# MỤC LỤC

Contents

[MỤC LỤC 1](#_Toc485830789)

[Tóm tắt luận văn 1](#_Toc485830790)

[Lời cảm ơn 3](#_Toc485830791)

[Lời cam đoan 4](#_Toc485830792)

[Danh mục các hình vẽ đồ thị 5](#_Toc485830793)

[PHẦN MỞ ĐẦU 6](#_Toc485830794)

[1. Lý do chọn đề tài 6](#_Toc485830795)

[**2.** **Mục tiêu của đề tài** 6](#_Toc485830796)

[**3.** **Giới hạn và phạm vi của đề tài** 6](#_Toc485830797)

[**4.** **Kết quả dự kiến đạt được** 7](#_Toc485830798)

[NỘI DUNG 7](#_Toc485830799)

[CHƯƠNG 1: Cơ sở lý thuyết 7](#_Toc485830800)

[**1.1.** **Codeigniter Framework** 7](#_Toc485830801)

[**1.1.1.** **Giới thiệu Codeigniter Framework** 7](#_Toc485830802)

[**1.1.2.** **Cài đặt CodeIgniter Framework** 9](#_Toc485830803)

[**1.2.** **Mysql** 12](#_Toc485830804)

[**1.2.1.** **Giới thiệu Mysql** 12](#_Toc485830805)

[**1.2.2.** **Cài đặt MySQL** 14](#_Toc485830806)

[**1.2.3.** **Sử dụng** 15](#_Toc485830807)

[CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 22](#_Toc485830808)

[**2.1.** **Khảo sát** 22](#_Toc485830809)

[**2.1.1.** **Hiện trạng thị trường việc làm hiện nay** 22](#_Toc485830810)

[**2.1.2.** **Mục đích và phương pháp giải quyết** 23](#_Toc485830811)

[**2.2.** **Phân tích yêu cầu hệ thống** 25](#_Toc485830812)

[**2.2.1.** **Yêu cầu chức năng nghiệp vụ** 25](#_Toc485830813)

[**2.2.2.** **Yêu cầu chức năng hệ thống** 25](#_Toc485830814)

[**2.2.3.** **Yêu cầu phi chức năng** 25](#_Toc485830815)

[**2.2.4.** **Dữ liệu đưa vào hệ thống quản lý** 26](#_Toc485830816)

[**2.3.** **Phân tích UML** 26](#_Toc485830817)

[**2.3.1.** **Use case diagram.** 26](#_Toc485830818)

[**2.3.2.** **Class diagram** 44](#_Toc485830819)

[**2.4.** **Cơ sở dữ liệu** 44](#_Toc485830820)

[CHƯƠNG 3: WEBSITE TUYỂN DỤNG VÀ TÌM VIỆC LÀM 50](#_Toc485830821)

[**3.1.** **Trang chủ** 50](#_Toc485830822)

[**3.2.** **Trang nhà tuyển dụng** 51](#_Toc485830823)

[**3.3.** **Trang người tìm việc** 53](#_Toc485830824)

[**3.4.** **Trang quản trị** 54](#_Toc485830825)

[KẾT LUẬN VÀ KIẾN NGHỊ 54](#_Toc485830826)

[A. Kết luận 54](#_Toc485830827)

[B. Kiến nghị 57](#_Toc485830828)

[DANH MỤC TÀI LIỆU THAM KHẢO 58](#_Toc485830829)

# Tóm tắt luận văn

Họ và tên học viên: **Trần Văn Sang**

Chuyên ngành: Công Nghệ Thông Tin

Cán bộ hướng dẫn: Ths. Đỗ Thanh Long

Tên đề tài: Xây dựng website tuyển dụng và tìm việc làm.

Tóm tắt: Hiện nay Internet là một công cụ, một môi trường làm việc và liên kết mọi người trên thế giới lại với nhau. Internet có mặt khắp nơi và hỗ trợ con người vào nhiều lĩnh vực. Tìm kiếm việc làm và ứng viên trên Internet là một loại hình được đa số người tìm việc và nhà tuyển dụng quan tâm, nó vừa đỡ mất thời gian và tiền bạc. Khi một công ty muốn tuyển dụng cho công ty mình một vị trí nào đó có lẽ việc liên lạc đến các công ty cung cấp nhân lực, hay nói nôm na là công ty săn đầu người chuyên cung cấp những ứng viên đủ trình độ đáp ứng yêu cầu công ty đặt ra. Hoặc có thể đăng lên báo đài để cho mọi người biết đến. Khi mội ứng viên muốn tìm việc làm họ thường lục tìm các thông tin tuyển dụng trên báo, hoặc họ đi đến một trung tâm giới thiệu việc làm, hoặc tham gia các cuộc hội thảo, hội chợ việc làm để kiếm việc làm cho mình. Tìm kiếm ứng viên cho nhà tuyển dụng và việc làm cho ứng viên trên mạng Internet hiện nay là một mô hình không còn mới mẻ tại Việt Nam. Vấn đề chúng ta đặt ra là làm thế nào để hỗ trợ ứng viên tìm việc và hỗ trợ nhà tuyển dụng tìm ứng viên vừa ý. Trong xu hướng phát triển hiện nay, Internet đã không còn xa lạ với rất nhiều người, rất nhiều thành phần khác nhau trong xã hội. Nó là một công cụ, một môi trường làm việc và liên kết mọi người trên thế giới lại với nhau. Internet có mặt khắp nơi và hỗ trợ con người vào nhiều lĩnh vực. Chính khả năng mạnh mẽ đó đã đem đến khả năng tiềm tàng cho việc xây dựng một hệ thống ”XÂY DỰNG WEBSITE TUYỂN DỤNG VÀ TÌM VIỆC LÀM”. Không còn quá xa lạ nhưng cũng không thật chắc chắn, rất nhiều công ty, doanh nghiệp có cung cấp các dịch vụ việc làm đã tự xây dựng cho mình các website nhằm thực hiện chiến lược thương mại điện tử của mình. Tuy nhiên, rất nhiều website trong số này đã không mang lại hiệu quả đầu tư xứng đáng. Vậy vấn đề ở đâu? Trước hết, đó là sự thiếu hiệu quả trong công tác quản lý nội dung thông tin của website, thông tin không được cập nhật thường xuyên dẫn đến nội dung nghèo nàn và hậu quả của nó là không thu hút được sự truy cập của khách hàng. Thứ hai đó chính là khả năng quản lý và công tác quảng bá cho website của mình đối với nhiều doanh nghiệp vẫn là vấn đề nan giải do kinh nghiệm cũng như kỹ năng quản lý còn thiếu, yếu. Qua khảo sát thực tế cùng với mong muốn hỗ trợ các nhà cung cấp dịch vụ việc làm cũng như những người có nhu cầu sử dụng các dịch vụ việc làm một cách hiệu quả, tạo sự thuận tiện trong giao dịch cũng như giảm được sự đầu tư không hiệu quả cho các website của công ty, doanh nghiệp. Em đã cố gắng tìm hiểu và “XÂY ĐỰNG WEBSITE TUYỂN DỤNG VÀ TÌM VIỆC LÀM”. Mặc dù đây là một nội dung không mới và cũng đã có rất nhiều website hỗ trợ các dịch vụ này. Nhưng cách giải quyết vấn đề thì lại hoàn toàn mới bởi hệ thống được xây dựng trên nền tảng hỗ trợ đa người dùng với phương thức hoạt động của một mô hình cổng thông tin điện tử. Đây chính là điều mà nhiều website với phương thức hoạt động truyền thống chưa làm được. Luận văn gồm:

**Nội dung luận văn**

***Chương 1: Tổng quan***

Tìm hiểu về Codeigniter Framework, cách thức cài đặt và triển khai một website.

Tìm hiểu tổng quan về mysql.

***Chương 2: Phân tích và thiết kế hệ thống***

Khảo sát: khảo sát thị trường việc làm hiện nay, nêu ra mục đích và phương hướng giải quyết.

Phân tích UML: phân tich usecase, class diagram.

***Chương 3: Website tuyển dụng và tìm việc làm***

Trang chủ

Trang nhà tuyển dụng

Trang người tìm việc

Trang quản trị

# Lời cảm ơn

Trong thời gian nghiên cứu và thực hiện luận văn này, tôi đã nhận được sự giúp đỡ nhiệt tình của các cơ quan, các tổ chức và các cá nhân. Tôi xin bày tỏ lời cảm ơn sâu sắc nhất tới tất cả các tập thể, cá nhân đã tạo điều kiện giúp đỡ tôi trong suốt quá trình thực hiện nghiên cứu luận văn này.

Trước hết tôi xin trân trọng cảm ơn Ban giám hiệu trường Đại học Công Nghệ Giao Thông Vận Tải Hà Nội cùng các thầy cô giáo, những người đã trang bị kiến thức cho tôi trong suốt quá trình học tập.

Với lòng biết ơn chân thành và sâu sắc nhất, tôi xin trân trọng cảm ơn thầy giáo - Ths. Đỗ Thanh Long, người thầy đã trực tiếp chỉ bảo, hướng dẫn khoa học và giúp đỡ tôi trong suốt quá trình nghiên cứu, hoàn thành luận văn này.

Xin chân thành cảm ơn tất các bạn bè, đồng nghiệp đã động viên, giúp đỡ nhiệt tình và đóng góp nhiều ý kiến quý báu để tôi hoàn thành luận văn này.

Do thời gian nghiên cứu có hạn, luận văn của tôi chắc hẳn không thể tránh khỏi những sơ suất, thiếu sót, tôi rất mong nhận đuợc sự đóng góp của các thầy cô giáo cùng toàn thể bạn đọc.

Xin trân trọng cảm ơn!

Hà Nội, tháng 6 năm 2017

Tác giả luận văn

Trần Văn Sang

# 

# Lời cam đoan

Tôi xin cam đoan đây là công trình nghiên cứu của bản thân, được xuất phát từ yêu cầu phát sinh trong công việc để hình thành hướng nghiên cứu. Các số liệu có nguồn gốc rõ ràng tuân thủ đúng nguyên tắc và kết quả trình bày trong luận văn được thu thập trong quá trình nghiên cứu là trung thực chưa từng được ai công bố trước đây.

Tôi xin cam đoan rằng mọi sự giúp đỡ cho việc thực hiện luận văn đã được cám ơn, các thông tin trích dẫn trong luận văn này đều được chỉ rõ nguồn gốc.

Hà Nội, tháng 6 năm 2017

Tác giả luận văn

Trần Văn Sang

# Danh mục các hình vẽ đồ thị

[Hình 1. Cấu trúc thư mục Codeigniter Framework 10](#_Toc485830005)

[Hình 2. Cấu trúc thư mục Aplication 10](#_Toc485830006)

[Hình 3. Chạy thử chương trình codeigniter đầu tiên 11](#_Toc485830007)

[Hình 4. Chương trình chạy trên trình duyệt 12](#_Toc485830008)

[Hình 5. Mô hình usecase với nhóm chức năng thuộc nhà tuyển dụng 27](#_Toc485830009)

[Hình 6. Sequence diagram đăng nhập 28](#_Toc485830010)

[Hình 7. Sequence diagram đăng tin tuyển dụng 29](#_Toc485830011)

[Hình 8. Sequence diagram xóa tin tuyển dụng 30](#_Toc485830012)

[Hình 9. Sequence diagram chỉnh sửa thông tin 31](#_Toc485830013)

[Hình 10. Tìm kiếm hồ sơ xin việc 32](#_Toc485830014)

[Hình 11. Mô hình usecase với nhóm chức năng thuộc người tìm việc 32](#_Toc485830015)

[Hình 12. Mô hình Sequence diagram Đăng hồ sơ 34](#_Toc485830016)

[Hình 13. Sequence diagram Xóa hồ sơ 36](#_Toc485830017)

[Hình 14. Mô hình usecase đối với chức năng thuộc người quản trị 37](#_Toc485830018)

[Hình 15. Sequence diagram Xóa nhà tuyển dụng 39](#_Toc485830019)

[Hình 16. Sequence diagram Xóa người tìm việc 40](#_Toc485830020)

[Hình 17. Sequence diagram Tìm kiếm nhà tuyển dụng 41](#_Toc485830021)

[Hình 18. Mô hình - Usecase khách viếng thăm 41](#_Toc485830022)

[Hình 19. Mô hình usecase tổng quát 44](#_Toc485830023)

[Hình 20. Biểu đồ lớp 44](#_Toc485830024)

[Hình 21. Mô hình cơ sở dữ liệu quan hệ 50](#_Toc485830025)

[Hình 22. Đăng nhập 52](#_Toc485830026)

[Hình 23. Giao diện người dùng sau khi đăng nhập 52](#_Toc485830027)

[Hình 24. Giao diện đăng tin 53](#_Toc485830028)

[Hình 25. Giao diện trang người tìm việc 54](#_Toc485830029)

[Hình 26. Giao diện trang quản trị 54](#_Toc485830030)

# PHẦN MỞ ĐẦU

## Lý do chọn đề tài

Một trong những vấn đề thiết yếu của con người là về vấn đề công ăn việc làm. Và việc làm là nhu cầu sống còn của người lao động. Nhưng làm sao để người lao động có thể tìm được đúng công việc bản thân mong muốn, thích hợp với năng lực, khả năng của mình. Đây là vấn đề không dễ giải quyết cho mỗi người lao động. Thời đại phát triển và vấn đề sử dụng máy tính với con người trở lên phổ cập. Họ tìm kiếm được các thông tin sức khỏe, giáo dục,… trên máy tính. Việc giúp người lao động có thể dễ dàng tìm được việc cho mình tốt hơn, chính xác hơn cũng là vấn đề rất nhiều người quan tâm.

Vì lý do này, đề tài “**Xây dựng website tuyển dụng và tìm việc làm**” đã là đề tài mà tôi lựa chọn làm đồ án tốt nghiệp. Website được phát triển thành công có thể đưa vào sử dụng ở các trung tâm trợ giúp tìm việc làm cho người lao động. Với sự trợ giúp của website này, các trung tâm tìm việc làm dễ dàng đảm nhận được trọng trách của mình và giảm được công sức lao động và chi phí. Người lao động và các tổ chức cần tuyển người cũng sẽ được hưởng lợi từ nó: tìm được việc làm và tuyển được người nhanh chóng và tiện lợi.

1. **Mục tiêu của đề tài**

***-*** Mục tiêu phát triển của đề tài là cung cấp các thông tin cần thiết và chi tiết của các ứng viên, đáp ứng nhu cầu tuyển dụng của các công ty, giúp nhà tuyển dụng chọn được các ứng viên xuất sắc cũng như giúp cho các ứng viên có thể tìm được một công việc phù hợp với mình trong thời gian nhanh nhất và hiệu quả nhất.

* Khảo sát được nhu cầu tìm người của nhà tuyển dụng, nhu cầu công việc của người tìm việc
* Phân tích hệ thống chương trình bao gồm giao diện, chức năng, cơ sở dữ liệu, nhằm đáp ứng nhu cầu người dùng

1. **Giới hạn và phạm vi của đề tài**

* Phân tích đối tượng sử dụng, chức năng người dùng
* Xây dựng website dựa vào các thao tác với HTML, PHP, MYSQL
* Sử dụng Framework codeigniter
* Website tập chung chức năng với người dùng là quản trị viên, nhà tuyển dụng và người tìm việc

1. **Kết quả dự kiến đạt được**

* Xây dựng website triển khai trên server
* Xây dựng website hoàn chỉnh đầy đủ các chức năng: Quản lý người tìm việc, quản lý doanh nghiệp, quản lý nhân viên, đăng tin tìm việc, đăng tin tuyển dụng,...
* Bổ sung thêm kiến thức về lập trình web

# NỘI DUNG

# CHƯƠNG 1: Cơ sở lý thuyết

* 1. **Codeigniter Framework**
     1. **Giới thiệu Codeigniter Framework**

**CodeIgniter** (CI) là một PHP Framework theo cấu trúc MVC được viết trên PHP4 và hỗ trợ cả PHP4 và PHP5. Nó là một tập hợp các thư viện viết sẵn trên PHP giúp chúng ta phát triển web bằng PHP nhanh hơn là cách viết lắp ghép thông thường:

* Các thư viện này giúp chúng ta thực hiện các tác vụ thông thường của một ứng dụng web như kết nối và thực hiện các công việc liên quan đến cơ sở dữ liệu , upload file, xử lý hình ảnh, phân trang, cookie, session, bảo mật … Đồng thời các thư viện này cũng giúp chúng ta tổ chức code tốt hơn với mô hình MVC.
* Để có thể học được CodeIgniter framework thì bắt buộc các bạn cần có kiến thức sau:

+ Lập trình hướng đối tượng ([OOP](http://hocphp.info/lap-trinh-huong-doi-tuong-oop/))

+ Mô hình theo đúng chuẩn [MVC](http://hocphp.info/mo-hinh-mvc/)

+ Làm việc và xử lý [mảng](http://hocphp.info/mang-trong-php/)thật tốt

Ngoài ta thì còn rất nhiều yếu tố khác nữa nhưng 3 yêu tố trên là 3 yếu tố bắt buộc để các bạn có thể dễ dàng tiếp cận với bất kỳ PHP framework nào

**Tại sao lại chọn CodeIgniter framework?**

+ Là một trong những PHP Framework có số lượng người dùng nhiều nhất hiện nay

+ Tính tiếp cận và tài liệu dễ học, dễ mở rộng

+ Bộ core thư viện của Codeigniter ít thay đổi

+ Codeigniter Framework có vòng đời ổn định hơn so với các PHP Framework khác

+ Dễ tích hợp thêm các thư viện từ framework khác

**Ưu điểm CodeIgniter framework:**

***Được thiết kế theo mô hình Model-View-Controller***

Mô hình MVC giúp tách thành phần hiển thị giao diện (presentation) và xử lý (business logic) của một phần mềm thành những thành phần độc lập, từ đó giúp cho việc thiết kế, xử lý và bảo trì mã nguồn dễ dàng, đồng thời tăng khả năng mở rộng của phần mềm. CodeIgniter vận dụng mô hình này trong thiết kế, giúp tách biệt các tập tin giao diện với các tập tin xử lý dữ liệu, nâng cao khả năng quản lý và dễ bảo trì. Nhỏ gọn:

Gói cài đặt chỉ 404KB (không bao gồm phần User Guide). So với các PHP framework khác như CakePHP (1.3MB), Symfony (5.08MB) hay Zend Framework (5.66MB)…kích thước của CodeIgniter giúp giảm thiểu đáng kể không gian lưu trữ.

*Tốc độ nhanh*

CodeIgniter được đánh giá là một PHP framework có tốc độ nhanh nhất hiện nay. Bằng cơ chế lưu nội dung vào bộ đệm (cache), kiểm tra bộ đệm trước khi tiến hành thực hiện yêu cầu, CodeIgniter giảm số lần truy cập và xử lý dữ liệu, từ đó tối ưu hóa tốc độ tải trang.

*Miễn phí*

CodeIgniter được phát hành dưới giấy phép Apache/BSD mở rộng, cho phép người dùng tự do thay đổi, phát triển và phân phối mã nguồn. Hỗ trợ Search Engine Optimization: Cấu trúc URL của CodeIgniter rất thân thiện với các robot tìm kiếm.

