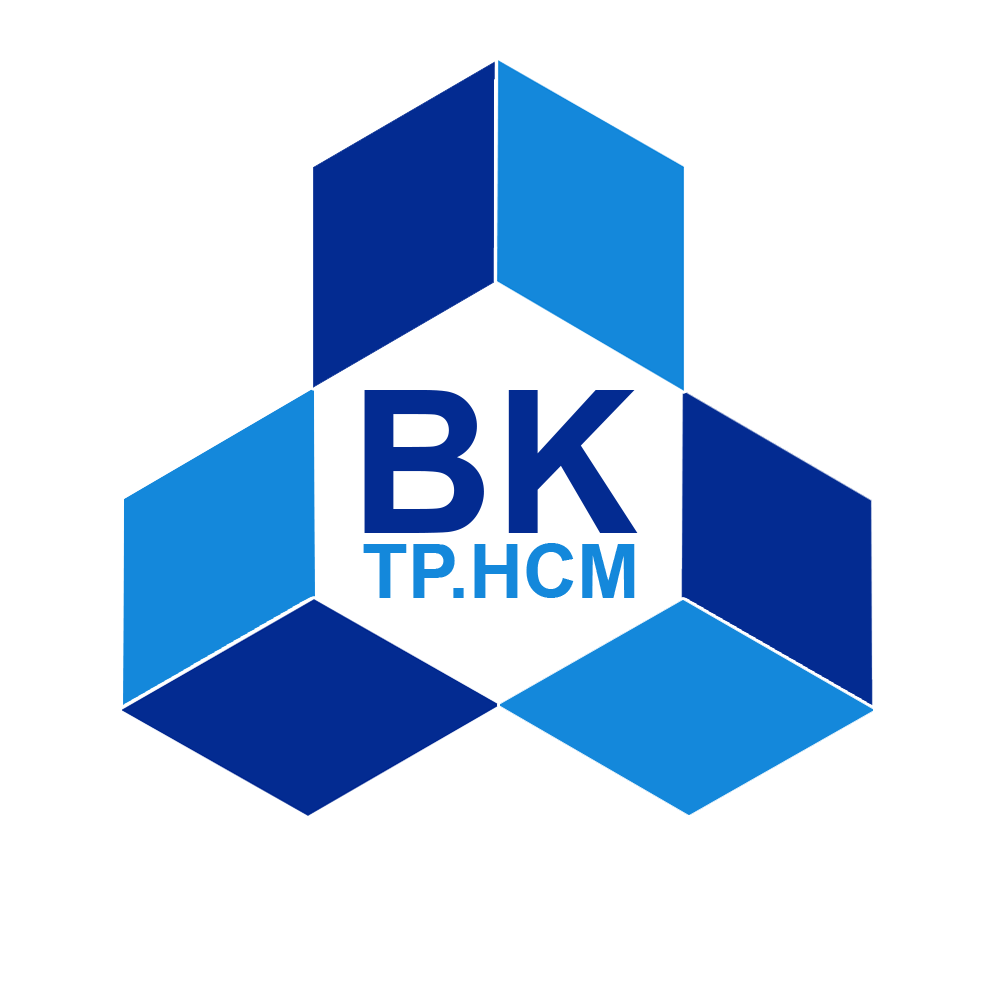
**ĐẠI HỌC QUỐC GIA TP.HCM**

**TRƯỜNG ĐẠI HỌC BÁCH KHOA**

**KHOA KHOA HỌC & KỸ THUẬT MÁY TÍNH**



**LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC**

**Xây dựng hệ thống đăng tin môi giới bất động sản**

**HỘI ĐỒNG: HỘI ĐỒNG 1 - HỆ THỐNG THÔNG TIN**

**GVHD: Trần Minh Quang**

**GVPB: Trần Thị Quế Nguyệt**

**---o0o---**

**SVTH: Nguyễn Ngọc Duy (51300595)**

TP. HỒ CHÍ MINH, THÁNG 05 NĂM 2018

# I. Lời cam đoan

Em xin cam đoan đây là công trình nghiên cứu và thực hiện bởi riêng em dưới sự hướng dẫn của giảng viên hướng dẫn. Nếu không đúng như đã nêu trên, em xin hoàn toàn chịu trách nhiệm về đề tài của mình.

# II. Lời cảm ơn

Em xin chân thành cảm ơn các thầy cô trường Đại học Bách Khoa TP.HCM nói chung, cũng như các thầy cô khoa Khoa học và Kỹ thuật Máy tính nói riêng, với những sự hỗ trợ nhiệt tình cũng như những bài học bổ ích về cả kiến thức chuyên ngành lẫn cho cuộc sống của em.

Trong quá trình thực hiện đề tài, do kỹ năng kinh nghiệm thực tế chưa nhiều nên không thể tránh khỏi những thiếu sót, em mong được quý thầy cô góp ý cho em để em có thể học tập được thêm nhiều kiến thức hay.

Cuối cùng, em xin kính chúc quý thầy cô dồi dào sức khỏe và ngày một thành công trong công việc, giúp cho các thế hệ tiếp theo được dìu dắt để có thể đóng góp được nhiều hơn cho gia đình và xã hội.

# III. Tóm tắt luận văn

Đề tài yêu cầu xây dựng một hệ thống đăng tin môi giới bất động sản, đáp ứng được các chức năng cần có của một hệ thống đăng tin bất môi giới động sản, đồng thời cung cấp các chức năng mới, hữu ích hơn, giúp người sử dụng được tiện lợi hơn.

Đề tài được thực hiện xây dựng giải pháp cho một hệ thống đăng tin môi giới bất động sản, hoạt động trên đa nền tảng, cung cấp các chức năng cơ bản của một hệ thống đăng tin môi giới bất động sản, cũng như cung cấp chức năng tìm kiếm thông minh qua hình ảnh, giúp người tìm tin đăng được chủ động được với những thông tin cần tìm kiếm. Giải pháp được xây dựng hướng đến mục đích hoạt động với hiệu năng tốt hơn các hệ thống đã có sẵn, dễ dàng mở rộng, cũng như bám sát với nghiệp vụ nhà đất của thị trường.

Mục lục

[I. Lời cam đoan 2](#_Toc515388414)

[II. Lời cảm ơn 2](#_Toc515388415)

[III. Tóm tắt luận văn 3](#_Toc515388416)

[IV. Danh mục hình 5](#_Toc515388417)

[V. Danh mục bảng 6](#_Toc515388418)

[VI. Nội dung chính 7](#_Toc515388419)

[VI.1. Đặt vấn đề 7](#_Toc515388420)

[VI.2. Các hệ thống hiện có trên thị trường 8](#_Toc515388421)

[VI.3. Giải pháp thực hiện 9](#_Toc515388422)

[VI.3. Công nghệ sử dụng 11](#_Toc515388423)

[VI.3.i. Ngôn ngữ PHP và SQL 11](#_Toc515388424)

[VI.3.ii. Ngôn ngữ Javascript 12](#_Toc515388425)

[VI.3.iii. Google Chart plugin 13](#_Toc515388426)

[VI.3.iv. Mô hình MVC 14](#_Toc515388427)

[VI.4. Quy trình thực hiện 16](#_Toc515388428)

[VI.4.i. Thiết kế cơ sở dữ liệu 16](#_Toc515388429)

[VI.4.ii. Thiết kế giao diện 18](#_Toc515388430)

[VI.4.iii. Viết chương trình 24](#_Toc515388431)

[VI.5. Các chức năng hỗ trợ 29](#_Toc515388432)

[VI.5.i. Cho phép sinh viên xem điểm các môn học 29](#_Toc515388433)

[VI.5.ii. Cho phép giảng viên xem và quản lý môn học của mình 29](#_Toc515388434)

[VI.5.iii. Cho phép quản trị viên quản lý thông tin thành viên 34](#_Toc515388435)

[VII. Danh mục tài liệu tham khảo 35](#_Toc515388436)

# IV. Danh mục hình

[Hình 1 Mô hình quy trình nghiệp vụ hệ thống môi giới bất động sản 8](#_Toc515388366)

[Hình 2 Tương tác giữa các thành phần thuộc hệ thống 10](#_Toc515388367)

[Hình 3 Luồng xử lý dữ liệu tại server 10](#_Toc515388368)

[Hình 4 Ví dụ về ngôn ngữ PHP 11](#_Toc515388369)

[Hình 5 Ví dụ về biến SESSION trong PHP 11](#_Toc515388370)

[Hình 6 Ví dụ về ngôn ngữ SQL 12](#_Toc515388371)

[Hình 7 Ví dụ sử dụng kết hợp ngôn ngữ SQL trong PHP 12](#_Toc515388372)

[Hình 8 Ví dụ về mã Javascript 13](#_Toc515388373)

[Hình 9 Ví dụ đoạn mã vẽ đồ thị bằng Google Chart 13](#_Toc515388374)

[Hình 10 Kết quả thực thi đoạn mã vẽ đồ thi bằng Google Chart 14](#_Toc515388375)

[Hình 11 Biểu diễn mô hình MVC 15](file:///D:\03_Source\DatToNhaDep\01_Documents\BaoCaoLVTN_28052018.docx#_Toc515388376)

[Hình 12 Lược đồ quan hệ 16](#_Toc515388377)

[Hình 13 Thiết kế màn hình chính 18](#_Toc515388378)

[Hình 14 Thiết kế màn hình đăng nhập thành công 19](#_Toc515388379)

[Hình 15 Thiết kế màn hình đăng nhập thất bại 19](#_Toc515388380)

[Hình 16 Thiết kế màn hình sinh viên xem bảng điểm cá nhân 20](#_Toc515388381)

[Hình 17 Thiết kế màn hình giảng viên xem các môn học mình quản lý 20](#_Toc515388382)

[Hình 18 Thiết kế màn hình giảng viên tạo môn học mới 21](#_Toc515388383)

[Hình 19 Thiết kế màn hình giảng viên xem bảng điểm 21](#_Toc515388384)

[Hình 20 Thiết kế màn hình giảng viên nhập điểm 22](#_Toc515388385)

[Hình 21 Thiết kế màn hình giảng viên tạo / sửa các thành phần điểm của môn học 22](#_Toc515388386)

[Hình 22 Thiết kế màn hình quản trị viên xem thông tin các thành viên 23](#_Toc515388387)

[Hình 23 Thiết kế màn hình quản trị viên thêm thành viên mới 23](#_Toc515388388)

[Hình 24 Thiết kế màn hình quản trị viên chỉnh sửa thông tin thành viên 24](#_Toc515388389)

[Hình 25 Tổ chức cấu trúc cho mô hình MVC 25](#_Toc515388390)

[Hình 26 Khởi tạo các trình tải config, library, view và model 25](#_Toc515388391)

[Hình 27 Tự động tải các controller và action cần thiết 26](#_Toc515388392)

[Hình 28 Mô tả cấu trúc của Controller và Action 26](#_Toc515388393)

[Hình 29 Thiết lập các hàm cơ bản trong lớp model 26](#_Toc515388394)

[Hình 30 Ví dụ một hàm phức tạp trong lớp model 27](#_Toc515388395)

