**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC MỞ THÀNH PHỐ HỒ CHÍ MINH**

**NGUYỄN TRƯƠNG ANH MINH**

**PHẠM MINH HOÀNG**

**ĐỀ TÀI THIẾT KẾ WEBSITE TUYỂN DỤNG H&M**

**DỰA TRÊN CÔNG NGHỆ SPRING MVC**

**ĐỒ ÁN NGÀNH**

**NGÀNH KHOA HỌC MÁY TÍNH**

**TP. HỒ CHÍ MINH, 2022BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC MỞ THÀNH PHỐ HỒ CHÍ MINH**

**ĐỀ TÀI THIẾT KẾ WEBSITE TUYỂN DỤNG H&M**

**DỰA TRÊN CÔNG NGHỆ SPRING MVC**

**NGUYỄN TRƯƠNG ANH MINH: 1951012071**

**PHẠM MINH HOÀNG: 1951012031**

**ĐỒ ÁN NGÀNH**

**NGÀNH KHOA HỌC MÁY TÍNH**

**Giảng viên hướng dẫn: ThS. Nguyễn Thị Mai Trang**

**TP. HỒ CHÍ MINH, 2022**

LỜI CÁM ƠN

Trước hết, em chân thành cảm ơn tất cả quý Thầy Cô khoa Công Nghệ Thông Tin, Trường Đại Học Mở Thành phố Hồ Chí Minh đã tạo điều kiện thuận lợi cho em trong suốt quá trình học tập và nghiên cứu đề tài này. Đồ án ngành là một minh chứng cho những kiến thức đã có sau gần bốn năm học tập.

Đặc biệt, em chân thành cảm ơn cô Nguyễn Thị Mai Trang người đã tận tình hướng dẫn, chỉ bảo em trong suốt quá trình làm báo cáo. Cô đã truyền đạt vốn kiến thức, kinh nghiệm, tâm huyết của cô đến chúng em và cả thời gian quý báu để giúp chúng em có thể hoàn thành báo cáo của mình. Nhờ có những lời hướng dẫn, dạy bảo ấy nên đề tài nghiên cứu của em mới có thể hoàn thiện tốt đẹp.

Qua đây, em cũng gửi lên cảm ơn đến anh chị, bạn bè đã tiếp thêm động lực để em có thể phát huy hết khả năng và nâng cao trình độ chuyên môn của bản thân.

Trong quá trình thực hiện Đồ Án Ngành, nhận thấy mình đã cố gắng hết sức nhưng vì kiến thức vẫn còn hạn hẹp nên vẫn còn nhiều thiếu sót, mong thầy cô bổ sung để Đồ án được hoàn thiện tốt hơn cũng như em có thể rút ra và tích lũy thêm được những kinh nghiệm quý báu cho bản thân, điều này sẽ giúp ích được nhiều cho em sau này.

Em xin chân thành cảm ơn.

NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

**MỤC LỤC**

[LỜI CÁM ƠN 2](#_Toc117754559)

[NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN 3](#_Toc117754560)

[DANH MỤC TỪ VIẾT TẮT 7](#_Toc117754561)

[DANH MỤC HÌNH VẼ 8](#_Toc117754562)

[DANH MỤC BẢNG 10](#_Toc117754563)

[MỞ ĐẦU 11](#_Toc117754564)

[Chương 1. Tổng quan về đề tài 12](#_Toc117754565)

[1.1. Giới thiệu tổng quan 12](#_Toc117754566)

[1.2. Lý do chọn đề tài 12](#_Toc117754567)

[1.3. Tóm tắt như cầu để thiết kế Website tuyển dụng 12](#_Toc117754568)

[1.3.1. Những khó khăn khi xây dựng hệ thống tuyển dụng 13](#_Toc117754569)

[1.3.2. Các yêu cầu cơ bản cần có trong chương trình 13](#_Toc117754570)

[1.4. So với các website tuyển dụng hiện nay 14](#_Toc117754571)

[1.5. Phương pháp cần lưu ý khi phát triển một website 14](#_Toc117754572)

[1.6. Tầm quan trọng của việc thiết kế website tuyển dụng 14](#_Toc117754573)

[Chương 2. Cơ sở lý thuyết 16](#_Toc117754574)

[2.1. Giới thiệu về Spring MVC. 16](#_Toc117754575)

[2.2. Giới thiệu về API. 16](#_Toc117754576)

[2.3. JSP – JavaServer Pages. 16](#_Toc117754577)

[2.4. Phương thức giao tiếp HTTP Request – HTTP Response. 17](#_Toc117754578)

[2.4.1. HTTP Request 17](#_Toc117754579)

[2.4.2. HTTP Response 18](#_Toc117754580)

[2.5. Xác thực và phân quyền người dùng với Spring Security. 18](#_Toc117754581)

[2.6. Controller trong SpringMVC. 20](#_Toc117754582)

[2.6.1. Giới thiệu về Controller. 20](#_Toc117754583)

[2.6.2. Luồng hoạt động của Controller. 20](#_Toc117754584)

[2.6.3. Tiếp nhận request trong Controller. 21](#_Toc117754585)

[2.6.4. Tiếp nhận dữ liệu từ request trong Controller. 21](#_Toc117754586)

[2.7. Chuyển đổi dữ liệu Serialization. 22](#_Toc117754587)

[2.8. Một số annotation quan trọng trong SpringMVC. 22](#_Toc117754588)

[2.9. Giới thiệu về Cloudinary. 23](#_Toc117754589)

[2.10. Đa ngôn ngữ. 23](#_Toc117754590)

[2.11. Hibernate. 24](#_Toc117754591)

[2.12. Apache Tomcat 26](#_Toc117754592)

[Chương 3. Khảo sát - Mô tả thực trạng – Thiết kế hệ thống 27](#_Toc117754593)

[3.1. Khảo sát và hoạch định hệ thống 27](#_Toc117754594)

[3.1.1. Đối tượng khảo sát 27](#_Toc117754595)

[3.1.2. Các câu hỏi phỏng vấn 27](#_Toc117754596)

[3.2. Khảo sát bằng phương pháp bảng câu hỏi 27](#_Toc117754597)

[3.3. Kết quả khảo sát 28](#_Toc117754598)

[3.3.1. Các chức năng của hệ thống tuyển dụng cần có 28](#_Toc117754599)

[3.3.2. Các thông tin hệ thống cần lưu trữ 30](#_Toc117754600)

[3.4. Mô tả nghiệp vụ 30](#_Toc117754601)

[3.4.1. Quy trình tìm việc 30](#_Toc117754602)

[3.4.2. Quy trình tìm ứng viên 31](#_Toc117754603)

[3.5. Xây dựng các chức năng dự kiến 31](#_Toc117754604)

[3.6. Sơ đồ cây chức năng (FDH) 34](#_Toc117754605)

[3.6.1. Sơ đồ cây chức năng Front-End 34](#_Toc117754606)

[3.6.2. Sơ đồ cây chức năng Back-End 35](#_Toc117754607)

[3.7. Mô tả chi tiết chức năng 35](#_Toc117754608)

[3.8. Mô hình thực thể ERD 37](#_Toc117754609)

[3.8.1. Liệt kê các thuộc tính của thực thể 37](#_Toc117754610)

[3.8.2. Các trường hợp liên kết N-N 42](#_Toc117754611)

[3.8.3. Các trường hợp liên kết 1-1. 43](#_Toc117754612)

[3.8.4. Thiết kế các các thuộc tính cho từng bảng. 43](#_Toc117754613)

[3.8.5. Mô hình ERD. 45](#_Toc117754614)

[3.9. Sơ đồ Use Case 46](#_Toc117754615)

[3.9.1. Use Case tổng quát 46](#_Toc117754616)

[3.9.2. Đặc tả Use Case 46](#_Toc117754617)

[3.9.3. Use Case đăng ký tài khoản 48](#_Toc117754618)

[3.9.4. Use Case tạo tin tuyển dụng 49](#_Toc117754619)

[3.9.5. Use Case nộp đơn 50](#_Toc117754620)

[3.9.6. Use Case tạo CV 50](#_Toc117754621)

[3.9.7. Use Case quản lý thông tin ứng viên 51](#_Toc117754622)

[3.9.8. Use Case quản lý thông tin doanh nghiệp 52](#_Toc117754623)

[3.9.9. Use Case thống kê 53](#_Toc117754624)

[3.9.10. Use Case kiểm duyệt thông tin 53](#_Toc117754625)

[3.9.11. Use Case quản lý tài khoản 54](#_Toc117754626)

[Chương 4. Nghiệp vụ hệ thống. 55](#_Toc117754627)

[4.1. Trang chủ. 55](#_Toc117754628)

[4.2. Đăng nhập. 55](#_Toc117754629)

[4.3. Đăng ký. 56](#_Toc117754630)

[4.3.1. Đăng ký ứng viên 56](#_Toc117754631)

[4.3.2. Đăng ký nhà tuyển dụng. 58](#_Toc117754632)

[4.4. Cập nhật thông tin người dùng. 60](#_Toc117754633)

[4.5. Danh sách việc làm. 61](#_Toc117754634)

[4.6. Danh sách ứng viên. 62](#_Toc117754635)

[4.7. Thông tin công ty. 63](#_Toc117754636)

[4.8. Thông tin tuyển dụng. 63](#_Toc117754637)

[4.9. Bình luận. 64](#_Toc117754638)

[4.10. Đăng tin tuyển dụng. 65](#_Toc117754639)

[4.11. Tạo đơn xin việc cho ứng viên. 66](#_Toc117754640)

[4.12. Nộp đơn xin việc cho nhà tuyển dụng. 68](#_Toc117754641)

[4.13. Quản lý. 69](#_Toc117754642)

[4.13.1. Quản lý của công ty. 69](#_Toc117754643)

[4.13.2. Quản lý của ứng viên. 71](#_Toc117754644)

[4.13.3. Quản lý của admin. 72](#_Toc117754645)

[4.14. Các tài khoản để sử dụng hệ thống. 75](#_Toc117754646)

[Chương 5. Tổng kết. 76](#_Toc117754647)

[5.1. Kết luận. 76](#_Toc117754648)

[5.2. Những vấn đề còn tồn tại. 76](#_Toc117754649)

[5.3. Hướng phát triển. 76](#_Toc117754650)

[TÀI LIỆU THAM KHẢO 77](#_Toc117754651)

DANH MỤC TỪ VIẾT TẮT

* CV : Curriculum vitae
* Web : website
* Admin : adminstrator
* App : application
* TH : trường hợp
* MVC: Model-View-Controller
* JSP: JavaServer Pages

DANH MỤC HÌNH VẼ

[Hình 2.1: Mô hình Model-View-Controller. 16](#_Toc117754467)

[Hình 2.2: Vị trí của JSP trong 1 trang web. 17](#_Toc117754468)

[Hình 2.3: Các luồng hoạt động xử lý request của Controller. 20](#_Toc117754469)

[Hình 2.4: Apache Tomcat hổ trợ được nhiều nền tảng ngôn ngữ. 26](#_Toc117754470)

[Hình 3.1: Hình vẽ mối quan hệ kết hợp ERD. 45](#_Toc117754471)

[Hình 3.2: Hình vẽ tổng thể Use Case. 46](#_Toc117754472)

[Hình 4.1: Giao diện trang chủ. 55](#_Toc117754473)

[Hình 4.2: Giao diện đăng nhập. 56](#_Toc117754474)

[Hình 4.3: Giao diện đăng ký. 56](#_Toc117754475)

[Hình 4.4: Báo lỗi tên đăng nhập đã tồn tại. 57](#_Toc117754476)

[Hình 4.5: Báo lỗi mật khẩu không đúng định dạng yêu cầu. 57](#_Toc117754477)

[Hình 4.6: Bổ sung thông tin tài khoản ứng viên. 58](#_Toc117754478)

[Hình 4.7: Giao diện đăng ký nhà tuyển dụng. 59](#_Toc117754479)

[Hình 4.8: Bổ sung thông tin cho nhà tuyển dụng. 60](#_Toc117754480)

[Hình 4.9: Giao diện trang tìm kiếm việc làm. 62](#_Toc117754481)

[Hình 4.10: Giao diện trang tìm kiếm ứng viên. 62](#_Toc117754482)

[Hình 4.11: Giao diện trang thông tin của công ty. 63](#_Toc117754483)

[Hình 4.12: Giao diện trang thông tin tuyển dụng. 63](#_Toc117754484)

