

**tài liệu dự án**

**HỆ THỐNG QUẢN LÝ ĐÀO TẠO - EduSys**



|  |  |
| --- | --- |
| Giảng viên | Lại Minh Đăng |
| Học viên | Phan Thành Vinh |
| Lớp/Kỳ | :SD19303/[Su24b1.Sof2041](https://www.facebook.com/groups/1190048635701440/) |

LẬP TRÌNH CITY EDUCATION CENTER

MỤC LỤC

[1 Giới thiệu dự án 4](#_Toc43201191)

[1.1 Giới thiệu công ty LapTrinhCity 4](#_Toc43201192)

[1.2 Yêu cầu của công ty 4](#_Toc43201193)

[1.3 Lập kế hoạch dự án 4](#_Toc43201194)

[2 Phân tích yêu cầu khách hàng 4](#_Toc43201195)

[2.1 Sơ đồ Use Case 4](#_Toc43201196)

[2.2 Đặc tả yêu cầu hệ thống (SRS) 4](#_Toc43201197)

[2.2.1 Quản lý nhân viên 5](#_Toc43201198)

[2.2.2 Quản lý chuyên đề 5](#_Toc43201199)

[2.2.3 Quản lý người học 5](#_Toc43201200)

[2.2.4 Quản lý khóa học 5](#_Toc43201201)

[2.2.5 Quản lý học viên 5](#_Toc43201202)

[2.2.6 Quản lý tổng hợp – thống kê 5](#_Toc43201203)

[2.2.7 Đăng nhập 5](#_Toc43201204)

[2.2.8 Đổi mật khẩu 5](#_Toc43201205)

[2.3 Sơ đồ triển khai và yêu cầu hệ thống 5](#_Toc43201206)

[2.3.1 Sơ đồ triển khai 6](#_Toc43201207)

[2.3.2 Yêu cầu hệ thống 6](#_Toc43201208)

[3 Thiết kế ứng dụng 6](#_Toc43201209)

[3.1 Mô hình công nghệ ứng dụng 6](#_Toc43201210)

[3.2 Thực thể 6](#_Toc43201211)

[3.2.1 Sơ đồ quan hệ thực thể (ERD) 6](#_Toc43201212)

[3.2.2 Chi tiết thực thể 7](#_Toc43201213)

[3.3 Giao diện 8](#_Toc43201214)

[3.3.1 Sơ đồ tổ chức giao diện 8](#_Toc43201215)

[3.3.2 Giao diện cửa sổ chính 8](#_Toc43201216)

[3.3.3 Giao diện chức năng 9](#_Toc43201217)

[3.3.4 Các giao diện hỗ trợ khác 11](#_Toc43201218)

[4 Thực hiện dự án 13](#_Toc43201219)

[4.1 Tạo giao diện với Swing 13](#_Toc43201220)

[4.1.1 Cửa sổ chính (EduSysJFrame) 13](#_Toc43201221)

[4.1.2 Các cửa sổ quản lý 15](#_Toc43201222)

[4.1.3 Cửa sổ tổng hợp thống kê (ThongKeJDialog) 16](#_Toc43201223)

[4.1.4 Các giao diện hỗ trợ khác 16](#_Toc43201224)

[4.2 Tạo CSDL với SQL Server 17](#_Toc43201225)

[4.2.1 Sơ đồ quan hệ 17](#_Toc43201226)

[4.2.2 Chi tiết các bảng 17](#_Toc43201227)

[4.2.3 Thủ tục lưu 18](#_Toc43201228)

[4.3 Lập trình CSDL 19](#_Toc43201229)

[4.3.1 Mô hình tổ chức lập trình CSDL 19](#_Toc43201230)

[4.3.2 JdbcHelper (XJdbc) 19](#_Toc43201231)

[4.3.3 Entity Class và DAO 20](#_Toc43201232)

[4.4 Thư viện tiện ích 23](#_Toc43201233)

[4.5 Lập trình nghiệp vụ 23](#_Toc43201234)

[4.5.1 Cửa sổ chính EduSysJFrame 23](#_Toc43201235)

[4.5.2 Các cửa sổ hỗ trợ tổ chức 23](#_Toc43201236)

[4.5.3 Các cửa sổ chức năng quản lý 24](#_Toc43201237)

[4.5.4 Cửa sổ chức năng tổng hợp - thống kê 26](#_Toc43201238)

[5 Kiểm thử phần mềm và sửa lỗi 26](#_Toc43201239)

[5.1 DangNhapJDialog 26](#_Toc43201240)

[5.2 DoiMatKhauJDialog 26](#_Toc43201241)

[5.3 NhanVienJDialog 26](#_Toc43201242)

[5.4 ChuyenDeJDialog 26](#_Toc43201243)

[5.5 NguoiHocJDialog 26](#_Toc43201244)

[5.6 KhoaHocJDialog 26](#_Toc43201245)

[5.7 HocVienJDialog 27](#_Toc43201246)

[5.8 ThongKeJDialog 27](#_Toc43201247)

[6 Đóng gói và triển khai 27](#_Toc43201248)

[6.1 Sản phẩm phần mềm 27](#_Toc43201249)

[6.2 Hướng dẫn cài đặt 27](#_Toc43201250)

[7 KẾT LUẬN 27](#_Toc43201251)

[7.1 Khó khăn 27](#_Toc43201252)

[7.2 Thuận lợi 27](#_Toc43201253)

# Giới thiệu dự án

## Giới thiệu công ty LapTrinhCity

* *Giới thiệu về công ty LapTrinhCity và hiện trạng của công ty này*
* Trung tâm đào tạo tin học ngắn hạn LapTrinhCity đang đào tạo các khóa tin học ngắn hạn theo chuyên đề như Photoshop, Java, Web… Việc quản lý khóa học, học viên, bảng điểm và doanh thu đang thực hiện thông qua excel. Hiện Trung tâm đang gặp khó khăn khi số lượng người học ngày một nhiều, dữ liệu ngày càng lớn nên việc quản lý excel gặp rất nhiều khó khăn, dễ sai sót mà không bảo mật quyết khó khăn trên.

## Yêu cầu của công ty

* *Nêu yêu cầu của công ty LapTrinhCity*
* *Đánh giá tính khả thi (năng lực, khả năng của mình có thể thực hiện dự án này)*

**Yêu cầu chức năng nghiệp vụ**

o Quản lý người đăng ký học

o Quản lý các chuyên đề

o Quản lý các khóa học được tạo ra từ các chuyên đề

o Quản lý học viên của các khóa học

o Tổng hợp bảng điểm của các khóa học

o Thống kê doanh thu, số lượng người học từng chuyên đề theo năm

o Tổng hợp số lượng người học đăng ký theo tháng

**Yêu cầu về bảo mật**

o Tất cả mọi thành viên phải đăng nhập mới sử dụng được phần mềm

o Trưởng phòng đào tạo được phép thực hiện tất cả các chức năng

o Nhân viên phòng đào tạo không được phép xóa dữ liệu và cũng không được xem thông tin về doanh thu

**Yêu cầu về môi trường công nghệ**

o Ứng dụng phải được thực với công nghệ Swing và JDBC chạy trên mọi hệ điều hành với môi trường JDK tối thiểu 1.8

o Hệ quản trị CSDL SQL Server 2008 trở lên

## Lập kế hoạch dự án

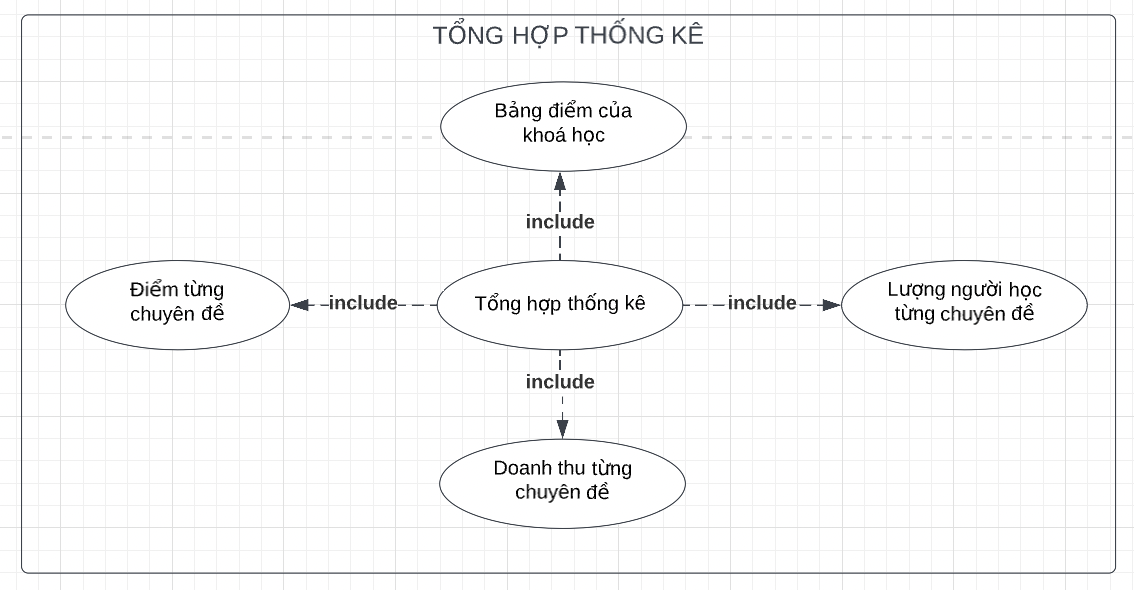
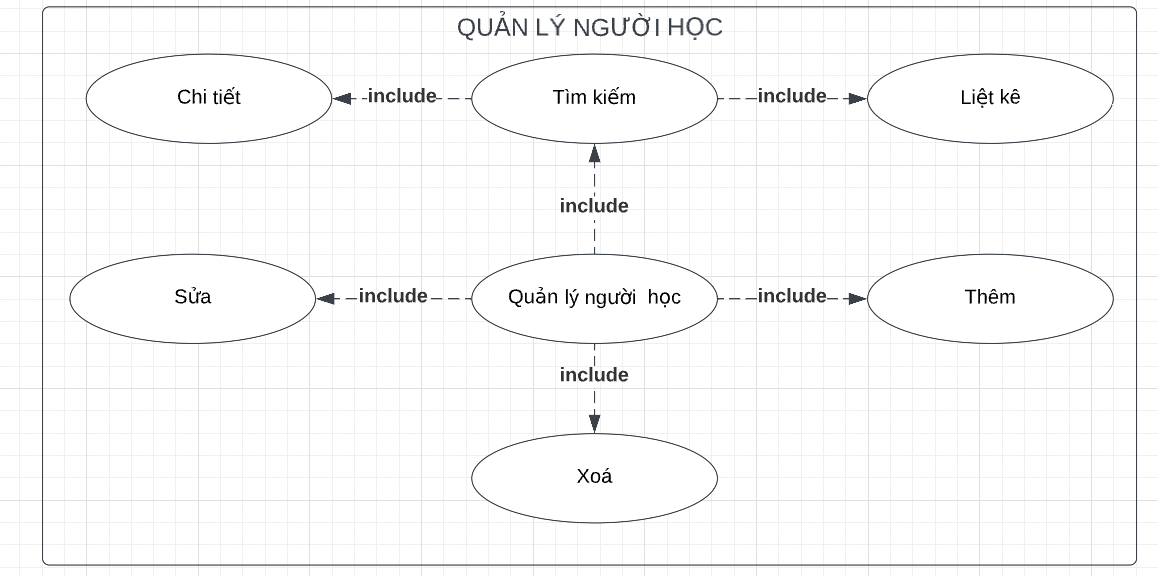
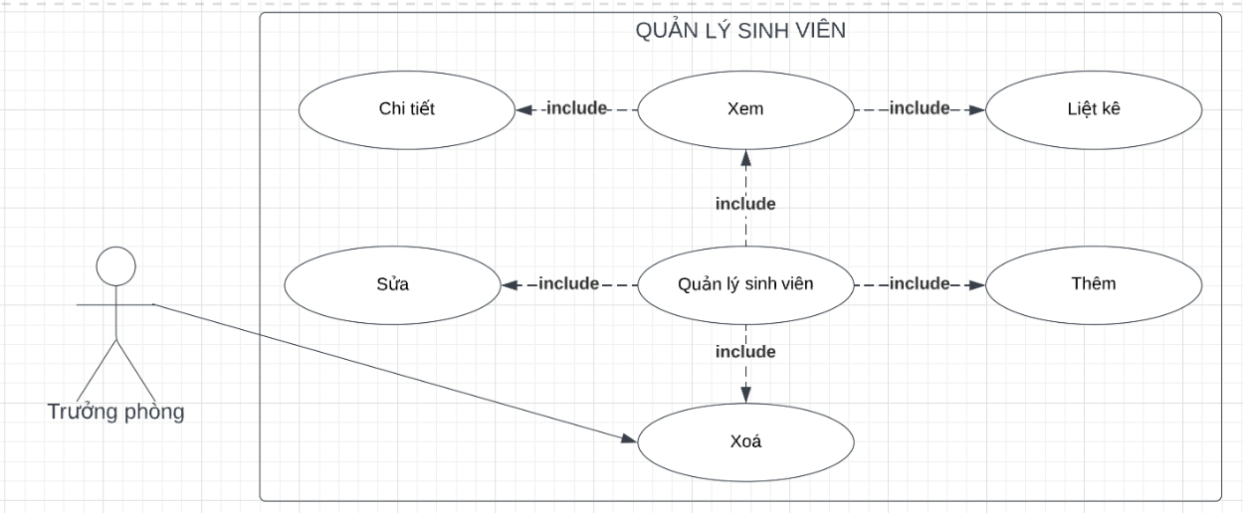
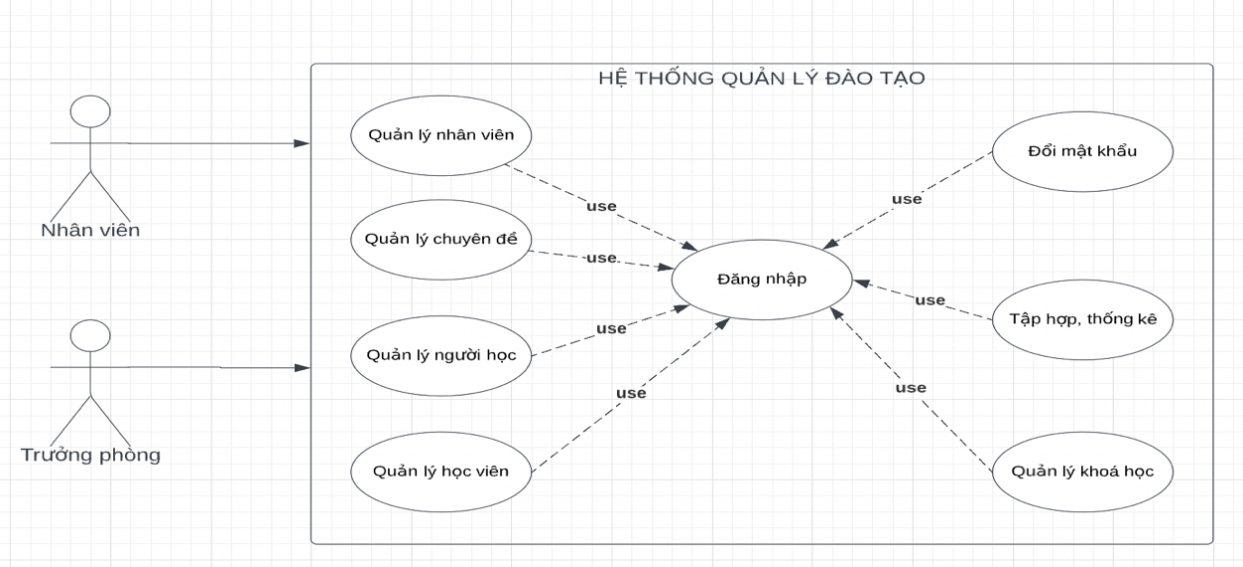
* *Lập kế hoạch kế hoạch dự án theo mẫu sau*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TT** | **Hạng mục** | **Bắt đầu** | **Kết thúc** | **Kết quả** |
| 1 | Phân tích yêu cầu khách hàng |  |  |  |
| 1.1 | Vẽ sơ đồ use cases |  |  |  |
|  | … |  |  |  |
| 2 | Thiết kế ứng dụng |  |  |  |
| 2.1 | Thiết kế mô hình công nghệ |  |  |  |
|  | … |  |  |  |
| 3 | Thực hiện dự án |  |  |  |
| 3.1 | Tạo giao diện |  |  |  |
|  | … |  |  |  |
| 4 | Kiểm thử |  |  |  |
| 4.1 | Xây dựng kịch bảng kiểm thử |  |  |  |
|  | … |  |  |  |
| 5 | Đóng gói & triển khai |  |  |  |
| 5.1 | Đóng gói sản phẩm |  |  |  |
|  | … |  |  |  |

