UBND TỈNH LÂM ĐỒNG
TRƯỜNG CAO ĐẲNG ĐÀ LẠT

GIÁO TRÌNH

MÔ ĐUN: XÂY DỰNG PHẦN MỀM QUẢN LÝ NHÂN SỰ

NGÀNH/NGHỀ: CÔNG NGHỆ THÔNG TIN (ƯDPM)

TRÌNH ĐỘ: CAO ĐẲNG

(Ban hành kèm theo Quyết định số: /QĐ-CĐNĐL ngày …tháng…năm… của Hiệu trưởng Trường Cao đẳng Đà Lạt)

LƯU HÀNH NỘI BỘ

Lâm Đồng, năm 2017

TUYÊN BỐ BẢN QUYỀN

Tài liệu này thuộc loại sách giáo trình nên các nguồn thông tin có thể được phép dùng nguyên bản hoặc trích dùng cho các mục đích về đào tạo và tham khảo.

Mọi mục đích khác mang tính lệch lạc hoặc sử dụng với mục đích kinh doanh thiếu lành mạnh sẽ bị nghiêm cấm.

LỜI GIỚI THIỆU

Hiện nay, trong nước cũng như trên thế giới đã có rất nhiều phần mềm quản lý nhân sự hỗ trợ công tác tổ chức, tuy nhiên các phần mềm quản lý này không thể đáp ứng đầy đủ nhu cầu của tất cả các đơn vị có những đặc thù khác nhau. Đối với phần mềm nước ngoài thì khó sử dụng cho người Việt có trình độ tin học không chuyên và giá thành lại cao. Đối với phần mềm trong nước chưa có sản phẩm phần mềm nào chuyên để hỗ trợ công tác tổ chức trong nhiều cơ quan, doanh nghiệp một cách đầy đủ nhất.

Mục tiêu của mô đun này là xây dựng một phần mề quản lý thông tin nhân viên, cán bộ với các chức năng: Phân quyền quản lý, Quản lý năng lực, Theo dõi diễn biến lương và tăng lương, Quản lý hồ sơ đi học, Quản lý khen thưởng – kỹ luật, Quản lý quá trình bổ nhiệm – miễn nhiệm.

Đà Lạt, ngày 07 tháng 7 năm 2017

Tham gia biên soạn

Chủ biên: Ths. Phạm Đình Nam

MỤC LỤC

TRANG

LỜI GIỚI THIỆU
BÀI 1: THIẾT KẾ CƠ SỞ DỮ LIỆU 1
1. Khảo sát và phân tích bài toán quản lý nhân sự 1
1.1 Xác định luồng dữ liệu 1
1.2. Mô tả ứng dụng 2
1.2.1 Chức năng hệ thống 2
1.2.2 Chức năng quản lý danh mục 2
1.2.3 Chức năng quản lý nhân sự 3
1.2.4 Chức năng quản lý chuyên môn, nghiệp vụ, quan hệ 4
1.2.5 Chức năng quản lý diễn biến lương 5
1.2.6 Chức năng thống kê báo cáo 5
1.2.7 Chức năng trợ giúp 5
2. Lựa chọn hệ quản trị cơ sở dữ liệu 6
2.1 Phân tích ưu điểm, hạn chế của các hệ quản trị cơ sở dữ liệu mạnh nhất hiện nay 6
2.1.1 MySQL 6
2.1.2. SQL Server 6
2.2 Những điểm mạnh của hệ quản trị cơ sở dữ liệu được chọn (SQL Server) 7
3. Thiết kế cơ sở dữ liệu Quản lý nhân sự 7
3.1 Xác định các bảng 7
3.2 Xác định các thuộc tính của từng bảng 8
4. Thiết kế các Table trong CSDL 9
4.1 Xác định kiểu dữ liệu từng trường của bảng 9
4.2 Tạo bảng bằng công cụ SQL Management Studio 9
5. Tạo liên kết cho các Table trong CSDL 18
5.1 Xác định khóa chính, khóa ngoại 18
5.2 Thiết lập quan hệ cho các bảng bằng giao diện SQL Management Studio 18
6. Nhập dữ liệu mẫu cho các Table 21
6.1 Nhập dữ liệu mẫu cho các bảng bên quan hệ 1 21
6.2 Nhập dữ liệu mẫu cho các bảng bên quan hệ n 22

7. Cài đặt cơ sở dữ liệu Quản lý nhân sự 24
7.1 Tạo user đăng nhập cơ sở dữ liệu 24
7.2 Cấp quyền user sử dụng cơ sở dữ liệu 25
7.3 Lưu trữ và quản lý cơ sở dữ liệu quản lý nhân sự 26
BÀI 2: THIẾT KẾ CÁC CHỨC NĂNG CHÍNH 28
1. Tạo project và kết nối cơ sở dữ liệu 28
1.1 Thiết lập chuỗi kết nối trong Properties của project 28
1.2 Khai báo các đối tượng dùng chung ADO.NET 28
1.3 Khởi tạo kết nối đến cơ sở dữ liệu 29
2. Thiết kế chức năng quản lý hồ sơ 30
2.1. Chức năng Cập nhật dữ liệu cho phòng ban 30
2.2. Chức năng Cập nhật dữ liệu cho Đơn vị/Chức vụ 31
2.3 Chức năng Cập nhật dữ liệu cho Ngành nghề, ngạch viên chức 31
2.4 Chức năng Cập nhật dữ liệu cho nhân viên 32
2.5. Chức năng Cập nhật dữ liệu cho quan hệ gia đình nhân viên 34
3 Thiết kế chức năng quản lý lương 35
3.1 Chức năng Chấm công 35
3.2 Chức năng Tính lương 36
4 Thiết kế chức năng Tra cứu - Tìm kiếm thông tin 37
4.1. Tra cứu hồ sơ lý lịch 37
4.2. Tra cứu theo lương 39
5. Thiết kế chức năng Báo cáo - Thống kê 40
5.1. Thống kê theo các tiêu chí về hồ sơ 40
5.2. Thống kê về lương 42
BÀI 3: THIẾT KẾ GIAO DIỆN CHƯƠNG TRÌNH 43
1. Tạo form đăng nhập với người dùng 43
1.1 Tạo bảng lưu thông tin đăng nhập 43
1.2 Thiết kế giao diện đăng nhập 45
1.3 Viết lệnh đăng nhập 45
2. Tạo form giao diện chính của chương trình 47
2.1 Thiết lập các thuộc tính của form chính 47
2.2 Thiết kế hệ thống menu, toolbar 47

2.2.1 Thiết kế hệ thống menu 47
2.2.2 Thiết kế hệ thống toolbar 47
3. Tạo các liên kết đến các chức năng trong chương trình 47
3.1 Chức năng mở form 47
3.2 Chức năng trợ giúp 48
3.3 Chức năng thiết lập hệ thống 48
BÀI 4: THIẾT KẾ CÁC BÁO CÁO 49
1. Xác định chức năng của báo cáo 49
1.1 Chức năng in danh sách 49
1.2 Chức năng in thống kê, tổng hợp 49
2. Xác định, tạo nguồn dữ liệu cho báo cáo 50
2.1 Xây dựng nguồn dữ liệu ảo để thiết kế báo cáo 50
2.2. Xác định nguồn dữ liệu thật để báo cáo 50
3. Tạo báo cáo với công cụ 50
3.1 Sử dụng công cụ Report Wizard để tạo báo cáo 50
3.2 Thêm tiêu đề cho báo cáo 53
4. Chỉnh sửa các thành phần, thuộc tính trong báo cáo 54
4.1 Định dạng ngày trong báo cáo 54
4.2 Kẽ viền cho báo cáo 54
5. Tạo form truyền tham số 54
6. Xử lý tham số với báo cáo 58
7. Thiết kế các báo cáo chi tiết 59
7.1 Tạo thủ tục SQL để kết xuất dữ liệu 59
7.2 Tạo báo cáo sử dụng nguồn dữ liệu từ thủ tục SQL 62
BÀI 5: HOÀN THIỆN, ĐÓNG GÓI PHẦN MỀM 64
1. Kiểm thử phần mềm 64
2. Hiệu chỉnh phần mềm 65
2.1 Xác định nguyên nhân gây lỗi 65
2.2 Khắc phục lỗi 65
3. Xác định công cụ đóng gói 66
4. Đặt bảo mật cho cơ sở dữ liệu 66
4.1 Xây dựng thư viện mã hóa 66

4.2 Đặt mật khẩu cho cơ sở dữ liệu và code. 66
5. Đóng gói phần mềm..... 67

GIÁO TRÌNH MÔ ĐUN

Tên mô đun: Xây dựng phần mềm quản lý nhân sự

Mã mô đun: MĐ30

Vị trí, tính chất, ý nghĩa và vai trò của mô đun:

- Vị trí: Mô đun này được học sau các mô đun Quản trị cơ sở dữ liệu với Access, Quản trị cơ sở dữ liệu với SQL Server, Lập trình windows 2.

- Tính chất: Là mô đun chuyên môn tự chọn, yêu cầu sinh viên phải sử dụng thành thạo công cụ lập trình và thiết kế cơ sở dữ liệu, cần trang bị cho sinh viên kỹ năng về quản trị cơ sở dữ liệu SQL Server và lập trình windows 2 (ADO.NET) để xây dựng phần mềm trên môi trường .NET.

- Ý nghĩa và vai trò của mô đun: Tài liệu này được thiết kế theo từng mô đun/ môn học thuộc hệ thống mô đun/môn học của một chương trình, để đào tạo hoàn chỉnh nghề Công nghệ thông tin (ứng dụng phần mềm) ở trình độ Trung cấp và Cao đẳng, đồng thời được dùng làm Giáo trình cho học viên trong các khoá đào tạo ngắn hạn hoặc cho các công nhân kỹ thuật, các nhà quản lý và người sử dụng nhân lực tham khảo.

Mục tiêu của mô đun:

Về kiến thức:

- Phân tích một số yêu cầu của bài toán thực tiễn để xây dựng được mô hình cơ sở dữ liệu;
- Trình bày được tính năng của các công cụ để xây dựng cơ sở dữ liệu vật lý;

Về kỹ năng:

- Xây dựng được cơ sở dữ liệu quản lý hồ sơ nhân sự cho đơn vị, doanh nghiệp;
- Xây dựng được phần mềm để quản lý nhân sự , với các tính năng quan trọng như cập nhật danh sách các phòng ban, nhân viên, lý lịch của từng nhân viên;

Về năng lực tự chủ và trách nhiệm:

- Có khả năng tự nghiên cứu, tự học, tham khảo tài liệu liên quan đến môn học để vận dụng vào hoạt động hoc tập.
- Vận dụng được các kiến thức tự nghiên cứu, học tập và kiến thức, kỹ năng đã được học để hoàn thiện các kỹ năng liên quan đến môn học một cách khoa học, đúng quy định.

Nội dung của mô đun:

BÀI 1: THIẾT KẾ CƠ SỞ DỮ LIỆU

Mã Bài: MĐ30_01

Giới thiệu:

Việc thiết kế cơ sở dữ liệu đã được các bạn thực hiện qua các mô đun như Quản trị CSDL với Microsoft Access, Quản trị CSDL với Microsoft SQL Server cũng như đã được học qua môn học phân tích và thiết kế hệ thống. Trong bài này chúng tôi nói về việc thiết kế cơ sở dữ liệu trên SQL Server.

Mục tiêu:

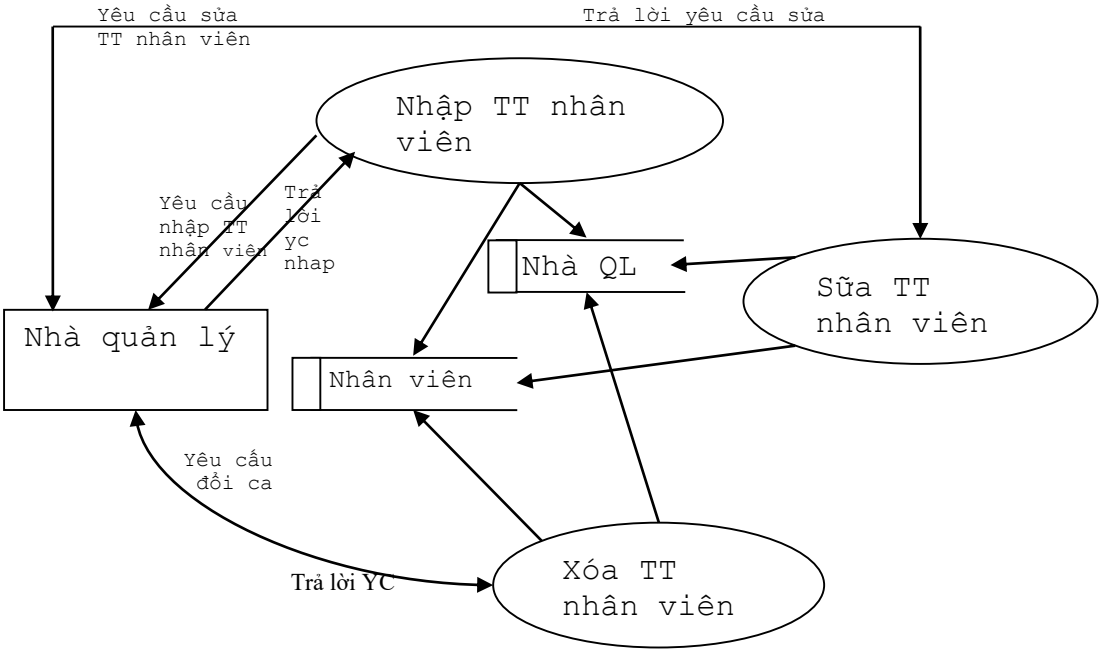
- Khảo sát và tiếp cận được bài toán quản lý nhân sự;
- Phân tích được các đặc trưng của bài toán quản lý nhân sự;
- Thiết kế được CSDL quản lý nhân sự bằng Microsoft Access hoặc SQL Server;
- Thiết kế được các bảng và tạo quan hệ giữa các bảng trong CSDL;
- Nhập được dữ liệu mẫu cho các bảng;
- Tích cực, chủ động trong việc tạo và cài đặt cơ sở dữ liệu quản lý nhân sự.

Nội dung chính:

1. Khảo sát và phân tích bài toán quản lý nhân sự

1.1 Xác định luồng dữ liệu

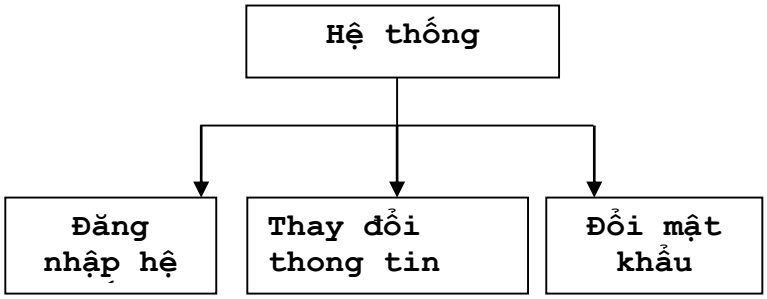
Mô hình dưới đỉnh của quản lý thông tin Nhân viên



1.2. Mô tả ứng dụng

1.2.1 Chức năng hệ thống

-Hệ thống yêu cầu đăng nhập trước khi người quản trị (hoặc nhân viên quản lý nhân sự) muốn quản lý và sử dụng. Các chức năng quản lý tương ứng với quyền hạn của người đăng nhập (quản lý người dùng).



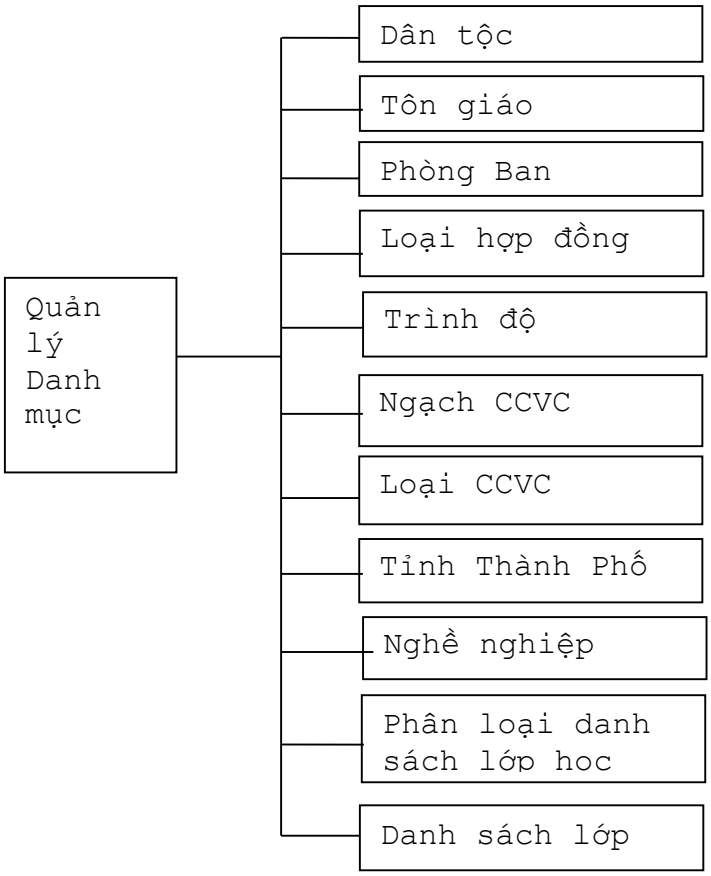
Hình 1.1

Ở đây người quản trị có thể thay đổi mật khẩu của chương trình để đảm bảo tính bảo mật của chương trình.

1.2.2 Chức năng quản lý danh mục

-Chức năng Quản lý danh mục chỉ được sử dụng bởi:

-Người quản trị hoặc nhân viên phòng nhân sự sử dụng để nhập, sửa, xoá, thay đổi về thông tin nhân viên. Các danh mục gồm có trình độ, dân tộc, tôn giáo, tỉnh thành phố, khoa/phòng ban, nghề nghiệp, phân loại danh sách lớp học, danh sách lớp học, loại hợp đồng, ngạch CCVC, loại CCVC.



Hình 1.2

1.2.3 Chức năng quản lý nhân sự

- Ở chức năng này người trực tiếp quản lý đó là nhân viên phòng nhân sự thực hiện được những yêu cầu sau:

+ Cập nhật tất cả các thông tin của nhân viên trong công ty như (Họ Tên, Quê quán, năm sinh, Điện thoại, chức vụ…) bổ sung vào danh sách nhân viên trong công ty

+ Nhập bộ phận của từng nhân viên vào phòng ban cụ thể nào đó

+ Sửa chi tiêt thông tin nhân viên: thực hiện việc chỉnh sửa, cập nhật bổ sung thêm các thông tin chi tiêt của từng nhân viên.

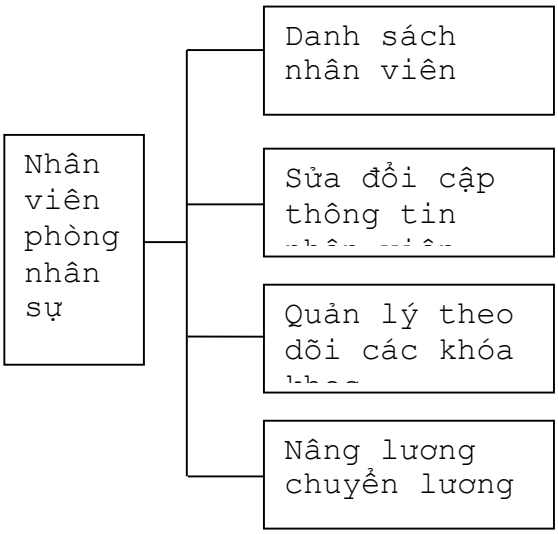
+Lập phân loại, thực hiện việc cập nhật bổ sung, chỉnh sửa phân loại nhân viên vào từng phòng ban cụ thể.

+Xoá thông tin: thực hiện việc xoá thông tin khi nhân viên nghỉ việc hoặc chyển đi nơi khác…

+ Quản lý danh sách nhân viên đi học.

+ Nâng lương nhân viên khi đến thời hạn.

+ Báo cáo cấp trên

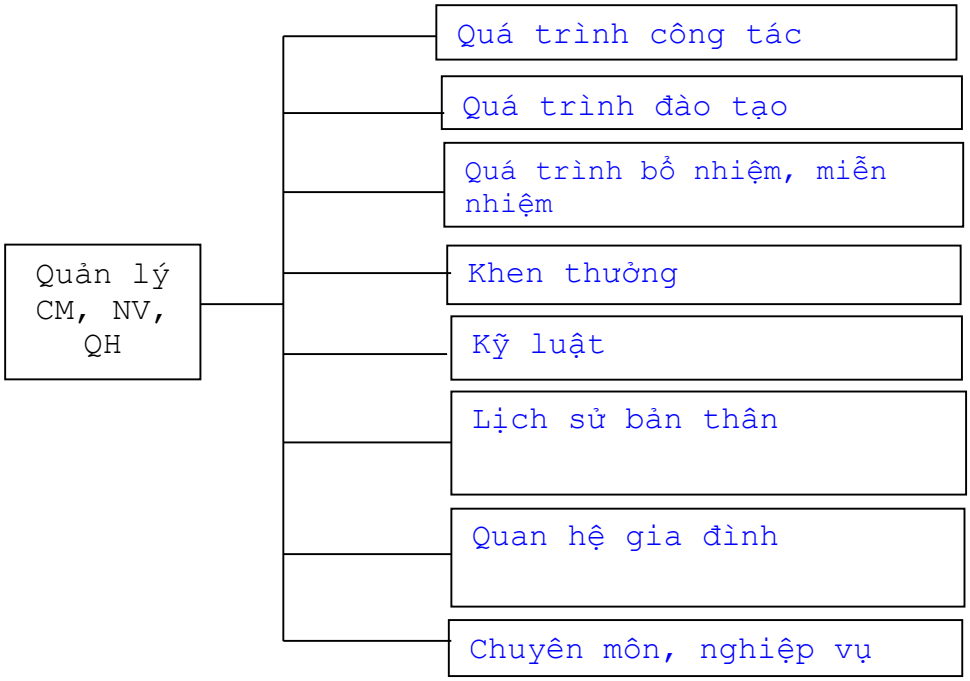


Hình 1.3

1.2.4 Chức năng quản lý chuyên môn, nghiệp vụ, quan hệ

-Chức năng này sẽ quản lý diễn biến của nhân viên như: quá trình công tác, quá trình đào tạo, quá trình diễn biến lương, quá trình khen thưởng-kỷ luật, lịch sử bản thân, quan hệ gia đình và trình độ chuyên môn nghiệp vụ của nhân viên.

+Cho phép người quản lý thêm thông tin mới, chỉnh sửa các thông tin và xoá bỏ các thông tin không cần thiết về chi tiết nhân viên. Thông qua chức năng này giúp cho việc quản lý dễ giàng hơn.

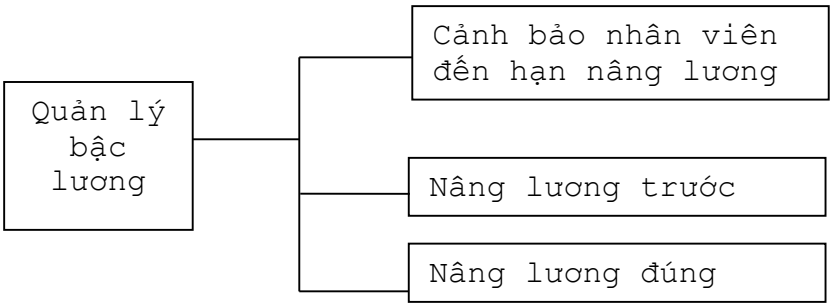


Hình 1.4

1.2.5 Chức năng quản lý diễn biến lương

+ Chức năng này sẽ được nhân viên phòng nhân sự theo dõi diễn biến lương của nhân viên trong đơn vị ở bất kỳ thời điểm nào.

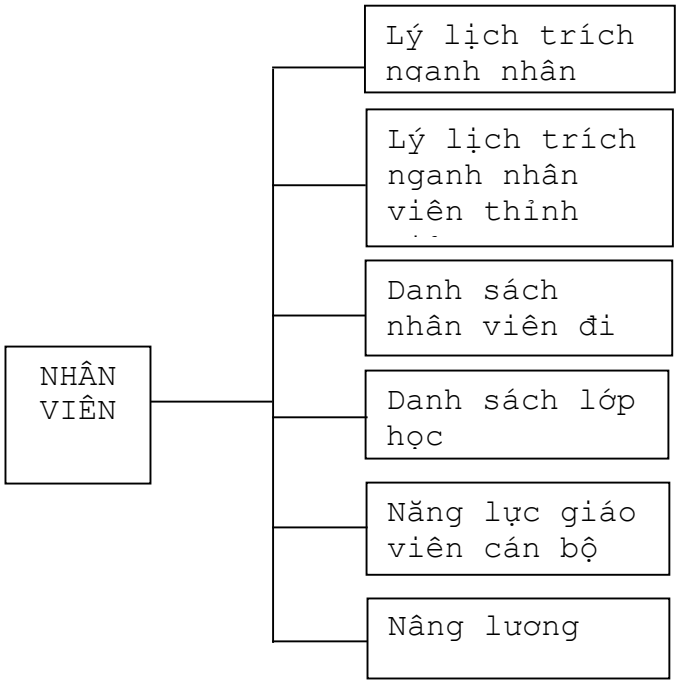
+ Cập nhật xem có nhân viên nào được khen thưởng hay bị kỷ luật, …



Hình 1.5

1.2.6 Chức năng thống kê báo cáo

Báo cáo theo định kỳ hoặc theo yêu cầu của cấp trên các thông tin của nhân viên như: Khen thưởng, hợp đồng, đào tạo, thôi việc….



Hình 1.6

1.2.7 Chức năng trợ giúp

-Chức năng này hướng dẫn sử dụng cho người dùng sử dụng được phần mềm của mình một cách hiệu quả nhất. khi chúng ta sử dụng phần mềm gặp khó khăn hay trở ngại gì thì một trong những biện pháp tốt nhất là chúng ta cần đến một sự

giúp đỡ nào đó.Vì vậychức năng trợ giúp là một phân không thể thiếu trong mỗi phần mềm.

2. Lựa chọn hệ quản trị cơ sở dữ liệu

2.1 Phân tích ưu điểm, hạn chế của các hệ quản trị cơ sở dữ liệu mạnh nhất hiện nay

2.1.1 MySQL

MySQL là một trong những cơ sở dữ liệu có khả năng mở rộng phổ biến nhất hiện nay. Nó giàu các tính năng, là một sản phẩm mã nguồn mở mạnh mẽ trên các website và các ứng dụng online. Việc bắt đầu với MySQL là cực kì dễ dàng và các nhà phát triển dễ dàng tiếp cận với một lượng lớn các thông tin về cơ sở dữ liệu trên internet.

*** Ưu điểm của MySQL**

Dễ dàng sử dụng: MySQL có thể dễ dàng cài đặt. Với các công cụ bên thứ 3 làm cho nó càng dễ đơn giản hơn để có thể sử dụng.

Giàu tính năng: MySQL hỗ trợ rất nhiều chức năng SQL được mong chờ từ một hệ quản trị cơ sở dữ liệu quan hệ-cả trực tiếp lẫn gián tiếp.

Bảo mật: Có rất nhiều tính năng bảo mật, một số ở cấp cao đều được xây dựng trong MySQL.

Khả năng mở rộng và mạnh mẽ: MySQL có thể xử lý rất nhiều dữ liệu và hơn thế nữa nó có thể được mở rộng nếu cần thiết.

