|  |  |
| --- | --- |
| logo (CMYK)-01 | BỘ GIÁO DỤC VÀ ĐÀO TẠO  **TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP. HCM** |

**ĐỒ ÁN TỔNG HỢP NGÀNH**

**Bảo bảo**

Ngành: **CÔNG NGHỆ THÔNG TIN**

Chuyên ngành: **HỆ THỐNG THÔNG TIN**

Giảng viên hướng dẫn :Th.S Văn Như Bích B

Sinh viên thực hiện :Tạ Tuấn Anh

MSSV:1711061011 Lớp:17DTHC1

TP. Hồ Chí Minh, 2021

**MỤC LỤC**

[**LỜI CAM ĐOAN** 3](#_Toc89368798)

[**LỜI CẢM ƠN** 3](#_Toc89368799)

[**CHƯƠNG 1: TỔNG QUAN** 4](#_Toc89368800)

[**CHƯƠNG 2: CƠ SỞ LÝ THUYẾT** 4](#_Toc89368801)

[**I.** **LÍ THUYẾT PHÂN TÍCH HỆ THỐNG THÔNG TIN** 4](#_Toc89368802)

[**II.** **Hệ quản trị cơ sở dữ liệu MySQL** 5](#_Toc89368803)

[**1.** **Khái niệm MySQL** 5](#_Toc89368804)

[**2.** **Ưa điểm của MySQL** 5](#_Toc89368805)

[**3.** **Giao diện phpMyadmin** 6](#_Toc89368806)

[**III.** **Ngôn ngữ lập trình PHP** 7](#_Toc89368807)

[**1.** **Tổng quan về PHP** 7](#_Toc89368808)

[**2.** **Lịch sử phát triển** 7](#_Toc89368809)

[**IV.** **Mô hình MVC** 8](#_Toc89368810)

[**1.** **MVC là gì?** 8](#_Toc89368811)

[**2.** **Các thành phần trong MVC** 9](#_Toc89368812)

[**3.** **Luồng dữ liệu trong MVC** 9](#_Toc89368813)

[**4.** **Ưa điểm và nhược điểm** 10](#_Toc89368814)

[**V.** **Khái quát về FRAMWORK LARAVEL** 10](#_Toc89368815)

[**1.** **Giới thiệu về Laravel** 10](#_Toc89368816)

[**2.** **Những ưu điểm của Laravel** 11](#_Toc89368817)

[**VI.** **BOOTSTRAP** 12](#_Toc89368818)

[**1.** **Bootstrap là gì?** 12](#_Toc89368819)

[**2.** **Ưu điểm** 12](#_Toc89368820)

[**VII.** **JAVASCRIPT** 14](#_Toc89368821)

[**1.** **Khái niệm về Javascript** 14](#_Toc89368822)

[**2.** **Ưu điểm và nhược điểm** 14](#_Toc89368823)

[**VIII.** **JQUERY** 15](#_Toc89368824)

[**1.** **Khái niệm về Jquery** 15](#_Toc89368825)

[**2.** **Ưa điểm và nhược điểm** 15](#_Toc89368826)

[**CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ** 16](#_Toc89368827)

[*3.1 Mục tiêu xây dựng:* 16](#_Toc89368828)

[*3.2 Thiết kế thành phần dữ liệu.* 17](#_Toc89368829)

[**1.** **Mô hình dữ liệu mức quan niệm (Conceptual Data Model - CDM)** 17](#_Toc89368830)

[**2.** **Mô hình dữ liệu mức luận lý (Logical Data Model - LDM)** 17](#_Toc89368831)

[**3.** **Mô hình dữ liệu mức vật lý (Physical Data Model - PDM)** 17](#_Toc89368832)

[*3.3 Danh sách các thực thể* 18](#_Toc89368833)

[**CHƯƠNG 4: GIỚI THIỆU TRANG WEB** 22](#_Toc89368834)

[**TÀI LIỆU THAM KHẢO** 31](#_Toc89368835)

**MỤC LỤC HÌNH**

[Hình 1. Mô hình MVC 10](#_Toc89370546)

[Hình 2. Thống kê sử dụng Framework Laravel 12](#_Toc89370547)

[Hình 3. Mô hình dữ liệu mức quan niệm (Conceptual Data Model - CDM) 18](#_Toc89370548)

[Hình 4. Mô hình dữ liệu mức luận lý (Logical Data Model - LDM) 18](#_Toc89370549)

[Hình 5. Mô hình dữ liệu mức vật lý (Physical Data Model - PDM) 19](#_Toc89370550)

[Hình 6. Giao diện đăng nhập 24](#_Toc89370551)

[Hình 7. Giao diện thông tin cá nhân 24](#_Toc89370552)

[Hình 8. Giao diện đăng kí môn học 25](#_Toc89370553)

[Hình 9. Giao diện yêu cầu phúc khảo 25](#_Toc89370554)

[Hình 10. Giao diện đang nhập giáo viên 26](#_Toc89370555)

[Hình 11. Giao diện thông tin giáo viên 26](#_Toc89370556)

[Hình 12. Giao diện danh sách môn học của sinh viên 27](#_Toc89370557)

[Hình 13. Giao diện nhập điểm của giáo viên 27](#_Toc89370558)

[Hình 14. Giao diện thống kê của giáo viên 28](#_Toc89370559)

[Hình 15. Giao diện in thông tin của giáo viên 28](#_Toc89370560)

[Hình 16. Giao diện thông tin Admin 29](#_Toc89370561)

[Hình 17. Giao diện thêm môn học 29](#_Toc89370562)

[Hình 18. Giao diện thêm môn học 29](#_Toc89370563)

[Hình 19. Giao diện thêm giáo viên 30](#_Toc89370564)

[Hình 20. Giao diện thêm giáo viên 30](#_Toc89370565)

[Hình 21. Giao diện thêm sinh viên 31](#_Toc89370566)

[Hình 22. Giao diện thêm sinh viên 31](#_Toc89370567)

[Hình 23. Giao diện thêm thêm học kì 32](#_Toc89370568)

**MỤC LỤC BẢNG**

[Bảng 1. SINH\_VIEN 20](#_Toc89370769)

[Bảng 2. LOP 21](#_Toc89370770)

[Bảng 3. PHUC\_KHAO 21](#_Toc89370771)

[Bảng 4. MON\_HOC\_SINH\_VIEN 21](#_Toc89370772)

[Bảng 5. NAM\_HOC 22](#_Toc89370773)

[Bảng 6. MON\_HOC 23](#_Toc89370774)

[Bảng 6. GIAO\_VIEN 23](#_Toc89370775)

[Bảng 8. HOC\_KI 24](#_Toc89370776)

# **LỜI CAM ĐOAN**

Tôi xin cam đoan đồ án tổng hợp “” là công trình nghiên cứu của bản thân dưới sự hướng dẫn của Các nội dung, kết quả nghiên cứu trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những thông tin được tham khảo của các tác giả, cơ quan tổ chức trong quá trình thực hiện đồ án đã được nêu rõ trong phần tài liệu tham khảo.

Tôi hoàn toàn chịu trách nhiệm về nội dung đồ án của mình.

Thành phố Hồ Chí Minh, năm 2021.

# **LỜI CẢM ƠN**

Chúng em xin gửi lời cảm ơn chân thành BBBBB người thầy đã hướng dẫn chúng em. Cảm ơn thầy vì đã tạo điều kiện cũng như giúp đỡ để chúng em có thể hoàn thành đồ án một cách thành công. Nhờ có thầy mà chúng em mới cố gắng tìm tòi nghiên cứu để xây dựng đồ án của mình. Tạo cho chúng em động lực học hỏi những kiến thức mới để chúng em ngày càng hoàn thiện bản thân mình hơn. Đây sẽ là cái đà để tụi em đi sâu vào chuyên ngành của mình và thành công trên con đường tương lai phía trước. Vì là lần đầu thực hiện cũng như trong lúc hoàn chỉnh phần mềm chúng em gặp nhiều khó khăn nên không tránh khỏi những thiếu xót. Mong cccc nhận xét và góp ý để chúng em có thêm nhiều hiểu biết và kinh nghiệm.

Một lần nữa chúng em xin chân thành cảm ơn thầy!

# **CHƯƠNG 1: TỔNG QUAN**

Đề tài “abc” được thực hiện với kiến thức tổng hợp từ các môn phân tích thiết kế hệ thống thông tin, cơ sở dữ liệu, cấu trúc dữ liệu,... Các module khác như giao diện nhập liệu, xử lý lưu trữ, v.. v.. được thực hiện với các kiến thức từ một số môn lập trình khác.

# **CHƯƠNG 2: CƠ SỞ LÝ THUYẾT**

## LÍ THUYẾT PHÂN TÍCH HỆ THỐNG THÔNG TIN

Hệ thống là một tập hợp các đối tượng, các thành phần có liên quan với nhau, tương tác với nhau theo những nguyên tắc, những cơ chế nào đó cùng tồn tại trong một thể thống nhất.

Hệ thống thông tin là một tập hợp và kết hợp của các phần cứng, phần mềm và các hệ mạng truyền thông được xây dựng và sử dụng để thu thập, tái tạo, phân phối và chia sẻ các dữ liệu, thông tin và tri thức nhằm phục vụ các mục tiêu của tổ chức.

Phân tích và thiết kế hệ thống thông tin là thực hiện các bước phân tích, xử lý có tổ chức một cách khoa học do một nhóm các chuyên gia về hệ thống thực hành nhằm mục đích phát triển và duy trì những hệ thống thông tin trên nền tảng máy tính.

Mô hình dữ liệu mức quan niệm (CDM – Conceptual Data Model) là mô hình chi tiết mô tả toàn bộ cấu trúc dữ liệu tổ chức mà nó không phụ thuộc bất kỳ một hệ quản trị cơ sở dữ liệu nào hay sự xem xét việc cài đặt.

## Hệ quản trị cơ sở dữ liệu MySQL

### Khái niệm MySQL

MySQL là [hệ quản trị cơ sở dữ liệu](https://vi.wikipedia.org/wiki/H%E1%BB%87_qu%E1%BA%A3n_tr%E1%BB%8B_c%C6%A1_s%E1%BB%9F_d%E1%BB%AF_li%E1%BB%87u) [tự do nguồn mở](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_m%E1%BB%81m_ngu%E1%BB%93n_m%E1%BB%9F) phổ biến nhất thế giới và được các nhà phát triển rất ưa chuộng trong quá trình phát triển ứng dụng. Vì MySQL là cơ sở dữ liệu tốc độ cao, ổn định và dễ sử dụng, có tính khả chuyển, hoạt động trên nhiều hệ điều hành cung cấp một hệ thống lớn các hàm tiện ích rất mạnh. Với tốc độ và tính bảo mật cao, MySQL rất thích hợp cho các ứng dụng có truy cập cơ sở dữ liệu (CSDL) trên internet. Người dùng có thể tải về MySQL miễn phí từ trang chủ. MySQL có nhiều phiên bản cho các hệ điều hành khác nhau: phiên bản Win32 cho các hệ điều hành dòng [Windows](https://vi.wikipedia.org/wiki/Microsoft_Windows), [Linux](https://vi.wikipedia.org/wiki/Linux), Mac OS X, [Unix](https://vi.wikipedia.org/wiki/Unix), [FreeBSD](https://vi.wikipedia.org/wiki/FreeBSD), [NetBSD](https://vi.wikipedia.org/w/index.php?title=NetBSD&action=edit&redlink=1), [Novell NetWare](https://vi.wikipedia.org/w/index.php?title=Novell_NetWare&action=edit&redlink=1), [SGI Irix](https://vi.wikipedia.org/w/index.php?title=SGI_Irix&action=edit&redlink=1), [Solaris](https://vi.wikipedia.org/wiki/Solaris), [SunOS](https://vi.wikipedia.org/w/index.php?title=SunOS&action=edit&redlink=1),..

