng mai sau. Vì vậy nhóm quyết định chọn đề tài xây dựng phần mềm quản lý tuyển sinh đại học.

### Mục tiêu chọn đề tài

Với mục tiêu tạo ra một phần mềm quản lý tuyển sinh dễ sử dụng và đáng tin cậy chính xác với đầy đủ các chức năng bằng cách sử dụng stored, trigger, view, scrip, index … Đặc biệt, áp dụng stored để lưu trữ thông tin và và kết nối sql server để đồng bộ dữ liệu được lưu trữ. Kết hợp với việc xây dựng giao diện dễ sử dụng và chính xác.

## Lựa chọn ngôn ngữ lập trình

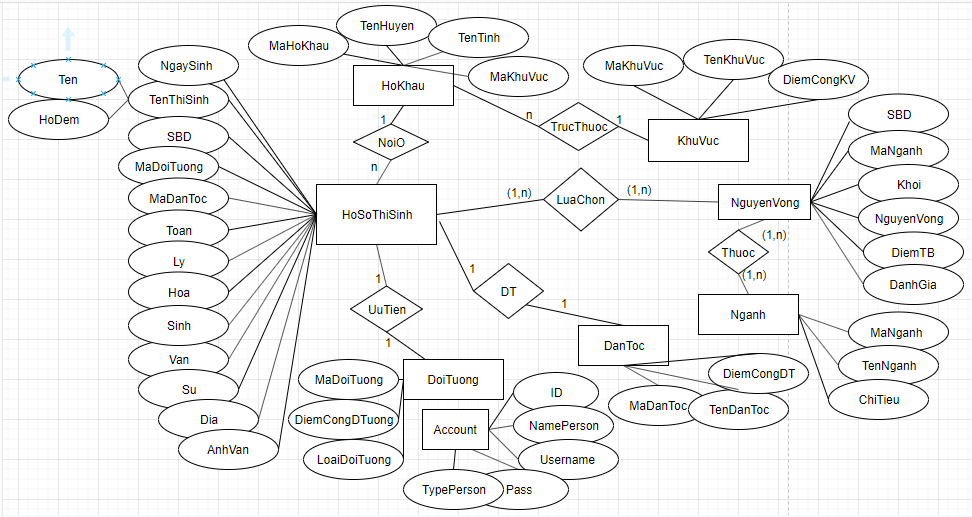
Vì chương trình mô phỏng của nhóm biễu diễn thuật toán dưới dạng các đối tượng thuộc tính, hành vi rõ ràng nên nhóm quyết định sử dụng ngôn ngữ Microsoft C#.NET với phiên bản .NET Framework 4.5.2; IDE và compiler nằm trong bộ Microsoft Visual Studio 2019 và Microsoft SQL Management Studio 2017 để cài đặt mô phỏng này.

# CHƯƠNG 2: NỘI DUNG ĐỒ ÁN

## 2.1. Phân tích yêu cầu

### 2.1.1. Yêu cầu về chức năng

Đồ án được nhóm thiết kế với sơ đồ ERD như sau:



Hình 1: Mô hình ERD

Xác định các đối tượng, mô tả thuộc tính và các phương thức.

* Hồ sơ Thí Sinh: SBD, Tên Thí Sinh, Ngày Sinh, Giới Tính, Nơi ở, Ngành học, Dân tộc, Đối tượng, Toán, Lý, Hóa, Sinh, Sử, Địa, Văn, Anh văn.
* Ngành: Mã Ngành, Tên Ngành, Chỉ Tiêu
* Dân tộc: Mã Dân Tộc, Tên Dân Tộc, Điểm Cộng Dân Tộc.
* Nguyện Vọng: SBD, Khối, Điểm Trung Bình, Mã Ngành, Đánh giá.
* Đối Tượng: Mã Đối Tượng, Loại Đối Tượng, Điểm Cộng Đối Tượng.
* Hộ Khẩu: Mã Hộ Khẩu, Tên Huyện, Tên Tỉnh
* Khu Vực: Mã Khu Vực, Tên Khu Vực, Điểm Cộng KV.
* Acount: ID, Tên Thường Dùng, Tên Đăng Nhập, Loại Đối Tượng.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Stt | Table | Form | User | Chức năng | | Chi tiết |
| 11 | Account | Quản lý | Nhân viên phòng tuyển sinh | Login  Administration (hiện quản lý Account, quản lý khu vực, quản lý đối tượng, quản lý ngành, tra cứu ngành, quản lý sinh viên) | | Chỉ có admin mới có thể đăng nhập và sử dụng các quyền được cấp. |
| Tra cứu nhanh | |
| 22 | Login | Đăng nhập  Xóa  Thoát | | User  Password |
| 33 | Quản lý Account | Thêm, lưu, Re pass, hủy, xóa, thoát  DataGridViewRow | | Tên hiển thị  User  Quyền |
| 44 | Thông tin tài khoản | Trở về, cập nhật, xóa, thoát | | Thông tin tài khoản (tên đăng nhập, tên hiển thị, quyền)  Chỉnh sửa tài khoản (mật khẩu cũ, tên hiển thị, mật khẩu mới, xác nhận) |
| 55 | Khuvuc | Quản lý khu vực | Danh mục | Thêm  Sửa  Xóa  Lưu  Hủy  DataGridView | Makhuvuc  Tenkhuvuc  Diemcongkhuvuc |
| Hokhau | Khu vực | Thêm  Sửa  Xóa  Lưu  Hủy  DataGridView | Mahokhau  Tenhuyen  Tentinh  Makhuvuc |
| 66 | Nganh | Quản lý ngành | Thêm, sửa, lưu, hủy, xóa, thoát  DataGridViewRow | | Manganh  Tennganh  Chitieu |
| 77 | Doituong | Quản lý đối tượng | Thêm, sửa, lưu, hủy, xóa, thoát  DataGridViewRow | | Madoituong  Loaidoituong  Diemcongdtuong |
| 88 | Hosothisinh | Quản lý sinh viên | Nhập lại, lưu, cập nhật, xóa, thoát  DataGridViewRow | | SBD  Hodem  Ten  Ngaysinh  Gioitinh  Mahokhau  Madoituong  Madantoc  Manganh  Nguyenvong |
| 99 | NguyenVong | Tra cứu điểm | Sinh viên  Nhân viên phòng quản lý | In, thoát  DataGridViewRow | | SBD |
| 110 | Nganh | Tra cứu ngành | Sinh viên | Tra cứu | | Manganh  Tennganh  Chitieu |