[Hình 31 Ví dụ tải dữ liệu sang lớp view để lấy thông tin hiển thị 27](#_Toc515388396)

[Hình 32 Ví dụ sử dụng dữ liệu nhận được để tạo nội dung hiển thị 27](#_Toc515388397)

[Hình 33 Ví dụ lấy dữ liệu tại lớp view và hiển thị ra cho người dùng 28](#_Toc515388398)

[Hình 34 Ví dụ thiết lập chức năng về chạy tại máy người dùng 28](#_Toc515388399)

[Hình 35 Mô tả chức năng xem điểm đa thành phần 29](#_Toc515388400)

[Hình 36 Mô tả chức năng xem danh sách các môn học 29](#_Toc515388401)

[Hình 37 Mô tả chức năng xem điểm của môn học đang quản lý 30](#_Toc515388402)

[Hình 38 Mô tả chức năng hiển thị thống kê 30](#_Toc515388403)

[Hình 39 Mô tả chức năng tải bảng điểm về máy tính giảng viên 31](#_Toc515388404)

[Hình 40 Mô tả chức năng tạo môn học mới với các thông tin môn học 31](#_Toc515388405)

[Hình 41 Mô tả chức năng tạo điểm thành phần liên hệ với outcome môn học 32](#_Toc515388406)

[Hình 42 Mô tả việc khai báo công thức tính điểm tổng kết 32](#_Toc515388407)

[Hình 43 Mô tả chức năng nhập điểm tại trang web 32](#_Toc515388408)

[Hình 44 Mô tả bảng nhập điểm mà giảng viên tải về từ hệ thống 33](#_Toc515388409)

[Hình 45 Mô tả kết quả sau khi chạy tính năng Tải lên tệp điểm 33](#_Toc515388410)

[Hình 46 Mô tả kết quả của quá trình nhập điểm qua tệp tin 33](#_Toc515388411)

[Hình 47 Mô tả bảng thông tin thành viên 34](#_Toc515388412)

[Hình 48 Mô tả chức năng tạo/chỉnh sửa thông tin thành viên 34](#_Toc515388413)

# V. Danh mục bảng

[Bảng 1 So sánh công năng các hệ thống đăng tin môi giới bất động sản nổi tiếng 9](#_Toc515388354)

[Bảng 2 Thiết kế cơ sở dữ liệu từ lược đồ quan hệ 18](#_Toc515388355)

# VI. Nội dung chính

## VI.1. Đặt vấn đề

Từ yêu cầu của đề tài, chương trình cần đáp ứng được các yêu cầu có thể được liệt kê chi tiết ra như sau:

1. Người môi giới:
   1. Môi giới cá nhân:
      1. Đăng thông tin BĐS muốn bán/cho thuê/chia sẻ.
      2. Liên hệ được với những người có nhu cầu (email, điện thoại, hệ thống nhắn tin được cung cấp).
      3. Chọn mức độ ưu tiên bài viết, dựa trên khoản chi phí bỏ ra được cho bài viết (hoặc chọn miễn phí cho mức ưu tiên thấp nhất).

Chi phí: Chi trả cho hệ thống để bài viết được ưu tiên.

Thu nhập: Được chi trả trực tiếp từ người mua (hoặc người gửi bán) khi giao dịch thành công.

* 1. Môi giới doanh nghiệp:
     1. Đăng thông tin BĐS muốn bán/cho thuê/chia sẻ.
     2. Đăng thông tin dự án, thông tin tham quan dự án.
     3. Liên hệ được với những người có nhu cầu (email, điện thoại, hệ thống nhắn tin được cung cấp).
     4. Chọn mức độ ưu tiên bài viết, dựa trên khoản chi phí bỏ ra được cho bài viết (hoặc chọn miễn phí cho mức ưu tiên thấp nhất).

Chi phí: Chi trả cho hệ thống để bài viết được ưu tiên.

Thu nhập: Được chi trả trực tiếp từ người mua (hoặc người gửi bán) khi giao dịch thành công.

1. Khách:
   * 1. Xem/Tìm kiếm các BĐS, dự án, người môi giới mà họ quan tâm.
     2. Tìm kiếm theo loại BĐS, vị trí hành chính, vị trí thực địa (map location), khoảng giá, khoảng diện tích, số tầng, hướng, chiều rộng đường, dự án.
     3. Liên lạc với người môi giới theo dự án, theo danh sách bạn bè đã liên kết sẵn qua điện thoại, email, hoặc công cụ hỗ trợ của hệ thống.

Chi phí: Chi phí mua BĐS, đưa cho bên bán.

Thu nhập: Không.

1. Điều hành viên:
   * 1. Quản lý các thành viên trong hệ thống.
     2. Quản lý (Xem, duyệt, gia hạn, chỉnh sửa, xóa) các bài đăng.
     3. Quản lý (Xem, tìm kiếm, tạo, sửa, xóa) các giao dịch nạp tiền vào hệ thống.
     4. Trả lời các liên hệ của thành viên thông qua công cụ do hệ thống cung cấp.

Chi phí: Không có.

Thu nhập: Tiền lương từ công ty quản lý công cụ.

1. Công ty phát hành, quản lý công cụ:

Chi phí: Lương cho nhân viên điều hành, chi phí marketing.

Thu nhập: Tiền nạp vào từ các thành viên, thu nhập từ quảng cáo.

Quy trình nghiệp vụ của hệ thống cần được đáp ứng có thể được biểu diễn như sau:

Khách

Môi giới

Điều hành viên

Truy cập

Đăng nhập

Tìm kiếm BĐS

Xem chi tiết BĐS

Liên hệ người đăng

Nạp tiền vào tài khoản

Đăng tin BĐS

Xem các bài cần duyệt

Duyệt bài

Xem các giao dịch nạp tiền

Cập nhật thông tin giao dịch

Hình 1 Mô hình quy trình nghiệp vụ hệ thống môi giới bất động sản

## VI.2. Các hệ thống hiện có trên thị trường

Qua nghiên cứu, tìm hiểu, trên thị trường hiện đang có các hệ thống đăng tin môi giới bất động sản lớn đang hoạt động. Trong đó có ba trang thông tin / hệ thống đăng tin môi giới bất động sản tiêu biểu như sau, đi kèm với các thông tin chi tiết của từng hệ thống:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Tên chức năng** | Image result for batdongsan.com.vn logo   |  | | --- | |  | | Image result for alonhadat   |  | | --- | |  | | Image result for muabannhadat   |  | | --- | |  | |
| 1 | Các tính năng cơ bản (Xem, Tìm kiếm, Lọc tin, Đăng Tin,…) | Có | Có | Có |
| 2 | Hỗ trợ đăng tin thường (Miễn phí) | Không | Có | Có |
| 3 | Hỗ trợ đăng tin VIP (Nạp tiền) | Có | Có | Có |
| 4 | Có bản đồ trong tin đăng | Có | Có | Có |
| 5 | Tin đăng có hình ảnh | Có | Có | Có |
| 6 | Tin đăng có hình ảnh 3D | Có | Không | Không |
| 7 | Hỗ trợ chụp hình ảnh 3D | Không | Không | Không |
| 8 | Hỗ trợ đăng tin dự án | Có | Có | Không |
| 9 | Hỗ trợ đăng tin phòng trọ | Có | Có | Không |
| 10 | Hỗ trợ liên hệ trợ giúp trực tuyến | Có | Không | Không |
| 11 | Chia sẻ thông tin lên mạng xã hội | Có | Không | Có |
| 12 | Nhiễu/Mập mờ thông tin | Có | Có | Có |
| 13 | Phương pháp thanh toán linh hoạt (Thẻ cào, Chuyển khoản, Tại chi nhánh,…) | Có | Có | Có |
| 14 | Đội ngũ nhân viên hỗ trợ sẵn sàng liên lạc qua điện thoại | Có | Không | Không |
| 15 | Tổng đài hỗ trợ khách hàng | Có | Có | Có |
| 16 | Cho phép bình luận tại tin đăng | Không | Không | Không |
| 17 | Cho phép phản ánh tin đăng | Có | Có | Không |
| 18 | Gợi ý các tin liên quan | Có | Có | Có |
| 19 | Tư vấn nhà/đất theo phong thủy | Không | Không | Có |
| 20 | Hỗ trợ tìm kiếm nhà môi giới | Có | Có | Không |

Bảng So sánh công năng các hệ thống đăng tin môi giới bất động sản nổi tiếng

## VI.3. Giải pháp thực hiện

Với yêu cầu được phân tích tại mục VI.1 và VI.2, hệ thống cần được thực hiện bao gồm các thành phần sau:

1. Trung tâm lưu trữ - truy xuất thông tin: Là một máy chủ có nhiệm vụ lưu trữ và xử lý các yêu cầu từ các thiết bị, nền tảng khác nhau.
2. Trang web quản lý: Là công cụ dành cho nghiệp vụ quản lý, phục vụ các thao tác quản lý thành viên, bài viết, tài khoản, cũng như hệ thống.
3. Phần mềm di động: Là công cụ dành cho người dung, phục vụ các thao tác xem, tìm kiếm và đăng các thông tin về môi giới bất động sản. Ngoài ra người dùng cũng có quyền (có giới hạn) đối với thông tin cá nhân tài khoản.

Mô hình liên kết giữa các thành phần này được biểu diễn như sau:



Mobile App



Web Client



Server

Request

JSON Result

Request

Web interface/JSON Result

Hình 2 Tương tác giữa các thành phần thuộc hệ thống

Kiểu dữ liệu được lựa chọn để truyền tải giữa các thành phần thuộc hệ thống là dưới dạng JSON, với phân tích như sau:

Ưu điểm:

1. Lưu trữ tạm data (JSON) trên local giúp phần mềm hoạt động được ngay cả khi có sự cố về mạng
2. Giúp đồng bộ phương thức trao đổi giữa phần mềm di động và trang web với máy server
3. Thời gian/Dung lượng tải nội dung cập nhật được giảm bớt so với việc tải lại toàn bộ nội dung website (Đối với website)

Nhược điểm:

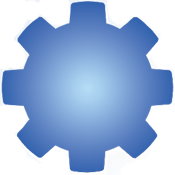
1. Giải pháp có thể khiến thiết bị quá tải khi phải xử lý cục bộ lượng lớn dữ liệu
2. Khi có sự cập nhật về cấu trúc dữ liệu, chương trình di động cần phải được cập nhật lại

Luồng xử lý dữ liệu được thực hiện theo mô hình sau:



Web Server

Database



Controller

Website request

Get data for index page

Data request

Get data for requested

Webpage

Data for index page

Website request

Data and Webpage

Data request

Data requested

JSON data

JSON data

Hình 3 Luồng xử lý dữ liệu tại server

## VI.3. Công nghệ sử dụng

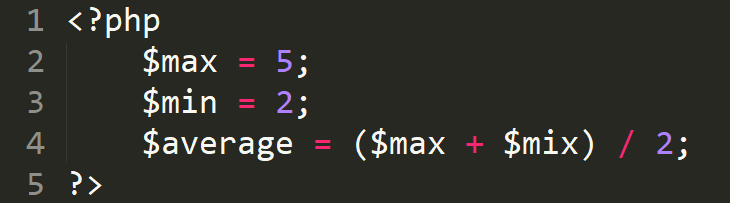
### VI.3.i. Ngôn ngữ PHP và SQL

Ngôn ngữ PHP (viết tắt hồi quy "PHP: Hypertext Preprocessor") là ngôn ngữ rất phổ biến, thường được lựa chọn để phát triển các ứng dụng web. PHP là một ngôn ngữ lập trình kịch bản được chạy ở phía server nhằm sinh ra mã HTML trên client. PHP đã trải qua rất nhiều phiên bản và được tối ưu hóa cho các ứng dụng web, với cách viết mã rõ rãng, tốc độ nhanh, dễ học. PHP chạy trên môi trường web server và lưu trữ dữ liệu thông qua hệ quản trị cơ sở dữ liệu nên PHP thường đi kèm với Apache, MySQL và hệ điều hành Linux.