MySQL là một trong các ví dụ rất cơ bản về Hệ Quản trị Cơ sở dữ liệu quan hệ sử dụng ngôn ngữ truy vấn có cấu trúc (SQL).

MySQL được sử dụng cho việc bổ trợ NodeJs, [PHP](https://vi.wikipedia.org/wiki/PHP), [Perl](https://vi.wikipedia.org/wiki/Perl), và nhiều ngôn ngữ khác, làm nơi lưu trữ những thông tin trên các trang web viết bằng NodeJs, PHP hay Perl...

### Ưa điểm của MySQL

Tốc độ: MySQL rất nhanh. Những nhà phát triển cho rằng MySQL là CSDL nhanh nhất mà bạn có thể có.

Dễ sử dụng: MySQL tuy có tính năng cao nhưng thực sự là một hệ thống CSDL rất đơn giản và ít phức tạp khi cài đặt và quản trị hơn các hệ thống lớn.

Giá thành: miễn phí cho hầu hết các việc sử dụng.

Hỗ trợ ngôn ngữ truy vấn: MySQL hiểu SQL, là ngôn ngữ của sự chọn lựa cho tất cả các hệ thống CSDL hiện đại.

Độ bảo mật cao: MySQL rất thích hợp cho các ứng dụng có truy cập CSDL trên Internet khi sở hữu nhiều tính năng bảo mật thậm chí là ở cấp cao.

Tính linh động: MySQL chạy trên nhiều hệ thống UNIX cũng như không phải UNIX chẳng hạn như Windows hay OS/2. MySQL chạy được với mọi phần cứng từ các máy PC ở nhà cho đến các máy server.

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

### Giao diện phpMyadmin

phpMyAdmin là một công cụ [nguồn mở miễn phí](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_m%E1%BB%81m_ngu%E1%BB%93n_m%E1%BB%9F) được viết bằng [PHP](https://vi.wikipedia.org/wiki/PHP) dự định để xử lý quản trị của [MySQL](https://vi.wikipedia.org/wiki/MySQL) thông qua một [trình duyệt web](https://vi.wikipedia.org/wiki/Tr%C3%ACnh_duy%E1%BB%87t_web). Nó có thể thực hiện nhiều tác vụ như tạo, sửa đổi hoặc xóa bỏ [cơ sở dữ liệu](https://vi.wikipedia.org/wiki/C%C6%A1_s%E1%BB%9F_d%E1%BB%AF_li%E1%BB%87u), [bảng](https://vi.wikipedia.org/w/index.php?title=B%E1%BA%A3ng_(c%C6%A1_s%E1%BB%9F_d%E1%BB%AF_li%E1%BB%87u)&action=edit&redlink=1), các [trường](https://vi.wikipedia.org/w/index.php?title=Tr%C6%B0%E1%BB%9Dng_(c%C6%A1_s%E1%BB%9F_d%E1%BB%AF_li%E1%BB%87u_quan_h%E1%BB%87)&action=edit&redlink=1) hoặc [bản ghi](https://vi.wikipedia.org/w/index.php?title=B%E1%BA%A3n_ghi&action=edit&redlink=1); thực hiện báo cáo [SQL](https://vi.wikipedia.org/wiki/SQL); hoặc quản lý người dùng và cấp phép.

## Ngôn ngữ lập trình PHP

### Tổng quan về PHP

PHP là viết tắt của từ "Hypertext Preprocessor" là một [ngôn ngữ lập trình](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh) kịch bản hay một loại [mã lệnh](https://vi.wikipedia.org/w/index.php?title=M%C3%A3_l%E1%BB%87nh&action=edit&redlink=1) chủ yếu được dùng để phát triển các [ứng dụng](https://vi.wikipedia.org/wiki/%E1%BB%A8ng_d%E1%BB%A5ng_web) viết cho máy chủ (server side). PHP là một open source, vì thế nó có tính cộng đồng cao và đồng thời có rất nhiều framework, CMS hỗ trợ như Laravel, Wordpress.

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 (LAMP).

Apache là một phần mềm web server có nhiệm vụ tiếp nhận request từ trình duyệt người dùng sau đó chuyển giao cho PHP xử lý và gửi trả lại cho trình duyệt.

MySQL cũng tương tự như các hệ quản trị cơ sở dữ liệu khác (Postgress, Oracle, SQL server...) đóng vai trò là nơi lưu trữ và truy vấn dữ liệu.

Linux: Hệ điều hành mã nguồn mở được sử dụng rất rộng rãi cho các web server. Thông thường các phiên bản được sử dụng nhiều nhất là RedHat, Enterprise Linux, Ubuntu...

### Lịch sử phát triển

PHP: được phát triển từ một sản phẩm có tên là [PHP/FI](https://vi.wikipedia.org/w/index.php?title=PHP/FI&action=edit&redlink=1). PHP/FI do [Rasmus Lerdorf](https://vi.wikipedia.org/w/index.php?title=Rasmus_Lerdorf&action=edit&redlink=1) tạo ra năm [1994](https://vi.wikipedia.org/wiki/1995), ban đầu được xem như là một tập con đơn giản của các [mã kịch bản](https://vi.wikipedia.org/w/index.php?title=M%C3%A3_k%E1%BB%8Bch_b%E1%BA%A3n&action=edit&redlink=1) [Perl](https://vi.wikipedia.org/wiki/Perl) để theo dõi tình hình truy cập đến bản sơ yếu lý lịch của ông trên mạng. Ông đã đặt tên cho bộ mã kịch bản này là 'Personal Home Page Tools'. Vào năm [1997](https://vi.wikipedia.org/wiki/1997), PHP/FI 2.0, lần viết lại thứ hai của phiên bản C, đã thu hút được hàng ngàn người sử dụng trên toàn thế giới với xấp xỉ 50.000 tên miền đã được ghi nhận là có cài đặt nó, chiếm khoảng 1% số tên miền có trên mạng Internet.

PHP 3: là phiên bản đầu tiên cho chúng ta thấy một hình ảnh gần gũi với các phiên bản PHP ngày nay được [Andi Gutmans](https://vi.wikipedia.org/w/index.php?title=Andi_Gutmans&action=edit&redlink=1) và [Zeev Suraski](https://vi.wikipedia.org/w/index.php?title=Zeev_Suraski&action=edit&redlink=1) tạo ra năm 1997 do họ nhận thấy PHP/FI 2.0 hết sức yếu kém trong việc phát triển các ứng dụng [thương mại điện tử](https://vi.wikipedia.org/wiki/Th%C6%B0%C6%A1ng_m%E1%BA%A1i_%C4%91i%E1%BB%87n_t%E1%BB%AD) mà họ đang xúc tiến trong một dự án của trường đại học. Trong một nỗ lực hợp tác và bắt đầu xây dựng dựa trên cơ sở người dùng đã có của PHP/FI, Andi, Rasmus và Zeev đã quyết định hợp tác và công bố PHP 3.0 như là phiên bản thế hệ kế tiếp của PHP/FI 2.0, và chấm dứt phát triển PHP/FI 2.0. PHP 3.0 đã chính thức được công bố vào tháng 6 năm 1998, sau thời gian 9 tháng được cộng đồng kiểm nghiệm.

PHP 4: mùa đông năm 1998, ngay sau khi PHP 3.0 chính thức được công bố, Andi Gutmans và Zeev Suraski đã bắt đầu bắt tay vào việc viết lại phần lõi của PHP. Với Một động cơ mới, có tên '[Zend Engine](https://vi.wikipedia.org/w/index.php?title=Zend_Engine&action=edit&redlink=1)' (ghép từ các chữ đầu trong tên của Zeev và Andi), đã đáp ứng được các nhu cầu thiết kế này một cách thành công, và lần đầu tiên được giới thiệu vào giữa năm [1999](https://vi.wikipedia.org/wiki/1999). PHP 4.0, dựa trên động cơ này, và đi kèm với hàng loạt các tính năng mới bổ sung, đã chính thức được công bố vào tháng 5 năm [2000](https://vi.wikipedia.org/wiki/2000). PHP 4.0 tốc độ xử lý được cải thiện rất nhiều đem đến các tính năng chủ yếu khác gồm có sự hỗ trợ nhiều máy chủ Web hơn, hỗ trợ phiên làm việc HTTP, tạo bộ đệm thông tin đầu ra, nhiều cách xử lý thông tin người sử dụng nhập vào bảo mật hơn và cung cấp một vài các cấu trúc ngôn ngữ mới.

PHP 5: Các bản beta 1, 2, 3 của PHP 5 đã lần lượt công bố để cộng đồng trãi nghiệm. PHP năm bản chính thức đã ra mắt ngày [13 tháng 7](https://vi.wikipedia.org/wiki/13_th%C3%A1ng_7) năm [2004](https://vi.wikipedia.org/wiki/2004) sau một chuỗi khá dài các bản kiểm tra thử bao gồm Beta 4, RC 1, RC2, RC3. Mặc dù coi đây là phiên bản sản xuất đầu tiên nhưng PHP 5.0 vẫn còn một số lỗi trong đó đáng kể là lỗi xác thực HTTP. Ngày [14 tháng 7](https://vi.wikipedia.org/wiki/14_th%C3%A1ng_7) năm [2005](https://vi.wikipedia.org/wiki/2005), PHP 5.1 Beta 3 được PHP Team công bố đánh dấu sự chín muồi mới của PHP với sự có mặt của PDO, một nỗ lực trong việc tạo ra một hệ thống API nhất quán trong việc truy cập cơ sở dữ liệu và thực hiện các câu truy vấn.

PHP 6: Phiên bản PHP 6 được kỳ vọng sẽ lấp đầy những khiếm khuyết của PHP ở phiên bản hiện tại, ví dụ: hỗ trợ [namespace](https://vi.wikipedia.org/wiki/Kh%C3%B4ng_gian_t%C3%AAn) (hiện tại các nhà phát triển vẫn chưa công bố rõ ràng về vấn đề này); hỗ trợ [Unicode](https://vi.wikipedia.org/wiki/Unicode); sử dụng [PDO](https://vi.wikipedia.org/w/index.php?title=PDO&action=edit&redlink=1) làm [API](https://vi.wikipedia.org/wiki/Giao_di%E1%BB%87n_l%E1%BA%ADp_tr%C3%ACnh_%E1%BB%A9ng_d%E1%BB%A5ng) chuẩn cho việc truy cập cơ sở dữ liệu, các [API](https://vi.wikipedia.org/wiki/Giao_di%E1%BB%87n_l%E1%BA%ADp_tr%C3%ACnh_%E1%BB%A9ng_d%E1%BB%A5ng) cũ sẽ bị đưa ra thành thư viện [PECL](https://vi.wikipedia.org/w/index.php?title=PECL&action=edit&redlink=1)... Phiên bản 6 này chỉ dùng ở việc nghiên cứu và thử nghiệm. Sau này PHP bỏ hẳn phiên bản 6 và lên 7.

PHP 7: Với việc sử dụng bộ nhân Zend Engine mới PHPNG cho tốc độ nhanh gấp 2 lần. Ngoài ra ở phiên bản này còn thêm vào rất nhiều cú pháp, tính năng mới giúp cho PHP trở nên mạnh mẽ hơn

## Mô hình MVC

### MVC là gì?

MVC là một kiểu kiến trúc đặc trưng phổ biến trong các ngôn ngữ lập trình, cho phép tách rời tầng xử lý và tầng giao diện (UI) thành các thành phần riêng biệt. Tầng xử lý có chức năng chính là xử lý, trao đổi thông tin giữa CSDL và giao diện vì vậy ta có thể chỉnh sửa các thành phần đó một các riêng biệt từ đó loại bỏ được rất nhiều lầm lẫn và đơn giản hóa toàn bộ quá trình phát triển.

