6/26/2020

**XÂY DỰNG CHƯƠNG TRÌNH QUẢN LÝ ĐỀ TÀI SINH VIÊN**

TP. Hồ Chí Minh, ngày 26 tháng 6 năm 2020

Chuyên ngành Kỹ thuật dữ liệu

KHOA CÔNG NGHỆ THÔNG TIN

**NHÓM SINH VIÊN THỰC HIỆN ĐỀ TÀI:**

**Nguyễn Tuấn Phi 18133039**

**Trần Tiến Đức 18133007**

**Nguyễn Thị Như Quỳnh 18133044**

**GVHD: TS. Huỳnh Xuân Phụng**

**ĐIỂM SỐ**

|  |  |  |  |
| --- | --- | --- | --- |
| TIÊU CHÍ | NỘI DUNG | TRÌNH BÀY | TỔNG |
| ĐIỂM |  |  |  |

**NHẬN XÉT CỦA GIẢNG VIÊN**

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

Giáo viên hướng dẫn

**(ký và ghi rõ họ tên)**

**PHUNG**

**Huỳnh Xuân Phụng**

**LỜI CẢM ƠN**

Nhóm chúng em xin gửi lời cảm ơn đến thầy Huỳnh Xuân Phụng đã cung cấp cho chúng em những kiến thức hữu ích về bộ môn lập trình window. Và cũng đưa ra cách thức làm đồ án môn học để lấy điểm cuối kỳ nhằm đảm bảo cho mọi người vận dụng những kiến thức đã học để làm ra một sản phẩm thực tế. Trong quá trình làm thầy cũng nhiệt tình giải đáp các thắc mắc đồng thời định hướng phát triển đồ án của nhóm chúng em. Đồng thời, nhóm chúng em cũng cảm ơn những thành viên trong nhóm đã nổ lực và hỗ trợ nhau hết mình để cùng nhau hoàn thành đồ án một cách tốt nhất.

TP. HCM, ngày 20 tháng 6 năm 2020

Nhóm sinh viên thực hiện

**MỤC LỤC**

[**1.** **Đặc tả chương trình quản lý đề tài sinh viên** 1](#_Toc44103021)

[***1.1.*** ***Quản lý thông tin sinh viên*** 1](#_Toc44103022)

[***1.2.*** ***Quản lý thông tin giảng viên*** 1](#_Toc44103023)

[***1.3.*** ***Quản lý thông tin đề tài*** 1](#_Toc44103024)

[**2.** **Phân công công việc** 1](#_Toc44103025)

[**3.** **Thiết kế** 2](#_Toc44103026)

[***3.1.*** ***Thuật toán*** 2](#_Toc44103027)

[***3.2.*** ***Thiết kế lớp*** 2](#_Toc44103028)

[***3.3.*** ***Thiết kế cơ sở dữ liệu*** 8](#_Toc44103029)

[***3.4.*** ***Bảng mô tả các field trong 1 table*** 9](#_Toc44103030)

[***3.5.*** ***Thiết kế giao diện*** 12](#_Toc44103031)

[**4.** **Cài đặt và kiểm thử** 19](#_Toc44103032)

[**5.** **Kết luận** 22](#_Toc44103033)

[***5.1.*** ***Kết luận*** 22](#_Toc44103034)

[***5.2.*** ***Hướng phát triển*** 23](#_Toc44103035)

