**Trường Cao Đẳng Công Nghệ Thông Tin**

**Đại học Đà Nẵng**

**Khoa : Công nghệ thông tin**

**BÁO CÁO MÔ TẢ DỰ ÁN**

****

**Môn học :** Dự Án Java

**Giáo viên hướng dẫn :** Trần Minh Thắng

**Sinh viên** : Hà Phước Yên

**Lớp** : 18IF1

Đà Nẵng, thứ 4 ngày 19 tháng 6 năm 2019

**Lời nói đầu**

Để hoàn thành bài tập thực hiện dự án phần mềm quản lý, em đã chọn làm phần mềm quản lý trường học với những mục đích như sau :

+ Có thể lưu lại thông tin các sinh viên theo từng lớp, cũng như thông tin các giáo viên.

+ Cung cấp một số thao tác như : thêm sinh viên (hoặc giáo viên, môn học, lớp học), cập nhật thông tin sinh viên (hoặc giáo viên, môn học, lớp học) , hoặc là xóa sinh viên (hoặc giáo viên, môn học, lớp học) khỏi danh sách trong trường hợp học viên chuyển trường.

Để hoàn thành dự án này, em đã tham khảo nhiều nguồn khác nhau như qua bạn bè, các bài học trên lớp, sự hướng dẫn thêm của thầy Thắng – giáo viên bộ môn.

**Mục lục :**

Lời nói đầu 2

Mục lục 2

Phần I : Phân tích cấu trúc chung 3

1. Giới thiệu chung 3
2. Các đối tượng xuất hiện trong dự án và mối quan hệ 3

Phần II : Triển khai dự án 4

1. Các hàm xuất hiện trong dự án 4
2. Các giao diện quản lý 4

Phần III : Ưu, nhược điểm và hướng phát triển 12

**Phần I : Phân tích cấu trúc chung:**

1. Giới thiệu chung:

Phần mềm quản lý trường học sẽ có các trang như sau:

+ Thông tin sinh viên

+ Thông tin giáo viên

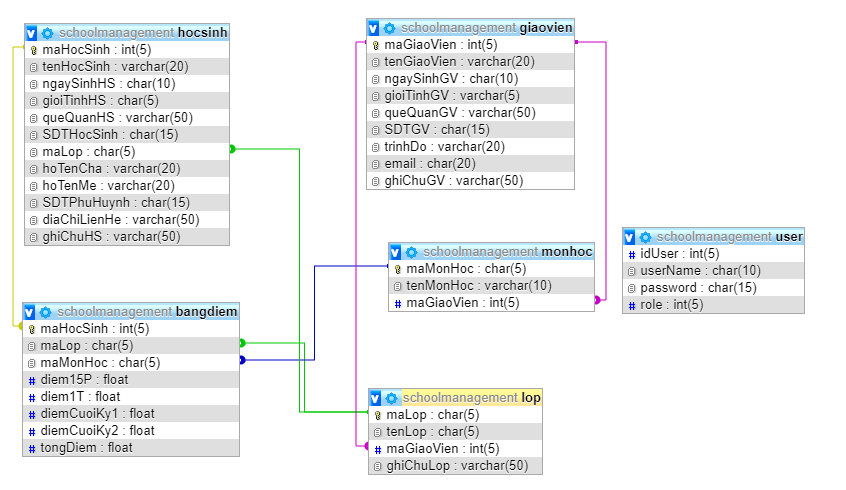
+ Môn học

+ Lớp

+ Điểm thi học phần

Trong mỗi trang sẽ có các giao diện để người dùng có thể thao tác với cơ sở dữ liệu bao gồm các button, textfield, table,….