[Hình 4.13: Giao diện bình luận. 64](#_Toc117754485)

[Hình 4.14: Giao diện ô nhập bình luận. 64](#_Toc117754486)

[Hình 4.15: Bình luận của bạn vừa được đăng. 65](#_Toc117754487)

[Hình 4.16: Quản lý tài khoản trên thanh điều hướng 65](#_Toc117754488)

[Hình 4.17: Giao diện đăng tin tuyển dụng 66](#_Toc117754489)

[Hình 4.18: Giao diện trang tạo CV. 67](#_Toc117754490)

[Hình 4.19: CV của bạn đã được tải xuống dưới dạng PDF. 67](#_Toc117754491)

[Hình 4.20: Hình ảnh CV của bạn đã được lưu. 68](#_Toc117754492)

[Hình 4.21: Giao diện nộp hồ sơ. 68](#_Toc117754493)

[Hình 4.22: Giao diện lựa chọn CV. 69](#_Toc117754494)

[Hình 4.23: Giao diện nộp CV thành công. 69](#_Toc117754495)

[Hình 4.24: Giao diện chức năng quản lý của công ty. 70](#_Toc117754496)

[Hình 4.25: Khi nhấn vào thông tin ứng viên bất kỳ. 70](#_Toc117754497)

[Hình 4.26: Hình ảnh công việc trước khi xóa. 71](#_Toc117754498)

[Hình 4.27: Hình ảnh công việc sau khi xóa. 71](#_Toc117754499)

[Hình 4.28: Giao diện trang quản lý ứng viên. 71](#_Toc117754500)

[Hình 4.29: Chức năng kiểm duyệt công ty. 72](#_Toc117754501)

[Hình 4.30: chức năng kiểm duyệt tin tuyển dụng. 72](#_Toc117754502)

[Hình 4.31: Chức năng quản lý công ty. 73](#_Toc117754503)

[Hình 4.32: Chức năng quản lý người dùng. 73](#_Toc117754504)

[Hình 4.33: Chức năng quản lý tin tuyển dụng. 74](#_Toc117754505)

[Hình 4.34: Thống kê số lượng tin tuyển dụng. 74](#_Toc117754506)

[Hình 4.35: Thống kê số lượng nộp hồ sơ. 75](#_Toc117754507)

DANH MỤC BẢNG

[Bảng 3.1: Bảng khảo sát bằng câu hỏi. 28](#_Toc117777352)

[Bảng 3.2: Thông tin bảng USER 38](#_Toc117777353)

[Bảng 3.3: Thông tin bảng COMPANY 38](#_Toc117777354)

[Bảng 3.4: Thông tin bảng CANDIDATE 39](#_Toc117777355)

[Bảng 3.5: Thông tin bảng ADMINSTRATOR 39](#_Toc117777356)

[Bảng 3.6: Thông tin bảng JOB 40](#_Toc117777357)

[Bảng 3.7: Thông tin bảng MAJOR 40](#_Toc117777358)

[Bảng 3.8: Thông tin bảng JOB-MAJOR 41](#_Toc117777359)

[Bảng 3.9: Thông tin bảng CURRICULUM-VITAE 41](#_Toc117777360)

[Bảng 3.10: Thông tin bảng CANDIDATE-JOB 42](#_Toc117777361)

[Bảng 3.11: Thông tin bảng COMMENT 42](#_Toc117777362)

[Bảng 3.12: Tất cả use case. 48](#_Toc117777363)

[Bảng 3.13: Use case đăng ký tài khoản. 49](#_Toc117777364)

[Bảng 3.14: Use case tạo tin tuyển dụng. 50](#_Toc117777365)

[Bảng 3.15: Use case nộp hồ sơ. 50](#_Toc117777366)

[Bảng 3.16: Use case tạo CV. 51](#_Toc117777367)

[Bảng 3.17: Use case quản lý thông tin ứng viên. 52](#_Toc117777368)

[Bảng 3.18: Use case quản lý thông tin doanh nghiệp. 52](#_Toc117777369)

[Bảng 3.19: Use case thống kê. 53](#_Toc117777370)

[Bảng 3.20: Use case kiểm duyệt thông tin. 54](#_Toc117777371)

[Bảng 3.21: Use case quản lý tài khoản. 54](#_Toc117777372)

[Bảng 4.1: Các tài khoản để sử dụng hệ thống. 75](#_Toc117777373)

MỞ ĐẦU

Với sự phát triển vượt bậc của Công Nghệ Thông Tin trong đời sống xã hội hiện nay hướng đến thời đại số trên toàn cầu. Các Công ty đã ứng dụng các phần mềm, website vào việc nâng cao quản lý các nghiệp vụ, tăng khả năng tiếp cận khách hàng, quản lý nhân sự, phát triển kinh doanh và xây dựng thương hiệu của mình.

Không thể không nhắc đến một vấn đề của giới trẻ hiện nay đó là khó khăn trong việc tìm kiếm việc làm. Người lao động không xin được việc làm trong khi các doanh nghiệp lại khó tuyển nhân sự.

Hiện nay, nhu cầu tìm việc làm chiếm tỷ lệ ngày càng cao. Sinh viên chính là nguồn lực rất quan trọng vì thế các doanh nghiệp cần biết cách sử dụng một cách hợp lý nhờ vào việc tạo nên các website tuyển dụng để thu hút nguồn nhân lực cho doanh nghiệp. Website Tuyển dụng là một trong những website mà hầu hết các doanh nghiệp nào cũng đều phải đáp ứng.

Chính vì thế, website Tuyển dụng H&M ra đời. Đây là một trang web giúp doanh nghiệp đăng tải những thông tin về mô tả công việc, địa điểm làm việc, yêu cầu ứng viên, chế độ lương thưởng và quyền lợi được hưởng,... Sự kết hợp hài hòa giữa giao diện thân thiện với người dùng và thông tin tìm kiếm nhanh chóng. Chúng cung cấp những thông tin về doanh nghiệp và việc làm một cách cụ thể. Bên cạnh đó, chúng còn cho phép người dùng đăng tải, theo dõi và quản lý CV. Ngoài ra, doanh nghiệp còn có thể liên hệ, trao đổi trực tiếp với ứng viên của mình.

Thực tế, các doanh nghiệp đều sử dụng nhiều phương tiện để đăng tải tìm kiếm việc làm. Tuy nhiên với tính tiện dụng của các công nghệ phần mềm, website Tuyển dụng H&M là sự lựa chọn tốt nhất cho các doanh nghiệp để nâng cao chất lượng ứng viên và dễ dàng tìm kiếm ứng viên thích hợp.

# Tổng quan về đề tài

## Giới thiệu tổng quan

Ngày xưa những vấn đề trong cuộc sống đều được giải quyết bằng phương pháp thủ công, hiện nay công nghệ đang ngày càng trở nên phổ biến, nó được tạo ra để giải quyết các vấn đề cơ bản khiến cho đời sống của con người trở nên đơn giản hơn như: mua hàng online, đặt vé, giao dịch, xem thông tin, tìm việc. Các ứng dụng hiện nay đa phần được dùng và quản lý dựa trên website và app. Với xu hướng phát triển ứng dụng hiện nay, website trở nên thân thiện với người dùng hơn nhờ vào sự tương thích với mọi thiết bị di dộng và sự quản lý một cách dễ dàng. Vì thế việc phát triễn website hiện nay vẫn còn là xu hướng và trong vài năm tới nữa.

## Lý do chọn đề tài

Công việc là lao động để tạo ra vật chất nuôi sống mỗi con người chúng ta, vì thế vấn đề tìm việc vẫn là vấn đề được quan tâm nhiều nhất từ xưa đến nay. Những năm gần đây với sự phát triễn của đất nước kèm theo những startup mọc lên như nấm và những doanh nghiệp nước ngoài đầu tư vào nước ta dẫn đến nhu cầu tìm việc tăng cao, dựa vào thống kê của Navigos Group, quý 02/2022 đã có hơn 36.000 việc làm được đăng lên trang tuyển dụng trực tiếp, tăng hơn 20% so với cùng kỳ năm 2021. Nắm bắt được nhu cầu tuyển dụng của thị trường, chính vì thế nhóm chúng em quyết định chọn đề tài “Thiết kế trang web tuyển dụng” cho môn đồ án ngành.

## Tóm tắt như cầu để thiết kế Website tuyển dụng

Rất nhiều năm về trước, mọi người mọi nhà đều đã quen với câu nói đi “xin việc”. Các cử nhân, sinh viên ở nước ngoài về đều phải chờ phân công công tác.

Người đi xin một công việc còn được phân công cho việc gì thì “tùy ý ông chủ”. Không chỉ vậy liên lạc khó khăn giữa đơn vị bổ nhiệm thêm nhân công với người đi xin việc và còn tốn cả tháng để duyệt công văn, đưa quyết định gọi đi làm đến người xin việc lại mất thêm vài tháng trời, chung ta còn quên thuộc với hình ảnh người tìm việc đọc báo mỗi buổi sáng để dò tìm những công việc đang tuyển. Đặc biệt sau đại dịch covid nhu cầu tuyển nhân viên và nhu cầu tìm việc tăng cao Vì thế thời gian sau này những website tuyển dụng rất hữu dụng cho vấn đề tìm việc cũng như tìm nhân viên phù hợp cho vị trí công ty. Đa phần các doanh nghiệp ngày xưa quản lý thông tin tuyển dụng thông qua giấy bút như thế sẽ rất tốn nhiều thời gian kèm theo đó là những rủi ro như mất hoặc trùng lắp thông tin ví dụ như khi nhiều người ứng tuyển vào cùng một vị trí mà không có sự hỗ trợ tự động giảm của hệ thống sẽ dẫn đến việc sai số liệu.

Cùng với sự đa dạng ngành nghề kèm theo sự gia tăng của nhu cầu ứng tuyển nên việc quản lý cần phải có sự hỗ trợ của công nghệ giúp cho người dùng có thể lựa chọn công việc trên website một cách dễ dàng.

Website tuyển dụng sẽ giúp các doanh nghiệp tối ưu những công việc quản lý thống kê những ứng viên đã nộp, những ứng viên đã ứng tuyển và công việc đã đăng tin ... nhờ vậy tiết kiệm được thời gian cũng như nguồn lực.

### Những khó khăn khi xây dựng hệ thống tuyển dụng

* Cần phải quản lý nhiều chức năng cũng như bộ phận: ứng viên, doanh nghiệp, quản lý CV, quản lý ứng viên đã nộp đơn và trúng tuyển, quản lý tin tuyển dụng...
* Thiết kế giao diện vừa đầy đủ thông tin các ngành nghề vừa không quá rối mắt cho người dùng.
* Tối ưu hệ thống nhằm tránh gây quá tải cho hệ thống.

### Các yêu cầu cơ bản cần có trong chương trình

* **Đối với người dùng**
* Giao diện dễ dàng sử dụng, không hiển thị quá nhiều thông tin.
* Thuận lợi trong việc xem thông tin doanh nghiệp và bài viết tuyển dụng.
* Có nhiều sự lựa chọn công việc, chuyên ngành cho người dùng để nộp đơn.
* Quản lý thông tin, những hành động một cách rõ ràng.
* Có thể lựa chọn doanh nghiệp dựa vào đánh giá của những người dùng khác.
* **Đối với hệ thống**
* Thao tác dễ sử dụng, chịu tải khi lượng người truy cập lớn.
* Cần có độ chính xác cao.
* **Đối với quản lý**
* Có thể cập nhật, thêm, xóa các thông tin như: tin tuyển dụng, thông tin người dùng, thông tin doanh nghiệp.
* Quản lý hệ thống.
* Thống kê số lượng công việc của từng ngành, số lượng ứng viên đã nộp đơn vào từng ngành.

## So với các website tuyển dụng hiện nay

* **Về sự quan sát**
* Giao diện dễ sử dụng.
* Hiển thị đầy đủ các chức năng tuyển dụng.
* **Về cải tiến**
* Nên áp dụng chức năng tư vấn.
* Nên áp dụng chức năng cho người dùng thêm CV bên ngoài và dựa vào CV để chọn lọc hiển thị những công việc phù hợp.
* **Về sáng tạo**
* Thêm thanh toán online mỗi khi đăng bài viết.
* Tạo thêm các blogs để phổ biến trang web truyên mạng xã hội nhiều hơn.