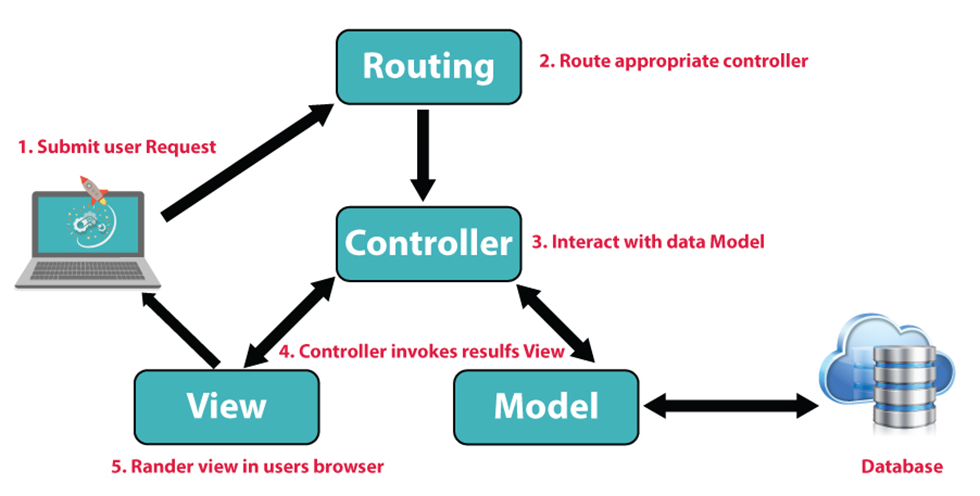
### Các thành phần trong MVC

Model (Tầng dữ liệu): là một đối tượng hoặc một tập hợp đối tượng biểu diễn cho phần dữ liệu, thường chịu trách nhiệm quản lý dữ liệu như kết nối đến cơ sở dữ liệu, truy vấn thêm, sửa, xóa dữ liệu.

View (Tầng giao diện): thực hiện việc trình bày dữ liệu lên giao diện và tương tác với người dùng. Thông thường các thông tin hiển thị lên View được lấy từ Model.

Controller (Tầng điều khiển): đóng vai trò trung gian, giúp Model và View tương tác với nhau và là tầng chịu trách nhiệm xử lý các tác động từ giao diện View. Sau khi xử lý các yêu cầu được gửi từ người dùng, Controller sẽ tương tác với đến Model để lấy dữ liệu cần thiết để đáp ứng các yêu cầu vừa được gửi. Cuối cùng Controller trả dữ liệu này về cho View.

### Luồng dữ liệu trong MVC



Hình 1. Mô hình MVC

Mô hình hoạt động như sau:

1. Người dùng sử dụng một BROWSER trình duyệt web bất kỳ (Firefox, Chrome, IE,…)để có thể gửi những yêu cầu (HTTP Request) có thể kèm theo những dữ liệu nhập tới những CONTROLLER xử lý tương ứng. Việc xác định Controllerr xử lý sẽ dựa vào một bộ Routing điều hướng.
2. Khi CONTROLLER nhận được yêu cầu gửi tới, nó sẽ chịu trách nhiệm kiểm tra yêu cầu đó có cần dữ liệu từ MODEL hay không? Nếu có, nó sẽ sử dụng các class/function cần thiết trong MODEL và nó sẽ trả ra kết quả (Resulting Arrays), khi đó CONTROLLER sẽ xử lý giá trị đó và trả ra VIEW để hiển thị. CONTROLLER sẽ xác định các VIEW tương ứng để hiển thị đúng với yêu cầu.
3. Khi nhận được dữ liệu từ CONTROLLER, VIEW sẽ chịu trách nhiệm xây dựng các thành phẩn hiển thị như hình ảnh, thông tin dữ liệu… và trả về GUI Content để CONTROLLER đưa ra kết quả lên màn hình BROWSER.
4. BROWSER sẽ nhận giá trị trả về (HTTP Response) và sẽ hiển thị với người dùng. Kết thúc một quy trình hoạt động.

### Ưa điểm và nhược điểm

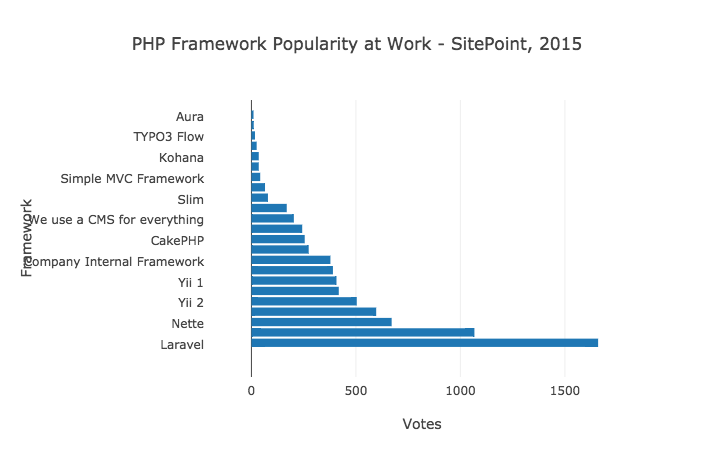
* **Ưu điểm:**
* Dự án được tổ chức một cách chặt chẽ và rõ ràng.
* Hệ thống được phân ra thành từng phần tách biệt nên dễ dàng phát triển cũng như cập nhật.
* Hệ thống được chia ra thành nhiều chức năng nhỏ (module) nên thuận tiện cho việc phát hiện và sửa lỗi chức năng.
* Thuận lợi cho làm việc nhóm.
* Dễ bảo trì và nâng cấp.
* **Nhược điểm:**
* Hệ thống xử lý chậm hơn so với việc sử dụng PHP thuần vì tốn thời gian trung chuyển dữ liệu giữa các thành phần.
* Đối với các dự án nhỏ, việc sử dụng mô hình MVC tạo ra sự cồng kềnh và tốc độ xử lý cũng chậm hơn.
* Tốn khá nhiều thời gian để xây dựng cấu trúc, thư viện.

## Khái quát về FRAMWORK LARAVEL

### Giới thiệu về Laravel

Laravel là một thư viện mã nguồn mở miễn phí, được tạo ra bởi Taylor OtWell giúp xây dựng những ứng dụng theo mô hình MVC (Model - View - Controller) nhanh chóng do Laravel không những học hỏi được nhiều tính năng hay từ các framework khác mà bản thân Laravel cũng tự phát triển, nâng cấp lên nhiều chức năng. Laravel được thiết kế với tính khoa học cao, tài liệu học tập đầy đủ và dễ dàng tiếp cận nhất là đối với những người mới lập trình PHP.

Hiện nay, có rất nhiều PHP Framework hỗ trợ phát triển website, nhưng Laravel vẫn là framework phổ biến nhất. Kết quả được thống kê theo www.thetoptens.com



Hình 2. Thống kê sử dụng Framework Laravel

Dù Laravel được ra đời sau (tháng 4 năm 2011) nhưng lại rất phổ biến và có đông đảo người sử dụng nhất tính đến thời điểm hiện tại. Một số lý do giúp Laravel trở thành Framework phổ biến nhất:

* Thừa hưởng được những thế mạnh và khắc phục được những nhược điểm của các framework ra đời trước đó.
* Được cập nhật liên tục, nên các lỗi được khắc phục nhanh.
* Tài liệu học tập được thiết kế đầy đủ, rõ ràng và chi tiết hơn so với các PHP framework khác.
* Khi gặp khó khăn, dễ dàng được hỗ trợ bởi cộng đồng người dùng Laravel.
* Dễ quản lý và tích hợp các thư viện cần sử dụng vào dự án.
* Tự động nạp thư viện mã lệnh (autoload) theo không gian tên (namespace).
* Tính năng Migration giúp quản lý cơ sở dữ liệu hiệu quả hơn khi làm việc nhóm.

Việc quản lý bố cục cũng trở nên dễ dàng hơn dựa vào Blade Template.

### Những ưu điểm của Laravel

* Tốc độ xử lý nhanh: sự ra đời của Laravel như một làn gió mới đến với cộng đồng framework PHP. Nền tảng này có thể hỗ trợ để tạo nên các trang web lớn, dự án lớn. Có rất nhiều công ty công nghệ đã sử dụng nền tảng này để phát triển dự án của họ.
* Dễ sử dụng: một trong những lý do giúp Laravel được người dùng đón nhận và sử dụng nhiều nhất là do nó rất dễ sử dụng, chỉ mất một khoảng thời gian khá ngắn là có thể bắt tay vào tạo nên một dự án (project) nhỏ với Laravel.
* Mã nguồn mở: Laravel được xây dựng với mã nguồn mở và hoàn toàn miễn phí. Mỗi khi nền tảng này được cập nhật, người dùng lại có thêm nhiều tính năng độc đáo để khám phá và ứng dụng vào các dự án công nghệ của mình.
* Được xây dựng theo mô hình MVC: với tiêu chuẩn này, giúp người dùng có thể duy trì, phát triển và mở rộng quy mô dự án.
* Tích hợp nhiều tính năng có sẵn: Laravel cung cấp cho người dùng rất nhiều các nhóm tính năng giúp quá trình phát triển trở nên nhanh chóng.
* Tính bảo mật cao: cung cấp cho người dùng các tính năng bảo mật cơ bản như: ORM (chống lại tấn công SQL Injection), field token ẩn (chống lại tấn công kiểu CSRF)
* Cộng đồng người dùng đông đảo: Laravel có đông đảo người sử dụng, sẵn sàng giúp đỡ và hỗ trợ bạn trong quá trình thiết lập và vận hành dự án. Đặc biệt việc tìm lỗi cũng trở nên nhanh chóng, dễ dàng và tiết kiệm thời gian.

## BOOTSTRAP

### Bootstrap là gì?

Bootstrap là 1 framework HTML, CSS, và JavaScript cho phép người dùng dễ dàng thiết kế website theo 1 chuẩn nhất định, tạo các website thân thiện với các thiết bị cầm tay như mobile, ipad, tablet.

Bootstrap là bao gồm các HTML templates, CSS templates và Javascript tao ra những cái cơ bản có sẵn như: typography, forms, buttons, tables, navigation, modals, image carousels và nhiều thứ khác. Trong bootstrap có thêm các plugin Javascript trong nó. Giúp cho việc thiết kế reponsive của bạn dễ dàng hơn và nhanh chóng hơn.

### Ưu điểm

Dễ sử dụng: vì nó được dựa trên HTML, CSS và Javascript nên chỉ cần có kiến thức cơ bản về ba cái đó là có thể sử dụng Bootstrap tốt.

Sự tương thích: tương thích tốt với tất cả các trình duyệt web (như Chrome, Firefox, Internet Explorer, Safari and Opera). Tuy nhiên, với IE Browser, Bootstrap chỉ hỗ trợ từ IE9 trở lên) vì IE8 không support HTML5 và CSS3.

Bootstrap giúp các nhà thiết kế website có thể giảm thiểu được thời gian trong việc thiết kế website. Khi sử dụng Bootstrap, có thể hạn chế được thời gian bỏ ra để viết code cho giao diện. Vì trong thư viện của nó đã có từ trước, những đoạn mã có thể chèn vào trong website.

Khi sử dụng Bootstrap hoàn toàn có thể tùy biến được giao diện theo ý muốn của cá nhân. Với hệ thống Grid System đã mặc định bao gồm 12 cột cùng với độ rộng 940px được tích hợp sẵn ở Bootstrap.

Ngày nay, khi mà xu hướng phát triển website tương thích được với các thiết bị thì Bootstrap nổi lên như một framework có sẵn responsive CSS phù hợp với mọi thiết bị như smartphone, tablets, desktop…

Gần đây, các nhà lập trình đã bổ sung thêm tính năng Customize (Tùy chỉnh). Chức năng này giúp cho các nhà thiết kế linh hoạt hơn trong việc lựa chọn những thuộc tính, phần tử phù hợp với dự án họ đang theo đuổi. Chức năng này cũng cho phép người dùng không cần phải tải toàn bộ mã nguồn về máy.