Bảng 1: Chức năng

### 2.1.2. Yêu cầu phi chức năng

- Hiệu năng hoạt động: Thông tin sinh viên, thông tin ngành, đối tượng ưu tiên, thông tin nguyện vọng…

- Giao diện đẹp và thân thiện dễ dùng: vì sự tra cứu và quản lý các thông tin tuyển sinh cần nhanh chóng và chính xác nên các form nên thiết kế đơn giản, thuận tiện, giao diện nên hướng đến cá nhân hóa theo người dùng.

- Phân quyền chặt chẽ: quyền người dung thông qua chức năng (màn hình) và dữ liệu. Các chức năng thêm, sửa, xóa, cập nhật, lưu được thiết kế độc lập giúp cho người quản lý dễ dàng sử dụng và kiểm soát dữ liệu.

- Ổn định và xử lý nhanh: Các thao tác thêm, sửa, xóa được quản lý chặt chẽ. Các số liệu đã nhập được rang buộc trên nền tảng hệ thống tổng thể. Các thủ tục xử lý và truy xuất dữ liệu được phân chia tối ưu, do tốc độ tính toán và xử lý nhanh.

- Tính năng mở và mềm dẻo: Giải pháp cung cấp nhiều lựa chọn để người dung có thể tùy biến chương trình phù hợp với phương thức hoạt động của mình. Người dùng có thể quy định chi tiết hệ thống các phương thức tính điểm.

- Tính kế thừa cao: hệ thống các phân hệ kế thừa và xử lý tự động dữ liệu và các báo cáo có thể xuất sang các file, dễ dàng kết nối với hệ thống khác.

- Báo cáo, biểu mẫu phong phú, đa dạng.

### 2.1.3. Yêu cầu khác

Tương tác với phần mềm quản lý của trường đại học và của bộ giáo dục.

## 2.2. Thiết kế chức năng

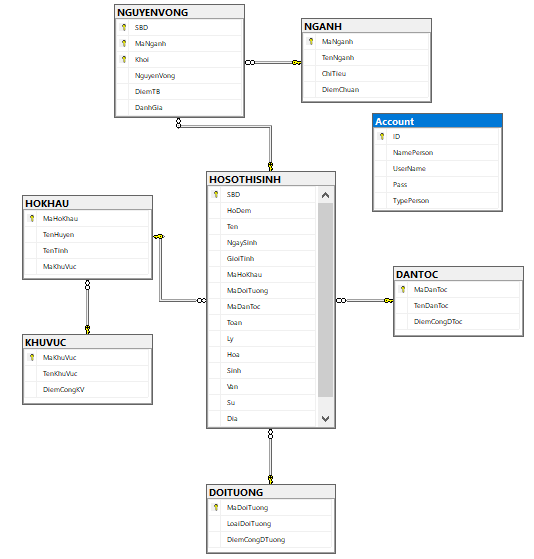
### 2.2.1. Thiết kế kiến trúc

Đồ án được thực hiện với cấu trúc 3 tầng:

* - DTO: tầng dữ liệu.
* - DAO: tầng xử lý.

- PresentationLayer: tầng giao diện.

### 2.2.2. Thiết kế chi tiết



Hình 2: Database Datagram

#### 2.2.2.1. Thiết kế chi tiết SQL

|  |  |  |
| --- | --- | --- |
| Table | Thuộc tính | Kiểu dữ liệu |
| Account | ID | int |
| NamePerson | varchar |
| UserName | varchar |
| Pass | varchar |
| TypePerson | int |
| DANTOC | MaDanToc | int |
| TenDanToc | varchar |
| DiemCongDToc | float |
| DOITUONG | MaDoiTuong | int |
| LoaiDoiTuong | varchar |
| DiemCongDTuong | float |
| KHUVUC | MaKhuVuc | varchar |
| TenKhuVuc | varchar |
| DiemCongKV | float |
| HOKHAU | MaHoKhau | int |
| TenHuyen | varchar |
| TenTinh | varchar |
| MaKhuVuc | varchar |
| HOSOTHISINH | SBD | varchar |
| HoDem | varchar |
| Ten | varchar |
| NgaySinh | smalldatetime |
| GioiTinh | bit |
| MaHoKhau | int |
| MaDoiTuong | int |
| MaDanToc | int |
| Toan | float |
| Ly | float |
| Hoa | float |
| Sinh | float |
| Van | float |
| Su | float |
| Dia | float |
| AnhVan | float |
| NGUYENVONG | SBD | varchar |
| MaNganh | varchar |
| Khoi | varchar |
| NguyenVong | int |
| DiemTB | float |
| DanhGia | varchar |
| NGANH | MaNganh | varchar |
| TenNganh | varchar |
| ChiTieu | int |
| DiemChuan | float |