Nhanh: Việc đưa ra một số tiêu chuẩn cho phép MySQL để làm việc rất hiệu quả và tiết kiệm chi phí, do đó nó làm tăng tốc độ thực thi.

*** Nhược điểm của MySQL**

Giới hạn: Theo thiết kế, MySQL không có ý định làm tất cả và nó đi kèm với các hạn chế về chức năng mà một vào ứng dụng có thể cần.

Độ tin cậy: Cách các chức năng cụ thể được xử lý với MySQL (ví dụ tài liệu tham khảo, các giao dịch, kiểm toán,...) làm cho nó kém tin cậy hơn so với một số hệ quản trị cơ sở dữ liệu quan hệ khác.

Sự phát triển chậm: Mặc dù MySQL vẫn là một sản phẩm công nghệ mã nguồn mở phổ biến nhưng có những phàn nàn liên quan đến quá trình phát triển kể từ khi nó được mua lại.

2.1.2. SQL Server

* Ưu điểm của SQL Server

Tích hợp chặt chẽ trọn bộ hệ thống và công cụ phát triển phần mề, đặc biệt ở mảng .NET. MSSQL còn hỗ trợ XML trực tiếp trong DB.

MS SQL Server có tính bảo mật cao, hệ thống xác thực người dùng của MSSQL cũng qua mặt MySQL.

Khả năng phục hồi (sau khi bị crash) của MSSQL tốt hơn MySQL. MSSQL cung cấp nhiều phương thức nhân bản nên việc phục hồi dữ liệu trên MSSQL dễ dàng hơn.

*** Nhược điểm của SQL Server**

Tính về mặt hiệu suất, có rất nhiều thử nghiệm và tường trình cho rằng MSSQL perform kém hơn MySQL nhiều mặt. MSSQL đòi hỏi tài nguyên rất lớn (CPU mạnh, nhiều RAM...), nếu không nó rất ì ạch. MySQL không đòi hỏi nhiều như MSSQL. MySQL có thể chạy trên các UNIX highend system và perform tốt hơn MSSQL trên Windows highend server trong nhiều trường hợp.

MSSQL là hệ quản trị có bản quyền, phải trả gần một ngàn rưỡi đô cho 1 license MSSQL standard và khi cần support, bạn phải trả thêm tiền support (tùy case). MS vẫn cung cấp bản MSSQL không thu phí dành cho mục đích development. Tài liệu về MSSQL cũng rất nhiều trên mạng.

2.2 Những điểm mạnh của hệ quản trị cơ sở dữ liệu được chọn (SQL Server)

Microsoft SQL Server 2008 là một hệ thống quản lý cơ sở dữ liệu sử dụng truy vấn SQL để trao đổi dữ liệu thực hiện các câu kết nối và truy vấn vào bên trong dữ liệu nhằm lưu trữ dữ liệu một cách an toàn hơn. Một hệ thống quản trị cơ sở dữ liệu bao gồm Databases, database engine và các ứng dụng dùng để quản lý dữ liệu. SQL Server 2008 được tối ưu để có thể chạy trên môi trường cơ sở dữ liệu rất lớn (Very Large Database Environment) lên đến Tera-Byte và có thể phục vụ cùng lúc cho hàng ngàn user. SQL Server 2008 có thể kết hợp "ăn ý" với các server khác như Microsoft Internet Information Server (IIS), E-Commerce Server, Proxy Server....SQL Server 2008 có rất nhiều phiên bản để cho từng đối tượng sử dụng. Hiện nay SQL Server đã phát triển lên đến phiên bản Microsoft SQL Server 2008, nhưngchúng tôi sử dụng phiên bản 2008 là vì cơ sở dữ liệu của phần mềm chúng tôi không lớn và do giới hạn về cấu hình máy, giá trị kinh tế….

3. Thiết kế cơ sở dữ liệu Quản lý nhân sự

3.1 Xác định các bảng

Đối với việc phân tích thông tin của hệ thống quản lý nhân sự là cán bộ, giáo viên, chúng tôi xác định các bảng dữ liệu sau:

tblDanToc
tblHangThuongBinh
tblHocVi
tblLoaiCCVC
tblLoaiChuongTrinh
tblLoaiHopDong
tblLopHoc
tblLopHoc_NguoiThamGia
tblNgachCCVC
tblNgheNghiep
tblNhanVien
tblNhanVien_ChiTietHopDong
tblNhanVien_DienBienLuong
tblNhanVien_LichSuBanThan
tblNhanVien_QTCongTac
tblNhanVien_QTDaoTao
tblNhanVien_QTKhenThuong
tblNhanVien_QTKyLuat
tblNhanVien_QuanHeGiaDinh
tblNhanVien_TrinhDo
tblPhongBan
tblTinh
tblToChuc
tblTonGiao
tblTrinhDo

Với mỗi bảng cần chỉ rõ:

+ Khoá chính (có thể gồm một hay nhiều thuộc tính).

+ Mô tả của tất cả các cột (trường).

3.2 Xác định các thuộc tính của từng bảng

Việc xác định thuộc tính của từng bảng phụ thuộc vào loại nhân viên hoặc cán bộ và đặt thù của cơ quan, doanh nghiệp, với ví dụ về cơ sở dữ liệu quản lý nhân sự chúng ta có thể liệt kê các thuộc tính của bảng tblNhanVien như sau:

NhanVienID	int
MaPhongBan	nvarchar(10)
HoTen	nvarchar(50)
TenGoiKhac	nvarchar(50)
NgaySinh	datetime
GioiTinh	bit
HinhAnh	image
NoiSinhXa	nvarchar(50)
NoiSinhHuyen	nvarchar(50)
NoiSinhTinh	nvarchar(50)
QueQuanXa	nvarchar(50)
QueQuanHuyen	nvarchar(50)
QueQuanTinh	nvarchar(50)
DanToc	nvarchar(20)
TonGiao	nvarchar(20)
HoKhauThuongTru	nvarchar(100)
DiaChiHienNay	nvarchar(100)
NgheNghiepKhiDuocTD	nvarchar(50)
NgayTuyenDung	datetime
CoQuanTuyenDung	nvarchar(100)
ChucVuHienNay	nvarchar(200)
NgayDuocBoNhiem	datetime

Và một số thuộc tính khác cần thiết cho việc in lý lịch đầy đủ của nhân viên. Các thuộc tính của đầy đủ cơ sở dữ liệu sẽ được chúng tôi trình bày ở nội dung sau.

4. Thiết kế các Table trong CSDL

4.1 Xác định kiểu dữ liệu từng trường của bảng

Với mỗi cột (trường) cần phải có:

+ Một tên duy nhất (trong bảng lưu giữ nó).

+ Một mô tả ngắn gọn.

+ Một kiểu dữ liệu (ví dụ: integer, char, date, logical,…) phụ thuộc vào hệ quản trị cơ sở dữ liệu cụ thể cài đặt cơ sở dữ liệu)

+ Một kích thước (mặc định hay chỉ rõ tuỳ từng kiểu dữ liệu)

4.2 Tạo bảng bằng công cụ SQL Management Studio

Với việc xác định các bảng như trên, chúng ta có thể sử dụng công cụ SQL Management Studio để tạo bảng như sau:

```
/****** Object:  Table [dbo].[tblNgheNghiep]     ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
```

```
GO
CREATE TABLE [dbo].[tblNgheNghiep](
	[NgheNghiepID] [int] IDENTITY(1,1) NOT NULL,
	[NgheNghiep] [nvarchar](50) NULL,
 CONSTRAINT [PK_tblNgheNghiep] PRIMARY KEY CLUSTERED
(
	[NgheNghiepID] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/****** Object:  Table [dbo].[tblLoaiHopDong]     ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblLoaiHopDong](
	[MaLoaiHopDong] [int] NOT NULL,
	[TenLoaiHopDong] [nvarchar](50) NULL,
	[MoTa] [nvarchar](50) NULL,
	[KieuHopDong] [int] NULL,
 CONSTRAINT [PK_tblLoaiHopDong] PRIMARY KEY CLUSTERED
(
	[MaLoaiHopDong] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/****** Object:  Table [dbo].[tblLoaiChuongTrinh]     ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblLoaiChuongTrinh](
	[MaLoaiChuongTrinh] [int] NOT NULL,
	[TenChuongTrinh] [nvarchar](200) NULL,
	[GhiChu] [nvarchar](50) NULL,
 CONSTRAINT [PK_tblLoaiChuongTrinh] PRIMARY KEY CLUSTERED
(
	[MaLoaiChuongTrinh] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/****** Object:  Table [dbo].[tblLoaiCCVC]     ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblLoaiCCVC](
	[MaLoaiCCVC] [nvarchar](10) NOT NULL,
	[TenLoaiCCVC] [nvarchar](50) NULL,
	[SoBac] [int] NULL,
	[SoNamNangBac] [int] NULL,
	[Bac01] [float] NULL,
	[Bac02] [float] NULL,
	[Bac03] [float] NULL,
	[Bac04] [float] NULL,
	[Bac05] [float] NULL,
	[Bac06] [float] NULL,
	[Bac07] [float] NULL,
	[Bac08] [float] NULL,
```

```
	[Bac09] [float] NULL,
	[Bac10] [float] NULL,
	[Bac11] [float] NULL,
	[Bac12] [float] NULL,
	[Bac13] [float] NULL,
	[Bac14] [float] NULL,
	[Bac15] [float] NULL,
	[Bac16] [float] NULL,
 CONSTRAINT [PK_tblLoaiCCVC] PRIMARY KEY CLUSTERED
(
	[MaLoaiCCVC] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/****** Object:  Table [dbo].[tblHocVi]     ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblHocVi](
	[MaHocVi] [int] NOT NULL,
	[TenHocVi] [nvarchar](50) NULL,
 CONSTRAINT [PK_tblHocVi] PRIMARY KEY CLUSTERED
(
	[MaHocVi] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/****** Object:  Table [dbo].[tblHangThuongBinh]     ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblHangThuongBinh](
	[HangThuongBinhID] [int] IDENTITY(1,1) NOT NULL,
	[HangThuongBinh] [nvarchar](50) NULL,
 CONSTRAINT [PK_tblHangThuongBinh] PRIMARY KEY CLUSTERED
(
	[HangThuongBinhID] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/****** Object:  Table [dbo].[tblDanToc]    Script Date: 07/16/2017 20:45:30
******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblDanToc](
	[DanTocID] [int] IDENTITY(1,1) NOT NULL,
	[DanToc] [nvarchar](20) NULL,
 CONSTRAINT [PK_tblDanToc] PRIMARY KEY CLUSTERED
(
	[DanTocID] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/****** Object:  Table [dbo].[tblTrinhDo]     ******/
```

```
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblTrinhDo](
	[MaTrinhDo] [nvarchar](10) NOT NULL,
	[TenTrinhDo] [nvarchar](50) NULL,
 CONSTRAINT [PK_tblTrinhDo] PRIMARY KEY CLUSTERED
(
	[MaTrinhDo] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/****** Object:  Table [dbo].[tblTonGiao]     ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblTonGiao](
	[TonGiaoID] [int] IDENTITY(1,1) NOT NULL,
	[TonGiao] [nvarchar](50) NULL,
 CONSTRAINT [PK_tblTonGiao] PRIMARY KEY CLUSTERED
(
	[TonGiaoID] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/****** Object:  Table [dbo].[tblToChuc]     ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblToChuc](
	[MaToChuc] [int] NOT NULL,
	[TenToChuc] [nvarchar](200) NULL,
	[DiaChi] [nvarchar](100) NULL,
	[DienThoai] [nvarchar](20) NULL,
 CONSTRAINT [PK_tblToChuc] PRIMARY KEY CLUSTERED
(
	[MaToChuc] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/****** Object:  Table [dbo].[tblTinh]     ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblTinh](
	[MaTinh] [nvarchar](5) NOT NULL,
	[TenTinh] [nvarchar](50) NULL,
 CONSTRAINT [PK_tblTinh] PRIMARY KEY CLUSTERED
(
	[MaTinh] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/****** Object:  Table [dbo].[tblPhongBan]     ******/
```

```
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblPhongBan](
	[MaPhongBan] [nvarchar](10) NOT NULL,
	[TenPhongBan] [nvarchar](100) NULL,
	[DienThoai] [nvarchar](50) NULL,
	[Fax] [nvarchar](50) NULL,
	[Email] [nvarchar](50) NULL,
	[MaToChuc] [int] NULL,
 CONSTRAINT [PK_tblPhongBan] PRIMARY KEY CLUSTERED
(
	[MaPhongBan] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/****** Object:  Table [dbo].[tblLopHoc]     ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblLopHoc](
	[LopHocID] [int] IDENTITY(1,1) NOT NULL,
	[MaLop] [nvarchar](50) NULL,
	[TenLop] [nvarchar](100) NULL,
	[CoSoDaoTao] [nvarchar](200) NULL,
	[MaLoaiChuongTrinh] [int] NULL,
 CONSTRAINT [PK_tblLopHoc] PRIMARY KEY CLUSTERED
(
	[LopHocID] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/****** Object:  Table [dbo].[tblNgachCCVC]     ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblNgachCCVC](
	[MaNgachCCVC] [nvarchar](10) NOT NULL,
	[TenNgachCCVC] [nvarchar](50) NULL,
	[MaLoaiCCVC] [nvarchar](10) NOT NULL,
 CONSTRAINT [PK_tblHeSoLuong_1] PRIMARY KEY CLUSTERED
(
	[MaNgachCCVC] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/****** Object:  Table [dbo].[tblLopHoc_NguoiThamGia]     ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblLopHoc_NguoiThamGia](
	[STT] [int] NOT NULL,
	[LopHocID] [int] NOT NULL,
	[ThoiGian] [nvarchar](50) NULL,
	[NguoiThamGia] [nvarchar](50) NULL,
```

```
	[ChucDanh] [nvarchar](50) NULL,
	[NamThamGia] [datetime] NULL,
	[GhiChu] [nvarchar](100) NULL,
 CONSTRAINT [PK_tblLopHoc_NguoiThamGia] PRIMARY KEY CLUSTERED
(
	[STT] ASC,
	[LopHocID] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/****** Object:  Table [dbo].[tblNhanVien]     ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblNhanVien](
	[NhanVienID] [int] IDENTITY(1,1) NOT NULL,
	[MaPhongBan] [nvarchar](10) NULL,
	[HoTen] [nvarchar](50) NULL,
	[TenGoiKhac] [nvarchar](50) NULL,
	[NgaySinh] [datetime] NULL,
	[GioiTinh] [bit] NULL,
	[HinhAnh] [image] NULL,
	[NoiSinhXa] [nvarchar](50) NULL,
	[NoiSinhHuyen] [nvarchar](50) NULL,
	[NoiSinhTinh] [nvarchar](50) NULL,
	[QueQuanXa] [nvarchar](50) NULL,
	[QueQuanHuyen] [nvarchar](50) NULL,
	[QueQuanTinh] [nvarchar](50) NULL,
	[DanToc] [nvarchar](20) NULL,
	[TonGiao] [nvarchar](20) NULL,
	[HoKhauThuongTru] [nvarchar](100) NULL,
	[DiaChiHienNay] [nvarchar](100) NULL,
	[NgheNghiepKhiDuocTD] [nvarchar](50) NULL,
	[NgayTuyenDung] [datetime] NULL,
	[CoQuanTuyenDung] [nvarchar](100) NULL,
	[ChucVuHienNay] [nvarchar](200) NULL,
	[NgayDuocBoNhiem] [datetime] NULL,
	[CongViecChinhDuocGiao] [nvarchar](100) NULL,
	[MaNgachCCVC] [nvarchar](10) NULL,
	[BacLuong] [int] NULL,
	[VuotKhung] [float] NULL,
	[NgayHuong] [datetime] NULL,
	[NgayNangLuongLanSau] [datetime] NULL,
	[NgayNangLuongTruocThoiHan] [datetime] NULL,
	[PhuCapChucVu] [float] NULL,
	[PhuCapKhac] [float] NULL,
	[LoaiHopDong] [int] NULL,
	[NgayKyHopDong] [datetime] NULL,
	[NgayThoiViec] [datetime] NULL,
	[ThamNienCongTac] [datetime] NULL,
	[NgayVaoCoQuanHienNay] [datetime] NULL,
	[NgayVaoDang] [datetime] NULL,
	[NgayVaoDangChinhThuc] [datetime] NULL,
	[NgayVaoDoan] [datetime] NULL,
	[NgayNhapNgu] [datetime] NULL,
	[NgayXuatNgu] [datetime] NULL,
	[QuanHamCaoNhat] [nvarchar](50) NULL,
	[DanhHieuDuocPhongTang] [nvarchar](100) NULL,
	[SoTruongCongTac] [nvarchar](100) NULL,
	[TinhTrangSucKhoe] [nvarchar](50) NULL,
```

```
	[ChieuCao] [nvarchar](50) NULL,
	[CanNang] [nvarchar](50) NULL,
	[NhomMau] [nvarchar](50) NULL,
	[LaThuongBinhHang] [nvarchar](50) NULL,
	[LaConGiaDinhChinhSach] [nvarchar](50) NULL,
	[SoCMND] [nvarchar](20) NULL,
	[NgayCap] [datetime] NULL,
	[NoiCap] [nvarchar](50) NULL,
	[SoBHXH] [nvarchar](20) NULL,
	[NhanXetCuaCoQuan] [nvarchar](1000) NULL,
	[LastEditUser] [nvarchar](50) NULL,
 CONSTRAINT [PK_tblNhanVien] PRIMARY KEY CLUSTERED
(
	[NhanVienID] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO
/****** Object:  Table [dbo].[tblNhanVien_TrinhDo]     ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblNhanVien_TrinhDo](
	[NhanVienID] [int] NOT NULL,
	[STT] [int] NOT NULL,
	[MaTrinhDo] [nvarchar](10) NULL,
	[MaHocVi] [int] NULL,
	[TenChungChi] [nvarchar](100) NULL,
	[QuyetDinhChungChi] [image] NULL,
	[LaCaoNhat] [smallint] NULL,
 CONSTRAINT [PK_tblNhanVien_TrinhDo_1] PRIMARY KEY CLUSTERED
(
	[NhanVienID] ASC,
	[STT] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO
/****** Object:  Table [dbo].[tblNhanVien_QuanHeGiaDinh]     ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblNhanVien_QuanHeGiaDinh](
	[NhanVienID] [int] NOT NULL,
	[STT] [int] NOT NULL,
	[LoaiQuanHe] [int] NULL,
	[MoiQuanHe] [nvarchar](20) NULL,
	[HoTenQuanHe] [nvarchar](50) NULL,
	[NamSinhQuanHe] [datetime] NULL,
	[CongViec] [nvarchar](1000) NULL,
 CONSTRAINT [PK_tblNhanVien_QuanHeGiaDinh_1] PRIMARY KEY CLUSTERED
(
	[NhanVienID] ASC,
	[STT] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/****** Object:  Table [dbo].[tblNhanVien_QTKyLuat]     ******/
SET ANSI_NULLS ON
```

```
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblNhanVien_QTKyLuat](
	[NhanVienID] [int] NOT NULL,
	[STT] [int] NOT NULL,
	[NgayQD] [datetime] NULL,
	[LyDo] [nvarchar](100) NULL,
	[CapQuyetDinh] [nvarchar](50) NULL,
	[LaCaoNhat] [smallint] NULL,
	[QuyetDinhKyLuat] [image] NULL,
 CONSTRAINT [PK_tblNhanVien_QTKyLuat] PRIMARY KEY CLUSTERED
(
	[NhanVienID] ASC,
	[STT] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO
/****** Object:  Table [dbo].[tblNhanVien_QTKhenThuong]     ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblNhanVien_QTKhenThuong](
	[NhanVienID] [int] NOT NULL,
	[STT] [int] NOT NULL,
	[NgayQD] [datetime] NULL,
	[LyDo] [nvarchar](100) NULL,
	[CapQuyetDinh] [nvarchar](50) NULL,
	[LaCaoNhat] [smallint] NULL,
	[QuyetDinhKhenThuong] [image] NULL,
 CONSTRAINT [PK_tblNhanVien_QTKhenThuong] PRIMARY KEY CLUSTERED
(
	[NhanVienID] ASC,
	[STT] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO
/****** Object:  Table [dbo].[tblNhanVien_QTDaoTao]     ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblNhanVien_QTDaoTao](
	[NhanVienID] [int] NOT NULL,
	[STT] [int] NOT NULL,
	[TenTruong] [nvarchar](100) NULL,
	[ChuyenNganhDaoTao] [nvarchar](100) NULL,
	[TuNgay] [datetime] NULL,
	[DenNgay] [datetime] NULL,
	[HinhThucDaoTao] [nvarchar](50) NULL,
	[LoaiVanBang] [nvarchar](50) NULL,
	[LaCaoNhat] [smallint] NULL,
	[QuyetDinhChungNhan] [image] NULL,
 CONSTRAINT [PK_tblNhanVien_QTDaoTao] PRIMARY KEY CLUSTERED
(
	[NhanVienID] ASC,
	[STT] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
```

```
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO
/****** Object:  Table [dbo].[tblNhanVien_QTCongTac]     ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblNhanVien_QTCongTac](
	[NhanVienID] [int] NOT NULL,
	[STT] [int] NOT NULL,
	[TuNgay] [datetime] NULL,
	[DenNgay] [datetime] NULL,
	[ChucDanh] [nvarchar](200) NULL,
	[LaCaoNhat] [smallint] NULL,
	[QuyetDinhBoNhiem] [image] NULL,
 CONSTRAINT [PK_tblNhanVien_QTCongTac] PRIMARY KEY CLUSTERED
(
	[NhanVienID] ASC,
	[STT] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO
/****** Object:  Table [dbo].[tblNhanVien_LichSuBanThan]     ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblNhanVien_LichSuBanThan](
	[NhanVienID] [int] NOT NULL,
	[STT] [int] NOT NULL,
	[LoaiQuanHe] [int] NULL,
	[TuNgay] [datetime] NULL,
	[DenNgay] [datetime] NULL,
	[CongViec] [nvarchar](1000) NULL,
 CONSTRAINT [PK_tblNhanVien_LichSuBanThan] PRIMARY KEY CLUSTERED
(
	[NhanVienID] ASC,
	[STT] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/****** Object:  Table [dbo].[tblNhanVien_DienBienLuong]     ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblNhanVien_DienBienLuong](
	[NhanVienID] [int] NOT NULL,
	[STT] [int] NOT NULL,
	[NgayHuong] [datetime] NULL,
	[MaNgachCCVC] [nvarchar](10) NULL,
	[BacLuong] [int] NULL,
	[HeSoLuong] [float] NULL,
	[QuyetDinhNangLuong] [image] NULL,
 CONSTRAINT [PK_tblNhanVien_DienBienLuong_1] PRIMARY KEY CLUSTERED
(
	[NhanVienID] ASC,
	[STT] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
```

```
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO
/****** Object:  Table [dbo].[tblNhanVien_ChiTietHopDong]     ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[tblNhanVien_ChiTietHopDong](
      [NhanVienID] [int] NOT NULL,
      [STT] [int] NOT NULL,
      [MonGiangDay] [nvarchar](100) NULL,
      [LopGiangDay] [nvarchar](100) NULL,
      [SoTiet] [int] NULL,
      [ThoiGianBatDau] [datetime] NULL,
      [ThoiGianKetThuc] [datetime] NULL,
 CONSTRAINT [PK_tblNhanVien_ChiTietHopDong] PRIMARY KEY CLUSTERED
(
      [NhanVienID] ASC,
      [STT] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
```

5. Tạo liên kết cho các Table trong CSDL

5.1 Xác định khóa chính, khóa ngoại

Các quan hệ giữa các bảng sẽ được thiết lập bởi các khoá ngoài, kiểm tra để các ràng buộc khoá ngoài (nếu có) sẽ được cài đặt.

Phải đảm bảo rằng giữu trường khóa chính và khóa ngoại phải cùng kiểu dữ liệu và kích thước thì chúng ta mới thiết lập quan hệ cho bước sau được.