# Phân tích yêu cầu khách hàng

* *Tiến hành phân tích yêu cầu khách hàng theo các bước sau đây*

## Sơ đồ Use Case

* *Vẽ sơ đồ use case để phác họa phạm vị của ứng dụng cũng như vai trò sử dụng. Nên chia nhiều mức để tránh rối sơ đồ*
* **

## Đặc tả yêu cầu hệ thống (SRS)

* *Hãy mô tả một cách cụ thể từng chức năng trong sơ đồ use case. Mỗi chức năng cần mô tả những vấn đề sau:*
  + *Làm công việc gì. Mục đích*
  + *Dữ liệu liên quan: cần thiết để chức năng hoạt động được*
  + *Đối tượng sử dụng*
  + ***CHI TIẾT CÁC CHỨC NĂNG***
  + *Mỗi chức năng quản lý bao gồm các chức năng con*
  + *Xem: xem tất cả và chi tiết một mục*
  + *Thêm: thêm mới vào cơ sở dữ liệu*
  + *Xóa: xóa theo mã*
  + *Sửa: cập nhật dữ liệu đang xem*
  + *Tìm kiếm: tìm kiếm theo điều kiện*
  + *Điều hướng: di chuyển đến dữ liệu của bản ghi chi tiết trước và sau*
  + *Các chức năng thao tác dữ liệu (thêm, sửa, xóa) cần được kiểm lỗi hợp lý với dữ*
  + *liệu*
  + *Chức năng tổng hợp thống kê số liệu*
  + *Bảng điểm theo khóa học phải có cấu trúc là:*
  + *Mã người học*
  + *Họ và tên*
  + *Điểm*
  + *Tổng hợp điểm từng chuyên đề*
  + *Chuyên đề*
  + *Số học viên*
  + *Điểm cao nhất*
  + *Điểm thấp nhất*
  + *Điểm trung bình*
  + *PROJECT DOCUMENT*
  + *SAMPLE PROJECT - ỨNG DỤNG PHẦN MỀM PAGE 6*
  + *Doanh thu từng chuyên đề theo năm phải có cấu trúc:*
  + *Chuyên đề*
  + *Số khóa*
  + *Số học viên*
  + *Doanh thu*
  + *Học phí thấp nhất*
  + *Học phí cao nhất*
  + *Học phí trung bình*
  + *Thống kê số người học từng năm phải có cấu trúc:*
  + *Năm*
  + *Số học viên*
  + *Ngày đăng ký sớm nhất*
  + *Ngày đăng ký muộn nhất*
  + *o Chức năng đăng nhập và đăng xuất*
  + *Đăng nhập: để sử dụng phần mềm, bắt buộc phải đăng nhập*
  + *Đăng xuất: đăng nhập lại với người dùng khác hoặc dừng lại đi đâu đó*
  + ***CHI TIẾT VỀ YÊU CẦU BẢO MẬT***
  + *o Tất cả phải đăng nhập mới được sử dụng các chức năng trong phần mềm*
  + *o Trưởng phòng: được phép sử dụng tất cả*
  + *Nhân viên:*
  + *Không được phép thực hiện chức năng xóa trong các chức năng quản lý*
  + *Không được phép xem chức năng thống kê doanh thu*
  + ***THÔNG TIN CÁC THỰC THỂ***
  + *Người học*
  + *Mã người đăng ký học*
  + *Họ và tên*
  + *Ngày sinh*
  + *Giới tính*
  + *Số điện thoại*
  + *Email*
  + *Ghi chú*
  + *Chuyên đề*
  + *Mã chuyên đề*
  + *Tên chuyên đề*
  + *Học phí*
  + *Thời lượng (tính theo giờ)*
  + *Hình logo*
  + *Mô tả chuyên đề*
  + *Khóa học*
  + *Mã khóa học*
  + *Mã chuyên đề*
  + *PROJECT DOCUMENT*
  + *SAMPLE PROJECT - ỨNG DỤNG PHẦN MỀM PAGE 7*
  + *Học phí*
  + *Thời lượng*
  + *Ngày khai giảng*
  + *Ghi chú*
  + *Học viên*
  + *Mã học viên (số báo danh)*
  + *Mã khóa học*
  + *Mã người học*
  + *Điểm trung bình*
  + *Nhân viên*
  + *Mã nhân viên (dùng để đăng nhập)*
  + *Mật khẩu*
  + *Họ và tên*
  + *Vai trò (trưởng phòng hay nhân viên)*
  + *Chú ý: Học phí và thời lượng của chuyên đề có thể bị thay đổi mà không làm ảnh hưởng*

*đến học phí và thời lượng của những người đã học*

### Quản lý nhân viên

* **Mô tả chức năng:**

Chức năng quản lý nhân viên được sử dụng để quản lý thông tin nhân viên. Yêu cầu của chức năng này là liệt kê danh sách nhân viên, xem thông tin chi tiết của mỗi nhân viên, thêm nhân viên mới, cập nhật thông tin hoặc xóa nhân viên đã tồn tài.

* **Dữ liệu liên quan:**

Thông tin của mỗi nhân viên gồm: mã nhân viên, họ và tên, mật khẩu đăng nhập, vai trò của nhân viên (gồm trưởng phòng và nhân viên bình thường).

* **Đối tượng sử dụng:**

Sau khi đăng nhập thì nhân viên nào cũng có thể sử dụng chức năng này, ngoại trừ xóa thì chỉ có trưởng phòng mới sử dụng được.

### Quản lý chuyên đề

* …**Mô tả chức năng:**

Chức năng quản lý nhân viên được sử dụng để quản lý thông tin nhân viên. Yêu cầu của chức năng này là liệt kê danh sách nhân viên, xem thông tin chi tiết của mỗi nhân viên, thêm nhân viên mới, cập nhật thông tin hoặc xóa nhân viên đã tồn tài.

* **Dữ liệu liên quan:**

Thông tin của mỗi nhân viên gồm: mã nhân viên, họ và tên, mật khẩu đăng nhập, vai trò của nhân viên (gồm trưởng phòng và nhân viên bình thường).

* **Đối tượng sử dụng:**

Sau khi đăng nhập thì nhân viên nào cũng có thể sử dụng chức năng này, ngoại trừ xóa thì chỉ có trưởng phòng mới sử dụng được

### Quản lý người học

* …**Mô tả chức năng:**

Chức năng quản lý nhân viên được sử dụng để quản lý thông tin nhân viên. Yêu cầu của chức năng này là liệt kê danh sách nhân viên, xem thông tin chi tiết của mỗi nhân viên, thêm nhân viên mới, cập nhật thông tin hoặc xóa nhân viên đã tồn tài.

* **Dữ liệu liên quan:**

Thông tin của mỗi nhân viên gồm: mã nhân viên, họ và tên, mật khẩu đăng nhập, vai trò của nhân viên (gồm trưởng phòng và nhân viên bình thường).

* **Đối tượng sử dụng:**

Sau khi đăng nhập thì nhân viên nào cũng có thể sử dụng chức năng này, ngoại trừ xóa thì chỉ có trưởng phòng mới sử dụng được

### Quản lý khóa học

* …**Mô tả chức năng:**

Chức năng quản lý nhân viên được sử dụng để quản lý thông tin nhân viên. Yêu cầu của chức năng này là liệt kê danh sách nhân viên, xem thông tin chi tiết của mỗi nhân viên, thêm nhân viên mới, cập nhật thông tin hoặc xóa nhân viên đã tồn tài.

* **Dữ liệu liên quan:**

Thông tin của mỗi nhân viên gồm: mã nhân viên, họ và tên, mật khẩu đăng nhập, vai trò của nhân viên (gồm trưởng phòng và nhân viên bình thường).

* **Đối tượng sử dụng:**

Sau khi đăng nhập thì nhân viên nào cũng có thể sử dụng chức năng này, ngoại trừ xóa thì chỉ có trưởng phòng mới sử dụng được

### Quản lý học viên

* …**Mô tả chức năng:**

Chức năng quản lý nhân viên được sử dụng để quản lý thông tin nhân viên. Yêu cầu của chức năng này là liệt kê danh sách nhân viên, xem thông tin chi tiết của mỗi nhân viên, thêm nhân viên mới, cập nhật thông tin hoặc xóa nhân viên đã tồn tài.

* **Dữ liệu liên quan:**

Thông tin của mỗi nhân viên gồm: mã nhân viên, họ và tên, mật khẩu đăng nhập, vai trò của nhân viên (gồm trưởng phòng và nhân viên bình thường).

* **Đối tượng sử dụng:**

Sau khi đăng nhập thì nhân viên nào cũng có thể sử dụng chức năng này, ngoại trừ xóa thì chỉ có trưởng phòng mới sử dụng được

### Quản lý tổng hợp – thống kê

* …**Mô tả chức năng:**

Chức năng quản lý nhân viên được sử dụng để quản lý thông tin nhân viên. Yêu cầu của chức năng này là liệt kê danh sách nhân viên, xem thông tin chi tiết của mỗi nhân viên, thêm nhân viên mới, cập nhật thông tin hoặc xóa nhân viên đã tồn tài.

* **Dữ liệu liên quan:**

Thông tin của mỗi nhân viên gồm: mã nhân viên, họ và tên, mật khẩu đăng nhập, vai trò của nhân viên (gồm trưởng phòng và nhân viên bình thường).

* **Đối tượng sử dụng:**

Sau khi đăng nhập thì nhân viên nào cũng có thể sử dụng chức năng này, ngoại trừ xóa thì chỉ có trưởng phòng mới sử dụng được

### Đăng nhập

* …**Mô tả chức năng:**

Chức năng quản lý nhân viên được sử dụng để quản lý thông tin nhân viên. Yêu cầu của chức năng này là liệt kê danh sách nhân viên, xem thông tin chi tiết của mỗi nhân viên, thêm nhân viên mới, cập nhật thông tin hoặc xóa nhân viên đã tồn tài.

* **Dữ liệu liên quan:**

Thông tin của mỗi nhân viên gồm: mã nhân viên, họ và tên, mật khẩu đăng nhập, vai trò của nhân viên (gồm trưởng phòng và nhân viên bình thường).

* **Đối tượng sử dụng:**

Sau khi đăng nhập thì nhân viên nào cũng có thể sử dụng chức năng này, ngoại trừ xóa thì chỉ có trưởng phòng mới sử dụng được

### Đổi mật khẩu

* …**Mô tả chức năng:**

Chức năng quản lý nhân viên được sử dụng để quản lý thông tin nhân viên. Yêu cầu của chức năng này là liệt kê danh sách nhân viên, xem thông tin chi tiết của mỗi nhân viên, thêm nhân viên mới, cập nhật thông tin hoặc xóa nhân viên đã tồn tài.

* **Dữ liệu liên quan:**

Thông tin của mỗi nhân viên gồm: mã nhân viên, họ và tên, mật khẩu đăng nhập, vai trò của nhân viên (gồm trưởng phòng và nhân viên bình thường).

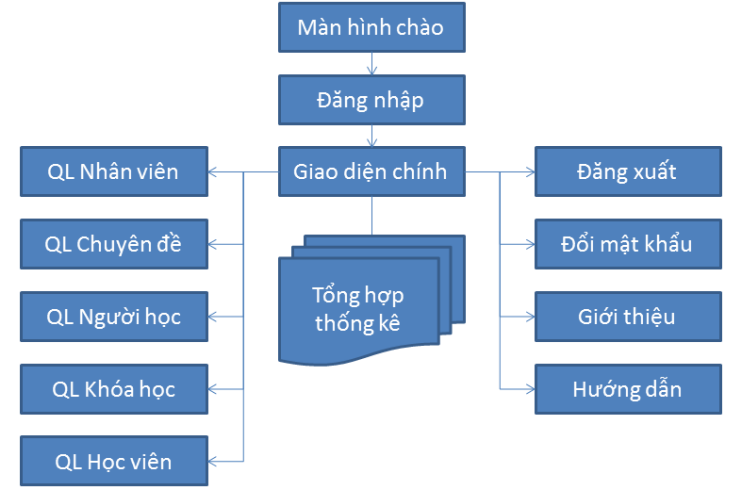
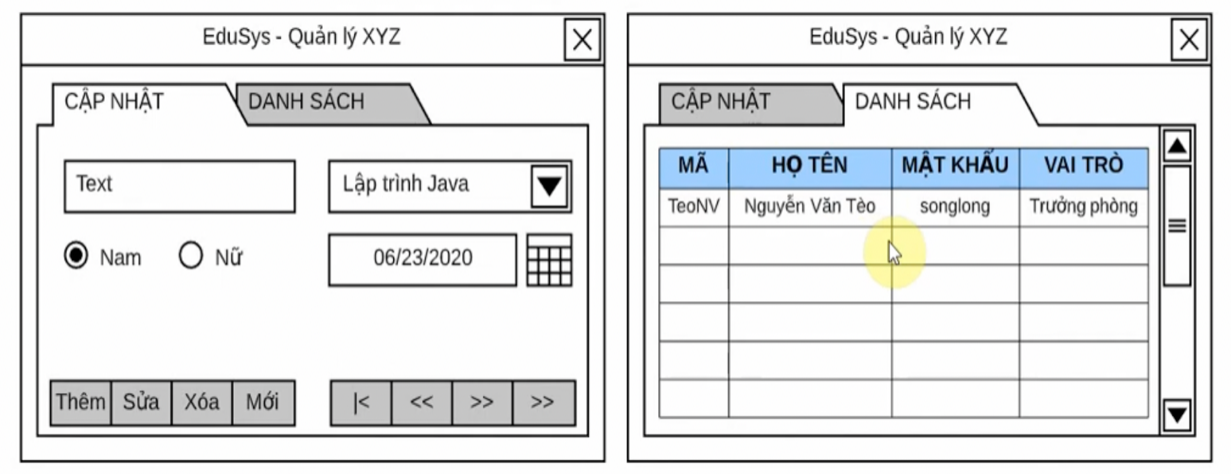
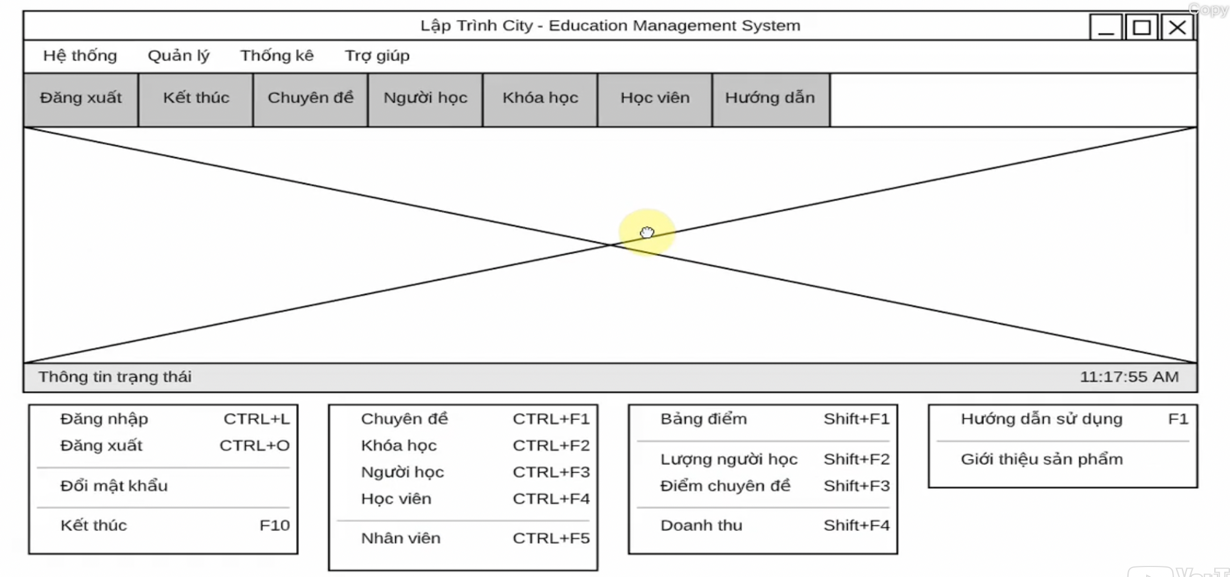
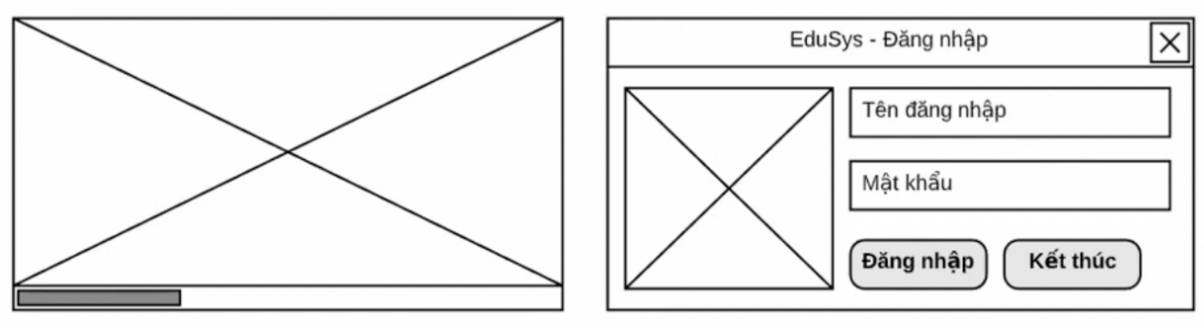
* **Đối tượng sử dụng:**