**DANH MỤC CÁC BẢNG VÀ HÌNH ẢNH**

**Bảng 1: Phân công công việc và đóng góp của các thành viên trong nhóm….2**

**Bảng 2: Danh mục các lớp của sinh viên Nguyễn Tuấn Phi…………………..3**

**Bảng 3: Danh mục các phương thức của sinh viên Nguyễn Tuấn Phi………..3**

**Bảng 4: Danh mục các lớp của sinh viên Trần Tiến Đức……………………...5**

**Bảng 5: Danh mục các phương thức của sinh viên Trần Tiến Đức…………..5**

**Bảng 6: Danh mục các lớp của sinh viên Nguyễn Thị Như Quỳnh…………...6**

**Bảng 7: Danh mục các phương thức của sinh viên Nguyễn Thị Như Quỳnh...7**

**Hình 1: Quan hệ giữa các bảng trong cơ sở dữ liệu……………………………8**

**Bảng 8: Danh sách các bảng trong cơ sở dữ liệu……………………………….9**

**Bảng 9: Danh sách các trường trong bảng dbo.SinhVien……………………..9**

**Bảng 10: Danh sách các trường trong bảng dbo.GiangVien…………………10**

**Bảng 11: Danh sách các trường trong bảng dbo.ChuyenNganh……………..10**

**Bảng 12: Danh sách các trường trong bảng dbo.MonHoc……………………10**

**Bảng 13: Danh sách các trường trong bảng dbo.NhomSinhVien…………….11**

**Bảng 14: Danh sách các trường trong bảng dbo.Type………………………...11**

**Bảng 15: Danh sách các trường trong bảng dbo.DeAn………………………..11**

**Bảng 16: Danh sách các trường trong bảng dbo.TienDo……………………...12**

**Bảng 17: Đặc tả giao diện………………………………………………………..13**

**Bảng 18: Cài đặt và kiểm thử chương trình……………………………………19**

1. **Đặc tả chương trình quản lý đề tài sinh viên**
   1. ***Quản lý thông tin sinh viên***

Quản lý thông tin sinh viên là một trong những chức năng chính của chương trình. Với chức năng này, người sử dụng có thể tìm thấy thông tin của sinh viên, hoặc thực hiện tìm kiếm sinh viên theo nhiều tiêu chí như là: Mã số sinh viên, họ tên, chuyên ngành, …. Đồng thời còn cho phép người quản trị thêm một sinh viên vào trong danh sách, sửa thông tin sinh viên, xóa sinh viên ra khỏi danh sách.

Thông tin đầu vào của chức năng quản lý sinh viên là các tiêu chí cần tìm kiếm, và đầu ra sẽ là danh sách các sinh viên thỏa mãn các tiêu chí đó.

* 1. ***Quản lý thông tin giảng viên***

Quản lý thông tin giảng viên là một trong những chức năng chính của chương trình. Với chức năng này, người sử dụng có thể tìm thấy thông tin của giảng viên, hoặc thực hiện tìm kiếm giảng viên theo nhiều tiêu chí như là: mã số giảng viên, họ và tên, chuyên ngành, …. Đồng thời cũng cho phép người quản trị thêm một giảng viên vào trong danh sách, sửa thông tin giảng viên, xóa giảng viên ra khỏi danh sách.

Thông tin đầu vào của chức năng quản lý giảng viên là các tiêu chí cần tìm kiếm và đầu ra sẽ là danh sách các giảng viên thỏa mãn các tiêu chí đó.

* 1. ***Quản lý thông tin đề tài***

Quản lý thông tin đề tài là chức năng quan trọng nhất của chương trình này. Ở chức năng cho phép người dùng biết được thông tin của các đề tài đã được đăng ký (tên đề tài, nhóm sinh viên thực hiện, tiến độ, …). Và người dùng cũng có thể tìm kiếm đề tài theo nhiều tiêu chí như là: ID đề tài, tên đề tài, loại đề tài (đồ án môn học, tiểu luận chuyên ngành, đồ án tốt nghiệp). Ngoài ra giảng viên còn có thể cập nhật tiến độ của các buổi báo cáo thông qua giao diện tiến độ.

1. **Phân công công việc**

Bảng 1: Phân công công việc và đóng góp của các thành viên trong nhóm

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên sinh viên | Mô tả khái quát mảng công việc SV thực hiện trong đồ án | Ước tính phần trăm đóng góp |
| 1 | Nguyễn Tuấn Phi | Thực hiện các chức năng: quản lý đề tài, đăng ký nhóm, đăng ký đề tài, cập nhật tiến độ báo cáo | 34% |
| 2 | Trần Tiến Đức | Thực hiện các chức năng: tìm kiếm, thêm, xóa, sửa sinh viên | 33% |
| 3 | Nguyễn Thị Như Quỳnh | Thực hiện các chức năng: tìm kiếm, thêm, xóa, sửa giảng viên. | 33% |