## JAVASCRIPT

### Khái niệm về Javascript

Javascript là một ngôn ngữ lập trình kịch bản Client – Side (Client – Side Script Language) dựa vào đối tượng (hướng đối tượng Object – Oriented programming) phát triển có sẵn hoặc tự định nghĩa ra, Javascript nhẹ và được sử dụng rộng rãi trong các ứng dụng website như là một phần của các trang web, được hỗ trợ hầu hết trên tất cả các trình duyệt như Firefox, Chrome… thậm chí các trình duyệt trên thiết bị di động cũng có hỗ trợ. Javascript có khả năng kết nối và thao tác CSDL trên server. Javascript được nhúng vào HTML và kết hợp với các ngôn ngữ lập trình web như ASP/ASP.NET, PHP, JSP tạo ra các trang web động.

### Ưu điểm và nhược điểm

**2.1 Ưu điểm:**

* Javascript được thiết kế độc lập với hệ điều hành. Nó có thể chạy trên bất kì hệ điều hành nào mà có trình duyệt hỗ trợ Javascript.
* Dễ dàng tương tác, điều khiển và tránh bớt việc xử lý từ phía server.
* Nắm vững kiến thức Javascript bây giờ rất hữu dụng cho các bạn sau này để có thể tiếp thu những công nghệ mới mà nó được gói họn vào những ngôn ngữ như: Ajax, Atlas…
* Phản hồi ngay lập tức tới khách truy cập: họ không phải chờ cho một trang web tải lại để thấy xem nếu họ đã quên nhập cái gì đó.
* Khả năng tương tác tăng lên: có thể tạo các giao diện mà phản ứng lại người sử dụng rẽ chuột qua chúng hoặc kích hoạt chúng thông qua bàn phím.
* Giao diện phong phú hơn: có thể sử dụng Javascript để bao gồm những mục như các thành phần Drag và Drop (DnD) và các con trượt (Slider) để cung cấp một Rich Interface (Giao diện giàu tính năng) tới site khách truy cập.

**2.2 Nhược điểm:**

* Javascript không có trình biên dịch riêng mà được diễn dịch và chạy bởi trình duyệt hỗ trợ nó. Chính vì thế, nếu trình duyệt không hỗ trợ, hoặc không bật JS nó sẽ không chạy được.
* Javascript có thể thể làm ứng web của bạn trở nên nặng nề hơn.
* Bảo mật kém. Không có khả năng giấu mã.

## JQUERY

### Khái niệm về Jquery

jQuery là một Framework được xây dựng dựa trên các tính năng của JavaScript. jQuery làm đơn giản hóa việc truyền tải HTML, xử lý sự kiện, tạo hiệu ứng động và tương tác Ajax. Một số tinh năng được hỗ trợ bởi jQuery:

* Thao tác DOM − jQuery giúp dễ dàng lựa chọn các phần tử DOM để traverse (duyệt) một cách dễ dàng như sử dụng CSS, và chỉnh sửa nội dung của chúng bởi sử dụng phương tiện Selector mã nguồn mở, mà được gọi là Sizzle.
* Xử lý sự kiện − jQuery giúp tương tác với người dùng tốt hơn bằng việc xử lý các sự kiện đa dạng mà không làm cho HTML code rối tung lên với các Event Handler.
* Hỗ trợ AJAX − jQuery giúp bạn rất nhiều để phát triển một site giàu tính năng và phản hồi tốt bởi sử dụng công nghệ AJAX.
* Hiệu ứng − jQuery đi kèm với rất nhiều các hiệu ứng đa dạng và đẹp mắt mà bạn có thể sử dụng trong các Website của mình.
* Gọn nhẹ − jQuery là thư viện gọn nhẹ - nó chỉ có kích cỡ khoảng 19KB (gzipped).
* Được hỗ trợ hầu hết bởi các trình duyệt hiện đại − jQuery được hỗ trợ hầu hết bởi các trình duyệt hiện đại, và làm việc tốt trên IE 6.0+, FF 2.0+, Safari 3.0+, Chrome và Opera 9.0+

### Ưa điểm và nhược điểm

**2.1 Ưu điểm:**

* Dễ sử dụng: Đây là lợi thế chính khi sử dụng jquery, nó dễ dàng hơn so với nhiều thư viện javascript chuẩn khác bởi cú pháp đơn giản và bạn chỉ phải viết ít dòng lệnh để tạo ra các chức năng tương tự. Chỉ với 10 dòng lệnh JQuery bạn có thể thay thế cả 20 chục dòng lệnh DOM javaScript, tiết kiệm thời gian của người lập trình.
* Là một thư viện lớn của javascript: Thực thi được nhiều chức năng hơn so với các thư viện javascript khác
* Cộng đồng mã nguồn mở mạnh mẽ (một số plugin jquery có sẵn): JQuery đang còn tương đối mới, có một cộng đồng dành thời gian của họ để phát triển các plugin của JQuery. Như vậy có hàng trăm plugin được viết trước đó có sẵn để tải về ngay lập tức để đẩy nhanh quá trình viết code của bạn. Một lợi thế khác đằng sau này là hiệu quả và an toàn của các script.
* Có nhiều tài liệu và hướng dẫn chi tiết: Các trang web JQuery có một toàn bộ tài liệu và hướng dẫn để ngay cả một người mới bắt đầu lập trình cũng có thể làm được quả bóng lăn với thư viện jquery này.
* Hỗ trợ ajax: JQuery cho phép bạn phát triển các template Ajax một cách dễ dàng. Ajax cho phép một giao diện kiểu dáng đẹp trên website, các chức năng có thể được thực hiện trên các trang mà không đòi hỏi toàn bộ trang được reload lại.

# **CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ**

## *3.1 Mục tiêu xây dựng:*

Đề tài “ ” gồm các tác nhân và những chức năng chính:

* Admin: quản trị hệ thống
  + Quản lí môn học:
    - Cho phép admin xem, thêm và cập nhật môn học theo từng học kì năm học và cả giáo viên dạy môn đó ở học kì cụ thể.
  + Quản lí giáo viên
    - Cho phép admin xem, thêm và cập nhật một số thông tin cá nhân của giáo viên
  + Quản lí sinh viên
    - Cho phép admin xem, thêm và cập nhật thông tin sinh viên
  + Quản lí năm học
    - Cho phép admin chọn học kì đang học, nhằm giúp thống kê thông tin chính xác rõ ràng.
* Giáo viên:
  + Xem danh sách môn học
    - Cho phép giáo viên xem thông tin các sinh viên đã đăng kí môn học mà mình phụ trách ở học kì hiện tại. Ngoài ra giáo viên còn có thể nhập điểm của môn học đó cho từng sinh viên.
  + Xem phúc khảo
    - Cho phép giáo viên xem lại và chấm lại điểm khi có sinh viên yêu cầu phúc khảo lại điểm số của mình
  + Thống kê
    - Cho phép giáo viên thông kê sinh viên và điểm số của các sinh viên theo từng môn học mà giáo viên đó đang dạng dạy.
* Sinh viên
  + Đăng ký môn học
    - Cho phép sinh viên xem danh sách các môn học và có thể đăng ký môn học ở học kì hiện tại.
  + Phúc khảo
    - Cho phép sinh viên xem lại điểm số của của mình theo từng môn học, có thể yêu cầu lại giáo viên phúc khảo lại điểm số nếu có vấn đề trong quá trình làm bài.

## *3.2 Thiết kế thành phần dữ liệu.*

### Mô hình dữ liệu mức quan niệm (Conceptual Data Model - CDM)



Hình 3. Mô hình dữ liệu mức quan niệm (Conceptual Data Model - CDM)

### Mô hình dữ liệu mức luận lý (Logical Data Model - LDM)



Hình 4. Mô hình dữ liệu mức luận lý (Logical Data Model - LDM)

### Mô hình dữ liệu mức vật lý (Physical Data Model - PDM)



Hình 5. Mô hình dữ liệu mức vật lý (Physical Data Model - PDM)

## *3.3 Danh sách các thực thể*

Bảng 1. SINH\_VIEN

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **TT** | **Tên** | **Kiểu** | **Kích thước** | **Khóa chính** | **Duy nhất** | **Not Null** | **RBTV Khóa Ngoại** | **Diễn Giải** |
| 1 | sv\_id | Integer |  | ✓ | ✓ | ✓ |  | Sinh viên id |
| 2 | sv\_ma | Varchar |  |  |  | ✓ |  | Mã sinh viên |
| 3 | sv\_ten | Varchar |  |  |  | ✓ |  | Tên sinh viên |
| 4 | sv\_namsinh | Varchar |  |  |  |  |  | Năm sinh sinh viên |
| 5 | sv\_email | Varchar |  |  |  |  |  | Email sinh viên |

Bảng 2. LOP

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **TT** | **Tên** | **Kiểu** | **Kích thước** | **Khóa chính** | **Duy nhất** | **Not Null** | **RBTV Khóa Ngoại** | **Diễn Giải** |
| 1 | l\_id | Integer |  | ✓ | ✓ | ✓ |  | Lớp id |
| 2 | l\_ten | Varchar |  |  |  | ✓ |  | Tên lớp |

Bảng 3. PHUC\_KHAO

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **TT** | **Tên** | **Kiểu** | **Kích thước** | **Khóa chính** | **Duy nhất** | **Not Null** | **RBTV Khóa Ngoại** | **Diễn Giải** |
| 1 | pk\_id | Integer |  | ✓ | ✓ | ✓ |  | Phúc khảo id |
| 2 | pk\_noidung | Varchar |  |  |  | ✓ | ✓ | Nội dung phúc khảo |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **TT** | **Tên** | **Kiểu** | **Kích thước** | **Khóa chính** | **Duy nhất** | **Not Null** | **RBTV Khóa Ngoại** | **Diễn Giải** |
| 1 | mhsv\_id | Integer |  | ✓ | ✓ | ✓ |  | Môn học sinh viên id |
| 2 | mhsv\_diem1 | Float |  |  |  | ✓ |  | Môn học sinh viên điểm giữa kì |
| 3 | mhsv\_diem2 | Float |  |  |  | ✓ |  | Môn học sinh viên điểm cuối kì |
| 4 | mhsv\_diem\_pk1 | Float |  |  |  | ✓ |  | Môn học sinh viên điểm phúc khảo |
| 5 | mhsv\_diem\_pk2 | Float |  |  |  | ✓ |  | Môn học sinh viên điểm phúc khảo |
| 6 | mhsv\_diemtong | Float |  |  |  | ✓ |  | Môn học sinh viên điểm tổng |
| 7 | mhsv\_diemchu | Varchar |  |  |  | ✓ |  | Môn học sinh viên điểm chữ |
| 8 | sv\_id | Integer |  |  |  |  | ✓ | Id sinh viên |
| 9 | pk\_id | Integer |  |  |  |  | ✓ | Phúc khảo id |

Bảng 4. MON\_HOC\_SINH\_VIEN

Bảng 5. NAM\_HOC

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **TT** | **Tên** | **Kiểu** | **Kích thước** | **Khóa chính** | **Duy nhất** | **Not Null** | **RBTV Khóa Ngoại** | **Diễn Giải** |
| 1 | nh\_id | Integer |  | ✓ | ✓ | ✓ |  | Id năm học |
| 2 | nh\_ten |  |  |  |  | ✓ |  | Tên năm học |