*(Trích: Wikipedia - PHP)*

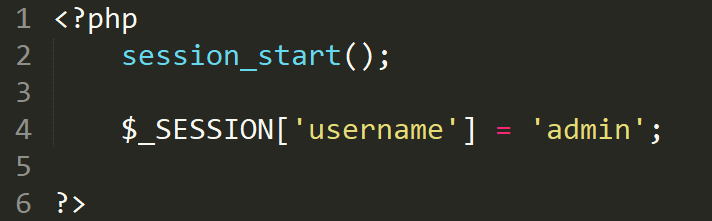
Mã PHP được nhận dạng dễ dàng vì được đặt giữa cặp thẻ: “<?php” và “?>”.

Ví dụ về ngôn ngữ PHP:



Hình 4 Ví dụ về ngôn ngữ PHP

Để quản lý phiên làm việc, PHP cung cấp các hàm chức năng và các biến truy cập để khởi tạo cũng như thao tác trên phiên làm việc:



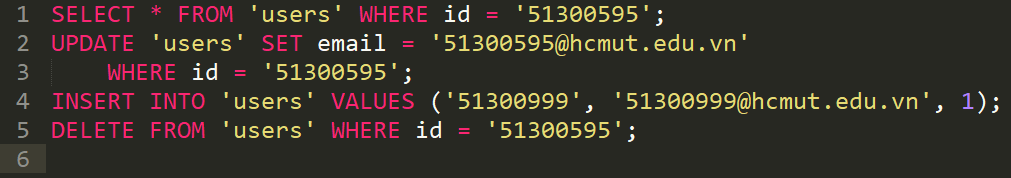
Hình 5 Ví dụ về biến SESSION trong PHP

Với đoạn như trên, thì trong cùng một tên miền, khi biến $\_SESSION['username'] được gọi đến thì nó đã có sẵn giá trị là 'admin', đồng thời ta cũng có thể chỉnh sửa lại giá trị của biến này tại một nơi khác cùng tên miền.

SQL (Structured Query Language) hay ngôn ngữ truy vấn mang tính cấu trúc, là một loại ngôn ngữ máy tính phổ biến để tạo, sửa, và lấy dữ liệu từ một hệ quản trị cơ sở dữ liệu quan hệ. Ngôn ngữ này phát triển vượt xa so với mục đích ban đầu là để phục vụ các hệ quản trị cơ sở dữ liệu đối tượng-quan hệ.

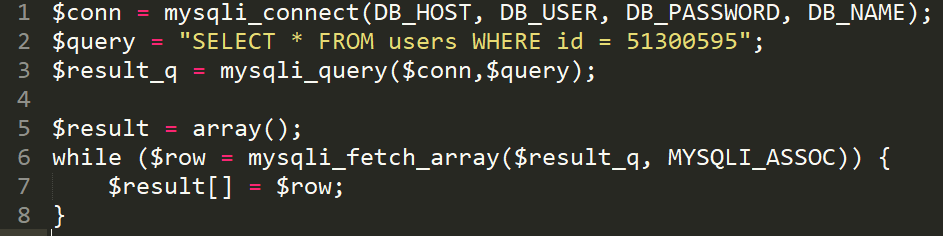
*(Trích: Wikipedia - SQL)*

Câu lệnh truy vấn SQL có thể đọc được, sửa được, thêm được và xóa được các dữ liệu trong cơ sở dữ liệu, ví dụ như sau:



Hình 6 Ví dụ về ngôn ngữ SQL

Để kết nối với cơ sở dữ liệu bằng SQL và thực hiện trên ngôn ngữ PHP, ta có thể thực hiện câu lệnh như sau:

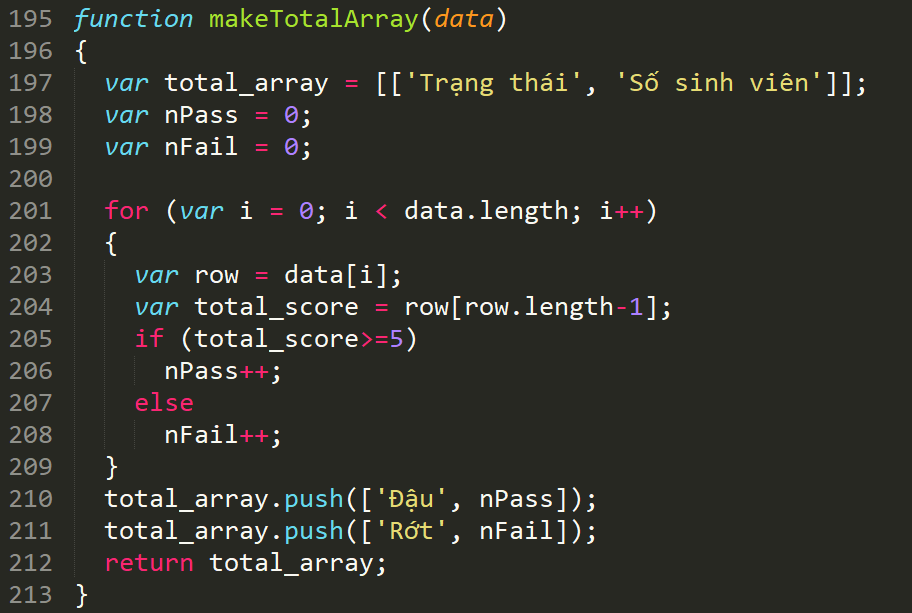


Hình 7 Ví dụ sử dụng kết hợp ngôn ngữ SQL trong PHP

### VI.3.ii. Ngôn ngữ Javascript

Ngôn ngữ Javascript là một ngôn ngữ lập trình kịch bản hướng đối tượng. Ngôn ngữ này được dùng rộng rãi cho các trang web, hỗ trợ thao tác trên các đối tượng của một trang web, giúp tạo hiệu ứng đa dạng và các chức năng phức tạp hơn cho trang web. Javascript thường được thực thi trên máy tính truy cập (client), và thường phụ thuộc vào trình duyệt được sử dụng để vào trang web.

Đoạn mã Javascript thường được đặt giữa cặp thẻ "<script>" và "</script>". Một ví cho mã Javascript như sau:



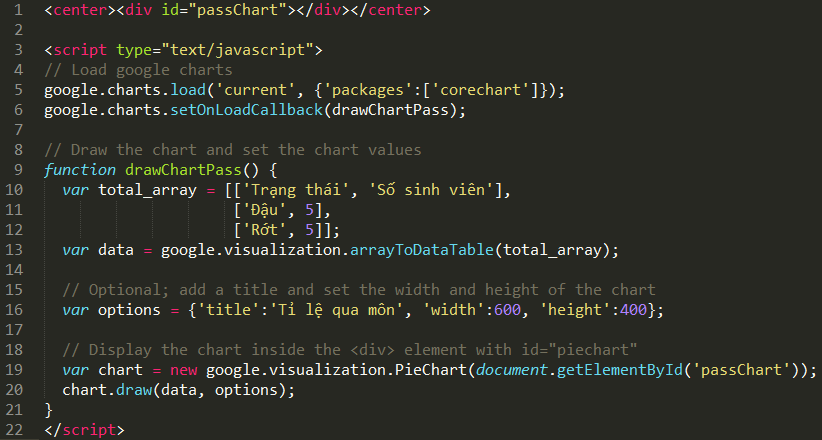
Hình 8 Ví dụ về mã Javascript

Trong phạm vi đề tài, Javascript có thể được sử dụng để đẩy việc tính điểm tổng kết, vẽ các đồ thị và thực hiện các thao tác phức tạp khác về cho máy tính người dùng, nhằm giảm tải cho máy chủ.

### VI.3.iii. Google Chart plugin

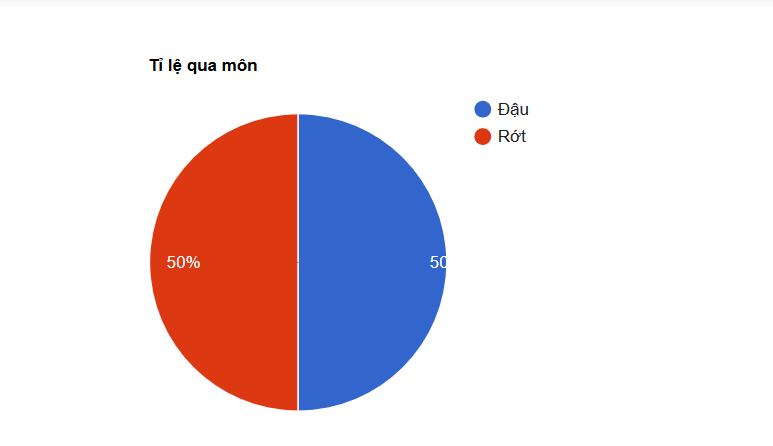
Google Chart là một thư viện jQuery, cung cấp các giao tiếp để người dùng có thể tạo ra các loại đồ thị bằng các lệnh javascript. Ưu điểm của việc sử dụng Google Chart là bộ thư viện rất gọn nhẹ, xử lý nhanh và công sức bỏ ra cho lập trình khá thấp.

Ví dụ đoạn mã lệnh sau vẽ ra một đồ thị theo số liệu nhập vào:



Hình 9 Ví dụ đoạn mã vẽ đồ thị bằng Google Chart

Khi thực thi, ta được kết quả như sau:



Hình 10 Kết quả thực thi đoạn mã vẽ đồ thi bằng Google Chart

### VI.3.iv. Mô hình MVC

MVC là viết tắt của đầy đủ ba thành phần: Model, View, Controller. Mô hình MVC là một mô hình phát triển phần mềm theo kiến trúc ba lớp, theo đó nhiệm vụ của từng lớp được quy định như sau:

* Model: Lớp này thực hiện giao tiếp với cơ sở dữ liệu, và cung cấp tác vụ cho lớp Controller gọi đến nó.
* View: Lớp này là lớp giao tiếp với người dùng, thường để hiển thị màn hình làm việc và chuyển những dữ liệu đã được xử lý từ lớp Controller đến người dùng.
* Controller: Lớp này có nhiệm vụ xử lý dữ liệu được gửi đến, truy xuất dữ liệu từ lớp Model và trả kết quả cho lớp View.