## Phương pháp cần lưu ý khi phát triển một website

* Cần phân tích rõ ràng đối tượng chính mà trang web này nhắm đến sẽ là đối tượng nào, nhu cầu họ muốn là gì.
* Phát triễn website cần tương thích với các thiết bị.
* Thông tin cần rõ ràng, chính xác.
* Dễ dàng bảo trì và nâng cấp hệ thống.
* Dữ liệu cần đảm bảo tính toàn vẹn.

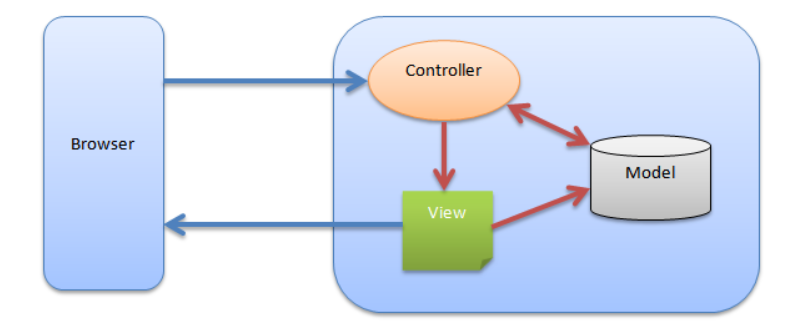
## Tầm quan trọng của việc thiết kế website tuyển dụng

Với tốc độ tăng trưởng mạnh của công nghệ , việc đơn giản hóa các công tác quản lý nhờ vào công nghệ dần là một điều thiết yếu trong cuộc sống . Nó đáp ứng được mọi nhu cầu của các doanh nghiệp và con người . Những trang web tuyển dụng đóng vai trò như một trợ thủ đắc lực cho các doanh nghiệp , và là địa điểm uy tin để người tìm việc có cơ hội tiếp cận việc làm , đồng thời nó còn giúp giảm thiểu thời gian thời gian tìm việc , giảm lượng lao động thất nhiệp đặt biệt rất giúp ích trong việc tăng ngân sách nhà nước.

# Cơ sở lý thuyết

## Giới thiệu về Spring MVC.

Spring MVC là một framework được sử dụng trong ứng dụng phát triển web. Nó dựa trên mô hình MVC (Model-View-Controller).



Hình 2.1: Mô hình Model-View-Controller.

Bao gồm 3 lớp:

* Model: bao gồm các class Pojo, Service, …
* View: là phần giao diện người dùng gồm các file html, jsp, …
* Controller: Front Controller, các Controller điều hướng các request, ...

## Giới thiệu về API.

Là viết tắt của Application Programming Interface, là chức năng trung gian giúp luân chuyển dữ liệu giữa các ứng dụng. Các dữ liệu sẽ được trả về dưới kiểu dữ liệu JSON hoặc XML bởi các giao thức HTTP và HTTPs.

## JSP – JavaServer Pages.

Là một công cụ hổ trợ phát triển trang web của JAVA. Nó là sự kết hợp giữa HTML hoặc XHTML, action và các phần tử XML, …



Hình 2.2: Vị trí của JSP trong 1 trang web.

Một số ưu điểm của các trang JSP so với CGI(Common Gateway Interface):

* Hiệu suất nhỉnh hơn vì không cần các tệp CGI riêng biệt.
* Không cần máy chủ cài đặt trình thông dịch.
* Có quyền truy cập vào JAVA APIs như là JDBC, JNDI, …

## Phương thức giao tiếp HTTP Request – HTTP Response.

### HTTP Request

HTTP Request là những thông tin yêu cầu của client gửi tới sever để yêu cầu xử lý một số công việc nào đó. Nó có thể là dữ liệu kiểu XML hoặc JSON.

Một số các phương thức của HTTP Request.

* Phương thức GET: client sẽ gửi dữ liệu lên sever trên đường dẫn URL. Phương thức này sẽ bắt đường dẫn đó, phân tích và trả lại kết quả cho client. Phương thức này được sử dụng khá phổ biến.
* Phương thức POST: client sẽ gửi dữ liệu lên sever thông qua request body. Tính bảo mật của phương thức POST cao hơn là GET.
* Phương thức PUT: tương tự như là POST nhưng nó chỉ dùng để cập nhật toàn bộ dữ liệu của một đối tượng đã tồn tại.
* Phương thức PATCH: giống như là POST và PUT nhưng thay vì cập nhật toàn bộ đối tượng thì nó chỉ cập nhật 1 phần dữ liệu.
* Phương thức DELETE: được sử dụng để yêu cầu xóa dữ liệu tới sever, và nó không có request body.
* Phương thức HEAD: tương tự GET nhưng nhanh hơn vì không có response body. Khi sử dụng GET sẽ trả về danh sách dữ liệu trong response body nhưng HEAD thì không như vậy khi cần xử lý kiểm tra API thì phương thức này sẽ tối ưu hơn.

Cấu trúc của HTTP Request bao gồm:

* Request line: bao gồm phương thức HTTP, URI (Uniform Resource Identifier) và phiên bản của HTTP.
* Request header: giúp cho client có thể gửi yêu cầu tới sever kèm theo các thông số (Header Parameters) như là User-Agent, Connection, Cache-Control, Accept-Language, …
* Request body: thường sử dụng cho các phương thức PUT, POST, PATCH.

### HTTP Response

HTTP Response là những thông tin mà sever gửi về cho phía client nó có thể là đoạn mã HTML đi kèm dữ liệu nằm trong header.

Cấu trúc của HTTP Response:

* HTTP-version: cung cấp phiên bản HTTP mà sever đang hổ trợ.
* Status-Code: mã phản hồi trả về cho client.
* Reason-Phrase: những mô tả dành cho Status-Code.

## Xác thực và phân quyền người dùng với Spring Security.

Là một công cụ hổ trợ của Spring Framework được sử dụng để xác thực và phân quyền người dùng.

Thao tác cài đặt thư viện Spring Security vào project của bạn:

* Thêm các dependency vào file cấu hình pom.xml:

<dependency>

<groupId>org.springframework.security</groupId>

<artifactId>spring-security-web</artifactId>

<version>5.2.1.RELEASE</version>

</dependency>

<dependency>

<groupId>org.springframework.security</groupId>

<artifactId>spring-security-config</artifactId>

<version>5.2.1.RELEASE</version>

</dependency>

<dependency>

<groupId>org.springframework.security</groupId>

<artifactId>spring-security-taglibs</artifactId>

<version>5.5.1</version>

</dependency>

* Cấu hình chứng thực người dùng trong file SpringSecurityConfig:

@Override

protected void configure(AuthenticationManagerBuilder authen) throws Exception{

authen.userDetailsService(userService).passwordEncoder(passwordEncoder());

}

* Cấu hình phân quyền người dùng trong file SpringSecurityConfig:

@Override

protected void configure(HttpSecurity http) throws Exception {

http.formLogin().loginPage("/login").usernameParameter("u").passwordParameter("psw").successHandler(/).failureUrl("/login?error");

http.logout().logoutSuccessUrl("/login");

http.exceptionHandling().accessDeniedPage("/?accessDenied");

http.authorizeRequests().antMatchers("/").permitAll()

.antMatchers("/admin/\*\*").access("hasRole('ROLE\_ADMIN')")

.antMatchers("/company/\*\*").access("hasRole('ROLE\_COMPANY')")

.antMatchers("/candidate/\*\*").access("hasRole('ROLE\_USER')");

http.csrf().disable();

}

Cấu trúc trên phân quyền chỉ cho người dùng thuộc ROLE\_COMPANY được phép truy cập vào các đường dẫn có chứa “/company” trong đó, và các ROLE khác cũng sẽ được phân quyền tương tự như vậy.

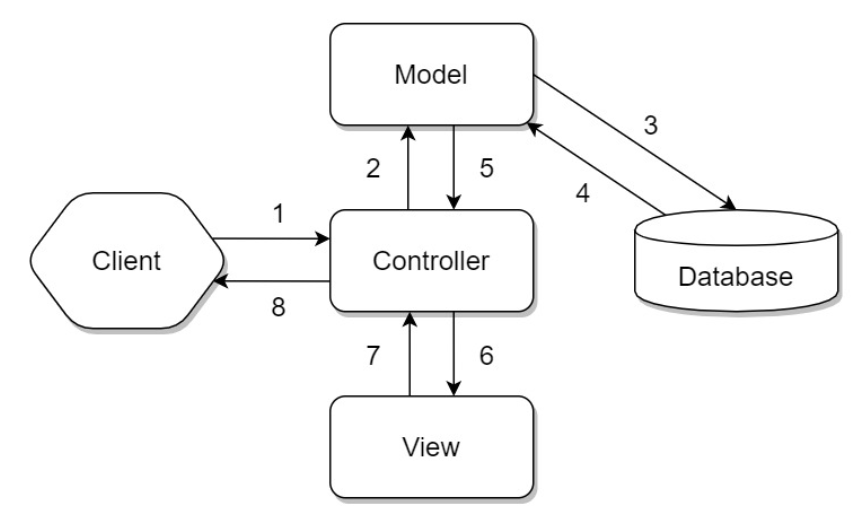
## Controller trong SpringMVC.

### Giới thiệu về Controller.

Controller trong SpringMVC là trung gian của sever và client và là nơi nhận và xử lý các request từ client. Bao gồm hai dạng:

* @Controller trả về 1 View hoặc kiểu dữ liệu JSON nằm trong response body.
* @RestController cũng giống như @Controller nhưng chỉ trả về data nằm trong response body và nó được sử dụng chính để xử lý và cung cấp API.

### Luồng hoạt động của Controller.



Hình 2.3: Các luồng hoạt động xử lý request của Controller.

Các bước nhận và xử lý request từ client của Controller:

* Đầu tiên khi client gửi một request tới sever nó sẽ đi qua một nơi gọi là DispatcherServlet hay còn gọi là Front Controller. Ở đây request của client sẽ được phân giải và kiểm tra request đó tới method nào để tìm đúng method đó. Các data trong request sẽ được xử lý, parse và mapping vào các tham số tương ứng(@RequestParam, @Header, @PathVariable, …)
* Trong trường hợp data không thể xử lý thì DispatcherServlet sẽ trả về bad request.
* Nếu request có yêu cầu dữ liệu thì Controller sẽ lấy dữ liệu trong Model và Model sẽ tương tác với cơ sở dữ liệu để nhận dữ liệu sau đó sẽ trả về cho Controller. Trong trường hợp request có yêu cầu chuyển trang thì Controller gọi đến View theo yêu cầu và sau đó sẽ trả kết quả về cho client. Còn trong trường hợp ngược lại, nếu không cần chuyển trang thì Controller sẽ trả kết quả về cho client theo dạng dữ liệu JSON thông qua API.

### Tiếp nhận request trong Controller.

Spring sử dụng các annotation để chỉ định các HTTP method tương ứng với các method của Controller. Một số annotation thường được sử dụng như: @GetMapping, @PutMapping, @PostMapping, … hoặc sẽ sử dụng @RequestMapping và chỉ định method chính cho nó:

@RequestMapping(value = "/register", method = RequestMethod.POST)

### Tiếp nhận dữ liệu từ request trong Controller.

Khi nhận các request từ client, tùy vào cách truyền dữ liệu thì Controller sẽ có các cách nhận khác nhau:

* Request param: dữ liệu sẽ đường truyền trực tiếp trên đường dẫn dưới dạng “/user?name=Minh” khi đó muốn lấy được dữ liệu name=Minh thì chúng ta sẽ cần sử dụng annotation @RequestParam(“name”) thì sẽ lấy được dữ liệu từ nó.
* Request body: chỉ nằm trong method PUT và POST, dữ liệu sẽ nằm dưới dạng JSON hoặc form-data và nhiệm vụ của Controller là sẽ parse nó về Object.
* Path variable: tương tự như Request param, dữ liệu cũng được truyền trực tiếp trên đường dẫn dưới dạng “/user/{id}” ví dụ như là “/user/1” thì 1 ở đây sẽ là Path variable và chúng ta sẽ cần sử dụng annotation @PathVariable(“id”) để lấy được dữ liệu.
* Header: dữ liệu của bạn được mã hóa và được truyền trong header. Nếu muốn lấy được dữ liệu trong header thì chúng ta sẽ cần đến annotation @Header.

## Chuyển đổi dữ liệu Serialization.

Trong lập trình JAVA chắc hẵn ai cũng đã từng nghe đến Serialization, JAVA không trao đổi dữ liệu dưới dạng Object vì vậy cần thể hiện chúng dưới dạng byte thì đây chính là một cơ chế chuyển đổi dữ liệu của Object thành một chuỗi byte để có thể đọc ghi Object ra file hoặc là để mapping với database.