*Hệ thống thư viện phong phú*

CodeIgniter cung cấp các thư viện phục vụ cho những tác vụ thường gặp nhất trong lập trình web, chẳng hạn như truy cập cơ sở dữ liệu, gửi email, kiểm tra dữ liệu, quản lý session, xử lý ảnh…đến những chức năng nâng cao như XML-RPC, mã hóa, bảo mật…

*Bảo mật hệ thống*

Cơ chế kiểm tra dữ liệu chặt chẽ, ngăn ngừa XSS và SQL Injection của CodeIgniter giúp giảm thiểu các nguy cơ bảo mật cho hệ thống.

**Nhược điểm**

*Chưa hỗ trợ Object Relational Mapping*

Object Relational Mapping (ORM) [25] là một kỹ thuật lập trình, trong đó các bảng của cơ sở dữ liệu được ánh xạ thành các đối tượng trong chương trình. Kỹ thuật này giúp cho việc thực hiện các thao tác trong cơ sở dữ liệu (Create Read Update Delete – CRUD) dễ dàng, mã nguồn ngắn gọn hơn. Hiện tại, CodeIgniter vẫn chưa hỗ trợ ORM.

*Chưa hỗ trợ AJAX*

AJAX (Asynchronous JavaScript and XML) [26] đã trở thành một phần không thể thiếu trong bất kỳ ứng dụng Web 2.0 nào. AJAX giúp nâng cao tính tương tác giữa người dùng và hệ thống, giúp cho người dùng có cảm giác như đang sử dụng ứng dụng desktop vì các thao tác đều diễn ra “tức thời”. Hiện tại, CodeIgniter vẫn chưa có thư viện dựng sẵn nào để hỗ trợ xây dựng ứng dụng AJAX. Lập trình viên phải sử dụng các thư viện bên ngoài, như jQuery, Script.aculo.us, Prototype hay Mootools…

*Chưa hỗ trợ một số module thông dụng*

So sánh với framework khác, CodeIgniter không có các module thực thi một số tác vụ thường gặp trong quá trình xây dựng ứng dụng web như Chứng thực người dùng (User Authorization), trình phân tích RSS (RSS Parser) hay trình xử lý PDF…

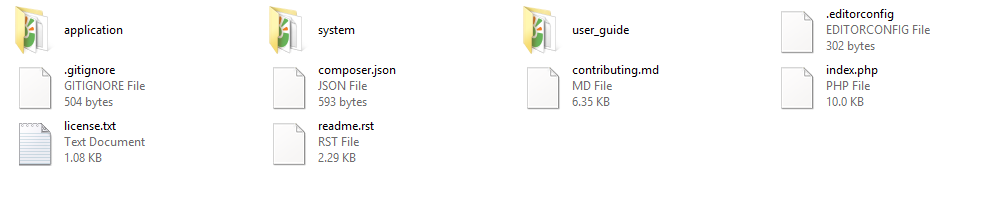
*Chưa hỗ trợ Event-Driven Programming*

Event-Driven Programming (EDP) là một nguyên lý lập trình, trong đó các luồng xử lý của hệ thống sẽ dựa vào các sự kiện, chẳng hạn như click chuột, gõ bàn phím không phải là một khuyết điểm to lớn của CodeIgniter vì hiện tại, chỉ có một số ít framework hỗ trợ EDP, bao gồm Prado, QPHP và Yii.

* + 1. **Cài đặt CodeIgniter Framework**

– Đầu tiên các bạn download phiên bản mới nhất của CI tại đây,hiện tại mình đang sử dụng phiên bản 3.1.4

– Sau khi download về các bạn giải nén và sẽ thấy các thư mục bên trong như sau:

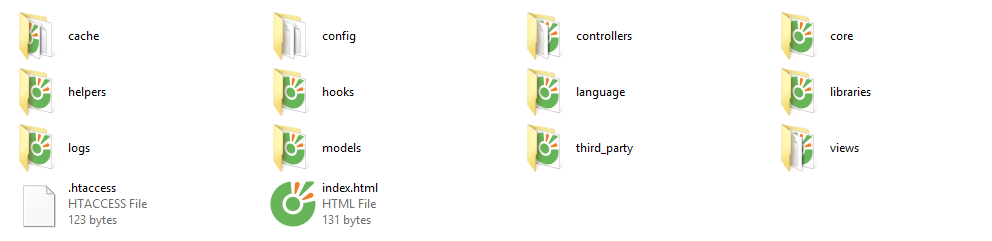


Hình 1. Cấu trúc thư mục Codeigniter Framework

Trong các thư mục download về thì ý nghĩa từng thư mục và file như sau:

* **application**: đây là thư mục rất quan trọng, thư mục chứa ứng dụng,nơi đây sẽ là nơi chúng ta viết code cho dự án.
* **system**: đây là thư mục chứa toàn bộ thư viện của CodeIgniter, các bạn không lên chỉnh sửa hay can thiệp gì vào đây,nếu muốn thì có thể mở ra xem thôi.
* **user\_guide**: đây là thư mục chứa toàn bộ hướng dẫn sử dụng, các bạn có thể xóa thư mục này.
* **index.php**: các bạn giữ nguyên file này.

Bây giờ các bạn hãy mở thư mục Application ra để xem cấu trúc và ý nghĩa của nó thể nào nhé:



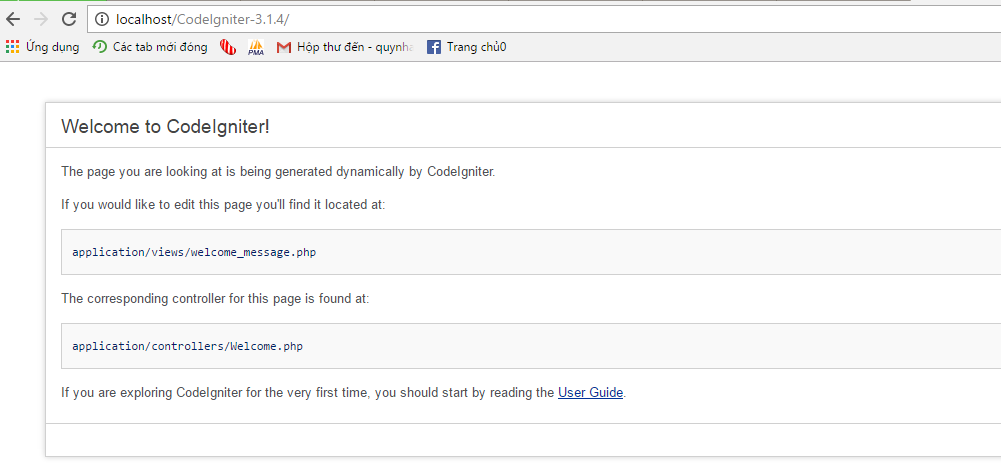
Hình 2. Cấu trúc thư mục Aplication

* **config**: đây là thư mục chứa toàn bộ cấu hình website, cấu hình **database**, đường dẫn,ngôn ngữ…
* **core**: Đây là thư mục viết các core của hệ thống,có thể viết các main **controller**,**model…** để kế thừa
* **controller**: Đây là thư mục chứa toàn bộ các file **controller** (xử lý dữ liệu)
* **model:** Là nơi viết các model của hệ thống (làm việc với csdl)
* **views**: là nơi chứa các views (hiển thị dữ liệu)
* **helpers**: chứa các helpers (các hàm tự xây dựng)
* **libraries:** chứa **các** thư viện mà chúng ta tự phát triển
* **language**: chứa các file ngôn ngữ,nếu bạn làm website đa ngôn ngữ thì cần sử dụng tới thư mục này

Ngoài ra còn 1 vài thư mục khác, nhưng ít được sử dụng hơn nên mình không đề cập trong tài liệu này.

**Chạy ứng dụng đầu tiên với Codeigniter**

Bây giờ các bạn hãy copy toàn bộ thư mục mà các bạn download về vào thư mục chạy web trên máy các bạn (mình sử dụng xampp lên copy vào thư mục htdocs),sau đó các bạn chạy đường link http://localhost/CodeIgniter-3.1.4/ sẽ được kết quả như sau:

****

Hình 3. Chạy thử chương trình codeigniter đầu tiên

Như vậy là các bạn đã chạy thành công rồi đó,khi các bạn chạy link website như vậy mặc định CodeIgniter sẽ chạy tới controller theo đường dẫn application/controllers/welcome.php,đó chính là những gì mà các bạn nhìn thấy trên màn hình.

– Bây giờ chúng ta cùng viết ứng dụng đầu tiên nhé:  
Bươc 1: tạo file index.php trong thư mục application/controlers với nội dung:

<?php

class Index extends CI\_Controller{

public function \_\_construct(){

parent::\_\_construct();

}

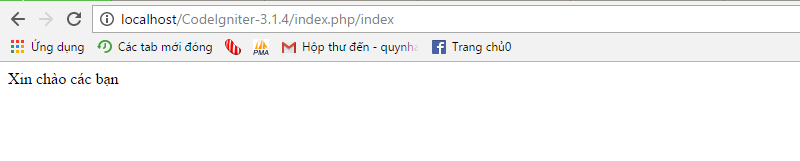
public function index(){

echo "Xin chào các bạn";

}

}

Bước 2: Sau đó các bạn thử chạy với link http://localhost/CodeIgniter-3.1.4/index.php/index và sẽ được kết quả như sau:



Hình 4. Chương trình chạy trên trình duyệt

Mình xin giải thích qua 1 chút về đoạn code trên:  
  
+ Khi khởi tạo controller thì tên file và tên lớp phải giống nhau. Cụ thể ở đây là tên lớp index và tên file index.php phải giống nhau.

+ Theo quy tắc của CI thì chữ cái đầu tiên của lớp phải viết hoa.

+ Mọi lớp tạo ra trên quy tắc là phải kế thừa lớp CI\_Controller

+ Trong classs index này mình viết hàm khởi tạo và sử dụng parent::\_\_construct(); nhằm tránh việc ghi đè, nếu bạn viết 1 hàm khởi tạo mà không sử dụng kế thừa từ \_\_construct(); trong CI\_Controller thì chắc chắn sẽ gặp lỗi

+ Để chạy các phương thức(action) thì chúng ta phải tạo nó như 1 phương thức bình thường.Và nếu không chạy action nào trong controller thì mặc định nó sẽ load phương thức (action) index

+ Để chạy ứng dụng ta phải đi qua file index.php. File này được hiểu là 1 front controller. Nó sẽ điều hướng dữ liệu mà chúng ta sẽ gởi đi.

* 1. **Mysql**
     1. **Giới thiệu Mysql**

**Cơ sở dữ liệu là gì?**

Một Database (Cơ sở dữ liệu) là một ứng dụng riêng rẽ mà lưu trữ một tập hợp dữ liệu. Mỗi cơ sở dữ liệu có một hoặc nhiều API riêng biệt để tạo, truy cập, quản lý, tìm kiếm và tái tạo dữ liệu nó đang giữ.

Một số loại kho lưu dữ liệu khác có thể được sử dụng, chẳng hạn như file trên hệ thống file hoặc các Hash Table lớn, nhưng việc lấy và ghi dữ liệu không thể nhanh và dễ dàng với các loại kho lưu dữ liệu này của các hệ thống.

Vì thế, ngày nay, chúng ta sử dụng các Hệ thống quản lý cơ sở dữ liệu quan hệ (RDBMS) lưu giữ và quản lý khối lượng lớn dữ liệu. Nó được gọi là cơ sở dữ liệu quan hệ, bởi vì tất cả dữ liệu được lưu giữ trong các bảng dữ liệu khác nhau và các mối quan hệ được thành lập bởi sử dụng các Primary Key (khóa chính) và một số khóa khác được biết đến như là Foreign Key.

Một Hệ thống quản lý cơ sở dữ liệu quan hệ (RDBMS) là một phần mềm mà:

* Cho bạn khả năng triển khai một Database với các bảng dữ liệu, cột (column), và các chỉ mục (Index).
* Bảo đảm Referential Integrity (có thể dịch là toàn vẹn quan hệ) giữa các hàng và các bảng đa dạng.
* Cập nhật tự động các chỉ mục.
* Thông dịch một truy vấn SQL và tổ hợp thông tin từ các bảng khác nhau.

**Thuật ngữ RDBMS**

Trước khi đi vào khái niệm hệ thống cơ sở dữ liệu MySQL, chúng ta cùng xem lại một số định nghĩa liên quan tới cơ sở dữ liệu:

* **Database:** Một cơ sở dữ liệu là một tập hợp các bảng dữ liệu, với dữ liệu có liên quan.
* **Bảng dữ liệu:** Một bảng là một ma trận dữ liệu. Một bảng trong một cơ sở dữ liệu trông giống như một bảng tính đơn giản.
* **Cột:** Một cột chứa cùng một kiểu dữ liệu, ví dụ như tên khách hàng.
* **Hàng:** Một hàng (row, entry, record) là một nhóm dữ liệu có liên quan.
* **Redundancy:** (có thể hiểu là dữ liệu dự phòng) Dữ liệu được lưu giữ hai lần, để làm cho hệ thống nhanh hơn.
* **Primary Key:** Một Primary Key (Khóa chính) là duy nhất. Một giá trị key không thể xuất hiện hai lần trong một bảng. Với một key, bạn có thể tìm thấy phần lớn trên một hàng.
* **Foreign Key:** Bạn tưởng tượng về Foreign Key như là cái ghim liên kết giữa hai bảng.
* **Compound Key:** Một Compound Key (hay composite key) là một key mà gồm nhiều cột, bởi vì một cột là không duy nhất.
* **Index:** Một chỉ mục trong một cơ sở dữ liệu tương tự như chỉ mục trong một cuốn sách.
* **Referential Integrity:** Đảm bảo rằng một giá trị Foreign Key luôn luôn trỏ tới một hàng đang tồn tại.

**Mysql là gì**

**MySQL** là một RDBMS nhanh và dễ dàng để sử dụng. MySQL đang được sử dụng cho nhiều công việc kinh doanh từ lớn tới nhỏ. MySQL được phát triển, được công bố, được hỗ trợ bởi MySQL AB, là một công ty của Thụy Điển. MySQL trở thành khá phổ biến vì nhiều lý do:

* MySQL là mã ngồn mở. Vì thế, để sử dụng nó, bạn chẳng phải mất một xu nào.
* MySQL là một chương trình rất mạnh mẽ.
* MySQL sử dụng một Form chuẩn của ngôn ngữ dữ liệu nổi tiếng là SQL.
* MySQL làm việc trên nhiều Hệ điều hành và với nhiều ngôn ngữ như PHP, PERL, C, C++, Java, …
* MySQL làm việc nhanh và khỏe ngay cả với các tập dữ liệu lớn.
* MySQL rất thân thiện với PHP, một ngôn ngữ rất đáng giá để tìm hiểu để phát triển Web.
* MySQL hỗ trợ các cơ sở dữ liệu lớn, lên tới 50 triệu hàng hoặc nhiều hơn nữa trong một bảng. Kích cỡ file mặc định được giới hạn cho một bảng là 4 GB, nhưng bạn có thể tăng kích cỡ này (nếu hệ điều hành của bạn có thể xử lý nó) để đạt tới giới hạn lý thuyết là 8 TB.
* MySQL là có thể điều chỉnh. Giấy phép GPL mã nguồn mở cho phép lập trình viên sửa đổi phần mềm MySQL để phù hợp với môi trường cụ thể của họ.
  + 1. **Cài đặt MySQL**

***Cài đặt MySQL trên Linux/Unix***

Bạn nên cài đặt MySQL trên hệ thống Linux thông qua RPM. MySQL AB cung cấp sẵn RPM cho bạn tải về.

**MySQL -** MySQL Database Server, quản lý cơ sở dữ liệu và bảng, điều khiển sự truy cập của người dùng, và xử lý các truy vấn SQL

**MySQL-client -** Các chương trình MySQL Client, giúp nó có thể để kết nối và tương tác với Server.

**MySQL-devel -** Các thư viện và Header file khá là thuận tiện khi biên dịch các chương trình khác mà sử dụng MySQL

**MySQL-shared -**Các thư viện có thể chia sẻ cho MySQL Client.

**MySQL-bench -**Tiêu chuẩn thử nghiệm các tool cho MySQL Database Server.

MySQL RPMs, mà được liệt kê ở đây, là được xây dựng trên một hệ thống SuSE Linux, nhưng chúng thường làm việc trên các phiên bản Linux khác mà không gặp phải vấn đề gì.

Dưới đây là các bước để cài đặt:

* Đăng nhập hệ thống bởi sử dụng root user.
* Chuyển tới thư mục chứa RPMs.
* Cài đặt MySQL Database Server bằng việc thực thi lệnh sau. Bạn nhớ thay thế filename trong chữ in nghiêng với tên tệp của RPM của bạn.

[root@host]# rpm -i *MySQL-5.0.9-0.i386.rpm*

* Lệnh trên sẽ xử lý việc cài đặt MySQL Server, tạo một user của MySQL, tạo cấu hình cần thiết và bắt đầu MySQL Server một cách tự động.
* Bạn có thể tìm tất cả /usr/bin và /usr/sbin liên quan tới MySQL. Tất cả bảng và cơ sở dữ liệu sẽ được tạo trong thư mục /var/lib/mysql.
* Phần này là tùy ý, nhưng cũng giới thiệu cho bạn để cài đặt RPMs theo cùng cách thức:

*[root@host]# rpm -i MySQL-client-5.0.9-0.i386.rpm*

*[root@host]# rpm -i MySQL-devel-5.0.9-0.i386.rpm*

*[root@host]# rpm -i MySQL-shared-5.0.9-0.i386.rpm*

*[root@host]# rpm -i MySQL-bench-5.0.9-0.i386.rpm*

**Cài đặt MySQL trên Windows**

Bây giờ, cài đặt mặc định trên bất kỳ phiên bản Windows nào là dễ dàng hơn để sử dụng, khi MySQL được đóng gói hoàn toàn với một Installer. Bạn chỉ cần tải Installer package này, unzip nó ở bất cứ đâu, và chạy setup.exe.

Installer mặc định setup.exe sẽ đưa bạn qua một tiến trình thông thường và theo mặc định sẽ cài đặt mọi thứ dưới C:\mysql.

Kiểm tra Server bằng việc kích hoạt nó từ Command Prompt lần đầu tiên. Tới location của mysqld server mà có thể là C:\mysql\bin, và soạn:

mysqld.exe --console

**Ghi chú:** Nếu bạn đang sử dụng NT, thì bạn sẽ phải sử dụng mysqld-nt.exe thay vì mysqld.exe.

Nếu tất cả hoạt động tốt, bạn sẽ thấy một số thông báo về Startup và InnoDB. Nếu không, bạn có thể có một thông báo về giấy phép. Đảm bảo rằng thư mục mà lưu giữ dữ liệu là có thể truy cập với bất kỳ người dùng nào (có thể là mysql).

MySQl không thêm phần Start Menu, và không có GUI để dừng Server. Vì thế, nếu bạn muốn kích đúp chuột để khởi động Server, bạn nên nhớ dừng tiến trình này bằng tay bởi sử dụng mysqladmin, Task List, Task Manager, hoặc các cách thức cụ thể cho Windows khác.

* + 1. **Sử dụng**

**Cú pháp kết hợp MySQL và PHP**

MySQL làm việc tốt khi kết hợp với nhiều ngôn ngữ lập trình đa dạng như PERL, C, C++, Java, và PHP. Trong đó, PHP được sử dụng phổ biến nhất bởi vì khả năng phát triển ứng dụng Web của nó. Loạt bài này trọng tâm sử dụng MySQL trong một môi trường PHP.