Sau khi đăng nhập thì nhân viên nào cũng có thể sử dụng chức năng này, ngoại trừ xóa thì chỉ có trưởng phòng mới sử dụng được

## Sơ đồ triển khai và yêu cầu hệ thống

* *Đưa ra mô hình triển khai của ứng dụng trong thực tế để từ đó chỉ ra những điều kiện cần thiết cho việc triển khai ứng dụng*
* **Ứng dụng phần mềm được xây dựng để phục vụ cho nhiều người dùng nhưng cơ sở dữ liệu thì lưu trữ tập trung.
* Cần một máy cài SQL Server 2008+. Máy này cần hệ điều hành window xp trở lên
* Các máy nhân viên phòng đào tạo cài phần mềm edusys. Các máy nhân viên cần JDK 1.8+ với hệ điều hành bất ký

### Sơ đồ triển khai

* *Đưa ra mô hình triển khai của ứng dụng trong thực tế*
* 
* 

### Yêu cầu hệ thống

* *Liệt kê các phần cứng, phần mềm cần thiết để hệ thống thực hiện được*

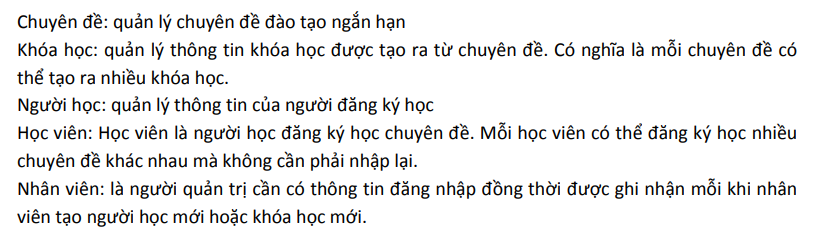
# Thiết kế ứng dụng

* *Dựa vào tài liệu đã phân tích ở mục 2 để phác thảo:*
  + *Mô hình công nghệ*
  + *Giao diện*
  + *Thực thể*

## Mô hình công nghệ ứng dụng

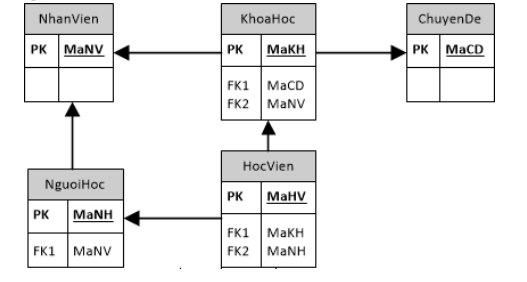
* *Phác thảo mô hình công nghệ ứng dụng của hệ thống phần mềm. Giải thích rõ các thành phần trong mô hình*

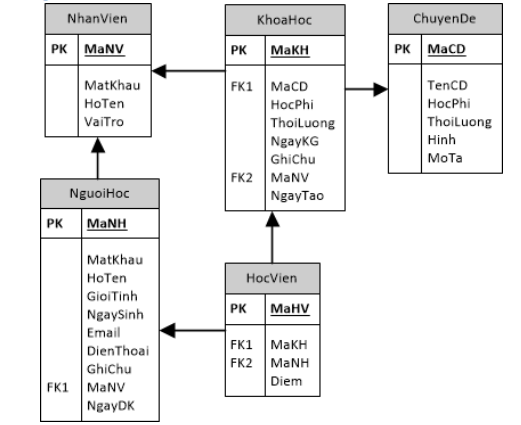
## Thực thể

* *Căn cứ vào SRS để thiết kế các thực thể. Cần phải hoàn thiện 2 phần theo yêu cầu sau đây.*
* **

### Sơ đồ quan hệ thực thể (ERD)

* *Vẽ sơ đồ thể hiện mối quan hệ giữa các thực thể. Nên chia thành 2 mức (tổng quát và chi tiết) để dễ đọc, dễ hiểu.*

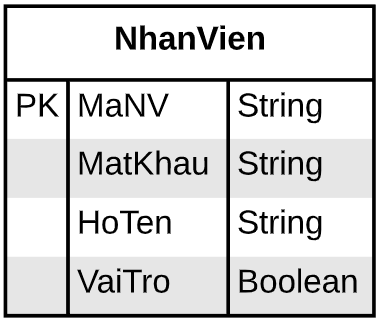


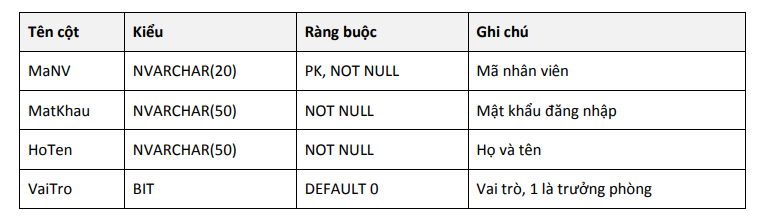


### Chi tiết thực thể

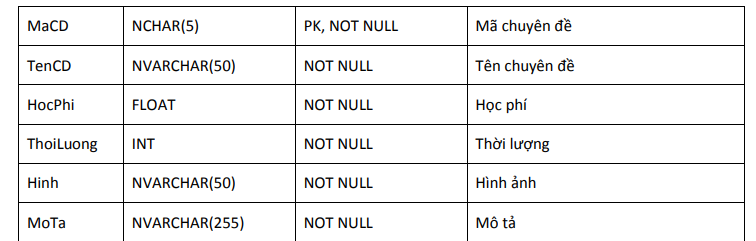
* *Thiết kế chi tiết thực thể có nghĩa là thiết kế các thuộc tính cần thiết cho mỗi thực thể (căn cứ vào phần dữ liệu liên quan của SRS).*

#### Thực thể Nhân viên



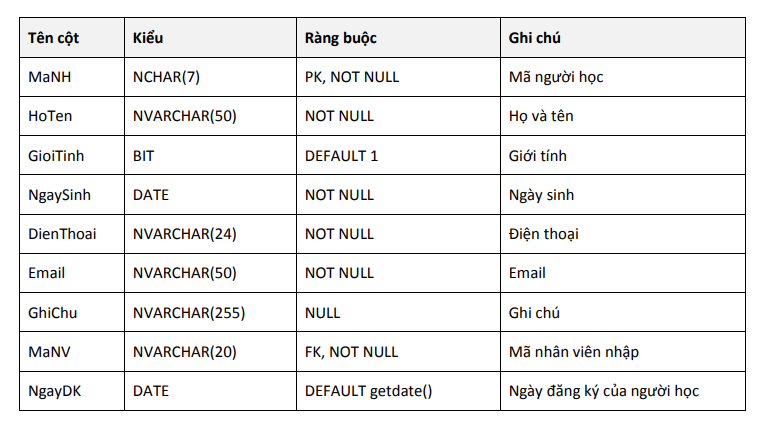


#### Thực thể Chuyên đề



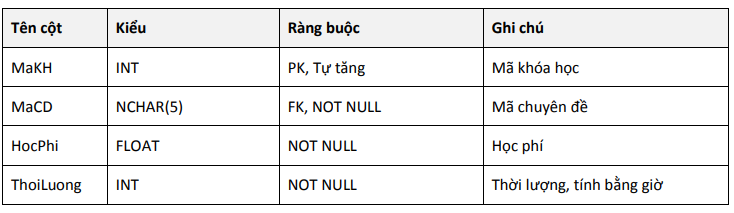
…

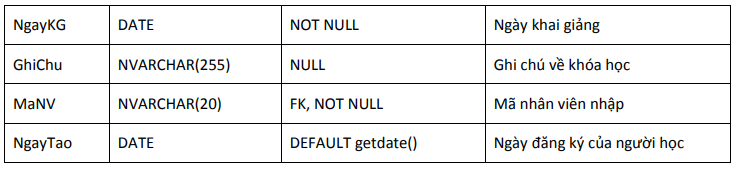
#### Thực thể Người học



…

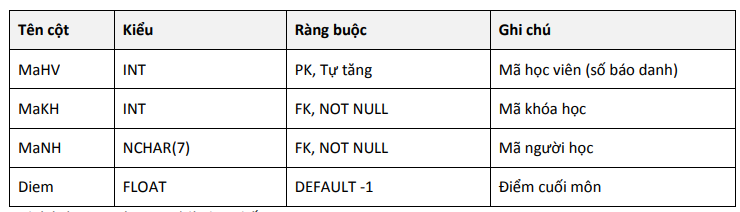
#### Thực thể Khóa học





…

#### Thực thể Học viên



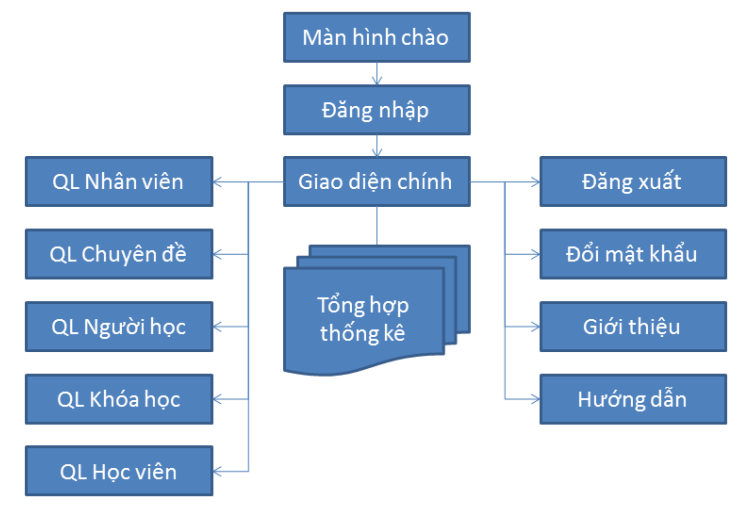
…

## Giao diện

* *Chú ý: phần này chỉ phác thảo giao diện chứ không phải xây dựng nên bạn có thể sử dụng bất ký công cụ gì thậm chí vẽ trên giấy rồi chụp hình đưa vào đây.*

### Sơ đồ tổ chức giao diện

* *Phác thảo sơ đồ liên kết giữa các giao diện chức năng trong ứng dụng*
* Giao diện chính là một cửa sổ chứa menu chính và toolbar. Thông qua đó để đi đến các giao diện thành viên để thực hiện các chức năng trong hệ thống. Theo yêu cầu thì mọi nhân viên phải đăng nhập trước khi sử dụng ứng dụng nên form đăng nhập xuất hiện trước để yêu cầu đăng nhập. Ngoài ra mỗi ứng dụng trong thời gian khởi động cần có một màn hình chào cùng với thanh tiến trình để người có cảm giác ứng dụng đang khởi động.
* *Mô tả sơ đồ*