Mô hình MVC cho phép tách biệt các chức năng hệ thống với nhau, giúp việc phát triển phần mềm trở nên rõ ràng, dễ hiểu, dễ tiếp cận, cũng như có thể nhanh chóng phát hiện, can tiệp khi có lỗi phát sinh. Khi muốn sửa chữa, thêm hay bớt ở lớp nào, thì sẽ ít làm phát sinh vấn đề với các lớp khác do đã có quy định giao diện làm việc (Interface) giữa các lớp.

Mô hình MVC có thể được biểu diễn như sau:

Hình Biểu diễn mô hình MVC

Database

Client

## VI.4. Quy trình thực hiện

### VI.4.i. Thiết kế cơ sở dữ liệu

Dựa vào phân tích yêu cầu như tại phần VI.1, ta có thể đưa ra các đối tượng chính bao gồm: Thành viên (user), Môn học (subject), Chuẩn đầu ra (outcome), Thành phần điểm (score element), và Điểm số (score). Từ đó ta có lược đồ quan hệ như sau:



Hình 12 Lược đồ quan hệ

Dựa vào lược đồ quan hệ như trên, ta xây dựng cơ sở dữ liệu như sau:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Bảng | Thuộc tính | Khóa | Kiểu dữ liệu | Chú thích |
| users | id | Khóa chính | VARCHAR |  |
| username |  | VARCHAR |  |
| password |  | VARCHAR |  |
| email |  | VARCHAR |  |
| fullname |  | VARCHAR |  |
| role |  | INT | 1: Sinh viên  2: Giảng viên  3: Quản trị viên |
| subjects | id | Khóa chính | VARCHAR |  |
| name |  | VARCHAR |  |
| description |  | TEXT |  |
| fomular |  | TEXT | Biểu diễn dạng công thức toán học cơ bản, là sự kết hợp của các ký tự (, ), +, -, \*, / và bộ tên biến dạng 'score'[0-9].  Ví dụ:  (score1\*4+score2\*6)/10 |
| outcomes | id | Khóa chính | INT | AUTOINCREASE |
| subject\_id | Khóa ngoại | VARCHAR | Khóa ngoại: subjects.id |
| description |  | TEXT |  |
| score\_element | id | Khóa chính | INT | AUTOINCREASE |
| subject\_id | Khóa ngoại | VARCHAR | Khóa ngoại: subjects.id |
| name |  | VARCHAR |  |
| teach | user\_id | Khóa chính  Khóa ngoại | VARCHAR | Khóa ngoại: users.id |
| subject\_id | Khóa chính  Khóa ngoại | VARCHAR | Khóa ngoại: subjects.id |
| semester | Khóa chính | VARCHAR |  |
| study | user\_id | Khóa chính  Khóa ngoại | VARCHAR | Khóa ngoại: users.id |
| subject\_id | Khóa chính  Khóa ngoại | VARCHAR | Khóa ngoại: subjects.id |
| semester | Khóa chính | VARCHAR |  |
| scores | id |  | INT | UNIQUE  AUTOINCREASE |
| user\_id | Khóa chính  Khóa ngoại | VARCHAR | Khóa ngoại: users.id |
| semester | Khóa chính | VARCHAR |  |
| score\_element\_id | Khóa chính  Khóa ngoại | INT | Khóa ngoại: score\_element.id |
| score |  | VARCHAR |  |
| outcomes\_of\_  score\_element | score\_element\_id | Khóa chính  Khóa ngoại | INT | Khóa ngoại: score\_element.id |
| outcome\_id | Khóa chính  Khóa ngoại | INT | Khóa ngoại: outcomes.id |

Bảng Thiết kế cơ sở dữ liệu từ lược đồ quan hệ

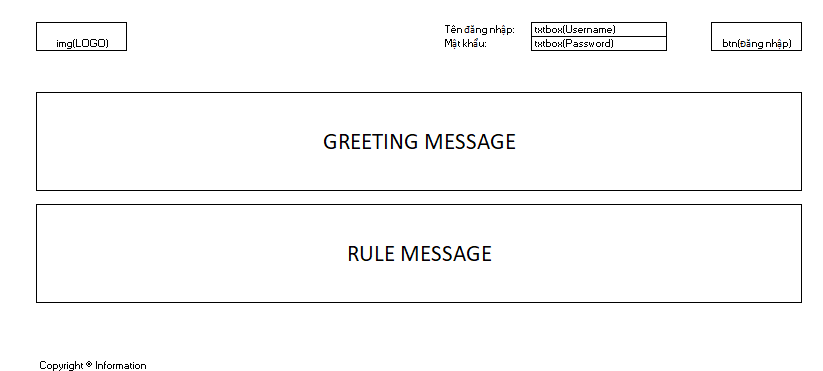
Nhận xét: Theo như cơ sở dữ liệu được thiết kế như trên:

* Đạt dạng chuẩn 1: Tất cả các thuộc tính của các quan hệ đều là thuộc tính đơn.
* Đạt dạng chuẩn 2: Đã đạt dạng chuẩn 1, và các thuộc tính không khóa của mỗi quan hệ đều phụ thuộc đầy đủ vào khóa chính của quan hệ đó.
* Không đạt dạng chuẩn 3: Đã đạt dạng chuẩn 2, tuy nhiên tại quan hệ tại bảng scores có thêm một thuộc tính ID dùng để xác định nhanh các phần tử điểm (score) để tránh truy xuất quá lâu và rườm rà.

Như vậy cơ sở dữ liệu được thiết kế đạt dạng chuẩn 2.

### VI.4.ii. Thiết kế giao diện

Giao diện trang web được thiết kế từng trang và có ghi chú kiểu cho từng đối tượng.



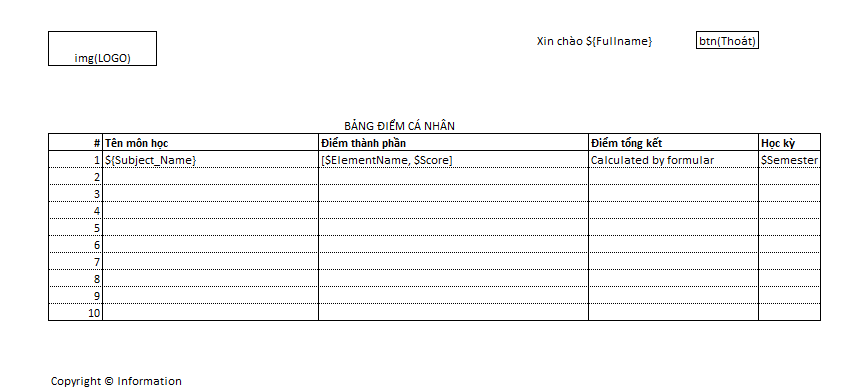
Hình 13 Thiết kế màn hình chính



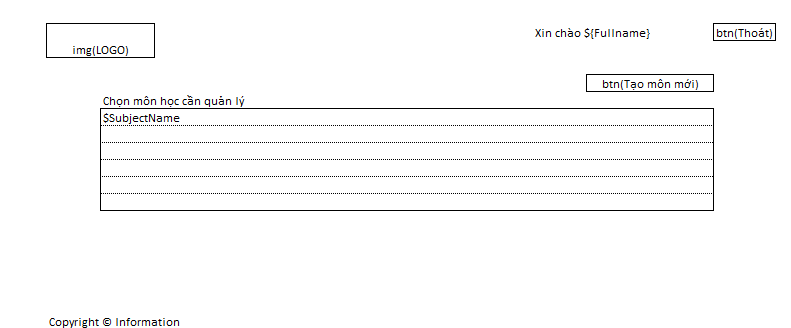
Hình 14 Thiết kế màn hình đăng nhập thành công



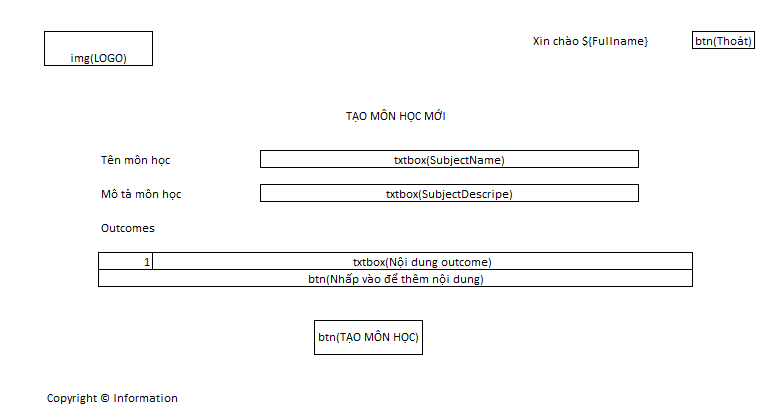
Hình 15 Thiết kế màn hình đăng nhập thất bại



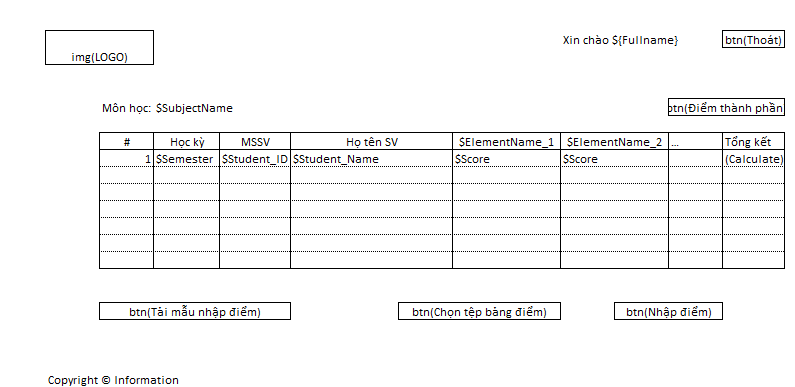
Hình 16 Thiết kế màn hình sinh viên xem bảng điểm cá nhân



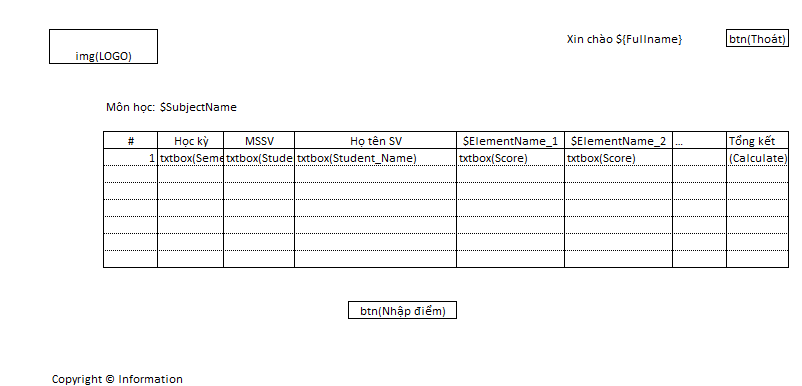
Hình 17 Thiết kế màn hình giảng viên xem các môn học mình quản lý