PHP cung cấp các hàm đa dạng để truy cập MySQL Database và để thao tác các bản ghi dữ liệu bên trong MySQL Database. Bạn sẽ cần gọi các hàm theo đúng như cách bạn gọi bất kỳ hàm PHP khác.

Các hàm PHP để sử dụng với MySQL có dạng tổng quát như sau:

mysql\_*Hàm*(giatri,giatri,...);

Phần thứ hai (*Hàm*) là cụ thể cho từng hàm, thường là một từ miêu tả những gì hàm đó thực hiện. Dưới đây là hai hàm sẽ được sử dụng trong loạt bài này.

mysql\_connect($connect);

mysql\_query($connect,"Lệnh SQL");

Ví dụ sau minh họa cú pháp chung của PHP để gọi bất kỳ hàm MySQL nào.

<html>

<head>

<title>Cu phap chung cua PHP va MySQL</title>

</head>

<body>

<?php

$retval = mysql\_Hàm(giatri, giatri,...);

if( !$retval )

{

die ( "Thong bao ve loi co lien quan" );

}

// Lenh MySQL hoac PHP khac

?>

</body>

</html>

**Tạo Database trong MySQL**

Để tạo cơ sở dữ liệu, bạn sử dụng lệnh CREATE trong MySQL. Dưới đây là cú pháp của lệnh CREATE:

CREATE DATABASE Ten\_co\_so\_du\_lieu;

Ví dụ

Ví dụ đơn giản sau sẽ tạo cơ sở dữ liệu là sinhvien:

CREATE DATABASE sinhvien;

Lệnh này sẽ tạo một cơ sở dữ liệu có tên là sinhvien trong MySQL Database.

**Xóa cơ sở dữ liệu trong MySQL**

Để xóa bất kỳ cơ sở dữ liệu nào mà bạn thấy không cần thiết, bạn sử dụng lệnh DROP DATABASE trong MySQL. Cẩn thận trong khi xóa bất kỳ cơ sở dữ liệu nào bởi vì bạn sẽ mất toàn bộ dữ liệu có trong cơ sở dữ liệu đó (Có lẽ hoạt động xóa này chỉ có trong quá trình thực hành của bạn, khi làm việc thực tế thì việc này rất ít xảy ra).

Cú pháp của lệnh DROP DATABASE trong MySQL là:

DROP DATABASE ten\_co\_so\_du\_lieu;

Dưới đây là ví dụ để xóa một cơ sở dữ liệu đã được tạo trong chương trước là sinhvien.

DROP DATABASE sinhvien;

**Chọn cơ sở dữ liệu trong MySQL**

Chọn một cơ sở dữ liệu cụ thể trong MySQL là rất đơn giản. Bạn có thể sử dụng lệnh **use** trong SQL để lựa chọn một cơ sở dữ liệu. Cú pháp của lệnh use là:

USE ten\_co\_so\_du\_lieu;

Ví dụ sau lựa chọn một cơ sở dữ liệu có tên **sinhvien**:

USE sinhvien;

Bây giờ, bạn đã lựa chọn cơ sở dữ liệu sinhvien và tất cả hoạt động sau đó sẽ được thực hiện trên cơ sở dữ liệu này.

**GHI CHÚ:** Tất cả tên cơ sở dữ liệu, tên bảng, tên các trường của bảng là phân biệt kiểu chữ. Vì thế, bạn phải sử dụng tên chính xác trong khi cung cấp cho bất kỳ lệnh SQL nào.

**Kiểu dữ liệu số trong MySQL**

MySQL sử dụng tất cả các kiểu dữ liệu số theo chuẩn ANSI SQL, vì thế nếu bạn đã làm quen với một hệ thống cơ sở dữ liệu khác, thì bạn sẽ thấy những định nghĩa này là khá thân thuộc với bạn khi học về MySQL. Dưới đây liệt kê các kiểu dữ liệu số phổ biến và miêu tả của chúng:

* **INT**- Một số nguyên với kích cỡ thông thường, có thể là signed hoặc unsigned. Nếu có dấu, thì dãy giá trị có thể là từ -2147483648 tới 2147483647, nếu không dấu thì dãy giá trị là từ 0 tới 4294967295. Bạn có thể xác định một độ rộng lên tới 11 chữ số.
* **TINYINT**- Một số nguyên với kích cỡ rất nhỏ, có thể là signed hoặc unsigned. Nếu có dấu, thì dãy giá trị có thể là từ -128 tới 127, nếu không dấu thì dãy giá trị là từ 0 tới 255. Bạn có thể xác định một độ rộng lên tới 4 chữ số.
* **SMALLINT**- Một số nguyên với kích cỡ nhỏ, có thể là signed hoặc unsigned. Nếu có dấu, thì dãy giá trị có thể là từ -32768 tới 32767, nếu không dấu thì dãy giá trị là từ 0 tới 65535. Bạn có thể xác định một độ rộng lên tới 5 chữ số.
* **MEDIUMINT**- Một số nguyên với kích cỡ trung bình, có thể là signed hoặc unsigned. Nếu có dấu, thì dãy giá trị có thể là từ -8388608 tới 8388607, nếu không dấu thì dãy giá trị là từ 0 tới 16777215. Bạn có thể xác định một độ rộng lên tới 9 chữ số.
* **BIGINT**- Một số nguyên với kích cỡ lớn, có thể là signed hoặc unsigned. Nếu có dấu, thì dãy giá trị có thể là từ -9223372036854775808 tới 9223372036854775807, nếu không dấu thì dãy giá trị là từ 0 tới 18446744073709551615. Bạn có thể xác định một độ rộng lên tới 20 chữ số.
* **FLOAT(M,D)**- Một số thực dấu chấm động không dấu. Bạn có thể định nghĩa độ dài hiển thị (M) và số vị trí sau dấy phảy (D). Điều này là không bắt buộc và sẽ có mặc định là 10,2: với 2 là số vị trí sau dấu phảy và 10 là số chữ số (bao gồm các phần thập phân). Phần thập phân có thể lên tới 24 vị trí sau dấu phảy đối với một số FLOAT.
* **DOUBLE(M,D)**- Một số thực dấu chấm động không dấu. Bạn có thể định nghĩa độ dài hiển thị (M) và số vị trí sau dấy phảy (D). Điều này là không bắt buộc và sẽ có mặc định là 16,4: với 4 là số vị trí sau dấu phảy và 16 là số chữ số (bao gồm các phần thập phân). Phần thập phân có thể lên tới 53 vị trí sau dấu phảy đối với một số DOUBLE. REAL là đồng nghĩa với DOUBLE.
* **DECIMAL(M,D)**- Một kiểu khác của dấu chấm động không dấu. Mỗi chữ số thập phân chiếm 1 byte. Việc định nghĩa độ dài hiển thị (M) và số vị trí sau dấy phảy (D) là bắt buộc. NUMERIC là một từ đồng nghĩa cho DECIMAL.

**Kiểu dữ liệu Date và Time trong MySQL**

Kiểu dữ liệu Date và Time được phân loại thành:

* DATE - Một date trong định dạng YYYY-MM-DD, giữa 1000-01-01 và 9999-12-31. Ví dụ, ngày 25 tháng 12 năm 2015 sẽ được lưu ở dạng 2015-12-25.
* DATETIME - Một tổ hợp Date và Time trong định dạng YYYY-MM-DD HH:MM:SS, giữa 1000-01-01 00:00:00 và 9999-12-31 23:59:59. Ví dụ, 3:30 chiều ngày 25 tháng 12, năm 2015 sẽ được lưu ở dạng 2015-12-25 15:30:00.
* TIMESTAMP - Một Timestamp từ giữa nửa đêm ngày 1/1/1970 và 2037. Trông khá giống với định dạng DATETIME trước, khác biệt ở chỗ không có dấu gạch nối giữa các số. Ví dụ, 3:30 chiều ngày 25 tháng 12, năm 2015 sẽ được lưu dưới dạng 20151225153000 ( YYYYMMDDHHMMSS ).
* TIME - Lưu time trong định dạng HH:MM:SS.
* YEAR(M) - Lưu 1 năm trong định dạng 2 chữ số hoặc 4 chữ số. Nếu độ dài được xác định là 2 (ví dụ: YEAR(2)), YEAR có thể từ 1970 tới 2069 (70 tới 69). Nếu độ dài được xác định là 4, YEAR có thể từ 1901 tới 2155. Độ dài mặc định là 4.

**Kiểu dữ liệu chuỗi trong MySQL**

Ngoài các kiểu dữ liệu số hoặc kiểu dữ liệu date và time, thì còn một kiểu dữ liệu mà bạn thường sử dụng nhất trong MySQL là kiểu dữ liệu chuỗi. Dưới đây liệt kê các kiểu dữ liệu chuỗi và phần miêu tả của chúng trong MySQL:

* CHAR(M) - Một chuỗi có độ dài cố định có độ dài từ 1 tới 255 ký tự (ví dụ CHAR(5)). Nếu giá trị thật của một trường kiểu Char không bằng với độ dài khai báo thì phần thiếu bên phải của nó sẽ được thêm bằng các kí tự trắng một cách tự động. Định nghĩa độ dài là không bắt buộc, giá trị mặc định là 1.
* VARCHAR(M) - Dữ liệu kiểu chuỗi có độ dài thay đổi, có độ dài từ 1 đến 255 kí tự (ví dụ Varchar(24)). Bạn phải định nghĩa độ dài khi tạo một trường VARCHAR.
* BLOB hoặc TEXT - Trường kiểu này có độ dài tối đa 65535 kí tự. BLOBs là viết tắt của "Binary Large Objects", và được sử dụng để lưu trữ một lượng lớn dữ liệu nhị phân như các bức ảnh hoặc các loại tập tin khác. Với TEXT, trường cũng lưu trữ được một lượng lớn dữ liệu. Điểm khác nhau giữa chúng là: khi sắp xếp và so sánh dữ liệu đã lưu trữ thì với BLOBs là phân biệt kiểu chữ, còn với TEXT là không phân biệt kiểu chữ. Bạn không phải xác định độ dài với BLOBs hoặc TEXT.
* TINYBLOB hoặc TINYTEXT - Một cột BLOB hoặc TEXT với độ dài tối đa là 255 ký tự. Bạn không cần xác định độ dài với TINYBLOB hoặc TINYTEXT.
* MEDIUMBLOB hoặc MEDIUMTEXT - Một cột BLOB hoặc TEXT với độ dài tối đa là 16777215 ký tự. Bạn không cần xác định độ dài với MEDIUMBLOB hoặc MEDIUMTEXT.
* LONGBLOB hoặc LONGTEXT - Một cột BLOB hoặc TEXT với độ dài tối đa là 4294967295 ký tự. Bạn không cần xác định độ dài với LONGBLOB hoặc LONGTEXT.
* ENUM - Khi định nghĩa một trường kiểu này, tức là, ta đã chỉ ra một danh sách các đối tượng mà trường phải nhận (có thể là Null). Ví dụ, nếu ta muốn một trường nào đó chỉ nhận một trong các giá trị "A" hoặc "B" hoặc "C" thì ta phải định nghĩa kiểu ENUM cho nó như sau: ENUM ('A', 'B', 'C'). Và chỉ có các giá trị này (hoặc NULL) có thể xuất hiện trong trường đó.

**Tạo bảng**

Dưới đây là cú pháp SQL cơ bản để tạo một bảng trong MySQL:

* CREATE TABLE ten\_bang (ten\_cot kieu\_du\_lieu\_cucot);

Dưới đây là ví dụ để tạo một bảng có tên là **sinhvienk60** với các trường **mssv, ho, ten, tuoi, diemthick** trong cơ sở dữ liệu **sinhvien**:

**Ghi chú**: Nếu bạn có nhiều cơ sở dữ liệu, thì để tạo bảng sinhvienk60 có trong cơ sở dữ liệu sinhvien thì trước hết bạn phải chọn cơ sở dữ liệu đó với lệnh **USE**.

CREATE TABLE sinhvienk60 (

mssv INT NOT NULL AUTO\_INCREMENT,

ho VARCHAR(255) NOT NULL,

ten VARCHAR(255) NOT NULL,

tuoi INT NOT NULL,

diemthi FLOAT(4,2) NOT NULL,

PRIMARY KEY (mssv)

);

Một số khái niệm cần giải thích:

* Thuộc tính **NOT NULL** của trường đang được sử dụng bởi vì chúng ta không muốn trường này là NULL. Vì thế, nếu người dùng cố gắng tạo một bản ghi có giá trị NULL, thì MySQL sẽ tạo một lỗi.
* Thuộc tính **AUTO\_INCREMENT** nói cho MySQL tự động tăng khóa chính và thêm giá trị có sẵn tiếp theo tới trường id.
* Từ khóa **PRIMARY KEY** được sử dụng để định nghĩa một cột là PRIMARY KEY (khóa chính). Bạn có thể sử dụng nhiều cột phân biệt nhau bởi dấu phảy để định nghĩa một PRIMARY KEY.

**Xóa bảng trong MySQL**

Cú pháp SQL chung để xóa một bảng MySQL là:

DROP TABLE ten\_bang;

Ví dụ

Để xóa bảng **sinhvienk60** đã tạo ở chương trước, bạn sử dụng:

DROP TABLE sinhvienk60;

**Truy vấn INSERT trong MySQL**

Cú pháp SQL cơ bản của lệnh INSERT INTO để chèn dữ liệu vào trong bảng MySQL là:

INSERT INTO ten\_bang ( truong1, truong2,...truongN )

VALUES

( giatri1, giatri2,...giatriN );

Để chèn kiểu dữ liệu chuỗi, điều bắt buộc là bạn phải giữ tất cả giá trị trong các trích dẫn đơn hoặc trích dẫn kép, ví dụ: **"giatri"**.

**Ví dụ chèn dữ liệu vào bảng trong MySQL**

Dưới đây là ví dụ minh họa việc sử dụng lệnh INSERT INTO để chèn dữ liệu vào trong bảng. Trong các chương trước, chúng ta đã tạo bảng **sinhvienk60**, bây giờ chúng ta chèn các bản ghi như sau:

INSERT INTO sinhvienk60 (ho, ten, diemthi)

VALUES ("Dinh Van", "Cao", 8);

INSERT INTO sinhvienk60 (ho, ten, diemthi)

VALUES ("Nguyen Van", "Thanh", 9);

INSERT INTO sinhvienk60 (ho, ten, diemthi)

VALUES ("Nguyen Hoang", "Manh", 7.5);

INSERT INTO sinhvienk60 (ho, ten, diemthi)

VALUES ("Tran Van", "Nam", 10);

Trong ví dụ trên, mình đã không cung cấp **mssv** bởi vì tại thời điểm tạo bảng này, mình đã cung cấp tùy chọn AUTO\_INCREMENT cho trường này. Vì thế, MySQL sẽ xử lý việc chèn các mssv của nó một cách tự động.

**Truy vấn SELECT trong MySQL**

Cú pháp SQL cơ bản của lệnh SELECT để lấy dữ liệu từ bảng MySQL là:

SELECT truong1, truong2,...truongN FROM ten\_bang

[menhDe WHERE]

[OFFSET M ][LIMIT N]

Bạn có thể lấy một hoặc nhiều trường trong một lệnh SELECT đơn.

Bạn có thể xác định một dấu sao (\*) thay cho các trường. Trong trường hợp này, lệnh SELECT sẽ trả về tất cả các trường.

Bạn có thể xác định bất kỳ điều kiện nào bởi sử dụng mệnh đề WHERE.

Bạn có thể xác định một offset bởi sử dụng **OFFSET** để từ đó lệnh SELECT này sẽ bắt đầu trả về các bản ghi. Theo mặc định, offset là 0.

Bạn có thể giới hạn số kết quả trả về bởi sử dụng thuộc tính **LIMIT**.

**Truy vấn UPDATE trong MySQL**

Dưới đây là cú pháp SQL chung của lệnh UPDATE để sửa đổi dữ liệu trong bảng MySQL:

UPDATE ten\_bang SET truong1=giaTri\_moi\_1, truong2=giaTri\_moi\_2

[MenhDe WHERE];

Bạn có thể cập nhật một hoặc nhiều trường.

Bạn có thể xác định bất kỳ điều kiện nào bởi sử dụng mệnh đề WHERE.

Bạn có thể cập nhật các giá trị trong một bảng đơn tại một thời điểm.

Mệnh đề WHERE là hữu ích khi bạn muốn cập nhật các hàng đã chọn trong một bảng.

**Cập nhật dữ liệu trong MySQL**

Bạn sử dụng lệnh UPDATE với mệnh đề WHERE trong SQL để cập nhật dữ liệu đã chọn vào trong bảng MySQL.

**Ví dụ**

Ví dụ sau sẽ cập nhật bản ghi có **mssv** là 3 trong trường **ten** trong bảng sinhvienk60:

UPDATE sinhvienk60 SET ten="Huong" WHERE mssv=3;

**Truy vấn DELETE trong MySQL**

Cú pháp SQL chung của lệnh DELETE để xóa dữ liệu từ một bảng MySQL là:

DELETE FROM ten\_bang [Menhde WHERE]

Nếu mệnh đề WHERE không được xác định, thì tất cả bản ghi sẽ bị xóa từ bảng MySQL đã cho.

Bạn có thể xác định bất kỳ điều kiện nào với mệnh đề WHERE.

Bạn có thể xóa các bản ghi trong một bảng đơn tại cùng một thời điểm.

Mệnh đề WHERE là thực sự hữu ích khi bạn muốn xóa các hàng đã chọn trong một bảng. Nếu không xác định mệnh đề WHERE thì thật sự rất nguy hiểm bởi vì **toàn bộ bảng của bạn sẽ bị xóa**. Do đó bạn phải thật cẩn thận khi sử dụng lệnh này.

***Xóa dữ liệu trong MySQL***

Ví dụ sau sẽ xóa một bản ghi trong bảng sinhvienk60 có mssv là 4.

DELETE FROM sinhvienk60 WHERE mssv=4;

**Hàm hữu ích trong MySQL**

Dưới đây liệt kê tất cả các hàm MySQl quan trọng. Mỗi hàm đều được giải thích chi tiết và có ví dụ phù hợp.

[Hàm COUNT trong MySQL](http://vietjack.com/mysql/ham_count_trong_mysql.jsp) - Hàm tập hợp COUNT trong MySQL được sử dụng để đếm số hàng trong một bảng dữ liệu.

[Hàm MAX trong MySQL](http://vietjack.com/mysql/ham_max_trong_mysql.jsp) - Hàm tập hợp MAX trong MySQL cho phép chúng ta lựa chọn giá trị lớn nhất (tối đa) trong một cột cụ thể.

[Hàm MIN trong MySQL](http://vietjack.com/mysql/ham_min_trong_mysql.jsp) - Hàm tập hợp MIN trong MySQL cho phép chúng ta lựa chọn giá trị nhỏ nhất (tối thiểu) trong một cột cụ thể.

[Hàm AVG trong MySQL](http://vietjack.com/mysql/ham_avg_trong_mysql.jsp) - Hàm tập hợp AVG trong MySQL cho phép chúng ta lựa chọn giá trị trung bình trong một cột cụ thể.

[Hàm SUM trong MySQL](http://vietjack.com/mysql/ham_sum_trong_mysql.jsp) - Hàm tập hợp AVG trong MySQL cho phép chúng ta lựa chọn giá trị trung bình trong một cột cụ thể.

[Hàm SQRT trong MySQL](http://vietjack.com/mysql/ham_sqrt_trong_mysql.jsp) - Được sử dụng để tính căn bậc hai của một số đã cho.