5.2 Thiết lập quan hệ cho các bảng bằng giao diện SQL Management Studio

```
/****** Object:  Default [DF_tblNhanVien_VuotKhung]     ******/
ALTER TABLE [dbo].[tblNhanVien] ADD  CONSTRAINT [DF_tblNhanVien_VuotKhung]
DEFAULT ((0)) FOR [VuotKhung]
GO
/****** Object:  Default [DF_tblNhanVien_TrinhDo_LaCaoNhat]     ******/
ALTER TABLE [dbo].[tblNhanVien_TrinhDo] ADD  CONSTRAINT
[DF_tblNhanVien_TrinhDo_LaCaoNhat]  DEFAULT ((0)) FOR [LaCaoNhat]
GO
/****** Object:  Default [DF_tblPhongBan_MaToChuc]     ******/
ALTER TABLE [dbo].[tblPhongBan] ADD  CONSTRAINT [DF_tblPhongBan_MaToChuc]
DEFAULT ((1)) FOR [MaToChuc]
GO
/****** Object:  ForeignKey [FK_tblLopHoc_tblLoaiChuongTrinh]     ******/
ALTER TABLE [dbo].[tblLopHoc]  WITH CHECK ADD  CONSTRAINT
[FK_tblLopHoc_tblLoaiChuongTrinh] FOREIGN KEY([MaLoaiChuongTrinh])
REFERENCES [dbo].[tblLoaiChuongTrinh] ([MaLoaiChuongTrinh])
ON UPDATE CASCADE
ON DELETE CASCADE
GO
ALTER TABLE [dbo].[tblLopHoc] CHECK CONSTRAINT
[FK_tblLopHoc_tblLoaiChuongTrinh]
GO
/****** Object:  ForeignKey [FK_tblLopHoc_NguoiThamGia_tblLopHoc]     ******/
ALTER TABLE [dbo].[tblLopHoc_NguoiThamGia]  WITH CHECK ADD  CONSTRAINT
[FK_tblLopHoc_NguoiThamGia_tblLopHoc] FOREIGN KEY([LopHocID])
```

```
REFERENCES [dbo].[tblLopHoc] ([LopHocID])
ON UPDATE CASCADE
ON DELETE CASCADE
GO
ALTER TABLE [dbo].[tblLopHoc_NguoiThamGia] CHECK CONSTRAINT
[FK_tblLopHoc_NguoiThamGia_tblLopHoc]
GO
/****** Object:  ForeignKey [FK_tblNgachCCVC_tblLoaiCCVC]     ******/
ALTER TABLE [dbo].[tblNgachCCVC]  WITH CHECK ADD  CONSTRAINT
[FK_tblNgachCCVC_tblLoaiCCVC] FOREIGN KEY([MaLoaiCCVC])
REFERENCES [dbo].[tblLoaiCCVC] ([MaLoaiCCVC])
ON UPDATE CASCADE
ON DELETE CASCADE
GO
ALTER TABLE [dbo].[tblNgachCCVC] CHECK CONSTRAINT
[FK_tblNgachCCVC_tblLoaiCCVC]
GO
/****** Object:  ForeignKey [FK_tblNhanVien_tblLoaiHopDong]     ******/
ALTER TABLE [dbo].[tblNhanVien]  WITH CHECK ADD  CONSTRAINT
[FK_tblNhanVien_tblLoaiHopDong] FOREIGN KEY([LoaiHopDong])
REFERENCES [dbo].[tblLoaiHopDong] ([MaLoaiHopDong])
ON UPDATE CASCADE
GO
ALTER TABLE [dbo].[tblNhanVien] CHECK CONSTRAINT
[FK_tblNhanVien_tblLoaiHopDong]
GO
/****** Object:  ForeignKey [FK_tblNhanVien_tblNgachCCVC]     ******/
ALTER TABLE [dbo].[tblNhanVien]  WITH CHECK ADD  CONSTRAINT
[FK_tblNhanVien_tblNgachCCVC] FOREIGN KEY([MaNgachCCVC])
REFERENCES [dbo].[tblNgachCCVC] ([MaNgachCCVC])
ON UPDATE CASCADE
GO
ALTER TABLE [dbo].[tblNhanVien] CHECK CONSTRAINT
[FK_tblNhanVien_tblNgachCCVC]
GO
/****** Object:  ForeignKey [FK_tblNhanVien_tblPhongBan]     ******/
ALTER TABLE [dbo].[tblNhanVien]  WITH CHECK ADD  CONSTRAINT
[FK_tblNhanVien_tblPhongBan] FOREIGN KEY([MaPhongBan])
REFERENCES [dbo].[tblPhongBan] ([MaPhongBan])
ON UPDATE CASCADE
GO
ALTER TABLE [dbo].[tblNhanVien] CHECK CONSTRAINT [FK_tblNhanVien_tblPhongBan]
GO
/****** Object:  ForeignKey [FK_tblNhanVien_ChiTietHopDong_tblNhanVien]
******/
ALTER TABLE [dbo].[tblNhanVien_ChiTietHopDong]  WITH CHECK ADD  CONSTRAINT
[FK_tblNhanVien_ChiTietHopDong_tblNhanVien] FOREIGN KEY([NhanVienID])
REFERENCES [dbo].[tblNhanVien] ([NhanVienID])
ON UPDATE CASCADE
ON DELETE CASCADE
GO
ALTER TABLE [dbo].[tblNhanVien_ChiTietHopDong] CHECK CONSTRAINT
[FK_tblNhanVien_ChiTietHopDong_tblNhanVien]
GO
/****** Object:  ForeignKey [FK_tblNhanVien_DienBienLuong_tblNhanVien]
******/
ALTER TABLE [dbo].[tblNhanVien_DienBienLuong]  WITH CHECK ADD  CONSTRAINT
[FK_tblNhanVien_DienBienLuong_tblNhanVien] FOREIGN KEY([NhanVienID])
REFERENCES [dbo].[tblNhanVien] ([NhanVienID])
ON UPDATE CASCADE
ON DELETE CASCADE
GO
```

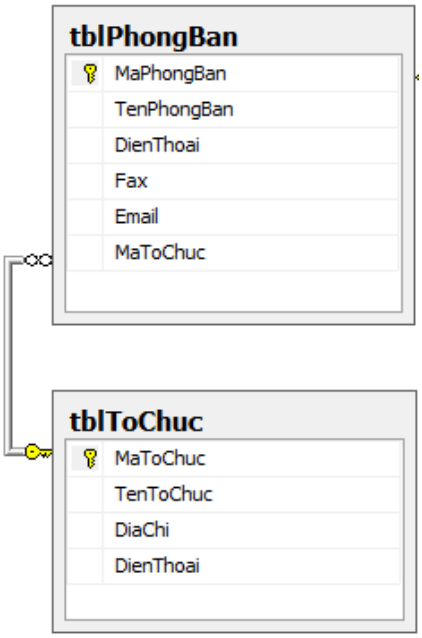
```
ALTER TABLE [dbo].[tblNhanVien_DienBienLuong] CHECK CONSTRAINT
[FK_tblNhanVien_DienBienLuong_tblNhanVien]
GO
/****** Object:  ForeignKey [FK_tblNhanVien_LichSuBanThan_tblNhanVien]
******/
ALTER TABLE [dbo].[tblNhanVien_LichSuBanThan]  WITH CHECK ADD  CONSTRAINT
[FK_tblNhanVien_LichSuBanThan_tblNhanVien] FOREIGN KEY([NhanVienID])
REFERENCES [dbo].[tblNhanVien] ([NhanVienID])
ON UPDATE CASCADE
ON DELETE CASCADE
GO
ALTER TABLE [dbo].[tblNhanVien_LichSuBanThan] CHECK CONSTRAINT
[FK_tblNhanVien_LichSuBanThan_tblNhanVien]
GO
/****** Object:  ForeignKey [FK_tblNhanVien_QTCongTac_tblNhanVien]
******/
ALTER TABLE [dbo].[tblNhanVien_QTCongTac]  WITH CHECK ADD  CONSTRAINT
[FK_tblNhanVien_QTCongTac_tblNhanVien] FOREIGN KEY([NhanVienID])
REFERENCES [dbo].[tblNhanVien] ([NhanVienID])
ON UPDATE CASCADE
ON DELETE CASCADE
GO
ALTER TABLE [dbo].[tblNhanVien_QTCongTac] CHECK CONSTRAINT
[FK_tblNhanVien_QTCongTac_tblNhanVien]
GO
/****** Object:  ForeignKey [FK_tblNhanVien_QTDaoTao_tblNhanVien]     ******/
ALTER TABLE [dbo].[tblNhanVien_QTDaoTao]  WITH CHECK ADD  CONSTRAINT
[FK_tblNhanVien_QTDaoTao_tblNhanVien] FOREIGN KEY([NhanVienID])
REFERENCES [dbo].[tblNhanVien] ([NhanVienID])
ON UPDATE CASCADE
ON DELETE CASCADE
GO
ALTER TABLE [dbo].[tblNhanVien_QTDaoTao] CHECK CONSTRAINT
[FK_tblNhanVien_QTDaoTao_tblNhanVien]
GO
/****** Object:  ForeignKey [FK_tblNhanVien_QTKhenThuong_tblNhanVien]
******/
ALTER TABLE [dbo].[tblNhanVien_QTKhenThuong]  WITH CHECK ADD  CONSTRAINT
[FK_tblNhanVien_QTKhenThuong_tblNhanVien] FOREIGN KEY([NhanVienID])
REFERENCES [dbo].[tblNhanVien] ([NhanVienID])
ON UPDATE CASCADE
ON DELETE CASCADE
GO
ALTER TABLE [dbo].[tblNhanVien_QTKhenThuong] CHECK CONSTRAINT
[FK_tblNhanVien_QTKhenThuong_tblNhanVien]
GO
/****** Object:  ForeignKey [FK_tblNhanVien_QTKyLuat_tblNhanVien]     ******/
ALTER TABLE [dbo].[tblNhanVien_QTKyLuat]  WITH CHECK ADD  CONSTRAINT
[FK_tblNhanVien_QTKyLuat_tblNhanVien] FOREIGN KEY([NhanVienID])
REFERENCES [dbo].[tblNhanVien] ([NhanVienID])
ON UPDATE CASCADE
ON DELETE CASCADE
GO
ALTER TABLE [dbo].[tblNhanVien_QTKyLuat] CHECK CONSTRAINT
[FK_tblNhanVien_QTKyLuat_tblNhanVien]
GO
/****** Object:  ForeignKey [FK_tblNhanVien_QuanHeGiaDinh_tblNhanVien]
******/
ALTER TABLE [dbo].[tblNhanVien_QuanHeGiaDinh]  WITH CHECK ADD  CONSTRAINT
[FK_tblNhanVien_QuanHeGiaDinh_tblNhanVien] FOREIGN KEY([NhanVienID])
REFERENCES [dbo].[tblNhanVien] ([NhanVienID])
ON UPDATE CASCADE
```

```
ON DELETE CASCADE
GO
ALTER TABLE [dbo].[tblNhanVien_QuanHeGiaDinh] CHECK CONSTRAINT
[FK_tblNhanVien_QuanHeGiaDinh_tblNhanVien]
GO
/****** Object:  ForeignKey [FK_tblNhanVien_TrinhDo_tblNhanVien]      ******/
ALTER TABLE [dbo].[tblNhanVien_TrinhDo]  WITH CHECK ADD  CONSTRAINT
[FK_tblNhanVien_TrinhDo_tblNhanVien] FOREIGN KEY([NhanVienID])
REFERENCES [dbo].[tblNhanVien] ([NhanVienID])
ON UPDATE CASCADE
ON DELETE CASCADE
GO
ALTER TABLE [dbo].[tblNhanVien_TrinhDo] CHECK CONSTRAINT
[FK_tblNhanVien_TrinhDo_tblNhanVien]
GO
/****** Object:  ForeignKey [FK_tblNhanVien_TrinhDo_tblTrinhDo]      ******/
ALTER TABLE [dbo].[tblNhanVien_TrinhDo]  WITH CHECK ADD  CONSTRAINT
[FK_tblNhanVien_TrinhDo_tblTrinhDo] FOREIGN KEY([MaTrinhDo])
REFERENCES [dbo].[tblTrinhDo] ([MaTrinhDo])
ON UPDATE CASCADE
ON DELETE CASCADE
GO
ALTER TABLE [dbo].[tblNhanVien_TrinhDo] CHECK CONSTRAINT
[FK_tblNhanVien_TrinhDo_tblTrinhDo]
GO
/****** Object:  ForeignKey [FK_tblPhongBan_tblToChuc]      ******/
ALTER TABLE [dbo].[tblPhongBan]  WITH CHECK ADD  CONSTRAINT
[FK_tblPhongBan_tblToChuc] FOREIGN KEY([MaToChuc])
REFERENCES [dbo].[tblToChuc] ([MaToChuc])
ON UPDATE CASCADE
GO
ALTER TABLE [dbo].[tblPhongBan] CHECK CONSTRAINT [FK_tblPhongBan_tblToChuc]
GO
```

6. Nhập dữ liệu mẫu cho các Table

6.1 Nhập dữ liệu mẫu cho các bảng bên quan hệ 1

Để nhập dữ liệu cho các bảng, chúng ta phải nhập dữ liệu cho các bảng bên quan hệ 1 trước. Ví dụ với các bảng có quan hệ sau:



Chúng ta phải nhập dữ liệu cho bảng tblToChuc trước, sau đó mới có thể nhập dữ liệu cho bảng tblPhongBan.

Khi nhập dữ liệu cho các bảng cần tránh sự trùng lặp về khóa chính.

```
INSERT [dbo].[tblToChuc] ([MaToChuc], [TenToChuc], [DiaChi],
[DienThoai]) VALUES (1, N'Cõ sở 1 - Cõ sở chính', N'01 Hoàng Văn
Thụ - Phýờng 4', N'0633836654')
INSERT [dbo].[tblToChuc] ([MaToChuc], [TenToChuc], [DiaChi],
[DienThoai]) VALUES (2, N'Cõ sở 2 - TTĐT lái xe', N'01 Nguyễn
Khuyến - Thành phố Đà Lạt', NULL)
```

6.2 Nhập dữ liệu mẫu cho các bảng bên quan hệ n

Như đã trình bày ở trên, khi nhập dữ liệu cho các bảng bên quan hệ bên n (bên nhiều), ngoài việc bảo toàn về sự không trùng lặp về khóa chính còn phải đảm bảo khóa ngoại phải tồn tại.

Ví dụ:

```
INSERT [dbo].[tblPhongBan] ([MaPhongBan], [TenPhongBan],
[DienThoai], [Fax], [Email], [MaToChuc]) VALUES (N'BGH', N'Ban
giám hiệu', N'', N'', N'', 1)
INSERT [dbo].[tblPhongBan] ([MaPhongBan], [TenPhongBan],
[DienThoai], [Fax], [Email], [MaToChuc]) VALUES (N'BQT', N'Ban
quản trị', NULL, NULL, NULL, 1)
INSERT [dbo].[tblPhongBan] ([MaPhongBan], [TenPhongBan],
[DienThoai], [Fax], [Email], [MaToChuc]) VALUES (N'DT', N'Khoa
Dân tộc nội trú', N'', N'', N'', 1)
INSERT [dbo].[tblPhongBan] ([MaPhongBan], [TenPhongBan],
[DienThoai], [Fax], [Email], [MaToChuc]) VALUES (N'KCB', N'Khoa
Khoa học cõ bản', N'063.3542535', N'', N'', 1)
INSERT [dbo].[tblPhongBan] ([MaPhongBan], [TenPhongBan],
[DienThoai], [Fax], [Email], [MaToChuc]) VALUES (N'KCK', N'Khoa
Cõ khí Động lực', N'063.3540803', N'', N'', 1)
INSERT [dbo].[tblPhongBan] ([MaPhongBan], [TenPhongBan],
[DienThoai], [Fax], [Email], [MaToChuc]) VALUES (N'KCN', N'Khoa
Công nghệ thông tin', N'063.3540106', N'', N'', 1)
```

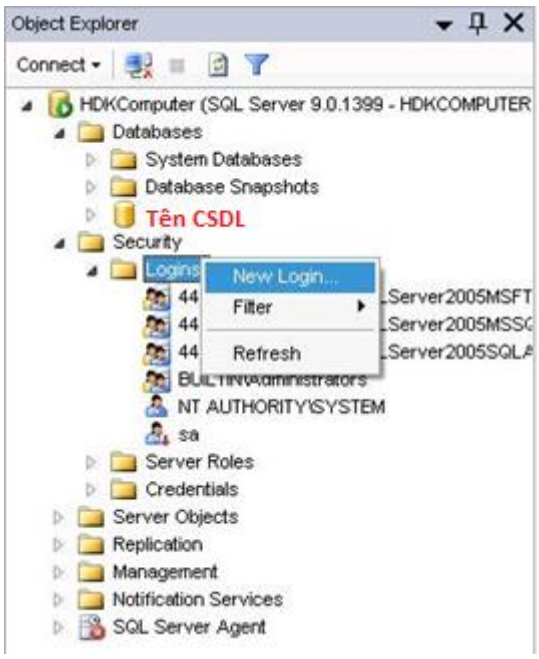
```
INSERT [dbo].[tblPhongBan] ([MaPhongBan], [TenPhongBan],
[DienThoai], [Fax], [Email], [MaToChuc]) VALUES (N'KDL', N'Khoa
Du lịch', N'063.3540804', N'', N'', 1)
INSERT [dbo].[tblPhongBan] ([MaPhongBan], [TenPhongBan],
[DienThoai], [Fax], [Email], [MaToChuc]) VALUES (N'KDN', N'Khoa
Điện - Điện tử', N'063.3832967', N'', N'', 1)
INSERT [dbo].[tblPhongBan] ([MaPhongBan], [TenPhongBan],
[DienThoai], [Fax], [Email], [MaToChuc]) VALUES (N'KKT', N'Khoa
Kinh tế', N'063.3541788', N'', N'', 1)
INSERT [dbo].[tblPhongBan] ([MaPhongBan], [TenPhongBan],
[DienThoai], [Fax], [Email], [MaToChuc]) VALUES (N'KNN', N'Khoa
Nông nghiệp & Sinh học ứng dụng', N'', N'', N'', 1)
INSERT [dbo].[tblPhongBan] ([MaPhongBan], [TenPhongBan],
[DienThoai], [Fax], [Email], [MaToChuc]) VALUES (N'KSP', N'Khoa
Sý phạm dạy nghề', N'', N'', N'', 1)
INSERT [dbo].[tblPhongBan] ([MaPhongBan], [TenPhongBan],
[DienThoai], [Fax], [Email], [MaToChuc]) VALUES (N'KT', N'Ký Túc
Xá', N'0633542266', N'', N'', 1)
INSERT [dbo].[tblPhongBan] ([MaPhongBan], [TenPhongBan],
[DienThoai], [Fax], [Email], [MaToChuc]) VALUES (N'PDT', N'Phòng
Đào tạo', NULL, NULL, NULL, 1)
INSERT [dbo].[tblPhongBan] ([MaPhongBan], [TenPhongBan],
[DienThoai], [Fax], [Email], [MaToChuc]) VALUES (N'PKH', N'Phòng
Nghiên cứu khoa học', NULL, NULL, NULL, 1)
INSERT [dbo].[tblPhongBan] ([MaPhongBan], [TenPhongBan],
[DienThoai], [Fax], [Email], [MaToChuc]) VALUES (N'PKK', N'Phòng
Khảo thí và Kiểm ðịnh chất lýợng', NULL, NULL, NULL, 1)
INSERT [dbo].[tblPhongBan] ([MaPhongBan], [TenPhongBan],
[DienThoai], [Fax], [Email], [MaToChuc]) VALUES (N'PKT', N'Phòng
Kế toán - Tài vụ', NULL, NULL, NULL, 1)
INSERT [dbo].[tblPhongBan] ([MaPhongBan], [TenPhongBan],
[DienThoai], [Fax], [Email], [MaToChuc]) VALUES (N'PSV', N'Phòng
Công tác HSSV', NULL, NULL, NULL, 1)
```

```
INSERT [dbo].[tblPhongBan] ([MaPhongBan], [TenPhongBan],
[DienThoai], [Fax], [Email], [MaToChuc]) VALUES (N'PTC', N'Phòng
Tổ chức - Hành chính', N'', N'', N'', 1)
INSERT [dbo].[tblPhongBan] ([MaPhongBan], [TenPhongBan],
[DienThoai], [Fax], [Email], [MaToChuc]) VALUES (N'TLX', N'Trung
tâm lái xe', NULL, NULL, NULL, 2)
INSERT [dbo].[tblPhongBan] ([MaPhongBan], [TenPhongBan],
[DienThoai], [Fax], [Email], [MaToChuc]) VALUES (N'TSX', N'Trung
tâm Thực nghiệm & Dịch vụ sản xuất', N'', N'', N'', 1)
INSERT [dbo].[tblPhongBan] ([MaPhongBan], [TenPhongBan],
[DienThoai], [Fax], [Email], [MaToChuc]) VALUES (N'TTH', N'Trung
tâm Đào tạo & Hướng nghiệp', N'', N'', N'', 1)
INSERT [dbo].[tblPhongBan] ([MaPhongBan], [TenPhongBan],
[DienThoai], [Fax], [Email], [MaToChuc]) VALUES (N'TTN', N'Nông
trại thực nghiệm và Sản xuất', N'', N'', N'', 1)
```

7. Cài đặt cơ sở dữ liệu Quản lý nhân sự

7.1 Tạo user đăng nhập cơ sở dữ liệu

+ Vào Security, nhắp chuột phải lên Logins → chọn New login :



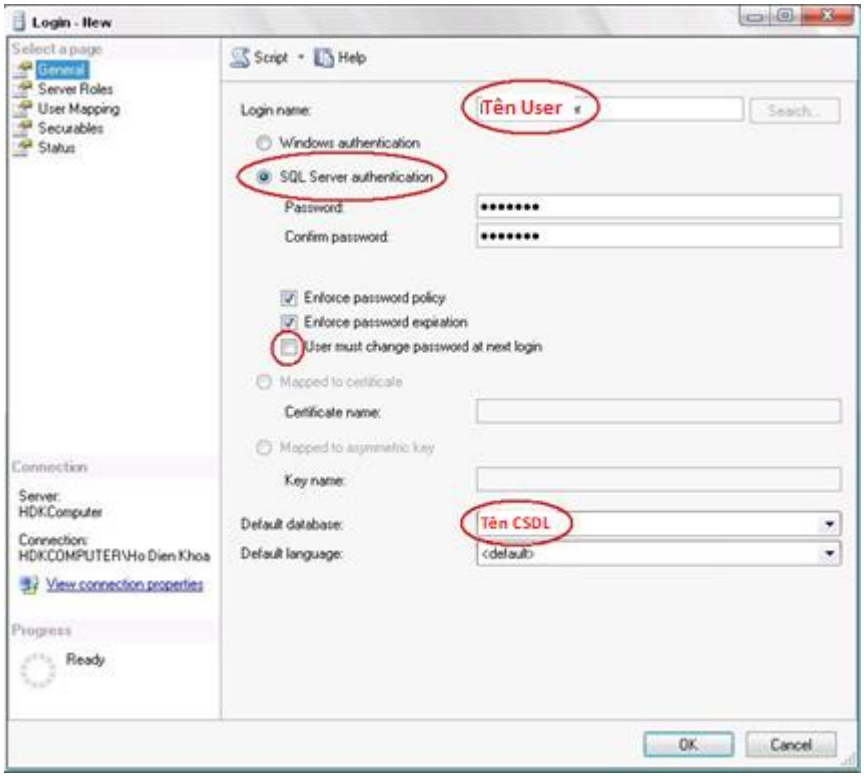
+ Trong tab "**General**":

Login name: tên tài khoản đăng nhập. Trong VD này chúng ta chọn **"Tên User"**.

Chọn **SQL Sever authenticaton**. Nhập password của tài khoản & confirm password.

Bỏ lựa chọn "User must change password at next login"

Default database: chọn CSDL chúng ta vừa tạo là **"Tên CSDL"**

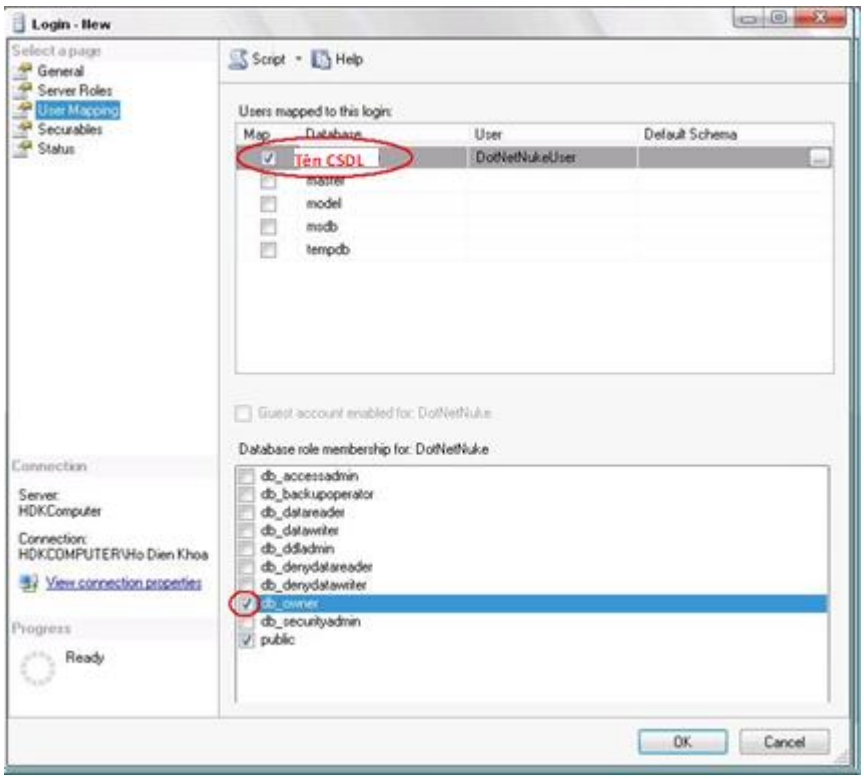


7.2 Cấp quyền user sử dụng cơ sử dữ liệu

+ Chuyển qua tab "**User Mapping**":

Trong khung "**Users mapped to this login**", check "**Tên CSDL**"

Trong khung "**Database role membership for: Tên CSDL**", check "**db_owner**"



*** Lưu ý:** Phần "**Authentication**" phải chọn "**SQL Server and Windows**"

Nhấn phải chuột vào computer, chọn **Properties**

Chọn tab "**Security**"

Trong khung Security, chọn "**SQL Server and Windows**"

Chọn **OK**.

7.3 Lưu trữ và quản lý cơ sở dữ liệu quản lý nhân sự

Loại 1: Full backup

Loại backup này sẽ backup dữ liệu đầy đủ nhất, vì vậy máy chủ sẽ mất nhiều thời gian để thực hiện nếu database lớn

Cú pháp

```
backup database Tên_Cơ_Sở_Dữ_Liệu to disk =
'Đường_dẫn\Tên_file.bak'
```

Ví dụ câu lệnh **full backup** cơ sở dữ liệu **QuanLyNhanSu** và lưu vào "**d:\bk**" (Lưu ý chúng ta phải tạo thư mục bk trong ổ đĩa D trước khi thực thi câu lệnh sau):

```
backup database QuanLyNhanSu to disk = 'd:\bk\QLNS.bak'
```

Loại 2: Different backup

Backup những dữ liệu phát sinh tính từ lần backup full gần nhất. Như vậy trước khi chúng ta thực hiện different backup thì full backup phải được thực hiện

trước. Chính vì vậy, khi chúng ta sử dụng loại backup này sẽ tiết kiệm được thời gian backup dữ liệu.

Cú pháp

```
backup database Tên_Cơ_Sở_Dữ_Liệu to disk =
'Đường_dẫn\Tên_File.bak' with differential
```

Ví dụ

```
backup database QuanLyNhanSu to disk = 'D:\sql\QLNS_Diff.bak'
with differential
```

Loại 3: Log backup

Backup dữ liệu sử dụng tập tin log và log backup được sử dụng để backup dữ liệu tại thời điểm cụ thể ví dụ như mỗi giờ trong ngày.

Cú pháp

```
backup log Tên_Cơ_Sở_Dữ_Liệu to disk = 'Đường_dẫn\Tên_File.trn'
```

Ví dụ

```
backup log QuanLyNhanSu to disk = 'D:\sql\QLNS.trn'
```

BÀI 2: THIẾT KẾ CÁC CHỨC NĂNG CHÍNH

Mã Bài: MĐ30_02

Giới thiệu:

Để xây dựng cho ứng dụng quản lý nhân sự nói chung rất đa dạng, trong phạm vi bài này chúng tôi chỉ xây dựng ứng dụng quản lý nhân sự cho một Trường Cao đẳng nghề.

Mục tiêu:

- Chọn lọc được một số nghiệp vụ trong tiến trình quản lý nhân sự của doanh nghiệp;
- Đặc tả được các chức năng chính của ứng dụng;
- Tạo các form giao diện cho từng chức năng của phần mềm;
- Viết code xử lý các biến cố, sự kiện cho các form nhằm thực hiện đầy đủ các yêu cầu xử lý dữ liệu, đồng thời tăng tính mềm dẻo, linh hoạt cho form;
- Khoa học, sáng tạo trong việc thiết kế các chức năng;
- Tư duy tích cực để cho các tính năng của phần mềm bám sát được thực tiễn.

Nội dung chính:

1. Tạo project và kết nối cơ sở dữ liệu

1.1 Thiết lập chuỗi kết nối trong Properties của project

Trong App.config

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
    <configSections>
    </configSections>
    <connectionStrings>
        <add name="QLNS.Properties.Settings.ChuoiKetNoi"
connectionString="Data Source=.;Initial
Catalog=QuanLyNhanSu;User ID=sa"
            providerName="System.Data.SqlClient" />
    </connectionStrings>
</configuration>
```

1.2 Khai báo các đối tượng dùng chung ADO.NET

Đối với ngôn ngữ C#, việc khai báo các đối tượng ADO.NET dùng chung phải khai báo dạng thành phần static (tĩnh) của lớp.