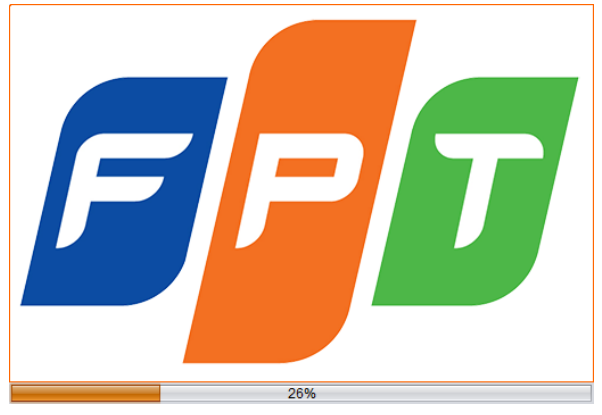


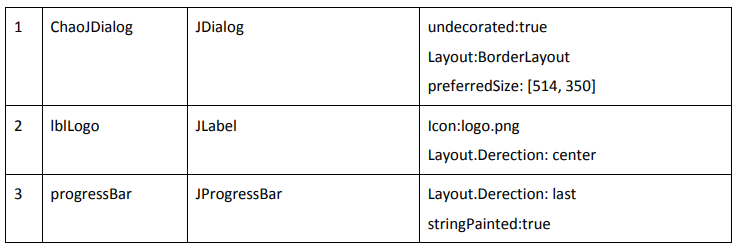
### Giao diện cửa sổ chính

* *Phác thảo giao diện cho cửa sổ chính và mô tả các hoạt động trên giao diện*

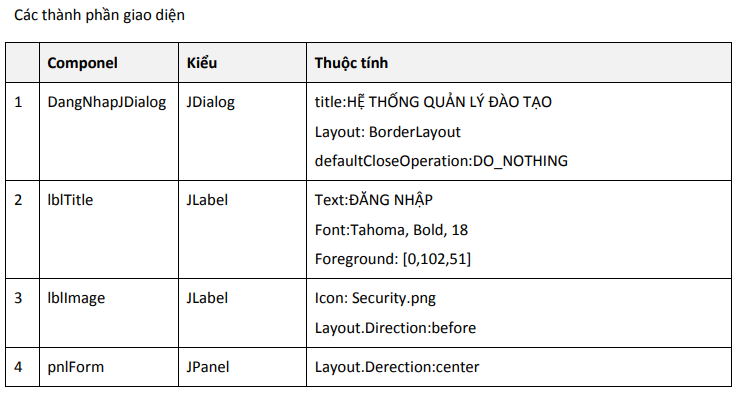
***Giao diện:***

**Màn hình chào**

****

****

***Đăng nhập***

******

******

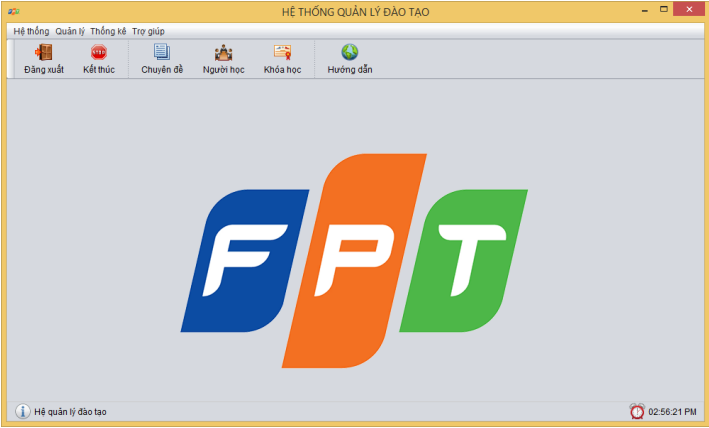
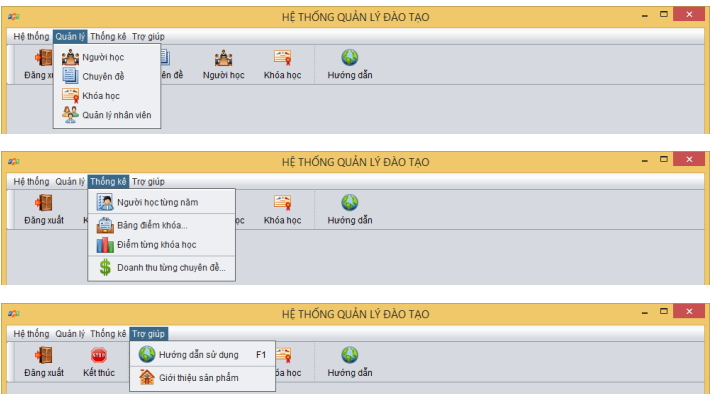
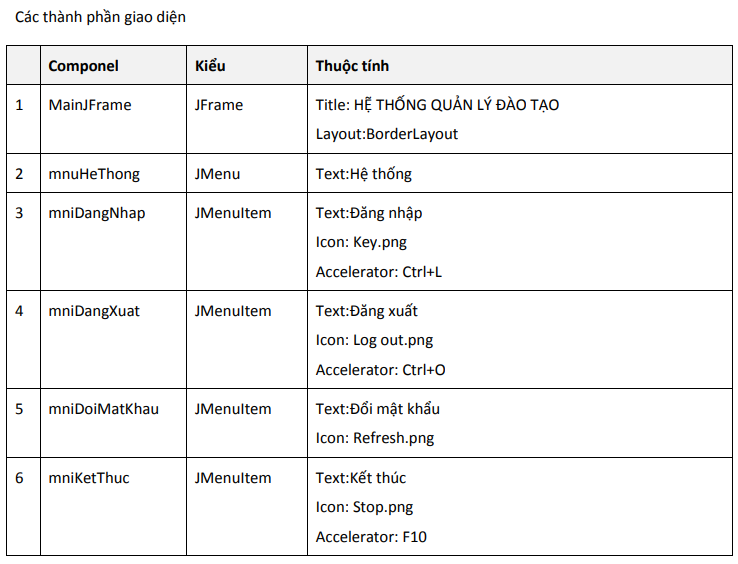
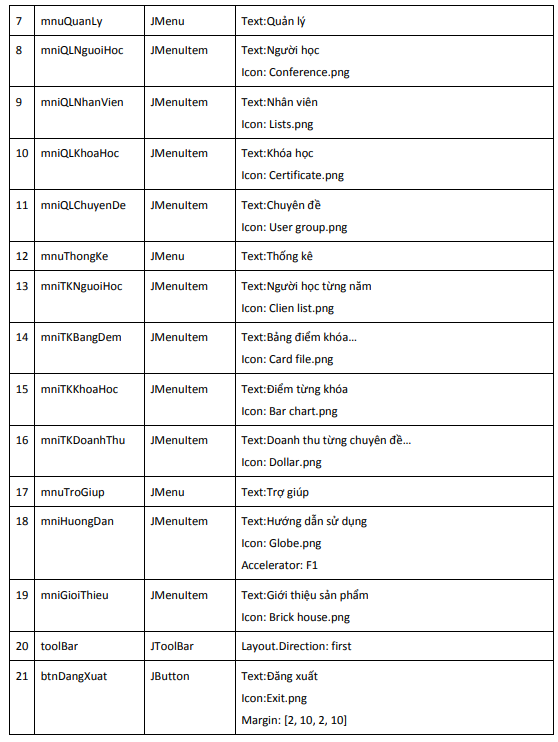
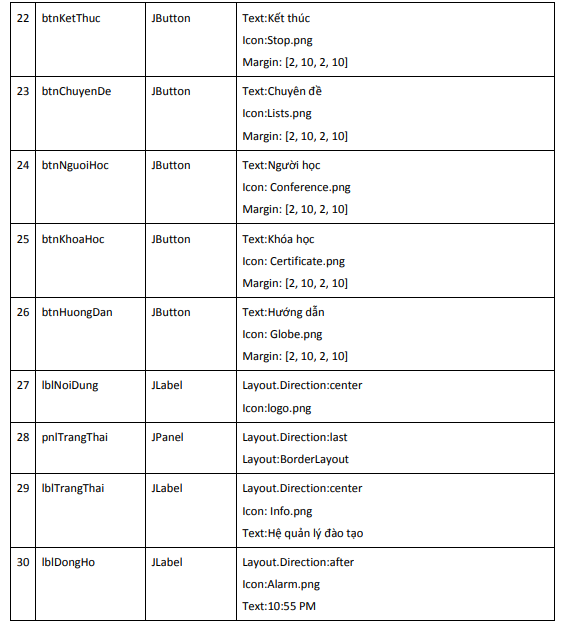
**Mô tả hoạt động**

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Điều khiển** | **Sự kiện** | **Mô tả hoạt động** |
| 1 | Cửa sổ | Initialize | Đưa cửa sổ ra giữa màn hình |
| 2 | [Đăng xuất] | Click | Đăng xuất và hiển thị DangNhapJDialog |
| 3 | [Đổi mật khẩu] | Click | Hiển thị DoiMatKhauJDialog |
| 4 | [Chuyên đề] | Click | Hiển thị ChuyenDeJDialog |
| 5 | [Bảng điểm] | Click | Hiển thị ThongKeJDialog.Tabs[0] |
|  | … |  |  |

### Giao diện chức năng

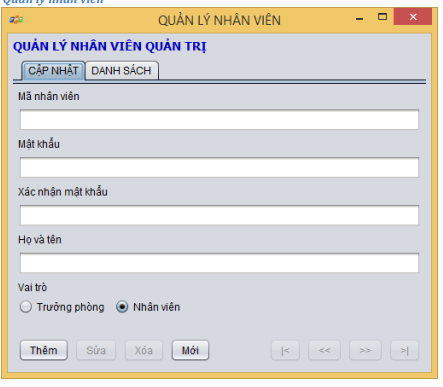
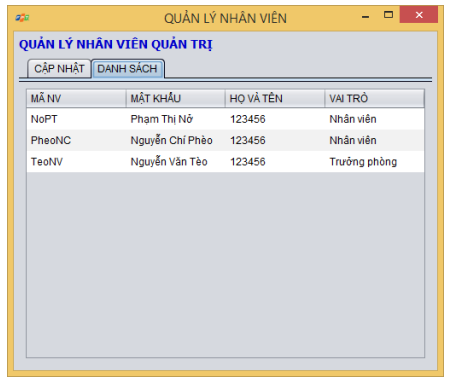
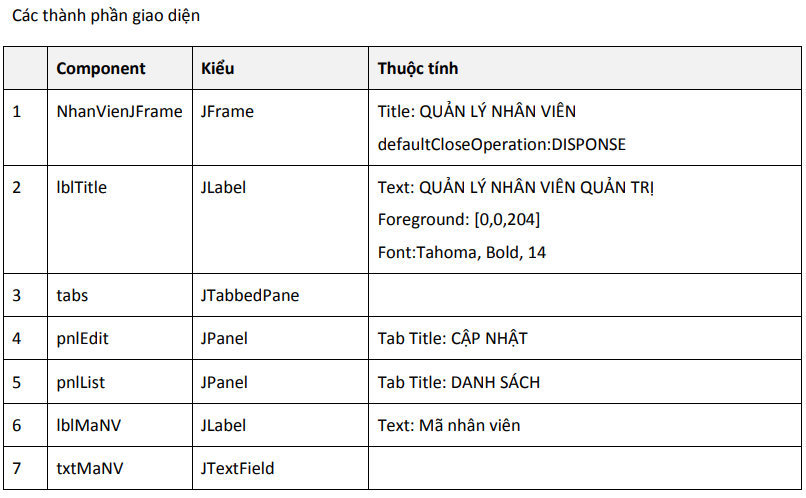
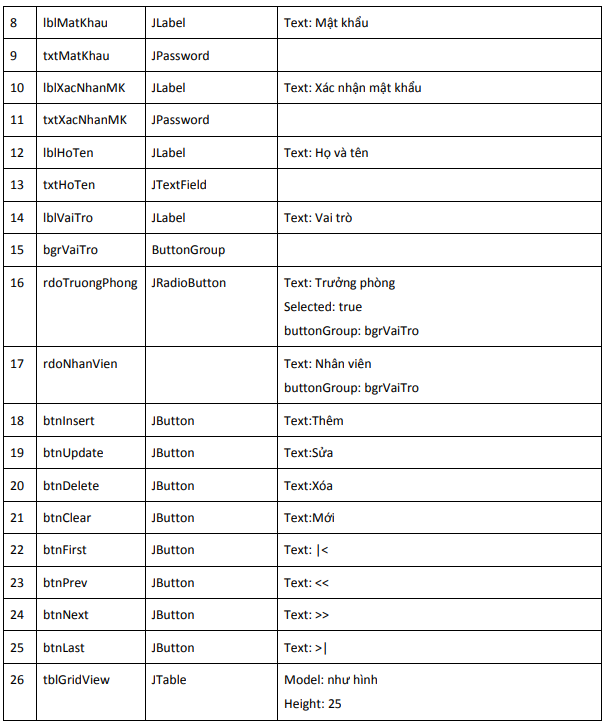
* *Phác thảo giao diện cho tất cả các cửa sổ chức năng và các cửa sổ hỗ trợ tổ chức*

*Giao Dien Chinh*

**   

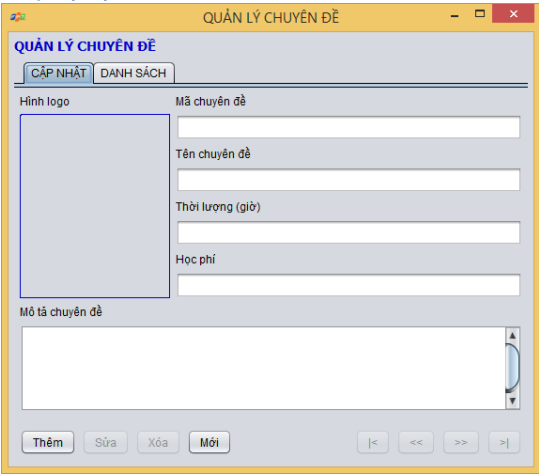
#### Cửa sổ quản lý nhân viên

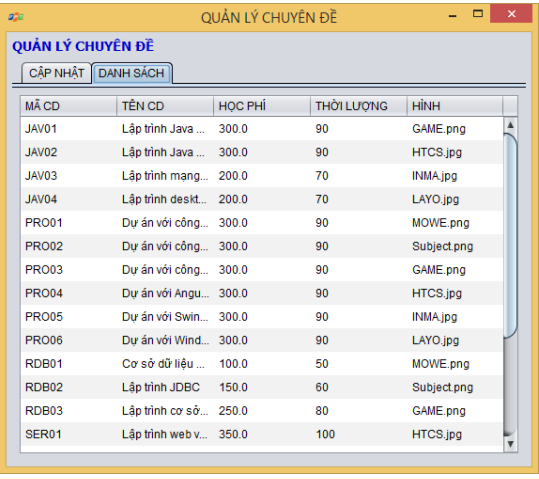
**Giao diện**

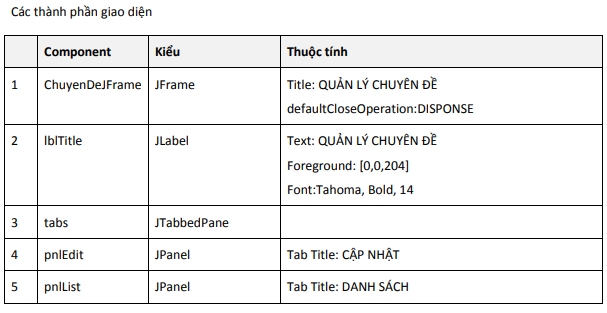
**** ****  

#### Cửa sổ quản lý chuyên đề

**Giao diện:**

****

****

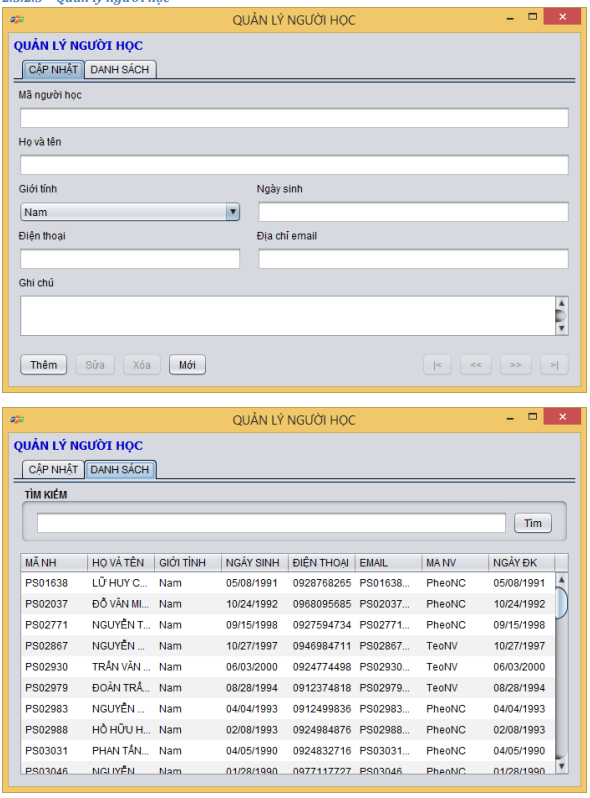
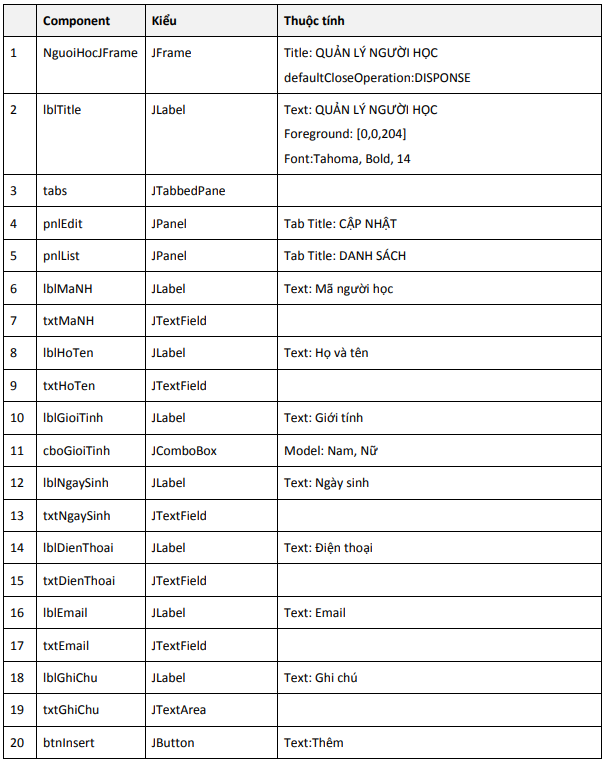
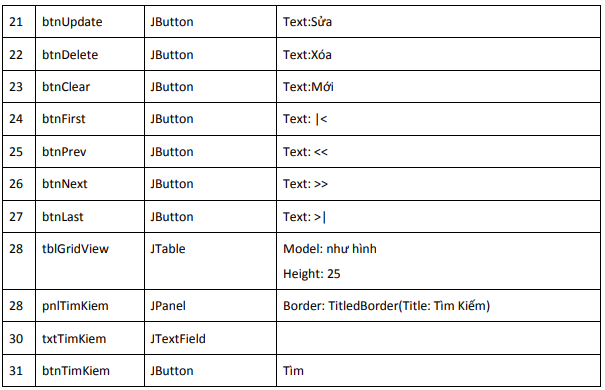
**** ****

**Mô tả hoạt động:**

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Điều khiển** | **Sự kiện** | **Mô tả hoạt động** |
| 1 | …. |  |  |
| 2 |  |  |  |