[Hàm RAND trong MySQL](http://vietjack.com/mysql/ham_rand_trong_mysql.jsp) - Được sử dụng để tạo một số ngẫu nhiên bởi sử dụng lệnh trong MySQL.

[Hàm CONCAT trong MySQL](http://vietjack.com/mysql/ham_concat_trong_mysql.jsp) - Được sử dụng để nối bất kỳ chuỗi nào trong bất kỳ lệnh MySQL nào.

[Hàm xử lý DATE và Time trong MySQL](http://vietjack.com/mysql/ham_xu_ly_date_time_trong_mysql.jsp) - Danh sách đầy đủ các hàm liên quan tới Date và Time trong MySQL.

[Hàm xử lý số trong MySQL](http://vietjack.com/mysql/ham_so_hoc_trong_mysql.jsp) - Danh sách đầy đủ các hàm cần thiết để thao tác với số trong MySQL.

[Hàm xử lý chuỗi trong MySQL](http://vietjack.com/mysql/ham_xu_ly_chuoi_trong_mysql.jsp) - Danh sách đầy đủ các hàm cần thiết để thao tác với chuỗi trong MySQL.

# CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

* 1. **Khảo sát**
     1. **Hiện trạng thị trường việc làm hiện nay**

Mỗi năm, Việt Nam có khoảng 1,2 triệu người đến tuổi lao động và được bổ sung vào lực lượng lao động của đất nước.

Tính chung cả nước, có 10,3% lao động có việc làm trong khu vực kinh tế nhà nước, 88,2% đang làm việc trong khu vực kinh tế ngoài nhà nước và 1,5% làm việc trong khu vực kinh tế có vốn đầu tư nước ngoài.

Theo báo cáo kết quả điều tra, trong lực lượng lao động từ độ tuổi 15 trở lên: khu vực thành thị 94,6% có việc làm và 5,4% thất nghiệp, khu vực nông thôn có 98,9% có việc làm và 1,1% thất nghiệp (theo VnExpress.com.vn).

Đa số các bạn sinh viên sau khi ra trường thường cố gắng trụ lại các thành phố lớn với mong muốn tìm được một công việc phù hợp, dù hiện nay, ở các tỉnh thành đang có rất nhiều chính sách thu hút nhân tài chất xám với nhiều ưu đãi đặc biệt. Việc ở lại thành phố cũng có nghĩa là họ đã đặt một chân vào công cuộc chạy đua tìm kiếm việc làm, và làm cho thị trường việc làm tại các đô thị trung tâm sôi động và nóng bỏng lên từng ngày.

Đáp ứng nhu cầu này, trên thị trường hiện nay có nhiều trung tâm giới thiệu việc làm đã được thành lập. Đa số các trung tâm hiện nay được thành lập không có giấy phép, chỉ có một số trung tâm đáng tin cậy như: trung tâm hỗ trợ việc làm nhà văn hoá thanh niên, trung tâm giới thiệu việc làm bộ thương binh xã hội…

Tuy vậy, các công việc ở các trung tâm này chỉ đáp ứng một phần nào nhu cầu công việc của thị trường việc làm. Đối tượng chủ yếu của các trung tâm này là sinh viên học sinh hay các kĩ sư, cử nhân mới ra trường không biết bắt đầu tìm việc từ đâu.

Vấn đề chủ yếu là số người tìm việc ngày càng tăng mà công việc ngày càng hạn chế và kén người, do đó số lượng hồ sơ tìm việc rất nhiều. Vì vậy việc quản lý, chọn lọc hồ sơ theo cách thủ công là không thoả đáng, mất nhiều thời gian…

* + 1. **Mục đích và phương pháp giải quyết**

Trong số chúng ta, chắc hẳn không ít người đã từng đi xin việc tại các trung tâm giới thiệu việc làm và cũng cảm thấy việc tìm kiếm một công việc phù hợp là gian nan, tốn nhiều thời gian và chi phí. Không chỉ đối với người tìm việc, mà ngay cả những người quản lý cũng khó khăn và phức tạp, nhất là với các trung tâm lớn, đặc biệt là việc tìm kiếm đầu việc, phân loại công việc theo ngành nghề, khu vực… Để hiểu rõ hơn những vấn đề này, chúng ta có thể đi chi tiết một ít về tình hình giới thiệu việc làm tại một trung tâm cụ thể. Qua tiếp xúc thực tế, chúng em nhận thấy, tại trung tâm, có một hệ thống sổ sách lưu trữ về chi tiết về các công việc bao gồm các thông tin về công ty cung cấp công viêc, tính chất công việc, thời gian, kinh nghiệm, lương… Tại các bảng tin, chỉ giới thiệu về công việc thông qua mã số và một vài chi tiết quan trọng. Khi người tìm việc muốn tham gia tuyển dụng công việc đó, có thể liên hệ nhân viên trung tâm, đóng lệ phí (đa số các trung tâm lấy phí bằng 30% tháng lương đầu tiên) và sẽ nhận được các thông tin cụ thể về công việc mà người đó lựa chọn và thông tin của người đó sẽ được lưu trữ lại để phòng trường hợp người đó không đồng ý về công việc thì có thể đến trung tâm lấy lại một phần lệ phí đã đóng. Tuy nhiên, đối với một trung tâm lớn, thì mỗi ngày có hàng trăm công việc được giới thiệu, việc lật lại hồ sơ tìm kiếm các khách hàng đã đăng kí trước đó là cực kì mất thời gian. Hàng tháng, trung tâm phải tổng kết lại số công việc đã giới thiệu trong tháng, tính doanh thu và lập kế hoạch cho tháng sau. Công việc tưởng nhẹ nhàng nhưng trên thực tế là rất phức tạp. Nhân viên phải lật lại các trang lưu trữ theo từng ngày để tìm số liệu và tính toán lại số đầu việc đã giới thiệu, số tiền thu được trong tháng…

Trong hầu hết các công việc trên đây, chúng ta cũng thấy những việc làm toán được làm bằng thủ công rất nhiều. Những công việc này được làm đi làm lại một cách thường xuyên, dữ liệu lưu trữ thì khá nhiều và phân phối một cách dày đặc. Nhận thấy vấn đề này chúng ta có thể đưa tin học vào vấn đề nêu trên nhằm giảm tối đa các tính toán không cần thiết, giảm công việc quản lý xuống. Và hơn thế nữa, để giảm thời gian cho người tìm việc, họ có thể tra cứu công việc trên máy, đăng kí dự tuyển và công việc còn lại là của người quản trị. Việc này khá thuận tiện cho cả hai phía, ngoài ra khi chúng ta thực hiện trên mạng còn có các ưu điểm như chúng ta có thể quảng cáo cho công ty đăng tuyển công việc, tư vấn các kinh nghiệm khi đi xin việc…Hơn nữa, đưa ứng dụng lên mạng, người xin việc không phải đi tìm kiếm khó khăn, không mất phí mà cũng có thể tìm được công việc phù hợp.

Hiện nay, có hàng loạt trang web việc làm đang tồn tại như: jobvn.com, 24h.com, mangtimviec.com… Với tính năng thuận tiện và dễ sử dung, giao diện dễ nhìn, gần gũi, các trang web này đã và đang thu hút đông đảo số thành viên tham gia.

Đồ án ra đời cũng không nằm ngoài mục đích trên: thiết kế website giới thiệu việc làm trực tuyến (kiemviecOnline). Thuận lợi của trang web là giảm tối đa mọi thủ tục rườm rà trong quá trình đi xin việc.

* 1. **Phân tích yêu cầu hệ thống**
     1. **Yêu cầu chức năng nghiệp vụ**
* Hệ thống phục vụ và trao đổi thông tin với các đối tượng sau:
  + Bộ phận nhà tuyển dụng
  + Bộ phận quản lý web
  + Bộ phận người tìm việc
* Cho phép người dùng đăng kí với vai trò là nhà tuyển dụng hay ứng viên.
* Lưu trữ thông tin của các đối tượng.
* Lưu trữ thông tin đăng kí tìm việc của ứng viên và thông tin tuyển dụng của nhà tuyển dụng.
* Cho phép các ứng viên đăng thông tin tìm việc.
* Cho phép các ứng viên tìm kiếm thông tin việc làm theo các tiêu chí khác nhau.
* Cho phép ứng viên hiệu chỉnh các thông tin khi cần thiết.
* Cho phép nhà tuyển dụng đăng thông tin tuyển dụng và có thể hiệu chỉnh bất cứ lúc nào.
* Cho phép nhà tuyển dụng đăng thông tin quảng cáo về các dịch vụ của mình.
* Hỗ trợ cho người quản lý trong việc quản lý hệ thống bao gồm: quản lý thông tin của các đối tượng (ứng viên, nhà tuyển dụng), quản lý các thông tin đăng tìm việc và đăng tuyển dụng và các thông tin liên quan.
* Hỗ trợ việc trao đổi thông tin lẫn nhau giữa nhà tuyển dụng và ứng viên.
  + 1. **Yêu cầu chức năng hệ thống**
* Cho phép người dùng đăng nhập vào hệ thống với các vai trò khác nhau ứng với từng đối tượng.
* Đảm bảo tính bảo mật thông tin đăng nhập của người dùng ở xa.
* Phân cấp quyền hạn của người dùng trong việc truy cập vào các tài nguyên của hệ thống.
  + 1. **Yêu cầu phi chức năng**

* Nhằm lợi ích của người dùng:

- Tính tiện dụng: giao diện thân thiện dễ sử dụng đối với các người dùng window.

- Tính hiệu quả: đảm bảo việc truy xuất nhanh đến các trang và khả năng kiểm soát lỗi tốt.

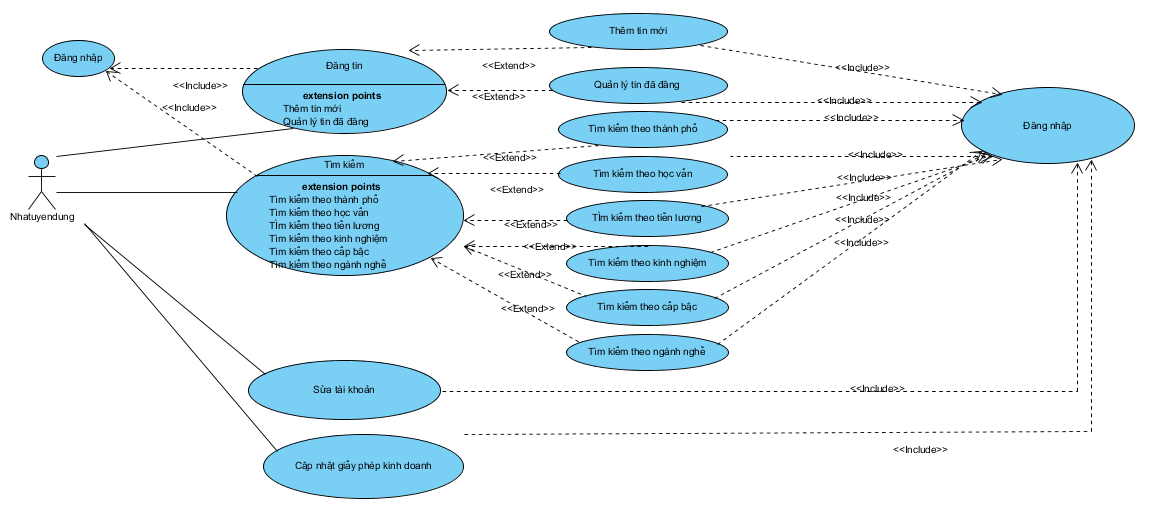
* Nhằm lợi ích của việc phát triển dự án

- Thiết kế theo mô hình MVC.

- Các tham số của hệ thống được thiết kế động, dễ điều chỉnh.

- Tính dùng lại của code.

* + 1. **Dữ liệu đưa vào hệ thống quản lý**
* **Công việc:**
  + Là đối tượng chủ yếu của website phục vụ cho đối tượng người tìm việc. Công việc có nhiều dạng khác nhau. Mỗi loại đòi hỏi một đối tượng, một trình độ phù hợp.
* **Người tìm việc**
  + Là đối tượng được phục vụ tại website. Người tìm việc bao gồm nhiều tầng lớp, trình độ học vấn và nghề nghiệp khác nhau. Họ có thể đăng kí để trở thành thành viên (thông tin được lưu trữ và họ có thể đăng nhập), tìm kiếm các công việc, đăng thông tin về bản thân… để có thể tìm kiếm một công việc thích hợp.
* **Nhà tuyển dụng**
  + Các nhà tuyển dụng sẽ đăng kí với website để trở thành nhà cung cấp công việc cho website và có thể đăng kí để quảng cáo trên tran..g web. Họ có nhiệm vụ cung cấp các thông tin về công ty họ, các ngành nghề cần tuyển, các yêu cầu… Các thông tin này sẽ được cập nhật vào các thời điểm thích hợp để đảm bảo tính chính xác của thông tin.
* **Các nhà quản trị website**:
* Nhà quản trị có thể đăng nhập để quản trị website ở bất cứ nơi đâu. Họ có quyền kiểm soát tất cả các hoạt động của website cụ thể như xem danh sách đăng kí (các công ty, thành viên), xoá…
  1. **Phân tích UML**
     1. **Use case diagram.**
        1. **Use case diagram – đối với nhóm chức năng thuộc nhà tuyển dụng.**



Hình 5. Mô hình usecase với nhóm chức năng thuộc nhà tuyển dụng

* + - * 1. **Actors**
* NhaTuyenDung (Nhà Tuyển dụng): Nhà tuyển dụng có vai trò quan trọng trong hệ thống, bao gồm đăng tuyển và tìm hồ sơ ứng viên.
* Mỗi nhà tuyển dụng có thể đăng nhiều hồ sơ tuyển dụng, sau khi đăng tuyển NTD có thể nhận được ngay kết quả so khớp các ứng viên phù hợp (các ứng viên này đã gửi đơn xin việc trực tuyến từ trước).
* Nhà tuyển dụng cũng có thể đăng các mục quảng cáo về các khoá học và các dịch vụ của công ty mình trên website.
* NTD có thể ngay lập tức tìm kiếm ứng viên với các lựa chọn thích hợp mà không phải mất thời gian đăng tuyển.
  + - * 1. **Usecase Đăng nhập vào hệ thống**

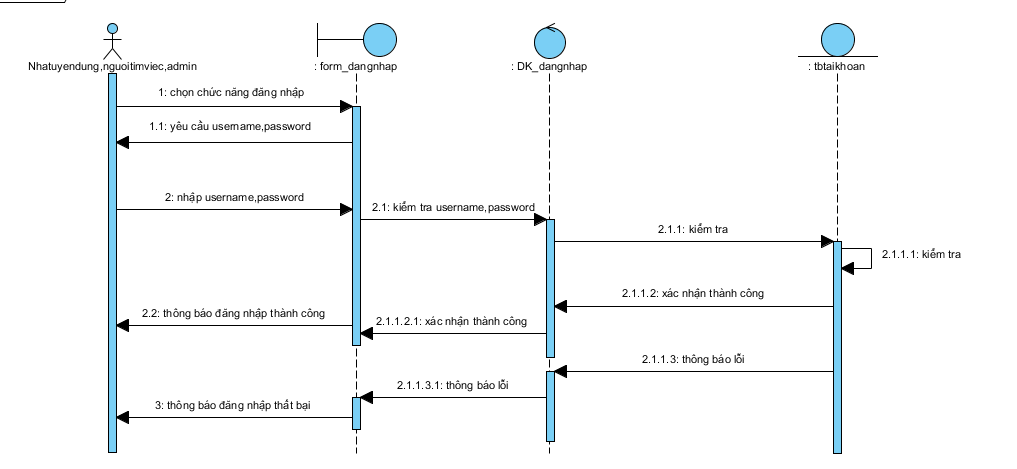
**A. Use case DangNhap**

1. Chức năng:
   * Use case này mô tả cách người dùng đăng nhập vào hệ thống với các vai trò khác nhau (nhà quản trị web, nhà tuyển dụng, ứng viên, khách hàng…)
2. Dòng sự kiện (hệ thống yêu cầu gì và sẽ thực hiên như thế nào):
   * + Dòng sự kiện chính:
   * Use case này bắt đầu khi một actor muốn đăng nhập vào hệ thống.

* Các bước thực hiện:
  + Khi người dùng chọn chức năng đăng nhập, hệ thống sẽ hiện thị trang đăng nhập, hệ thống sẽ yêu cầu người dùng nhập các thông tin đăng nhập (username và password) và nhấn nút đăng nhập.
  + Hệ thống sẽ kiểm tra các thông tin đăng nhập và quyết định có cho người dùng đăng nhập hay không.
    - Dòng sự kiện khác
  + Tên/mật khẩu bị sai
  + Nếu trong quá trình đăng nhập, người dùng nhập sai tên hoặc mật khẩu, hệ thống sẽ hiện thị một thông báo lỗi, người dùng có thể chọn nhập lại hoặc huỷ bỏ việc đăng nhập. Nếu huỷ bỏ, use case kết thúc.
    - 1. Điều kiện tiên quyết:
  + Không có

1. Post-Conditions:
   * Nếu use case này thành công, người dùng sẽ đăng nhập vào hệ thống. Nếu không trạng thái hệ thống không thay đổi.

**B. Sequence diagram DangNhap:**



Hình 6. Sequence diagram đăng nhập

* + - * 1. **Usecase Đăng tin**

**A. Use-case DangTin:**

a. Chức năng:

* Cung cấp chức năng cho nhà tuyển dụng đăng tuyển dụng với những yêu cầu về TDVH, chứng chỉ… Cung cấp chức năng cho nhà tuyển dụng đăng tuyển dụng với những yêu cầu về TDVH, chứng chỉ…

b. Dòng sự kiện (*Hệ thống yêu cầu gì và sẽ thực hiện như thế nào*).

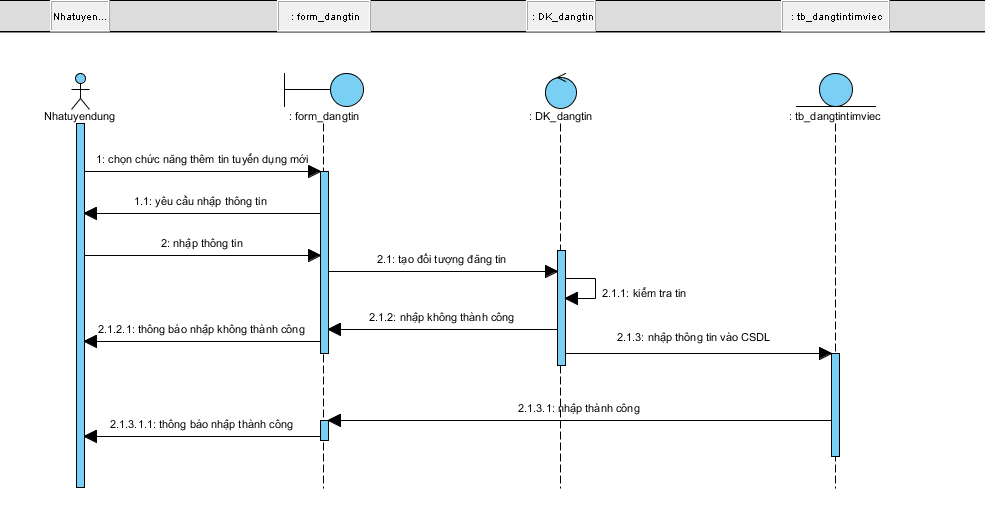
* + Dòng sự kiện chính
    - Use case này bắt đầu khi nhà tuyển dụng muốn đăng tuyển công việc.
      * Các bước thực hiện
* Hệ thống yêu cầu đăng nhập trước khi thực hiện tác vụ khác.
* Đăng nhập thành công, hệ thống yêu cầu nhập vào form đăng tuyển với các thông tin cần thiết (Hệ thống đưa ra các yêu cầu bắt buộc phải nhập, khi nhập sai sẽ được cảnh báo).
* Đăng tuyển thành công, hệ thống sẽ hiển thị trong form danh sách công việc mới nhất.
  + Dòng sự kiện khác
* Nếu trong quá trình đăng tuyển nhà tuyển dụng chọn chức năng huỷ bỏ thì việc đăng tuyển sẽ kết thúc và trạng thái của hệ thống không thay đổi.

c. Điều kiện tiên quyết.