Bảng 2: Thiết kế SQL

#### 2.2.2.2. Thiết kế chi tiết đối tượng.

|  |  |  |
| --- | --- | --- |
| STT | Class | Chức Năng. |
| 1 | Account | * Đại diện cho đối tượng Account. Chứa các thuộc tính của Account. |
| 2 | DANTOC | * Đại diện cho đối tượng Dân Tộc. Chứa các thuộc tính của Dân Tộc. |
| 3 | DOITUONG | * Đại diện cho đối tượng mang tên Đối Tượng. Chứa các thuộc tính của Đối Tượng. |
| 4 | HOKHAU | * Đại diện cho đối tượng Hộ Khẩu. Chứa các thuộc tính của Hộ Khẩu. |
| 5 | HOSOTHISINH | * Đại diện cho đối tượng Thí Sinh. Chứa các thuộc tính của Thí Sinh. |
| 6 | KHUVUC | * Đại diện cho đối tượng Khu Vực. Chứa các thuộc tính của Khu Vực. |
| 7 | NGANH | * Đại diện cho đối tượng Ngành. Chứa các thuộc tính của Ngành. |
| 8 | NGUYENVONG | * Đại diện cho đối tượng Nguyện Vọng. Chứa các thuộc tính của Nguyện Vọng. |

Bảng 3: Thiết kế đối tượng

### 2.2.3. Thiết kế giao diện

Phần mềm Quản lý Tuyển sinh của nhóm thực hiện bao gồm các form làm việc như sau:

- Form Loading.

- Form Login.

- Form quản lý.

- Form thông tin tài khoản.

- Form quản lý sinh viên.

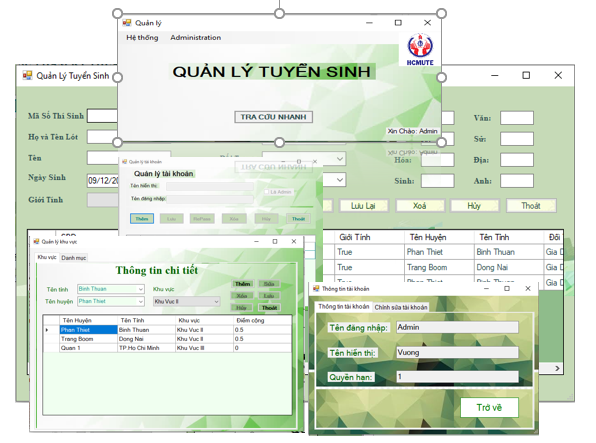
- Form quản lý ngành.

- Form quản lý khu vực.

- Form quản lý đối tượng.

- Form tra cứu điểm.

- Form tra cứu ngành.

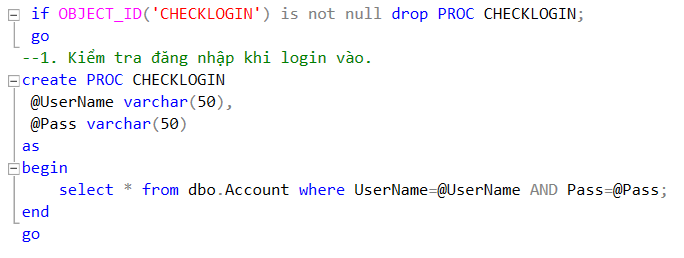


Hình 3: Giao diện chính

### 2.2.4. Thiết kế cơ sở dữ liệu

#### 2.2.4.1. Stored

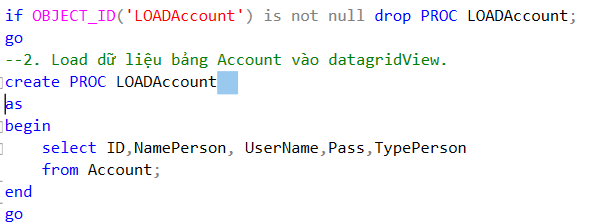
##### Check Login



Hình 4: CHECKLOGIN

Sao khi người dùng đăng nhập, phần mềm sẽ truyền tên đăng nhập và password xuống cơ sở dữ liệu bằng cách gọi Store Procduce để kiểm tra. Nếu tên đăng nhập và password là đúng thì SQL sẽ trả về giá trị thực hiện dòng lệnh đã thực thi được cho visual. Khi đó người dùng mới đăng nhập được vào form quản lý.

##### Load Table

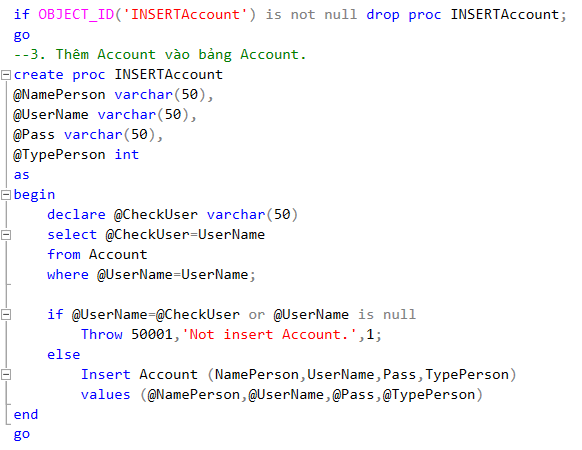


Hình 5: LoadTable

Khi vào giao diện một form, gọi Store Procduce để lấy dữ liệu datatable từ SQL lên visual truyền vào các đối tượng đã tạo sẵn, và tải lên các trên giao diện để người dùng xem.

Một số hàm tương tự: LOADNganh, LOADDoiTuong, LOADHoKhau, …

##### Insert data

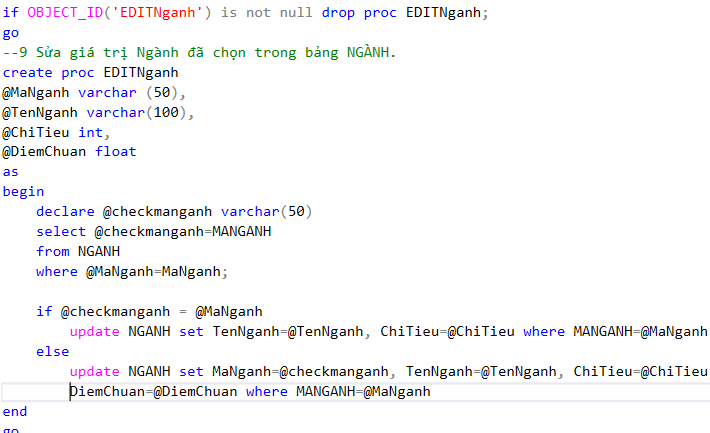


Hình 6: InsertAccount

Khi người dùng thêm một giá trị từ trên form, sẽ được gọi đến Store Procduce Insert… đã có sẵn, và thực hiện chức năng insert vào bảng của đối tượng. Nếu thành công SQL sẽ trả ra giá trị cho visual để thông báo cho người dùng biết.