#### Cửa sổ quản lý người học

**Giao diện:**

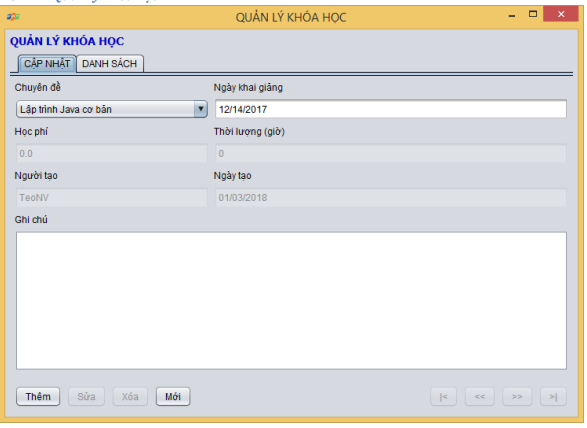
**** **** 

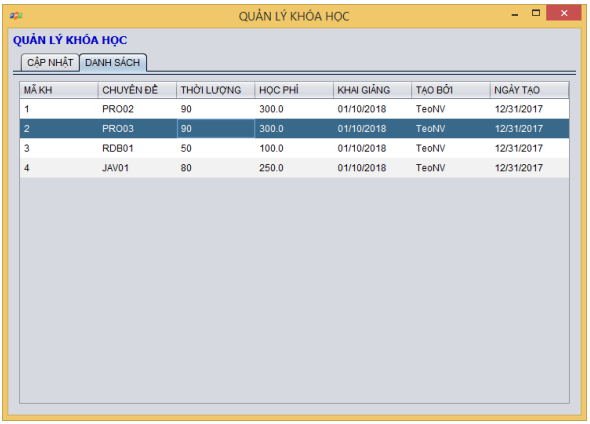
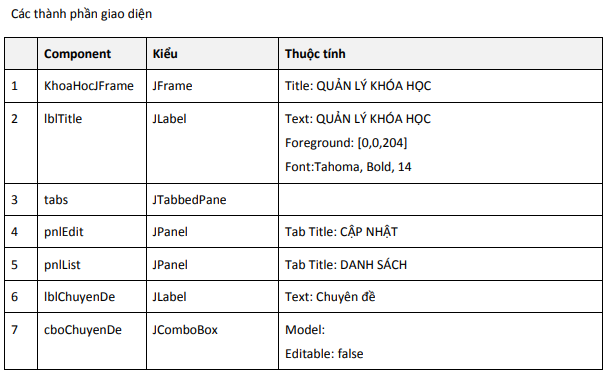
**Mô tả hoạt động:**

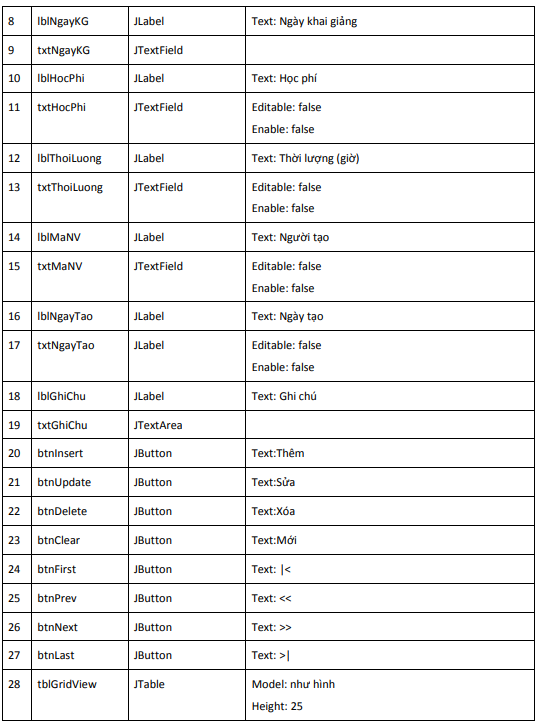
|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Điều khiển** | **Sự kiện** | **Mô tả hoạt động** |
| 1 | …. |  |  |
| 2 |  |  |  |

#### Cửa sổ quản lý khóa học

**Giao diện:**

****

**** ****

****

****

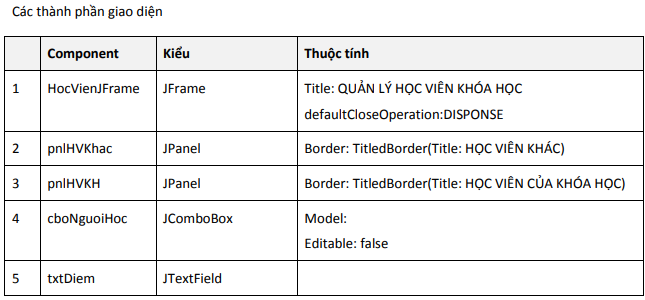
**Mô tả hoạt động:**

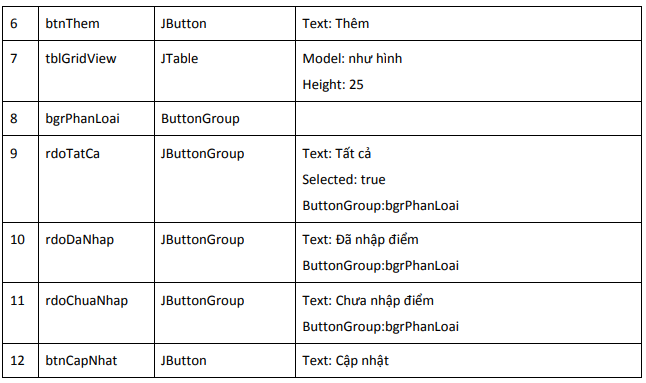
|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Điều khiển** | **Sự kiện** | **Mô tả hoạt động** |
| 1 | …. |  |  |
| 2 |  |  |  |

#### Cửa sổ quản lý học viên

**Giao diện:**

****

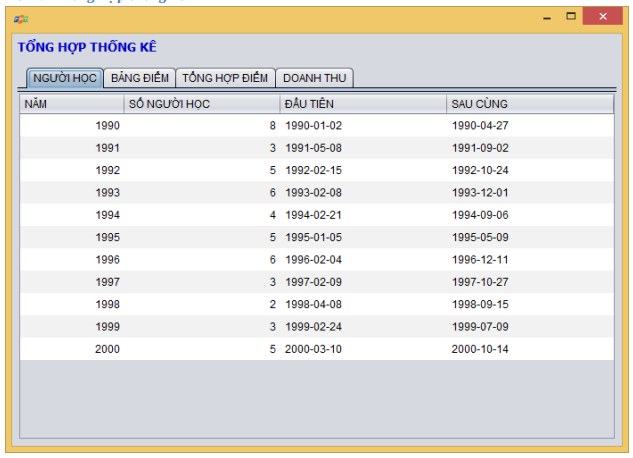
****

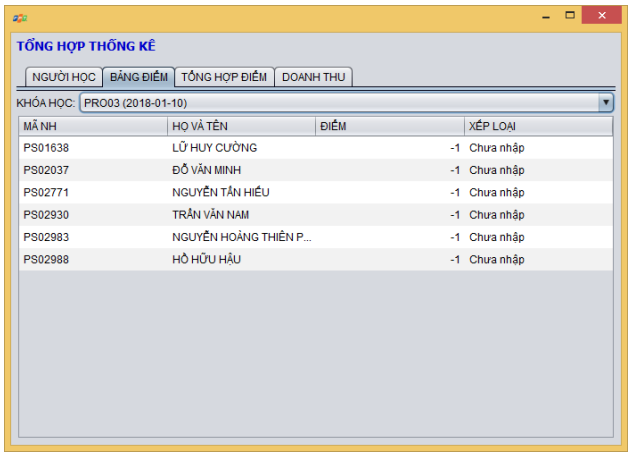
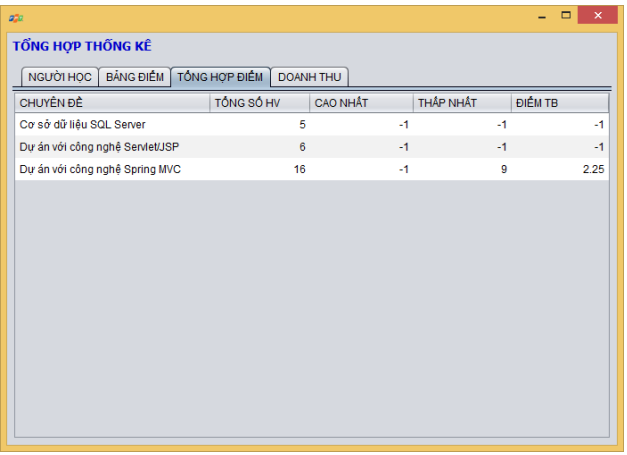
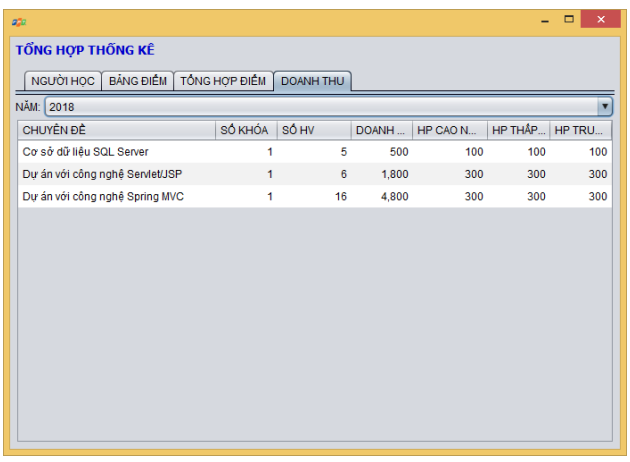
****

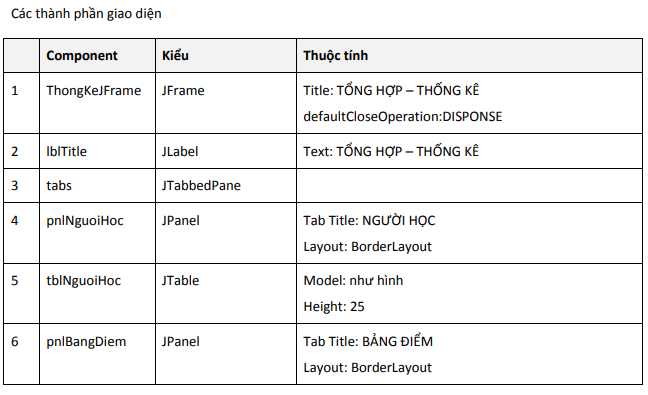
**Mô tả hoạt động:**

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Điều khiển** | **Sự kiện** | **Mô tả hoạt động** |
| 1 | …. |  |  |
| 2 |  |  |  |

#### Cửa sổ thống kê

**Giao diện**

**** **** 

****

****

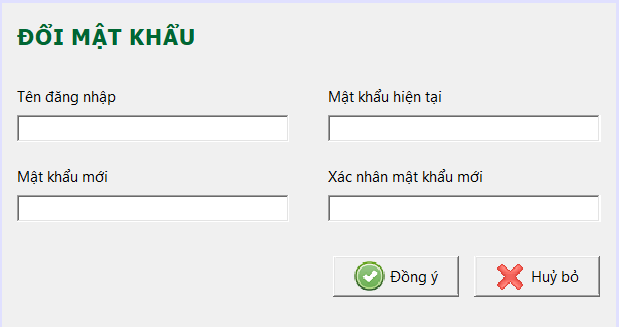
**Mô tả hoạt động:**

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Điều khiển** | **Sự kiện** | **Mô tả hoạt động** |
| 1 | …. |  |  |
| 2 |  |  |  |

### Các giao diện hỗ trợ khác

#### Cửa sổ đổi mật khẩu

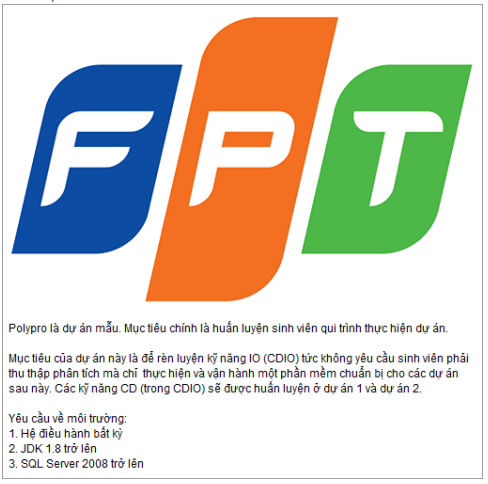
**Giao diện:**

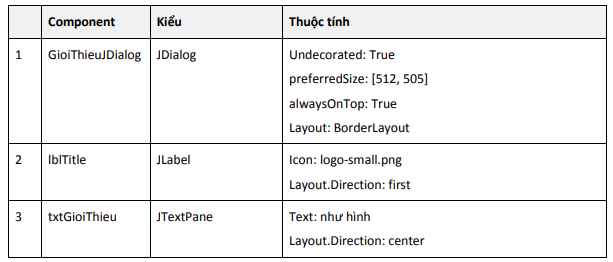
**Mô tả hoạt động:**

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Điều khiển** | **Sự kiện** | **Mô tả hoạt động** |
| 1 | …. |  |  |
| 2 |  |  |  |

#### Cửa sổ giới thiệu

**Giao diện:**

****

****

**Mô tả hoạt động:**

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Điều khiển** | **Sự kiện** | **Mô tả hoạt động** |
| 1 | …. |  |  |
| 2 |  |  |  |

#### Cửa sổ hướng dẫn

**Giao diện:**

****

Thiết kế giao diện:

Thiết kế các trang web hướng dẫn sử dụng phần mềm. Trong phần viết mã sẽ hướng dẫn cách mở các trang web này khi người dùng nhấp vào mục hướng dẫn trong menu trợ giúp.

**Mô tả hoạt động:**

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Điều khiển** | **Sự kiện** | **Mô tả hoạt động** |
| 1 | …. |  |  |
| 2 |  |  |  |

## Tạo CSDL với SQL Server

### Sơ đồ quan hệ

* *Hình: Sơ đồ quan hệ*
* *Ràng buộc: Mô tả ràng buộc của các quan hệ*
* **

### Chi tiết các bảng

* *Mỗi bảng cần viết theo hướng dẫn sau đây:*

#### Bảng NhanVien

**Cấu trúc bảng**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| MaNV | NVARCHAR(20) | PK, NOT NULL | Mã nhân viên |
| MatKhau | NVARCHAR(50) | NOT NULL | Mật khẩu |
| HoTen | NVARCHAR(50) | NOT NULL | Họ và tên |
| VaiTro | BIT | DEFAULT 0 | 1 là trưởng phòng, 0 là nhân viên |

**Mã lệnh tạo bảng**

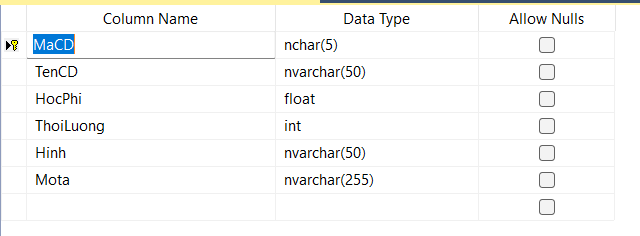
|  |
| --- |
| CREATE TABLE NhanVien  (  MaNV NVARCHAR(50) NOT NULL,  MatKhau NVARCHAR(50) NOT NULL,  HoTen NVARCHAR(50) NOT NULL,  VaiTro BIT NOT NULL,  PRIMARY KEY(MaNV)  ) |

**Câu lệnh SQL cơ bản**

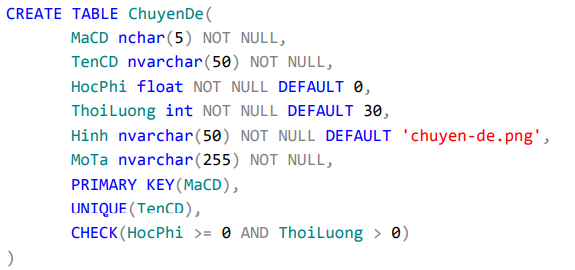
|  |
| --- |
| INSERT INTO NhanVien (MaNV, MatKhau, HoTen, VaiTro) VALUES (?, ?, ?, ?)  UPDATE NhanVien SET MatKhau = ?, HoTen = ?, VaiTro = ? WHERE MaNV = ?  DELETE FROM NhanVien WHERE MaNV = ?  SELECT \* FROM NhanVien WHERE MaNV = ? |