1. **Thiết kế**
   1. ***Thuật toán***

Chương trình sử dụng chủ yếu phương pháp tìm kiếm dữ liệu trong cơ sở dữ liệu dựa trên truy vấn các thông tin đầu vào. Sử dụng kỹ thuật tách chuỗi và ghép chuỗi để lưu trữ và truy xuất các giá trị trong thuộc tính đa trị ở cơ sở dữ liệu (Ví dụ: danh sách thành viên nhóm, danh sách link tài liệu, …)

* 1. ***Thiết kế lớp***

*Bảng 2: Danh mục các lớp của sinh viên Nguyễn Tuấn Phi*

Sinh viên phụ trách: Nguyễn Tuấn Phi

|  |  |  |
| --- | --- | --- |
| TT | Tên lớp | Mục đích |
| 1 | DanhSachDeAn | Để chứa thông tin của các đề án, làm source cho datagridview ở form FrmQuanLyDeAn |
| 2 | TienDoBaoCao | Để chứa thông tin của các buổi báo cáo thuộc một đề án nào đó, làm source cho datagridview ở form FrmTienDo |

Bảng 3: Danh sách các phương thức của sinh viên Nguyễn Tuấn Phi

Sinh viên phụ trách: Nguyễn Tuấn Phi

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo |
|  |  |  |  |
| 1 | TachIDNhom(string IDNhom)  Input: IDNhom  Output: List các chuỗi được tách ra bởi dấu ‘-’ | Nhầm lấy ra các phần khác nhau của chuỗi IDNhom được lưu trong database. Phần đầu đại diện cho loại đồ án, phần 2 là đại diện cho bộ môn hoặc chuyên ngành, phần 3 là số thứ tự | DeAnController.cs  (11) |
| 2 | CreateIDNhom(string type, string subject, string stt)  Input: type, subject, stt  Output: Một chuỗi bao gồm 3 phần đại diện cho type, subject, stt được nối bởi ký tự ‘-’ | Tạo ID nhóm tự động | NhomController.cs  (13) |
| 3 | SoThuTuTiepTheo(int x)  Input: x  Output: số tiếp theo của x được thể hiện ở dạng string gồm 3 ký tự. Ví dụ: 003, 012, 134. | Tính số thứ tự tiếp theo. | NhomController.cs  (20) |
| 4 | GetListThanhVienNhom(string s)  Input: s  Output: List các sinh viên | Lấy ra danh sách sinh viên dựa trên việc tách mã số sinh viên từ chuỗi s | NhomController.cs  (30) |
| 5 | GetDanhSachThanhVien(string s)  Input: s  Output: List các chuỗi có dạng: <MSSV>: <Họ và tên> | Tạo ra một danh sách có dạng:  <MSSV>: <Họ và tên> để hiển thị lên datagridview | NhomController.cs  (47) |
| 6 | GetDanhSachNoiDungBaoCao(string s)  Input: s  Output: List các nội dung báo cáo của các buổi | Lấy ra danh sách các nội dung đã báo cáo trong các buổi báo cáo | BaoCaoTienDoController.cs  (11) |
| 7 | GetDanhSachLinkTaiLieu(string s)  Input: s  Output: List các link tài liệu của các buổi báo cáo | Lấy ra danh sách các link tài liệu đã sử dụng trong các buổi báo cáo | BaoCaoTienDoController.cs  (26) |
| 8 | GetHoanThanh(string s)  Input: s  Output: List các chuỗi true, false | Lấy ra danh sách các chuỗi true, false để biết các buổi đã báo cáo | BaoCaoTienDoController.cs  (41) |

Bảng 4: Danh mục các lớp của sinh viên Trần Tiến Đức

Sinh viên phụ trách: Trần Tiến Đức

|  |  |  |
| --- | --- | --- |
| TT | Tên lớp | Mục đích |
| 1 | StudentController | Là controller chứa các phương thức điều khiển đối tượng sinh viên trong database |
| 2 | SinhVienViewModel | Dùng để hiển thị trên DataGridView có trong form FrmSinhVien |