```
using System.Data;
using System.Data.SqlClient;
namespace QLNS
{
    class Class1
    {
        static SqlCommand lenh = new SqlCommand();
        static SqlDataAdapter da = new SqlDataAdapter();
     }
}
```

1.3 Khởi tạo kết nối đến cơ sở dữ liệu

Để thuận lợi cho việc khởi tạo kết nối đến cơ sở dữ liệu, trước tiên là khai báo chuỗi:

```
string  connStr =Properties.Settings.Default.ChuoiKetNoi;
```

Sau đó chúng ta có thể sử dụng phương thức tĩnh của lớp để khởi tạo kết nối một lần duy nhất đến cơ sở dữ liệu:

```
static private SqlConnection _ketnoi
                                = new SqlConnection(connStr);
static public SqlConnection ketnoi()
        {
            if (_ketnoi.State == ConnectionState.Closed)
                _ketnoi.Open();
            return _ketnoi;
         }
```

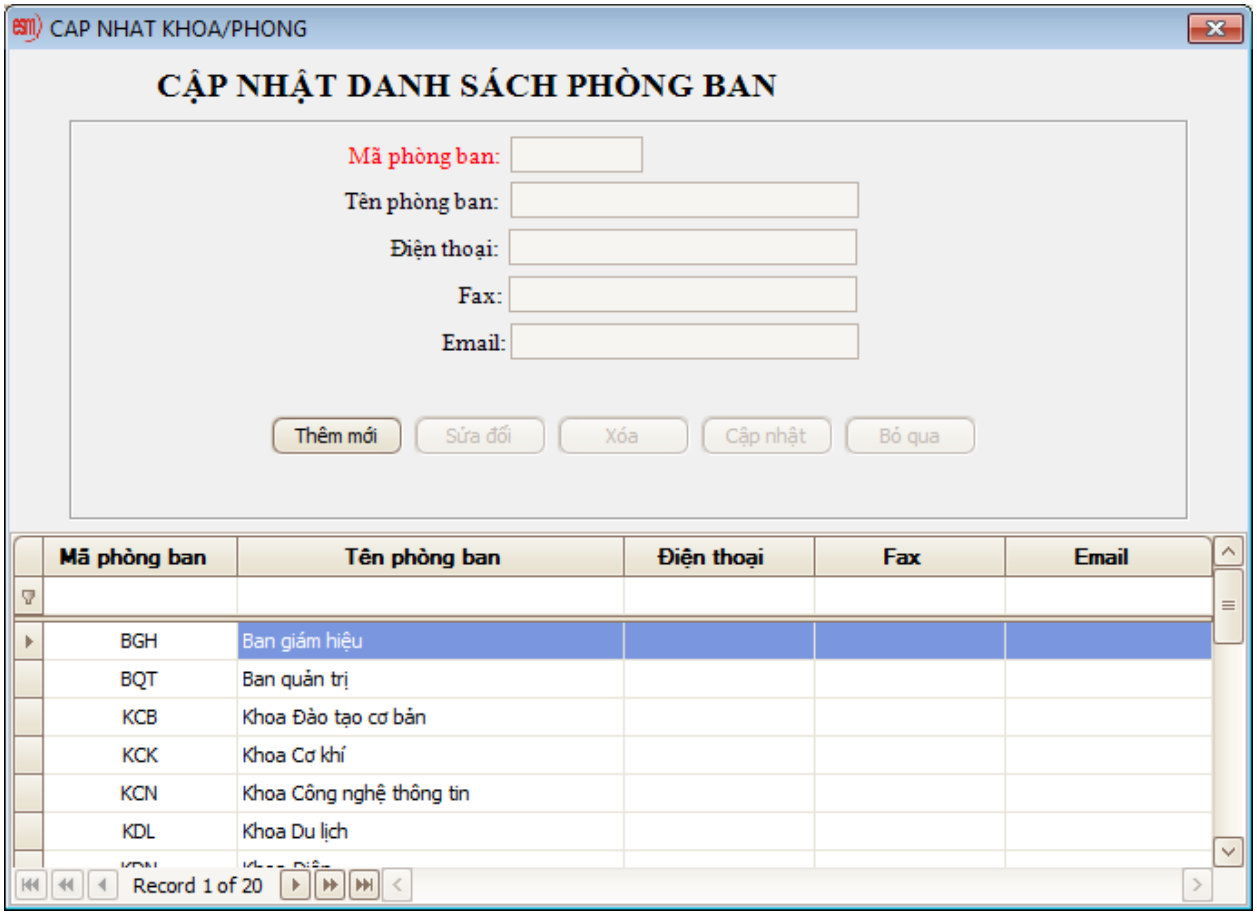
Hoặc chúng ta có thể không thông qua biến chuỗi, code hoàn thiện của lớp kết nối cơ sở dữ liệu như sau:

```
using System.Data;
using System.Data.SqlClient;
namespace QLNS
{
    class Class1
    {
        static SqlCommand lenh = new SqlCommand();
```

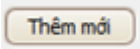
```
        static SqlDataAdapter da = new SqlDataAdapter();