Và để thực hiện công việc trên thì class của Object đó phải hiện thực một lớp interface đó là java.io.Serializable:

public class User implements Serializable{}

## Một số annotation quan trọng trong SpringMVC.

Trong Spring MVC có rất nhiều annotation(chú thích) được sử dụng cho đoạn code Java của bạn. Và đây là một số annotation phổ biến và công dụng của nó:

* @Autowired: Annotation này được sử dụng để tự động wiring đến một trong các bean trong Application Context hoặc các bean đã được Spring container tạo ra bằng phương thức @ComponentScan. Sau đó sẽ tự động cung cấp các thuộc tính và hàm cho nó.
* @Configuration: thường được khai báo cho class chứa các bean cấu hình.
* @ComponentScan: hướng dẫn cho Spring container biết cần tạo các bean từ package nào.
* @Component: dùng để khải báo 1 bean, bạn sẽ không thể nào @Autowired đến một lớp nếu lớp đó không chứa @Component.
* @Service: là annotation thuộc @Component, thường được xử dụng để xử lý các nghiệp vụ.
* @Repository: là annotation thuộc @Component, thường được dùng để trao đổi với cơ sở dữ liệu.
* @Controller: dùng để khai báo 1 lớp là Controller.

## Giới thiệu về Cloudinary.

Cloudinary – một giải pháp giúp người dùng quản lý upload, lưu trữ hình ảnh trên cloud. Giúp các lập trình viên tích hợp vào các website hoặc các ứng dụng để lưu trữ dữ liệu của người dùng.

Các bước cấu hình cloudinary:

* Thêm các dependency vào file cấu hình pom.xml.

<dependency>

<groupId>com.cloudinary</groupId>

<artifactId>cloudinary-http44</artifactId>

<version>1.29.0</version>

</dependency>

* Tạo bean trong SpringContext.

@Bean

public Cloudinary cloudinary() {

Cloudinary cloud = new Cloudinary(ObjectUtils.asMap(

"cloud\_name", "*your-cloud-name*",

"api\_key", "*your-cloud-api-key*",

"api\_secret", "*your-api-secret*",

"secure", true

));

return cloud;

}

* Upload 1 file lên cloudinary.

@Autowired

private Cloudinary cloudinary;

Map r = cloudinary.uploader().upload(*Object.getFile().*getBytes(), ObjectUtils.asMap("resource\_type", "auto"));

* Lấy secure url.

String link = (String)r.get("secure\_url");

## Đa ngôn ngữ.

SpringMVC cũng có công cụ hổ trợ lập trình viên về đa ngôn ngữ, thay vì phải hard code thì nó sẽ giúp chúng ta hiển thị các loại ngôn ngữ khác nhau. Công việc của lập trình viên sẽ là tạo ra các file .properties chứa các hiển thị ngôn ngữ, mỗi khi chọn file .properties của ngôn ngữ nào thì sẽ hiển thị ngôn ngữ đó.

Ví dụ:

user.username.notNullErr = Tên đăng nhập không được để trống

user.username.emailErr = Vui lòng điền đúng địa chỉ email

user.username.sameErr = Tên đăng nhập này đã tồn tại

user.password.notNullErr = Mật khẩu không được để trống

user.password.notMatch = Mật khẩu không khớp

Một vài class hổ trợ:

* AbstractLocaleResolver: lớp trừu tượng của LocaleResolver dùng để lấy thông tin địa phương của người dùng từ cookies để lấy được ngôn ngữ mà người dùng đã sử dụng trước đó.
* AcceptHeaderLocaleResolver: chỉ định ngôn ngữ chính là ngôn ngữ trong header của request HTTP từ client.
* CookieLocaleResolver: chỉ định ngôn ngữ chính là ngôn ngữ trong cookies mà người dùng đã sử dụng.
* FixedLocaleResolver: chỉ định ngôn ngữ chính là 1 ngôn ngữ mặc định đã chọn trước.
* LocaleChangeInterceptor: ngôn ngữ được thay đổi theo yêu cầu thông qua các tham số.
* SessionLocaleResolver: triển khai ngôn ngữ trong session của người dùng, dự phòng cho ngôn ngữ mặc định.

## Hibernate.

Là một ORM(Object Relational Mapping) với cơ chế hổ trợ lập trình viên trong việc mapping các lớp Pojo tới cơ sở dữ liệu, hổ trợ việc lập trình hướng đối tượng tới các bảng trong cơ sở dữ liệu quan hệ.

Để giao tiếp với nó, phải tạo một class Pojo đại diện cho 1 bảng trong cở sở dữ liệu. Dữ liệu của bảng trong cơ sở dữ liệu sẽ được hibernate tự động ràng buộc tới class trên.

@Entity //Cho biết đây là 1 thực thể chịu sự quản lý của hibernate

@Table(name = "job") //Cho biết thực thể này đại diện cho bảng JOB

public class Job implements Serializable{

@Id //Cho biết đây là khóa chính

@GeneratedValue(strategy = GenerationType.IDENTITY) //Tự động tăng

@Column(name = "ID")

private Integer id;

@Column(name = "Name") //Trường name đại diện cho cột name

private String name;

}

Cách cấu hình hibernate cho việc lập trình của bạn:

* Thêm các dependency vào file pom.xml.

<dependency>

<groupId>org.hibernate</groupId>

<artifactId>hibernate-core</artifactId>

<version>5.4.10.Final</version>

</dependency>

* Cấu hình file HibernateConfig.

@Configuration

public class HibernateConfig {

@Autowired

private Environment env;

@Bean

public LocalSessionFactoryBean getSessionFactory() {

LocalSessionFactoryBean sessionFactory

= new LocalSessionFactoryBean();

sessionFactory.setPackagesToScan(new String[]{

"com.hm.pojo" //Chọn package chứa các Pojo muốn scan

});

sessionFactory.setDataSource(dataSource());

sessionFactory.setHibernateProperties(hibernateProperties());

return sessionFactory;

}

@Bean

public DataSource dataSource() {

DriverManagerDataSource dataSource

= new DriverManagerDataSource();

dataSource.setDriverClassName(

env.getProperty("com.mysql.cj.jdbc.Driver "));

dataSource.setUrl(env.getProperty("*url-database-cua-ban*"));

dataSource.setUsername(

env.getProperty("*usname-mysql-cua-ban*"));

dataSource.setPassword(

env.getProperty("*passwwork-mysql-cua-ban*"));

return dataSource;

} //xác định các datasource để kết nối tới mysqlWorkbench

private Properties hibernateProperties() {

Properties props = new Properties();

props.put(DIALECT, env.getProperty("org.hibernate.dialect.MySQLDialect "));

props.put(SHOW\_SQL, env.getProperty(true));

return props;

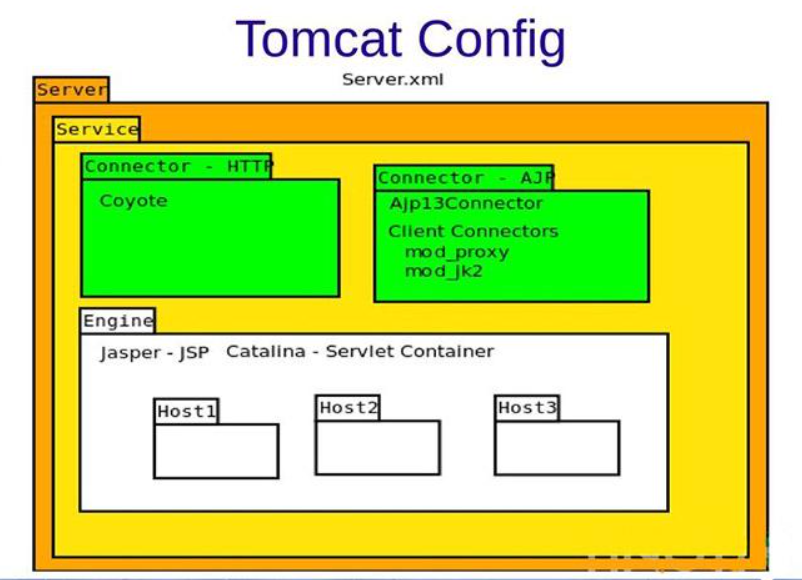
}

}

## Apache Tomcat

Apache HTTP Sever là một máy chủ cung cấp khả năng giao tiếp tới các giao thức HTTP, hổ trợ nhiều hệ điều hành như: Windows, Linux, … Nó đóng vai trò vô cùng quan trọng trong việc phát triển web.

Apache Tomcat – một web sever hổ trợ mạnh mẽ cho ngôn ngữ JAVA, nó có thể sử dụng trên nhiều bản JAVA như là: JSP, Java Servlet, … Nó hổ trợ hầu hết các tính năng của một trang web thương mại và tính ổn định tương đối cao. Nhưng bên cạnh đó, nếu sở hữu lượng truy cập lớn có thể sẽ ảnh hưởng tới hiệu năng và vẫn tồn tại một vài điểm yếu liên quan đến bảo mật.



Hình 2.4: Apache Tomcat hổ trợ được nhiều nền tảng ngôn ngữ.

# Khảo sát - Mô tả thực trạng – Thiết kế hệ thống

## Khảo sát và hoạch định hệ thống

### Đối tượng khảo sát

* Đối tượng 1: Nhân viên tuyển dụng.
* Đối tượng 2: Ứng viên.

### Các câu hỏi phỏng vấn

1. Đối tượng 1 : Người tuyển dụng

* Quy trình tuyển dụng bao gồm mấy bước
* Tuyển dụng truyền thống có gặp nhiều trở ngại không?
* Tiêu chí tuyển chọn ứng viên của Anh (chị) là như thế nào ?
* Thời gian trung bình để tuyển ứng viên cho một công việc là bao lâu?

1. Đối tượng 2 : Ứng viên

* Chất lượng phục vụ của các dịch vụ tuyển dụng online như thế nào?
* Tiêu chí lựa chọn một doanh nghiệp để nộp đơn của Anh (chị)?
* Anh(chị) có quan trọng giao diện đẹp mắt hay chỉ quan tâm chất lượng hệ thống?
* Những điều mà Anh (chị) cảm thấy khó chịu khi sử dụng một dịch vụ website?
* Những điều mà Anh (chị) thấy thích khi sử dụng một dịch vụ website?

## Khảo sát bằng phương pháp bảng câu hỏi

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ①**Rất kém** | | ② **Kém** | ③**Trung bình** | ④ **Tốt** | ⑤ **Rất tốt** |
| 1 | Quy trình tuyển dụng có được công khai minh bạch không | | | | ① ② ③ ④ ⑤ |
| 2 | Quy trình tuyển dụng có diễn ra một cách nhanh chóng không | | | | ① ② ③ ④ ⑤ |
| 3 | Dịch vụ chăm sóc khách hàng có tốt không | | | | ① ② ③ ④ ⑤ |
| 4 | Chất lượng trang web có tốt không | | | | ① ② ③ ④ ⑤ |
| 5 | Thông tin tuyển dụng có đầy đủ hay không | | | | ① ② ③ ④ ⑤ |
| 6 | Các tiêu chí chọn lọc ứng viên có hợp lý ? | | | | ① ② ③ ④ ⑤ |
| 7 | Các chức năng tìm kiếm có trong trang web có tốt không? | | | | ① ② ③ ④ ⑤ |
| 8 | Giao diện thích hợp để sử dụng ? | | | | ① ② ③ ④ ⑤ |
| 9 | Quản lý thông tin có dễ dàng ? | | | | ① ② ③ ④ ⑤ |
| 10 | Khi sử dụng hệ thống có sự ổn định ? | | | | ① ② ③ ④ ⑤ |

Bảng 3.1: Bảng khảo sát bằng câu hỏi.

## Kết quả khảo sát

### Các chức năng của hệ thống tuyển dụng cần có

* Chức năng tìm ứng viên

- Đối tượng : Nhân viên tuyển dụng.

- Chức năng :

* Đăng nhập cho doanh nghiệp.
* Đăng ký tài khoản.
* Xem thông tin doanh nghiệp.
* Xem thông tin các bài đăng tuyển.
* Xem thông tin các ứng viên đã nộp đơn.
* Tạo các tin tuyển dụng.
* Tìm kiếm các ứng viên phù hợp với vị trí doanh nghiệp đang tuyển.
* Quản lý bài viết, ứng viên đã nộp, thông tin của doanh nghiệp.
* Chức năng tìm kiếm công việc

- Đối tượng : Người tìm việc.