Bảng 5: Danh mục các phương thức của sinh viên Trần Tiến Đức

Sinh viên phụ trách: Trần Tiến Đức

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, dòng chứa khai báo |
| 1 | GetStudent(string studentID)  Input: studentID  Output: đối tượng sinh viên có mã số sinh viên là studentID | Để trích đối tượng có mã số sinh viên được cho trước | StudentController.cs (15) |
| 2 | AddStudent(SinhVien student)  Input: student  Output: True nếu thêm sinh viên thành công hoặc False nếu có lỗi xảy ra | Để thêm đối tượng sinh viên vào database | StudentController.cs (25) |
| 3 | UpdateStudent(SinhVien student)  Input: student  Output: True nếu cập nhật sinh viên thành công hoặc False nếu có lỗi xảy ra | Để cập nhật đối tượng sinh viên có sẵn trong database | StudentController.cs (34) |
| 4 | DeleteStudent(string studentID)  Input: studentID  Output: True nếu xoá sinh viên thành công hoặc False nếu có lỗi xảy ra | Để xoá sinh viên trong database | StudentController.cs (43) |

Bảng 6: Danh sách các lớp của sinh viên Nguyễn Thị Như Quỳnh

Sinh viên phụ trách: Nguyễn Thị Như Quỳnh

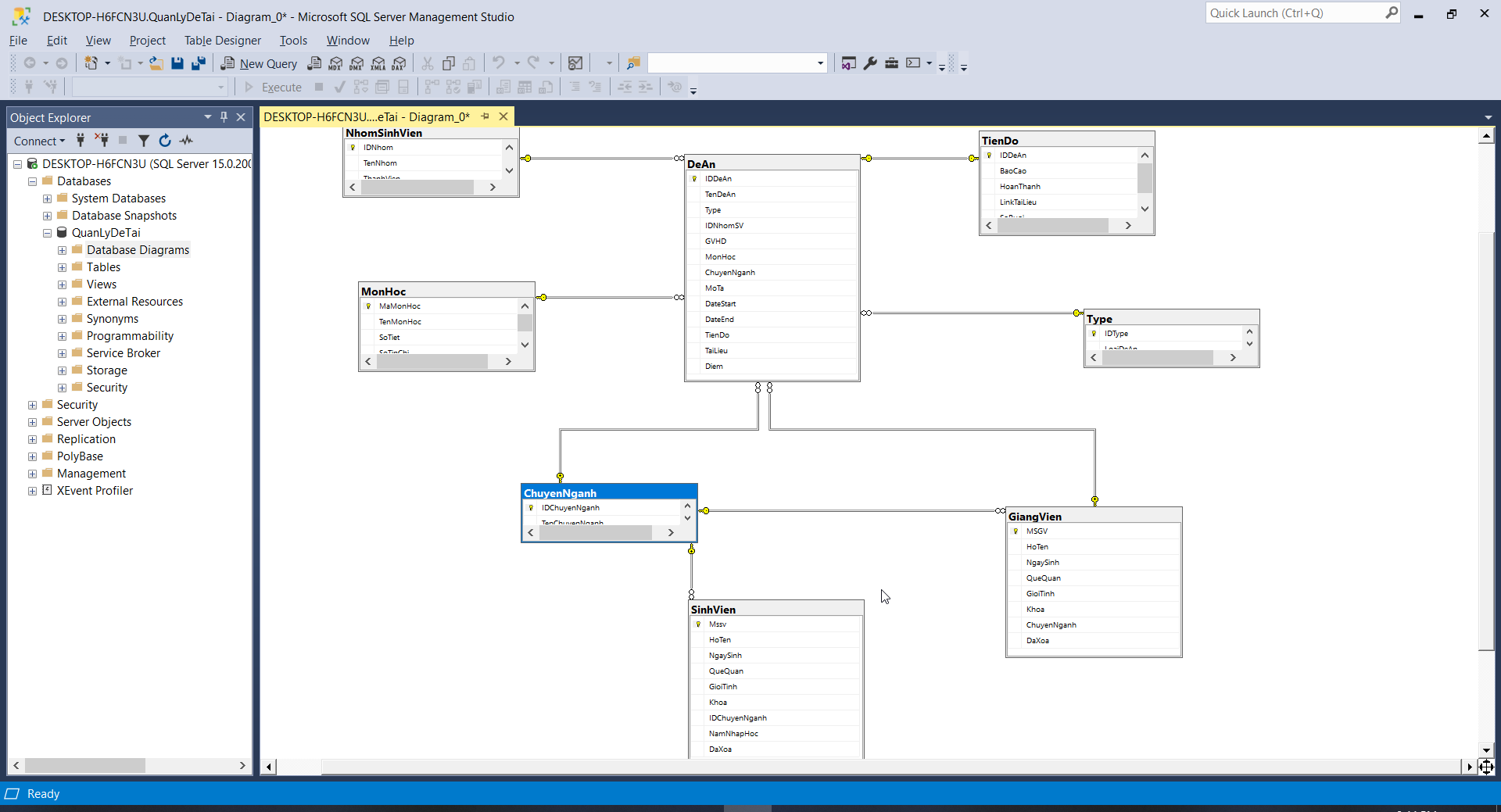
|  |  |  |
| --- | --- | --- |
| TT | Tên lớp | Mục đích |
| 1 | GiangVienController | Là controller chứa các phương thức điều khiển đối tượng giảng viên trong database |
| 2 | GiangVienViewModel | Dùng để hiển thị trên datagridview có trong form FrmGiangVien |
| 3 | ChuyenNganhController | Là controller chứa phương thức điều khiển đối tượng chuyên ngành trong database |
| 4 | ChuyenNganhViewModel | Dùng để hiển thị chuyên ngành lên combobox |
| 5 | AddDeAnThanhCongEventArgs | Là một class kế thừa một tham số sự kiện dùng để truyền dữ liệu. |
| 6 | DangKyNhomThanhCongEventArgs | Là một class kế thừa một tham số sự kiện dùng để truyền dữ liệu. |