Một số hàm tương tự: INSERTNganh, INSERTDoiTuong, INSERTKhuVuc, …

##### Edit data

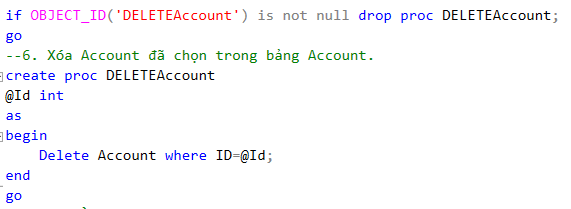


Hình 7: Edit data

Khi người dùng chỉnh sửa giá trị của một đối tượng tren form, sẽ được gọi đến Store Procduce EDIT… đã được tạo sẵn, và thực hiện chức năng update đối tượng trong bảng của đối tượng đó. Nếu thành công SQL sẽ trả ra giá trị cho visual để thông báo cho người dùng biết.

Một số hàm tương tự: EDITDoiTuong, EDITHoKhau hay UPDATEAccount, …

##### Delete data



Hình 8: DeleteAccount

Khi người dùng xóa giá trị của một đối tượng trên form, sẽ được gọi đến Store Procduce DELETE… đã được tạo sẵn, và thực hiện chức năng delete đối tượng trong bảng. Nếu thành công SQL sẽ trả giá trị báo cho visual để thông báo cho người dùng biết.

Một số hàm tương tự: DELETENganh, DELETEDoiTuong, DETELEHoKhau, …

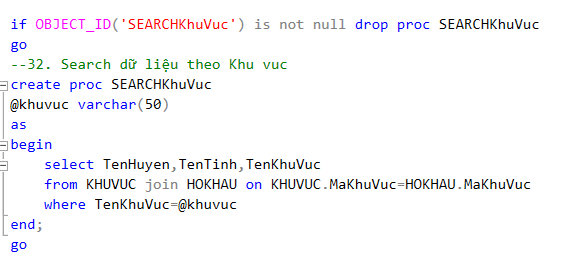
##### Reset Pass

##### 

Hình 9: Reset Pass

Khi người dùng chọn sự kiện Reset pass trong form Quản Lý Accont. Sẽ thực hiện Store Procduce Reset Pass, thực hiện chức năng đổi mật khẩu hiện tại của ID đã chọn về thành “1”. Nếu thay đổi thành công sẽ báo lại cho người dùng để theo dõi.

##### Search data



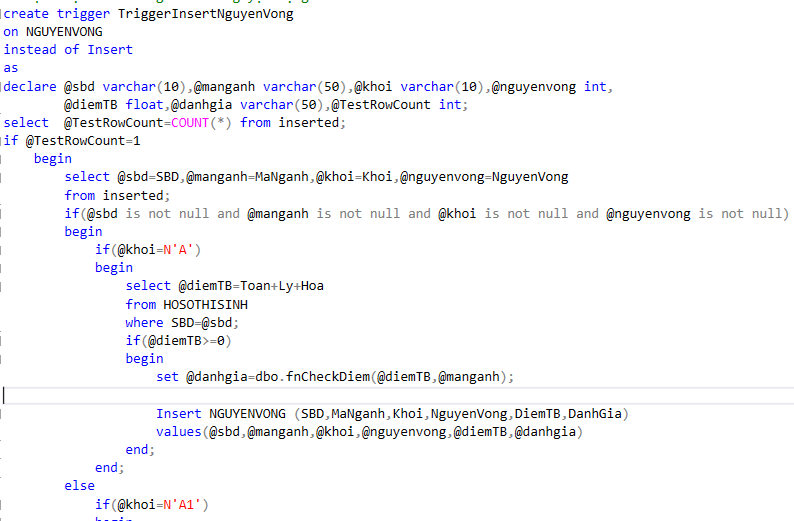
Hình 10: Search data

Khi người dùng tìm kiếm dữ liệu với từ khóa, sẽ gọi hàm Store Procduce đã được tạo sẵn, và thực hiện chức năng Sreach, sẽ trả về cho người dùng xem những giá trị sau khi Sreach được.

Một số hàm tương tự: LOADTraCuuNganh, LOADTraCuuDiem.

#### 2.2.4.2. Trigger

##### Insert



Hình 11: Trigger Insert

Khi người dùng thêm giá trị cho một đối tượng trên form, giá trị sẽ được truyền vào Store Procduce INSERTNguyenVong và thực hiện chức insert. Khi đó TriggerInsertNguyenVong sẽ được sử dụng để thực hiện chức năng của chính hàm. Hàm sẽ tự tính @diemTB dựa vào giá trị @khoi, tiếp theo đó giá trị @danhgia sẽ được tính dựa vào function với biến truyền vào là @diemTB và @manganh. Khi có đầy đủ giá trị sẽ bắt đầu thực hiện insert vào bảng NGUYENVONG, và báo giá trị về cho người dùng khi đã thực hiện thành công.

##### Edit



Hình 12: Trigger Edit

Khi người dùng thay đổi giá trị cho một đối tượng trên form, giá trị sẽ được truyền vào Store Procduce EDITNguyenVong và thực hiện chức insert. Khi đó TriggerUpdateNguyenVong sẽ được sử dụng để thực hiện chức năng của chính hàm. Hàm sẽ tự tính @diemTB dựa vào giá trị @khoi, tiếp theo đó giá trị @danhgia sẽ được tính dựa vào function với biến truyền vào là @diemTB và @manganh. Khi có đầy đủ giá trị sẽ bắt đầu thực hiện update giá trị trong bảng NGUYENVONG, và báo giá trị về cho người dùng khi đã thực hiện thành công.