- Chức năng :

* Đăng ký tài khoản.
* Đăng nhập cho người dùng.
* Xem thông tin cá nhân, CV.
* Tạo CV.
* Xem thông tin các doanh nghiệp.
* Xem danh sách công việc đang tuyển.
* Xem chi tiết công việc.
* Nộp đơn cho công việc.
* Tìm kiếm doanh nghiệp và công việc theo các tiêu chí.
* Chức năng quản lý

- Đối tượng : quản trị viên (admin).

- Chức năng :

* Kiểm duyệt bài đăng.
* Kiểm duyệt doanh nghiệp.
* Thống kê số lượng công việc của từng ngành.
* Thống kê số lượng ứng viên đã nộp vào từng ngành.
* Quản lý tài khoản người dùng, tài khoản doanh nghiệp.

### Các thông tin hệ thống cần lưu trữ

* Thông tin người dùng: tên tài khoản, email, năm sinh, giới tính, trình độ, ngành nghề, mức lương mong muốn, thành phố, mô tả bản thân, ảnh cá nhân, số điện thoại.
* Thông tin doanh nghiệp: tên doanh nghiệp, địa chỉ, thành phố, logo công ty, mô tả công ty, email công ty.
* Thông tin tin tuyển dụng: tên công việc, mô tả công việc, mức lương, yêu cầu độ tuổi, yêu cầu bằng cấp, số lượng nhân viên muốn tuyển, yêu cầu giới tính, trạng thái đã kiểm tra hoặc xóa, ngày tạo tin tuyển dụng , ngày hết hạn tuyển dụng.
* Thông tin CV: thông tin cá nhân, mục tiêu nghề nghiệp, kinh nghiệm, kĩ năng, ngoại ngữ, hình ảnh cv được lưu lại.
* Thông tin ứng viên đã nộp đơn vào một công việc.
* Thông tin tin tuyển dụng đã xóa.
* Thông tin người quản trị.
* Thông tin ngành: mã ngành, tên ngành.
* Thông tin nghề : mã nghề, tên nghề, mã ngành tương ứng.

## Mô tả nghiệp vụ

### Quy trình tìm việc

Đầu tiên người dùng sẽ truy cập website thông qua các quảng cáo, hoặc biết trên mạng xã hội, trang web sẽ cho người dùng tìm kiếm thông tin các thông tin của các doanh nghiệp mà không cần đăng nhập tài khoản, nếu có nhu cầu người dùng sẽ bấm nút apply để nộp đơn ứng tuyển nhưng để thực hiện hành động thì người phải tiến hành đăng nhập, nếu chưa có tài khoản người dùng sẽ tiến hành điền thông tin để đăng ký tài khoản. Sau đó có thể thực hiện các chức năng như tìm kiếm công ty, công việc dựa theo các tiêu chí chọn lọc (ngành, thành phố, mức lương, từ khóa tìm kiếm), quản lý thông tin, xem chi tiết thông tin công việc và doanh nghiệp… , người dùng có thể liên hệ trực tiếp với doanh nghiệp qua thông tin được cung cấp trên website.

### Quy trình tìm ứng viên

Cũng giống như người dùng doanh nghiệp sẽ truy cập trang web xem thông tin, nếu có nhu cầu doanh nghiệp sẽ chọn phần đăng ký tin tuyển dụng, tại đây doanh nghiệp sẽ phải điền đầy đủ thông tin (tên công ty, logo, địa chỉ, thành phố, email và phần mô tả công ty), sau đó doanh nghiệp đc phép tìm kiếm các ứng viên dựa theo các tiêu chí (bằng cấp, ngành nghề, nơi sinh sống), đặc biệt doanh nghiệp có thể đăng tin tuyển dụng sau khi đăng ký tài khoản và phải thanh toán tiền cho mỗi bài đăng, tuy nhiên trước khi bài đăng được post lên giao diện thì phải qua bước kiểm duyệt tin của admin, sau khi đăng tin tuyển dụng doanh nghiệp có thể xem thông tin những ứng viên đã nộp hồ sơ để liên hệ.

## Xây dựng các chức năng dự kiến

* Người dùng chưa có tài khoản.
* Các chức năng dự kiến:

- Xem trang chủ.

- Xem danh sách công ty.

- Xem chi tiết thông tin của công ty.

- Tìm kiếm doanh nghiệp dựa theo tiêu chí.

- Tìm kiếm các ứng viên khác theo các tiêu chí.

- Xem danh sách công việc của mỗi ngành nghề.

- Xem chi tiết công việc và công việc gợi ý.

- Tìm kiếm công việc dựa theo các tiêu chí.

- Đăng ký thông tin người dùng.

* Người dùng đã có tài khoản.
* Các chức năng dự kiến:

- Xem trang chủ.

- Xem danh sách công ty.

- Xem chi tiết thông tin của công ty.

- Tìm kiếm doanh nghiệp dựa theo tiêu chí.

- Xem danh sách công việc của mỗi ngành nghề.

- Xem chi tiết công việc và công việc gợi ý.

- Tìm kiếm công việc dựa theo các tiêu chí.

- Tìm kiếm các ứng viên khác theo các tiêu chí.

- Đăng ký thông tin người dùng.

- Đăng nhập tài khoản.

- Có thể bình luận đánh giá các doanh nghiệp.

- Tạo CV.

- Quản lý thông tin và CV.

- Nộp đơn và lựa chọn CV tương ứng để nộp.

* Doanh nghiệp chưa có tài khoản.
* Các chức năng dự kiến:

- Xem trang chủ.

- Xem danh sách các doanh nghiệp.

- Xem và tìm kiếm các ứng viên.

- Tìm kiếm doanh nghiệp theo các tiêu chí.

- Đăng ký thông tin doanh nghiệp.

- Xem danh sách công việc của mỗi ngành nghề.

- Xem chi tiết công việc và công việc gợi ý.

- Tìm kiếm công việc dựa theo các tiêu chí.

- Xem thông tin ứng viên.

- Tìm kiếm ứng viên dựa theo các tiêu chí.

* Doanh nghiệp đã có tài khoản.
* Các chức năng dự kiến:

- Xem trang chủ.

- Xem danh sách các doanh nghiệp.

- Xem và tìm kiếm các ứng viên.

- Tìm kiếm doanh nghiệp theo các tiêu chí.

- Đăng ký thông tin doanh nghiệp.

- Xem danh sách công việc của mỗi ngành nghề.

- Xem chi tiết công việc và công việc gợi ý.

- Tìm kiếm công việc dựa theo các tiêu chí.

- Xem thông tin ứng viên.

- Tìm kiếm ứng viên dựa theo các tiêu chí.

- Đăng nhập tài khoản với vai trò là doanh nghiệp.

- Tạo tin tuyển dụng.

- Xem các ứng viên đã nộp đơn vào bản tin tuyển dụng.

- Quản lý thông tin doanh nghiệp.

- Quản lý tin tuyển dụng.

* Doanh nghiệp đã có tài khoản.
* Các chức năng dự kiến:

- Thống kê số lượng.

- Kiểm duyệt đăng ký doanh nghiệp.

- Kiểm duyệt tin tuyển dụng.

- Quản lý tài khoản người dùng và doanh nghiệp.

## Sơ đồ cây chức năng (FDH)

### Sơ đồ cây chức năng Front-End

Hệ thống website Tuyển dụng (Front-End)

Xem và tra cứu thông tin

Trang thông tin user

Tạo thông tin

Bình luận , đánh giá

Tài khoản và đăng nhập

Nộp đơn

Đăng ký tài khoản người dùng

Lựa chọn CV

Xem doanh nghiệp

Xem bình luận

Nộp đơn ứng tuyển

Tạo tin tuyển dụng

Xem thông tin cá nhân

Xem ứng viên

Đăng ký tài khoản công ty

Nhập bình luận

Tạo CV

Cập nhật thông tin

Tìm kiếm doanh nghiệp

Đăng nhập người dùng

Chọn số sao

Tìm kiếm ứng viên

Đăng nhập tài khoản doanh nghiệp

Xem ứng viên đã nộp đơn

Xem công ty mình đã nộp

### **Sơ đồ cây chức năng Back-End**

Hệ thống website Tuyển dụng (Back-End)

Quản lý doanh nghiệp

Thống kê báo cáo

Quản lý tin tuyển dụng

Quản lý admin

Quản lý người dùng

Quản lý CV

Thông kê công việc trong ngành

Sửa CV

Xóa CV

Thêm doanh nghiệp

Thêm tài khoản

Duyệt tin tuyển dụng

Thêm CV

Tạo tin tuyển dụng

Thống kê ứng viên đã nộp theo ngành

Sửa tin tuyển dụng

Sửa thông tin doanh nghiệp

Sửa thông tin người dùng

Duyệt doanh nghiệp

Xóa tin tuyển dụng

Tìm kiếm ứng viên

Tìm kiếm doanh nghiệp

Xóa doanh nghiệp

Xóa doanh nghiệp

Xóa người dùng

Xóa người dùng

## Mô tả chi tiết chức năng

* Thông tin chung
  + - * + Trang chủ: hiển thị các doanh nghiệp, điều hướng đến các trang khác.
        + Trang tin tuyển dụng: hiển thị các tin tuyển dụng theo ngành nghề, phân loại và tìm kiếm tin tuyển dụng dựa vào các tiêu chí.
        + Chi tiết tin tuyển dụng: xem chi tiết tất cả các yêu cầu của công việc đó và bao gồm thông tin công ty.
        + Chi tiết công ty: xem chi tiết tất cả thông tin về doanh nghiệp.

Tìm kiếm thông tin: tìm kiếm doanh nghiệp, tìm kiếm ứng viên.

* Người dùng và doanh nghiệp chưa có tài khoản ( người tham quan )
* Xem và dùng được các thông tin chung.
* Phải đăng ký tài khoản mới có thể đăng nhập và sử dụng các chức năng của người dùng.
* Click vào các chức năng khi chưa đăng nhập sẽ được yêu cầu đăng nhập.
* Người dùng đã có tài khoản (Khách hàng)
* Có thể xem và dùng những thông tin chung.
* Sử dụng những chức năng của vai trò người dùng: Nộp đơn, tạo cv, đánh giá doanh nghiệp, xem danh sách công việc mà bản thân đã nộp đơn.
* Khi người dùng xem xong chi tiết công việc và muốn nộp đơn, người dùng sẽ ấn vào nút nộp đơn, sau đó hệ thống sẽ hiện danh sách CV của người dùng để lựa chọn, sau khi xem xét kĩ nên dùng CV nào thì người dùng sẽ click vào CV đó và ấn nút hoàn tất.
* Xem công việc đã nộp đơn: để ghi nhớ những hành động của người dùng thực hiện trong quá khứ, hệ thống sẽ lưu lại những công việc mà người dùng đã nộp đơn tiện cho việc quản lý và chờ đợi kết quả.
* Quản lý thông tin bao gồm: sửa xóa thông tin cá nhân
* Tạo cv: người dùng muốn tạo cv sẽ bắt buộc phải điền đầy đủ thông tin, sau đó bấm nút SAVE, CV sẽ tự động lưu vào thông tin cá nhân và CV cũng sẽ tự động tải về máy 1 file CV PDF.
* Quản lý CV bao gồm: xem danh sách CV của bản thân, nếu muốn xem chi tiết rõ ràng chi tiết từng CV thì người dùng sẽ phải click vào con mắt góc phải CV, xóa CV sẽ thực hiện xóa mềm sau khi xóa CV đó vẫn còn lưu trữ ở database nhưng chỉ là không hiện lên giao diện cho người dùng, sửa CV người dùng sẽ tiến hành thay đổi thông tin có trên CV và tiến hành lưu CV.
* Đánh giá doanh nghiệp: người dùng được phép đánh giá doanh nghiệp nếu đã đăng nhập.
* Doanh nghiệp đã có tài khoản
  + Doanh nghiệp cũng có thể xem những thông tin chung.
  + Có thể sử dụng các chức năng dành riêng cho doanh nghiệp như: tạo tin tuyển dụng, xem những ứng viên đã nộp đơn cho từng tin tuyển dụng.
  + Doanh nghiệp trước khi tạo tin tuyển dụng phải cần đăng nhập tài khoản, sau đó điền đầy đủ thông tin cần có cho 1 bài tuyển dụng, sau khi tạo xong, tin tức sẽ chưa đc hiện lên website mà phải đợi qua bước kiểm duyệt của admin.
  + Doanh nghiệp có thể vào mục xem ứng viên đã nộp đơn để xem và lập ra các kế hoạch ứng tuyển phù hợp.
  + Quản lý thông tin của doanh nghiệp , quản lý các bản tin tuyển dụng.
* Người quản trị
  + - * Người quản trị chỉ có thể xem những thông tin mà không thực hiện được các chức năng.
      * Có chức năng thống kê các số lượng công việc của từng ngành, số lượng người đã nộp đơn vào từng nghề từ đó đưa ra các báo cáo, biểu đồ để nắm bắt xu hướng tuyển dụng thị trường.
      * Có quyền kiểm duyệt tin tuyển dụng tránh những nội dung không phù hợp trước khi đưa lên trang web.
      * Kiểm duyệt các doanh nghiệp đăng ký vào trừ trường hợp những công ty ma tạo tài khoản giả.
      * Quản lý tất cả tài khoản bao gồm cả doanh nghiệp lẫn người dùng.