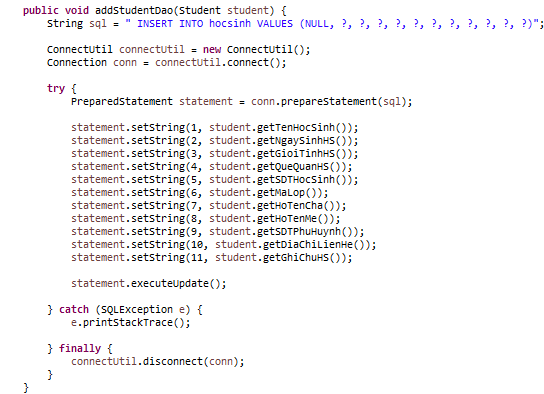
1. Sơ đồ mối quan hệ giữa các đối tượng:



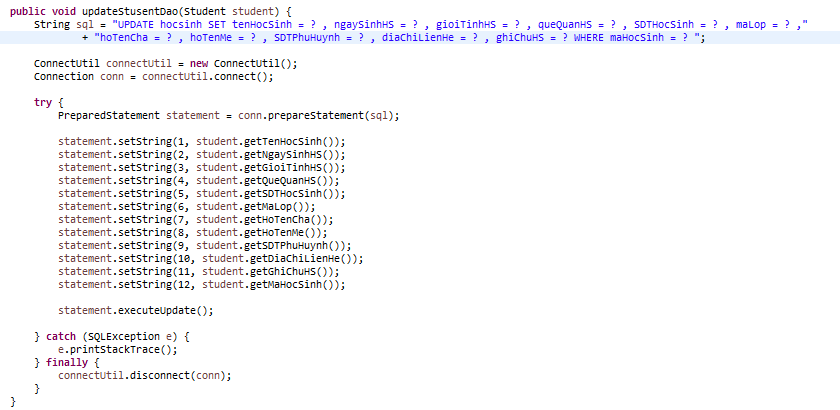
**Phần II : Triển khai dự án**

1. Các hàm xuất hiện trong dự án:

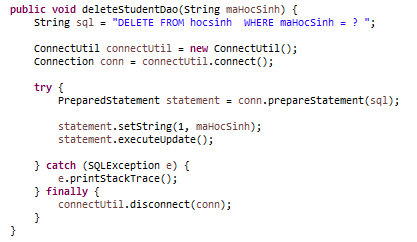
* Hàm thêm ( tạm gọi là ADD để phân biệt) ( Sinh viên, giáo viên, môn học,… ) : hàm này có vai trò thêm đối tượng vào database. Ví dụ dưới đây là thêm sinh viên, sử dụng các phương thức có trong đối tượng là Student.



* Hàm sửa ( UPDATE ) ( Sinh viên, giáo viên, môn học,… ) : hàm này được sử dụng với mục đích sửa lại các thông tin đã có từ trước của các đối tượng trên database.



* Hàm xóa ( DELETE ) ( Sinh viên, giáo viên, môn học,… ) : hàm này được dùng để xóa thông tin các đối tượng đã có trước trên database trong trường hợp sinh viên chuyển trường hoặc giáo viên chuyển nơi công tác,…

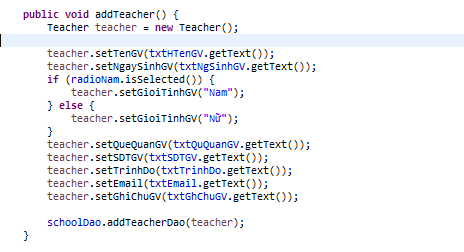


Đây là các hàm cơ bản dùng để tương tác trực tiếp với database. Sau đây là các hàm lấy thông tin từ giao diện để chuẩn bị các yếu tố cần thiết để sử dụng các hàm ở trên. ( Dưới đây đều là các hàm liên quan đến đối tượng sinh viên để làm đại diện cho các đối tượng khác)

* Hàm ghi đè ActionPerformed : hàm được sử dụng với vai trò như điều hướng các tính năng của giao diện ví dụ như kick nút ADD => gọi đến hàm ADD,…



* Hàm addStudent : lấy các thông tin từ giao diện cung cấp cho hàm ADD.