#### Bảng ChuyenDe

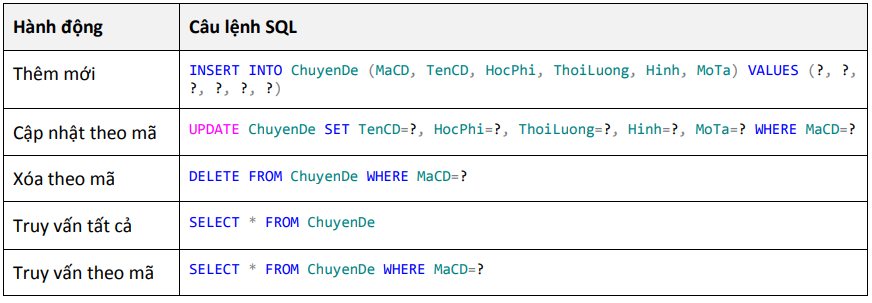
**Cấu trúc bảng**

****

**Mã lệnh tạo bảng**

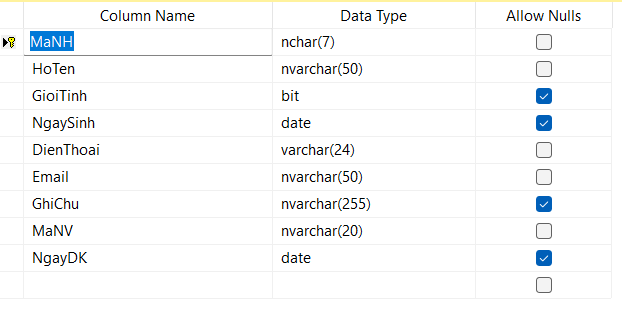
****

**Câu lệnh SQL cơ bản**

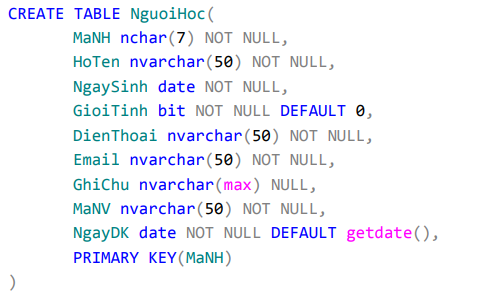
****

#### Bảng NguoiHoc

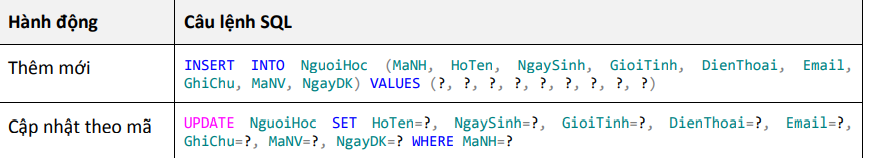
**Cấu trúc bảng**

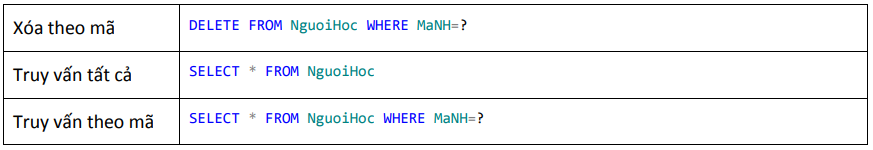
****

**Mã lệnh tạo bảng**

****

**Câu lệnh SQL cơ bản**

****

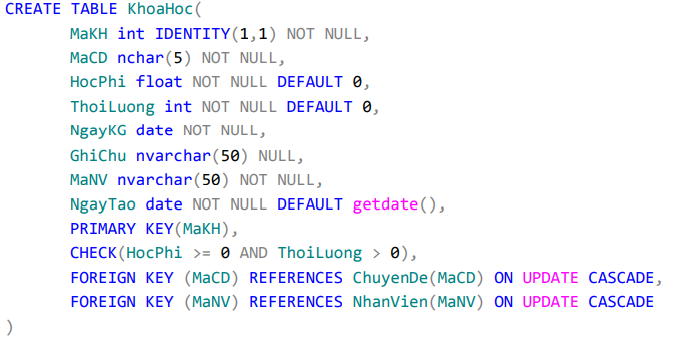
****

#### Bảng KhoaHoc

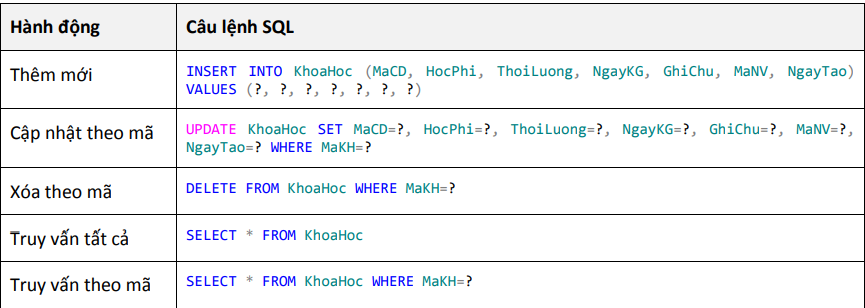
**Cấu trúc bảng**

****

**Mã lệnh tạo bảng**

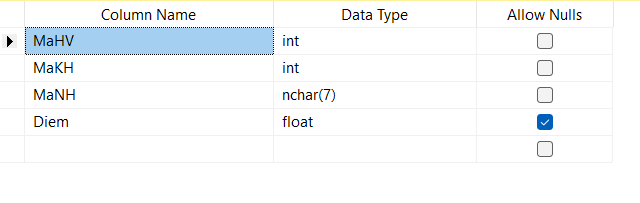
****

**Câu lệnh SQL cơ bản**

****

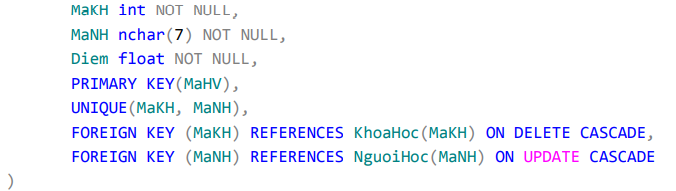
#### Bảng HocVien

**Cấu trúc bảng**

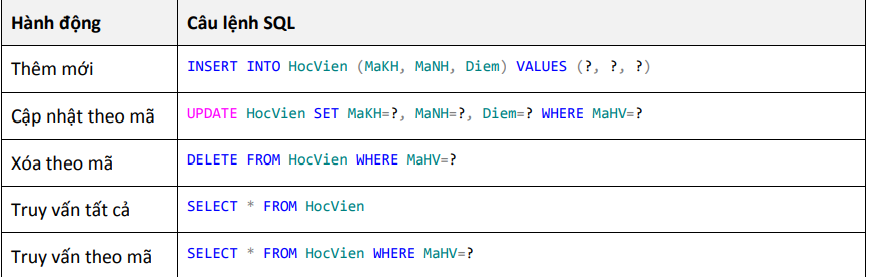
****

**Mã lệnh tạo bảng**

****

****

**Câu lệnh SQL cơ bản**

****

### Thủ tục lưu

* *Mỗi thủ tục lưu cần làm rõ các vấn đề sau đây:*
  + *Mô tả, mục đích sử dụng*
  + *Tham số vào*
  + *Kết quả ra*

#### Sp\_BangDiem()

|  |  |
| --- | --- |
| **Mã nguồn** | CREATE PROC sp\_BangDiem(@MaKH INT)  AS BEGIN  SELECT  nh.MaNH,  nh.HoTen,  hv.Diem  FROM HocVien hv  JOIN NguoiHoc nh ON nh.MaNH=hv.MaNH  WHERE hv.MaKH = @MaKH  ORDER BY hv.Diem DESC  END |
| **Mô tả** | Proc này được sử dụng để truy vấn bảng điểm của khóa học |
| **Tham số** | @MaKH là mã khóa học cần truy vấn bảng điểm |
| **Kết quả** | [MaNH, TenNH, Diem] |

#### Sp\_DoanhThu()

|  |  |
| --- | --- |
| **Mã nguồn** |  |
| **Mô tả** |  |
| **Tham số** |  |
| **Kết quả** |  |

#### Sp\_LuongNguoiHoc()

|  |  |
| --- | --- |
| **Mã nguồn** |  |
| **Mô tả** |  |
| **Tham số** |  |
| **Kết quả** |  |

#### Sp\_DiemChuyenDe()

|  |  |
| --- | --- |
| **Mã nguồn** |  |
| **Mô tả** |  |
| **Tham số** |  |
| **Kết quả** |  |
|  |  |
|  |  |

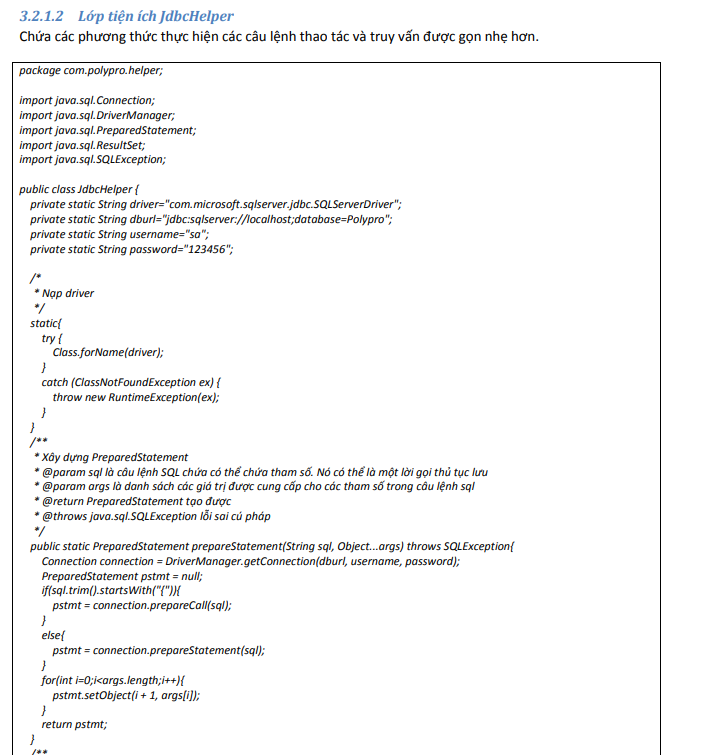
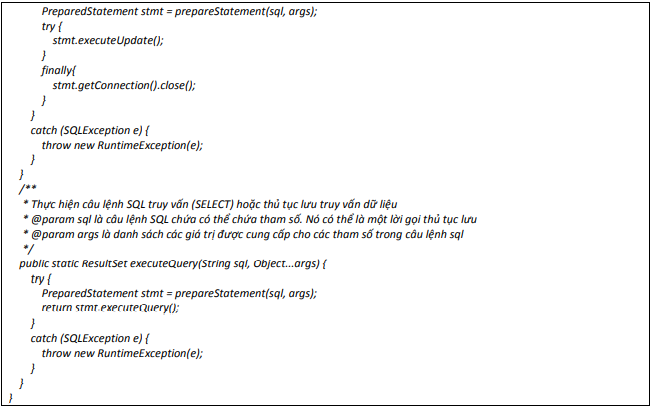
## 

## Lập trình CSDL

### Mô hình tổ chức lập trình CSDL

* *Hình và mô tả (DAO+JdbcHelper+Entity+CSDL)*
* **

### JdbcHelper (XJdbc)

* *Mô tả lớp XJdbc*
* *Hướng dẫn sử dụng các phương thức tiện ích của XJdbc*
* **
* public static void executeUpdate(String sql, Object...args) {
* try {
* **

**Mô tả:**

XJdbc là lớp tiện ích hỗ trợ lập trình JDBC. Lớp này gồm 3 phương thức được mô tả trong phần hướng dẫn sử dụng sau đây.

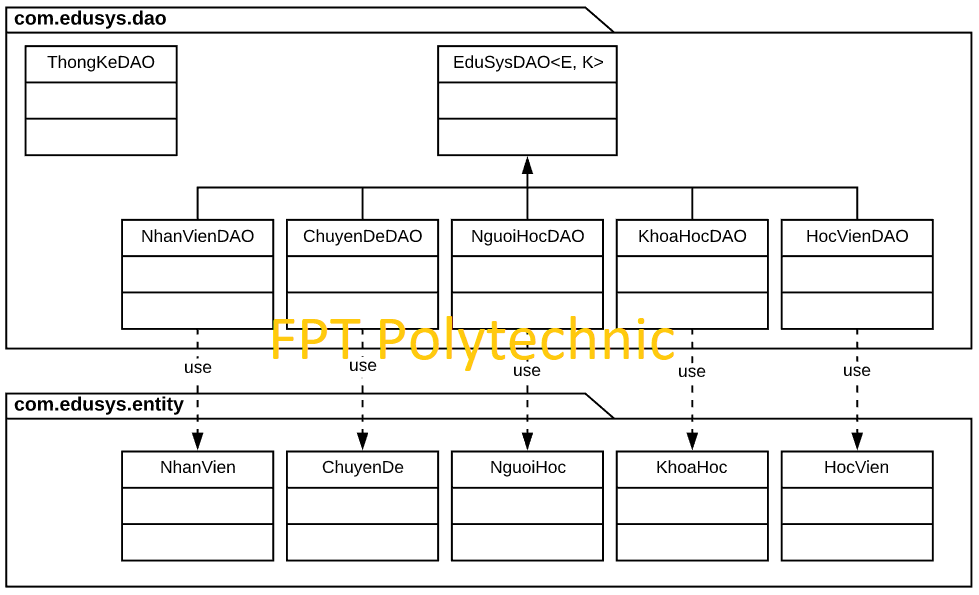
**Hướng dẫn sử dụng:**

* XJdbc.**query**(String sql, Object…args): ResultSet
  + Truy vấn dữ liệu từ CSDL
  + Tham số sql: câu lệnh SELECT truy vấn có chứa ?
  + Tham số: args là các giá trị cung cấp cho các dấu ? của SQL
  + Kết quả: ResultSet
* XJdbc.**update**(String sql, Object…args): int
  + Thao tác CSDL (INSERT, UPDATE, DELETE)
  + Tham số sql: câu lệnh INSERT, UPDATE, DELETE có chứa ?
  + Tham số: args là các giá trị cung cấp cho các dấu ? của SQL
  + Kết quả: số bản ghi có ảnh hưởng
* XJdbc.**value**(String sql, Object…args): Object
  + Truy vấn một giá trị từ CSDL
  + Tham số sql: câu lệnh SELECT truy vấn có chứa ?
  + Tham số: args là các giá trị cung cấp cho các dấu ? của SQL
  + Kết quả: đối tượng chứa giá trị truy vấn

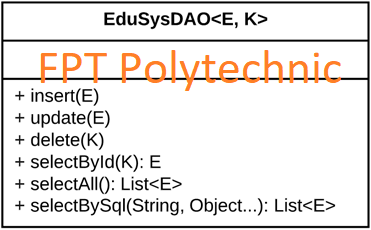
### Entity Class và DAO

* *Mỗi mục cần làm rõ 2 vấn đề sau*
  + *Mô hình và mô tả lớp thực thể*
  + *Mô hình và mô tả lớp DAO*

#### Class Diagram



#### EduSysDAO

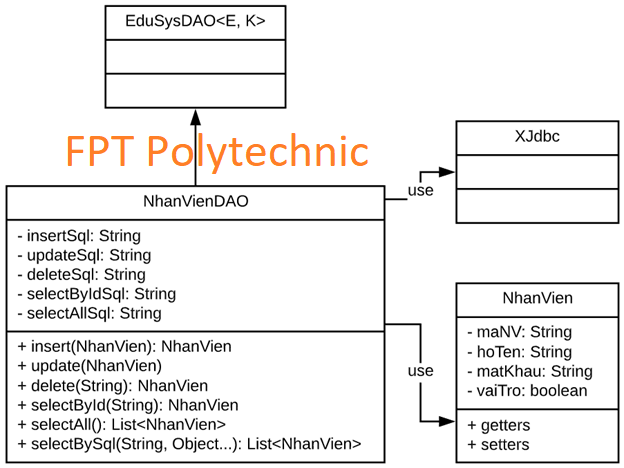


EduSysDAO là lớp trừu tượng, tổng quát (Generic) đặc tả những hành động truy xuất CSDL cơ bản nhất mà một lớp DAO cụ thể phải có. Các phương thức thực hiện truy xuất CSDL được mô tả như sau

|  |  |  |
| --- | --- | --- |
| **TT** | **Phương thức** | **Mô tả hoạt động** |
| 1 | insert(E) | Thêm mới một bản ghi với dữ liệu là entity E |
| 2 | update(E) | Cập nhật một bản ghi với dữ liệu là entity E |
| 3 | delete(K) | Xóa một bản ghi với mã K |
| 4 | selectById(K): E | Truy vấn một bản ghi với mã là K |
| 5 | selectAll(): List<E> | Truy vấn tất cả các bản ghi |
| 6 | selectBySql(String, Object…): List<E> | Truy vấn các bản ghi tùy vào sql và args |

Trên đây chỉ là các hoạt động truy xuất CSDL cơ bản. Trong một lớp DAO cụ thể, các DAO con sẽ viết mã cho các phương thức này với E là lớp Entity và K là kiểu khóa chính cụ thể đồng thời có thể bổ sung thêm các hoạt động truy xuất đặc thù riêng của từng DAO cần thiết.