Bảng 7: Danh sách các phương thức của sinh viên Nguyễn Thị Như Quỳnh

Sinh viên phụ trách: Nguyễn Thị Như Quỳnh

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, dòng chứa khai báo |
| 1 | GetAllChuyenNganh()  Input:  Output: list các chuyên ngành | Lấy ra IDChuyenNganh và TenChuyenNganh để bỏ vào combobox và hiển thị lên datagridview | ChuyenNganhController.cs (12) |
| 2 | GetGiangVien(string IDGiangVien)  Input: IDGiangVien  Output: đối tượng giảng viên có MSGV là IDGiangVien | Lấy ra giảng viên có MSGV cho trước | GiangVienController.cs (13) |
| 3 | GetAllGiangVien(string searchString)  Input: searchString  Output: list các giảng viên | Lấy ra list các giảng viên và hiển thị lên datagridview | GiangVienController.cs (25) |
| 4 | DeleteGiangVien(string ID)  Input: ID (IDGiangVien)  Output: đối tượng giảng viên có MSGV là ID | Xóa giảng viên có MSGV là ID | GiangVienController.cs (47) |

* 1. ***Thiết kế cơ sở dữ liệu***

******

Hình 1: Quan hệ giữa các bảng trong cơ sở dữ liệu

Bảng 8: Danh sách các bảng trong cơ sở dữ liệu

|  |  |  |
| --- | --- | --- |
| TT | Tên Bảng | Mục đích |
| 1 | dbo.SinhVien | Quản lý các thông tin của sinh viên |
| 2 | dbo.GiangVien | Quản lý các thông tin của giảng viên |
| 3 | dbo.ChuyenNganh | Quản lý các thông tin của các chuyên ngành |
| 4 | dbo.NhomSinhVien | Quản lý các thông tin của các nhóm sinh viên |
| 5 | dbo.MonHoc | Quản lý các thông tin của các môn học |
| 6 | dbo.Type | Quản lý các loại của đề án |
| 7 | dbo.DeAn | Quản lý các thông tin của các đề án |
| 8 | dbo.TienDo | Quản lý tiến độ báo cáo của đề án |

* 1. ***Bảng mô tả các field trong 1 table***

Bảng 9: Danh sách các trường trong bảng dbo.SinhVien

Table dbo.SinhVien

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | MSSV | char(8) | Lưu trữ mã số sinh viên của mỗi sinh viên. Phân biệt các sinh viên |
| 2 | HoTen | nvarchar(60) | Lưu trữ họ và tên của sinh viên |
| 3 | NgaySinh | date | Lưu trữ ngày sinh của sinh viên |
| 4 | QueQuan | nvarchar(MAX) | Lưu trữ quê quán của sinh viên |
| 5 | GioiTinh | nvarchar(5) | Lưu trữ giới tính của sinh viên |
| 6 | Khoa | nvarchar(30) | Lưu trữ khoa của sinh viên |
| 7 | IDChuyenNganh | char(4) | Lưu trữ ID chuyên ngành của sinh viên để có thể truy vấn biết chuyên ngành |
| 8 | NamNhapHoc | char(4) | Lưu trữ năm nhập học của sinh viên |