* Nhà tuyển dụng phải đăng nhập trước khi thực hiện chức năng này.

1. Post-Conditions.
   * Nếu use case này thành công thì thông tin đăng tuyển của nhà tuyển dụng sẽ được cập nhật vào cơ sở dữ liệu và hệ thống sẽ tự động thống kê các ứng viên phù hợp cho công việc vừa được đăng, nếu không thành công thì trạng thái của hệ thống không thay đổi.

B. **Sequence diagram DangTin:**

****

Hình 7. Sequence diagram đăng tin tuyển dụng

* + - * 1. **Usecase Xóa thông tin**

**A**. **Use case XoaTTTuyenDung**

a. Chức năng:

* Cho phép nhà tuyển dụng xoá thông tin đã đăng.

b. Dòng sự kiện (*hệ thống yêu cầu gì và sẽ thực hiện như thế nào*)

* Use case này bắt đầu khi nhà tuyển dụng muốn xoá một mẫu tin đã đăng.
  + Các bước thực hiện:
* Nhà tuyển dụng đăng nhập vào hệ thống.
* Nhà tuyển dụng chọn một mẫu tin muốn xoá, nhấp “Delete”.

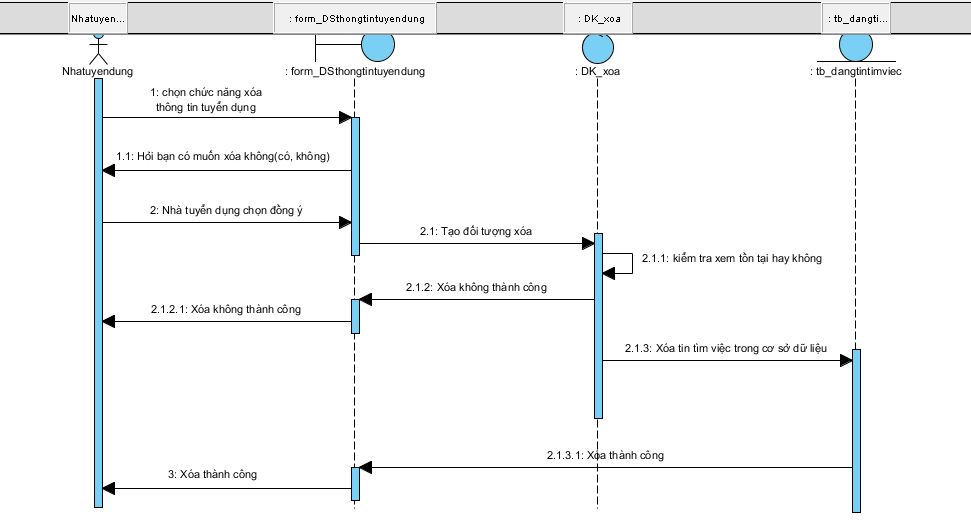
c. Điều kiện tiên quyết

* Nhà tuyển dụng phải đăng nhập trước khi xoá.

d. Post – Condition:

* Nếu use case này thành công thì hệ thống sẽ mất đi một mẫu tin.

**B. Sequence diagram XoaTTTuyenDung**

****

Hình 8. Sequence diagram xóa tin tuyển dụng

* + - * 1. **Usecase chỉnh sửa thông tin**

**A. Use case ChinhSuaThongTin**

* + - * 1. Chức năng:
* Cho phép nhà tuyển dụng chỉnh sửa thông tin đăng tuyển nếu các thông tin đó có sự thay đổi.
  + - * 1. Dòng sự kiện (*Hệ thống yêu cầu gì và sẽ thực hiện như thế nào*)

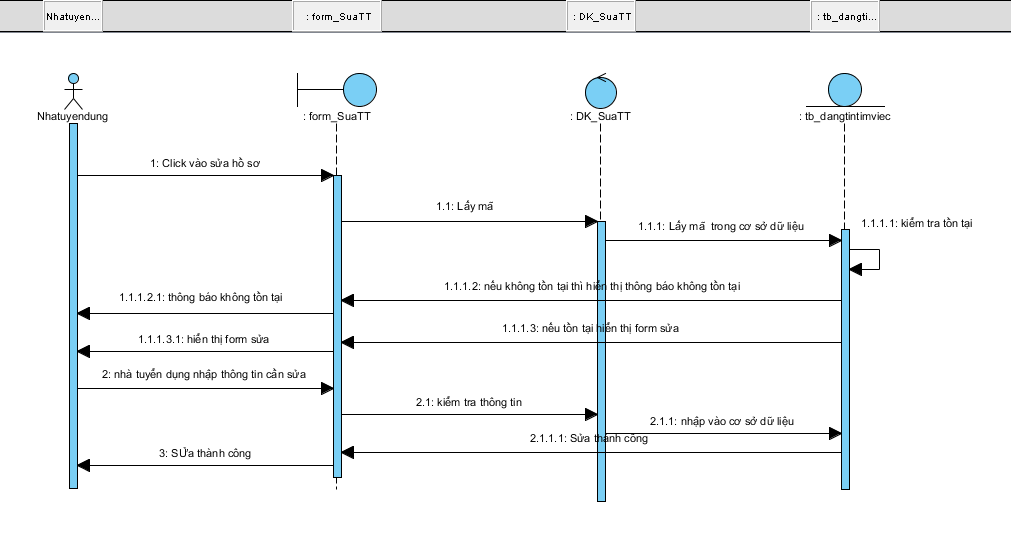
Dòng sự kiện chính

* Use case này bắt đầu khi nhà tuyển dụng muốn chỉnh sửa các thông tin đã đăng.
* Nhà tuyển dụng nhấp “chỉnh sửa thông tin”.
* Hệ thống yêu cầu đăng nhập trước khi chỉnh sửa thông tin.
* Sau khi đăng nhập thành công hệ thống sẽ hiển thị các thông tin về nhà tuyển dụng đó và nhà tuyển dụng sẽ nhập vào các thông tin cần chỉnh sửa.
* Chỉnh sửa thông tin thành công, hệ thống sẽ cập nhật các thông tin mới nhất này.

Dòng sự kiện khác:

* Nếu trong quá trình chỉnh sửa thông tin, nhà tuyển dụng chọn chức năng huỷ bỏ thì việc chỉnh sửa thông tin và trạng thái của hệ thống sẽ không thay đổi.
  + - * 1. Điều kiện tiên quyết:
* Nhà tuyển dụng phải đăng nhập trước khi chỉnh sửa thông tin.
  + - * 1. Post – Condition:
* Nếu use case này thành công thì hệ thống sẽ tự động cập nhật các thông tin mới nhất. Nếu không thành công thì trạng thái hệ thống sẽ không thay đổi.

**B. Sequence diagram ChinhSuaThongTin:**



Hình 9. Sequence diagram chỉnh sửa thông tin

* + - * 1. **Tìm hồ sơ ứng viên**

**A. Use case TimHoSoUngVien**

* + - * 1. Chức năng:
    - Cung cấp chức năng cho nhà tuyển dụng tìm kiếm những ứng viên phù hợp với mục tiêu tuyển dụng.

Dòng sự kiện (*Hệ thống yêu cầu gì và sẽ thực hiện như thế nào*)

* + - Dòng sự kiện chính

Use case này bắt đầu khi nhà tuyển dụng muốn tìm kiếm thông tin về các ứng viên theo những tiêu chí nhất định.

Các bước thực hiện:

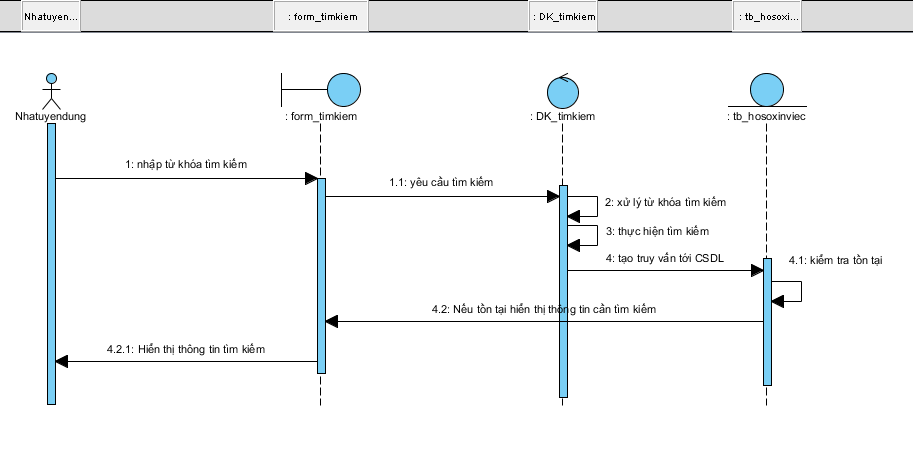
* Nhà tuyển dụng nhấp nút tìm kiếm.
* Hệ thống yêu cầu đăng nhập trước khi tìm kiếm hồ sơ ứng viên
* Khi đăng nhập thành công hệ thống sẽ yêu cầu nhập vào thông tin cần thiết mà nhà tuyển dụng cần tìm kiếm.
* Sau khi tìm thấy, hệ thống sẽ hiển thị tất cả các thông tin đó cho nhà tuyển dụng xem.
  + - Dòng sự kiện phụ:
* Nếu thông tin mà nhà tuyển dụng cần tìm kiếm không có trong cơ sở dữ liệu thì hệ thống sẽ thông báo không tìm thấy

1. Điều kiện tiên quyết:

* Nhà tuyển dụng phải đăng nhập nếu muốn tìm kiếm thông tin ứng viên.
  1. Post – Conditions:
* Nếu việc tìm kiếm thành công thì hệ thống sẽ hiển thị các thông tin tìm thấy trên trang tìm kiếm. Ngược lại hiển thị kết quả tìm kiếm thất bại.

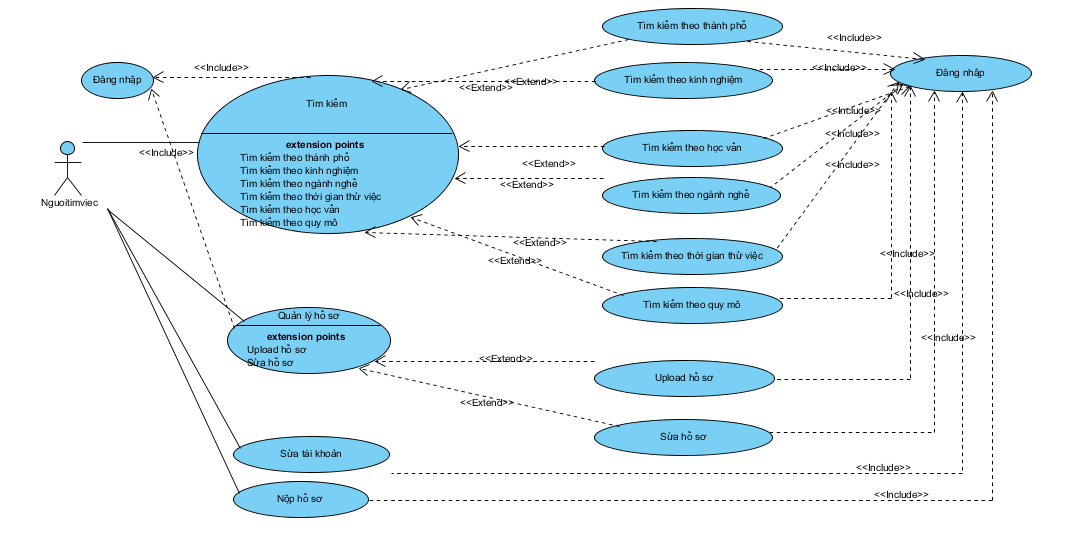
**1.1.2.6.1. Use-case Tim\_Kiem:**

**-** Tìm kiếm theo chuyên ngành, thành phố, nơi làm việc,…



Hình 10. Tìm kiếm hồ sơ xin việc

* + - 1. **Use-case diagram – đối với nhóm chức năng thuộc người tìm việc.**

****

Hình 11. Mô hình usecase với nhóm chức năng thuộc người tìm việc

* + - * 1. **Actors:**
* NguoiTimViec (Người Tìm Việc): Nguời tìm việc có vai trò quan trọng trong hệ thống, bao gồm tìm việc, đăng hồ sơ, nộp hồ sơ, nhận thông báo việc làm, nhận kỹ năng xin việc, hỏi tư vấn kỹ năng xin việc.
* Người tìm việc có thể đăng hồ sơ để tìm kiếm công việc thích hợp với mình mà không phải mất thời gian để đi đến các trung tâm xin việc nộp hồ sơ.
* Người tìm việc có thể tìm kiếm các công việc phù hợp với ngành nghề, chuyên môn…
* Người tìm việc có thể hỏi và nhận các kỹ năng xin việc làm hành trang cho mình trong quá trình đi tìm việc.
* Ngoài ra, họ còn nhận được các thông báo trúng tuyển nếu hồ sơ của họ phù hợp với yêu cầu của nhận tuyển dụng.
  + - * 1. **Usecase Đăng hồ sơ**

**A. Use-case DangHoSo**

1. Chức năng:

* Cung cấp cho người xin việc các thức đăng một mẫu tin tìm việc.

1. Dòng sự kiện (*Hệ thống yêu cầu gì và sẽ thực hiện ntn*):
   * Dòng sự kiện chính:
     + Use case này bắt đầu khi ứng viên muốn đăng thông tin tìm việc
       - Các bước thực hiện:

* Ứng viên nhấp đăng hồ sơ.
* Hệ thống yêu cầu đăng nhập trước khi thực hiện tác vụ khác.
* Đăng nhập thành công, hệ thống yêu cầu nhập vào các thông tin cần thiết (Hệ thống đưa ra các yêu cầu bắt buộc phải nhập, khi nhập sai sẽ được cảnh báo).
* Đăng ký công việc thành công, hệ thống sẽ hiển thị trong ra danh sách cá ứng viên một cách chi tiết.
  + Dòng sự kiện khác:
* Ứng viên không nhập đủ các thông tin mà trang web yêu cầu.
* Trong quá trình đăng hồ sơ, ứng viên chọn chức năng huỷ bỏ thì việc đăng hồ sơ thất bại và trạng thái hệ thống sẽ không thay đổi.

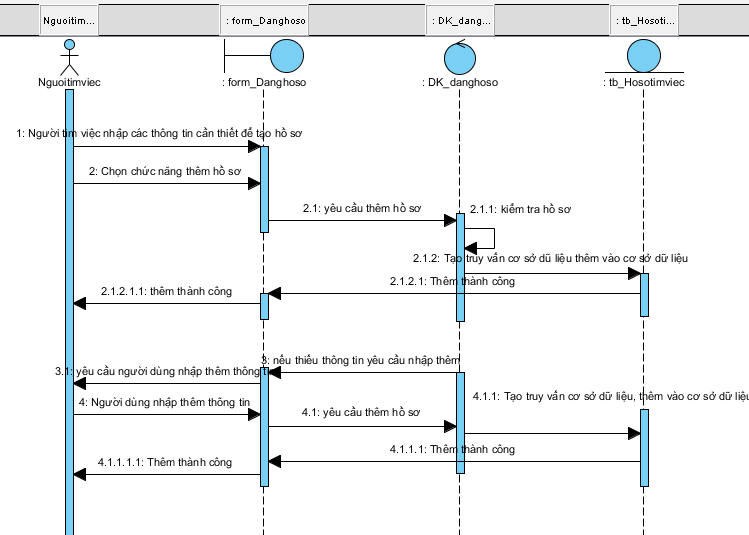
1. Điều kiện tiên quyết:

* Người tìm việc phải đăng nhập trước khi thực hiện chức năng này.

1. Post-Conditions:

* Nếu việc đăng hồ sơ thành công thì hệ thống sẽ cập nhật các thông tin về ứng viên đó vào cơ sở dữ liệu và nhà tuyển dụng sẽ có thêm một ứng cử viên để lựa chọn.

**B. Sequence diagram DangHoSo**



Hình 12. Mô hình Sequence diagram Đăng hồ sơ

* + - * 1. **Chỉnh sửa hồ sơ**

1. **Use case Chinhsuahoso**
   1. Chức năng:

* Cho phép ứng viên sửa chữa, thay đổi thông tin về bản thân đã đăng trước đó.
  1. Dòng sự kiện (*Hệ thống yêu cầu gì và sẽ thực hiện ntn*):
  + Dòng sự kiện chính:
    - Use case này bắt đầu khi ứng viên muốn hiệu chỉnh các thông tin đã đăng.
      * Các bước thực hiện:
    - Ứng viên chọn chức năng hiệu chỉnh hồ sơ.
    - Hệ thống yêu cầu đăng nhập trước khi thực hiện hiệu chỉnh hồ sơ.
    - Sau khi đăng nhập thành công hệ thống sẽ hiển thị các thông tin về mà ứng cử viên đã đăng trước đó và ứng viên sẽ nhập vào các thông tin cần chỉnh sửa.
    - Chỉnh sửa thông tin thành công, hệ thống sẽ cập nhật các thông tin mới nhất này.
  + Dòng sự kiên phụ:
    - Nếu trong quá trình chỉnh sửa thông tin, nhà tuyển dụng chọn chức năng huỷ bỏ thì việc chỉnh sửa thông tin thất bại và trạng thái của hệ thống sẽ không thay đổi.
  1. Điều kiện tiên quyết:
     + Ứng viên phải đăng nhập trước khi hiệu chỉnh thông tin.
  2. Post – Condition:
     + Nếu use case này thành công thì hệ thống sẽ tự động cập nhật các thông tin mới nhất. Nếu không thành công thì trạng thái hệ thống sẽ không thay đổi.

**B. Sequence diagram HieuChinhHoSo**

* + - * 1. **Xóa hồ sơ**

**A. Use case XoaHoSo**

a. Chức năng:

* Cho phép người tìm việc xoá thông tin của bản thân đã đăng trước đó.

b.Dòng sự kiện (hệ thống yêu cầu gì và sẽ thực hiện như thế nào)

* Người tìm việc đăng nhập vào hệ thống.
* Chọn hồ sơ đã đăng và nhấp “Delete”.