#### NhanVien và NhanVienDAO



**Diễn giải**

|  |  |
| --- | --- |
| **Thành phần** | **Mô tả** |
| NhanVien | Là lớp thực thể mô tả dữ liệu nhân viên |
| NhanVienDAO | Là lớp DAO cụ thể hóa các hoạt động truy xuất CSDL với bảng NhanVien   * E được cụ thể hóa là NhanVien * K được cụ thể hóa là String * SQL là các câu lệnh thao tác dữ liệu trên bảng NhanVien |

#### ChuyenDe & ChuyenDeDAO

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package DAO;

import MODEL.ChuyenDe;

import com.edusys.utils.XJdbcHelper;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.ArrayList;

import java.util.List;

import java.util.logging.Level;

import java.util.logging.Logger;

/\*\*

\*

\* @author nhatt

\*/

public class ChuyenDeDAO extends EduSysDAO<ChuyenDe, String>{

String INSERT\_SQL = "INSERT INTO ChuyenDe(MaCD,TenCD,HocPhi,ThoiLuong,Hinh,MoTa) VALUES(?,?,?,?,?,?)";

String UPDATE\_SQL = "UPDATE ChuyenDe SET TenCD =? ,HocPhi = ?, ThoiLuong = ?,Hinh=?,MoTa=? WHERE MaCD =?";

String DELETE\_SQL = "DELETE FROM ChuyenDe WHERE MaCD =?";

String SELECT\_ALL\_SQL ="SELECT \* FROM ChuyenDe";

String SELECT\_BY\_ID\_SQL = "SELECT \* FROM ChuyenDe WHERE MaCD = ?";

@Override

public void insert(ChuyenDe entity) {

XJdbcHelper.update(INSERT\_SQL, entity.getMaCD(),entity.getTenCD(),entity.getHocPhi(),entity.getThoiLuong(),entity.getHinh(),entity.getMoTa());

}

@Override

public void update(ChuyenDe entity) {

XJdbcHelper.update(UPDATE\_SQL, entity.getTenCD(),entity.getHocPhi(),entity.getThoiLuong(),entity.getHinh(),entity.getMoTa(),entity.getMaCD());

}

@Override

public void delete(String id) {

XJdbcHelper.update(DELETE\_SQL,id);

}

@Override

public List<ChuyenDe> selectAll() {

return selectBySQL(SELECT\_ALL\_SQL);

}

@Override

public ChuyenDe selectById(String id) {

List<ChuyenDe> list = selectBySQL(SELECT\_BY\_ID\_SQL, id);

return list.size() > 0 ? list.get(0) : null;

}

@Override

protected List<ChuyenDe> selectBySQL(String sql, Object... args) {

List<ChuyenDe> list = new ArrayList<ChuyenDe>();

try {

ResultSet rs = null;

try {

rs = XJdbcHelper.query(sql, args);

while(rs.next()){

ChuyenDe cd = new ChuyenDe(rs.getString("MaCD"),rs.getString("TenCD"),rs.getDouble("HocPhi"),rs.getInt("ThoiLuong"),rs.getString("Hinh"),rs.getString("MoTa"));

list.add(cd);

}

} finally {

rs.getStatement().getConnection().close();

}

} catch (Exception e) {

throw new RuntimeException(e);

}

return list;

}

public List<ChuyenDe> selectByKeywords(String key){

String sql = "SELECT \* FROM ChuyenDe WHERE HoTen LIKE ?";

return this.selectBySQL(sql, "%"+key+"%");

}

}

…

#### NguoiHoc & NguoiHocDAO

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package DAO;

import MODEL.NguoiHoc;

import com.edusys.utils.XDate;

import com.edusys.utils.XJdbcHelper;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.ArrayList;

import java.util.Date;

import java.util.List;

import java.util.logging.Level;

import java.util.logging.Logger;

/\*\*

\*

\* @author nhatt

\*/

public class NguoiHocDAO extends EduSysDAO<NguoiHoc, String> {

java.sql.Date sqlDate;

String INSERT\_SQL = "INSERT INTO NguoiHoc(MaNH,HoTen,GioiTinh,NgaySinh,DienThoai,Email,GhiChu,MaNV,NgayDK) VALUES(?,?,?,?,?,?,?,?,?)";

String UPDATE\_SQL = "UPDATE NguoiHoc SET HoTen =? ,GioiTinh = ?, NgaySinh = ?,DienThoai=?,Email=?,GhiChu=?,MaNV=?,NgayDK=? WHERE MaNH =?";

String DELETE\_SQL = "DELETE FROM NguoiHoc WHERE MaNH =?";

String SELECT\_ALL\_SQL ="SELECT \* FROM NguoiHoc";

String SELECT\_BY\_ID\_SQL = "SELECT \* FROM NguoiHoc WHERE MaNH = ?";

@Override

public void insert(NguoiHoc entity) {

XJdbcHelper.update(INSERT\_SQL, entity.getMaNH(),entity.getHoTen(),entity.getGioiTinh(),entity.getNgaySinh(),entity.getDienThoai(),entity.getEmail(),entity.getGhiChu(),entity.getMaNV(),entity.getNgayDK());

}

public Date convertDate(java.util.Date date){

sqlDate = new java.sql.Date(date.getTime());

return sqlDate;

}

@Override

public void update(NguoiHoc entity) {

XJdbcHelper.update(UPDATE\_SQL, entity.getHoTen(),entity.getGioiTinh(),entity.getNgaySinh(),entity.getDienThoai(),entity.getEmail(),entity.getGhiChu(),entity.getMaNV(),entity.getNgayDK(),entity.getMaNH());

}

@Override

public void delete(String id) {

XJdbcHelper.update(DELETE\_SQL,id);

}

@Override

public List<NguoiHoc> selectAll() {

return this.selectBySQL(SELECT\_ALL\_SQL);

}

@Override

public NguoiHoc selectById(String id) {

List<NguoiHoc> list = this.selectBySQL(SELECT\_BY\_ID\_SQL, id);

return list.size() > 0 ? list.get(0) : null;

}

@Override

protected List<NguoiHoc> selectBySQL(String sql, Object... args) {

List<NguoiHoc> list = new ArrayList<NguoiHoc>();

try {

ResultSet rs = null;

try {

rs = XJdbcHelper.query(sql, args);

while(rs.next()){

NguoiHoc nv = new NguoiHoc(rs.getString("MaNH"),rs.getString("HoTen"),rs.getBoolean("GioiTinh"),rs.getDate("NgaySinh"),rs.getString("DienThoai"),rs.getString("Email"),rs.getString("GhiChu"),rs.getString("MaNV"),rs.getDate("NgayDK"));

list.add(nv);

}

} finally {

rs.getStatement().getConnection().close();

}

} catch (Exception e) {

throw new RuntimeException(e);

}

return list;

}

public List<NguoiHoc> selectByKeywords(String key){

String sql = "SELECT \* FROM NguoiHoc WHERE HoTen LIKE ?";

return this.selectBySQL(sql, "%"+key+"%");

}

public List<NguoiHoc> selectNotInCourse(int makh, String keyword) {

String sql="SELECT \* FROM NguoiHoc "

+ " WHERE HoTen LIKE ? AND "

+ " MaNH NOT IN (SELECT MaNH FROM HocVien WHERE MaKH=?)";

return this.selectBySQL(sql, "%"+keyword+"%", makh);

}

}

…

#### KhoaHoc & KhoaHocDAO

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package DAO;

import MODEL.KhoaHoc;

import com.edusys.utils.XJdbcHelper;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.ArrayList;

import java.util.Date;

import java.util.List;

import java.util.logging.Level;

import java.util.logging.Logger;

/\*\*

\*

\* @author nhatt

\*/

public class KhoaHocDAO extends EduSysDAO<KhoaHoc, Integer>{

String INSERT\_SQL = "INSERT INTO KhoaHoc(MaCD,HocPhi,ThoiLuong,NgayKG,GhiChu,MaNV) VALUES(?,?,?,?,?,?)";

String UPDATE\_SQL = "UPDATE KhoaHoc SET MaCD =? ,HocPhi = ?, ThoiLuong = ?,NgayKG=?,GhiChu=?,MaNV=? WHERE MaKH =?";

String DELETE\_SQL = "DELETE FROM KhoaHoc WHERE MaKH =?";

String SELECT\_ALL\_SQL ="SELECT \* FROM KhoaHoc";

String SELECT\_BY\_ID\_SQL = "SELECT \* FROM KhoaHoc WHERE MaKH = ?";

@Override

public void insert(KhoaHoc entity) {

XJdbcHelper.update(INSERT\_SQL, entity.getMaCD(),entity.getHocPhi(),entity.getThoiLuong(),entity.getNgayKG(),entity.getGhiChu(),entity.getMaNV());

}

java.sql.Date sqlDate;

public Date convertDate(java.util.Date date){

sqlDate = new java.sql.Date(date.getTime());

return sqlDate;

}

@Override

public void update(KhoaHoc entity) {

XJdbcHelper.update(UPDATE\_SQL, entity.getMaCD(),entity.getHocPhi(),entity.getThoiLuong(),entity.getNgayKG(),entity.getGhiChu(),entity.getMaNV(),entity.getMaKH());

}

@Override

public void delete(Integer id) {

XJdbcHelper.update(DELETE\_SQL,id);

}

@Override

public List<KhoaHoc> selectAll() {

return this.selectBySQL(SELECT\_ALL\_SQL);

}

@Override

public KhoaHoc selectById(Integer id) {

List<KhoaHoc> list = this.selectBySQL(SELECT\_BY\_ID\_SQL, id);

return list.size() > 0 ? list.get(0) : null;

}

@Override

protected List<KhoaHoc> selectBySQL(String sql, Object... args) {

List<KhoaHoc> list = new ArrayList<KhoaHoc>();

try {

ResultSet rs = null;

try {

rs =XJdbcHelper.query(sql, args);

while(rs.next()){

KhoaHoc kh = new KhoaHoc(rs.getInt("MaKH"),rs.getString("MaCD"),rs.getDouble("HocPhi"),rs.getInt("ThoiLuong"),rs.getDate("NgayKG"),rs.getString("GhiChu"),rs.getString("MaNV"),rs.getDate("NgayTao"));

list.add(kh);

}

} finally {

rs.getStatement().getConnection().close();

}

} catch (Exception e) {

throw new RuntimeException(e);

}

return list;

}

public List<KhoaHoc> selectByChuyenDe(String maCD){

String sql="SELECT \* FROM KhoaHoc WHERE MaCD=?";

return this.selectBySQL(sql, maCD);

}

public List<Integer> selectYears() {

String sql="SELECT DISTINCT year(NgayKG) Year FROM KhoaHoc ORDER BY Year DESC";

List<Integer> list=new ArrayList<>();

try {

ResultSet rs = XJdbcHelper.query(sql);

while(rs.next()){

list.add(rs.getInt(1));

}

rs.getStatement().getConnection().close();

return list;

}

catch (SQLException ex) {

throw new RuntimeException(ex);

}

}

}

…

#### HocVien & HocVienDAO

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package DAO;

import MODEL.HocVien;

import com.edusys.utils.XJdbcHelper;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.ArrayList;

import java.util.List;

import java.util.logging.Level;

import java.util.logging.Logger;

public class HocVienDAO extends EduSysDAO<HocVien, Integer> {

String INSERT\_SQL = "INSERT INTO HocVien(MaKH,MaNH,Diem) VALUES(?,?,?)";

String UPDATE\_SQL = "UPDATE HocVien SET MaKH =? ,MaNH = ?, Diem = ? WHERE MaHV =?";

String DELETE\_SQL = "DELETE FROM HocVien WHERE MaHV =?";

String SELECT\_ALL\_SQL ="SELECT \* FROM HocVien";

String SELECT\_BY\_ID\_SQL = "SELECT \* FROM HocVien WHERE MaHV = ?";

@Override

public void insert(HocVien entity) {

XJdbcHelper.update(INSERT\_SQL, entity.getMaKH(),entity.getMaNH(),entity.getDiem());

}

@Override

public void update(HocVien entity) {

XJdbcHelper.update(UPDATE\_SQL, entity.getMaKH(),entity.getMaNH(),entity.getDiem(),entity.getMaHV());

}

@Override

public void delete(Integer id) {

XJdbcHelper.update(DELETE\_SQL,id);

}

@Override

public List<HocVien> selectAll() {

return this.selectBySQL(SELECT\_ALL\_SQL);

}

@Override

public HocVien selectById(Integer id) {

List<HocVien> list = this.selectBySQL(SELECT\_BY\_ID\_SQL, id);

return list.size() > 0 ? list.get(0) : null;

}

@Override

protected List<HocVien> selectBySQL(String sql, Object... args) {

List<HocVien> list = new ArrayList<HocVien>();

try {

ResultSet rs = null;

try {

rs = XJdbcHelper.query(sql, args);

while(rs.next()){

HocVien hv = new HocVien(rs.getInt("MaHV"),rs.getInt("MaKH"),rs.getString("MaNH"),rs.getDouble("Diem"));

list.add(hv);

}

} finally {

rs.getStatement().getConnection().close();

}

} catch (Exception e) {

throw new RuntimeException(e);

}

return list;

}

public List<HocVien> selectByKhoaHoc(int maKH) {

String sql="SELECT \* FROM HocVien WHERE MaKH=?";

return this.selectBySQL(sql, maKH);

}

}

…

#### ThongKeDAO

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package DAO;

import com.edusys.utils.XJdbcHelper;

import java.util.ArrayList;

import java.util.List;

import java.sql.ResultSet;

/\*\*

\*

\* @author nhatt

\*/

public class ThongKeDAO {

private List<Object[]> getListOfArray(String sql,String[] cols,Object...args){

try {

List<Object[]> list = new ArrayList<>();

ResultSet rs = XJdbcHelper.query(sql, args);

while(rs.next()){

Object[] vals = new Object[cols.length];

for(int i = 0;i<cols.length;i++){

vals[i] = rs.getObject(cols[i]);

}

list.add(vals);

}

rs.getStatement().getConnection().close();

return list;

} catch (Exception e) {

throw new RuntimeException(e);

}

}

public List<Object[]> getBangDiem(Integer maKH){

String sql = "{CALL SP\_BangDiem(?)}";

String[] cols ={"MaNH","HoTen","Diem"};

return this.getListOfArray(sql, cols, maKH);

}

public List<Object[]> getLuongNguoiHoc(){

String sql= "{CALL SP\_LuongNguoiHoc}";

String[] cols = {"Nam","SoLuong","DauTien","CuoiCung"};

return this.getListOfArray(sql, cols);

}

public List<Object[]> getDiemChuyenDe(){

String sql = "{CALL SP\_ThongKeDiem}";

String[] cols ={"ChuyenDe","SoHV","ThapNhat","CaoNhat","TrungBinh"};

return this.getListOfArray(sql, cols);

}

public List<Object[]> getDoanhThu(int nam){

String sql = "{CALL SP\_ThongKeDoanhThu(?)}";

String[] cols ={"ChuyenDe","SoKH","SoHV","DoanhThu","ThapNhat","CaoNhat","TrungBinh"};

return this.getListOfArray(sql, cols, nam);

}

}

….