        static private SqlConnection _ketnoi = new
SqlConnection(Properties.Settings.Default.ChuoiKetNoi );
        static public SqlConnection ketnoi()
        {
            if (_ketnoi.State == ConnectionState.Closed)
                _ketnoi.Open();
            return _ketnoi;
        }
    }
}
```

2. Thiết kế chức năng quản lý hồ sơ

2.1. Chức năng Cập nhật dữ liệu cho phòng ban



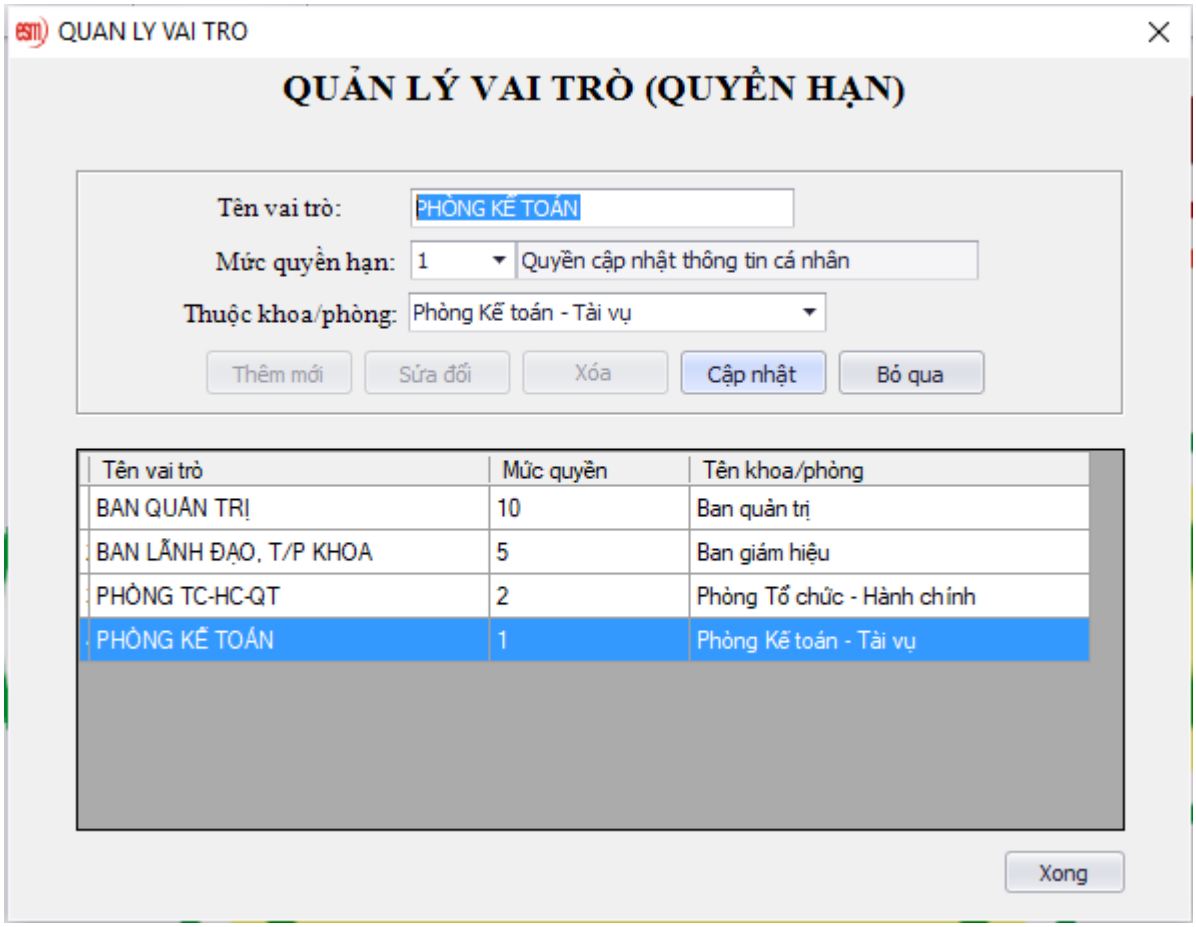
Hình 2.1

Khi bạn nhấp vào nút này bạn sẽ có một bản thông tin về Khoa/phòng ban của đơn vị, nếu bạn muốn thêm mới bạn nhấp vào nút Thêm mới 

khi đó bạn

sẽ điền thông tin ở các ô trên và sau đó bạn muôn thay đổi bạn nhấp vào nút Cập nhật Cập nhật hoăc nút Bỏ qua Bỏ qua để không thực thi, bạn muốn thay đổi tên Khoa/phòng ban bạn chỉ cấn nhấp chuột vào danh sách Khoa/phòng ban ở bảng danh sách dưới khi đó bạn mới Sửa đổi Sửa đổi tin hoặc Xóa Xóa được, nếu ban muốn lọc theo từng mục một bạn nhấp vào từng cộ thông tin bạn sẽ thấy chức năng lọc và thông tin lọc hình cái phễu.

2.2. Chức năng Cập nhật dữ liệu cho Đơn vị/Chức vụ



Hình 2.2

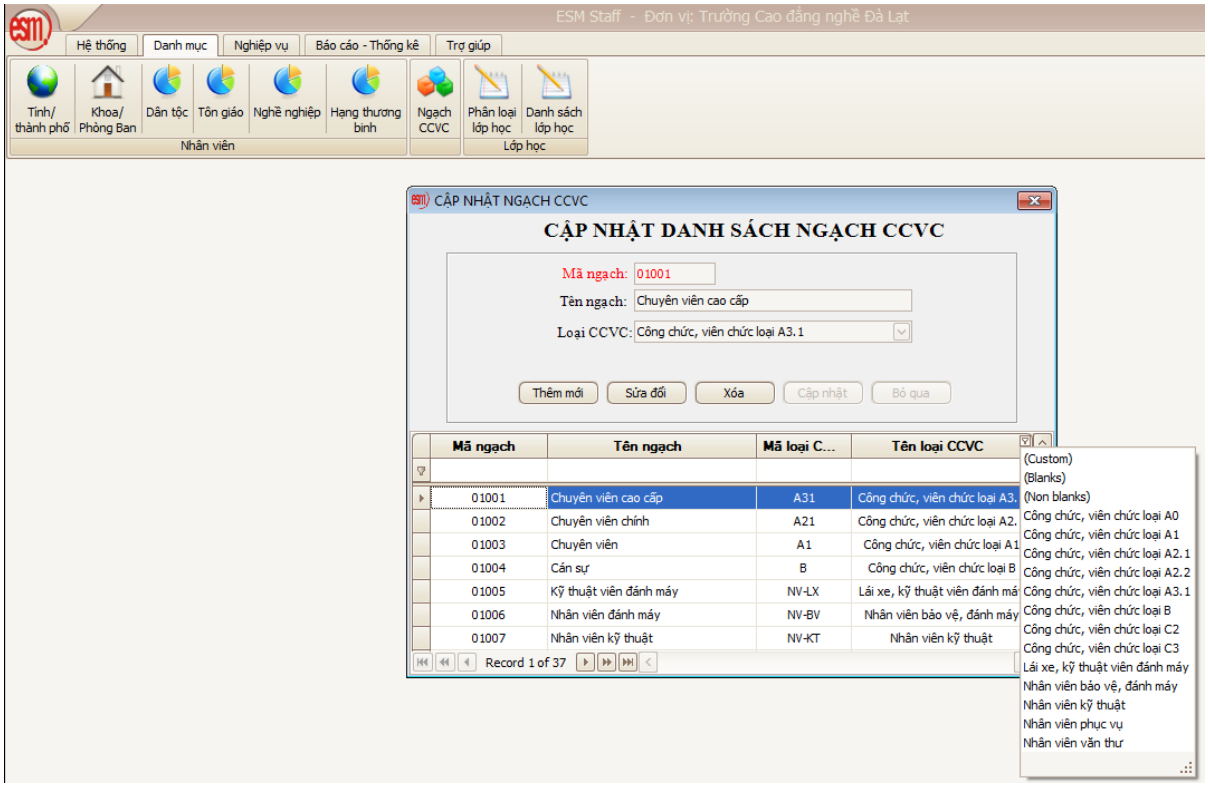
Việc thêm, sửa, xóa vai trò như đối với danh sách phòng ban.

2.3 Chức năng Cập nhật dữ liệu cho Ngành nghề, ngạch viên chức

Ở bảng chức năng nghạch CCVC, khi bạn muốn thêm mới bạn nhấp bàn nút

Thêm mới Thêm mới và khi đó các ô trên sẽ hiển thị và bạn điền thông tin và ô trên sau đó nhấp nút Cập nhật Cập nhật để thực thi hoặc là nút Bỏ qua Bỏ qua để bỏ qua không thực thi, khi bạn muốn sủa thông tin đã có bạn nhấp và thông tin mà bạn muốn sửa có ở danh sách dưới thì nút Sử đôi Sửa đổi và Xóa Xóa mới hiển thi khi

đó bạn mới thực thi được công việc của mình, ở đây bạn cũng thấy chức năng lọc theo từng điều kiện hình cái phễu nhỏ ở góc bên phải mỗi cột thông tin.

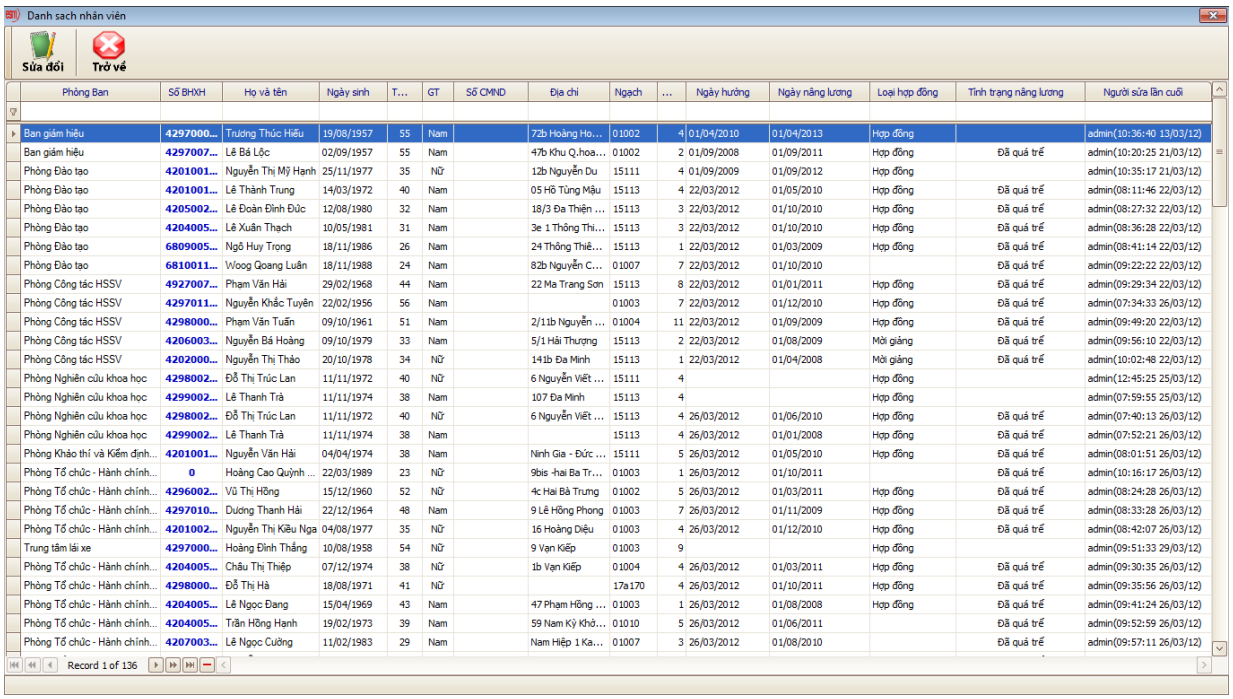


Hình 2.3

2.4 Chức năng Cập nhật dữ liệu cho nhân viên

Danh sách nhân viên

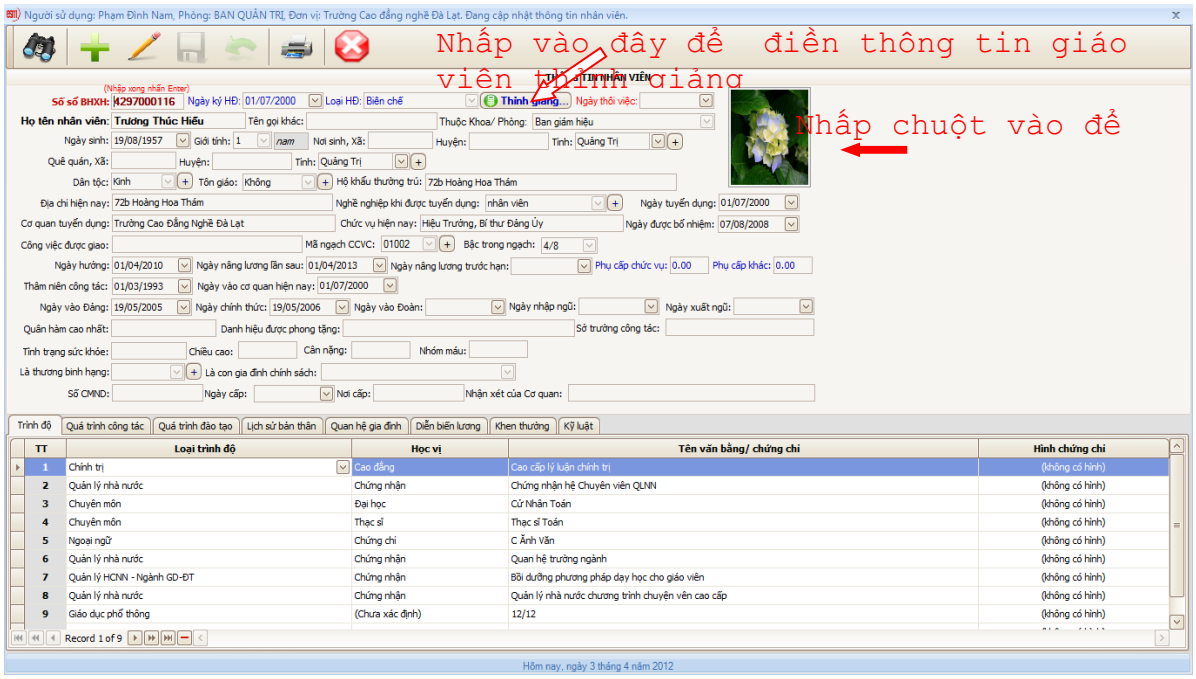
+ Bảng nhân sự bạn sẽ có chức năng quản lý thông tin nhân viên cơ bản.



Hình 2.4

Trên đây là sanh sách nhân viên của đơn vị, khi bạn muốn sửa thông tin nhân viên thì bạn nhấp vào biểu tượng Sửa đổi khi đó sẽ hiển thi bảng chức năng sau.

Cập nhập thông tin nhân viên

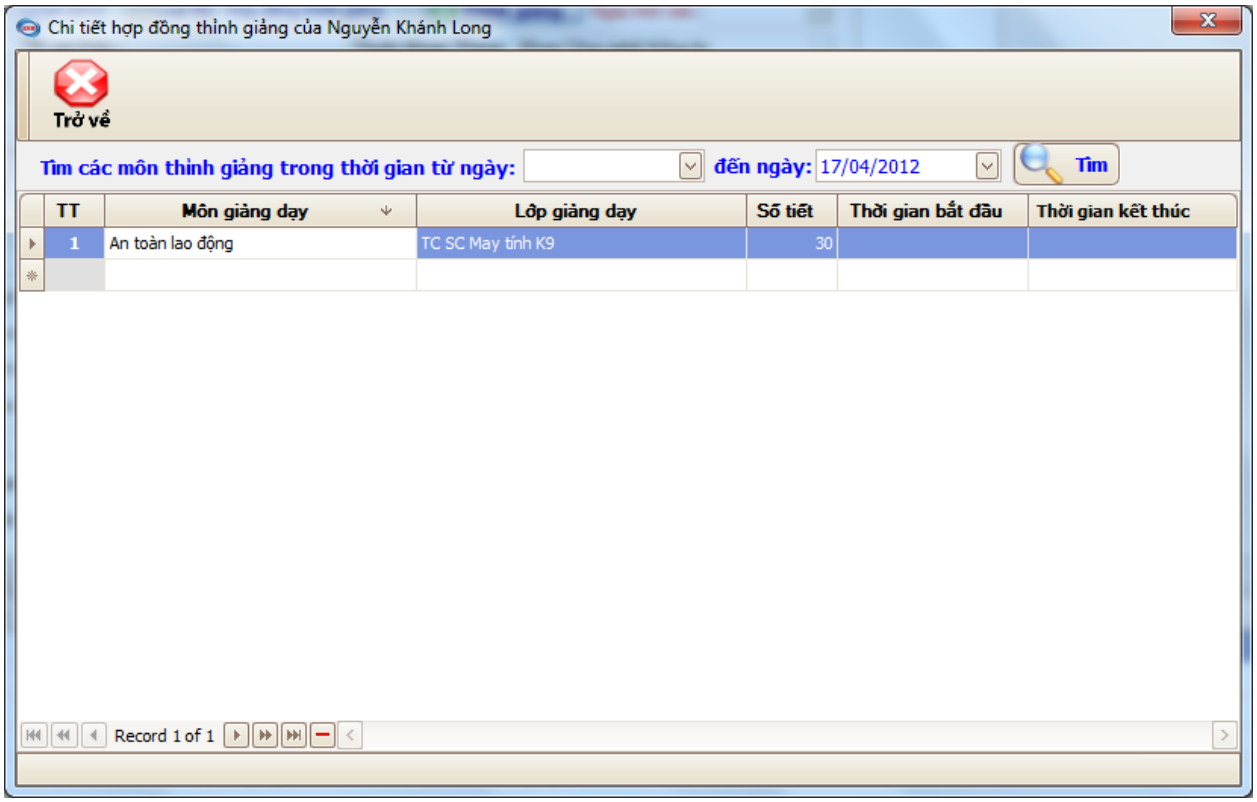


Hình 2.5

Khi này bạn nhấp vào biểu tượng Bút chì khi đó bạn sẽ sửa đổi được thông tin của nhân viên mà bạn muốn sửa đôi, sau khi sửa xong bạn nhấp vào biểu tượng đĩa mềm là khi đó bạn đã lưu thông tin vừa sửa khi đó bạn mới sửa đổi hoặc

thêm thông tin ở danh sách biên dưới được. Nếu bạn muốn thêm nhân viên bạn chỉ cần nhấp vào biểu tượng dấu Cộng ở thanh công cụ trên là bạn có thể thêm thông tin nhân viên, khi bạn thêm thông tin nhân nhân viên bạn nhấp vào biểu tượng đĩa mềm để lưu mà lưu không được thì khi đó có thể bạn đã không nhập một số thông tin mặc định bắt bạn phải nhập, khi đó bạn sẽ thấy dòng thông báo chữ đỏ ở trên bảng là phần thông tin nào bạn chưa nhập, sau đó bạn nhập lại và nhấp và biểu tượng đĩa mềm để lưu thông tin nhân viên cơ bản, khi bạn nhấp lưu ở danh sách dưới sẽ hiển thị và khi đó bạn sẽ thêm thông tin chi tiết của nhân viên vào danh sách dưới của nhân viên mà bạn vừa nhập ở trên, khi bạn nhập vào hệ thống sẽ tự lưu và thay đổi khi bạn thay đổi.

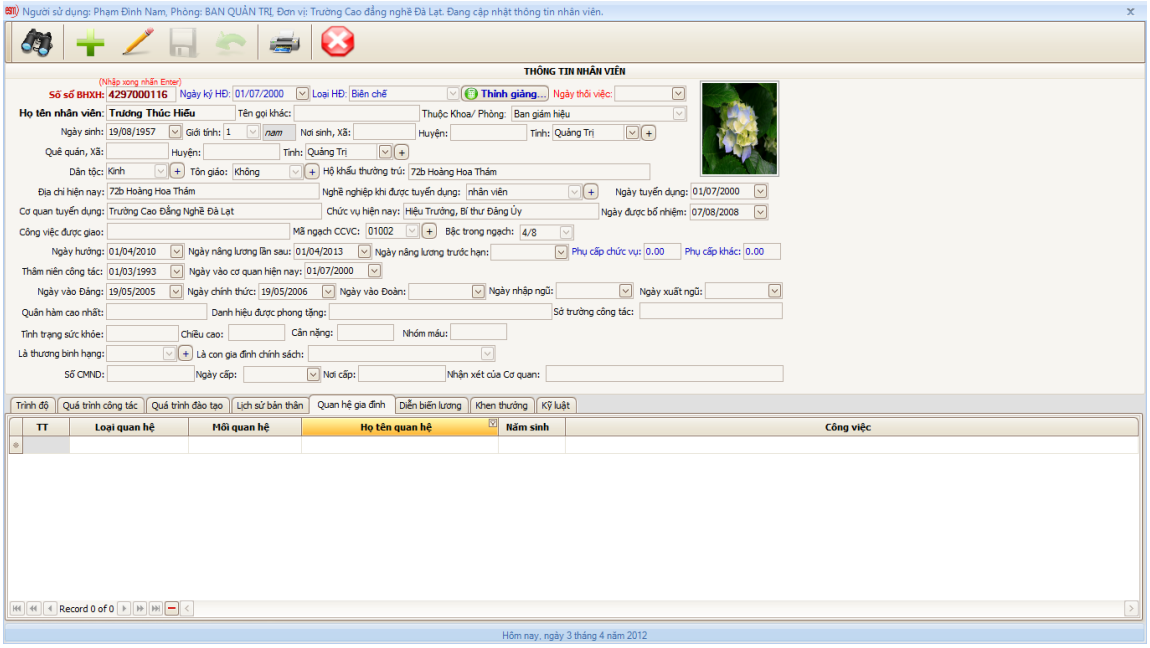
Khi nhân viên là Giảng viên thỉnh giảng bạn lưu thông tin nhân viên và sau đó nhấn vào nút Thỉnh giảng sẽ hiển thị bảng sau để bạn điền thông tin giáo viên thỉnh giảng.



Hình 2.6

2.5. Chức năng Cập nhật dữ liệu cho quan hệ gia đình nhân viên

Bảng chức năng quan hệ gia đình, lưu chữ thông tin và mối quan hệ các thân nhân của nhân viên.

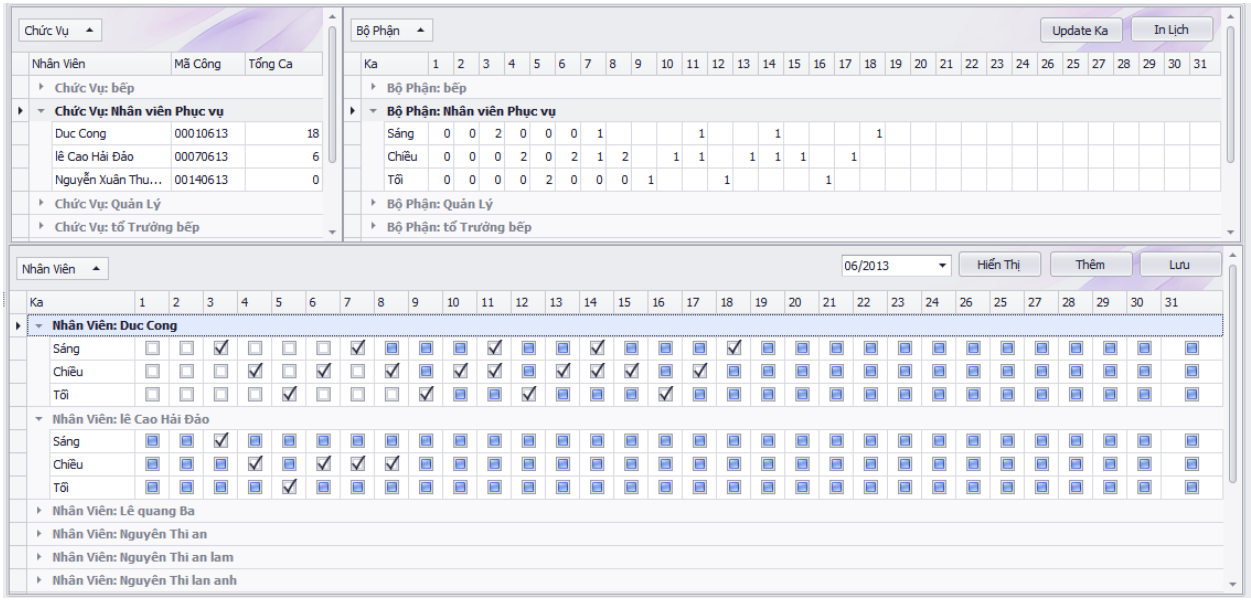


Hình 2.7

Và một số thông tin chi tiết của nhân viên cũng được phân trong mỗi tab thuận tiện cho quá trình nhập dữ liệu.

3 Thiết kế chức năng quản lý lương

3.1 Chức năng Chấm công



Hình 2.8

Gridcontrol trên trái: hiển thị tổng ca của tường nhân viên gom nhóm theo bộ phận

Gridcontrol trên phải: hiển thị tổng nhân viên đi làm trong ca theo từng ngày, từng bộ phận

Gridcontrol dưới: giao diện để ghi công và xếp lịch

Nút chức năng:

Hiên thị: hiển thị bảng ghi công theo từng tháng

Thêm: thêm bảng ghi công khi vào tháng mới

Lưu: lưu khi đã xếp công, ghi công xong

Lương tháng

Drag a column header here to group by that column 06/2013 Lương Ca : 35000 Tính Lương In DS Lương

Mã NV	Tên NV	Tên CV	Số Điện Thoại	Ngày Sinh	Tổng Ca	Lương	Tiền Trừ	Tiền Cộng	Thực Lĩnh
1	Duc Cong	Nhân viên Phục vụ	0123456789	03/12/1991	18	756000	25000	12000	743000
2	Nguyên Thi an	tổ Trưởng Phục Vụ	0123456789	03/12/1993	0	0	150000	0	-150000
3	Lê quang Ba	tổ Trưởng Phục Vụ	0123456789	03/12/1994	0	0	0	15555	15555
4	Nguyễn Văn c	Thủ Quỹ, Thủ Kho	0123456789	03/12/1990	0	0	0	0	0
5	Nguyễn thu thủy	tổ Trưởng bếp	0123456789	03/12/1990	0	0	0	0	0
6	Trần Đức Cống	bếp	0123456789	03/12/1989	0	0	0	0	0
7	lê Cao Hải Đảo	Nhân viên Phục vụ	0123456789	03/12/1989	6	252000	0	0	252000
11	Nguyên Thi lan anh	bếp	0123456789	03/12/1993	0	0	0	0	0
12	Nguyên Thi an lam	Quản Lý	0123456789	03/12/1993	0	0	0	0	0
14	Nguyễn Xuân Thuyên	Nhân viên Phục vụ	0123456789	03/12/1993	0	0	0	0	0
15	yyyyyyyyyyyyy	Quản Lý	0979419515	17/06/1991	0	0			

Hình 2.9

Gridcontrol: hiển thị thông tin nhân viên, tổng ca, lương (lương = tổng ca * lương ca * hệ số lương), tiền trừ(tiền phạt, tiền ứng), tiền cộng(tiền thưởng,tiền bo…), thực lỉnh (thực lĩnh = lương + tiền cộng – tiền trừ)

Nút chức năng:

- Tính lương: tính lương tháng cho nhân viên
- In ds lương: in những thông tin hiển hị trên gridcontrol

3.2 Chức năng Tính lương

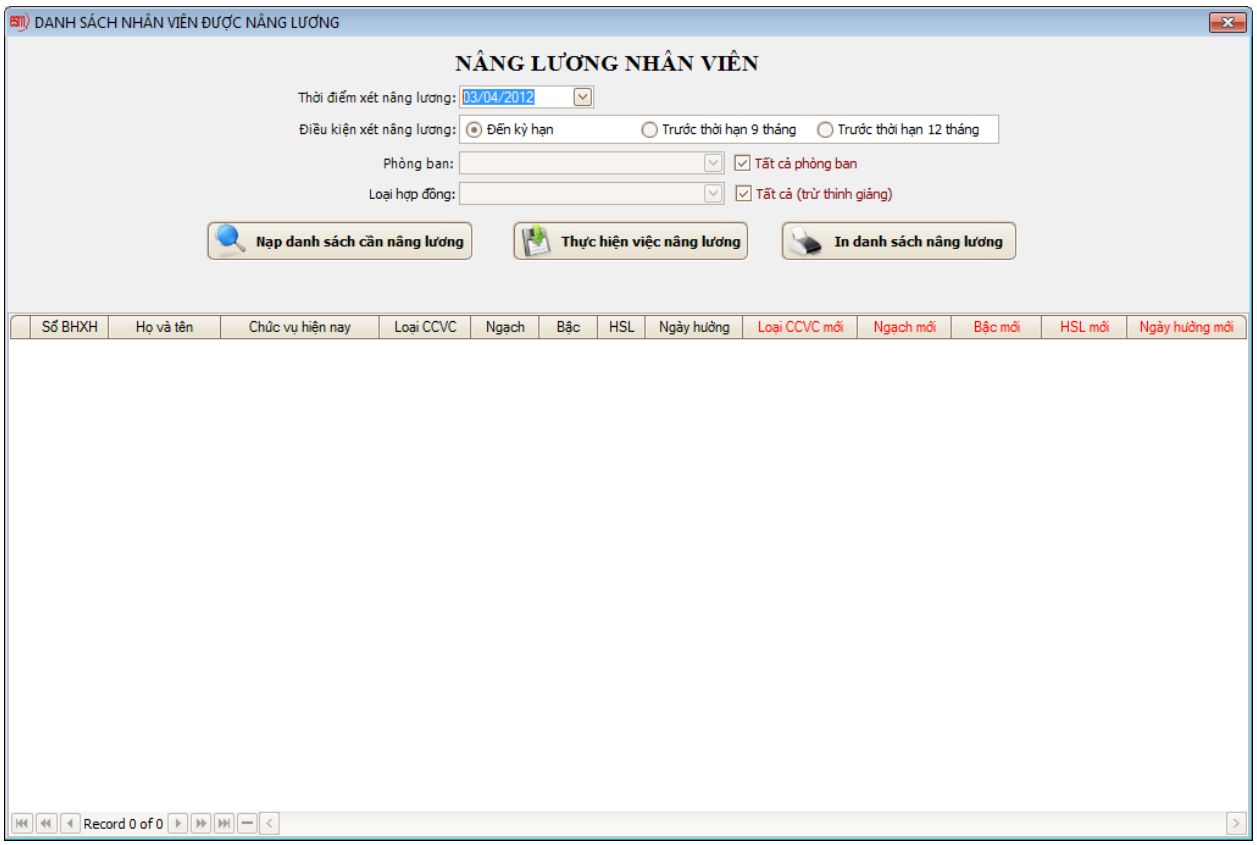
Ở nút Nâng lương

, chuyển ngach, ký lại hợp đồng, bạn nhấp vô để thực hiện các chức năng như đã hiển thị ở trên nút. Bạn nhập thời gian lúc nâng lương, điều kiện xét là gì, phòng nào và loại hợp đông, khi bạn muốn nâng lương tất cả thì bạn dánh dấu bào ô vuông tất cả. Nhung trước khi nâng lương bạn phải nạp danh sách nhân sự vô trong danh sách bằng nút

, sau khi đã điền đầy đủ thông tin banh nhấp vào nút Thực hiện việc nâng lương

, sau đó bạn chỉ cấn nhấn nút

(In danh sách nâng lương) để in ra thông tin nhân viên đã được nâng lương.

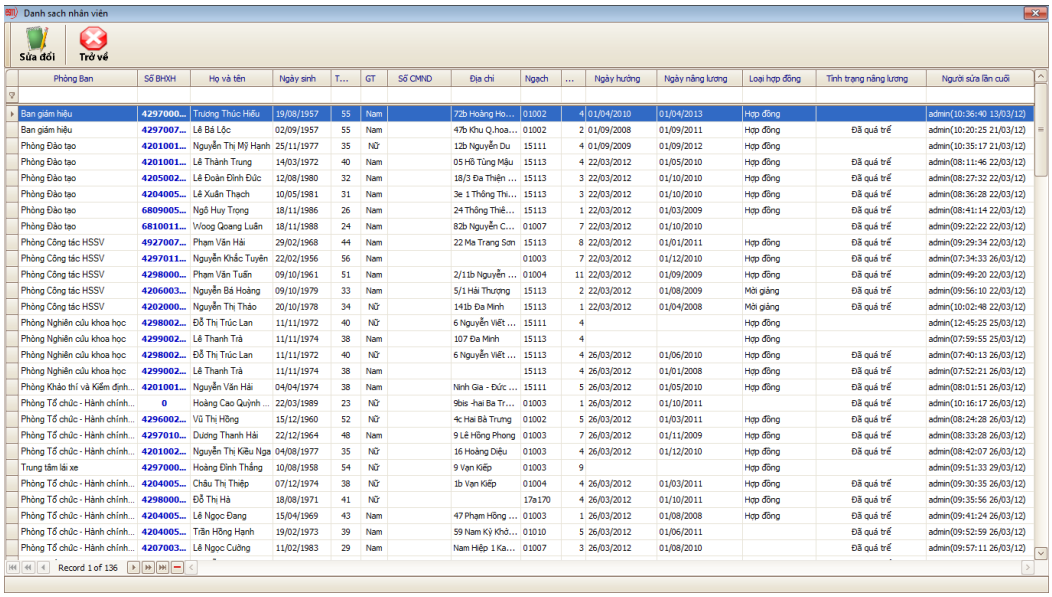


Hình 2.10

4 Thiết kế chức năng Tra cứu - Tìm kiếm thông tin

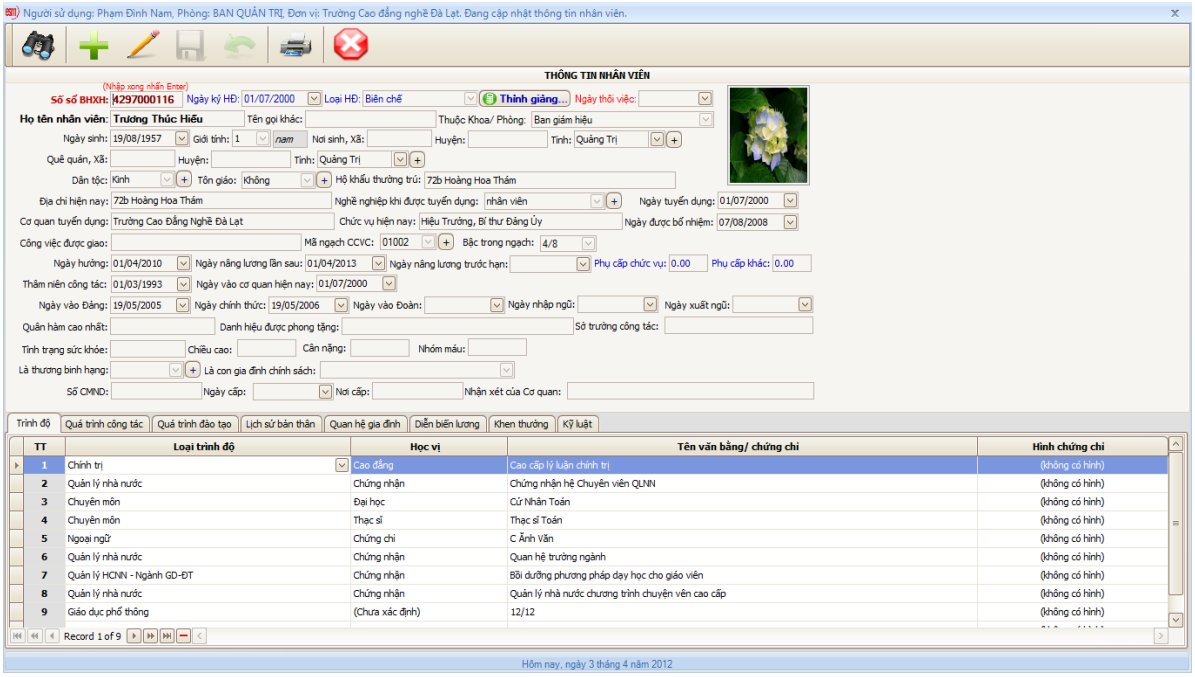
4.1. Tra cứu hồ sơ lý lịch

Từ danh sách nhân viên, nếu bạn muốn tìm kiếm thông tin nhân viên nào bạn có thể tìm kiếm theo bất kỳ trường nào trên thanh lọc (dòng trên cùng). Khi tìm thấy nhân viên bạn nhấp đôi vào dòng tương ứng để mở cho việc xem thông tin:



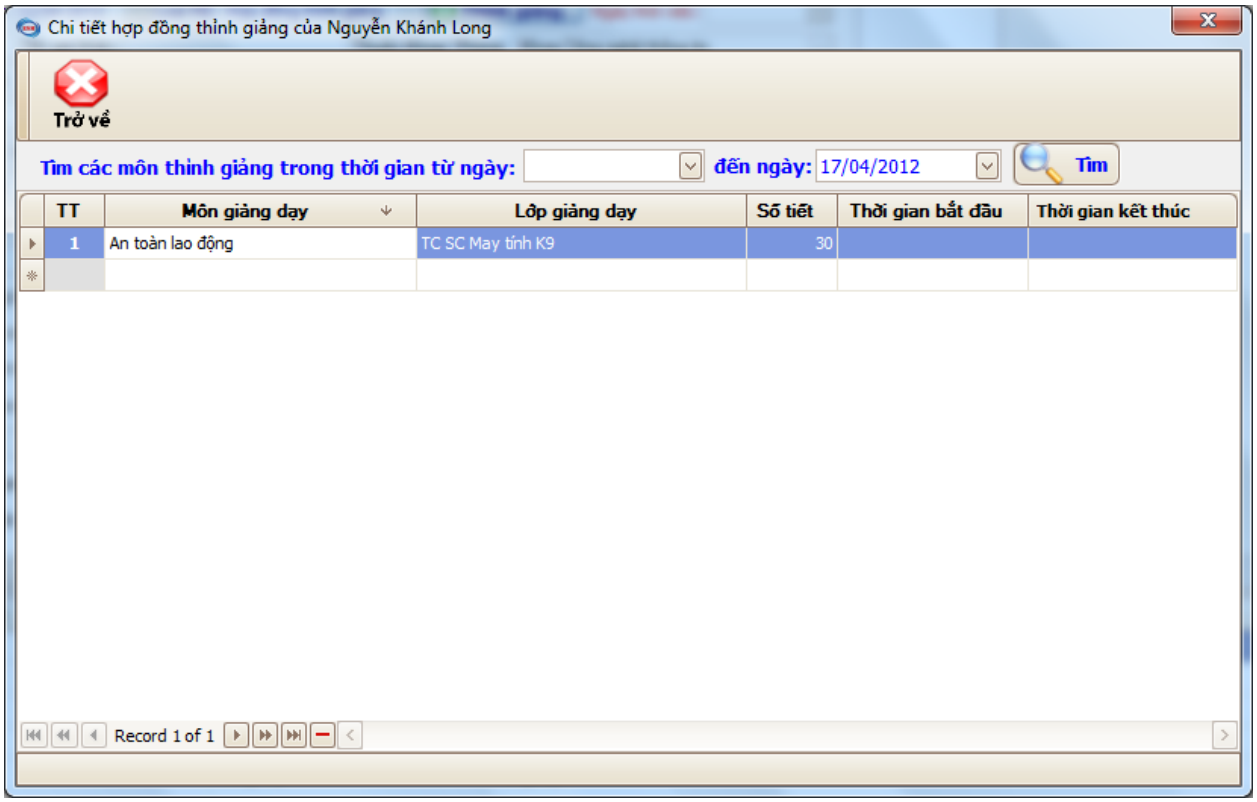
Hình 2.11

Từ cửa sổ thông tin chi tiết nhân viên:



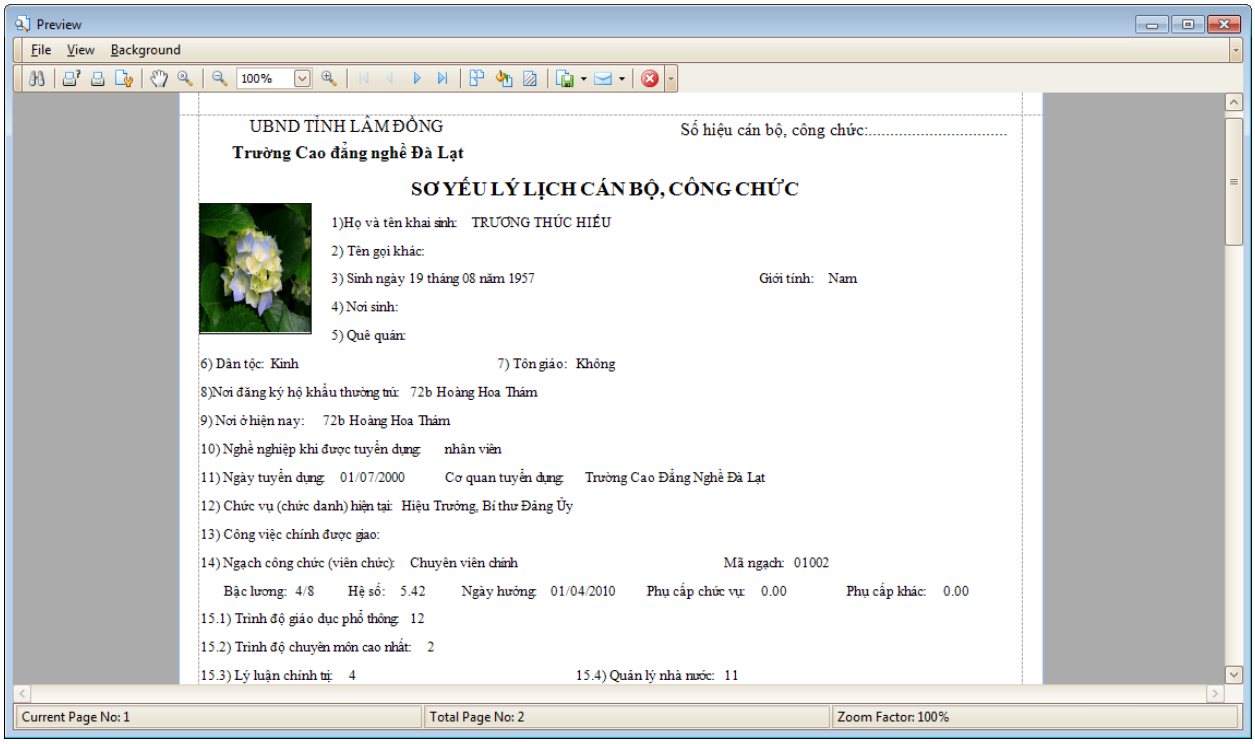
Hình 2.12

Khi nhân viên là Giảng viên thỉnh giảng bạn lưu thông tin nhân viên và sau đó nhấn vào nút **Thỉnh giảng** sẽ hiển thị bảng sau để bạn điền thông tin giáo viên thỉnh giảng.



Hình 2.13

Nếu bạn muốn xem danh sách tất cả nhân viên bạn nhấp vào biểu tượng ống nhòm . Nếu bạn muốn in thông tin nhân viên bạn nhấp vào biểu tượng máy in trên thanh công cụ, bạn sẽ có thông tin liên quan đến nhân viên mà bạn muốn in như hình rưới.



Hình 2.14

Khi bạn muộn thoát chương trình bạn nhấp vào biểu tượng chữ X là xong .

4.2. Tra cứu theo lương

Danh sách nhân viên đã nâng lương trước thời hạn

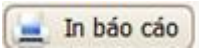
Ở bảng THEO DÕI TÌNH HÌNH NÂNG LƯƠNG. Tại thời điểm bạn muỗn xét và nhập thông tin cần xét ở ô rưới bạn sẽ có thông tinh nhân viên nâng lương trước thời hạn, sau đó bạn nhấn và nút In báo cáo để có thông tin.

Hình 2.15

5. Thiết kế chức năng Báo cáo - Thống kê

5.1. Thống kê theo các tiêu chí về hồ sơ

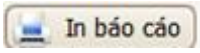
Lý lịch trích ngang giáo viên cơ hữu

Từ nút chức năng lý lịch trích ngang giáo viên cơ hữu bạn có bảng rưới. Trong bảng chức năng dưới, khi bạn nhấn nút In báo cáo 

bạn sẽ có lý lịch trích ngang của nhân viên thuộc các phòng bạn theo điều kiện.

Hinhg 2.16

Lý lịch trích ngang Giáo viên thỉnh giảng

Bạn nhấp vào Lý lịch trích ngang Giáo viên thỉnh giảng và nhấp và nút 

(In báo cáo) để có thông tin lý lịch trích ngang giáo viên thỉnh giảng.

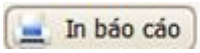
Hình 2.17

Năng lực giáo viên, cán bộ giảng dạy

Tại bảng năng lực giáo viên và cán bộ giảng dạy, bạn sẽ có thông tin về năng lực của giáo viên và cán bộ từ đó sẽ có điều chỉnh phù hợp, sau đó bạn nhấn nút In báo cáo để có thông tin năng lực giáo viên.

Hình 2.18

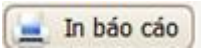
Danh sách đi học chuyên môn, nghiệp vụ

Cũng như bảng trên bảng BÁO CÁO TÌNH HÌNH ĐI HỌC CHUYÊN MÔN - NGHIỆP VỤ. Bạn nhấp thời gian bắt đầu và kết thúc với điều kiện lọc ở dưới bạn sẽ có thông tin mà bạn muốn biết, sau đó bạn nhấp vào nút In báo cáo  để có thông tin.

Hình 2.19

5.2. Thống kê về lương

Danh sách nhân viên đủ điều kiện nâng lương

Bạn chon thông tin ở các ô trên và nhấp và nút In báo cáo 

để có danh sách.

Hình 2.20

BÀI 3: THIẾT KẾ GIAO DIỆN CHƯƠNG TRÌNH

Mã Bài: MĐ30_03

Giới thiệu:

Hệ thống giao diện cần đẹp mắt và dễ sử dụng để tạo hứng thú cho người làm việc, nhất là với phần mềm thương mại điều này là hết sức cần thiết để tạo ấn tượng ngay từ giai đoạn tiếp thị ban đầu đến khách hàng.

Mục tiêu:

- Vẽ được quy trình của phần mềm quản lý nhân sự;
- Sử dụng được một số phần mềm thiết kế giao diện;
- Xây dựng giao diện thân thiện, cấu trúc logic phù hợp với tiến trình công việc;
- Xây dựng giao diện đăng nhập cho người dùng, nhằm tăng cường tính bảo mật của dữ liệu;
- Cẩn thận, khoa học, sáng tạo trong việc thiết kế giao diện phần mềm.

Nội dung chính:

1. Tạo form đăng nhập với người dùng

1.1 Tạo bảng lưu thông tin đăng nhập

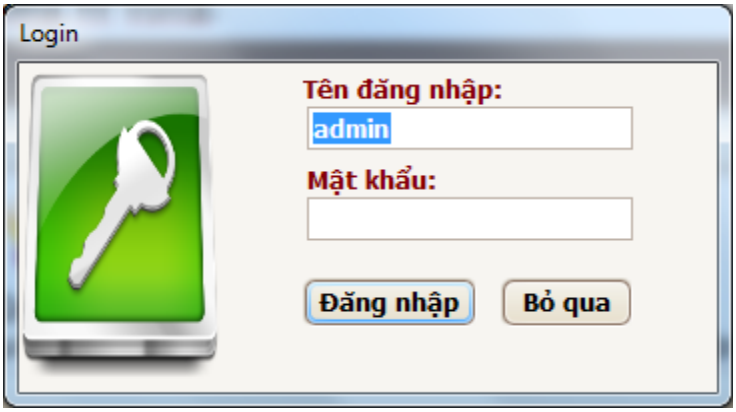
```
CREATE TABLE [dbo].[_Roles](
	[RoleID] [int] IDENTITY(1,1) NOT NULL,
	[RoleName] [nvarchar](50) NULL,
	[RoleLevel] [int] NULL,
	[MaPhongBan] [nvarchar](10) NULL,
 CONSTRAINT [PK__Roles] PRIMARY KEY CLUSTERED
(
	[RoleID] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  =
ON) ON [PRIMARY]
) ON [PRIMARY]
GO
CREATE TABLE [dbo].[_Users](
	[UserID] [int] IDENTITY(1,1) NOT NULL,
	[UserFullName] [nvarchar](50) NULL,
	[UserName] [nvarchar](50) NOT NULL,
```

```
	[UserPass] [nvarchar](50) NULL,
	[RoleID] [int] NULL,
	[Status] [int] NULL,
	[Note] [nvarchar](100) NULL,
 CONSTRAINT [PK__Users] PRIMARY KEY CLUSTERED
(
	[UserID] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS  = ON, ALLOW_PAGE_LOCKS  =
ON) ON [PRIMARY]
) ON [PRIMARY]
GO
ALTER TABLE [dbo].[_Roles]  WITH CHECK ADD  CONSTRAINT
[FK__Roles__RoleLevels] FOREIGN KEY([RoleLevel])
REFERENCES [dbo].[_RoleLevels] ([RoleLevel])
ON UPDATE CASCADE
GO
ALTER TABLE [dbo].[_Roles] CHECK CONSTRAINT
[FK__Roles__RoleLevels]
GO
ALTER TABLE [dbo].[_Roles]  WITH CHECK ADD  CONSTRAINT
[FK__Roles_tblPhongBan] FOREIGN KEY([MaPhongBan])
REFERENCES [dbo].[tblPhongBan] ([MaPhongBan])
ON UPDATE CASCADE
GO
ALTER TABLE [dbo].[_Roles] CHECK CONSTRAINT
[FK__Roles_tblPhongBan]
GO
ALTER TABLE [dbo].[_Users]  WITH CHECK ADD  CONSTRAINT
[FK__Users__Roles] FOREIGN KEY([RoleID])
REFERENCES [dbo].[_Roles] ([RoleID])
ON UPDATE CASCADE
ON DELETE CASCADE
GO
ALTER TABLE [dbo].[_Users] CHECK CONSTRAINT [FK__Users__Roles]
```

```
GO
ALTER TABLE [dbo].[_Users]  WITH CHECK ADD  CONSTRAINT
[FK__Users__Status] FOREIGN KEY([Status])
REFERENCES [dbo].[_Status] ([Status])
ON UPDATE CASCADE
GO
ALTER TABLE [dbo].[_Users] CHECK CONSTRAINT [FK__Users__Status]
GO
```

1.2 Thiết kế giao diện đăng nhập

Khi bạn nhấn vào một bảng chức năng đăng nhập hiện ra để bạn đăng nhập lại với tên đăng nhập khác.



Hình 3.1

1.3 Viết lệnh đăng nhập

```
try {
      conn.ConnectionString = My.Settings.mySqlConnString;
      conn.Open();
      cmd.Connection = conn;
      cmd.CommandType = CommandType.Text;

} catch (Exception ex) {
}
if (conn.State != ConnectionState.Open) {
      MessageBox.Show("Lỗi kết nối cơ sở dữ liệu!", "Lỗi kết
nối", MessageBoxButtons.OK, MessageBoxIcon.Error);
      this.Hide();
      frmConnect.ShowDialog();
```