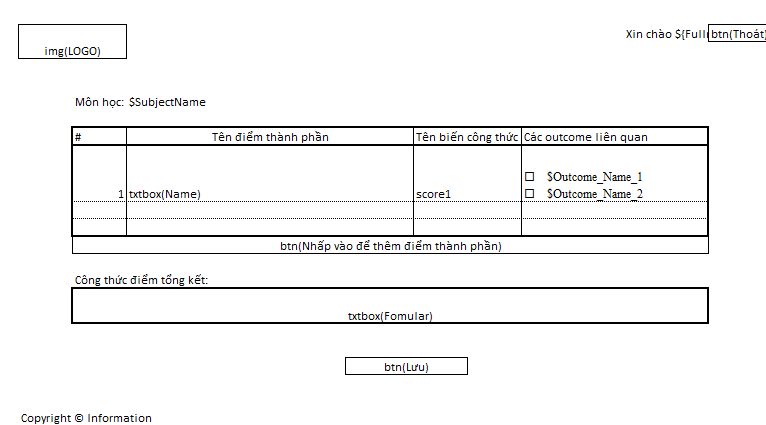
Hình 18 Thiết kế màn hình giảng viên tạo môn học mới



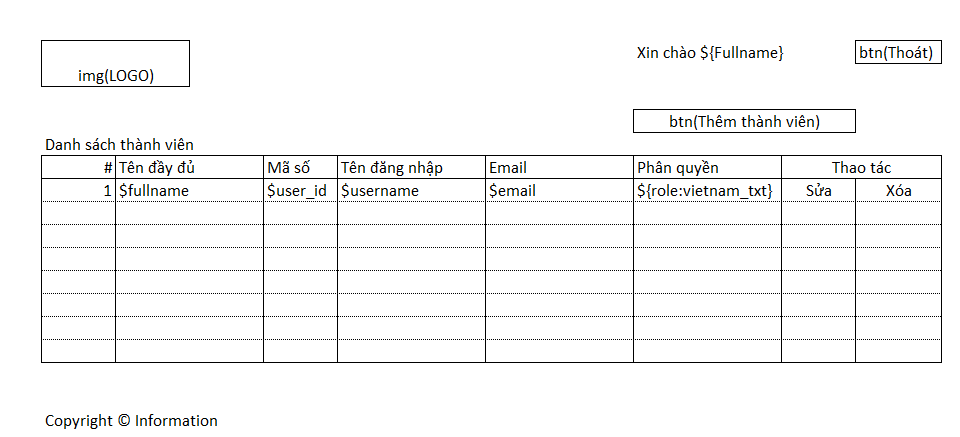
Hình 19 Thiết kế màn hình giảng viên xem bảng điểm



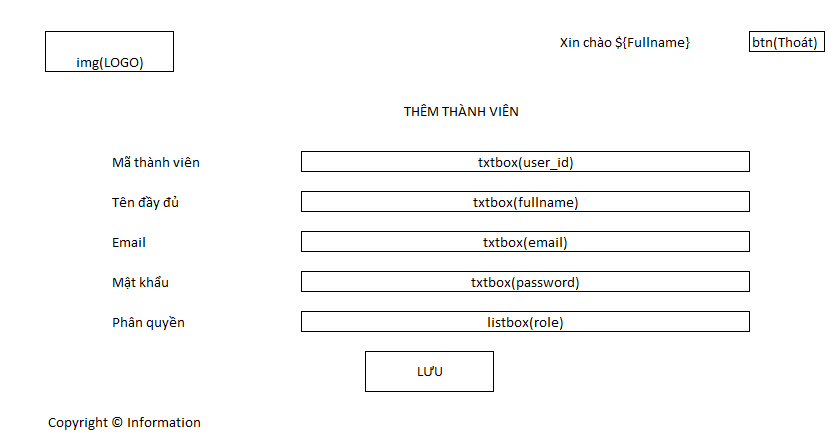
Hình 20 Thiết kế màn hình giảng viên nhập điểm



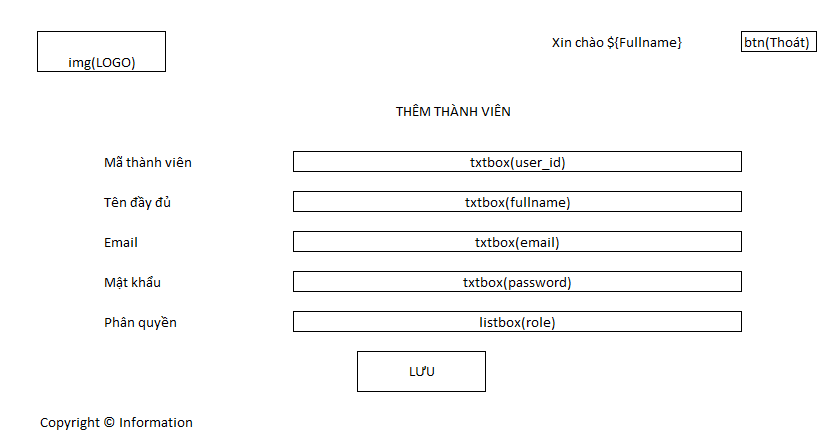
Hình 21 Thiết kế màn hình giảng viên tạo / sửa các thành phần điểm của môn học



Hình 22 Thiết kế màn hình quản trị viên xem thông tin các thành viên



Hình 23 Thiết kế màn hình quản trị viên thêm thành viên mới



Hình 24 Thiết kế màn hình quản trị viên chỉnh sửa thông tin thành viên

### VI.4.iii. Viết chương trình

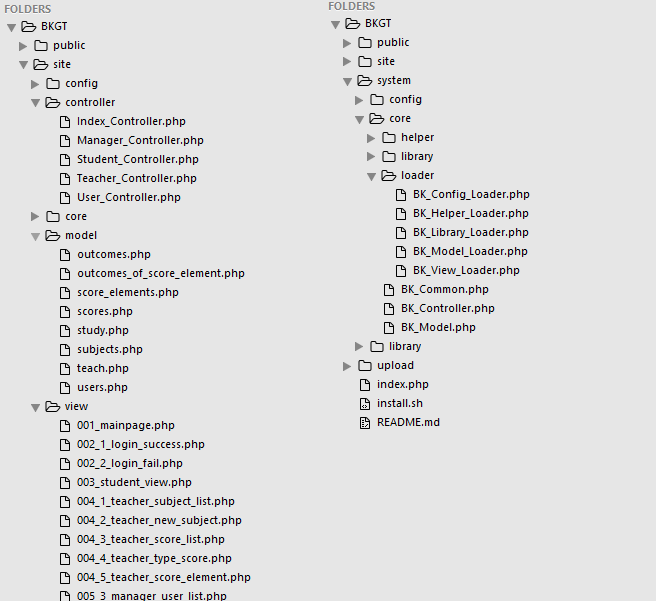
a. Thiết lập môi trường

Để thiết lập môi trường phát triển chương trình, ta cần một môi trường hỗ trợ ngôn ngữ PHP, một hệ quản trị cơ sở dữ liệu và một chương trình lập trình tốt.

XAMPP là một trong những công cụ mạnh mẽ cho phép thiết lập một môi trường web, hỗ trợ cả Apache, PHP, MySQL cũng như các hệ quản trị khác.

Để thực hiện việc viết chương trình, Sublime Text được chọn vì hỗ trợ tốt các thao tác lập trình từ cơ bản đến nâng cao; hỗ trợ nhận diện rất nhiều ngôn ngữ lập trình; thậm chí cho phép tạo chỉ mục cho phép truy vết (trace) nhanh chóng.

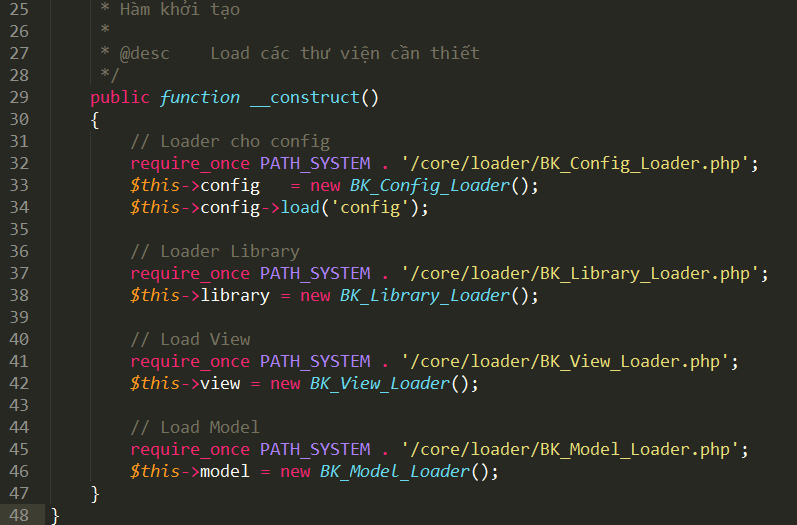
Mô hình MVC được thiết lập như sau:



Hình 25 Tổ chức cấu trúc cho mô hình MVC

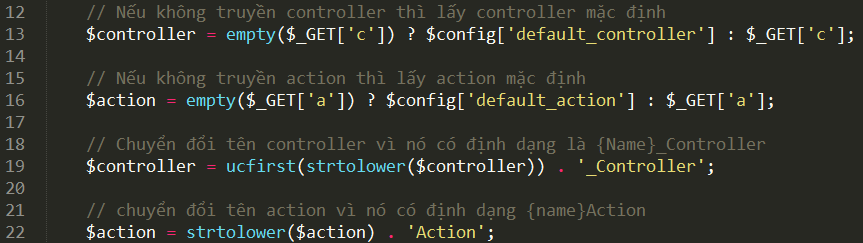
b. Tạo MVC framework

Đoạn mã khởi động sẽ khởi tạo các trình tải config, library, view và model mà chương trình sẽ sử dụng:



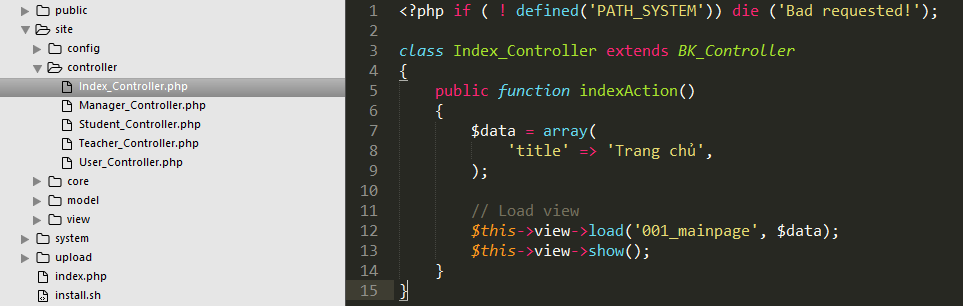
Hình 26 Khởi tạo các trình tải config, library, view và model

Để thiết thập cơ chế tự động tải cho các controller, ta có đoạn mã sau. Chương trình có thể tự động chọn controller và tự động chọn các action tương ứng với các tình huống khác nhau do người dùng yêu cầu đến:



Hình 27 Tự động tải các controller và action cần thiết

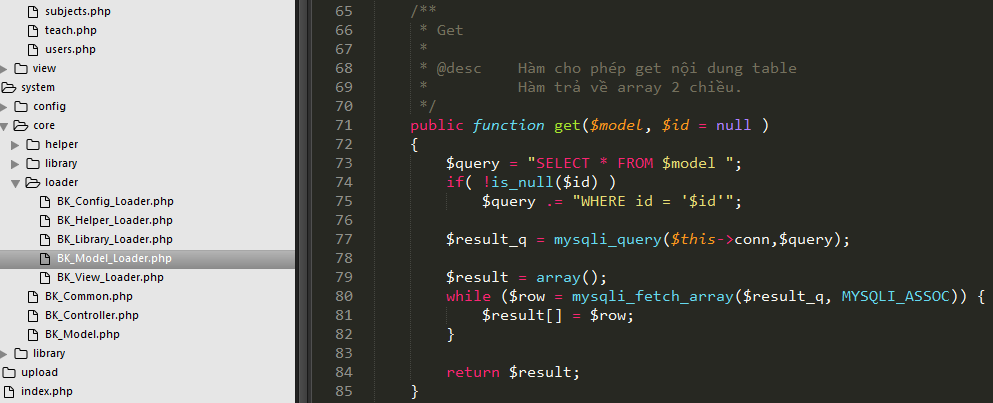
Các controller hỗ trợ các action khác nhau tùy vào thao trác mà người dùng gửi đến:



Hình 28 Mô tả cấu trúc của Controller và Action

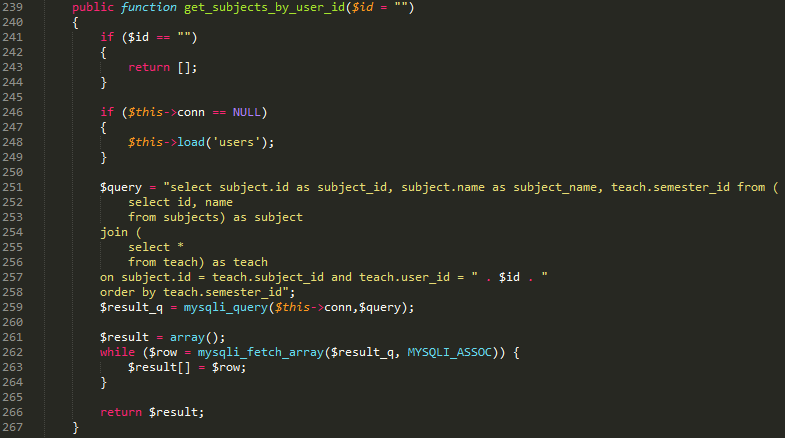
Như thế, khi người dùng truy cập đến địa chỉ: 'index.php?c=index&a=index' hoặc địa chỉ: 'index.php' (Index được chọn là nội dung mặc định nếu người dùng không cung cấp thông tin), controller Index sẽ lấy nội dung từ view '001\_mainpage' và gửi ngược lại cho người dùng.

Ta thiết lập các hàm mặc định trong tệp tải model để thực hiện các thao tác cơ bản như đọc bảng, thêm nội dung vào bảng, sửa bảng, xóa nội dung từ bảng.



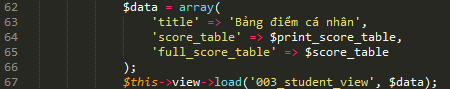
Hình 29 Thiết lập các hàm cơ bản trong lớp model

Hay thiết lập các hàm phức tạp hơn dùng trong các trường hợp riêng biệt cần thiết để tiết kiệm thời gian xử lý tại lớp controller. Ta có ví dụ về một hàm phức tạp trong lớp model như sau:



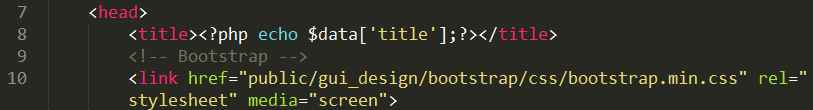
Hình 30 Ví dụ một hàm phức tạp trong lớp model

Mỗi khi xử lý xong dữ liệu cần thiết, controller sẽ gửi dữ liệu sang lớp view bằng hàm load như sau:



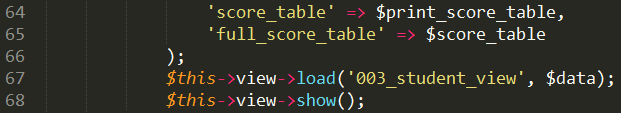
Hình 31 Ví dụ tải dữ liệu sang lớp view để lấy thông tin hiển thị

Tại lớp view, các biến này được sử dụng để hiện thị dữ liệu một cách hợp lý:



Hình 32 Ví dụ sử dụng dữ liệu nhận được để tạo nội dung hiển thị

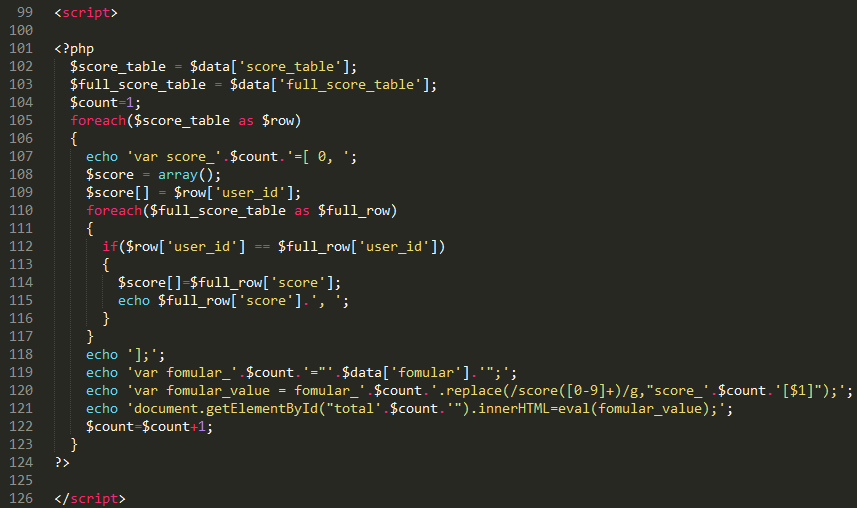
Sau đó, tại lớp controller, dữ liệu tạo ra tại lớp view sẽ được hiển thị ra cho người dùng. Ví dụ như sau:



Hình 33 Ví dụ lấy dữ liệu tại lớp view và hiển thị ra cho người dùng

c. Tạo các chức năng chạy tại máy người dùng

Chuyển việc tính toán điểm tổng kết, hay vẽ các đồ thị báo cáo, cũng như các tính năng nâng cao khác sang chạy tại máy người dùng, ta tiết kiệm được tài nguyên của máy chủ, giúp trang web có thể đáp ứng được nhiều yêu cầu cùng lúc hơn, tốc độ truy cập trở nên ổn định hơn. Ta có thể sử dụng javascript để thực hiện việc này, để xử lý và hiển thị các dữ liệu (các dữ liệu không mật).



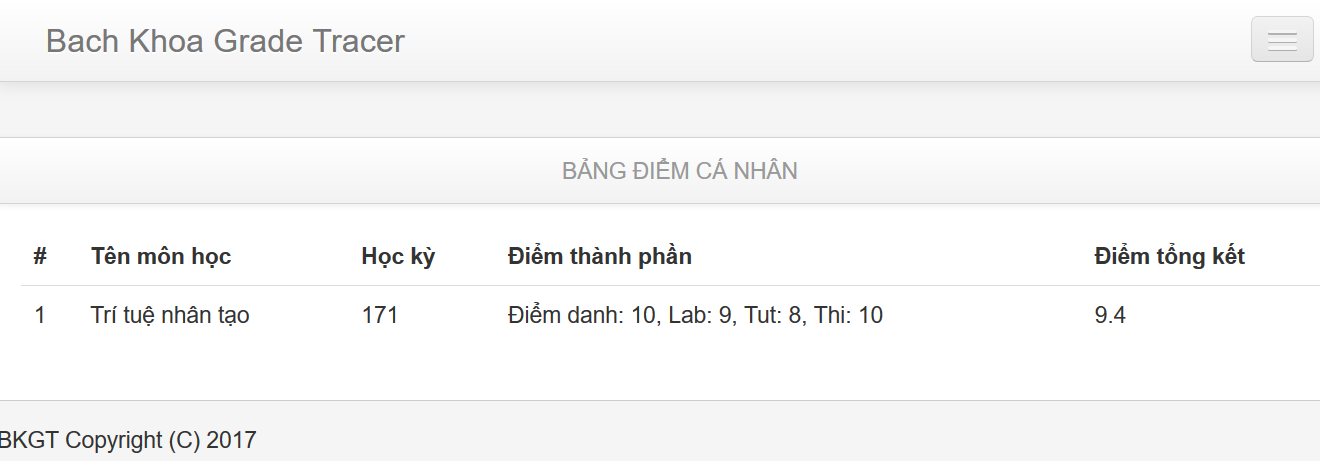
Hình 34 Ví dụ thiết lập chức năng về chạy tại máy người dùng

## VI.5. Các chức năng hỗ trợ

Qua các bước phát triển được sơ lược tại mục VI.4, sản phẩm thu được đã có được các chức năng cơ bản cũng như một số tính năng nâng cao của một trang quản lý điểm.

### VI.5.i. Cho phép sinh viên xem điểm các môn học

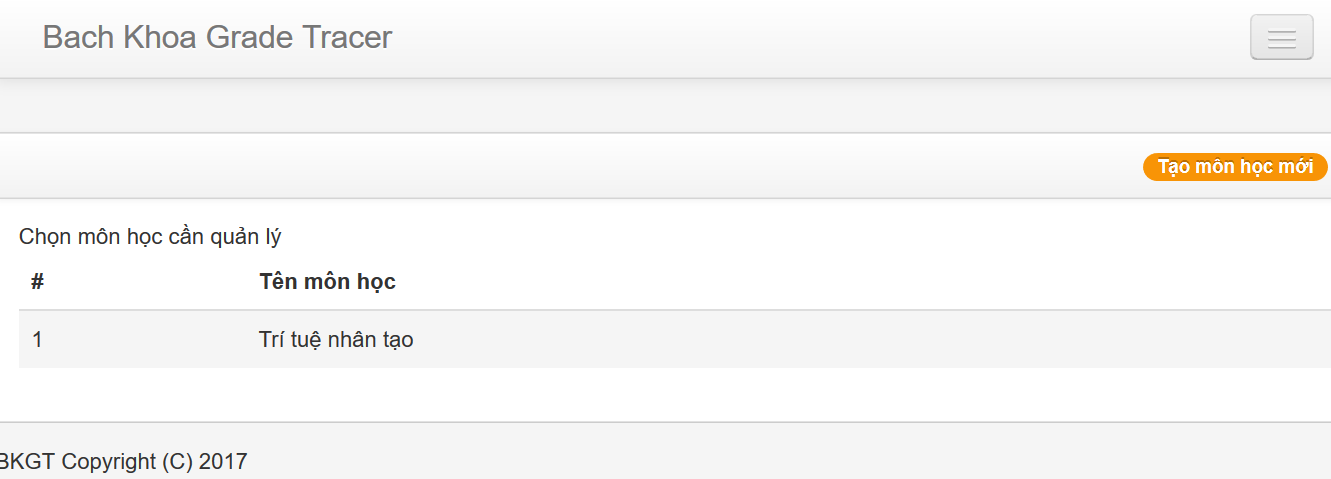
Công cụ cho phép người dùng có tài khoản sinh viên xem điểm các môn học mà sinh viên đó có điểm tại hệ thống. Điểm số mỗi môn có cấu trúc thành phần có thể giống hoặc khác nhau, phục vụ mục đích chấm điểm đa thành phần cho môn học.