#### 2.2.4.3. Function

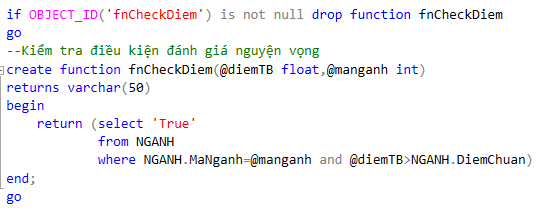
##### Load data

##### 

Hình 13: Function Loaddata

Khi vào giao diện form Quản lý Ngành, gọi function để lấy dữ liệu datatable từ SQL lên visual truyền vào các đối tượng đã tạo sẵn, và tải lên các trên giao diện để người dùng xem.

##### Check Điểm

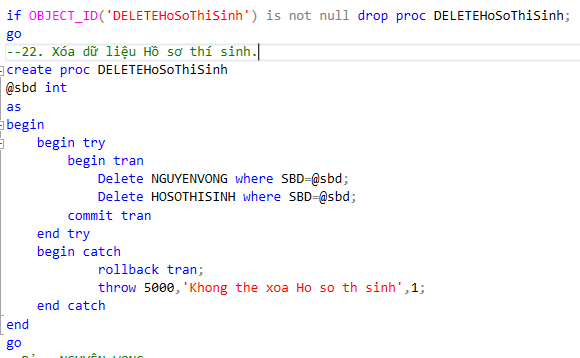


Hình 14: Check Điểm

Hàm function được sử dụng trong TriggerInsertNguyenVong và, TriggerUpdateNguyenVong để thực hiện việc kiểm tra @diemTB so với DiemChuan nếu lớn hơn thì sẽ trả ra giá trị ‘True’.

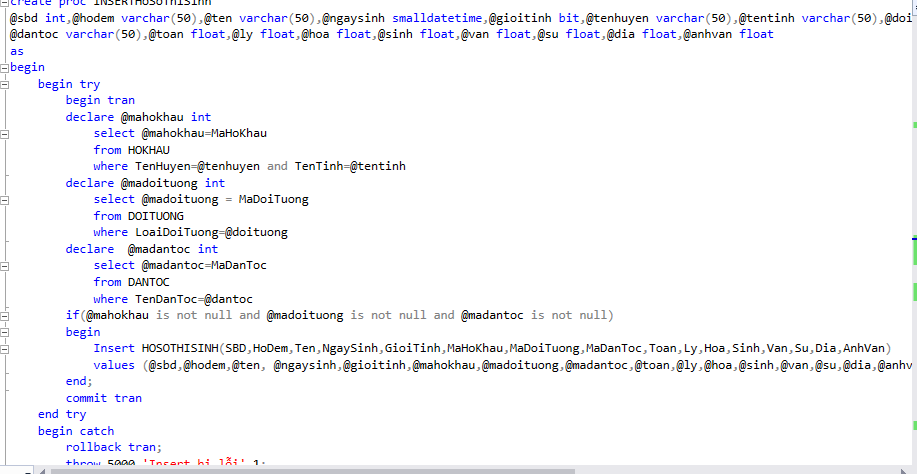
#### 2.2.4.4. Transaction

Thực hiện xóa dữ liệu hồ sơ thí sinh và nguyện vọng, nếu giữa chừng lỗi thì sang catch và rollback giá trị.



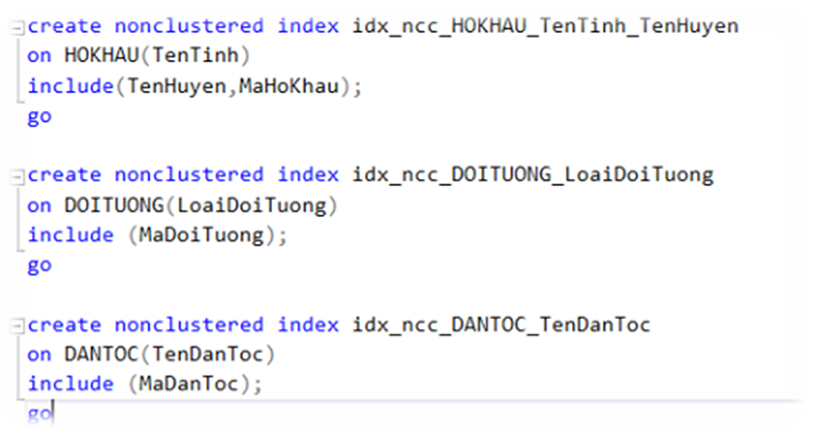
Hình 15: Transaction

Thực hiện bình thường insert hồ sơ thí sinh nếu có lỗi thì chuyển sang catch và rollback giá trị.



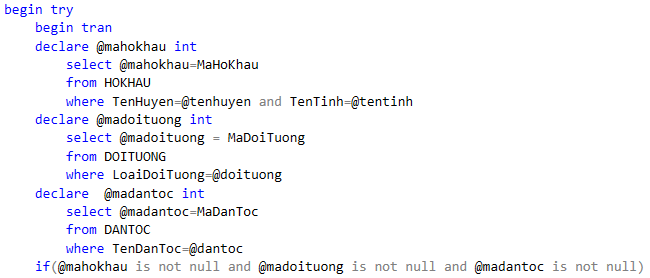
Hình 16: Transaction1

#### 2.2.4.5. Index



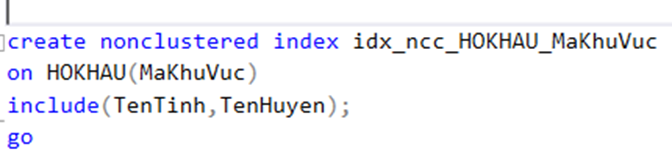
Hình 17: Index

Tiến hành tạo index nonclustered cho các bảng HOKHAU, DOITUONG, DANTOC thay vì scan tất cả dòng thì nó chỉ lấy thẳng những thông tin cần thiết. Hỗ trợ cho việc tìm kiếm dữ liệu trong bảng HOSOTHISINH để thêm dữ liệu.