Bảng 6. MON\_HOC

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **TT** | **Tên** | **Kiểu** | **Kích thước** | **Khóa chính** | **Duy nhất** | **Not Null** | **RBTV Khóa Ngoại** | **Diễn Giải** |
| 1 | mh\_id | Integer |  | ✓ | ✓ | ✓ |  | Id môn học |
| 2 | mh\_ten | Varchar |  |  |  | ✓ |  | Tên môn học |
| 3 | nh\_id | Integer |  |  |  | ✓ | ✓ | Id năm học |
| 4 | mh\_ma | Varchar |  |  |  |  | ✓ | Mã môn học |
| 5 | hk\_id | Integer |  |  |  |  | ✓ | Id học kì |
| 6 | gv\_id | Integer |  |  |  |  | ✓ | Id giáo viên |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **TT** | **Tên** | **Kiểu** | **Kích thước** | **Khóa chính** | **Duy nhất** | **Not Null** | **RBTV Khóa Ngoại** | **Diễn Giải** |
| 1 | gv\_id | Integer |  | ✓ | ✓ | ✓ |  | Id giáo viên |
| 2 | gv\_ten | Varchar |  |  |  | ✓ |  | Tên giáo viên |
| 3 | gv\_email | Varchar |  |  |  | ✓ |  | Email giáo viên |
| 4 | gv\_sdt | Varchar |  |  |  |  |  | Số điện thoại giáo viên |
| 5 | gv\_diachi | Varchar |  |  |  |  |  | Địa chỉ giáo viên |

Bảng 7. GIAO\_VIEN

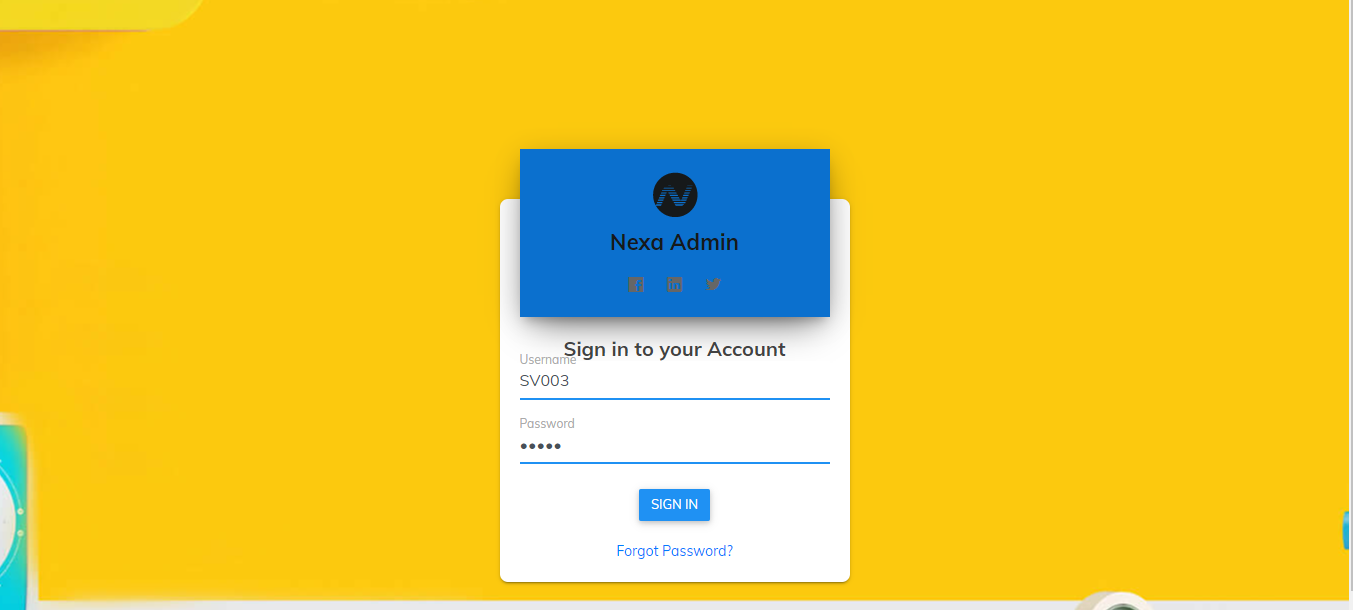
Bảng 8. HOC\_KI

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **TT** | **Tên** | **Kiểu** | **Kích thước** | **Khóa chính** | **Duy nhất** | **Not Null** | **RBTV Khóa Ngoại** | **Diễn Giải** |
| 1 | hk\_id | Integer |  | ✓ | ✓ | ✓ |  | Id học kì |
| 2 | hk\_ten | Varchar |  |  |  | ✓ |  | Tên học kì |

# **CHƯƠNG 4: GIỚI THIỆU TRANG WEB**

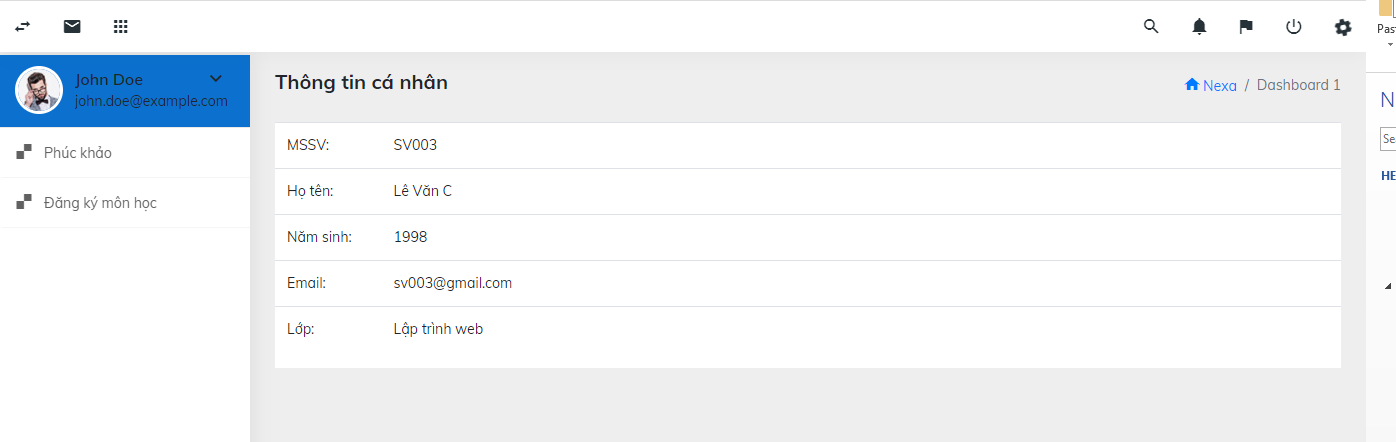
## *Giao diện người dùng là Sinh viên:*

Giao diện đăng nhập của sinh viên

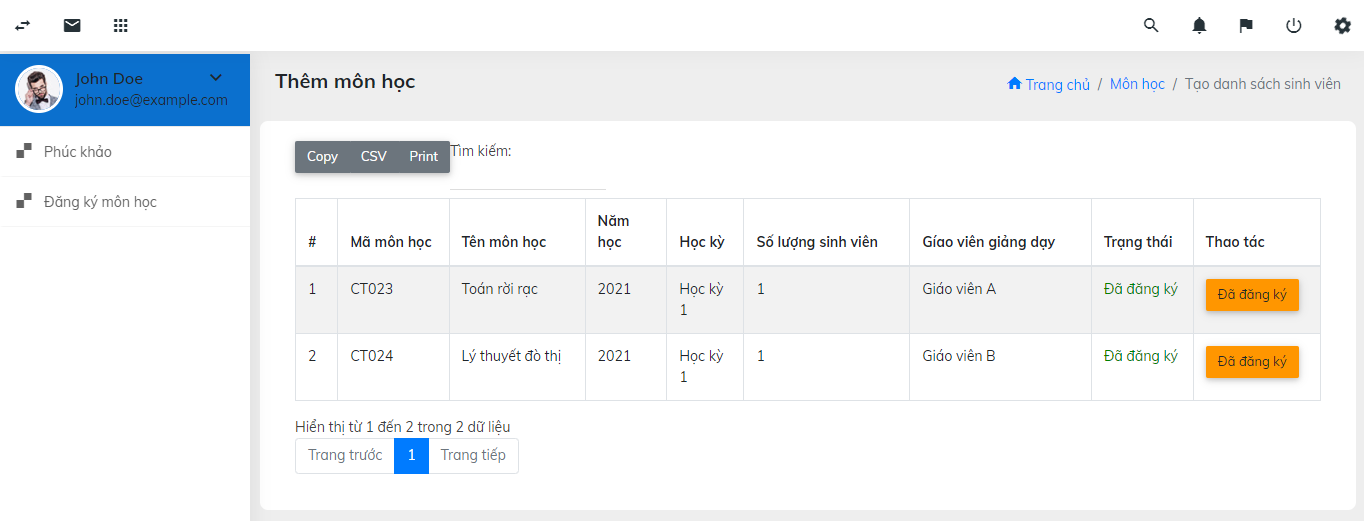


Hình 6. Giao diện đăng nhập

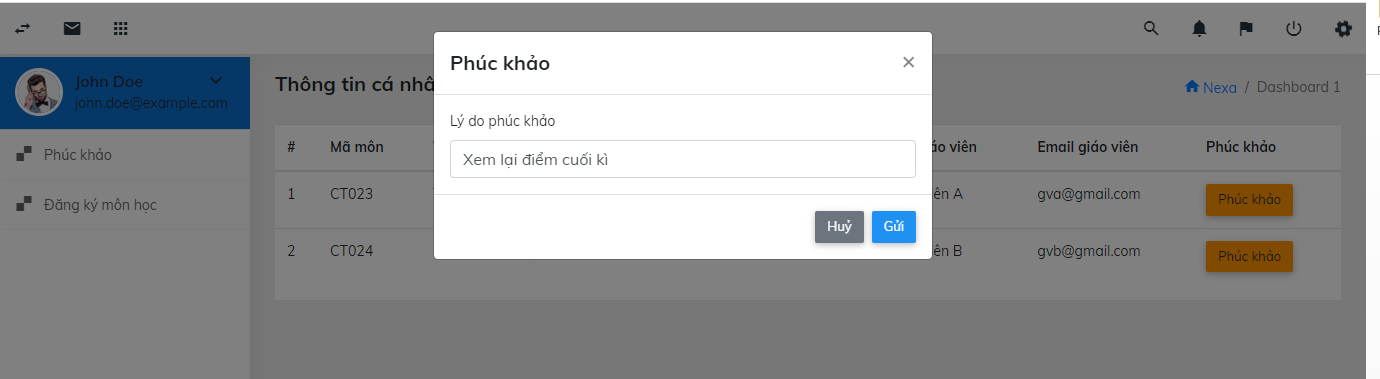
Hiện thị thông tin cá nhân của sinh viên