Hình 35 Mô tả chức năng xem điểm đa thành phần

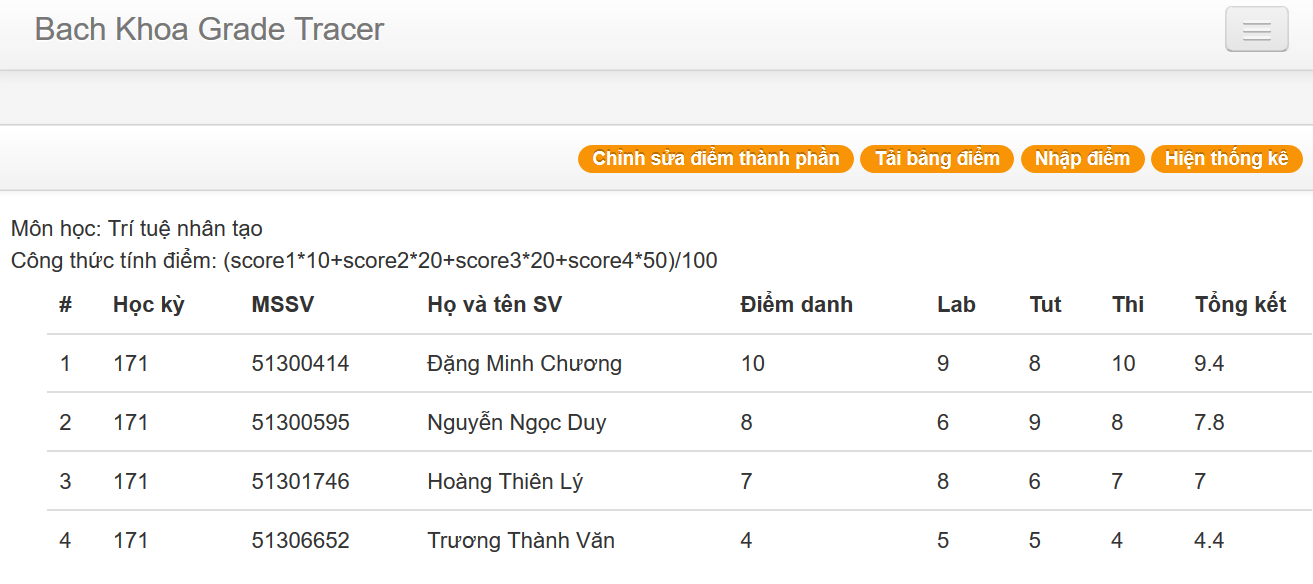
### VI.5.ii. Cho phép giảng viên xem và quản lý môn học của mình

Trang web liệt kê các môn học mà người dùng có tài khoản giảng viên tạo ra và quản lý:



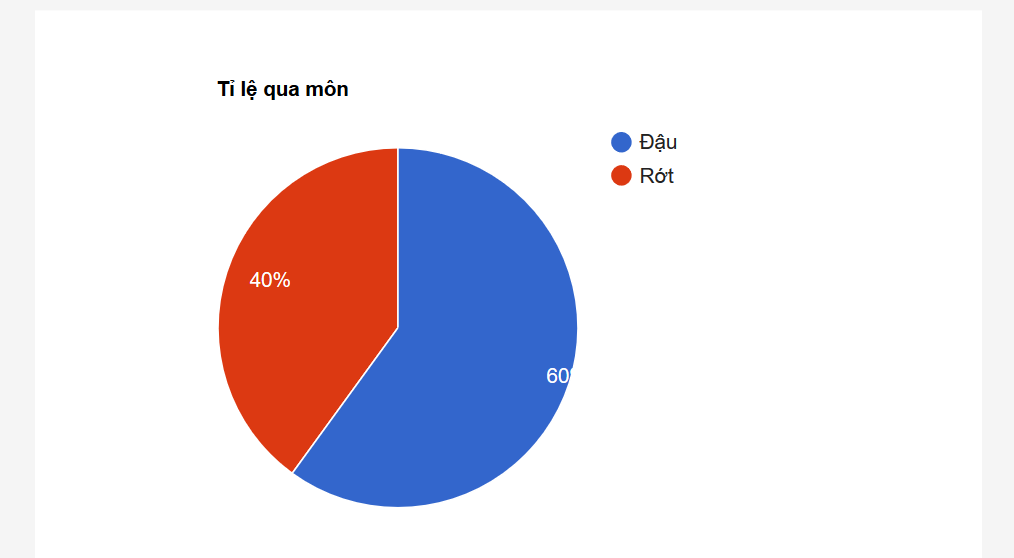
Hình 36 Mô tả chức năng xem danh sách các môn học

Khi chọn vào một môn học nhất định, trang web sẽ thể hiện danh sách các sinh viên cùng với thông tin điểm số tương ứng của sinh viên đó trong môn học đang chọn. Điểm tổng kết được tính toán tại máy tính đang truy cập trang web và công thức tính toán được hiển thị cho giảng viên biết.



Hình 37 Mô tả chức năng xem điểm của môn học đang quản lý

Với cái nhìn trực quan hơn về kết quả giảng dạy, giảng viên có thể nhấn vào nút "Hiện thốn kê", đồ thị mô tả số liệu dựa vào điểm tổng kết sẽ hiện ra.



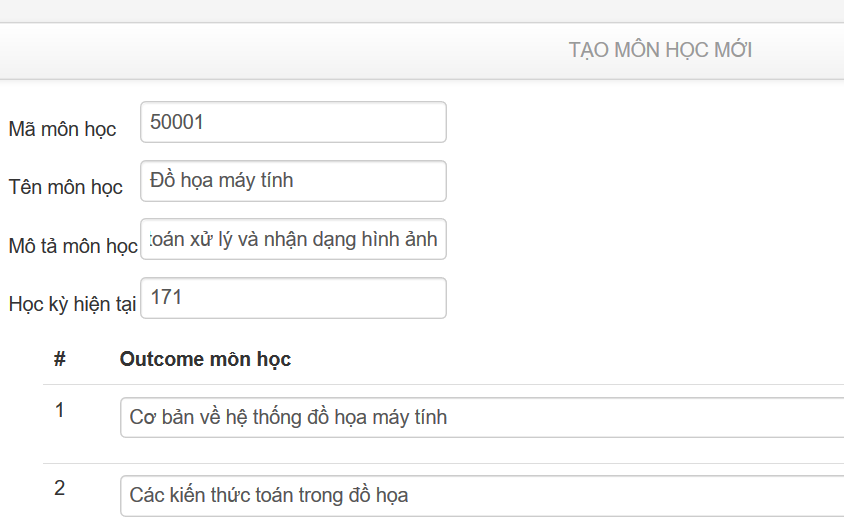
Hình 38 Mô tả chức năng hiển thị thống kê

Bên cạnh đó, trang web còn cho phép giảng viên tải bảng điểm đang hiển thị về máy dưới địnhh dạng CSV. Bảng điểm lưu trữ nội dung điểm số cũng như bảng thống kê dựa vào điểm tổng kết môn học của các sinh viên.



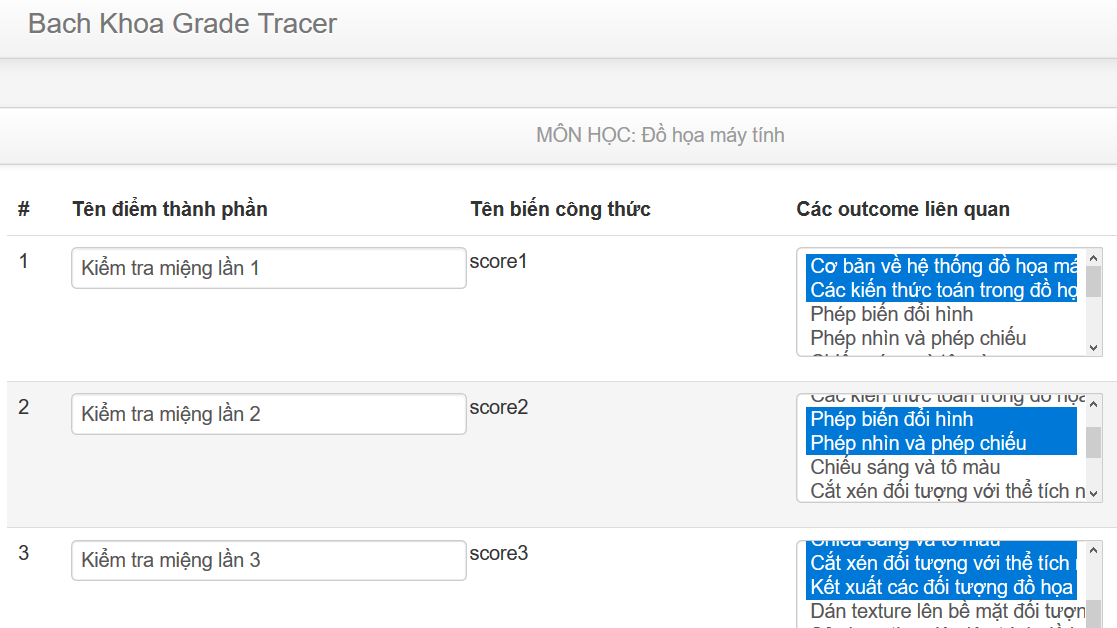
Hình 39 Mô tả chức năng tải bảng điểm về máy tính giảng viên

Giảng viên có thể tạo môn học mới và nhập các thông tin cần thiết liên quan đến môn học, chuẩn đầu ra, cũng như thao tác với các điểm thành phần của môn học.



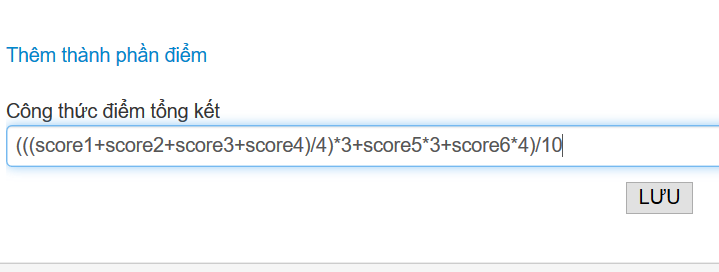
Hình 40 Mô tả chức năng tạo môn học mới với các thông tin môn học

Sau khi chọn tạo môn học, giảng viên sẽ khai báo các điểm thành phần có trong môn học, mà mối liên hệ giữa chúng với các mục tiêu đầu ra (outcome). Mỗi điểm thành phần có thể liên quan đến một hoặc nhiều mục tiêu đầu ra khác nhau.