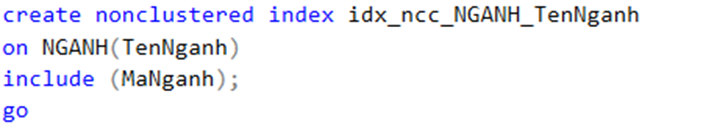
Hình 18: Select index

Vì câu truy vấn này từ bảng hộ khẩu, truy vấn theo tên tỉnh và tên huyện. Nên clustered MaHoKhau không được sử dụng. Vì thế ta cần đặt nonclustered dựa theo TenTinh, và include TenHuyen và MaHoaDon để truy vấn được thực hiện nhanh hơn.



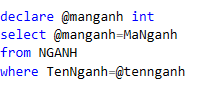
Hình 19: Index HoKhau

Tạo index cho Hokhau, nội dung truy suất chính là Makhuvuc và bao gồm thêm tentinh, tenhuyen.



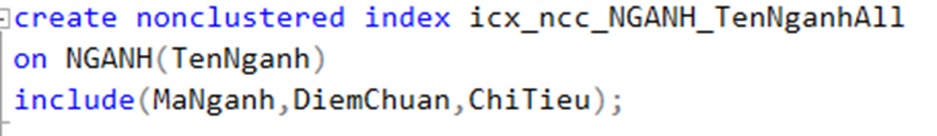
Hình 20: Index Nganh

Tạo index cho Nganh với nội dung truy suất chính là Tennganh và truy vấn đến MaNganh.



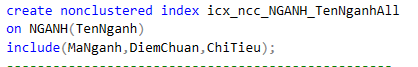
Hình 21: declare index

Hỗ trợ cho viện tìm theo TenNganh để thực hiện truy vấn nhanh hơn.



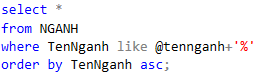
Hình 22: Index Nganh1

Tạo index nonclustered cho Nganh, truy vấn đến Manganh, DiemChuan, ChiTieu phục vụ cho việc tìm kiếm ngành được nhanh hơn.



Hình 23: Index NGANH

Vì đây là show tất cả dữ liệu hiện có nên nonclustered đã có sẵn không tối ưu được câu truy vấn này. Vì thế đặt một nonclustered khác hỗ trợ cho việc tìm kiếm tất cả bằng cách include (MaNganh, ChiTieu, DiemChuan).



Hình 24: Select NGANH

# Chương 3: HIỆN THỰC HÓA HỆ THỐNG

## 3.1. Class đối tượng DTO.

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên lớp** | **Mục đích sử dụng** |
| 1 | HOSOTHISINH | Đại điện cho đối tượng Thí Sinh. Chứa các thuộc tính của Thí Sinh. |
| 2 | HOKHAU | Đại diện cho đối tượng Hộ Khẩu. Chứa các thuộc tính của Hộ Khẩu. |
| 3 | KHUVUC | Đại diện cho đối tượng Khu Vực. Chứa các thuộc tính của Khu Vực. |
| 4 | DANTOC | Đại diện cho đối tượng Dân Tộc. Chứa các thuộc tính của Dân Tộc. |
| 5 | DOITUONG | Đại diện cho Đối Tượng. Chứa các thuộc tính của Đối Tượng. |
| 6 | NGANH | Đại diện cho đối tượng Ngành. Chứa các thuộc tính của Ngành. |
| 7 | NGUYENVONG | Đại diện cho đối tượng Nguyện Vọng. Chứa các thuộc tính của Nguyện Vọng. |
| 8 | ACCOUNT | Đại điện cho đối tượng Account. Chứa các thuộc tính của Account. |

Bảng 4: DTO

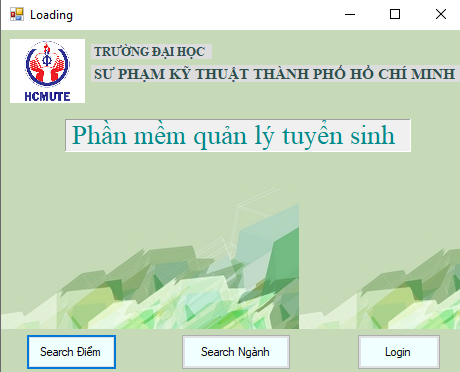
## 3.2. Class chức năng DAO.

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên lớp** | **Mục đích sử dụng** |
| 1 | QuanLyTuyenSinhConText | Kết nối đến cơ sở dữ liệu |
| 2 | AccountDAO | Xử lí bảng ACCOUNT: Đáp ứng các yêu cầu thao tác dữ liệu của GUI layer, có các nghiệp vụ xử lý như: insert, delete, update, … |
| 3 | DanTocDAO | Xử lí bảng DANTOC: Đáp ứng các yêu cầu thao tác dữ liệu của GUI layer, có các nghiệp vụ xử lý như: insert, delete, update, … |
| 4 | DoiTuongDAO | Xử lí bảng DOITUONG: Đáp ứng các yêu cầu thao tác dữ liệu của GUI layer, có các nghiệp vụ xử lý như: insert, delete, update, … |
| 5 | HoKhauDAO | Xử lí bảng HOKHAU: Đáp ứng các yêu cầu thao tác dữ liệu của GUI layer, có các nghiệp vụ xử lý như: insert, delete, update, … |
| 6 | HoSoThiSinhDAO | Xử lí bảng HOSOTHISINH: Đáp ứng các yêu cầu thao tác dữ liệu của GUI layer, có các nghiệp vụ xử lý như: insert, delete, update, … |
| 7 | KhuVucDAO | Xử lí bảng KHUVUC: Đáp ứng các yêu cầu thao tác dữ liệu của GUI layer, có các nghiệp vụ xử lý như: insert, delete, update, … |
| 8 | NganhDAO | Xử lí bảng NGANH: Đáp ứng các yêu cầu thao tác dữ liệu của GUI layer, có các nghiệp vụ xử lý như: insert, delete, update, … |
| 9 | NguyenVongDAO | Xử lí bảng NGUYENVONG: Đáp ứng các yêu cầu thao tác dữ liệu của GUI layer, có các nghiệp vụ xử lý như: insert, delete, update, … |