## Thư viện tiện ích

* *Mô tả các lớp tiện ích sau tương tự JdbcHelper(XJdbc) ở mục 4.3.2*
  + *Ximage*
  + */\**
  + *\* To change this license header, choose License Headers in Project Properties.*
  + *\* To change this template file, choose Tools | Templates*
  + *\* and open the template in the editor.*
  + *\*/*
  + *package com.edusys.utils;*
  + *import java.awt.Image;*
  + *import java.io.File;*
  + *import java.net.URL;*
  + *import java.nio.file.Files;*
  + *import java.nio.file.Path;*
  + *import java.nio.file.Paths;*
  + *import java.nio.file.StandardCopyOption;*
  + *import javax.swing.ImageIcon;*
  + */\*\**
  + *\**
  + *\* @author nhatt*
  + *\*/*
  + *public class XImage {*
  + *public static Image getAppIcon(){*
  + *URL url = XImage.class.getResource("/com/edusys/icon/fpt.png");*
  + *return new ImageIcon(url).getImage();*
  + *}*
  + *public static void save(File src){*
  + *File dst = new File("logos", src.getName());*
  + *if(!dst.getParentFile().exists()){*
  + *dst.getParentFile().mkdirs();*
  + *}*
  + *try {*
  + *Path from = Paths.get(src.getAbsolutePath());*
  + *Path to = Paths.get(dst.getAbsolutePath());*
  + *Files.copy(from, to, StandardCopyOption.REPLACE\_EXISTING);*
  + *}*
  + *catch (Exception ex) {*
  + *throw new RuntimeException(ex);*
  + *}*
  + *}*
  + *public static ImageIcon read(String fileName){*
  + *File path = new File("logos", fileName);*
  + *return new ImageIcon(path.getAbsolutePath());*
  + *}*
  + *}*
  + *Xdate*
* */\**
* *\* To change this license header, choose License Headers in Project Properties.*
* *\* To change this template file, choose Tools | Templates*
* *\* and open the template in the editor.*
* *\*/*
* *package com.edusys.utils;*
* *import java.text.SimpleDateFormat;*
* *import java.util.Date;*
* *public class XDate {*
* *static SimpleDateFormat formater = new SimpleDateFormat();*
* *public static Date toDate(String date, String pattern){*
* *try {*
* *if(date == null){*
* *return XDate.now();*
* *}*
* *if(pattern.length()>0){*
* *formater.applyPattern(pattern);*
* *}*
* *return formater.parse(date);*
* *} catch (Exception e) {*
* *throw new RuntimeException(e);*
* *}*
* *}*
* *public static String toString(Date date, String pattern){*
* *if(date == null){*
* *date = XDate.now();*
* *}*
* *if(pattern.length()>0){*
* *formater.applyPattern(pattern);*
* *}*
* *return formater.format(date);*
* *}*
* *public static Date addDays(Date date,long days){*
* *date.setTime(date.getTime() + days\*24\*60\*60\*1000);*
* *return date;*
* *}*
* *public static Date now(){*
* *return new Date();*
* *}*
* *}*
  + *MsgBox*
  + */\**
  + *\* To change this license header, choose License Headers in Project Properties.*
  + *\* To change this template file, choose Tools | Templates*
  + *\* and open the template in the editor.*
  + *\*/*
  + *package com.edusys.utils;*
  + *import java.awt.Component;*
  + *import javax.swing.JOptionPane;*
  + */\*\**
  + *\**
  + *\* @author nhatt*
  + *\*/*
  + *public class MsgBox {*
  + *public static void alert(Component parent, String message){*
  + *JOptionPane.showMessageDialog(parent, message,"Hệ thống quản lý đào tạo",JOptionPane.INFORMATION\_MESSAGE);*
  + *}*
  + *public static boolean confirm(Component parent, String message){*
  + *int result = JOptionPane.showConfirmDialog(parent, message,"Hệ thống quản lý đào tạo",JOptionPane.YES\_NO\_OPTION,*
  + *JOptionPane.QUESTION\_MESSAGE);*
  + *return result ==JOptionPane.YES\_NO\_OPTION;*
  + *}*
  + *public static String prompt(Component parentComponent,String message){*
  + *return JOptionPane.showInputDialog(parentComponent,message,"Hệ thống quản lý đào tạo",JOptionPane.INFORMATION\_MESSAGE);*
  + *}*
  + *}*
  + *Auth*
  + */\**
  + *\* To change this license header, choose License Headers in Project Properties.*
  + *\* To change this template file, choose Tools | Templates*
  + *\* and open the template in the editor.*
  + *\*/*
  + *package com.edusys.utils;*
  + *import MODEL.NhanVien;*
  + *public class Auth {*
  + *public static NhanVien user = null ;*
  + *public static void clear(){*
  + *Auth.user = null;*
  + *}*
  + *public static boolean isLogin(){*
  + *return Auth.user != null;*
  + *}*
  + *public static boolean isManager(){*
  + *return Auth.isLogin() && user.getVaiTro();*
  + *}*
  + *}*

## Lập trình nghiệp vụ

* *Mỗi cửa sổ chức năng sau đây cần* 
  + *Sơ đồ tổ chức công nghệ*
  + *Mô tả tất cả các hành vi của người sử dụng tương tác lên các chức năng của các giao diện (có thể sử dụng lưu đồ thuật toán hoạt các loại sơ đồ Activity Diagram, Sequence Diagram - các loại sơ đồ này được học trong môn Nhập môn kỹ thuật phần mềm)*

### Cửa sổ chính EduSysJFrame

|  |  |  |
| --- | --- | --- |
| **TT** | **Phương thức** | **Mô tả** |
| 1 | init() | Mở cửa sổ chào  Mở cửa sổ đăng nhập  Bắt đầu hiển thị đồng hồ hệ thống |
| 2 | openDoiMatKhau() | Mở cửa sổ đổi mật khẩu |
|  | … |  |

### Các cửa sổ hỗ trợ tổ chức

#### ChaoDialog

|  |  |  |
| --- | --- | --- |
| **TT** | **Phương thức** | **Mô tả** |
| 1 | init() | Hiển thị cửa sổ ở giữa màn hình  Tăng giá trị ProgressBar, đóng cửa sổ khi giá trị bằng max |
| 2 | … |  |

#### DangNhapJDialog

|  |  |  |
| --- | --- | --- |
| **TT** | **Phương thức** | **Mô tả** |
| 1 | init() | Hiển thị cửa sổ ở giữa màn hình |
| 2 | … |  |

#### DoiMatKhauJDialog

|  |  |  |
| --- | --- | --- |
| **TT** | **Phương thức** | **Mô tả** |
| 1 | init() | Hiển thị cửa sổ ở giữa màn hình |
| 2 | … |  |

#### GioiThieuJDialog

|  |  |  |
| --- | --- | --- |
| **TT** | **Phương thức** | **Mô tả** |
| 1 | init() | Hiển thị cửa sổ ở giữa màn hình |
| 2 | … |  |

### Các cửa sổ chức năng quản lý

#### NhanVienJDialog

|  |  |  |
| --- | --- | --- |
| **TT** | **Phương thức** | **Mô tả** |
| 1 | init() | Hiển thị cửa sổ ở giữa màn hình |
| 2 | Insert() | Thêm mới nhân viên |
| 3 | Update() | Sửa thông tin nhân viên |
| 4 | Delete() | Xóa nhân vien |
| 5 | getForm() | Lấy dữ liệu từ form |
| 6 | setForm() | Thay đổi thông tin trên form |
| 7 | clearForm() | Xóa form |
| 8 | Edit() | Sửa thông tin trong database |
| 9 | fillTable() | Đỗ dữ liêu lên table |
| 10 | updateStatus() | Sửa thông tin |
| 11 | First() | Tiến tới vị trí đầu |
| 12 | Prev() | Lùi về trước 1 đơn vị |
| 13 | Next() | Lùi về sau 1 đơn vị |
| 14 | Last() | Tiến tới vị trí cuối |
|  | … |  |

#### ChuyenDeJDialog

|  |  |  |
| --- | --- | --- |
| **TT** | **Phương thức** | **Mô tả** |
| 1 | init() | Hiển thị cửa sổ ở giữa màn hình |
| 2 | Insert() | Thêm mới Chuyen đề |
| 3 | Update() | Sửa thông tin Chuyên đề |
| 4 | Delete() | Xóa chuyên đề |
| 5 | getForm() | Lấy dữ liệu từ form |
| 6 | setForm() | Thay đổi thông tin trên form |
| 7 | clearForm() | Xóa form |
| 8 | Edit() | Sửa thông tin trong database |
| 9 | fillTable() | Đỗ dữ liêu lên table |
| 10 | updateStatus() | Sửa thông tin |
| 11 | First() | Tiến tới vị trí đầu |
| 12 | Prev() | Lùi về trước 1 đơn vị |
| 13 | Next() | Lùi về sau 1 đơn vị |
| 14 | Last() | Tiến tới vị trí cuối |
|  | … |  |

#### NguoiHocJDialog

|  |  |  |
| --- | --- | --- |
| **TT** | **Phương thức** | **Mô tả** |
| 1 | init() | Hiển thị cửa sổ ở giữa màn hình |
| 2 | Insert() | Thêm mới nhân viên |
| 3 | Update() | Sửa thông tin nhân viên |
| 4 | Delete() | Xóa nhân vien |
| 5 | getForm() | Lấy dữ liệu từ form |
| 6 | setForm() | Thay đổi thông tin trên form |
| 7 | clearForm() | Xóa form |
| 8 | Edit() | Sửa thông tin trong database |
| 9 | fillTable() | Đỗ dữ liêu lên table |
| 10 | updateStatus() | Sửa thông tin |
| 11 | First() | Tiến tới vị trí đầu |
| 12 | Prev() | Lùi về trước 1 đơn vị |
| 13 | Next() | Lùi về sau 1 đơn vị |
| 14 | Last() | Tiến tới vị trí cuối |
|  | … |  |

#### KhoaHocJDialog

|  |  |  |
| --- | --- | --- |
| **TT** | **Phương thức** | **Mô tả** |
| 1 | init() | Hiển thị cửa sổ ở giữa màn hình |
| 2 | Insert() | Thêm mới nhân viên |
| 3 | Update() | Sửa thông tin nhân viên |
| 4 | Delete() | Xóa nhân vien |
| 5 | getForm() | Lấy dữ liệu từ form |
| 6 | setForm() | Thay đổi thông tin trên form |
| 7 | clearForm() | Xóa form |
| 8 | Edit() | Sửa thông tin trong database |
| 9 | fillTable() | Đỗ dữ liêu lên table |
| 10 | updateStatus() | Sửa thông tin |
| 11 | First() | Tiến tới vị trí đầu |
| 12 | Prev() | Lùi về trước 1 đơn vị |
| 13 | Next() | Lùi về sau 1 đơn vị |
| 14 | Last() | Tiến tới vị trí cuối |
|  | … |  |

#### HocVienJDialog

|  |  |  |
| --- | --- | --- |
| **TT** | **Phương thức** | **Mô tả** |
| 1 | init() | Hiển thị cửa sổ ở giữa màn hình |
| 2 | Insert() | Thêm mới nhân viên |
| 3 | Update() | Sửa thông tin nhân viên |
| 4 | Delete() | Xóa nhân vien |
| 5 | getForm() | Lấy dữ liệu từ form |
| 6 | setForm() | Thay đổi thông tin trên form |
| 7 | clearForm() | Xóa form |
| 8 | Edit() | Sửa thông tin trong database |
| 9 | fillTable() | Đỗ dữ liêu lên table |
| 10 | updateStatus() | Sửa thông tin |
| 11 | First() | Tiến tới vị trí đầu |
| 12 | Prev() | Lùi về trước 1 đơn vị |
| 13 | Next() | Lùi về sau 1 đơn vị |
| 14 | Last() | Tiến tới vị trí cuối |
|  | … |  |

### Cửa sổ chức năng tổng hợp - thống kê

|  |  |  |
| --- | --- | --- |
| **TT** | **Phương thức** | **Mô tả** |
| 1 |  | Hiển thị cửa sổ ở giữa màn hình |
| 2 |  |  |

# Kiểm thử phần mềm và sửa lỗi

## DangNhapJDialog

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TH** | **Mô tả tình huống** | **Dữ liệu mẫu** | **Kết quả trông đợi** | **Fix** |
| 1 | Đăng nhập để trống tên và mật khẩu |  | Yêu cầu nhập | ok |
| 2 | Đăng nhập sai tên | TeoVN, songsong | Sai tên đăng nhập | not |
| 3 | Đăng nhập sai mật khẩu | TeoNV, 123 | Sai mật khẩu | 95% |
| 4 | Đăng nhập đúng | TeoNV, songlong | Đóng cửa sổ |  |

## DoiMatKhauJDialog

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TH** | **Mô tả tình huống** | **Dữ liệu mẫu** | **Kết quả trông đợi** | **Fix** |
| 1 | Nhập tài khoản | TeoNV | Xác nhận có tồn tại | OK |
| 2 | Nhập mặt khẩu hiện tại | Songsong | Xác nhận đúng | Ok |
| 3 | Nhập sai pass | Nvac0 | Thông báo lỗi | OK |
| 4 |  |  |  |  |

## NhanVienJDialog

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TH** | **Mô tả tình huống** | **Dữ liệu mẫu** | **Kết quả trông đợi** | **Fix** |
| 1 | Đăng nhập để trống tên và mật khẩu |  | Yêu cầu nhập | ok |
| 2 | Đăng nhập sai tên | TeoVN, songsong | Sai tên đăng nhập | not |
| 3 | Đăng nhập sai mật khẩu | TeoNV, 123 | Sai mật khẩu | 95% |
| 4 | Đăng nhập đúng | TeoNV, songlong | Đóng cửa sổ |  |

## ChuyenDeJDialog

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TH** | **Mô tả tình huống** | **Dữ liệu mẫu** | **Kết quả trông đợi** | **Fix** |
| 1 | Đăng nhập để trống tên và mật khẩu |  | Yêu cầu nhập | ok |
| 2 | Đăng nhập sai tên | TeoVN, songsong | Sai tên đăng nhập | not |
| 3 | Đăng nhập sai mật khẩu | TeoNV, 123 | Sai mật khẩu | 95% |
| 4 | Đăng nhập đúng | TeoNV, songlong | Đóng cửa sổ |  |

## NguoiHocJDialog

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TH** | **Mô tả tình huống** | **Dữ liệu mẫu** | **Kết quả trông đợi** | **Fix** |
| 1 | Đăng nhập để trống tên và mật khẩu |  | Yêu cầu nhập | ok |
| 2 | Đăng nhập sai tên | TeoVN, songsong | Sai tên đăng nhập | not |
| 3 | Đăng nhập sai mật khẩu | TeoNV, 123 | Sai mật khẩu | 95% |
| 4 | Đăng nhập đúng | TeoNV, songlong | Đóng cửa sổ |  |

## KhoaHocJDialog

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TH** | **Mô tả tình huống** | **Dữ liệu mẫu** | **Kết quả trông đợi** | **Fix** |
| 1 | Đăng nhập để trống tên và mật khẩu |  | Yêu cầu nhập | ok |
| 2 | Đăng nhập sai tên | TeoVN, songsong | Sai tên đăng nhập | not |
| 3 | Đăng nhập sai mật khẩu | TeoNV, 123 | Sai mật khẩu | 95% |
| 4 | Đăng nhập đúng | TeoNV, songlong | Đóng cửa sổ |  |

## HocVienJDialog

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TH** | **Mô tả tình huống** | **Dữ liệu mẫu** | **Kết quả trông đợi** | **Fix** |
| 1 | Đăng nhập để trống tên và mật khẩu |  | Yêu cầu nhập | ok |
| 2 | Đăng nhập sai tên | TeoVN, songsong | Sai tên đăng nhập | not |
| 3 | Đăng nhập sai mật khẩu | TeoNV, 123 | Sai mật khẩu | 95% |
| 4 | Đăng nhập đúng | TeoNV, songlong | Đóng cửa sổ |  |

## ThongKeJDialog

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TH** | **Mô tả tình huống** | **Dữ liệu mẫu** | **Kết quả trông đợi** | **Fix** |
| 1 | Đăng nhập để trống tên và mật khẩu |  | Yêu cầu nhập | ok |
| 2 | Đăng nhập sai tên | TeoVN, songsong | Sai tên đăng nhập | not |
| 3 | Đăng nhập sai mật khẩu | TeoNV, 123 | Sai mật khẩu | 95% |
| 4 | Đăng nhập đúng | TeoNV, songlong | Đóng cửa sổ |  |

# Đóng gói và triển khai

## Sản phẩm phần mềm

|  |  |  |
| --- | --- | --- |
| **TT** | **Thành phần** | **Mô tả** |
| 1 | Setup.exe | File cài đặt |
| 2 | EduSys.sql | Cơ sở dữ liệu |
| 3 | Readme.txt | Hướng dẫn cài đặt |
| 4 | Guide.docx | Hướng dẫn sử dụng |

## Hướng dẫn cài đặt

# 

# KẾT LUẬN

DỰ ÁN MANG TÍNH THAM KHẢO

## Khó khăn

QUÁ NHIỀU

## Thuận lợi

CÓ SẲN TÀI LIỆU THAM KHẢO