## Mô hình thực thể ERD

### Liệt kê các thuộc tính của thực thể

* Bảng USER

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | ID | INT | Khóa chính | Khóa chính sẽ tự động tăng |
| 2 | Username | VARCHAR(45) | Not null | Tài khoản |
| 3 | Password | VARCHAR(200) | Not null | Mật khẩu |
| 4 | RegisterDate | DATE | Not null | Ngày đăng ký |
| 5 | UserRole | VARCHAR(20) | Not null | Vai trò |
| 6 | IsDeleted | INT | Not null | Đã xóa hay chưa, mặc định là 0 |

Bảng 3.2: Thông tin bảng USER

* Bảng COMPANY

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | ID | INT | Khóa chính | Khóa chính sẽ tự động tăng |
| 2 | Name | VARCHAR(100) | Not null | Tên công ty |
| 3 | Address | VARCHAR(100) | Not null | Địa chỉ |
| 4 | City | VARCHAR(45) | Not null | Thành phố |
| 5 | Avatar | VARCHAR(300) | Not null | Ảnh công ty |
| 6 | Description | VARCHAR(1000) |  | Mô tả của công ty |
| 7 | Email | VARCHAR(45) | Not null | Địa chỉ email |
| 8 | Ischecked | INT | Not null | Đã được duyệt hay chưa |
| 9 | isDeleted | INT | Not null | Đã xóa hay chưa, mặc định là 0 |
| 10 | User\_ID | INT | Not null | Khóa ngoại tới bảng USER |

Bảng 3.3: Thông tin bảng COMPANY

* Bảng CANDIDATE

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | ID | INT | Khóa chính | Khóa chính sẽ tự động tăng |
| 2 | Full\_Name | VARCHAR(100) | Not null | Tên người dùng |
| 3 | Email | VARCHAR(100) | Not null | Địa chỉ email |
| 4 | DateOfBirth | DATE | Not null | Ngày sinh |
| 5 | City | VARCHAR(45) | Not null | Thành phố |
| 6 | Avatar | VARCHAR(300) | Not null | Ảnh đại diện |
| 7 | Description | VARCHAR(500) | Not null | Mô tả bản thân |
| 8 | Phone | VARCHAR(10) | Not null | Số điện thoại |
| 9 | Degree | VARCHAR(45) | Not null | Bằng cấp |
| 10 | Major | VARCHAR(45) | Not null | Chuyên ngành |
| 11 | Sex | VARCHAR(45) | Not null | Giới tính |
| 12 | User\_ID | INT | Not null | Khóa ngoại tới bảng User |

Bảng 3.4: Thông tin bảng CANDIDATE

* Bảng ADMINSTRATOR

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | ID | INT | Khóa chính | Khóa chính sẽ tự động tăng |
| 2 | Full\_Name | VARCHAR(100) | Not null | Tên người dùng |
| 3 | Email | VARCHAR(100) | Not null | Địa chỉ email |
| 4 | User\_ID | INT | Not null | Khóa ngoại tới bảng User |

Bảng 3.5: Thông tin bảng ADMINSTRATOR

* Bảng JOB

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | ID | INT | Khóa chính | Khóa chính sẽ tự động tăng |
| 2 | Company\_ID | INT | Not null | Khóa ngoại tới bảng Company |
| 3 | Name | VARCHAR(100) | Not null | Tên của công việc |
| 4 | Description | VARCHAR(100) | Not null | Mô tả công việc |
| 5 | SalaryFrom | INT | Not null | Lương từ |
| 6 | SalaryTo | INT | Not null | Lương đến |
| 7 | AgeFrom | INT | Not null | Tuổi từ |
| 8 | AgeTo | INT | Not null | Tuổi đến |
| 9 | EndDate | DATE | Not null | Ngày kết thúc |
| 10 | CreateDate | DATE | Not null | Ngày khởi tạo |
| 11 | Quantity | INT | Not null | Số lượng tuyển |
| 12 | IsChecked | INT | Not null | Đã được duyệt hay chưa |
| 13 | DegreeRequired | VARCHAR(100) | Not null | Yêu cầu bằng cấp |
| 14 | IsDeleted | INT | Not null | Đã xóa hay chưa |
| 15 | SexRequired | VARCHAR(100) | Not null | Yêu cầu giới tính |
| 16 | ProbationaryPriod | INT | Not null | Số tháng làm việc |
| 17 | Benefit | VARCHAR(100) | Not null | Lợi ích |
| 18 | Position | VARCHAR(100) | Not null | Vị trí tuyển dụng |
| 19 | Type | VARCHAR(100) | Not null | Loại công việc |
| 20 | City | VARCHAR(100) | Not null | Thành phố |
| 21 | Address | VARCHAR(100) | Not null | Địa chỉ |
| 22 | ExperienceRequired | INT | Not null | Kinh nghiệm làm việc |
| 23 | JobRequired | VARCHAR(100) | Not null | Yêu cầu của công việc |

Bảng 3.6: Thông tin bảng JOB

* Bảng MAJOR

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | ID | INT | Khóa chính | Khóa chính sẽ tự động tăng |
| 2 | Name | VARCHAR(100) | Not null | Tên của ngành nghề |

Bảng 3.7: Thông tin bảng MAJOR

* Bảng JOB-MAJOR

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | ID | INT | Khóa chính | Khóa chính sẽ tự động tăng |
| 2 | Job\_ID | INT | Not null | Khóa ngoại tới bảng Job |
| 3 | Major\_ID | INT | Not null | Khóa ngoại tới bảng Major |
| 4 | Point | INT | Not null | Số điểm được tính |

Bảng 3.8: Thông tin bảng JOB-MAJOR

* Bảng CURRICULUM-VITAE

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | ID | INT | Khóa chính | Khóa chính sẽ tự động tăng |
| 2 | CareerGoals | VARCHAR(100) | Not null | Mục tiêu nghề nghiệp |
| 3 | DegreeDetail | VARCHAR(100) | Not null | Bằng cấp |
| 4 | ExperienceDetail | VARCHAR(100) | Not null | Kinh nghiệm làm việc |
| 5 | Skill | VARCHAR(100) | Not null | Kỹ năng |
| 6 | ForeignLanguage | VARCHAR(100) | Not null | Ngoại ngữ |
| 7 | Candidate\_ID | INT | Not null | Khóa ngoại tới bảng Candidate |
| 8 | CV\_Link | VARCHAR(100) | Not null | Đường dẫn hình ảnh CV |
| 9 | Is\_Deleted | INT | Not null | Đã xóa hay chưa |

Bảng 3.9: Thông tin bảng CURRICULUM-VITAE

* Bảng CANDIDATE-JOB

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | ID | INT | Khóa chính | Khóa chính sẽ tự động tăng |
| 2 | Candidate\_ID | INT | Not null | Khóa ngoại tới bảng Candidate |
| 3 | Job\_ID | INT | Not null | Khóa ngoại tới bảng Job |
| 4 | CV\_ID | INT | Not null | Khóa ngoại tới bảng CV |
| 5 | AppliedDate | DATE | Not null | Ngày nộp đơn |
| 6 | Isliked | INT | Not null | Đã thích Job đó hay chưa |

Bảng 3.10: Thông tin bảng CANDIDATE-JOB

* Bảng COMMENT

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Ràng buộc | Mô tả |
| 1 | ID | INT | Khóa chính | Khóa chính sẽ tự động tăng |
| 2 | Candidate\_ID | INT | Not null | Khóa ngoại tới bảng Candidate |
| 3 | Company\_ID | INT | Not null | Khóa ngoại tới bảng Company |
| 4 | Rating | INT | Từ 0 đến 5 | Đánh giá |
| 5 | Assessment | VARCHAR(200) | Not null | Bình luận |
| 6 | CreatedDate | DATE | Not null | Ngày khởi tạo |

Bảng 3.11: Thông tin bảng COMMENT

### Các trường hợp liên kết N-N

* TH1 : COMPANY - JOB

Trong bảng JOB 1 COMPANY có thể có nhiều JOB, 1 JOB chỉ thuộc 1 COMPANY, thế nên ta có mối quan hệ 1-n và lấy khóa chính của bảng COMPANY làm khóa ngoại cho bảng JOB.

* TH2 : CV - CANDIDATE

Trong bảng CV, 1 CANDIDATE có thể có nhiều CV, 1 CV chỉ thuộc 1 CANDIDATE, thế nên ta có mối quan hệ 1-n và lấy khóa chính của bảng CANDIDATE làm khóa ngoại cho bảng CV.

* TH1 : JOB – MAJOR.

Trong bảng JOB\_MAJOR, 1 JOB có thể thuộc nhiều MAJOR, 1 MAJOR có thể chứa nhiều JOB, thế nên ta có mối quan hệ n-n và lấy khóa chính của bảng JOB, MAJOR làm 2 khóa cho bảng JOB\_MAJOR.

* TH2 : CANDIDATE – JOB.

Trong bảng CANDIDATE\_JOB, 1 CANDIDATE có thể nộp đơn vào nhiều JOB, 1 JOB có thể có nhiều CANDIDATE nộp đơn ứng tuyển, thế nên ta có mối quan hệ n-n và lấy khóa chính của bảng CANDIDATE, JOB làm 2 khóa ngoại cho bảng CANDIDATE\_JOB.

* TH3 : COMPANY - MAJOR

Trong bảng COMPANY\_MAJOR, 1 COMPANY có thể có nhiều lĩnh vực MAJOR, 1 lĩnh vực MAJOR có thể có trong nhiều COMPANY, thế nên ta có mối quan hệ n-n và lấy khóa chính của bảng COMPANY, MAJOR làm 2 khóa ngoại cho bảng COMPANY\_MAJOR.

* TH4 : COMMENT

Trong bảng COMMENT có thể, 1 COMPANY có thể được bình luận bởi nhiều CANDIDATE, 1 CANDIDATE có thể bình luận vào nhiều COMPANY,

Thế nên ta có mối quan hệ n-n và lấy khóa chính của bảng COMPANY, CANDIDATE làm khóa ngoại cho bảng COMMENT.

### Các trường hợp liên kết 1-1.

* TH1: USER – ADMIN.

1 ADMIN chỉ có riêng 1 tài khoản USER và 1 tài khoản USER chỉ dành cho 1 ADMIN, thế nên ta có mối quan hệ 1-1.

* TH2: USER – COMPANY.

1 COMPANY chỉ có riêng 1 tài khoản USER và 1 tài khoản USER chỉ dành cho 1 COMPANY, thế nên ta có mối quan hệ 1-1.

* TH3 : USER – CANDIDATE.

1 CANDIDATE chỉ có riêng 1 tài khoản USER và 1 tài khoản USER chỉ dành cho 1 CANDIDATE , thế nên ta có mối quan hệ 1-1.

### Thiết kế các các thuộc tính cho từng bảng.

**USER** (ID, UserName, Password, RegisterDate, UserRole).

**CANDIDATE** (ID, Full\_Name, Email, DateOfBirth, City, Avatar, Description, Phone, Degree, Major, Sex, #User\_ID).

**ADMINISTRATOR** (ID, User\_Name, Email, #User\_ID).

**COMPANY** (ID, Name, Address, City, Avatar, Description, Email, IsChecked, #User\_ID).

**JOB** (ID, #Company\_ID, Name, Description, SalaryFrom, SalaryTo, AgeFrom, AgeTo, EndDate, CreateDate, Quantity, IsChecked, DegreeRequired, IsDeleted, SexRequired, ProbationaryPriod, Benefit, Position, Type, City, Address, ExperienceRequired, JobRequired).