Bảng 5: DAO

## 3.3. PresentationLayer

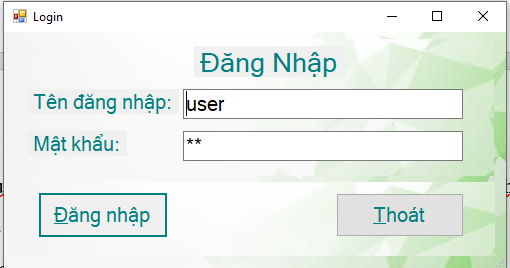
- Form Loading



Hình 25: From Loading

Giao diện xuất hiện đầu tiên sau khi chạy chương trình, với chức năng chính là cho phép thí sinh nếu không có login thì vẫn có thể tìm kiếm điểm số của mình hay tìm kiếm thông tin ngành. Còn khi vào login sẽ bắt buộc người dùng phải đăng nhập vào, nếu đăng nhập chính xác sẽ vào được trang quản lý chính của phần mềm quản lý tuyển sinh.

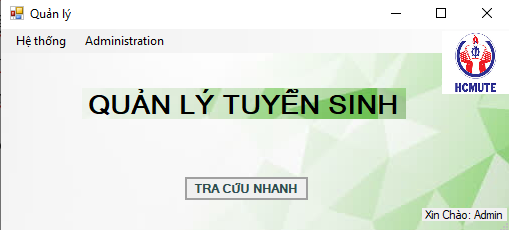
- Form Login



Hình 26: Form Login

Giao diện Login bắt buộc người dùng cần phải đăng nhập để có thể sử dụng phần mềm quản lý tuyển sinh. Ở đây khi nhấn đăng nhập giá trị tên đăng nhập và mật khẩu sẽ được gửi xuống cơ sở dữ liệu để xử lý kiểm tra tính đúng sai. Nếu đúng thì sẽ được vào phần mềm chính.

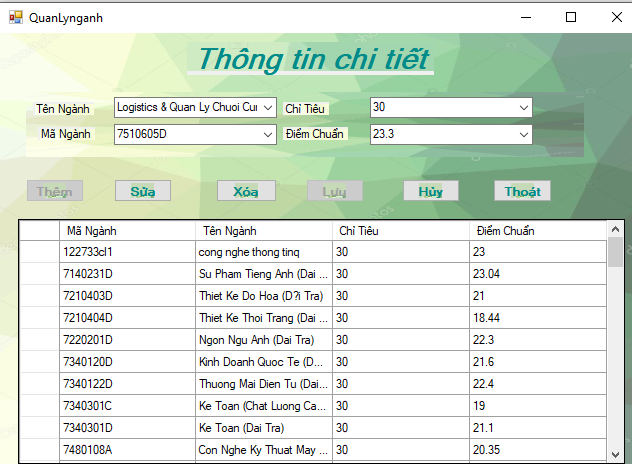
- Form quản lý



Hình 27: Form Quản lý

Đây là giao diện chính của phần mềm quản lý tuyển sinh. Có thể xem thông tài khoản trong menustrip hệ thống, hay thực hiện các chức năng dành cho quản lý trong menustrip Administration. Ngoài ra Admin còn có thể tra cứu điểm một đối tượng thí sinh, thông qua button tra cứu nhanh.

- Form quản lý ngành



Hình 28: Form Quản Lý Ngành

Ở giao diện này người quản lí xem được dữ liệu được truyền từ cơ sở dữ liệu và có thể thao tác nghiệp vụ thêm, sửa, xóa cho đối tượng ngành. Và các chức năng này được thao tác dưới SQL Server.

- Form quản lý đối tượng



Hình 29: Form quản lý đối tượng

Ở giao diện này người quản lí xem được dữ liệu được truyền từ cơ sở dữ liệu và có thể thao tác nghiệp vụ thêm, sửa, xóa cho loại đối tượng. Và các chức năng này được thao tác dưới SQL.

- Form quản lý khu vực

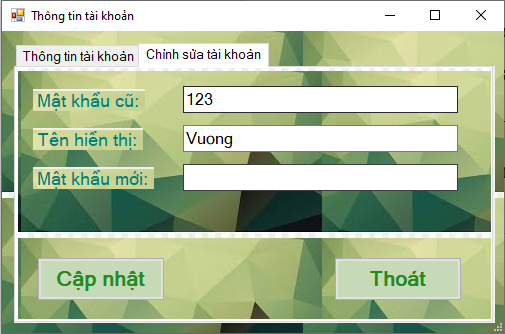


Hình 30: Form quản lý khu vực

Ở giao diện này người quản lí xem được dữ liệu được truyền từ cơ sở dữ liệu và có thể thao tác nghiệp vụ thêm, sửa, xóa cho loại đối tượng Hộ Khẩu. Và các chức năng này được thao tác dưới SQL.