Hình 7. Giao diện thông tin cá nhân

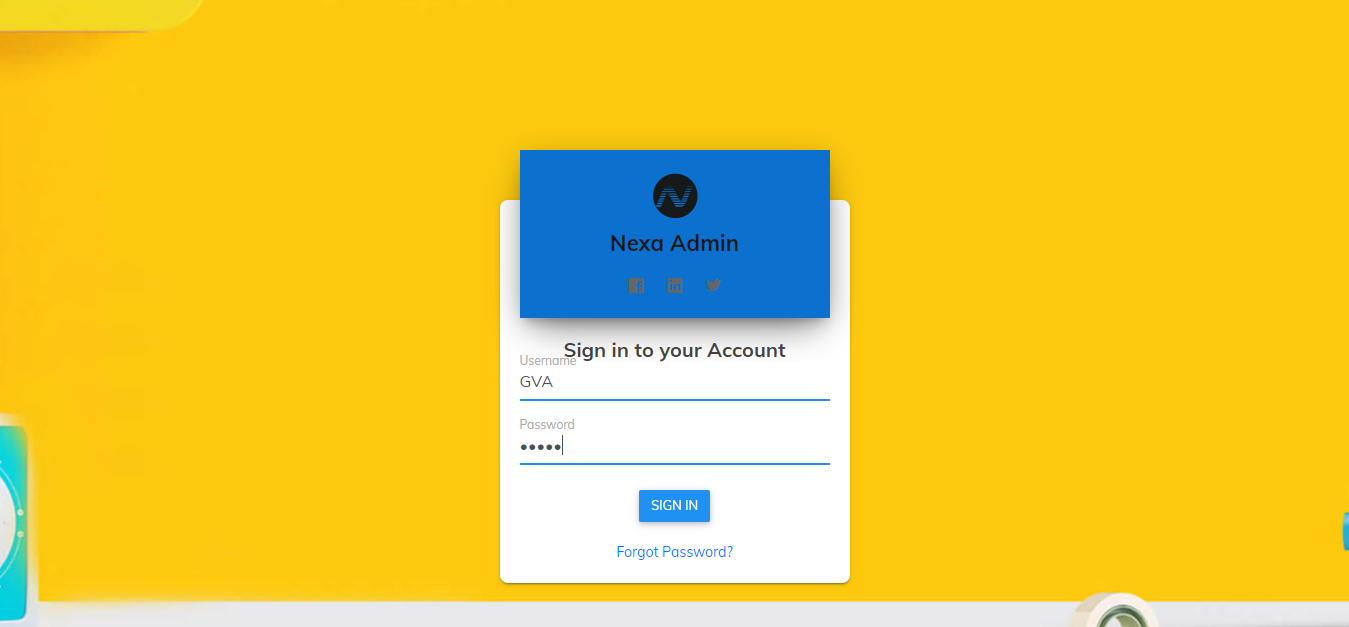


Hình 8. Giao diện đăng kí môn học

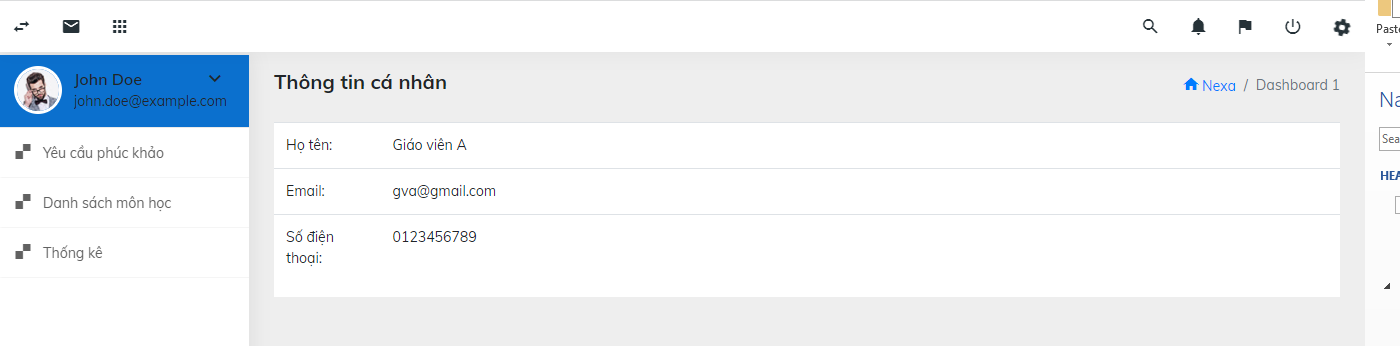


Hình 9. Giao diện yêu cầu phúc khảo

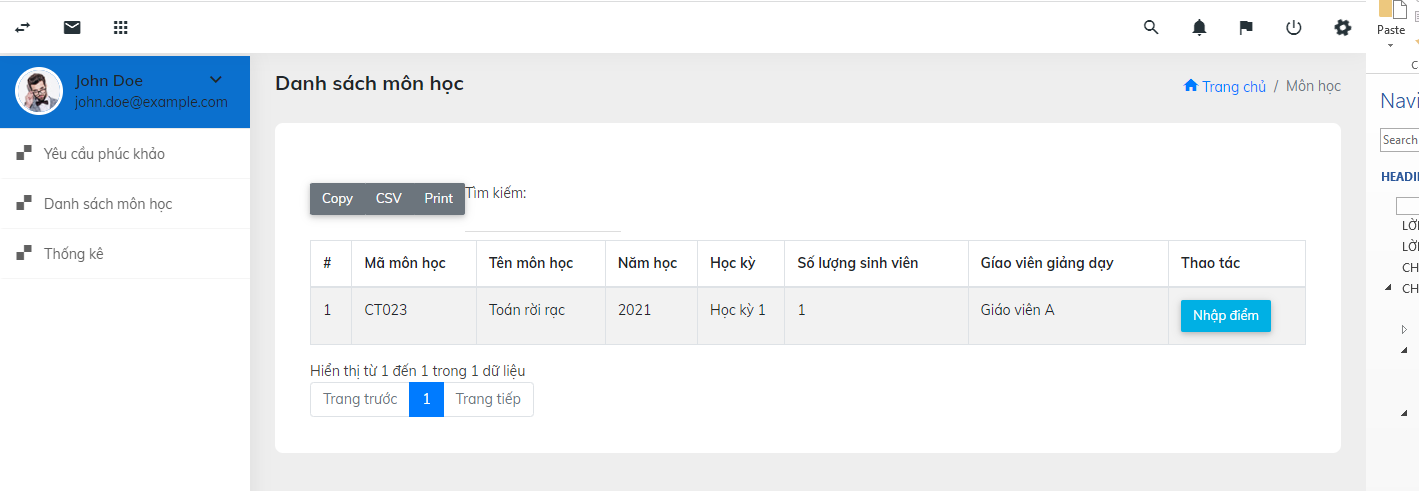
## *Giao diện người dùng là Giáo viên:*



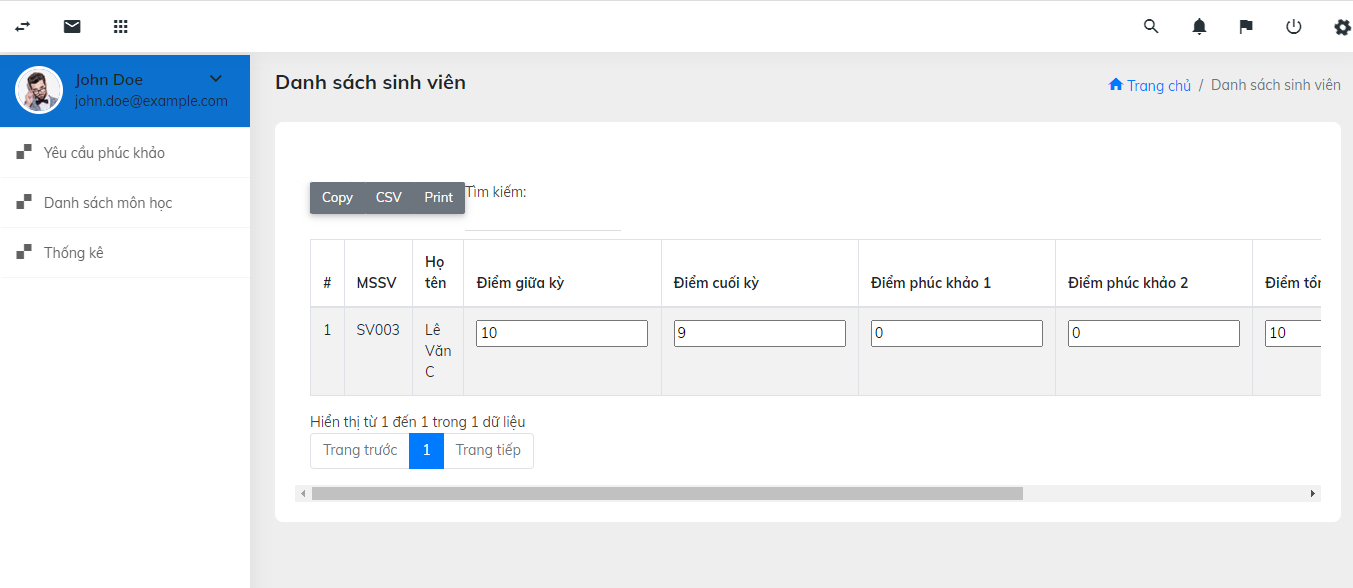
Hình 10. Giao diện đang nhập giáo viên



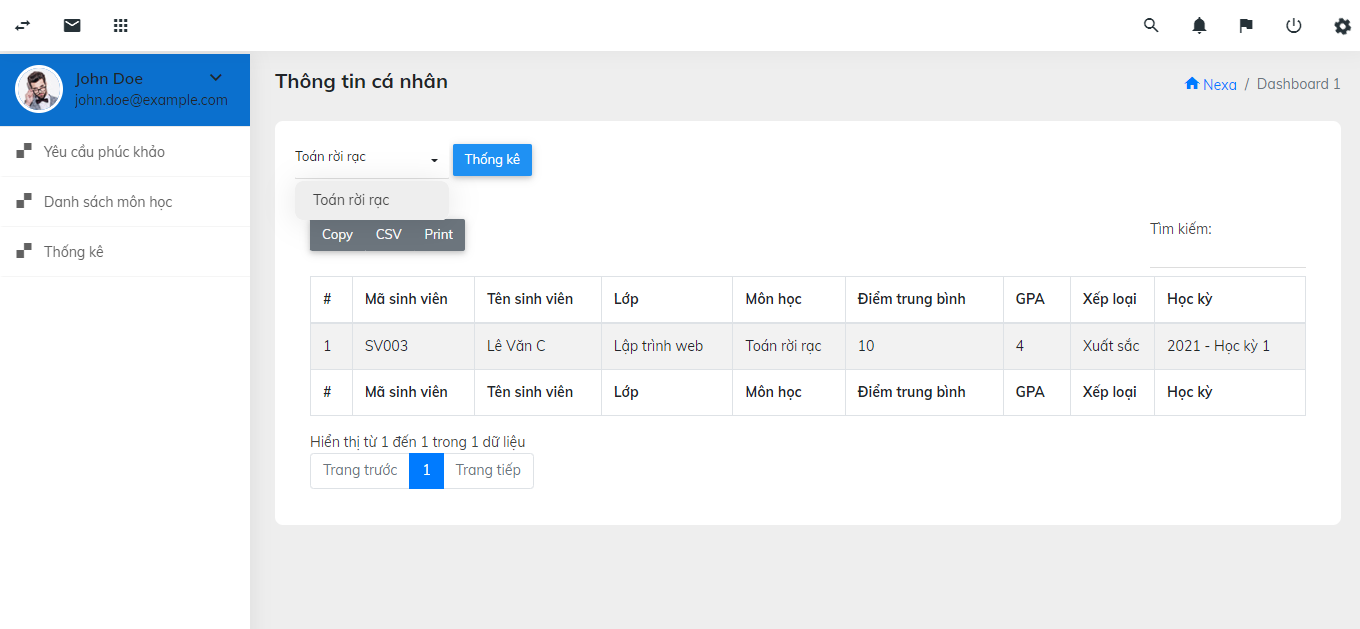
Hình 11. Giao diện thông tin giáo viên



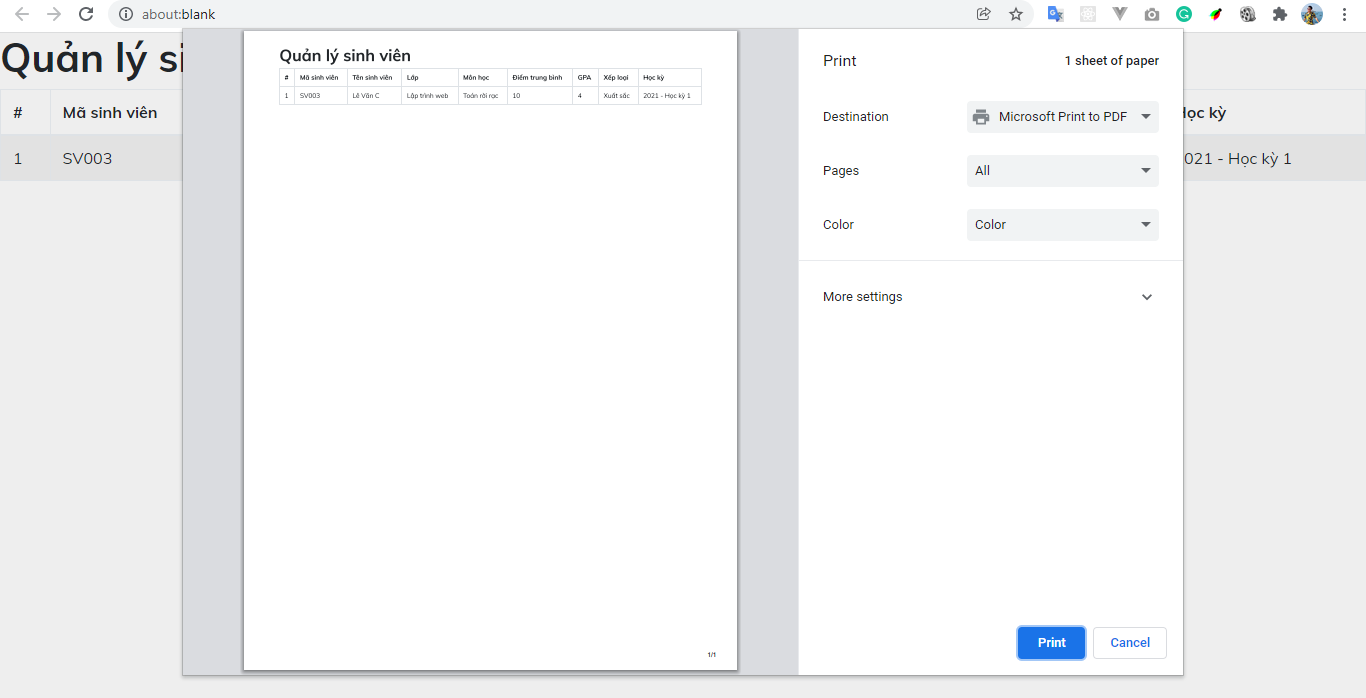
Hình 12. Giao diện danh sách môn học của sinh viên