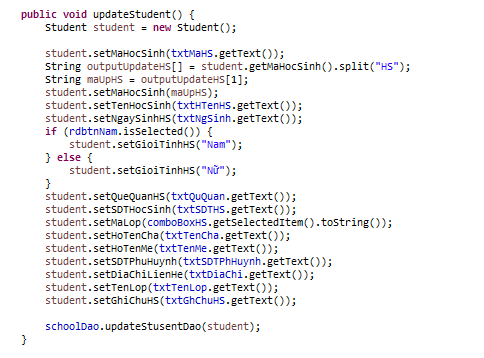


* Hàm updateStudent : dùng kết hợp với thao tác mouseClick để cập nhật thông tin đối tượng mà ở đây chính là sinh viên.

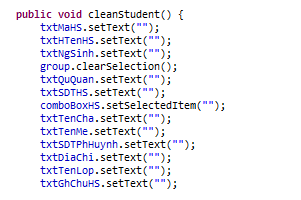
\*mouseClicked: dùng để lấy thông tin từ database và xuất ra vùng nhập\*



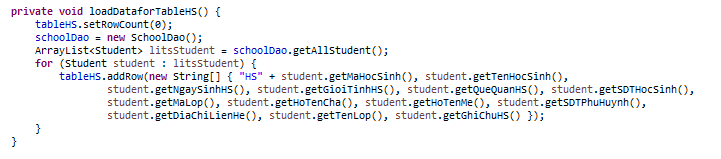
\*updateStudent\*



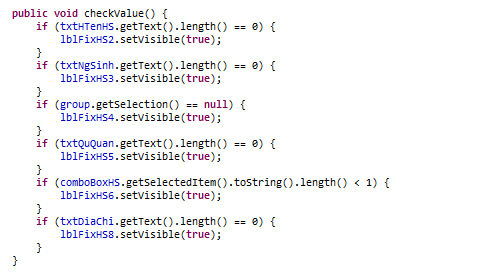
.- Hàm cleanStudent : làm trống vùng nhập.



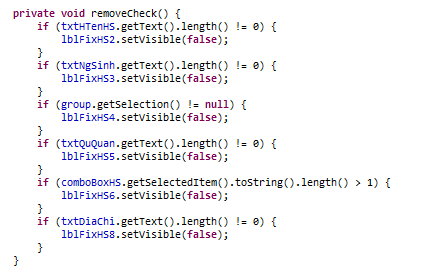
* Hàm dùng để loadDatabase:



* Hàm Check để ràng buộc 1 số ô phải có dữ liệu:

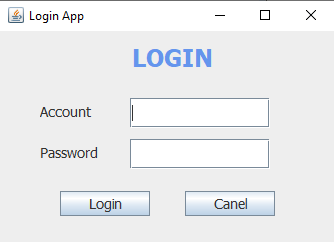


* Hàm RemoveCheck để reset hàm Check lại như ban đầu:

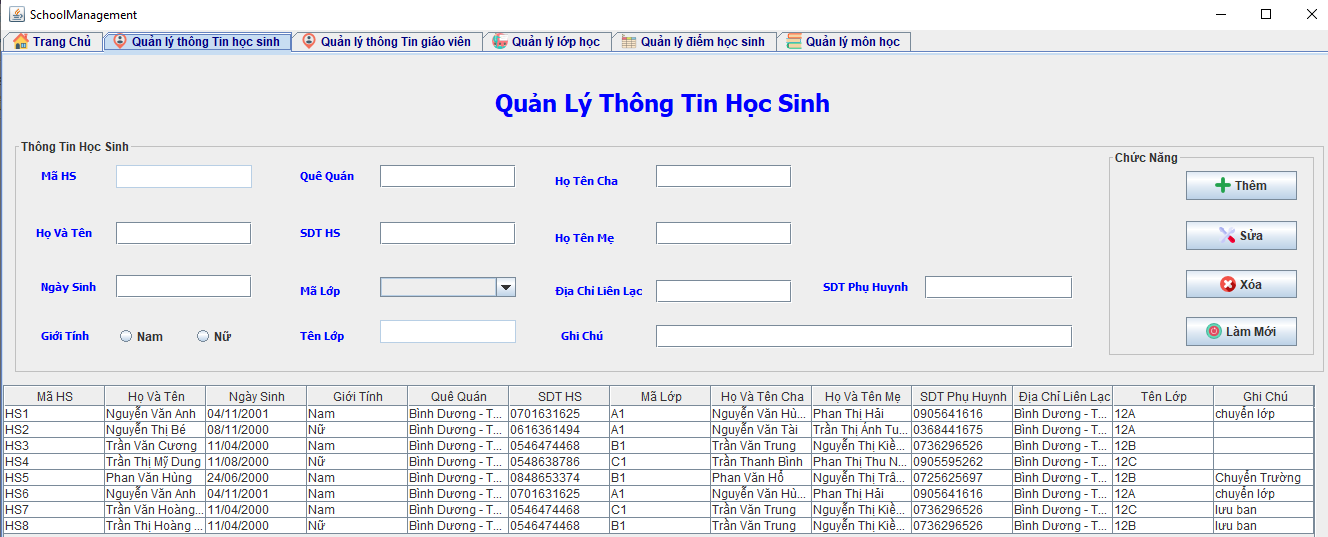


1. Các giao diện của phần mền quản lý :

+ Đăng nhập :



+ Quản lý sinh viên :

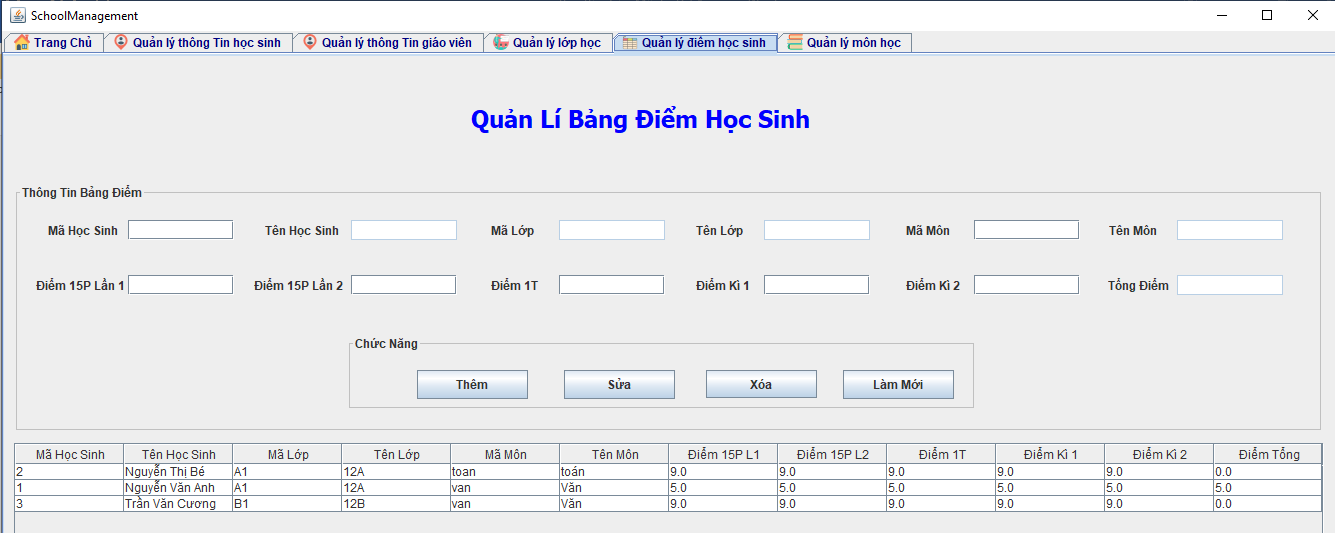


+ Quản lý giáo viên: 

+ Quản lý lớp học :