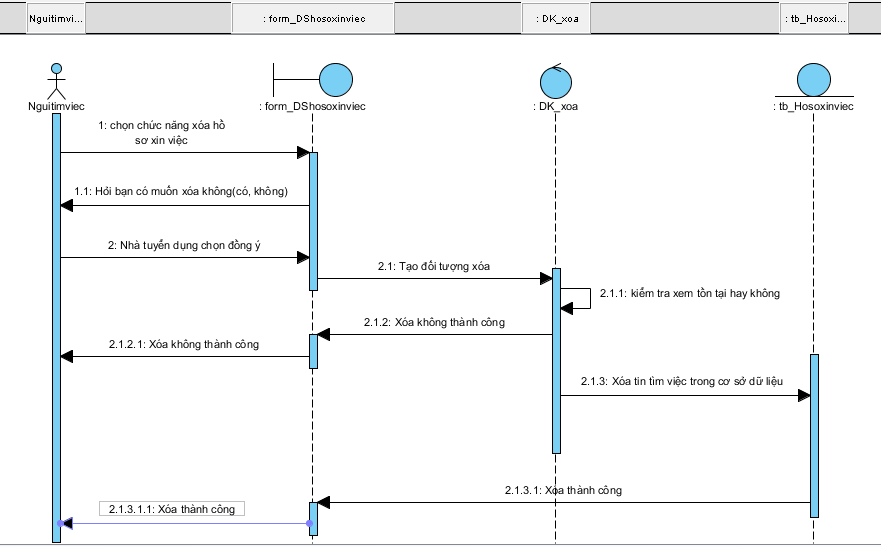
c. Điều kiện tiên quyết

* Hệ thống yêu cầu đăng nhập trước khi xoá hồ sơ

d. Post – Condition:

* Hệ thống sẽ mất đi một hồ sơ nếu use case này thành công

**B. Sequence diagram XoaHoSo()**



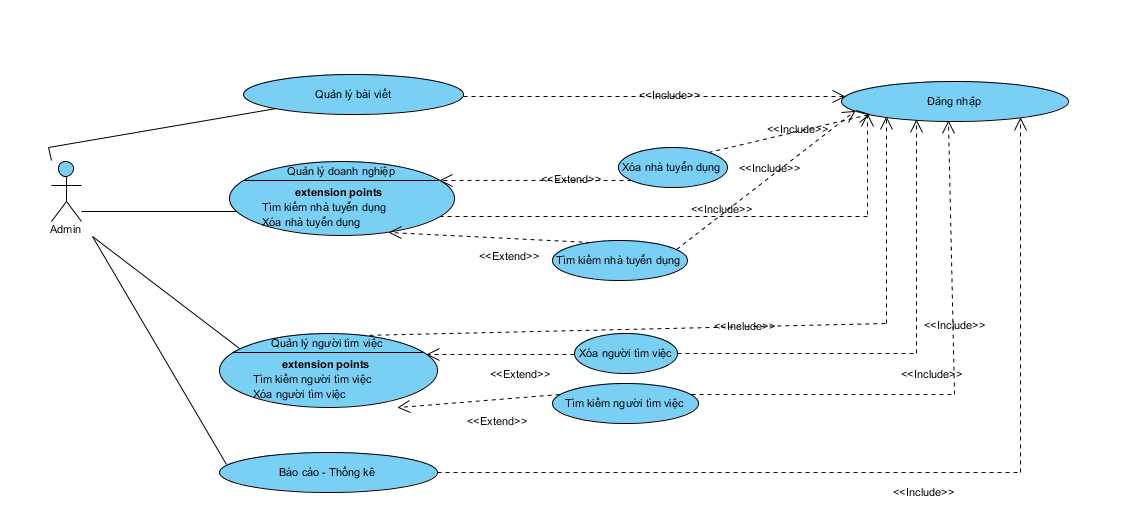
Hình 13. Sequence diagram Xóa hồ sơ

* + - * 1. **Nộp hồ sơ trực tiếp nhà tuyển dụng**

**A. Use case NopHoSoTrucTiepNTD:**

* + 1. Chức năng:
* Cho phép ứng viên gửi mail trực tiếp đến nhà tuyển dụng nếu thấy công việc phù hợp với bản thân.

1. Dòng sự kiện (*Hệ thống yêu cầu gì và sẽ thực hiện ntn*):
   * Ứng cử viên nhấn gửi mail cho nhà tuyển dụng để tham gian sơ tuyển trực tiếp mà không cần đăng tin tìm việc trên trang web.
2. Điều kiện tiên quyết:
   * Không có
3. Post – Condition:
   * Không có.
     + 1. **Usecase diagram – đối với chức năng thuộc người quản trị**



Hình 14. Mô hình usecase đối với chức năng thuộc người quản trị

* + - * 1. **Actors:**
* Người quản trị (NguoiQuanTri): người quản trị có vai trò quan trọng trong hệ thống, bao gồm quản trị diễn đàn, thống kê danh sách trúng tuyển, xoá thông tin tuyển dụng, sơ khớp điều kiện tuyển dụng và gửi danh sách các ứng cử viên cho nhà tuyển dụng.
* Người quản trị quản lý diễn đàn, giải đáp thắc mắc đối của các thành viên.
* Người quản trị thống kê số lượng các ứng cử viên được lựa chọn sau khi nhận được thông báo của nhà tuyển dụng.
* Người quản trị có thể xoá thông tin tuyển dụng nếu mẫu tin đó tồn tại quá lâu.
  + - * 1. **Tìm kiếm nhà tuyển dụng**

**A. Usecase TimKiemNTD:**

Chức năng:

Cho phép người quản trị tìm kiếm thông tin của nhà tuyển dụng

Dòng sự kiện (*Hệ thống yêu cầu gì và sẽ thực hiện ntn*):

* + Nhà tuyển dụng nhập vào các tiêu chuẩn tìm kiếm
  + Nếu tìm thấy hệ thống sẽ hiển thị các thông tin.

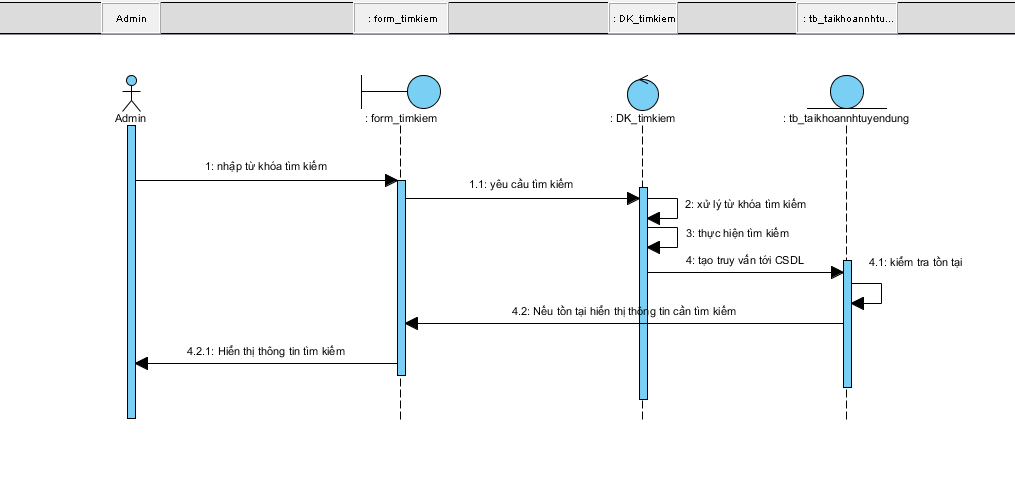
Điều kiện tiên quyết:

* + Người quản trị phải đăng nhập trước khi thực hiện tác vụ này.

Post – Condition:

* + Nếu use case này thành công hệ thống sẽ hiển thị tất cả các thông tin theo yêu cầu tìm kiếm.
  + Nếu không thành công, hệ thống không thay đổi

1. **Sequence diagram TimKiemNTD**



* + - * 1. **Xóa nhà tuyển dụng**

**A. Usecase XoaNTD:**

Chức năng:

Cho phép người quản trị xóa các thông tin của nhà tuyển dụng.

Dòng sự kiện (*Hệ thống yêu cầu làm gì và sẽ thực hiện ntn*):

Người quản trị chọn nhà tuyển dụng cần xóa.

Chọn “Delete”.

Điều kiện tiên quyết:

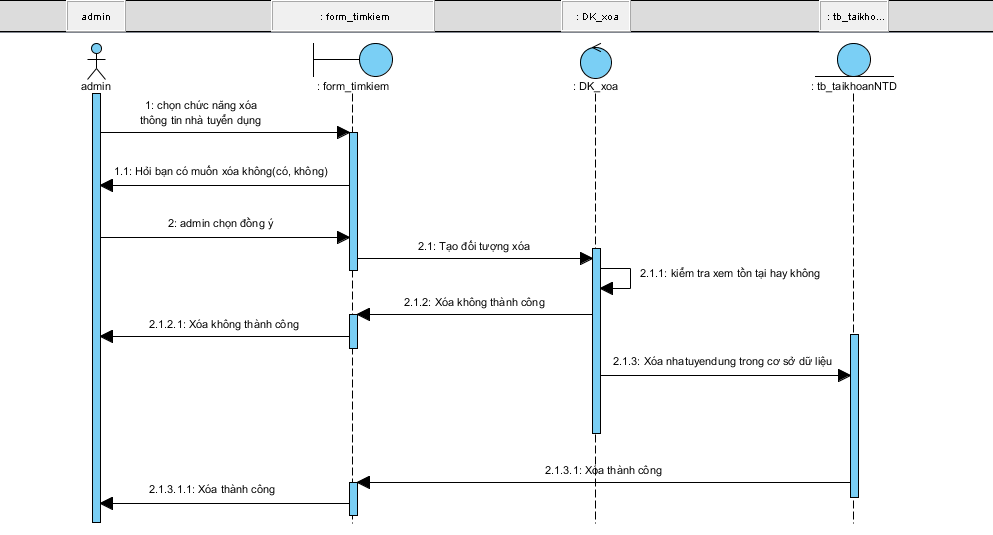
Người quản trị phải đăng nhập trước khi thực hiện tác vụ này.

Post – Condition:

Nếu use case này thành công hệ thống sẽ mất đi thông tin của nhà tuyển dụng đã chọn.

Nếu không thành công, hệ thống không thay đổi

**B. Sequence diagram XoaNTD**

****

Hình . Sequence diagram Xóa nhà tuyển dụng

* + - * 1. **Xóa người tìm việc**

**A. Usecase XoaNTV**

Chức năng:

Cho phép người quản trị xóa các thông tin của người tìm việc.

Dòng sự kiện (*Hệ thống yêu cầu làm gì và sẽ thực hiện ntn*):

Người quản trị chọn người tìm việc cần xóa.

Chọn “Delete”.

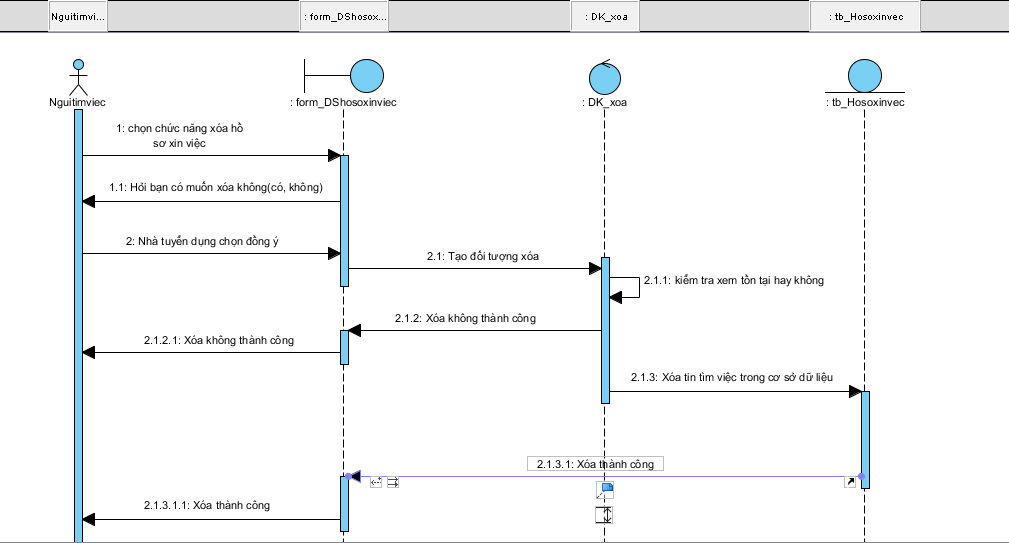
Điều kiện tiên quyết:

* + Người quản trị phải đăng nhập trước khi thực hiện tác vụ này.

Post – Condition:

* + Nếu use case này thành công hệ thống sẽ mất đi thông tin của người tìm việc đã chọn.
  + Nếu không thành công, hệ thống không thay đổi

**B. Sequence diagram XoaNTV**



Hình 16. Sequence diagram Xóa người tìm việc

* + - * 1. **Tìm kiếm người tìm việc**

**A. Usecase TimKiemNTV**

1. Chức năng:

Cho phép người quản trị tìm kiếm thông tin của người tìm việc

1. Dòng sự kiện (*Hệ thống yêu cầu gì và sẽ thực hiện ntn*):

Nhà tuyển dụng nhập vào các tiêu chuẩn tìm kiếm

Nếu tìm thấy hệ thống sẽ hiển thị các thông tin.

1. Điều kiện tiên quyết:

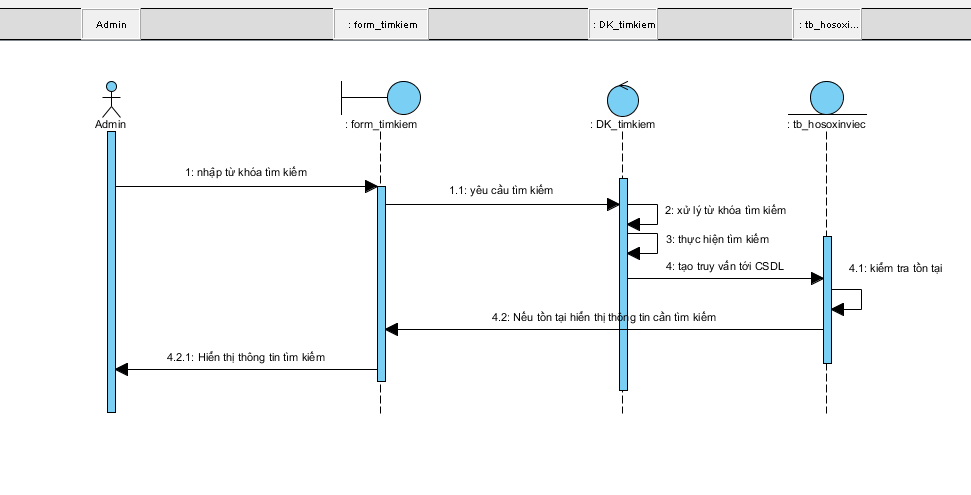
Người quản trị phải đăng nhập trước khi thực hiện tác vụ này.

1. Post – Condition:

Nếu use case này thành công hệ thống sẽ hiển thị tất cả các thông tin theo yêu cầu tìm kiếm.

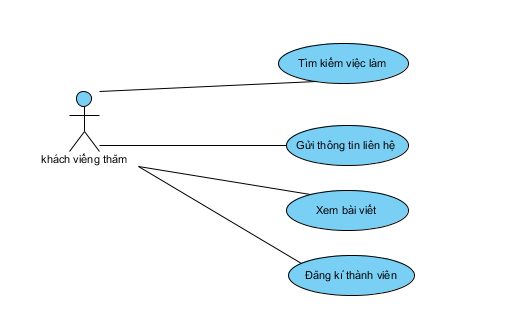
Nếu không thành công, hệ thống không thay đổi

**B. Sequence diagram TimKiemNTV**

****

Hình 17. Sequence diagram Tìm kiếm nhà tuyển dụng

* + - 1. **Usecase diagram – đối với chức năng thuộc khách viếng thăm.**



Hình 18. Mô hình - Usecase khách viếng thăm

* + - * 1. **Đăng kí thành viên**

**A. Usecase DangKiThanhVien:**

1. Chức năng:
   * Use case này cho phép người sử dụng đăng kí với trang web để trở thành thành viên (có thể là nhà tuyển dụng hay ứng viên).
2. Dòng sự kiện (*Hệ thống yêu cầu gì và sẽ thực hiện ntn*):
   * + - Dòng sự kiện chính:
   * Use case này bắt đầu khi người dùng muốn trở thành thành viên của trang web hoặc khi họ muốn sử dụng dịch vụ khi chưa có account.
     + - * Các bước thực hiện:
   * Người dùng chọn chức năng đăng kí thành viên.
   * Hệ thống sẽ yêu cầu nhập các thông tin (username và password)
   * Hệ thống tiến hành kiểm tra tính hợp lệ của thông tin nhập vào và lưu vào cơ sở dữ liệu.
     + - Dòng sự kiện khác:
   * Nếu trong quá trình đăng kí người dùng quyết định huỷ bỏ các thông tin đăng kí thì usecase kết thúc và trạng thái hệ thống không thay đổi.
   * Nếu quá trình kiểm tra phát hiện thông tin nhập vào không hợp lệ thì hệ thống trở về trạng thái trước đó (mở trang nhập liệu với các thông tin nhập trước đó vẫn còn để người dùng có thể chỉnh sửa).
3. Điều kiện tiên quyết:
   * Không có.
4. Post – Condition:
   * Nếu qua trình đăng kí thành công thì các thông tin của người dùng sẽ được lưu vào cơ sở dữ liệu. Nếu không trạng thái của hệ thống không hề thay đổi.
     + - 1. **Tìm kiếm việc làm**

**A. Use case TimViec**

* + - 1. Chức năng:
* Cung cấp chức năng cho ứng viên tìm kiếm những công việc phù hợp với khả năng bằng các từ khoá xác định.
  1. Dòng sự kiện (*Hệ thống yêu cầu gì và sẽ thực hiện ntn*)
  + Dòng sự kiện chính:
* Use case này bắt đầu khi ứng viên muốn tìm kiếm các công việc trên trang web.
  + - * Các bước thực hiện:
* Ứng viên chọn chức năng tìm kiếm.
* Hệ thống yêu cầu nhập vào thông tin cần tìm kiếm.
* Sau khi tìm thấy, hệ thống sẽ hiển thị tất cả các thông tin đó cho người tìm việc.
  + Dòng sự kiện phụ:
* Nếu thông tin mà nhà tuyển dụng cần tìm kiếm không có trong cơ sở dữ liệu thì hệ thống sẽ thông báo không tìm thấy

1. Điều kiện tiên quyết:

* Không có

1. Post – Conditions:

* Nếu việc tìm kiếm thành công thì hệ thống sẽ hiển thị các thông tin tìm thấy trên trang tìm kiếm. Ngược lại hiển thị kết quả tìm kiếm thất bại.
  + - * 1. **Gửi thông tin liên hệ**

**A. Usecase GuiThongTinLienLac:**

* 1. Chức năng:
  + Cho phép người dùng gửi các thông tin đến người quản trị.
    1. Dòng sự kiện (*Hệ thống yêu cầu gì và sẽ thực hiện ntn*):
       - Dòng sự kiện chính:
       - Usecase này bắt đầu khi người dùng muốn gửi thông tin liên hệ đến người quản trị.
         * Các bước thực hiện:
       - Người dùng chọn chức năng liên hệ.
       - Hệ thống yêu cầu nhập thông tin liên hệ.
       - Hệ thống sẽ tiến hành lưu thông tin của người dùng vào cơ sở dữ liệu.
       - Dòng sự kiện khác:
* Nếu trong quá trình liên hệ, người dùng thực hiện huỷ bỏ thì việc liên hệ sẽ thất bại và trạng thái của hệ thống sẽ không đổi.

Điều kiện tiên quyết:

* + Không có.