Bảng 10: Danh sách các trường trong bảng dbo.GiangVien

Table dbo.GiangVien

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | MSGV | char(4) | Lưu trữ mã số giảng viên của mỗi giảng viên. Phân biệt các giảng viên |
| 2 | HoTen | nvarchar(60) | Lưu trữ họ và tên của giảng viên |
| 3 | NgaySinh | date | Lưu trữ ngày sinh của giảng viên |
| 4 | QueQuan | nvarchar(50) | Lưu trữ quê quán của giảng viên |
| 5 | GioiTinh | nvarchar(5) | Lưu trữ giới tính của giảng viên |
| 6 | Khoa | nvarchar(50) | Lưu trữ khoa của giảng viên |
| 7 | IDChuyenNganh | char(4) | Lưu trữ ID chuyên ngành của giảng viên để có thể truy vấn biết chuyên ngành |

Bảng 11: Danh sách các trường trong bảng dbo.ChuyenNganh

Table dbo.ChuyenNganh

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | IDChuyenNganh | char(4) | Lưu trữ ID chuyên ngành, phân biệt giữa các chuyên ngành |
| 2 | TenChuyenNganh | nvarchar(70) | Lưu trữ tên chuyên ngành |

Bảng 12: Danh sách các trường trong bảng dbo.MonHoc

Table dbo.MonHoc

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | MaMonHoc | char(10) | Lưu trữ mã môn học. Phân biệt giữa các môn học |
| 2 | TenMonHoc | nvarchar(50) | Lưu trữ tên môn học |
| 3 | SoTiet | int | Lưu trữ số tiết của môn học |
| 4 | SoTinChi | int | Lưu trữ số tín chỉ của môn học |

Bảng 13: Danh sách các trường trong bảng dbo.NhomSinhVien

Table dbo.NhomSinhVien

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | IDNhom | varchar(15) | Lưu trữ ID nhóm. Phân biệt giữa các nhóm |
| 2 | TenNhom | nvarchar(200) | Lưu trữ tên nhóm |
| 3 | ThanhVien | varchar(200) | Lưu trữ chuỗi các mã số sinh viên của các thành viên trong nhóm được nối bởi dấu ‘,’ |

Bảng 14: Danh sách các trường trong bảng dbo.Type

Table dbo.Type

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | IDType | char(1) | Lưu trữ ID loại đề án. Phân biệt giữa các loại đề án |
| 2 | LoaiDeAn | nvarchar(100) | Tên của loại đề án |

Bảng 15: Danh sách các trường trong bảng dbo.DeAn

Table dbo.DeAn

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | IDDeAn | varchar(15) | Lưu trữ ID đề án. Phân biệt giữa các đề án |
| 2 | TenDeAn | nvarchar(200) | Lưu trữ tên đề án |
| 3 | Type | char(1) | Lưu trữ ID của loại đề án |
| 4 | IDNhomSV | varchar(15) | Lưu trữ ID nhóm sinh viên |
| 5 | GVHD | char(4) | Lưu trữ ID giảng viên hướng dẫn |
| 6 | MonHoc | char(10) | Lưu trữ ID môn học |
| 7 | ChuyenNganh | char(4) | Lưu trữ ID chuyên ngành |
| 8 | MoTa | nvarchar(MAX) | Lưu trữ mô tả đề án |
| 9 | DateStart | datetime | Lưu trữ ngày bắt đầu đề án |
| 10 | DateEnd | datetime | Lưu trữ ngày kết thúc đề án |
| 11 | TienDo | float | Lưu trữ tiến độ hoàn thành đề án |
| 12 | TaiLieu | nvarchar(200) | Lưu trữ chuỗi cái link tài liệu sử dụng khi báo cáo |
| 13 | Diem | varchar(10) | Lưu trữ điểm của đề án |