```
        return;
}

if (string.IsNullOrEmpty(this.txtUserName.Text)) {
        this.DxErrorProvider1.SetError(this.txtUserName, "Tên đăng
nhập không được để trống");
        this.txtUserName.Focus();
        return;
}
cmd.CommandText = "Select * from _Users Where UserName=@UserName
And UserPass=@UserPass";
cmd.Parameters.Clear();
cmd.Parameters.AddWithValue("@UserName", this.txtUserName.Text);
cmd.Parameters.AddWithValue("@UserPass",
Des.EnCrypt(this.txtUserPass.Text));
SqlDataReader dr = cmd.ExecuteReader;
if (dr.Read) {
        UserID = dr("UserID");
} else {
        MessageBox.Show("Tên đăng nhập hoặc mật khẩu không hợp
lệ!", "Đăng nhập thất bại", MessageBoxButtons.OK,
MessageBoxIcon.Stop);
}
dr.Close();

if (UserID != 0) {
        if (GetCurrentUser("Status") < 0) {
                MessageBox.Show("Tên đăng nhập này đã bị khóa!" +
Strings.Chr(13) + "Hãy liên hệ với người quản trị.", "Đăng nhập
thất bại", MessageBoxButtons.OK, MessageBoxIcon.Stop);
                return;
        }

        KiemTraChucNang(GetCurrentUser("RoleLevel"));
        this.Close();
```

```
}
```

Trong đó hàm KiemTraChucNang() sẽ dựa vào quyền hạn của user đăng nhập để cấp quyền sử dụng menu tương ứng với chức danh của nhân viên đó.

2. Tạo form giao diện chính của chương trình

2.1 Thiết lập các thuộc tính của form chính

StartPosition: CenterParent

Text: Phần mền quản lý nhân sự

Window State: Maximized

2.2 Thiết kế hệ thống menu, toolbar

2.2.1 Thiết kế hệ thống menu

Hệ thống menu trong Visual Studio cũng đáp ứng khá tốt, tuy nhiên để khai thác những công cụ thiết kế giao diện mạnh nhất hiện nay, chúng tôi tích hợp công cụ DevExpres để thiết kế hệ thống menu.

2.2.2 Thiết kế hệ thống toolbar

Hệ thống toobar trong ứng dụng này cũng được chúng tôi tích hợp công cụ DevExpres vào để giao diện thêm đẹp mắt:

Hình 3.2

Hệ thống menu, toobar này được tích hợp với nhau thành một điều khiển chung.

3. Tạo các liên kết đến các chức năng trong chương trình

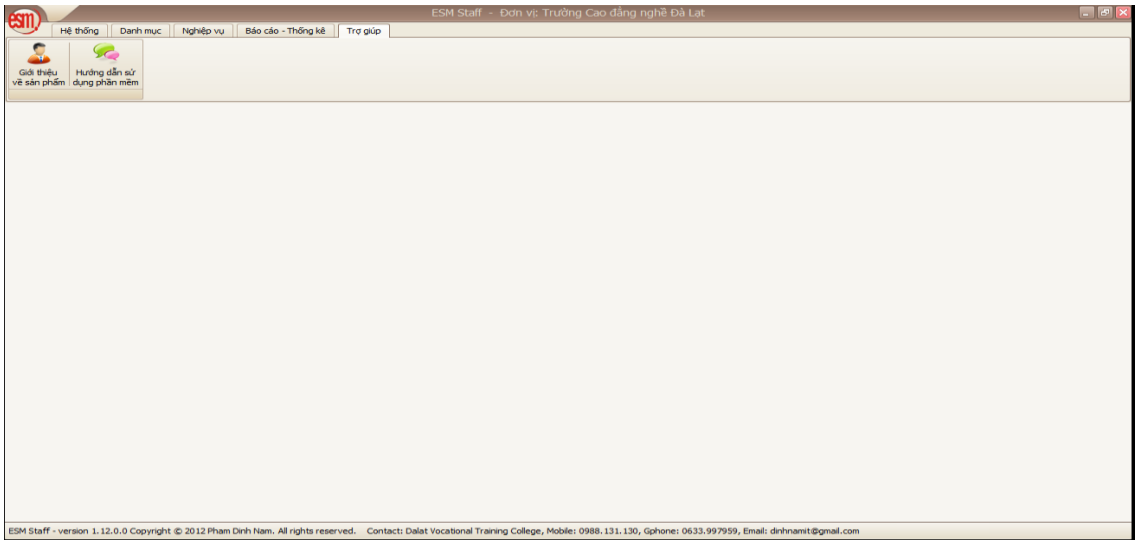
3.1 Chức năng mở form

Để mở các form từ hệ thống menu hoặc toolbar của phần mềm, chúng ta lần lượt viết code bằng cách nhắp đôi chuột vào menu hoặc toolbar chức năng tương ứng và thực hiện lệnh sau:

```
<tên form> f = new <tên form>();
f.Show();
```

3.2 Chức năng trợ giúp

Để hỗ trợ cho người sử dụng, chúng ta cần thêm hai menu về việc giới thiệu phần mềm và hướng dẫn sử dụng.



Hình 3.3

Trong menu trợ giúp này, phần mềm sẽ cung cấp cho người sử dụng thông tin về tác giả, thông tin phần mềm và hướng dẫn sử dụng phần mềm.

3.3 Chức năng thiết lập hệ thống

Khi bạn thiết lập hệ thống, bạn điền thông tin mà bạn cần thiết lập, thông thường là hệ thống đã được người quản trị thiết lập mặc định, khi bạn điền thông tin xong bạn nhấp vào nút Đồng ý

để thực thi hoặc là Bỏ qua để bỏ qua,

Hình 3.4

BÀI 4: THIẾT KẾ CÁC BÁO CÁO

Mã Bài: MĐ30_04

Giới thiệu:

Thiết kế báo cáo để in ấn là một phần không thể thiếu của phần mềm, dù công cụ Crystal Report của Microsoft cũng được chúng tôi giới thiệu trong bài này, tuy nhiên trong nội dung bài này chúng tôi khuyến khích sử dụng công cụ XtraReport của DevExpress để khai thác những tinh hoa của công cụ đó.

Mục tiêu:

- Sử dụng được một số công cụ thiết kế báo cáo;
- Xác định nguồn dữ liệu, các phương pháp xử lý dữ liệu tránh sai lệch cho kết quả báo cáo, thống kê;
- Xây dựng giao diện báo cáo thân thiện, lôgic phù hợp với trình tự công việc;
- Có thái độ tỉ mỉ, chính xác, khoa học.

Nội dung chính:

1. Xác định chức năng của báo cáo

1.1 Chức năng in danh sách

Với phương pháp tạo báo cáo như trình bày ở trên, chúng ta phải có hệ thống menu in danh sách từng danh mục khi có yêu cầu.

1.2 Chức năng in thống kê, tổng hợp

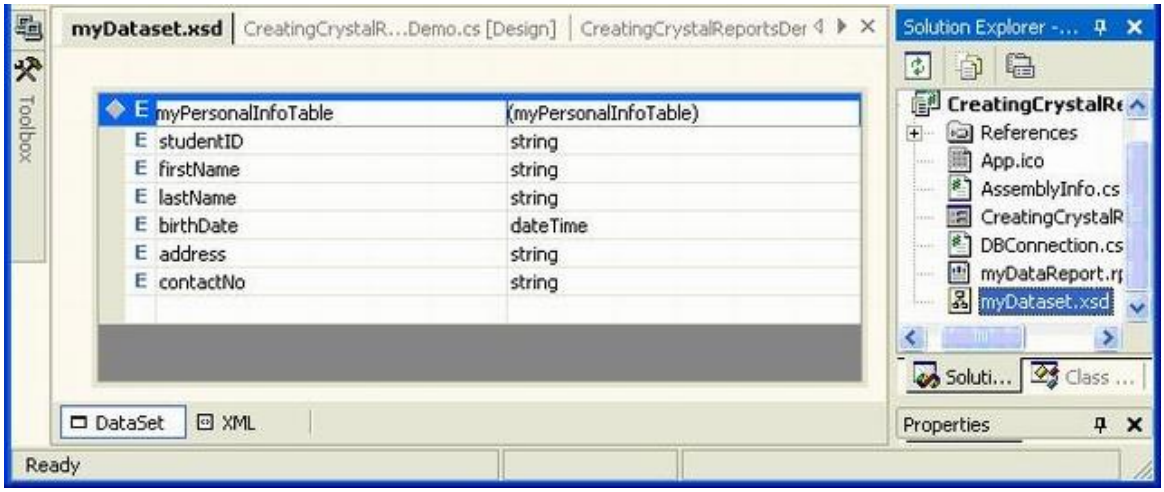
Hình 4.1

Trong hệ thống phần mềm này, chúng tôi tổng hợp các báo cáo gồm: Lý lịch trích ngang giáo viên cơ hữu, Lý lịch trích ngang giáo viên thỉnh giảng, Năng lực giáo viên – cán bộ giảng dạy, Danh sách giáo viên đi học, Danh sách nhân viên đã nâng lương trước thời hạn, Danh sách nhân viên đến hạn nâng lương, …

2. Xác định, tạo nguồn dữ liệu cho báo cáo

2.1 Xây dựng nguồn dữ liệu ảo để thiết kế báo cáo

Để thuận lợi cho việc thiết kế các báo cáo đầu tiên vào menu Project → "Add New Item" → add a Dataset để xây dựng các trường cần in tương ứng với từng báo cáo:



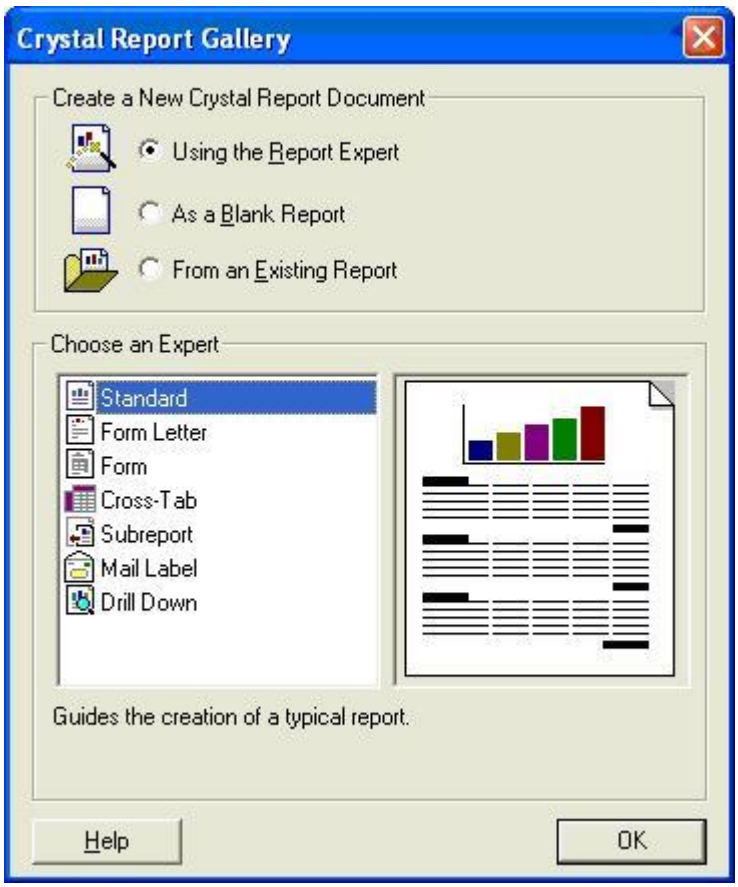
2.2. Xác định nguồn dữ liệu thật để báo cáo

Để có nguồn dữ liệu thật cho báo cáo, giải pháp của chúng tôi đưa ra là viết tất cả các truy vấn tổng hợp ở dạng thủ tục (Stored Procedures) của SQL Server.

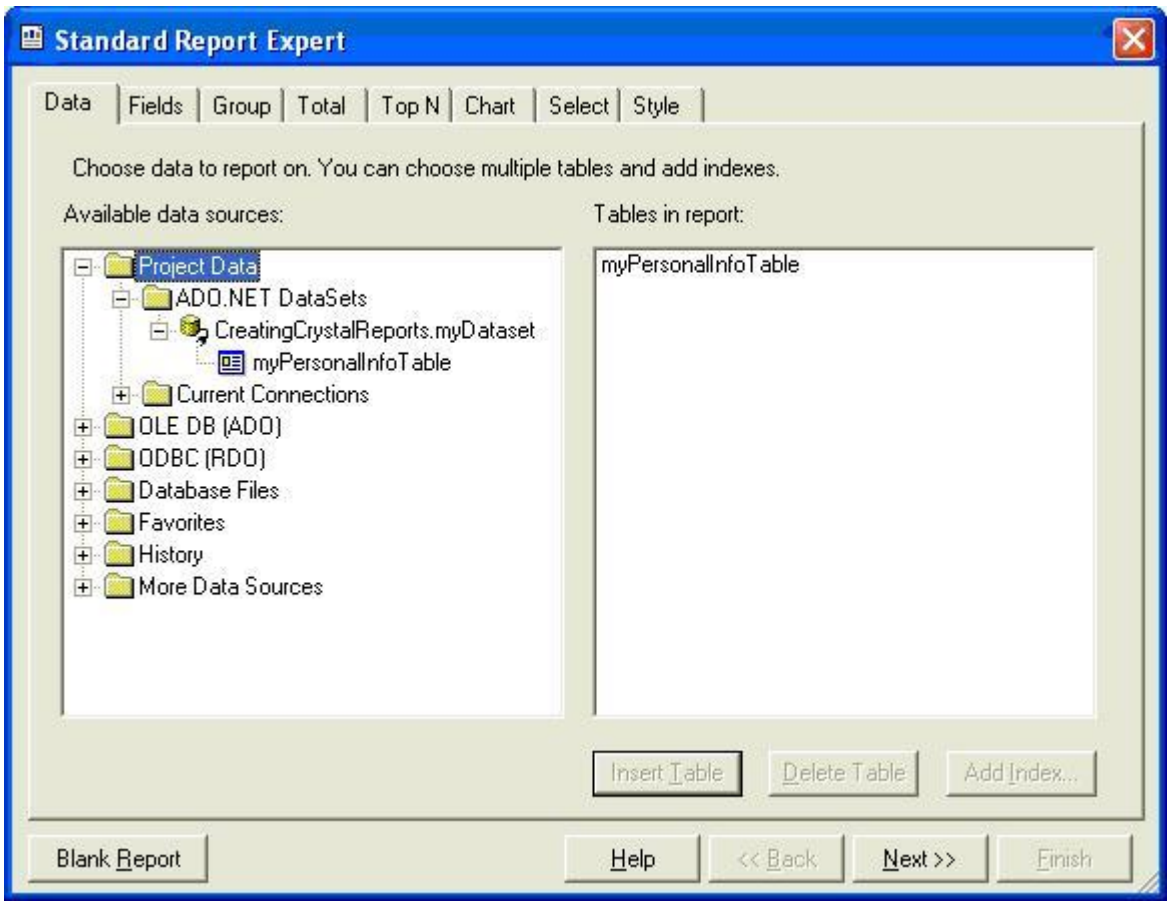
3. Tạo báo cáo với công cụ

3.1 Sử dụng công cụ Report Wizard để tạo báo cáo

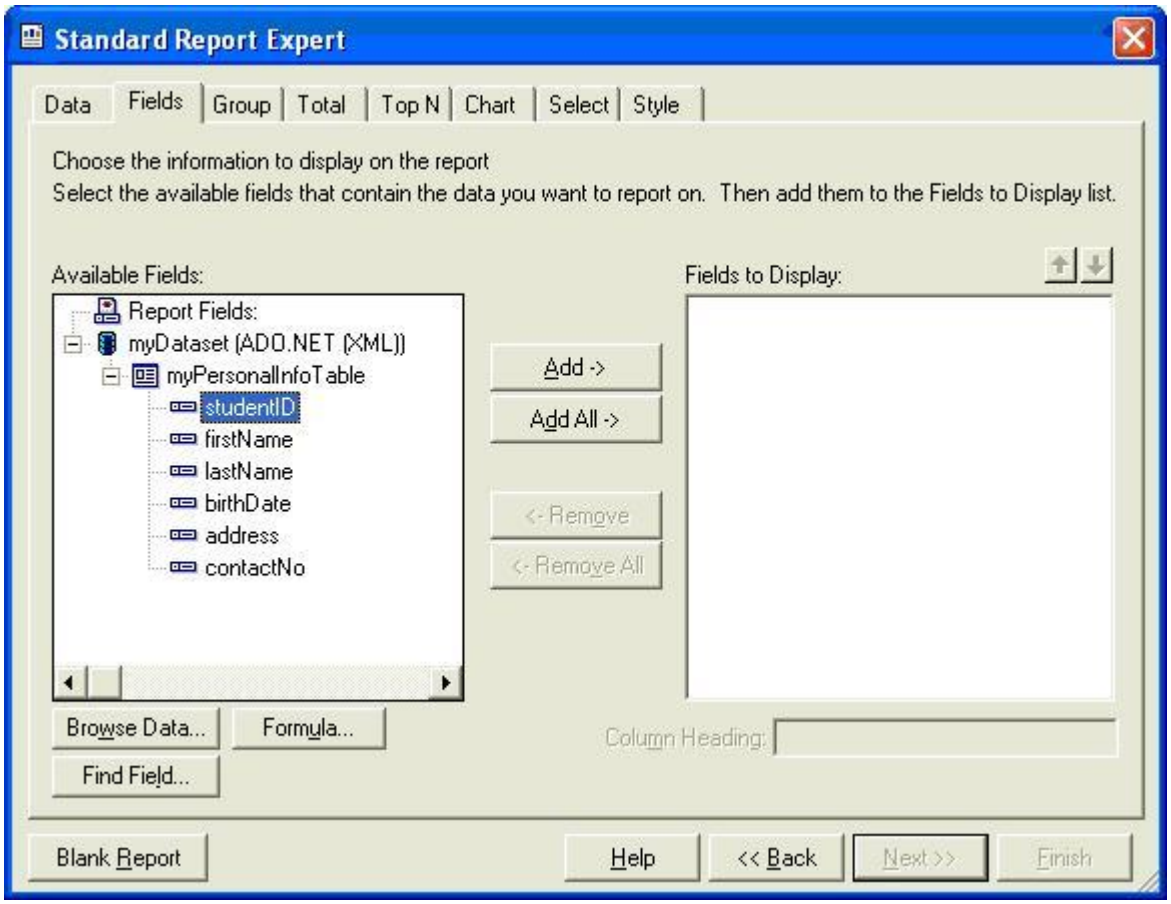
Sau khi tao xong dataset bạn add a Crystal Report vào project bằng cách "Add New Item" trong "Project" menu. Sau đó chọn tên xuất hiện rồi nhấn nút ok.



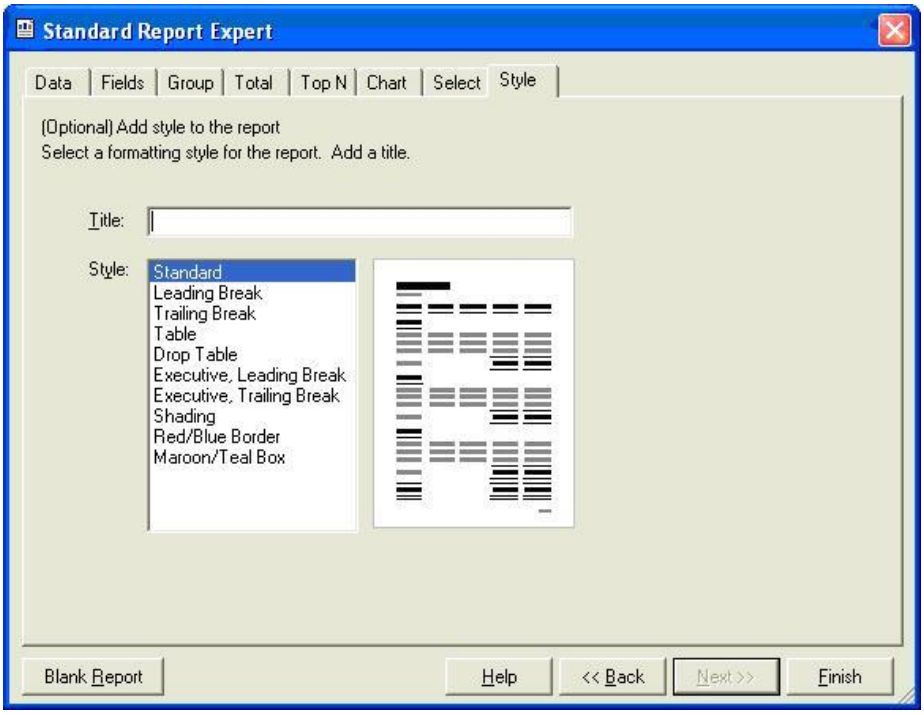
Màn hình sẽ hiển thị như sau bạn chọn mặt định rồi ấn OK Sau đó chọn table rồi insert table.



sau đó chọn các fields hiển thị trong report --> nhần nút add



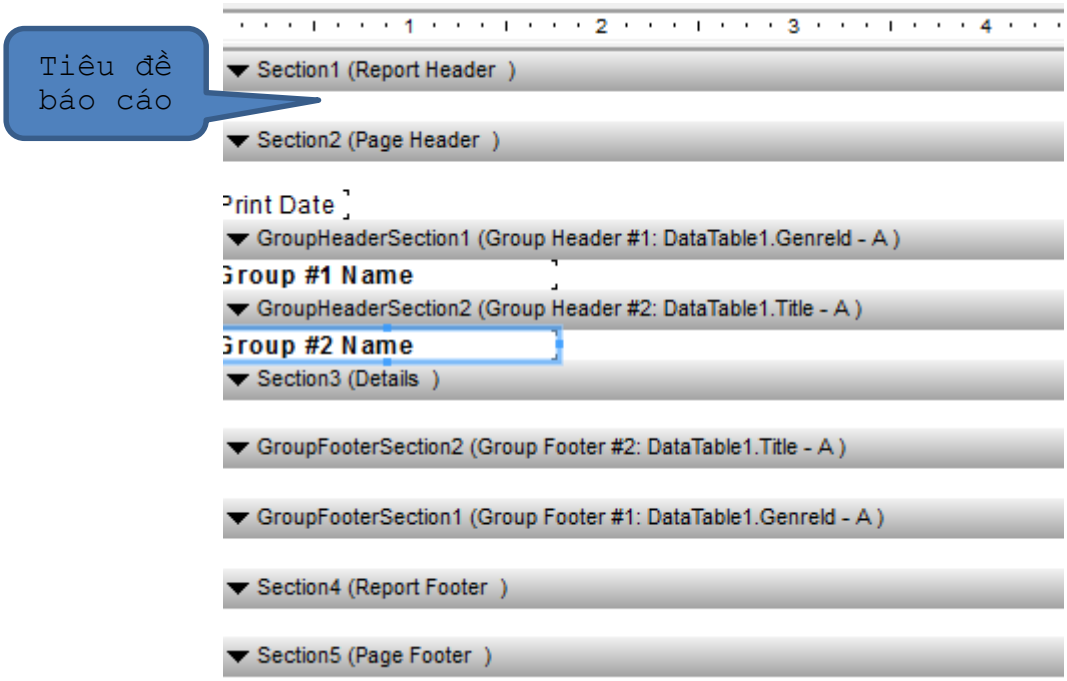
Xong xu bạn nhấn nút next-->cuối cùng chọn style kiểu bạn muốn report hiển thị



Kế tiếp bạn sẽ cho report hiển thị trên form

3.2 Thêm tiêu đề cho báo cáo

Trong trường hợp muốn thêm tiêu đề cho báo cáo thì thêm phần Report Header (Session 1) vào báo cáo.



4. Chỉnh sửa các thành phần, thuộc tính trong báo cáo

4.1 Định dạng ngày trong báo cáo

Để định dạng ngày trong Crystal Report bạn nháy chuột phải vào ô cần sửa và chọn Format Field, sau đó Customize Format.

Ví dụ: 3/13/2009 11:24:16PM cần chuyển về dạng dd/MM/yyyy

Tab DateTime chọn Order=Date,Tab Date chọn Order → DMY

+ Chọn lại phần format cho ngày tháng 2 số → OK

100,000,000.00 → cần loại bỏ từ dấu chấm và phần thập phân Tab Number Bỏ phần Thousands Separator, phần Decimal chọn 1 → OK.

4.2 Kẽ viền cho báo cáo

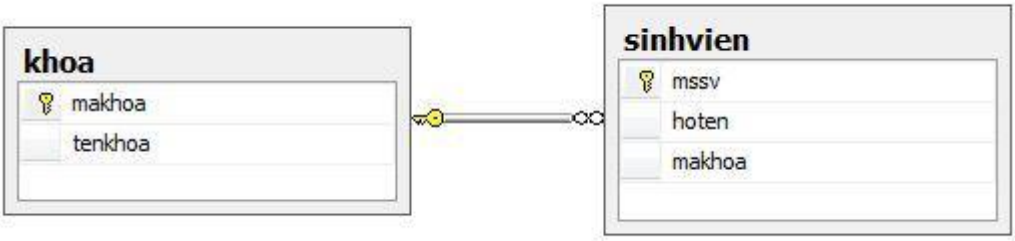
Để tạo viền cho báo cáo chúng ta có thể sử dụng đối tượng line để vẽ các đường ngang, dọc bao quanh các TextObject. Đây là một điểm yếu của Crystal report so với các công cụ thiết kế report của các hãng thứ ba.

5. Tạo form truyền tham số

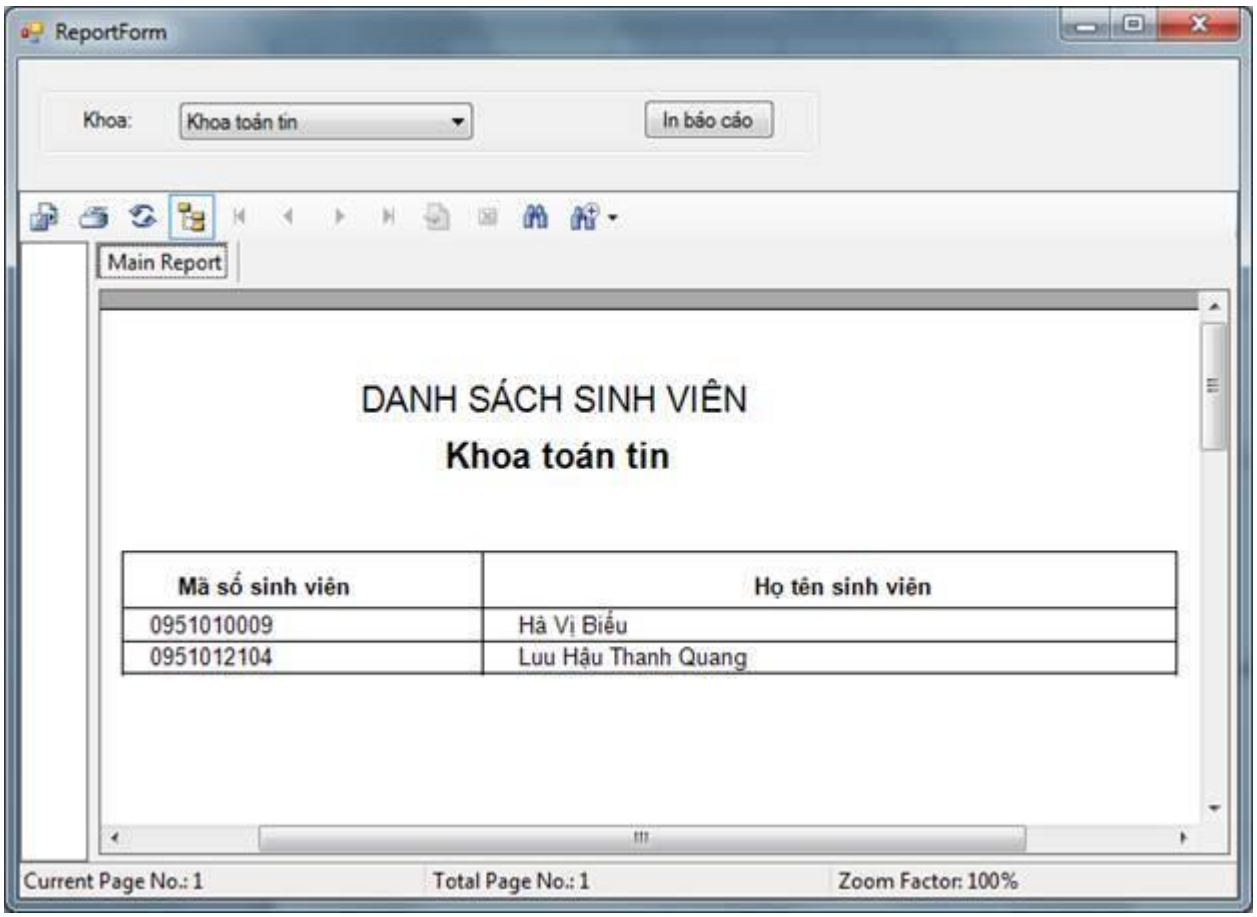
Ngoài cách sử dụng StoredProcedure mà các bạn đã biết, thì các bạn còn có thể sử dụng Parameter Field để truyền thông tin ngắn gọn dễ dàng.

Xét một ví dụ như sau:

Với cơ sở dữ liệu:



Form:

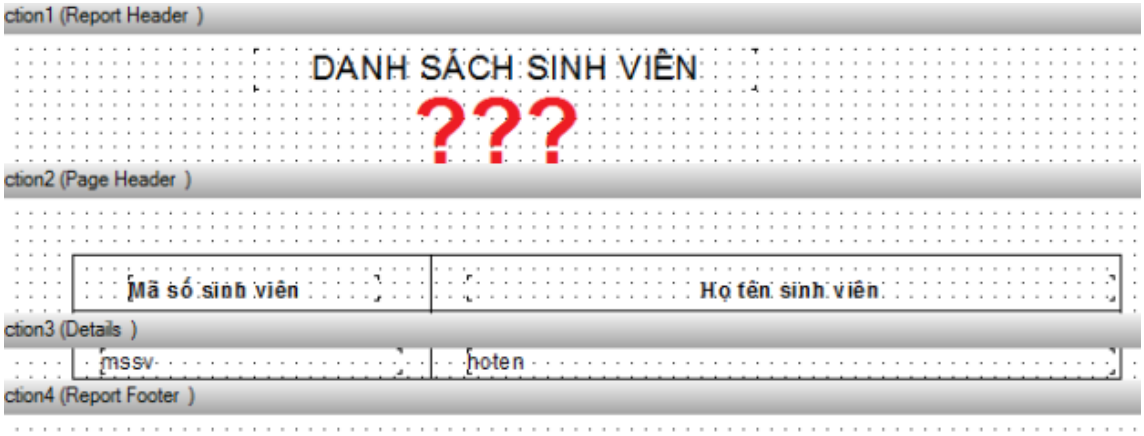


Sau khi chọn khoa trên combobox và nhấn nút In báo cáo thì báo cáo in ra sẽ có tiêu đề tên khoa tương ứng, phía dưới là danh sách sinh viên thuộc khoa đó.

Chúng ta đã quen thuộc với những bài tập tạo báo cáo dạng này bằng cách sử dụng StoredProcedure đưa dữ liệu vào. Thông thường, chúng ta sẽ tạo ngay 1 StoredProcedure lấy mã số sinh viên và họ tên sinh viên theo tham số là mã khoa, cụ thể SQL thế này:

```
CREATE PROCEDURE Sp_BCSVTHEOKHOA
      @makhoa char(4)
AS
BEGIN
      select mssv, hoten from sinhvien where makhoa = @makhoa
END
```

Sau đó là khâu thiết kế báo cáo, chúng ta gặp ngay vấn đề là tham số tên khoa không có trong StoredProcedure trên.



Vấn đề là làm thế nào thêm vào trường dữ liệu "tên khoa" đây?

Có một cách là ngay lập tức về lại bên cơ sở dữ liệu sửa lại StoredProcedure đó và kế tiếp là verify bên crystalreport. Chấp nhận rằng chúng ta sẽ truy vấn dư thừa.

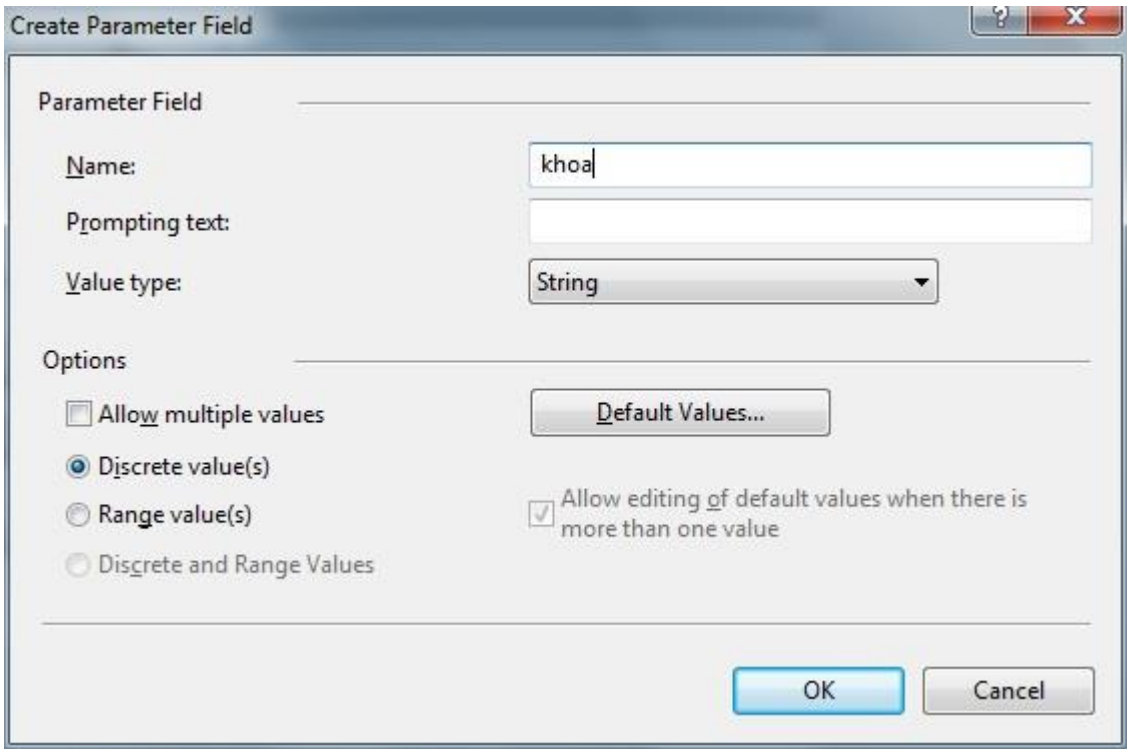
	tenkhoa	mssv	hoten
1	Khoa thủy điện	09501012XXZ	Yugi Mutou
2	Khoa thủy điện	0951012XXX	Nguyễn Thủy Thủ
3	Khoa thủy điện	0951012XXY	Võ Thị Thủy Trang

Dư

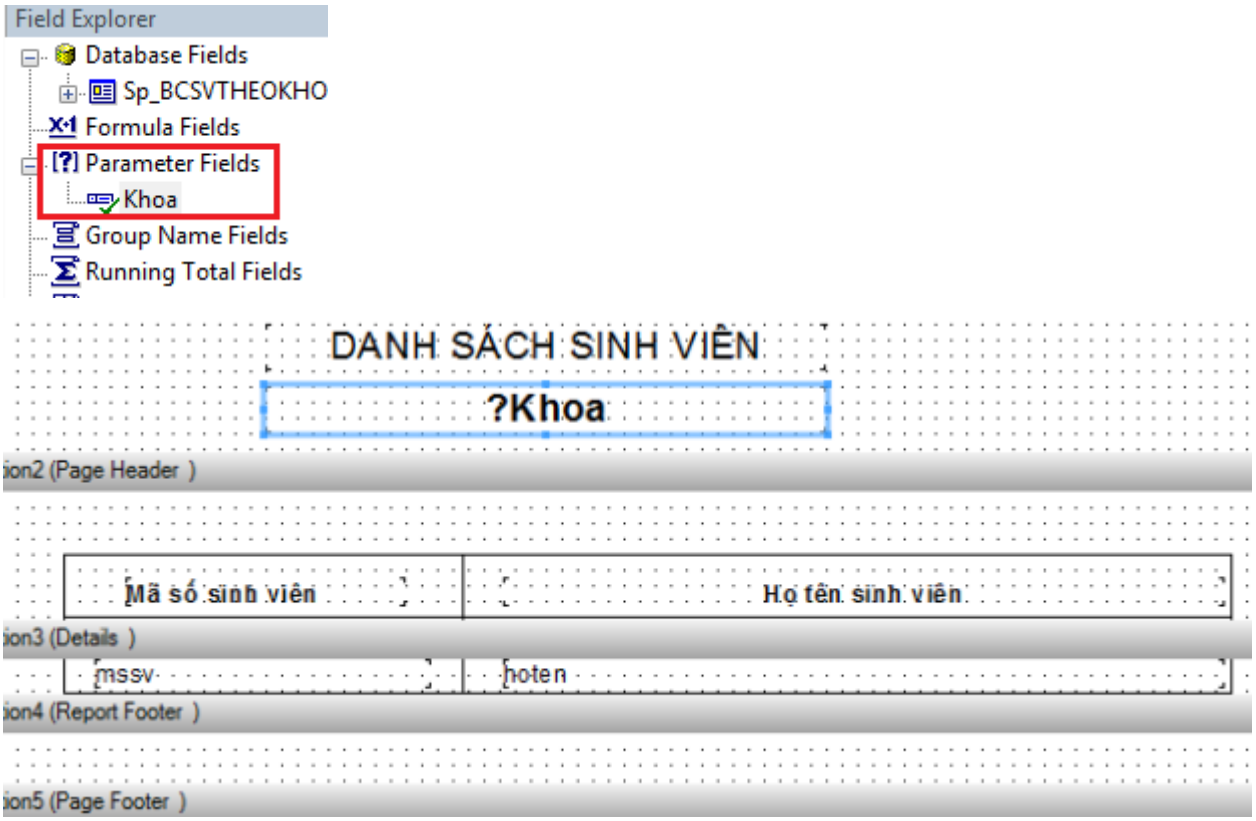
Cách khác đó là truyền "tên khoa" từ form vào crystal report theo đường Parameter Field.

Cách thực hiện như sau:

Đầu tiên, tôi vào Report, bên cửa sổ Field Explorer, tạo mới một Parameter Field



Đặt tên là "khoa". Tôi bỏ trống phần prompting text, vì tôi không có dự định sử dụng prompt để người dùng nhập vào parameter, mà tôi sẽ truyền vào bằng code C#. Sau khi tạo xong, thì kéo Field Khoa đến vị trí cần hiển thị.



Lưu lại và về Form để thực hiện gán vào Parameter đó khi bấm vào nút in:

```
        private void btnPrintReport_Click(object sender,
EventArgs e)
        {
            clssForm1BLL.MaKhoa =
cboKhoa.SelectedValue.ToString();
            crystalReportSVTHEOKHOA1.SetDataSource(clssForm1BLL.
GetSinhVienByMaKhoa());

            ParameterDiscreteValue pa = new
ParameterDiscreteValue();
            pa.Value = cboKhoa.Text;

            ParameterFieldDefinition crParameterFieldDefinition
=
crystalReportSVTHEOKHOA1.DataDefinition.ParameterFields["khoa"];
            crParameterFieldDefinition.CurrentValues.Clear();
            crParameterFieldDefinition.CurrentValues.Add(pa);
            crParameterFieldDefinition.ApplyCurrentValues(crPara
meterFieldDefinition.CurrentValues);

            crystalReportViewer1.ReportSource =
crystalReportSVTHEOKHOA1;
        }
```

Những người bạn mới ở đây là:

– ParameterDiscreteValue : khai báo đối tượng thuộc lớp này để chứa dữ liệu rời rạc. Trong ví dụ này thì là tên khoa đang chọn được hiển thị trên combobox.

– ParameterFieldDefinition : khai báo đối tượng thuộc lớp này để tìm đến định nghĩa ParameterField muốn truyền trên báo cáo. Đó chính là ParameterField có tên là "khoa" (giống với tên ParameterField ban nãy tạo bên báo cáo). Sau đó là vài động tác xóa, thêm, sử dụng dữ liệu ta đã truyền vào theo đối tượng ParameterDiscreteValue.

6. Xử lý tham số với báo cáo

Chúng ta có thể viết ngắn gọn lại theo cách này:

```
        private void btnPrintReport_Click(object sender,
EventArgs e)
        {
            clssForm1BLL.MaKhoa =
cboKhoa.SelectedValue.ToString();
            crystalReportSVTHEOKHOA1.SetDataSource(clssForm1BLL.
GetSinhVienByMaKhoa());

            ParameterDiscreteValue pdv = new
ParameterDiscreteValue();
            pdv.Value = cboKhoa.Text;
            ParameterValues pv = new ParameterValues();
            pv.Add(pdv);
crystalReportSVTHEOKHOA1.DataDefinition.ParameterFields["khoa"].
ApplyCurrentValues(pv);

            crystalReportViewer1.ReportSource =
crystalReportSVTHEOKHOA1;
        }
    }
```

7. Thiết kế các báo cáo chi tiết

7.1 Tạo thủ tục SQL để kết xuất dữ liệu

Tất cả công việc kết xuất dữ liệu được thực hiện trong thủ tục của SQL Server. Chẳng hạn với việc kết xuất dữ liệu lý lịch cá nhân, chúng ta có thể có thủ tục sau:

```
Create PROCEDURE [dbo].[usp_LyLichCaNhan]
(
@NhanVienID int
)

AS

SELECT
      [NhanVienID],
      tblPhongBan.[MaPhongBan],
```

```
        TenPhongBan,
        UPPER([HoTen]) as HoTen,
        [TenGoiKhac],
        [NgaySinh],
        case when [GioiTinh]=1 then NgaySinh else null end as
GioiTinhNam,
        case when [GioiTinh]=0 then NgaySinh else null end as
GioiTinhNu,
        case when [GioiTinh]=0 then N'Nữ' else N'Nam' end as
GioiTinh,
        [HinhAnh],
        [NoiSinhXa]+ case when [NoiSinhHuyen]<>'' then ' - '+
[NoiSinhHuyen] end + case when [NoiSinhTinh]<>'' then ' -
'+[NoiSinhTinh] end as NoiSinh,
        [QueQuanXa]+ case when [QueQuanHuyen]<>'' then ' - '+
[QueQuanHuyen] end + case when [QueQuanTinh]<>'' then ' -
'+[QueQuanTinh] end as QueQuan,
        [DanToc],
        [TonGiao],
        [HoKhauThuongTru],
        [DiaChiHienNay],
        [NgheNghiepKhiDuocTD],
        [NgayTuyenDung],
        [CoQuanTuyenDung],
        [ChucVuHienNay],
        [NgayDuocBoNhiem],
        [CongViecChinhDuocGiao],

        dbo.GetChucVu(NhanVienID,1) as ChucVuCaoNhat,
        dbo.GetTrinhDoCaoNhat(NhanVienID,'01PT') as
TrinhDoPhoThong,
        dbo.GetTrinhDoCaoNhat(NhanVienID,'02CM') as
TrinhDoChuyenMon,
        dbo.GetTrinhDoCaoNhat(NhanVienID,'03CT') as
TrinhDoChinhTri,
```

```
        dbo.GetTrinhDoCaoNhat(NhanVienID,'04QL') as TrinhDoQLNN,
        dbo.GetTrinhDoCaoNhat(NhanVienID,'05NN') as
TrinhDoNgoaiNgu,
        dbo.GetTrinhDoCaoNhat(NhanVienID,'06TH') as TrinhDoTinHoc,
        dbo.GetTrinhDoCaoNhat(NhanVienID,'07SP') as TrinhDoSuPham,
        dbo.GetTrinhDoCaoNhat(NhanVienID,'08HC') as TrinhDoQLHC,
        dbo.GetTrinhDoCaoNhat(NhanVienID,'09LX') as TrinhDoGPLX,

        dbo.GetKhenThuongCaoNhat(NhanVienID) as KhenThuongCaoNhat,
        dbo.GetKyLuatCaoNhat(NhanVienID) as KyLuatCaoNhat,
        tblNgachCCVC.MaLoaiCCVC,
        tblNgachCCVC.TenNgachCCVC,
        tblNhanVien.MaNgachCCVC,
        [BacLuong],
        dbo.LayTenBacLuong(tblNhanVien.MaNgachCCVC,BacLuong) as
BacTrongNgach,
        (1+VuotKhung/100)*dbo.LayHeSoLuong(tblNhanVien.MaNgachCCVC
,BacLuong) as HeSoLuong,
        [NgayHuong],
        [NgayNangLuongLanSau],
        [NgayNangLuongTruocThoiHan],
        [PhuCapChucVu],
        [PhuCapKhac],
        case KieuHopDong when 0 then N'Biên chế' when 1 then N'Hợp
đồng' when 2 then N'Mời giảng' end as LoaiHopDong,
        [NgayKyHopDong],
        NgayThoiViec,
        [ThamNienCongTac],
        [NgayVaoCoQuanHienNay],
        [NgayVaoDang],
        [NgayVaoDangChinhThuc],
        [NgayVaoDoan],
        [NgayNhapNgu],
        [NgayXuatNgu],
        [QuanHamCaoNhat],
```

```
        [DanhHieuDuocPhongTang],
        [SoTruongCongTac],
        [TinhTrangSucKhoe],
        [ChieuCao],
        [CanNang],
        [NhomMau],
        [LaThuongBinhHang],
        [LaConGiaDinhChinhSach],
        [SoCMND],
        [NgayCap],
        [NoiCap],
        [SoBHXH],
        [NhanXetCuaCoQuan],
        LastEditUser,
        case when NgayNangLuongLanSau<getdate() then N'Đã quá trể'
when NgayNangLuongLanSau between getdate() and
dateadd(month,1,getdate())  then N'Sắp tăng lýõng' else ''  end
as TinhTrangNangLuong,
        year(getdate())-year(NgaySinh) as Tuoi
FROM tblNhanVien Left Join tblPhongBan On
tblNhanVien.MaPhongBan=tblPhongBan.MaPhongBan
        Left Join tblNgachCCVC On
tblNhanVien.MaNgachCCVC=tblNgachCCVC.MaNgachCCVC
        Left Join tblLoaiHopDong On
tblNhanVien.LoaiHopDong=tblLoaiHopDong.MaLoaiHopDong
Where tblNhanVien.NhanVienID=@NhanVienID
```

7.2 Tạo báo cáo sử dụng nguồn dữ liệu từ thủ tục SQL

```
cmd.CommandType = CommandType.StoredProcedure; //cmd là
SqlCommand
cmd.CommandText = "usp_LyLichCaNhan";
cmd.Parameters.Clear() ;
cmd.Parameters.AddWithValue("@Tên_Tham_Số",Giá_trị);
da.SelectCommand = cmd ; //da là SqlDataAdapter
ds.Clear(); //ds là DataSet
```