Post – Condition:

* + Nếu qua trình liên hệ thành công thì các thông tin cần liên hệ của người dùng sẽ được lưu vào cơ sở dữ liệu. Nếu không trạng thái của hệ thống không hề thay đổi.
    - * 1. **Xem bài viết**

**A. Use case Xembaiviet:**

1. Chức năng:

* Cung cấp các thông tin về các kỹ năng cần thiết khi đi xin việc: từ cách thức làm hồ sơ, đến cách trả lời phỏng vấn…
* Cung cấp cho ứng viên các thông tin về việc làm mới nhất của trang web.
  1. Dòng sự kiện:

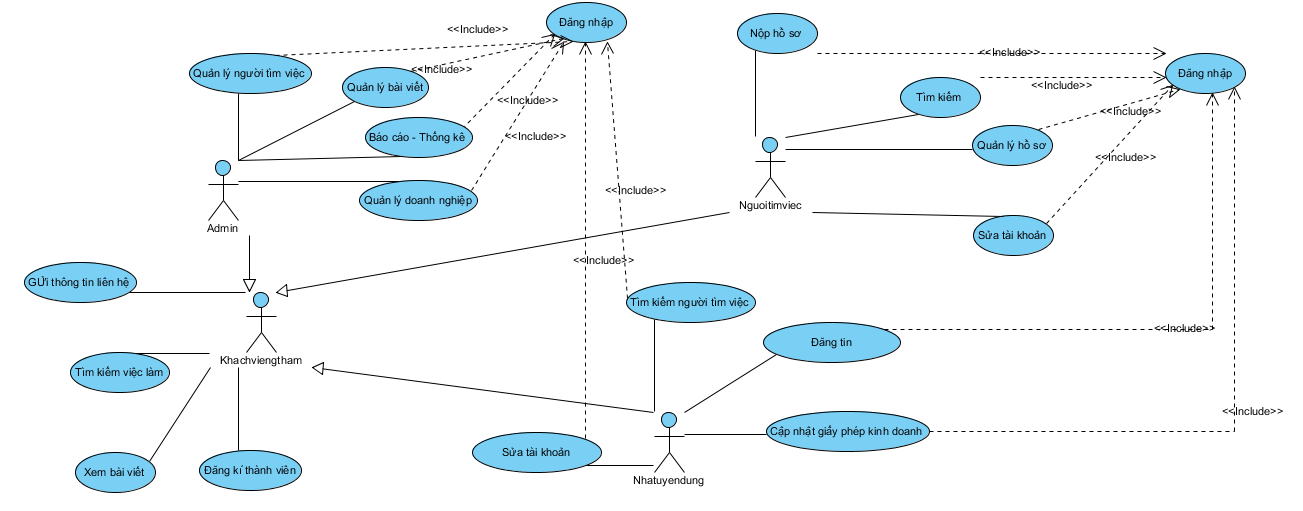
Hệ thống sẽ gửi các thông tin về các kỹ năng xin việc, hướng nghiệp.. về địa chỉ email của người tìm việc.

1. Điều kiện tiên quyết:

* Không có

1. Post – Conditions.

* Không có
  + - 1. **Usecase tổng quát**

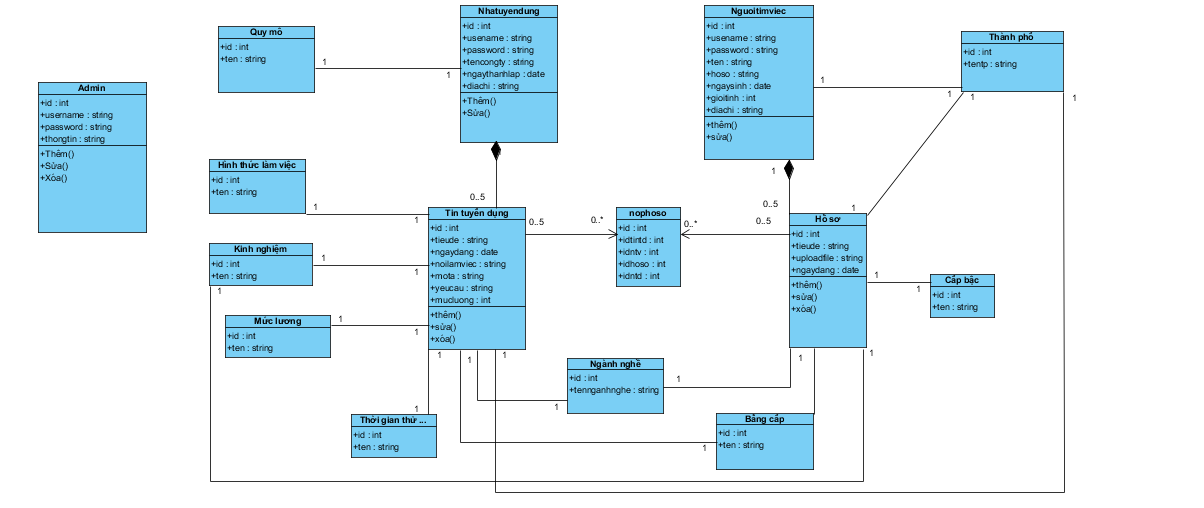
****

Hình 19. Mô hình usecase tổng quát

* + 1. **Class diagram**

**Class Diagram là một trong những bản vẽ quan trọng nhất của thiết kế phần mềm, nó cho thấy cấu trúc và quan hệ giữa các thành phần tạo nên phần mềm. Từ đó làm cơ sở thiết kế cơ sở dữ liệu.**

**Từ các luồng dữ liệu bên trên, tôi đưa ra biểu đồ class diagram như sau:**

****

Hình 20. Biểu đồ lớp

* 1. **Cơ sở dữ liệu**

Từ biểu đồ lớp bên trên, tôi đưa ra được các bảng dữ liệu:

* + 1. **Bảng Admin(Người quản lý).**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên cột | Ý nghĩa | Kiểu | Độ rộng |
| id | Mã người quản lý | int | 11 |
| hoten | Họ tên nhà quản lý | varchar | 100 |
| email | Địa chỉ email | varchar | 100 |
| password | Mật khẩu người quản lý | varchar | 100 |
| ngaysinh | Ngày sinh | date |  |
| anh | ảnh người quản lý | varchar | 100 |
| gioitinh | Giới tính | int | 1 |
| diachi | Địa chỉ | varchar | 100 |

* + 1. **Bảng Nhatuyendung**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên cột | Ý nghĩa | Kiểu | Độ rộng |
| id | Mã nhà tuyển dụng | int | 11 |
| tenNTD | Tên nhà tuyển dụng(tên công ty) | varchar | 100 |
| email | Địa chỉ email | varchar | 100 |
| password | Mật khẩu người quản lý | varchar | 100 |
| ngaysinh | Ngày sinh | date |  |
| logo | Logo nhà tuyển dụng | varchar | 100 |
| giayphekd | Giâý phép kinh doanh | varchar | 100 |
| gioitinh | Giới tính | int | 1 |
| diachi | Địa chỉ | varchar | 100 |
| Sodt | Số điện thoại | int |  |
| tenNLH | Tên người liên hệ | varchar | 100 |
| sodtNLH | Số điện thoại người liên hệ | int |  |
| emailNLH | Địa chỉ email người liên hệ | varchar | 100 |

* + 1. **Bảng Tintuyendung**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên cột | Ý nghĩa | Kiểu | Độ rộng |
| id | Mã tin tuyển dụng | int | 11 |
| tieude | Tiêu đề tin tuyển dụng | varchar | 100 |
| mota | Mô tả công việc | varchar | 500 |
| yeucau | Yêu cầu kĩ năng | varchar | 500 |
| quyenloi | Quyền lợi | Varchar | 500 |
| ngayhethan | Ngày hết hạn | date | 100 |
| noilamviec | Nơi làm việc | varchar | 100 |
| idtinhchatcv | Mã tính chất công việc | int | 11 |
| idbangcap | Mã bằng cấp | int | 11 |
| idkinhnghiem | Mã kinh nghiệm | int | 11 |
| idmucluong | Mã mức lương | int | 11 |
| idhinhthuccv | Mã hình thức công viẹc | int | 11 |
| idthoigianthuviec | Mã thời gian thử việc | int | 11 |
| idnganhnghe | Mã ngành nghề | int | 11 |

* + 1. **Bảng Nguoitimviec**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên cột | Ý nghĩa | Kiểu | Độ rộng |
| id | Mã người tìm việc | int | 11 |
| hoten | Tên người tìm việc | varchar | 100 |
| email | Email người tìm việc | varchar | 500 |
| password | Mật khẩu đăng nhập người tìm việc | varchar | 100 |
| gioitinh | Giới tính người tìm việc | int | 1 |
| ngaysinh | Ngày sinh người tìm việc | date |  |
| diachi | Địa chỉ | varchar | 100 |

* + 1. **Bảng Hoso**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên cột | Ý nghĩa | Kiểu | Độ rộng |
| id | Mã hồ sơ | int | 11 |
| tieude | Tiểu đề hồ sơ | varchar | 100 |
| mucluong | Mức lương | int |  |
| upload | Upload file CV xin việc | varchar | 100 |
| ngaydang | Ngày đăng | date | 1 |
| idthanhpho | Mã thành phố | int | 11 |
| idkinhnghiem | Mã kinh nghiệm | int | 11 |
| idcapbac | Mã cấp bậc | int | 11 |
| idnganhnghe | Mã ngành nghề | int | 11 |

* + 1. **Bảng nộp hồ sơ**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên cột | Ý nghĩa | Kiểu | Độ rộng |
| id | Mã nộp hồ sơ | int | 11 |
| idNTD | Mã nhà tuyển dụng | int | 11 |
| idNTV | Mã người tìm việc | int | 11 |
| idhoso | Mã hồ sơ | int | 11 |
| idTinTD | Mã tin tuyển dụng | int | 11 |

* + 1. **Bảng mucluong( mức lương)**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên cột | Ý nghĩa | Kiểu | Độ rộng |
| id | Mã mức lương | int | 11 |
| mucluong | Mức lương | varchar | 100 |

* + 1. **Bảng quymo( quy mô)**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên cột | Ý nghĩa | Kiểu | Độ rộng |
| id | Mã mức lương | int | 11 |
| quymo | Quy mô | varchar | 100 |

* + 1. **Bảng capbac(cấp bậc)**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên cột | Ý nghĩa | Kiểu | Độ rộng |
| id | Mã mức lương | int | 11 |
| capbac | Cấp bậc | varchar | 100 |

* + 1. **Bảng tinhchatcv(tính chất công việc)**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên cột | Ý nghĩa | Kiểu | Độ rộng |
| id | Mã tính chất công việc | int | 11 |
| tinhchatcv | Tính chất công việc | varchar | 100 |

* + 1. **Bảng hinhthuccv(hình thức công việc)**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên cột | Ý nghĩa | Kiểu | Độ rộng |
| id | Mã tính chất công việc | int | 11 |
| hinhthuccv | Hình thức công việc | varchar | 100 |

* + 1. **Bảng kinhnghiem( kinh nghiệm)**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên cột | Ý nghĩa | Kiểu | Độ rộng |
| id | Mã tính chất công việc | int | 11 |
| kinhnghiem | Kinh nghiệm | varchar | 100 |

* + 1. **Bảng bangcap( bằng cấp)**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên cột | Ý nghĩa | Kiểu | Độ rộng |
| id | Mã tính chất công việc | int | 11 |
| bangcap | Bằng cấp | varchar | 100 |

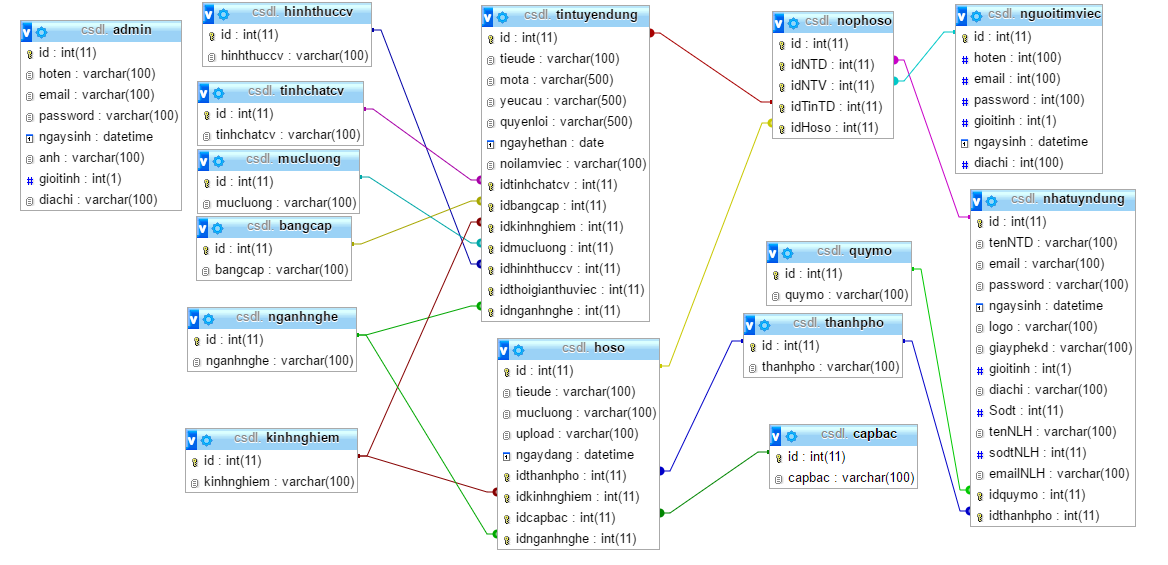
* + 1. **Bảng thanhpho( thành phố)**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên cột | Ý nghĩa | Kiểu | Độ rộng |
| id | Mã tính chất công việc | int | 11 |
| thanhpho | Thành phố | varchar | 100 |

* + 1. **Bảng nganhnghe(ngành nghề)**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên cột | Ý nghĩa | Kiểu | Độ rộng |
| id | Mã tính chất công việc | int | 11 |
| nganhnghe | Ngành nghề | varchar | 100 |

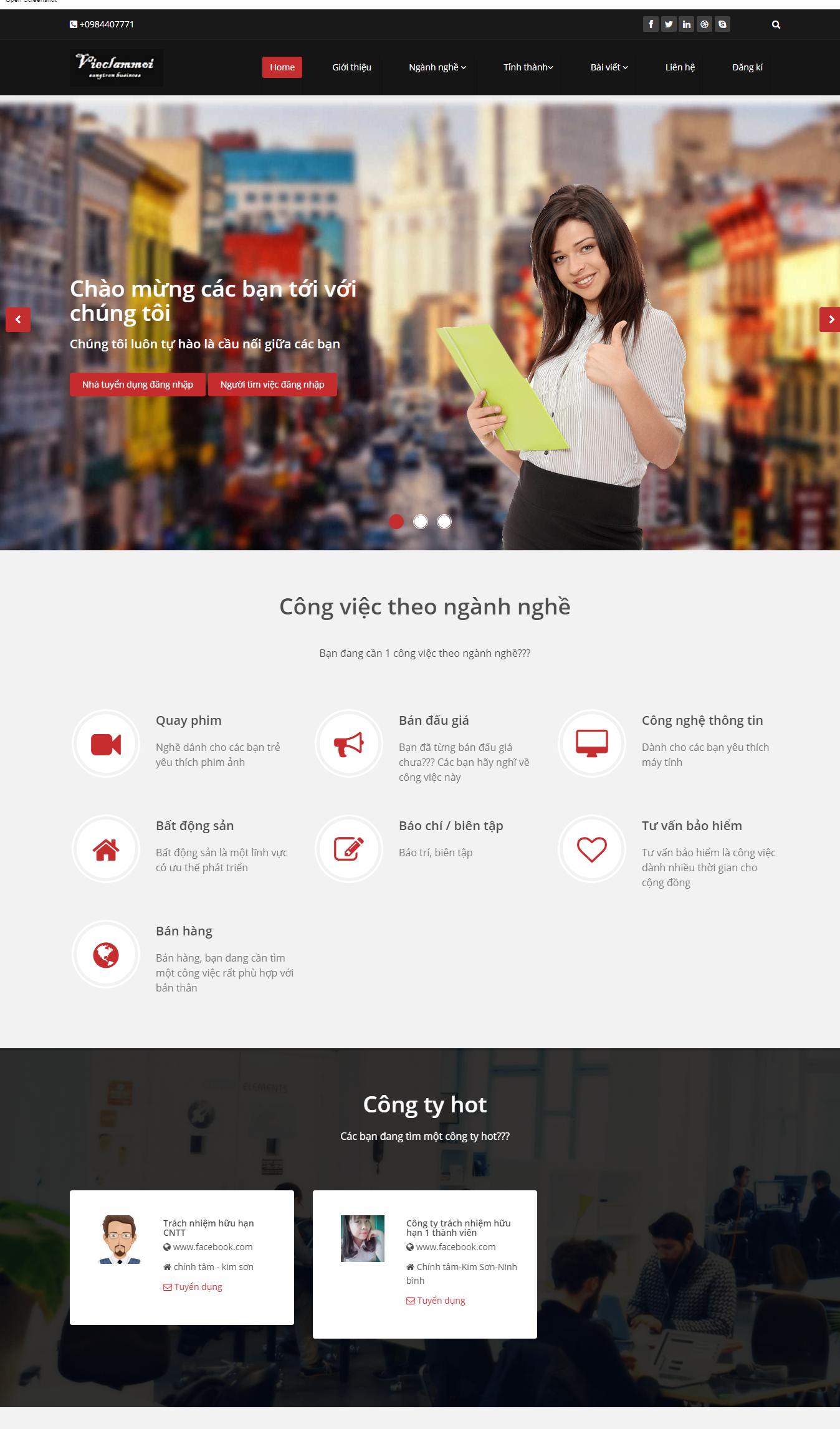
**Từ các bảng biểu diễn trên tôi đưa ra mô hình liên kết cơ sở dữ liệu**