**MAJOR** (ID, Name).

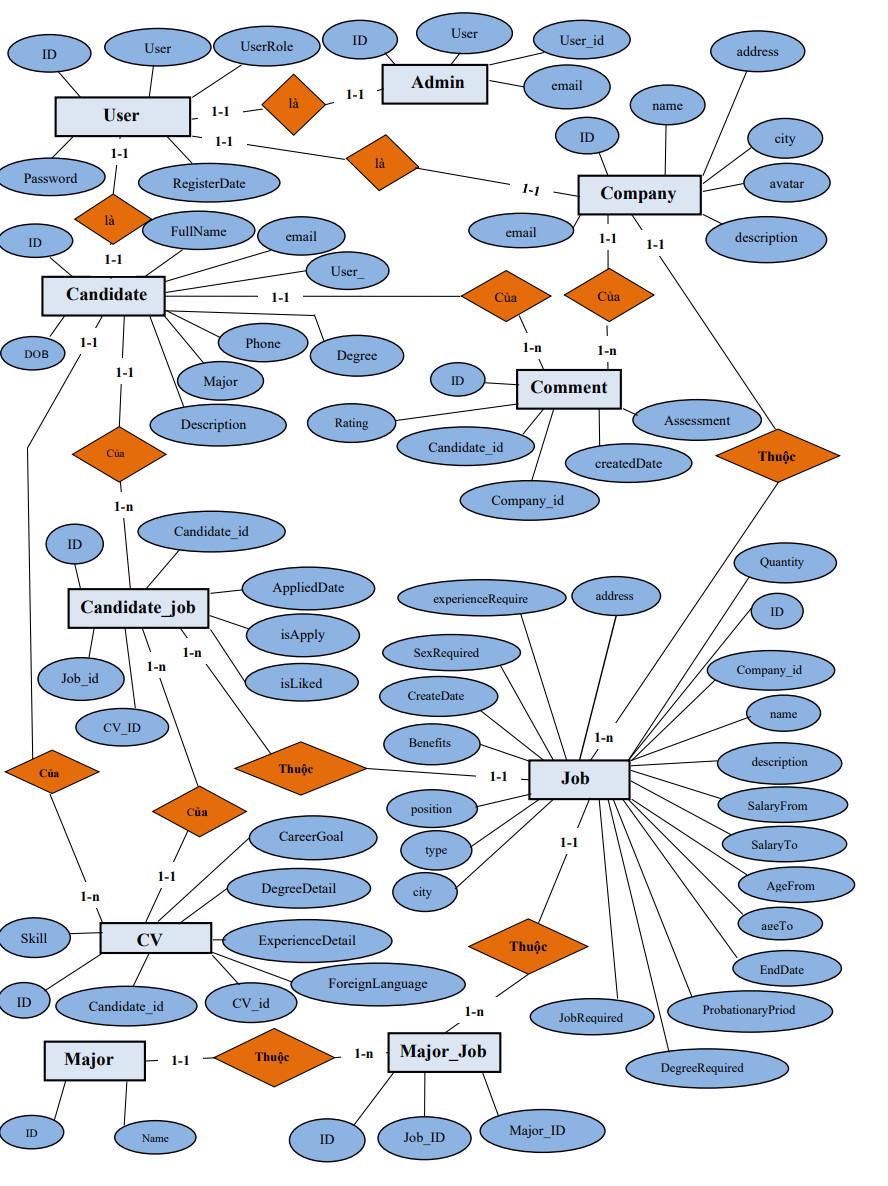
**JOB\_MAJOR** (ID, #Job\_ID, #Major\_ID, point).

**CURRICULUM\_VITAE** (ID, CareerGoals, DegreeDetail, ExperienceDetail, Skill, ForeignLanguage, #Candidate\_ID, CV\_Link, Is\_Deleted).

**CANDIDATE\_JOB** (ID, #Candidate\_ID, #Job\_ID, #CV\_ID, AppliedDate, IsLiked, IsApplied).

**COMMENT** (ID, #Candidate\_ID, #Company\_ID, Rating, Assessment, CreatedDate).

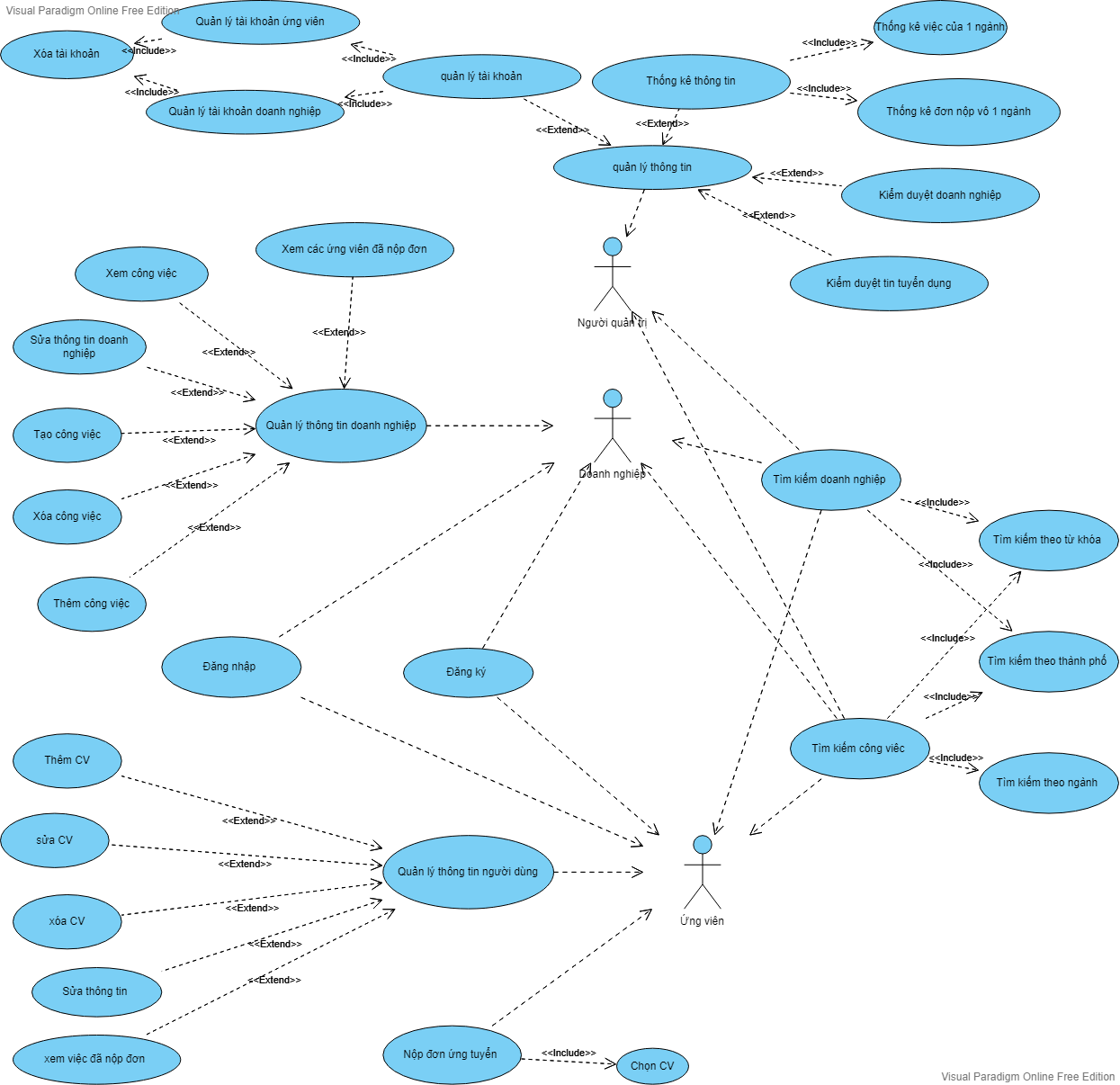
### Mô hình ERD.



Hình 3.1: Hình vẽ mối quan hệ kết hợp ERD.

## Sơ đồ Use Case

### Use Case tổng quát

****

Hình 3.2: Hình vẽ tổng thể Use Case.

### Đặc tả Use Case

Các đặc tả chức năng của hệ thống được liệt kê thông qua bảng sau:

|  |  |  |
| --- | --- | --- |
| **Mã Use case** | **Tên Use Case** | **Mô tả Use case** |
| UC1 | Đăng ký tài khoản. | Doanh nghiệp và ứng viên đăng ký tài khoản để dùng được các chức năng của hệ thống. |
| UC2 | Tạo tin tuyển dụng. | Doanh nghiệp đăng bài tuyển dụng với đầy đủ thông tin yêu cầu để tìm kiếm ứng viên nhưng chưa được đưa vào hệ thống. |
| UC3 | Nộp đơn. | Sau khi ứng viên tìm kiếm được công việc phù hợp tiến hành lựa chọn CV và bấm vào nộp đơn. |
| UC4 | Tạo CV. | Ứng viên điền đầy đủ thông tin để tạo CV sau đó CV sẽ lưu vào hệ thống và tự động tải file CV xuống máy người dùng. |
| UC5 | Quản lý thông tin ứng viên. | Khách hàng quản lý thông tin cá nhân và CV bao gồm thêm sửa xóa , và xem danh sách các tin tuyển dụng đã nộp đơn vào. |
| UC6 | Quản lý thông tin doanh nghiệp. | Doanh nghiệp quản lý thông tin và tin tuyển dụng bằng thêm sửa xóa thông tin , và doanh nghiệp có thể xem cụ thể thông tin ứng viên nộp đơn vào từng tin tuyển dụng |
| UC7 | Thống kê. | Người quản trị có nhiệm vụ thống kế những số liệu về công việc và đơn nộp vào của từng ngành, từ đó đưa ra những báo cáo phù hợp với xu hướng tuyển dụng. |
| UC8 | Kiểm duyệt thông tin. | Người quản trị sẽ được quyền duyệt doanh nghiệp đăng ký vào hệ thống và tin tuyển dụng được tạo nhằm mục đích ngăn chặn những nội dung xấu. |
| UC9 | Quản lý tài khoản. | Người quản trị có thể quản lý tất cả tài khoản có trong hệ thống. |

Bảng 3.12: Tất cả use case.

### Use Case đăng ký tài khoản

Luồng hoạt động use case đăng ký tài khoản được mô tả dưới đây:

|  |  |
| --- | --- |
| **Mã Use case** | UC1 |
| **Tên Use case** | Đăng ký tài khoản |
| **Mô tả** | Doanh nghiệp và ứng viên phải đăng ký tài khoản để sử dụng được các chức năng trong hệ thống |
| **Actor chính** | Doanh nghiệp , ứng viên |
| **Actor phụ** |  |
| **Tiền điều kiện** |  |
| **Hậu điều kiện** |  |
| **Luồng hoạt động** | - Người dùng chọn đăng ký tài khoản.  - Điền thông tin người dùng đầy đủ vào các ô input.  - Chọn nút đăng ký và hiển thị thông báo đã đăng ký thành công. |
| **Luồng thay thế** |  |
| **Luồng ngoại lệ** | Nếu người dùng chưa điền đủ thông tin hoặc điền sai thông tin, hệ thống sẽ thông báo lỗi cho người dùng. |

Bảng 3.13: Use case đăng ký tài khoản.

### Use Case tạo tin tuyển dụng

Luồng hoạt động use case tạo tin tuyển dụng được mô tả dưới đây:

|  |  |
| --- | --- |
| **Mã Use case** | UC2 |
| **Tên Use case** | Tạo tin tuyển dụng |
| **Mô tả** | Doanh nghiệp có nhu cầu tạo một bài đăng tuyển dụng nhân viên. |
| **Actor chính** | Doanh nghiệp |
| **Actor phụ** |  |
| **Tiền điều kiện** | - Phải đăng nhập với vai trò *Doanh Nghiệp* |
| **Hậu điều kiện** |  |
| **Luồng hoạt động** | - Người dùng chọn Tạo tin tuyển dụng.  - Điền thông tin yêu cầu của 1 tin tuyển dụng vào các ô input.  - Chọn nút tạo tin nhưng lúc này tin tuyển dụng chưa được đưa lên hệ thống vì phải chờ kiểm duyệt. |
| **Luồng thay thế** |  |
| **Luồng ngoại lệ** | Nếu doanh nghiệp chưa điền đủ thông tin hoặc điền sai thông tin, hệ thống sẽ thông báo lỗi cho doanh nghiệp. |

Bảng 3.14: Use case tạo tin tuyển dụng.