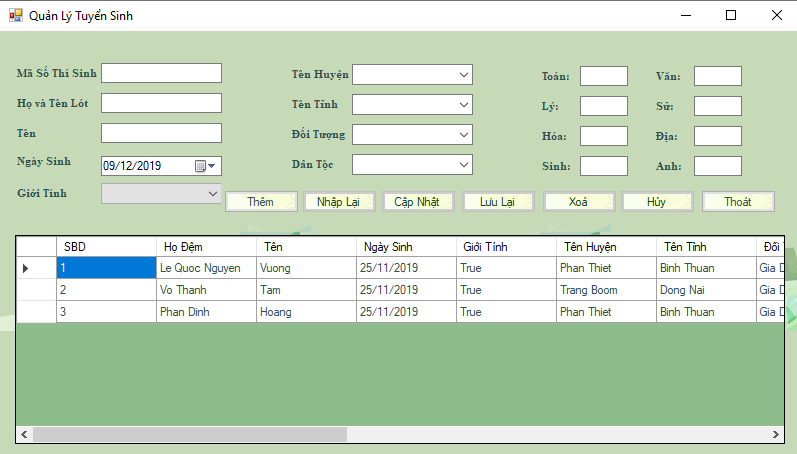
- Form thông tin tài khoản

Khi thao tác với form này thì người quản lý được xem và thay đổi thông tin đăng nhập.



Hình 31: Form Thông tin tài khoản

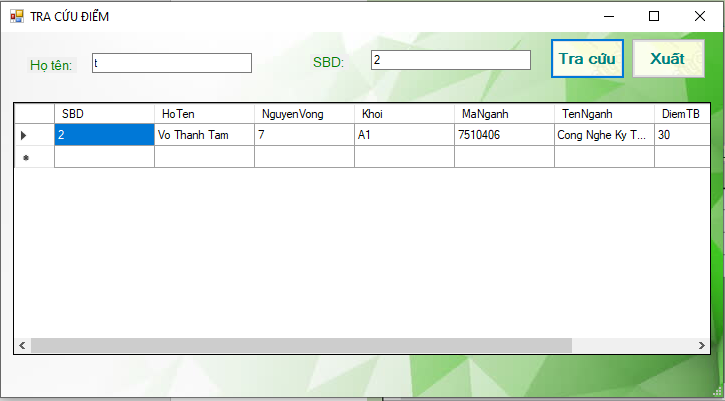
- Form quản lý sinh viên



Hình 32: Quản lý sinh viên

Người quản lý có thể thêm cập nhật lưu hay các thao tác trên form này. Và chức năng này được thao tác dưới sql.

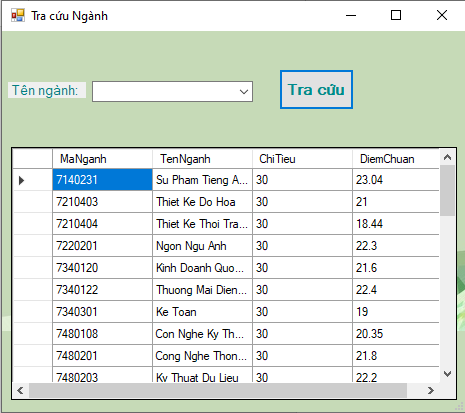
- Form tra cứu điểm



Hình 33: Tra cứu điểm

Người dùng cả sinh viên hay người quản lý có thể tra cứu kết quả của sinh viên với họ tên và SBD. Mọi thao tác điều được thực hiện dưới sql.

- Form tra cứu ngành



Hình 34: Tra cứu ngành

# CHƯƠNG 4: KẾT LUẬN

## 4.1. Tổng kết

Nhóm đã hoàn thành hơn 90% mục tiêu đã đề ra qua đề tài quản lý tuyển sinh bằng cách xây dựng giao diện dễ nhìn và thông tin được đồng bộ quản lý chặt chẽ. Với việc ứng dụng công nghệ thông tin trong trong tác quản lý tuyển sinh là một việc cần thiết nhằm năng cao chấp lượng của việc quản lý cũng như tính an toàn của thông tin.

## 4.2. Ưu điểm

Với đồ án này, nhóm thực hiện đã hoàn thành những chức năng sau:

- Chương trình được xây dựng với giao diện dễ nhìn, dễ sử dụng, thích hợp với nhiều đối tượng sử dụng bao gồm người quản lý và sinh viên.

- Hệ thống các câu lệnh hàm được xây dựng từ Sql datacript dễ thao tác và dễ hiểu.

- Có tính bảo mật ổn định, Việc sử dụng dụng tính toán thông qua sql giúp xét kết quả nhanh hơn.

- Lập trình giúp quản lý, tra cứu và lấy thông tin từ cơ sở dữ liệu dễ dàng.

## 4.3. Nhược điểm

- Dữ liệu dễ bị chỉnh sửa, tính bảo mật kém.

- Có nhiều hàm chức năng đã học nhưng chưa áp dụng triệt để.

- Chưa đồng bộ được với các phần mềm khác.

## 4.4. Hướng phát triển

- Nâng cấp các chức năng và giao diện người dùng.

- Phát triển trên website khác.

- Tăng cường chế độ bảo mật hệ thống.

+ Bảo mật Server.

+ Bảo mật cơ sở dữ liệu trên SQL Server.

# BẢNG PHÂN CÔNG CÔNG VIỆC

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Công việc** | **Lê Quốc Nguyên Vương** | **Võ Thị**  **Kiều Diễm** |
| 1 | Thiết kế cơ sở dữ liệu | X | X |
| 2 | Viết class: NGANH, DANTOC, HOKHAU, KHUVUC |  | X |
| 3 | Viết class: DOITUONG, HOSOTHISINH, NGUYENVONG | X |  |
| 4 | Viết stored | X | X |
| 5 | Trigger | X |  |
| 6 | Function | X | X |
| 7 | Transaction, Index | X |  |
| 8 | Nhập liệu | X | X |
| 9 | Thiết Kế Form |  | X |
| 10 | Debug, thử nghiệm | X | X |
| 11 | Báo cáo | X | X |
| 12 | Thuyết trình | X | X |

Bảng 6: Phân công