Bảng 16: Danh sách các trường trong bảng dbo.TienDo

Table dbo.TienDo

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | IDDeAn | varchar(15) | Lưu trữ ID đề án. Phân biệt giữa các tiến độ đề án |
| 2 | BaoCao | nvarchar(MAX) | Lưu trữ chuỗi nội dung báo cáo |
| 3 | HoanThanh | nvarchar(MAX) | Lưu trữ chuỗi các true, false để cho biết buổi báo cáo đã hoàn thành chưa |
| 4 | LinkTaiLieu | varchar(MAX) | Lưu trữ chuỗi các link tài liệu |
| 5 | SoBuoi | int | Lưu trữ số buổi báo cáo |

* 1. ***Thiết kế giao diện***

Bảng 17: Đặc tả giao diện

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Màn hình/Cửa sổ/Dialog | Mục đích | Giải thích ngắn gọn các quyết định thiết kế khi thiết kế màn hình |
| 1 | Form Trang chủ | Quản lý toàn bộ các form quản lý sinh viên, giảng viên và đề án | Nguyễn Thị Như Quỳnh |
| 2 | Form quản lý sinh viên | Quản lý sinh viên | Trần Tiến Đức  Form được thiết kế theo tiêu chí dễ dung, dễ tiếp cận và phải bao quát hết các chức năng cần thiết. |
| 3 | Form thêm sinh viên | Thêm sinh viên vào cơ sở dữ liệu | Trần Tiến Đức  Form được thiết kế có rang buộc để nhập đủ các thông tin cần thiết. |
| 4 | Form cập nhật thông tin sinh viên | Cập nhật thông tin sinh viên trong cơ sở dữ liệu | Trần Tiến Đức  Form được thiết kế để có thể cập nhật thông tin một cách dễ dàng. |
| 5 | Form quản lý giảng viên | Quản lý toàn bộ giảng viên trong cơ sở dữ liệu | Nguyễn Thị Như Quỳnh  Form được thiết kế với hướng thiết kế tương tự form quản lý sinh viên. |
| 6 | Form thêm giảng viên | Thêm giảng viên vào cơ sở dữ liệu | Nguyễn Thị Như Quỳnh  Tương tự form them sinh viên. |
| 7 | Form cập nhật thông tin giảng viên | Cập nhật thông tin giảng viên trong cơ sở dữ liệu | Nguyễn Thị Như Quỳnh  Tương tự form cập nhật thông tin sinh viên. |
| 8 | Form quản lý đề án | Quản lý các đề án của sinh viên | Nguyễn Tuấn Phi  Thiết kế trực quan, dễ quan sát, dễ dùng. |
| 9 | Form đăng ký nhóm | Đăng ký nhóm làm đề án | Nguyễn Tuấn Phi  Thiết kế trực quan, dễ sử dụng. |
| 10 | Form đăng ký đề án | Đăng ký đề án vào cơ sở dữ liệu | Nguyễn Tuấn Phi  Tương tự form đăng ký nhóm sinh viên. |
| 11 | Form quản lý tiến độ | Quản lý và tính toán tiến độ của đề án dựa theo số buổi báo cáo và kết quả báo cáo | Nguyễn Tuấn Phi  Tương tự form quản lý đề án. |