Hình 13. Giao diện nhập điểm của giáo viên

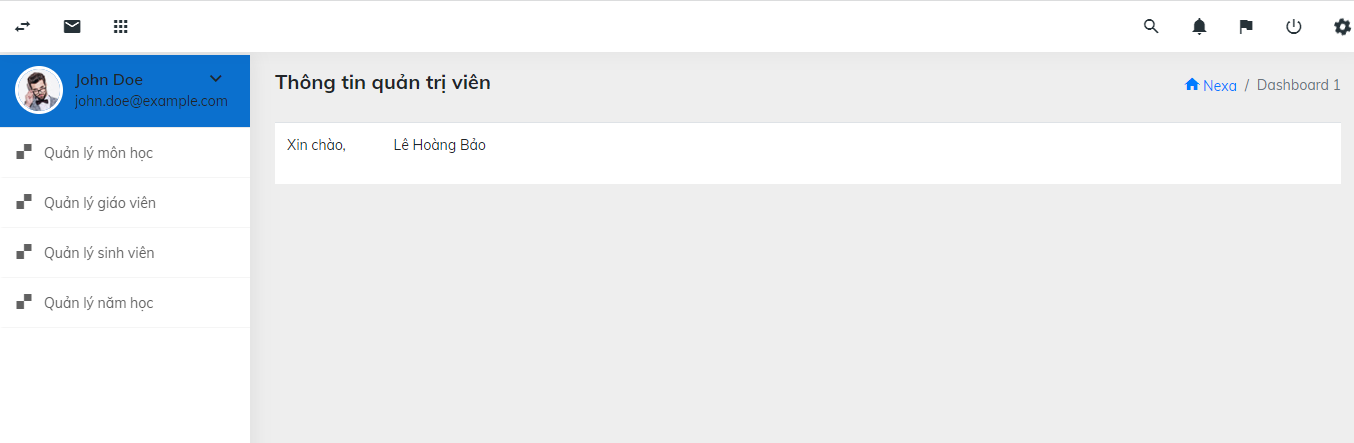


Hình 14. Giao diện thống kê của giáo viên

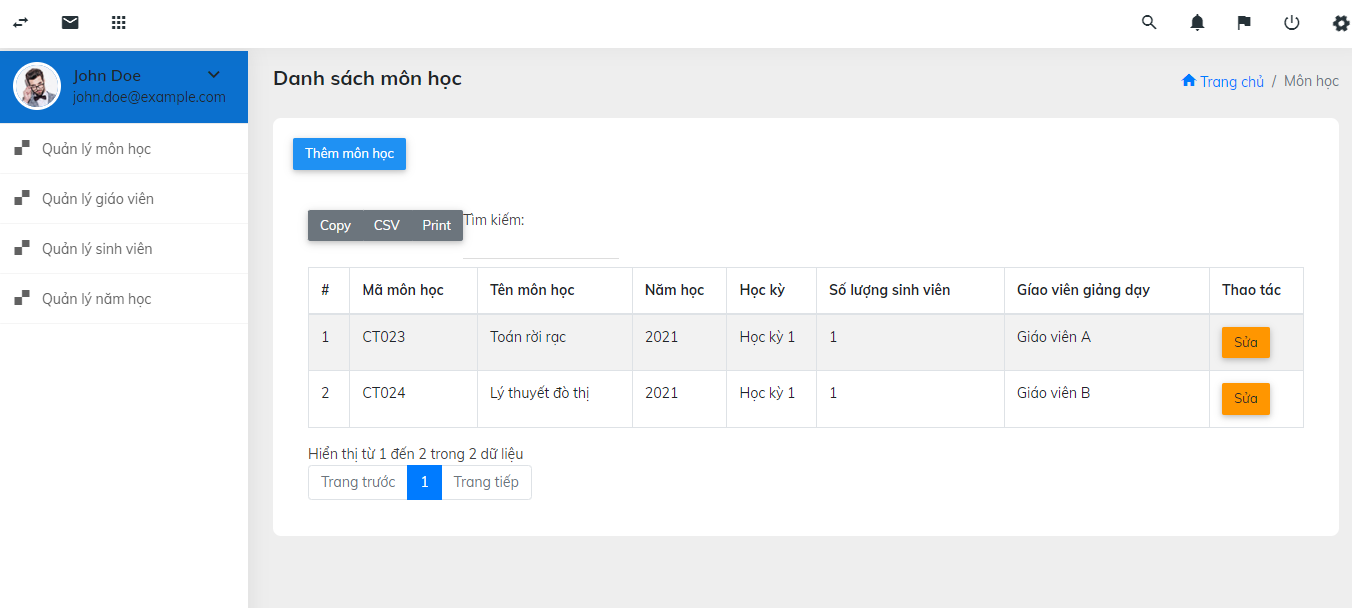


Hình 15. Giao diện in thông tin của giáo viên

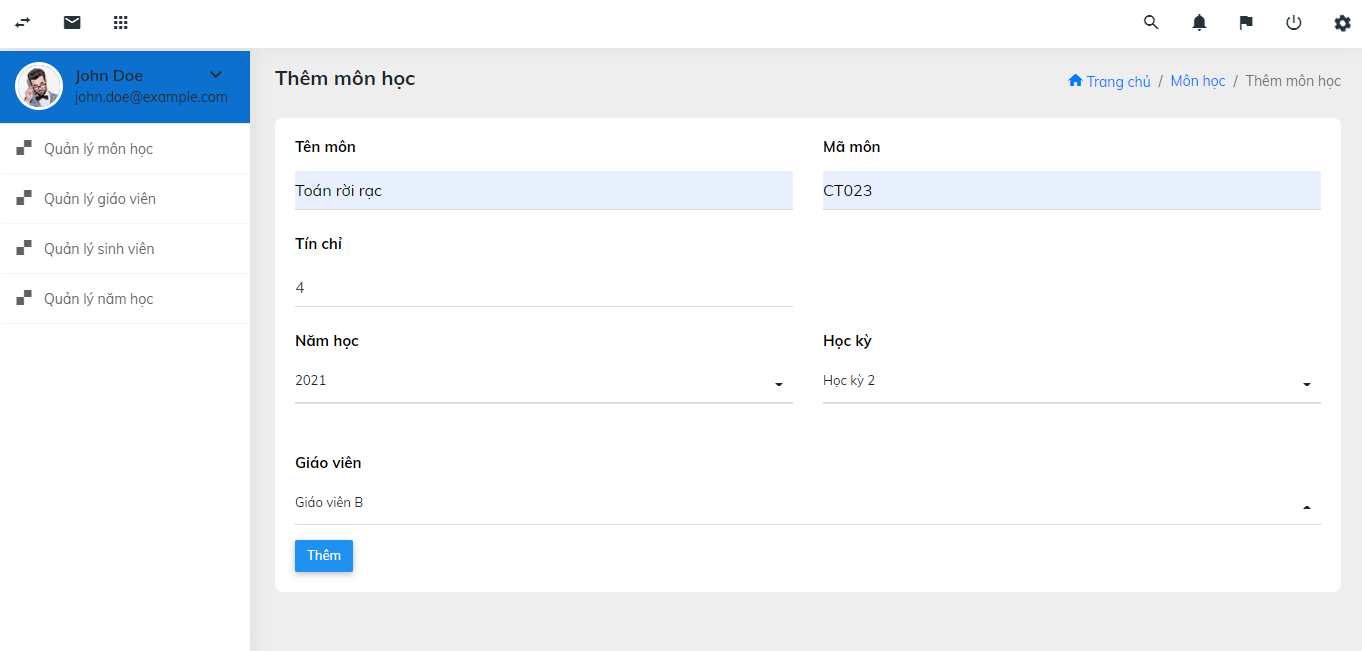
## *Giao diện người dùng là Admin:*



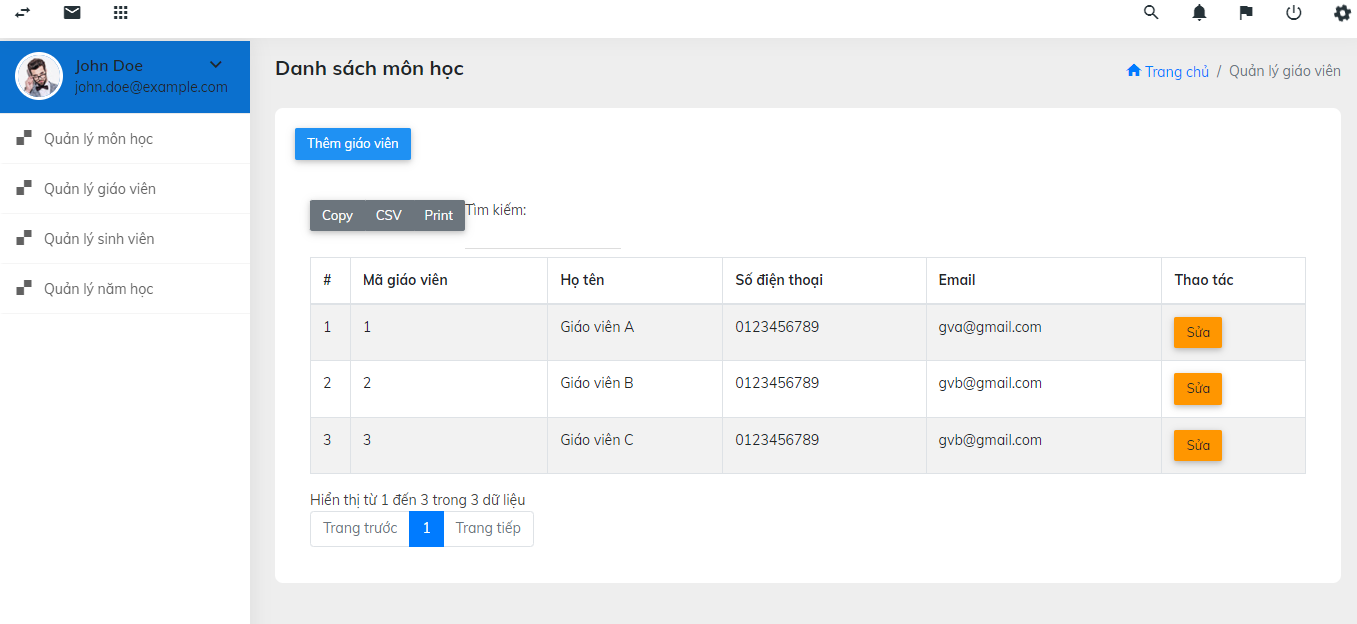
Hình 16. Giao diện thông tin Admin



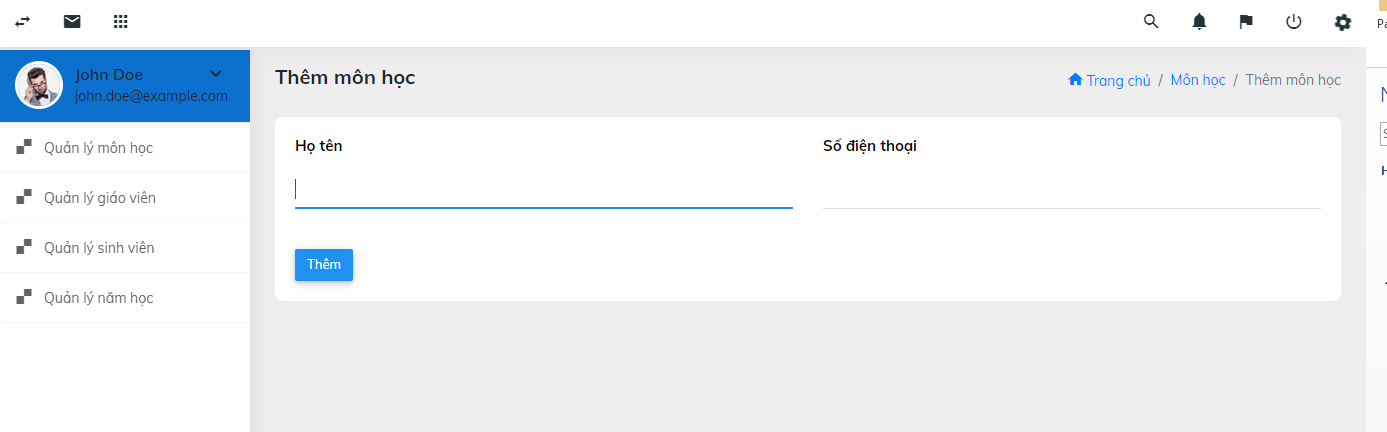
Hình 17. Giao diện thêm môn học



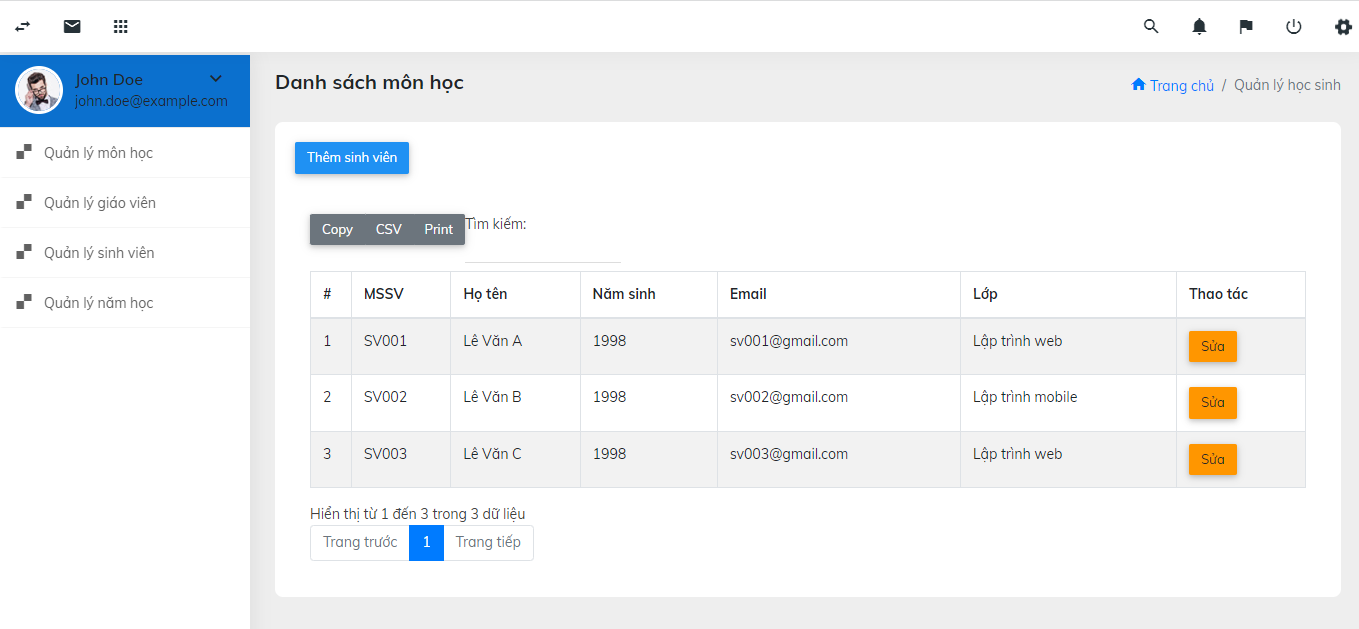
Hình 18. Giao diện thêm môn học



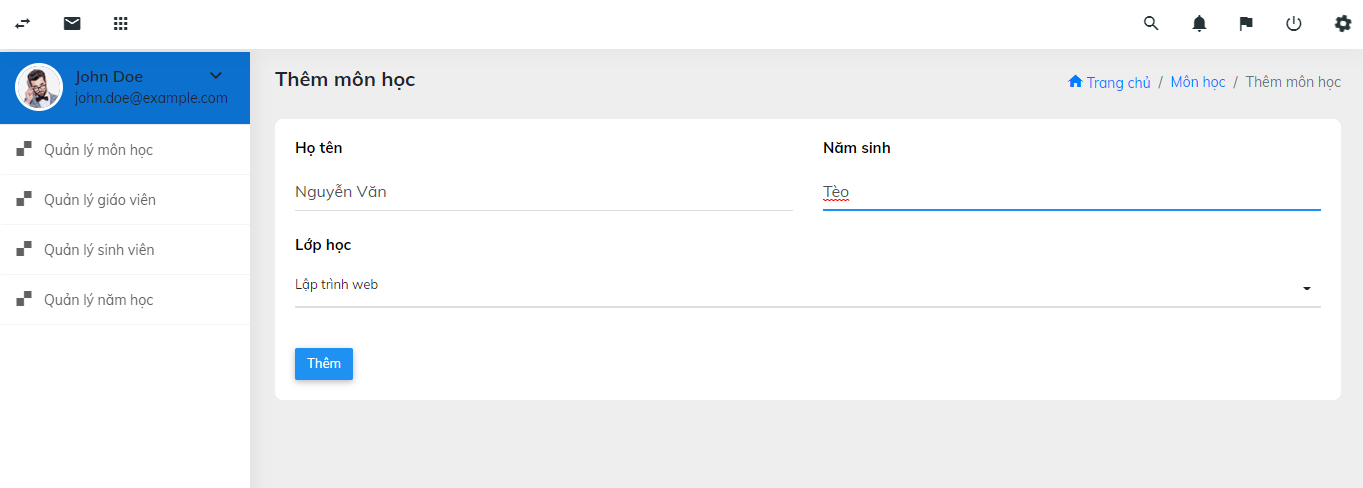
Hình 19. Giao diện thêm giáo viên



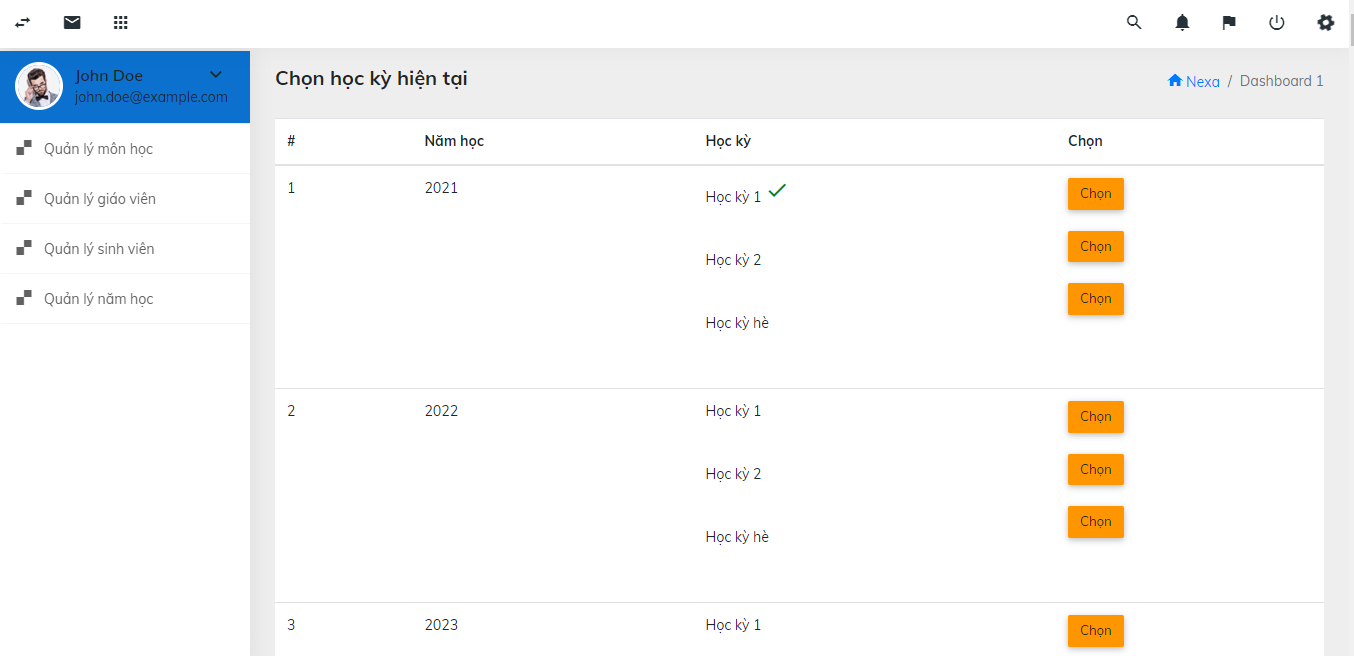
Hình 20. Giao diện thêm giáo viên



Hình 21. Giao diện thêm sinh viên



Hình 22. Giao diện thêm sinh viên

****

Hình 23. Giao diện thêm thêm học kì

# **TÀI LIỆU THAM KHẢO**

[8] W3school.com. [Online] Available at: *https://www.w3schools.com*

[9] Laravel. [Online] Available at: *https://laravel.com*

[10] Larval-dompdf. [Online] Available at: *https://github.com/barryvdh/laravel-dompdf*

[11] StackOverFlow. [Online] Available at: *https://stackoverflow.com*

[12] Google cloud platform. [Online] Available at: *https://cloud.google.com/*

[13] Viblo. [Online] Available at: *https://viblo.asia*

[22] StackOverFlow. [Online] Available at: *https://stackoverflow.com*