****

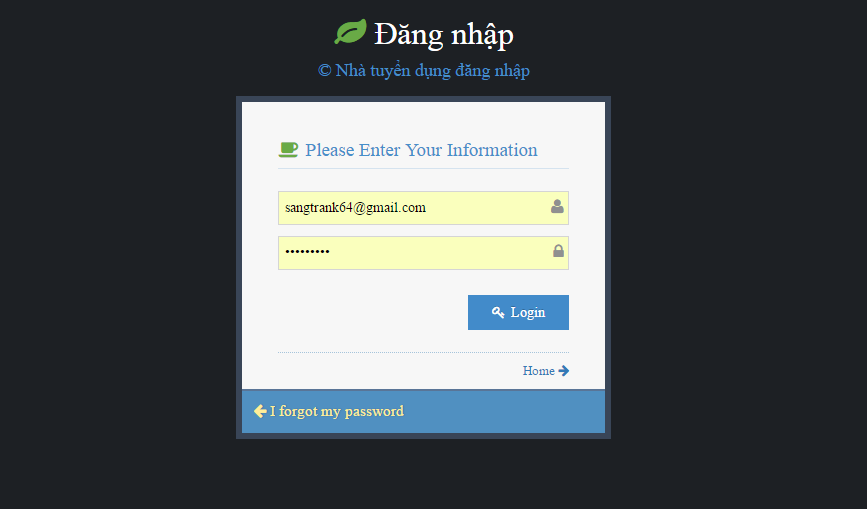
Hình 21. Mô hình cơ sở dữ liệu quan hệ

# CHƯƠNG 3: WEBSITE TUYỂN DỤNG VÀ TÌM VIỆC LÀM

* 1. **Trang chủ**

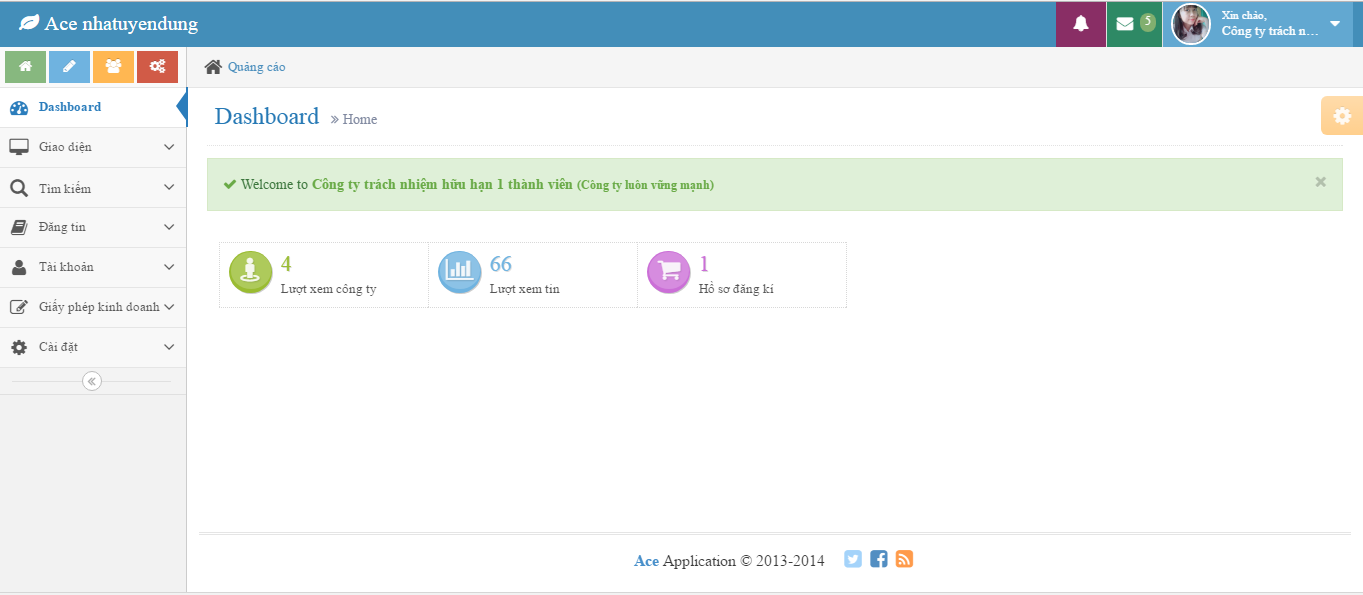
****

* 1. **Trang nhà tuyển dụng**
* Muốn vào trang nhà tuyển dụng thì phải đăng nhập

****

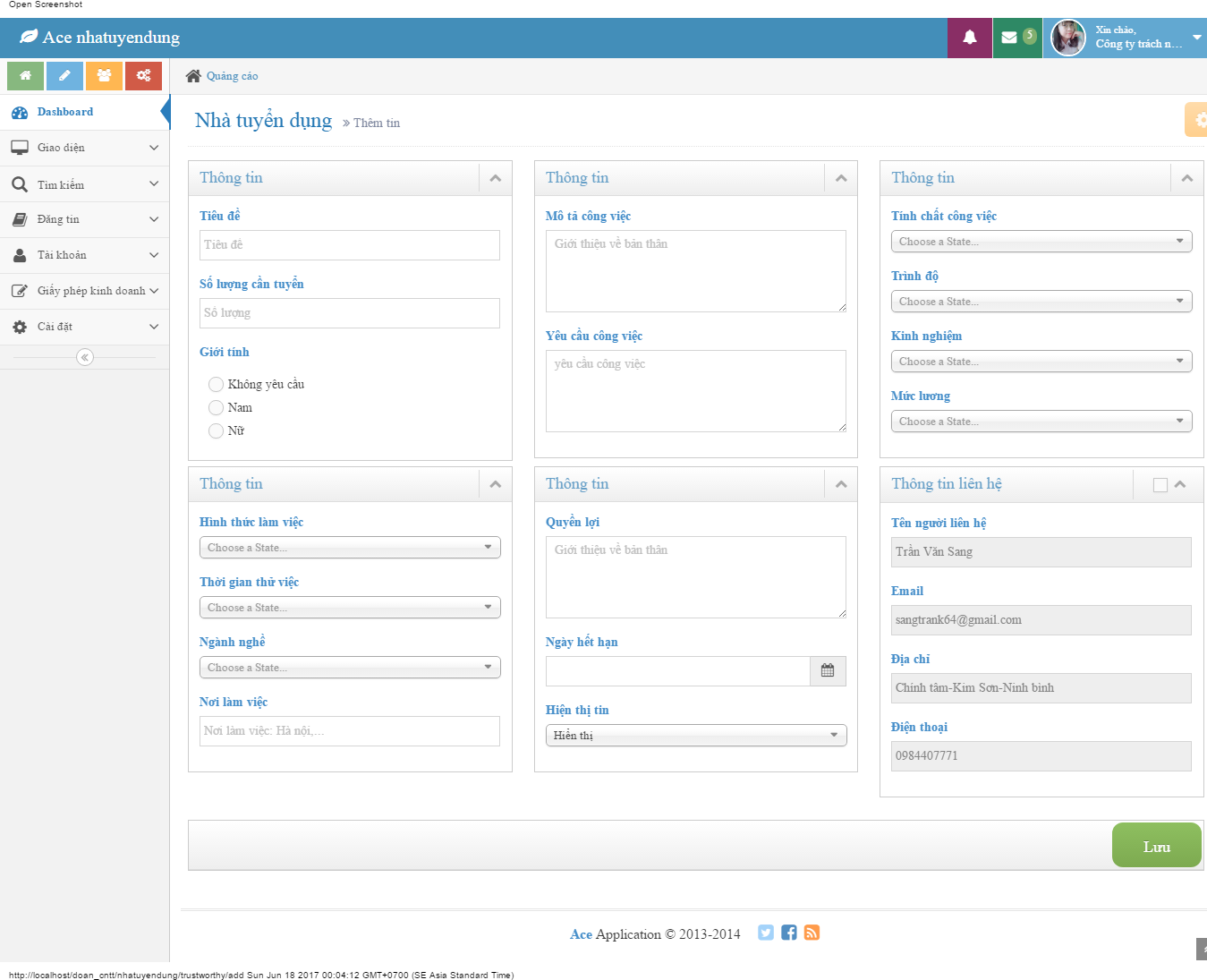
Hình 22. Đăng nhập

* + 1. **Giao diện người dùng sau khi đăng nhập**

****

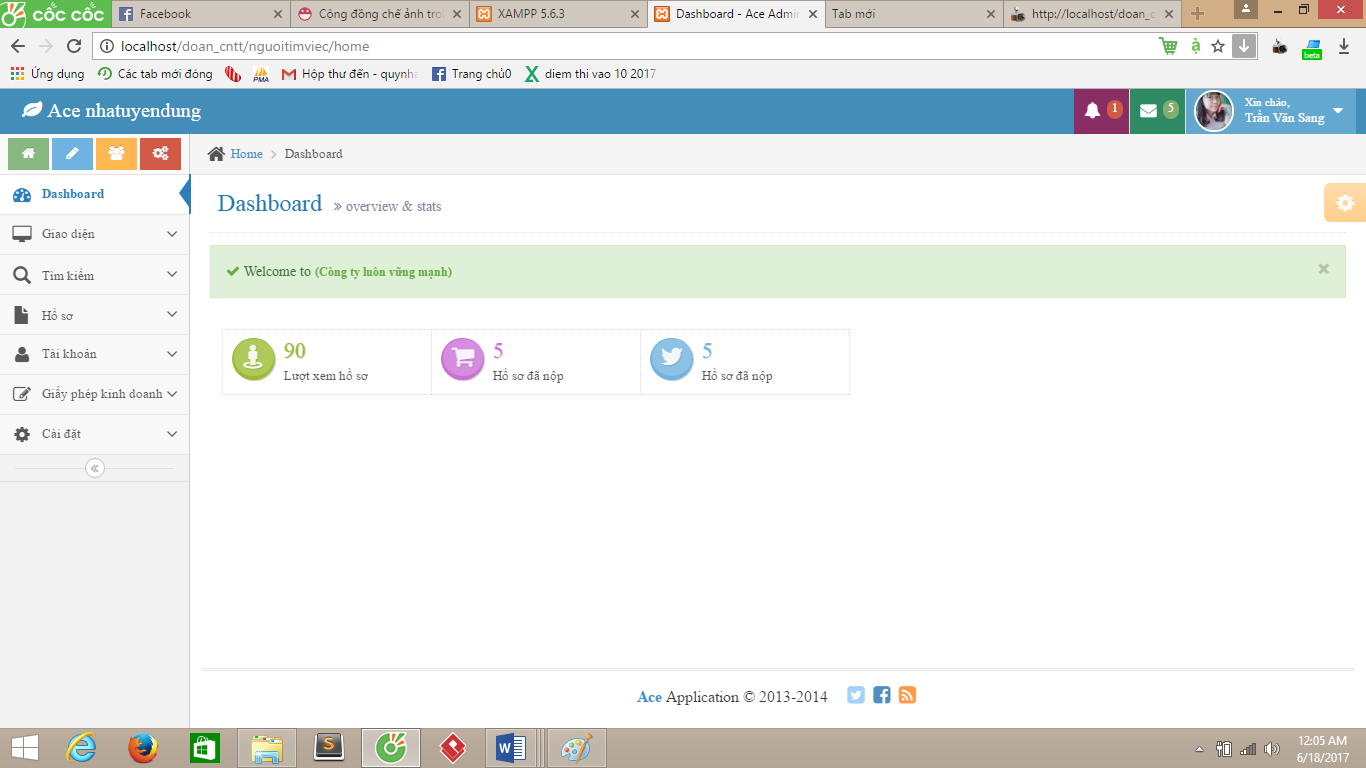
Hình 23. Giao diện người dùng sau khi đăng nhập

* + 1. **Đăng tin**
* Sau khi đăng nhập người dùng có thể thực hiện đăng tin tuyển dụng



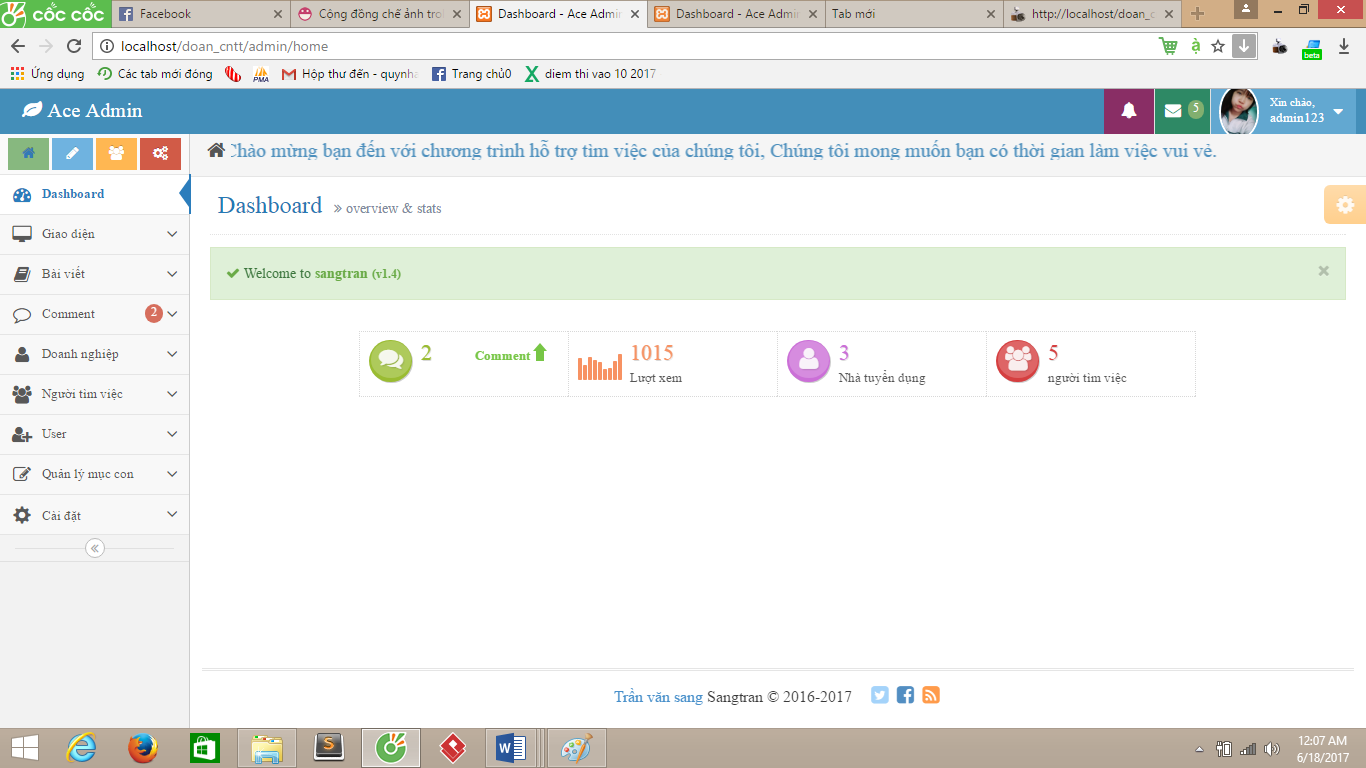
Hình 24. Giao diện đăng tin

* 1. **Trang người tìm việc**



Hình 25. Giao diện trang người tìm việc

* 1. **Trang quản trị**



Hình 26. Giao diện trang quản trị

# KẾT LUẬN VÀ KIẾN NGHỊ

## A. Kết luận

* Xây dựng một ứng dụng thương mại sử dụng công nghệ lập trình mới, hiện đại là một điều thú vị song không ít những thách thức. Thứ nhất, mặc dù thương mại điện tử không còn xa lạ gì nhưng đối với chúng em, thời gian tiếp cận không thể nói là nhiều. Thứ hai, đây là lần đầu tiên chúng em thực sự là việc với Codeigniter Framework để xây dựng một ứng dụng web. Hơn nữa, thời gian làm luận văn cũng có giới hạn, nên chúng em cũng gặp nhiều khó khăn. Tuy nhiên, chúng em đã cố gắng rất nhiều để hoàn thành luận văn đúng thời hạn. Sau đây là những phần chúng em đã hoàn thành trong luận văn (mặc dù còn rất nhiều ý tưởng và dự định chưa thể thực hiện được do thời gian và khả năng có hạn, hy vọng có thể hoàn thiện trong thời gian tới).
* Giao diện được xây dựng hoàn chỉnh, tương ứng với từng người dùng có các thanh menu riêng, chứa các khả năng có thể thực hiện.
* Với đối tượng là người tìm việc có thể thực hiện các chức năng sau:
  + Đăng kí là thành viên
  + Đăng nhập vào hệ thống (nếu đã đăng kí).
  + Đăng hồ sơ xin việc
  + Xem các hồ sơ mình đã đăng, có thể chỉnh sửa thông tin của hồ sơ xin việc nếu thấy cần thiết.
  + Tìm kiếm công việc:
    - Tìm kiếm công việc: tìm theo các thông tin mở rộng
      * Tìm theo thời gian tuyển dụng
      * Tìm theo thông tin công việc bao gồm: chuyên ngành, tên công việc, mức lương khởi điểm, yêu cầu kinh nghiệm, lương khởi điểm, nơi làm việc
      * Tìm theo tên nhà tuyển dụng
  + Gửi mail trực tiếp đến cho nhà tuyển dụng nếu thấy công việc phù hợp.
  + Xem các kỹ năng xin việc.
  + Đăng kí email để nhận các thông báo việc làm hàng ngày, hàng tuần
  + Tham gia diễn đàn.
  + Tham gia góp ý, liên hệ với người quản trị mạng.
* Với đối tượng là nhà tuyển dụng có thể thực hiện các chức năng sau:
* Đăng kí là nhà tuyển dụng
* Đăng nhập vào hệ thống (nếu đã đăng kí), có hỗ trợ chức năng tìm lại mật khẩu khi quên mật khẩu và có thể đổi mật khẩu.
* Đăng tuyển dụng.
* Chỉnh sửa thông tin đã đăng
* Đăng quảng cáo
* Tìm kiếm ứng viên
  + Tìm kiếm: tìm theo các thông tin mở rộng
    - Tìm theo chuyên ngành
    - Tìm theo trình độ văn hoá
    - Tìm theo thành phố
    - Tìm theo giới tính
    - Tìm theo số năm kinh nghiệm
    - Tìm theo mức lương yêu cầu
    - Tìm theo tên chứng chỉ
* Góp ý kĩ năng xin việc.
* Với đối tượng là nhà quản trị có thể thực hiện các chức năng sau:
* Quản lý nhà tuyển dụng
* Quản lý người tìm việc
* Các yêu cầu về bảo mật: Sử dụng kiến trúc bảo mật codeigniter framework
* Khả năng tương thích với các trình duyệt khác nhau: Chạy tốt trên các trình duyệt khác nhau.
* Tuy nhiên, do thời gian và năng lực có hạn, nên đề tài của chúng em còn một vài thiếu sót:
* Về giao diện: do hạn chế về khả năng thẩm mỹ nên giao diện chưa thật sự thân thiện với người dùng
* Về nghiệp vụ: một số nghiệp vụ còn chưa hoàn chỉnh, chưa thực hiện được việc liên kết với hệ thống ngân hàng để tiến hành thanh toán tự động.
* Về nghiệp vụ của Admin: do thời gian hạn chế nên phần Admin còn sơ sài.

## B. Kiến nghị

* + Đề tài có thể mở rộng và phát triển theo các hướng sau:
    - Liên kết với hệ thống ngân hàng:
      * Hiện nay, hệ thống ngân hàng ở nước ta chưa phổ biến hình thức thanh toán tín dụng qua mạng. Do đó, nếu trong tương lai dịch vụ này phát triển thì việc liên kết với các ngân hàng để thực hiện quá trình thanh toán trực tuyến được nhanh chóng và hiệu quả hơn
    - Liên kết với các nhà tuyển dụng để thực hiện và đăng kí tự động nếu nhà tuyển dụng có nhu cầu muốn đăng quảng cáo trên trang web.
      * Điều này chưa thực hiện được do hai vấn đề sau: thứ nhất, công nghệ trong nước chưa thực hiện được. Thứ 2, trình độ của nhóm còn hạn chế.
    - Đưa vào triển khai ứng dụng thực tế:
      * Trong tương lai, nếu có khả năng, nhóm sẽ đưa ứng dụng vào triển khai thực tế để đem lại cho mọi người một địa chỉ đáng tin cậy

# DANH MỤC TÀI LIỆU THAM KHẢO

**Tiếng Việt**

[1] Phân tích thiết kế hướng đối tượng PGS.TS Đặng Văn Đức

[2] Trang phát triển codeigniter: <https://codeigniter.com>

[3] Trang học codeigniter: <https://freetuts.net>

[4] Trang học codeigniter: <http://hocphp.info/codeigniter/>

[5] Trang tìm kiếm việc làm: <https://www.timviecnhanh.com/>

[6] Trang tìm kiếm việc làm: <https://vieclam24h.vn/>