```
da.Fill(ds, "Tên_bảng_ảo");
DataGridView1.DataSource = ds.Tables("Tên_bảng_ảo ") ;
```

BÀI 5: HOÀN THIỆN, ĐÓNG GÓI PHẦN MỀM

Mã Bài: MĐ30_05

Giới thiệu:

Công việc cuối cùng sau khi hoàn thiện sản phẩm là phải đóng gói để thuận lợi trong việc triển khai đến khách hàng.

Mục tiêu:

- Xác định được một số công cụ đóng gói sản phẩm;
- Kiểm tra, hiệu chỉnh nhằm đảm bảo phần mềm đạt được các yêu cầu đặt ra;
- Bảo mật cho cơ sở dữ liệu nhằm tránh những truy cập trái phép;
- Đóng gói phần mềm, tạo bộ cài đặt;
- Rèn luyện được nghiêm túc, cẩn thận, tỉ mỉ.

Nội dung chính:

1. Kiểm thử phần mềm

Ở đây chỉ nói đến việc Kiểm thử phần mềm cho ứng dụng máy tính và mobile, không bàn đến các nền tảng khác (nhúng, …)

Kiểm thử phần mềm (software testing) là hoạt động nhằm tìm kiếm, phát hiện các lỗi của phần mềm được được kiểm thử về thiết kế, mã nguồn, chức năng, dữ liệu, bảo mật, thân thiện với người dùng, tài liệu kèm theo, môt trường hoạt động, tốc độ hoạt động, khả năng tải của hệ thống, … Thường được chia thành các nhóm là Nhóm thuộc về chức năng (Functionality), Nhóm không thuộc chức năng (Non-Functionality), Nhóm thuộc về cấu trúc (Structural) và Nhóm liên quan đến các thay đổi (Change Related). (Tham khảo thêm ISO9126

Kiểm thử phần mềm còn hướng đến mục tiêu xa hơn mà tôi gọi nôm na là "phòng bệnh hơn chữa bệnh". Tức là nâng cao khả năng kiểm soát và hạn chế các lỗi xảy ra khi phát triển phần mềm ngay từ ban đầu, chứ không đơn thuần chỉ là việc tìm những lỗi sẵn có khi nhóm phát triển đã đưa ra những phiên bản cụ thể của phần mềm.

Kiểm thử phần mềm là một ngành không quá mới tại Việt Nam. Tuy nhiên, hình thức tồn tại chủ yếu của Kiểm thử phần mềm tại Việt Nam là nằm phụ thuộc trong một nhóm phát triển phần mềm (vừa lập trình, vừa kiểm thử bởi một nhóm phát triển). Vấn đề đảm bảo chất lượng phần mềm cũng chưa được các công ty phát triển phần mềm trong nước coi trọng đúng mức. Do đó, mặc dù đã có ở Việt Nam từ lâu, nhưng chỉ một vài năm gần đây, Kiểm thử phần mềm chuyên nghiệp mới được biết đến như một ngành riêng biệt với tiềm năng phát triển lớn. Công ty

LogiGear Việt Nam hiện là công ty hàng đầu Việt Nam trong lĩnh vực kiểm thử phần mềm chuyên nghiệp

2. Hiệu chỉnh phần mềm

2.1 Xác định nguyên nhân gây lỗi

Có nhiều thuật ngữ tiếng Anh có thể dịch sang tiếng Việt là "lỗi". Tuy nhiên, ý nghĩa của nó trong Kiểm thử hoàn toàn khác nhau:

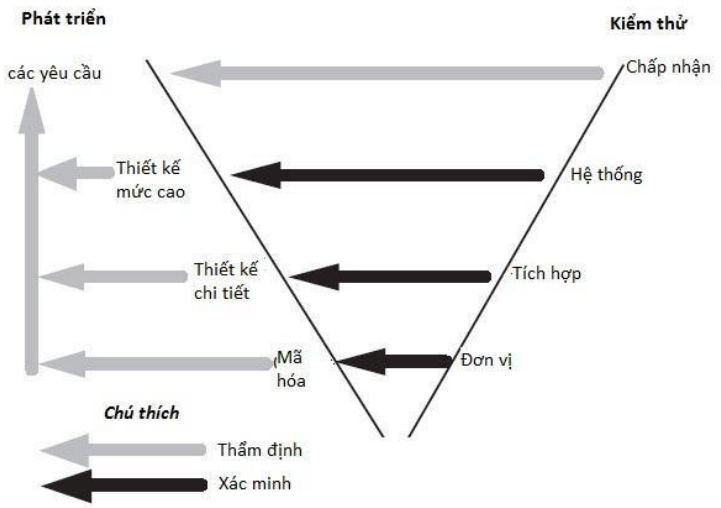
Error, Mistake: Thường là chỉ một lỗi của con người trong quá trình xây dựng phần mềm.

Bug, Defect, Fault: Là lỗi nằm trong mã nguồn, tài liệu của chương trình. Loại lỗi này có nhiều nguyên nhân như: do error của con người, do công nghệ phức tạp, áp lực công việc, do các thành phần của hệ thống tương tác với nhau, … Kiểm thử viên chủ yếu là bắt các loại lỗi này.

Fail, Failure: Dùng để chỉ các lỗi dưới góc độ của hệ thống. Khi một hệ thống không thực hiện được chức năng cần thiết, hoặc thực hiện chức năng không được phép làm thì được gọi là fail/failure. Một bug có thể là nguyên nhân của nhiều fail khi hệ thống hoạt động.

2.2 Khắc phục lỗi

Các giai đoạn kiểm thử tương ứng với các giai đoạn khác nhau trong tiến trình phát triển phần mềm và được khái quát hóa qua mô hình chữ V như hình dưới gồm:



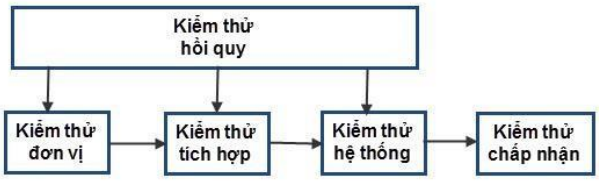
Các giai đoạn phát triển và kiểm thử trong mô hình chữ V

1) Kiểm thử đơn vị

2) Kiểm thử tích hợp

3) Kiểm thử hệ thống

4) Kiểm thử hồi quy



Kiểm thử hồi quy tại các mức kiểm thử phần mềm khác

5) Kiểm thử chấp nhận

3. Xác định công cụ đóng gói

Build và Release Version đều được dùng để chỉ một phiên bản của phần mềm. Tuy nhiên, ý nghĩa và trường hợp sử dụng thì khác nhau:

Build: Thường được dùng để chỉ phiên bản phần mềm trong quá trình phát triển tại nhóm dự án. Những thay đổi về chức năng giữa các build liên tiếp nhau thường là nhỏ (có thể là thêm một vài chức năng, sửa một số lỗi, tối ưu mã nguồn, …). Các build được tạo ra trong quá trình phát triển phần mềm của nhóm dự án. Một ngày có thể có một hoặc nhiều build.

Release Version: Được dùng để chỉ một bản build. Tuy nhiên, bản build này sẽ được gởi đến cho khách hàng kiểm tra chấp nhận. Những thay đổi giữa các Release Version liên tiếp nhau thường là khá lớn. Phải có nhiều build được viết và kiểm thử tại nhóm dự án thì mới có một Release Version.

Trong bài này chúng ta chọn công cụ đóng gói của chính bộ phần mềm Microsoft Visual Studio .NET.

4. Đặt bảo mật cho cơ sở dữ liệu

4.1 Xây dựng thư viện mã hóa

Để đảm bảo về tính bảo mật, ngoài việc phải mã hóa mật khẩu trong code trước khi lưu vào cơ sở dữ liệu thì các thủ tục (Stored Procedures) và hàm (Functions) trong SQL cũng phải được mã hóa để tránh người sử dụng can thiệp gây ra những lỗi không mong muốn.

4.2 Đặt mật khẩu cho cơ sở dữ liệu và code.

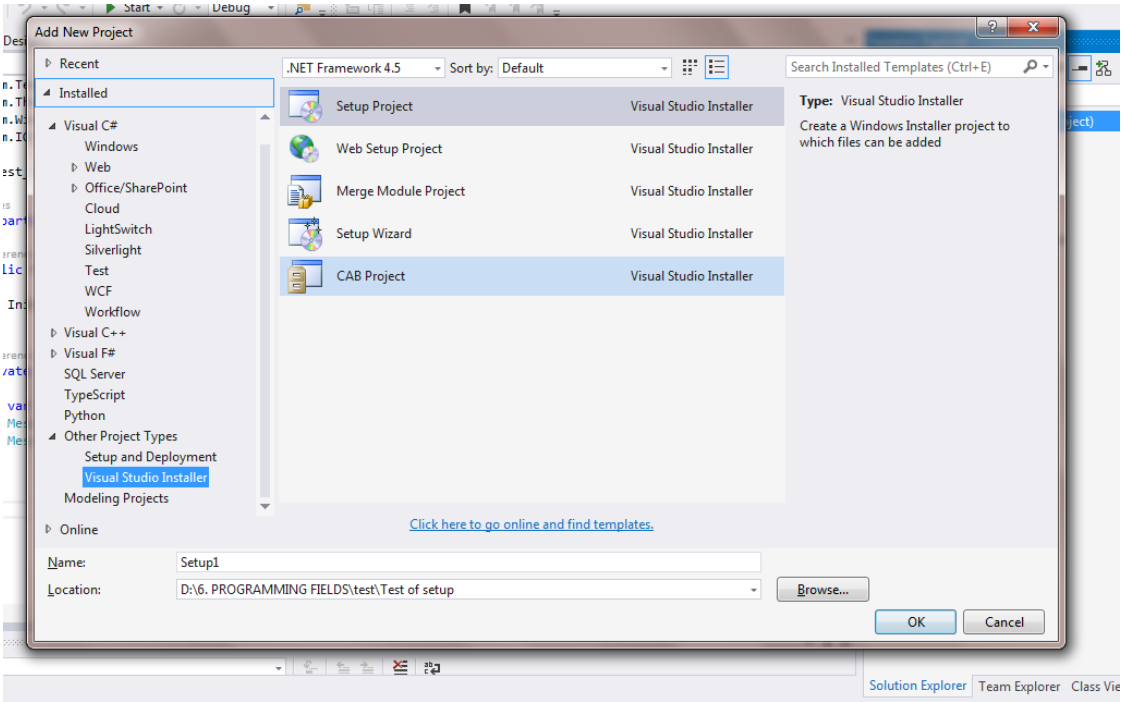
Với cơ sở dữ liệu SQL Server, để đảm bảo an toàn chúng ta không sử dụng user đăng nhập mặc định là sa mà nên tạo một user riêng, đồng thời tạo mật khẩu cho user này để đăng nhập cơ sở dữ liệu.

Đối với các user đăng nhập trong phần mềm cũng nên tạo mật khẩu mặc định không được quá đơn giản, tránh người sử dụng có thể đoán được. Đặc biệt không nên để trống.

5. Đóng gói phần mềm

B1. Tạo Setup project

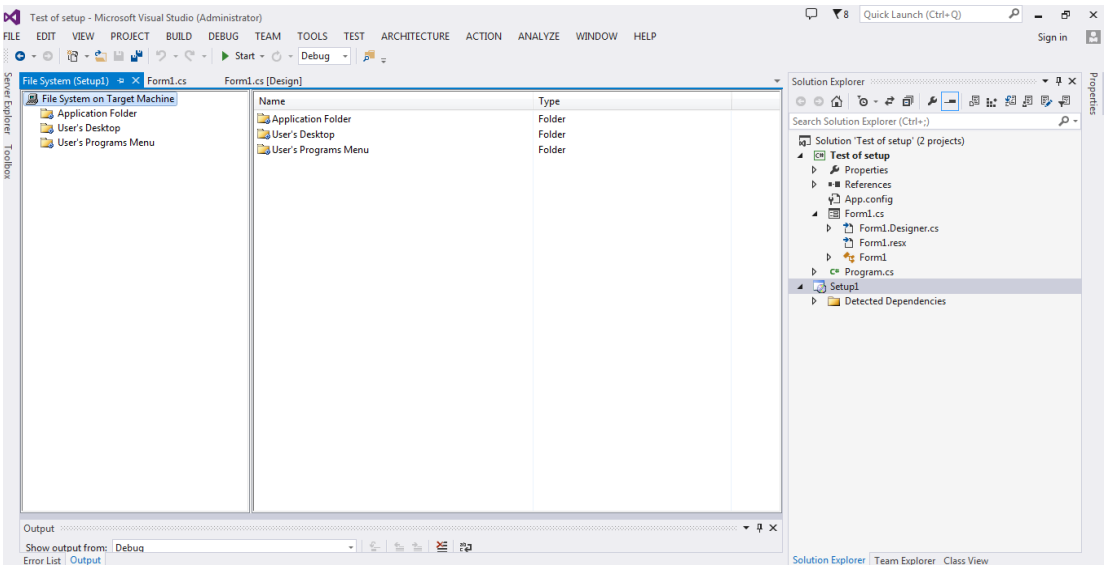
Bây giờ chúng ta sẽ tạo Setup project cho chương trình vừa viết ở trên. Chuột phải vào Solution "Test of Setup" => Add => New Project => Other project type => Setup Project:



Chú ý: đối với những phiên bản VS 2012 trở đi, chúng ta phải lên mạng và tìm cách down thêm Visual Studio Installer và cài đặt thì mới xuất hiện Setup Project trong Other Project Type được. Tải Visual Studio Installer TẠI ĐÂY.

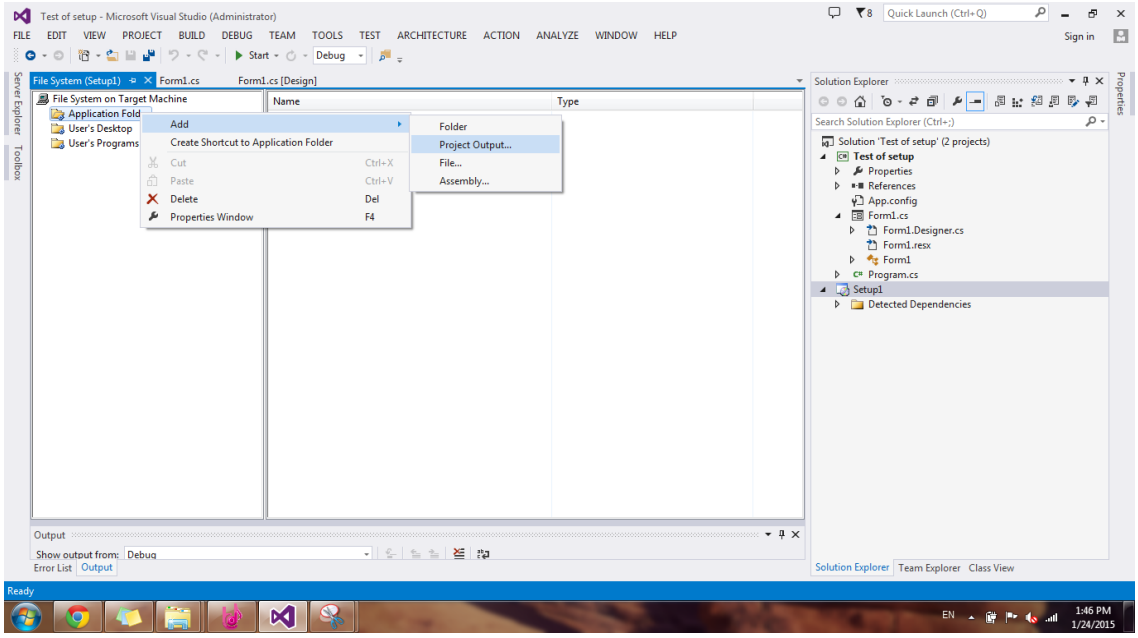
Tôi giữ nguyên tên là Setup1 cho nó đỡ lằng nhằng.

Kết quả, giao diện như sau:

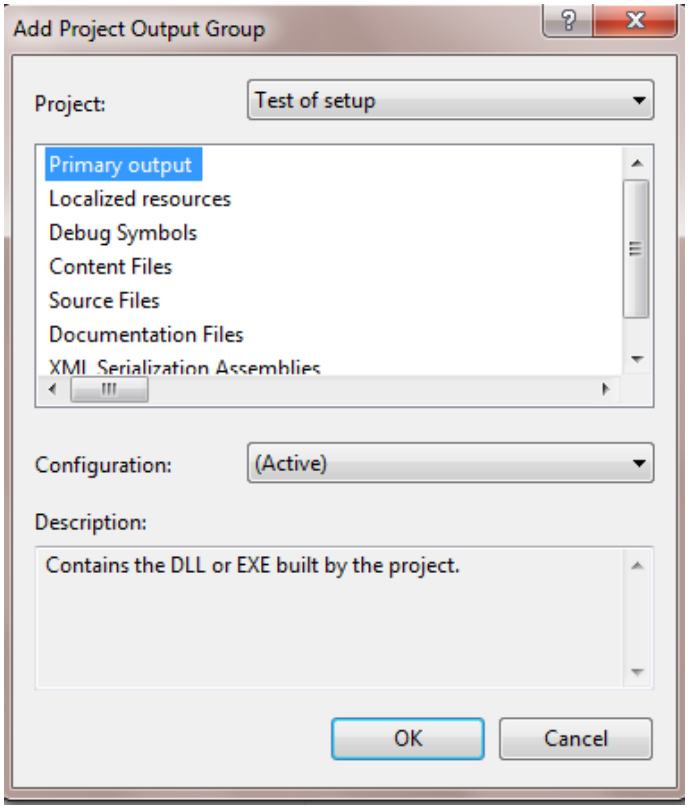


B2. Các bước cài đặt cơ bản

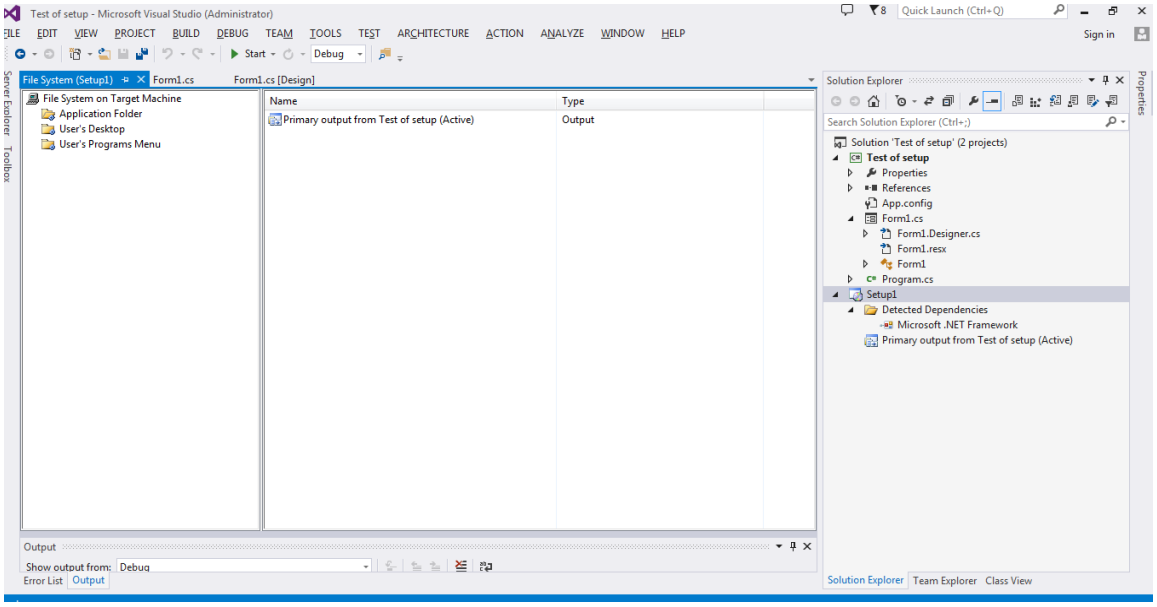
Bây giờ chúng ta cần tạo một Output cho setup. Chuột phải vào Application Folder => Add => Project Output... như bên dưới:



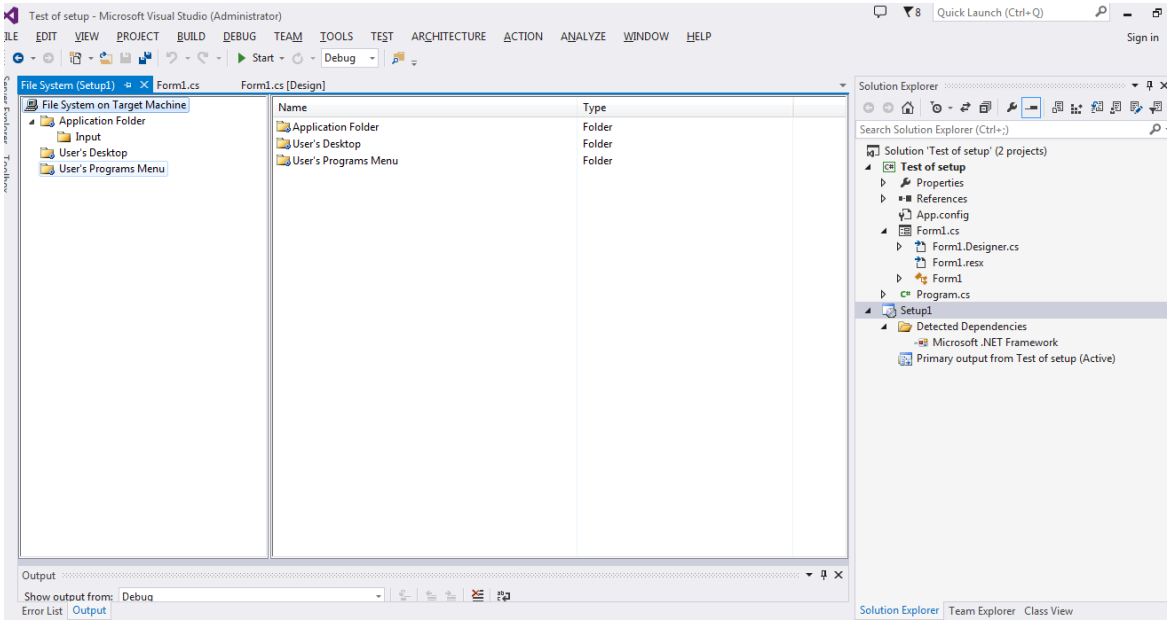
Sau đó chọn Primary Output => Ok:



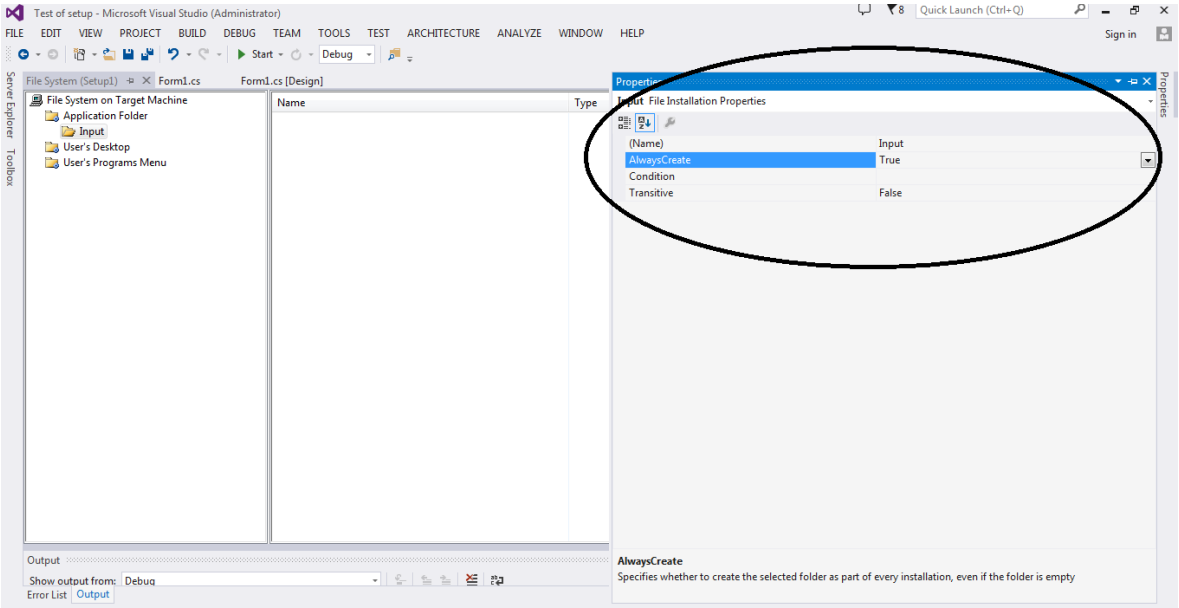
Và kết quả như sau:



Trong Project, chương trình của chúng ta sẽ phải lấy thông tin từ file Sample.txt để hiển thị ra ngoài màn hình. Do đó, file Sample.txt cũng phải đi kèm với file Setup luôn, có nghĩa là chúng ta cần phải add Sample.txt vào trong Setup Project (nếu không add thì chương trình của chúng ta sẽ bị lỗi). Để Add được thì chúng ta chuột phải vào Application Folder => Add => Folder => thế là một Folder mới xuất hiện, sửa thành Input, kết quả như sau:



Chuột phải vào thư mục Input => Properties => chọn thuộc tính AlwaysCreate là True:



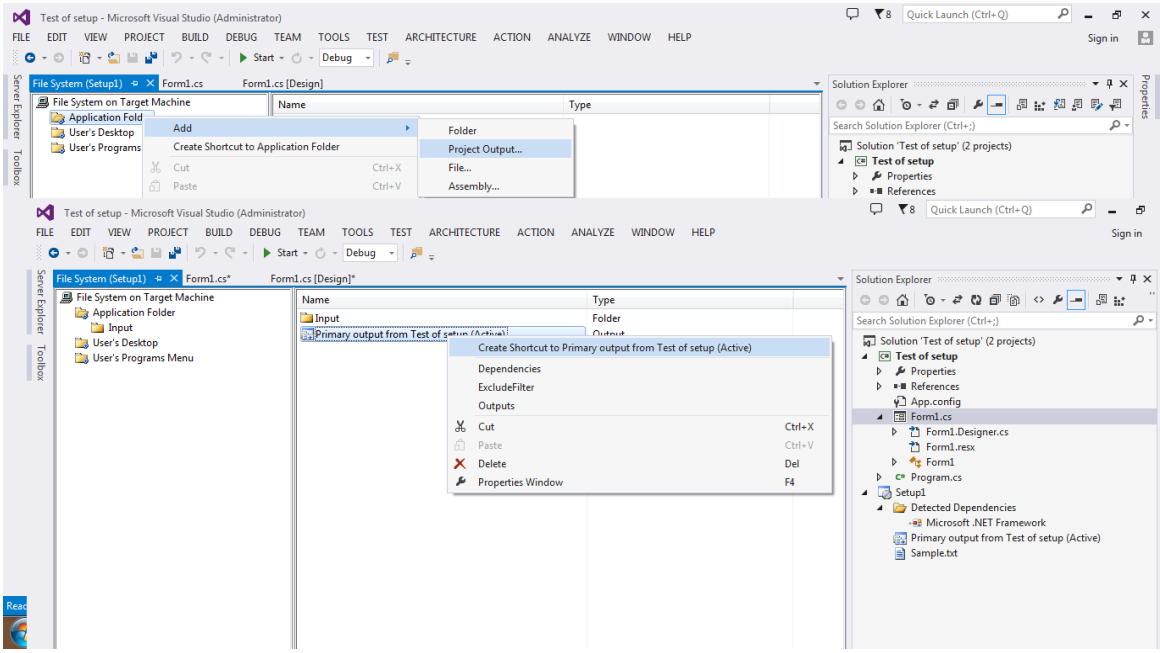
Làm như thế có nghĩa là: khi chạy file Setup thì chương trình luôn tạo một Folder tên là Input trong ổ C, còn nó ở chỗ nào thì ta chẳng quan tâm, chỉ cần biết khi mà cài xong thì chương trình cần dùng những gì từ thư mục Input đó thì nó sẽ tự biết tìm đến và lấy, thế thôi.

Chúng ta tiếp tục add Sample.txt vào trong thư mục Input để sau này khi phần mềm của mình đem sang máy khác cài thì nó còn biết đường mà tìm đến Input/Sample.txt.

Động tác Add như sau: chuột phải vào Input => Add => File => chọn đến file Sample.txt.

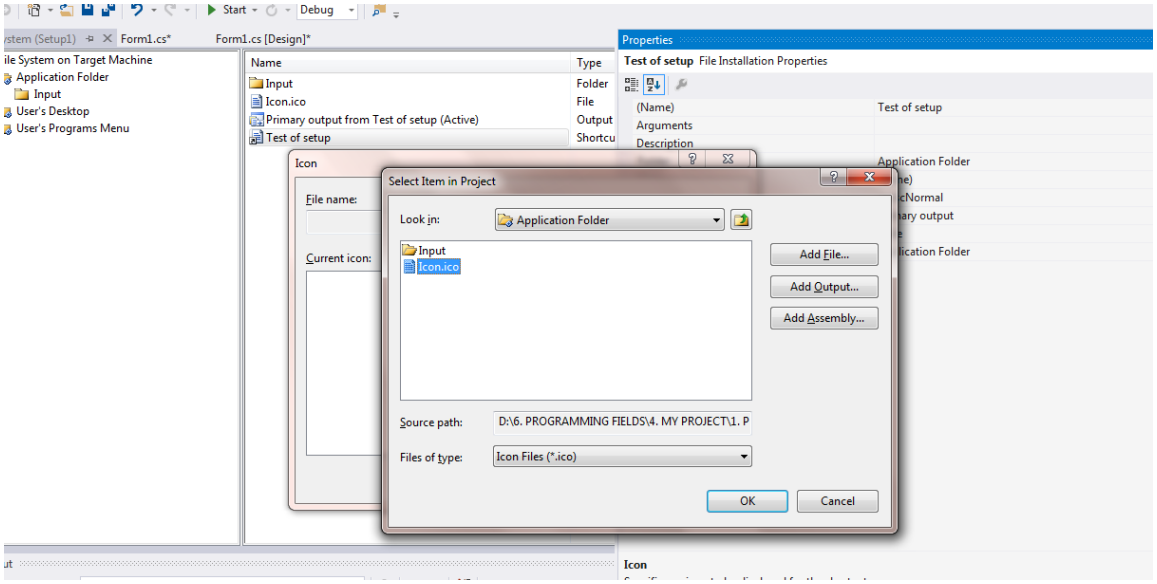
B3. Tạo Shortcut

Chuột phải vào Primary Output ở giữa màn hình và chọn Create Shortcut:



Sửa lại tên thành "Test of setup", cái shortcut này sẽ là cái xuất hiện trên nền desktop khi mà chúng ta cài đặt xong.

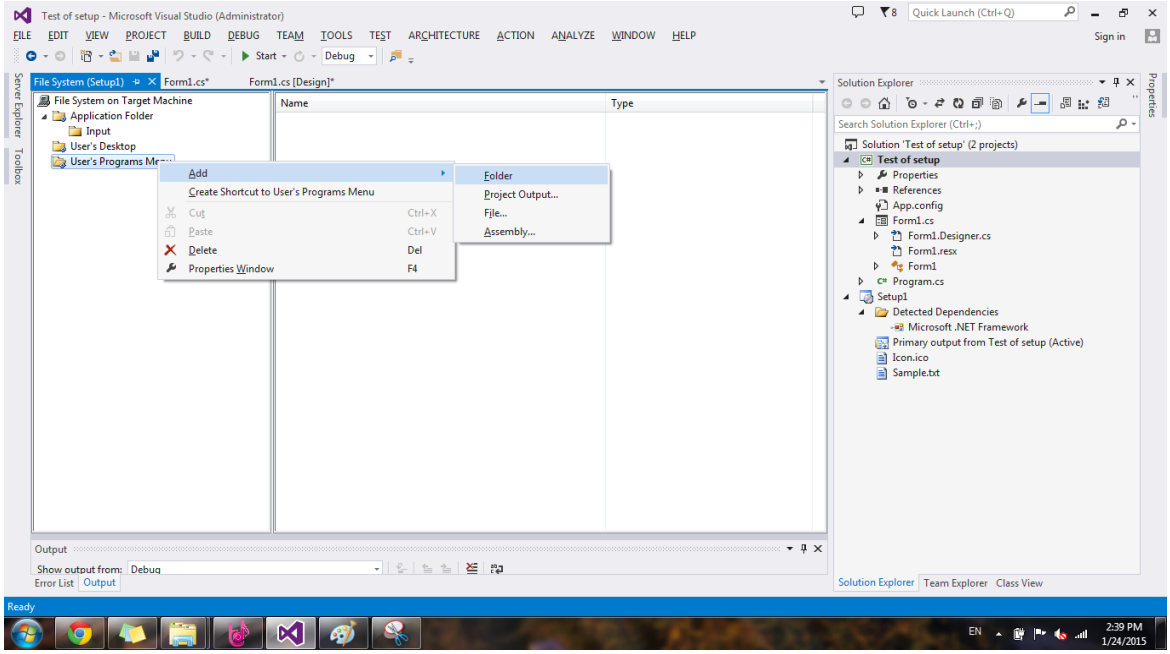
Bây giờ chúng ta cần thiết đặt Icon cho Shortcut: chuột phải vào Shortcut vừa mới tạo => Properties => trong cửa sổ Properties, chọn Icon => Browse => Browse => nháy kép vào Application Folder thì nút Add File sẽ có hiệu lực, bấm vào Add File => chọn đến một file ảnh có đuôi dạng .ico để làm Icon cho Shorcut, chọn được file .ico rồi thì bấm OK:



Sau đó chúng ta cắt cái Shortcut vừa tạo và paste vào trong thư mục User's Desktop

Tiếp theo, chúng ta sẽ quan tâm đến thư mục User Program Menu, nhìn cái tên thì biết ngay thư mục này liên quan đến các hoạt động ở Program Menu (cái Menu mà khi bấm Start thì hiện ra đấy). Bây giờ chúng ta sẽ Add thư mục và Short cut vào thư mục User Program Menu. Điều đó có nghĩa là khi mà chương trình của chúng ta được cài thì sau khi bấm Start, thư mục chương trình và shortcut chương trình của chúng ta sẽ nằm trên cả Program Menu nữa (như vậy trông nó mới chuyên nghiệp).

Trước tiên là Add thư mục (tên thư mục thì tùy chọn, trong trường hợp này tôi cũng vẫn đặt tên là "Test of setup"), làm giống như add thư mục Input vào thư mục Application Folder:



Sau đó, chọn thư mục Test of setup vừa tạo, tiếp đó đưa chuột ra khung giữa chọn Create Shortcut (cái này là Shortcut nằm ở trên Program Menu), một hộp thoại sổ ra, chúng ta nháy kép vào thư mục Application Folder sau đó chọn Primary ouput from Test of setup (Active) và ấn OK. Sau đó đặt tên cho Shortcut đó là được.

TÀI LIỆU THAM KHẢO

[1] Nguyễn Thị Ngọc Mai (2004), *Lập trình CSDL với Visual Basic 6.0*, NXB Lao động - Xã hội;

[2] Đinh Xuân Lâm (2004), *Những bài thực hành Cơ sở dữ liệu Visual Basic căn bản*, NXB thống kê;

[3] Nguyễn Ngọc Bình Phương – Nguyễn Quang Nam (2004), *Thủ thuật lập trình Visual basic 6.0*, NXB Giao thông vận tải.

[4] Thuận Thành (2008), *Tự học lập trình ADO.Net*, NXB Thanh niên;

[5] Nguyễn Tiến, Nguyễn Vũ Thịnh, Hồ Bích thủy (2009), *Professional ADO.NET lập trình và ứng dụng*, NXB Thống kê;

[6] Dương Quang Thiện (2005), *.NET Toàn Tập - Tập 4: Lập trình căn cứ dữ liệu dùng ADO.NET*, NXB Thành phố Hồ Chí Minh;

[7] Bill Hamilton (2007), *ADO.NET Cookbook*, O'Reilly.