+ Quản Lý Điểm:



+ Quản Lý Môn Học:



\* Ngoài ra em còn có phân quyền cho hệ thống:

- Có 3 phân quyền cơ bản cho hệ thống là :

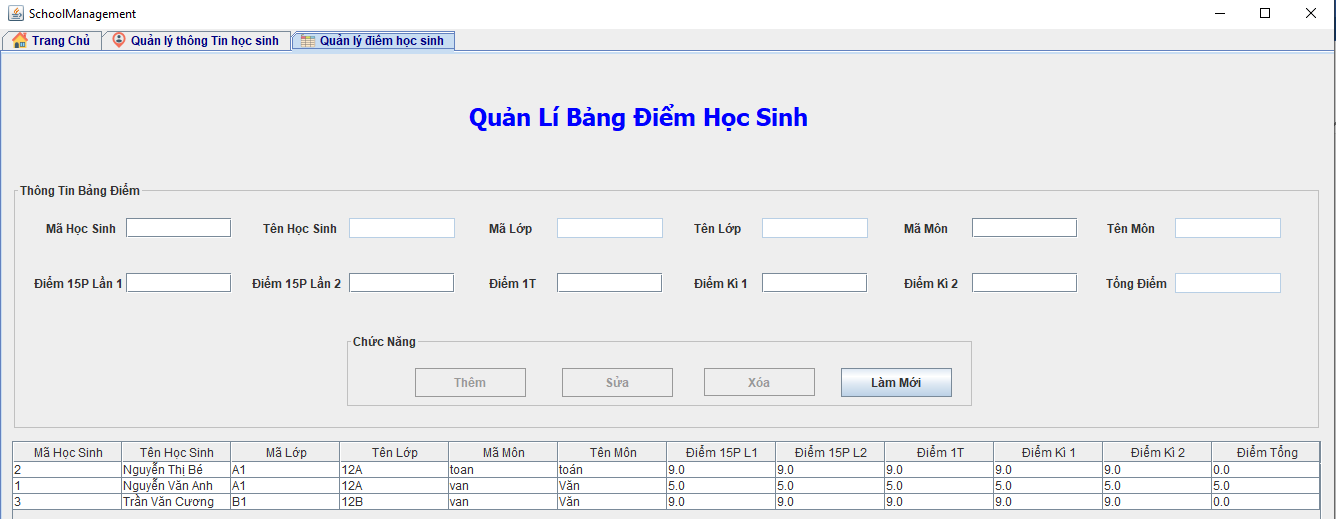
+ Giáo vụ của nhà trường sẽ nhập toàn quyền.



+ Giáo viên của nhà trường sẽ được phép xem thông tin của các học sinh, thông tin của các giáo viên, thông tin lớp học và them, sửa, xóa điểm.



+ Học sinh sẽ được quyền xem điểm và xem thông tin của học sinh.



**Phần III : Ưu, nhược điểm của hệ thống và hướng phát triển**

1. Ưu và nhược điểm :

+ Ưu điểm :

- Hệ thống xây dựng gọn nhẹ.

- Hệ thống có sự ràng buộc về dữ liệu.

- Giao diện khá thân thiện với người dùng.

+ Nhược điểm :

* Cách tổ chức chưa chặt chẽ.
* Bắt lỗi còn hạn chế, chưa hoàn thiện
* Các chức năng còn đơn giản, chưa linh động.

1. Hướng phát triển:

* Khả năng xử lý các sự kiện, các lỗi ngoài ý muốn tốt hơn của chương trình và dùng thao tác lên chương trình.
* Hoàn thiện tốt hơn về lập trình Java core, thiết kế Desktop và các ràng buột dữ liệu.
* Nâng cao sự linh hoạt của chương trình.
* Nâng cao kỹ thuật, hoàn thiện các phần còn thiếu của chương trình.

***Bảng báo cáo của em đến đây là hết. Xin cảm ơn mọi người đã giúp đỡ và đọc bảng báo cáo này.***