1. **Cài đặt và kiểm thử**

Bảng 18: Cài đặt và kiểm thử chương trình

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tình huống | Mục đích | Giải thích ngắn gọn quyết định lựa chọn tình huống và dữ liệu đầu vào |
| 1 | Tình huống 1: thêm 1 sinh viên  Dữ liệu vào: một sinh viên với thông tin của sinh viên đó.  Kết quả dự kiến: hiện đúng thông tin lên datagridview. | Thêm thành công thông tin của sinh viên và hiển thị thông tin lên datagridview | Thử xem có thêm được sinh viên thành công hay không |
| 2 | Tình huống 2: sửa 1 sinh viên  Dữ liệu vào: một sinh viên với thông tin của sinh viên đó.  Kết quả dự kiến: hiện đúng thông tin lên datagridview. | Sửa thành công thông tin của sinh viên và hiển thị thông tin lên datagridview | Thử xem có sửa được sinh viên thành công hay không |
| 3 | Tình huống 3: xóa 1 sinh viên  Dữ liệu vào: một sinh viên.  Kết quả dự kiến: sinh viên đó không còn hiển thị trên datagridview | Xóa thành công sinh viên đó hay không | Thử xem có xóa được sinh viên thành công hay không |
| 4 | Tình huống 4: tìm kiếm sinh viên  Dữ liệu vào: thông tin bất kì (mssv, họ tên,…)  Kết quả dự kiến: tìm kiếm ra được sinh viên với những thông tin tìm kiếm đúng và hiển thị lên datagridview | Tìm kiếm sinh viên thành công hay không | Thử xem có tìm kiếm được sinh viên hay không. |
| 5 | Tình huống 5: thêm 1 giảng viên  Dữ liệu vào: một giảng viên với thông tin của giảng viên đó.  Kết quả dự kiến: hiện đúng thông tin lên datagridview. | Thêm thành công thông tin của giảng viên và hiển thị thông tin lên datagridview | Thử xem có thêm được giảng viên thành công hay không |
| 6 | Tình huống 6: sửa 1 giảng viên  Dữ liệu vào: một giảng viên với thông tin của giảng viên đó.  Kết quả dự kiến: hiện đúng thông tin lên datagridview. | Sửa thành công thông tin của giảng viên và hiển thị thông tin lên datagridview | Thử xem có sửa được giảng viên thành công hay không |
| 7 | Tình huống 7: xóa 1 giảng viên  Dữ liệu vào: một giảng viên  Kết quả dự kiến: giảng viên đó không còn hiển thị trên datagridview | Xóa thành công giảng viên đó hay không | Thử xem có xóa được giảng viên thành công hay không |
| 8 | Tình huống 8: tìm kiếm giảng viên  Dữ liệu vào: thông tin bất kì (msgv, họ tên,…)  Kết quả dự kiến: tìm kiếm ra được giảng viên với những thông tin tìm kiếm đúng và hiển thị lên datagridview | Tìm kiếm giảng viên thành công hay không | Thử xem có tìm kiếm được giảng viên hay không. |
| 9 | Tình huống 9: tìm kiếm đề án  Dữ liệu vào: thông tin bất kì (ID đề án, tên đề án,…)  Kết quả dự kiến: tìm kiếm ra được dự án với những thông tin tìm kiếm đúng và hiển thị lên datagridview | Tìm kiếm đề án thành công hay không | Thử xem có tìm kiếm được đề án hay không |
| 10 | Tình huống 10: đăng ký nhóm  Dữ liệu vào: thông tin của một nhóm  Kết quả dự kiến: đăng ký nhóm thành công, hiển thị ID nhóm lên form đăng ký đề án | Đăng ký nhóm thành công hay không | Thử xem có đăng ký nhóm được hay không |
| 11 | Tình huống 11: đăng ký đề án  Dữ liệu vào: thông tin của một đề án  Kết quả dự kiến: đăng ký đề án thành công và hiển thị lên datagridview | Đăng ký đề án thành công và hiển thị lên datagridview | Thử xem có đăng ký đề án được hay không |
| 12 | Tình huống 12: cập nhật tiến độ  Dữ liệu vào: đề án cần cập nhật  Kết quả dự kiến: cập nhật tiến độ của đề án và hiển thị lên datagridview | Cập nhật tiến độ đề án và hiển thị lên datagridview | Thử xem có cập nhật được tiến độ hay không |

1. **Kết luận**
2. ***Kết luận***

Về cơ bản, nhóm tự nhận xét phần mềm của nhóm đã giải quyết được 89% yêu cầu mà đồ án đặt ra. Sau đây là những ưu nhược điểm của phần mềm.

* Ưu điểm:
* Giao diện gọn gàng, dễ tiếp cận.
* Khả năng xử lý nhanh.
* Chương trình tốn khá ít tài nguyên hệ thống để hoạt động.
* Chương trình hoạt động ổn định, hiếm khi xảy ra tình trạng crash trong quá trình hoạt động.
* Nhược điểm:
* Giao diện chương trình chưa được đẹp.
* Do được thiết kế theo mô hình Database – First, chương trình cần có database được thiết lập sẵn trước khi sử dụng.
* Còn thiếu một số ràng buộc.

1. ***Hướng phát triển***

* Bổ sung ràng buộc cho các lớp xử lý dữ liệu.
* Cải thiện giao diện người dùng.
* Phân lớp, cấp quyền cho người dùng sử dụng các chức năng nhất định trong quyền hạn của mình.