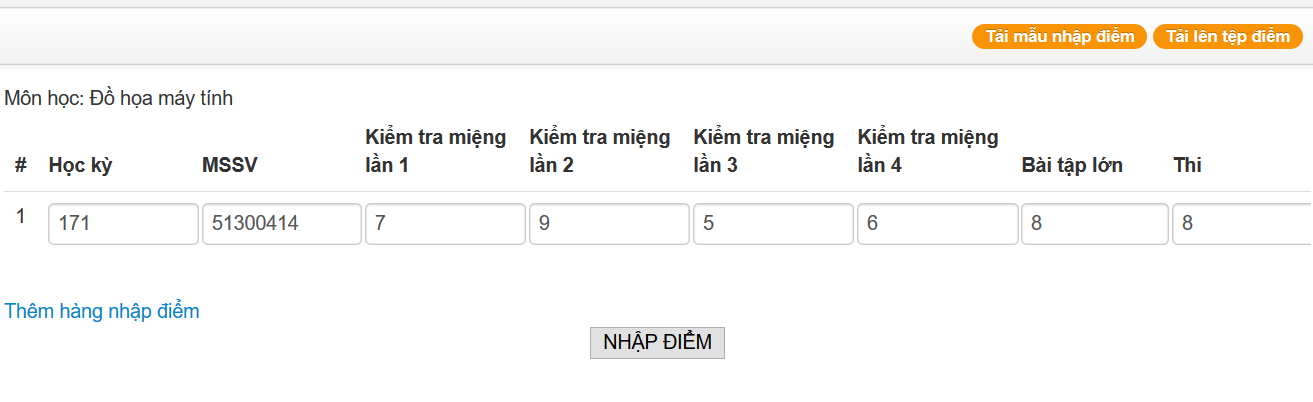
Hình 41 Mô tả chức năng tạo điểm thành phần liên hệ với outcome môn học

Giảng viên cần khai báo công thức tính điểm tổng kết dựa vào các biến được cung cấp như ở Hình 38. Công thức này là căn cứ để hệ thống tính điểm tổng kết mỗi khi được truy suất.



Hình 42 Mô tả việc khai báo công thức tính điểm tổng kết

Hệ thống cho phép giảng viên nhập điểm ngay tại giao diện của trang web.



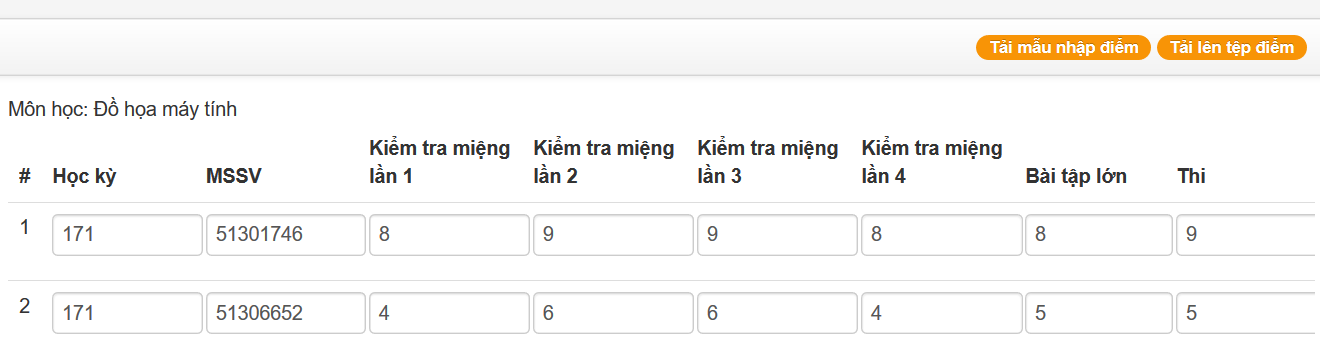
Hình 43 Mô tả chức năng nhập điểm tại trang web

Bằng một cách khác, giảng viên có thể nhấn vào nút "Tải mẫu nhập điểm" và nhập điểm số vào tệp tin CSV mà hệ thống gửi về. Việc tạo tệp nhập điểm cũng được thực hiện bằng mã lệnh javascript và được chạy trên máy đang truy cập đến hệ thống. Lưu ý tệp tin sau khi nhập điểm xong cần được lưu dưới dạng tệp CSV, với mỗi cột phân cách nhau bằng ký tự phẩy (',').



Hình 44 Mô tả bảng nhập điểm mà giảng viên tải về từ hệ thống

Sau đó, giảng viên tải lên tệp tin CSV đã nhập điểm bằng cách nhấn vào nút 'Tải lên tệp điểm' và chọn tệp tin CSV tương ứng. Khi đó, hệ thống sẽ tự động đọc tệp tin và chuyển dữ liệu sang bảng điểm tại trang web.



Hình 45 Mô tả kết quả sau khi chạy tính năng Tải lên tệp điểm

Suốt quá trình từ việc trích suất tạo tệp tin mẫu nhập điểm, cho đến việc mở tệp tin nhập điểm này và nhập thông tin vào biểu mẫu tại trang web, đều được thực hiện bằng mã javascript chạy trên máy người dùng. Trong trường hợp bảng điểm lớn, chức năng này giúp tiết kiệm việc xử lý cho máy chủ, nhằm nâng cao hiệu suất và giữ tính sẵn sàng để phục vụ các yêu cầu khác từ người dùng.

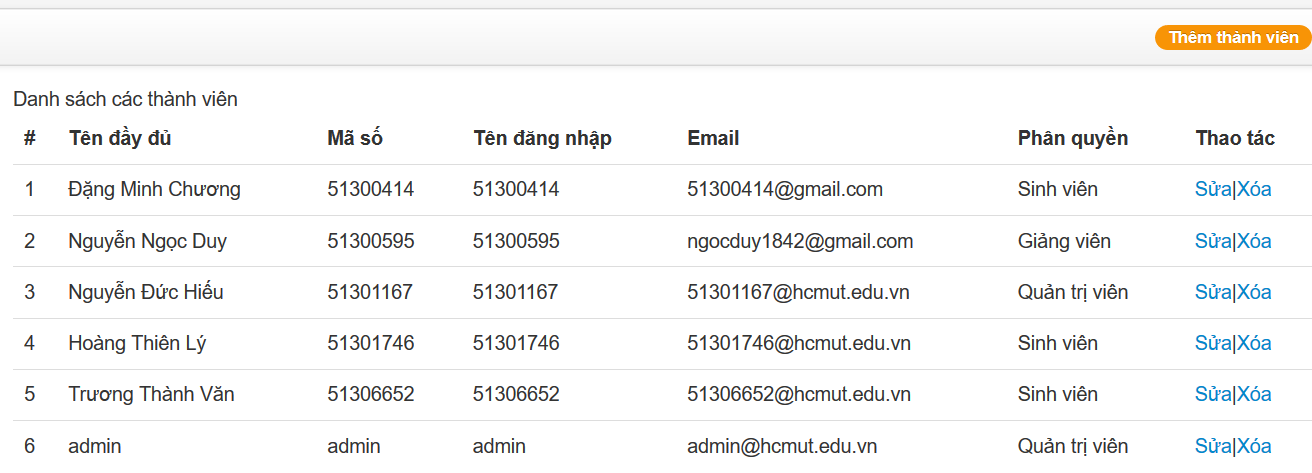
Kết quả của quá trình được hệ thống lưu lại như sau:



Hình 46 Mô tả kết quả của quá trình nhập điểm qua tệp tin

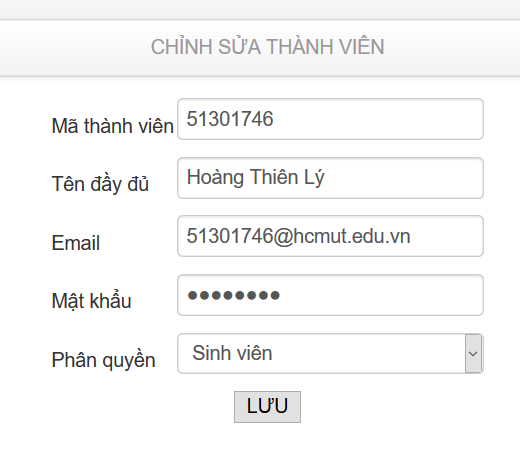
### VI.5.iii. Cho phép quản trị viên quản lý thông tin thành viên

Hệ thống cung cấp bảng thông tin tất cả các thành viên có trong hệ thống cho quản trị viên, để quản trị viên nắm rõ và kịp thời hỗ trợ thành viên khi cần thiết.



Hình 47 Mô tả bảng thông tin thành viên

Đồng thời quản trị viên cũng có quyền tạo mới thành viên, hay chỉnh sửa thông tin của thành viên có sẵn. Ngoài ra quản trị viên cũng có quyền xóa thành viên ra khỏi hệ thống khi thành viên không còn nhu cầu sử dụng.



Hình Mô tả chức năng tạo/chỉnh sửa thông tin thành viên

# VII. Danh mục tài liệu tham khảo

1. Freetuts.net. Học MVC PHP nâng cao <https://freetuts.net/hoc-php/hoc-mvc-php-nang-cao> [Ngày truy cập: 11 tháng 11 năm 2017].
2. Wikipedia. PHP <https://vi.wikipedia.org/wiki/PHP> [Ngày truy cập: 18 tháng 12 năm 2017].
3. Wikipedia. SQL <https://vi.wikipedia.org/wiki/SQL> [Ngày truy cập: 18 tháng 12 năm 2017].
4. Stackoverflow. Hiding an element <https://stackoverflow.com/questions/20518143/hiding-an-element-difference-between-javascript-attribute-and-css-style> [Ngày truy cập: 10 tháng 12 năm 2017].
5. Stackoverflow. Convert HTML table to array in javascript <https://stackoverflow.com/questions/9579721/convert-html-table-to-array-in-javascript> [Ngày truy cập: 10 tháng 12 năm 2017].
6. Hackermoon. 9 best javascript charting libraries <https://hackernoon.com/9-best-javascript-charting-libraries-46e7f4dc34e6> [Ngày truy cập: 10 tháng 12 năm 2017].
7. ASPSnippets. Read, Parse and display CSV (Text) file using JavaScript, jQuery and HTML5 <https://www.aspsnippets.com/Articles/Read-Parse-and-display-CSV-Text-file-using-JavaScript-jQuery-and-HTML5.aspx> [Ngày truy cập: 12 tháng 12 năm 2017].
8. Trang web khoa Khoa học và Kỹ thuật Máy tính, Đại học Bách khoa TP.HCM. Đề cương môn học Đồ họa máy tính <http://cse.hcmut.edu.vn/site/Media/PDF\_CS/CO3059.pdf> [Ngày truy cập: 12 tháng 11 năm 2017].