### Use Case nộp đơn

Luồng hoạt động use case nộp đơn được mô tả dưới đây:

|  |  |
| --- | --- |
| **Mã Use case** | UC3 |
| **Tên Use case** | Nộp đơn |
| **Mô tả** | Ứng viên chọn được một công việc phù hợp và muốn nộp đơn ứng tuyển |
| **Actor chính** | Ứng viên |
| **Actor phụ** |  |
| **Tiền điều kiện** | - Phải đăng nhập với vai trò Ứng viên |
| **Hậu điều kiện** |  |
| **Luồng hoạt động** | - Người dùng chọn nút nộp đơn.  - Chọn CV phù hợp mà ứng viên muốn nộp.  - Bấm nút con mắt để xem rõ từng CV để lựa chọn.  - Chọn vào nút xác nhận để hoàn tất nộp đơn, hiển thị thông báo nộp đơn thành công. |
| **Luồng thay thế** |  |
| **Luồng ngoại lệ** |  |

Bảng 3.15: Use case nộp hồ sơ.

### Use Case tạo CV

Luồng hoạt động use case tạo CV được mô tả dưới đây:

|  |  |
| --- | --- |
| **Mã Use case** | UC4 |
| **Tên Use case** | Tạo CV |
| **Mô tả** | Ứng viên có nhu cầu tạo CV cho bản thân. |
| **Actor chính** | Ứng viên |
| **Actor phụ** |  |
| **Tiền điều kiện** | - Phải đăng nhập với vai trò Ứng viên |
| **Hậu điều kiện** |  |
| **Luồng hoạt động** | - Người dùng chọn Tạo CV.  - Điền đầy đủ thông tin cho các trường input trong bảng CV.  - Bấm nút lưu CV. |
| **Luồng thay thế** | - Những thông tin cá nhân sẽ được tự động hiển thị lên CV.  - CV sẽ được lưu vào hệ thống và tự động tải file PDF về máy người dùng. |
| **Luồng ngoại lệ** |  |

Bảng 3.16: Use case tạo CV.

### Use Case quản lý thông tin ứng viên

Luồng hoạt động use case quản lý thông tin ứng viên được mô tả dưới đây:

|  |  |
| --- | --- |
| **Mã Use case** | UC5 |
| **Tên Use case** | Quản lý thông tin ứng viên |
| **Mô tả** | Quản lý thông tin cá nhân của ứng viên, xem các công việc đã nộp đơn và quản lý CV. |
| **Actor chính** | Ứng viên |
| **Actor phụ** |  |
| **Tiền điều kiện** | - Phải đăng nhập với vai trò Ứng viên |
| **Hậu điều kiện** |  |
| **Luồng hoạt động** | - Chọn vào mục quản lý thông tin cá nhân.  - Thêm, sửa thông tin cá nhân, xem danh sách công việc đã nộp.  - Xem, sửa, xóa , thêm CV.  - Bấm vào con mắt trên mỗi CV để xem chi tiết CV.  - Bấm vào hình thùng rác để tiến hành xóa.  - Sau khi thêm, sửa, xóa sẽ thông báo thao tác thành công. |
| **Luồng thay thế** |  |
| **Luồng ngoại lệ** |  |

Bảng 3.17: Use case quản lý thông tin ứng viên.

### Use Case quản lý thông tin doanh nghiệp

Luồng hoạt động use case quản lý thông tin doanh nghiệp được mô tả dưới đây:

|  |  |
| --- | --- |
| **Mã Use case** | UC6 |
| **Tên Use case** | Quản lý thông tin doanh nghiệp |
| **Mô tả** | Quản lý thông tin doanh nghiệp, xem các ứng viên đã nộp đơn vào từng tin tuyển dụng và quản lý tin tuyển dụng. |
| **Actor chính** | Doanh nghiệp |
| **Actor phụ** |  |
| **Tiền điều kiện** | - Phải đăng nhập với vai trò Doanh nghiệp |
| **Hậu điều kiện** |  |
| **Luồng hoạt động** | - Chọn vào mục quản lý thông tin doanh nghiệp.  - Thêm, sửa thông tin doanh nghiệp, xem danh các ứng viên đã nộp cho từng tin tuyển dụng.  - Xem, sửa, xóa, thêm tin tuyển dụng.  - Bấm vào chữ *i* để xem chi tiết công việc.  - Sau khi thêm, sửa, xóa sẽ thông báo thao tác thành công. |
| **Luồng thay thế** |  |
| **Luồng ngoại lệ** |  |

Bảng 3.18: Use case quản lý thông tin doanh nghiệp.

### Use Case thống kê

Luồng hoạt động use case thống kê được mô tả dưới đây:

|  |  |
| --- | --- |
| **Mã Use case** | UC7 |
| **Tên Use case** | Thống kê |
| **Mô tả** | Thống kế số liệu liên quan đến số lượng công việc , số đơn đã nộp vào từng ngành |
| **Actor chính** | Người quản trị |
| **Actor phụ** |  |
| **Tiền điều kiện** | - Phải đăng nhập với vai trò Người quản trị |
| **Hậu điều kiện** |  |
| **Luồng hoạt động** | - Chọn vào mục quản lý thông tin.  - Xem biểu đồ chart của số lượng công việc và số lượng đơn nộp vào trong từng ngành. |
| **Luồng thay thế** |  |
| **Luồng ngoại lệ** |  |

Bảng 3.19: Use case thống kê.

### Use Case kiểm duyệt thông tin

Luồng hoạt động use case kiểm duyệt thông tin được mô tả dưới đây:

|  |  |
| --- | --- |
| **Mã Use case** | UC8 |
| **Tên Use case** | Kiểm duyệt thông tin |
| **Mô tả** | Kiểm duyệt doanh nghiệp đăng ký vào hệ thống , và tin tuyển dụng mới được tạo |
| **Actor chính** | Người quản trị |
| **Actor phụ** |  |
| **Tiền điều kiện** | - Phải đăng nhập với vai trò Người quản trị |
| **Hậu điều kiện** |  |
| **Luồng hoạt động** | - Chọn vào mục quản lý thông tin.  - Bấm vào duyệt công việc hoặc duyệt doanh nghiệp nếu kiểm tra thấy đủ tiêu chuẩn. |
| **Luồng thay thế** |  |
| **Luồng ngoại lệ** |  |

Bảng 3.20: Use case kiểm duyệt thông tin.

### Use Case quản lý tài khoản

Luồng hoạt động use case quản lý tài khoản được mô tả dưới đây:

|  |  |
| --- | --- |
| **Mã Use case** | UC9 |
| **Tên Use case** | Quản lý tài khoản |
| **Mô tả** | Người quản trị có quyền kiểm soát tất cả các tài khoản trong hệ thống |
| **Actor chính** | Người quản trị |
| **Actor phụ** |  |
| **Tiền điều kiện** | - Phải đăng nhập với vai trò Người quản trị |
| **Hậu điều kiện** |  |
| **Luồng hoạt động** | - Chọn vào mục quản lý thông tin.  - Xem chi tiết tất cả tài khoản có trong hệ thống , và bấm xóa sẽ tiến hành xóa tài khoản đó và thống báo xóa thành công. |
| **Luồng thay thế** |  |
| **Luồng ngoại lệ** |  |

Bảng 3.21: Use case quản lý tài khoản.

# Nghiệp vụ hệ thống.

## Trang chủ.

Trang chủ của trang web tìm kiếm việc làm H&M sẽ có một số chức năng như là:

* Tìm kiếm việc làm theo thành phố.
* Thanh tìm kiếm việc làm theo từ khóa người dùng nhập và địa chỉ nơi muốn tìm kiếm.
* Hiển thị các công ty đang đứng đầu

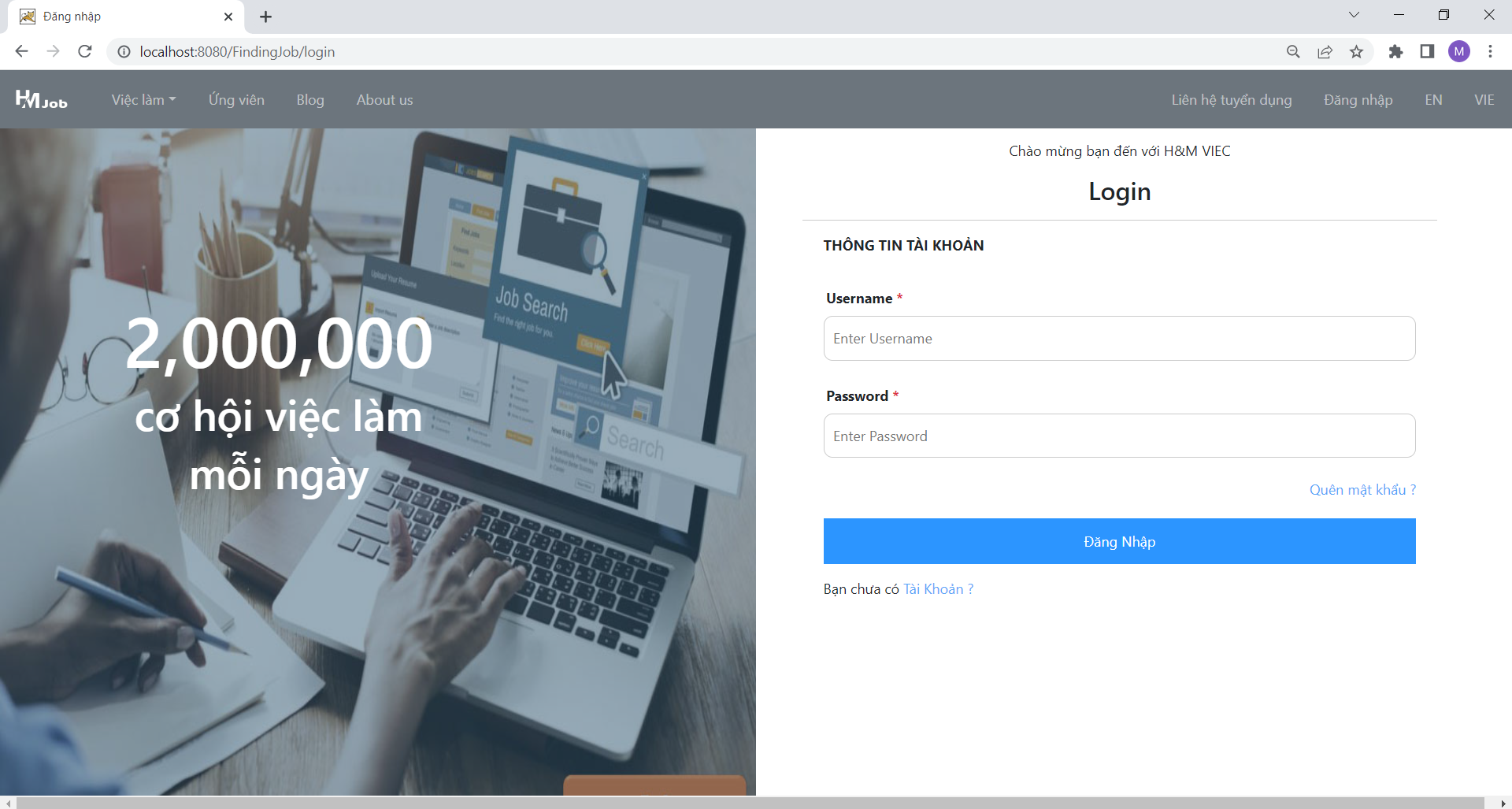


Hình 4.1: Giao diện trang chủ.

## Đăng nhập.

Khi nhấn vào “đăng nhập” trên thanh điều hướng sẽ dẫn bạn tới trang đăng nhập của trang web. Ở đây khi bạn nhập thông tin tên đăng nhập và mật khẩu của bạn và nhấn đăng nhập, nhóm đã sử dụng Spring Security để hổ trợ việc xác thực tài khoản cho người dùng nếu như thông tin đăng nhập của người dùng là đúng thì sẽ trả về trang chủ ngược lại sẽ hiển thị lỗi.

Riêng nhà tuyển dụng khi đăng nhập sẽ kiểm tra thuộc tính isChecked có bằng 1 hay không, việc này kiểm tra tài khoản của nhà tuyển dụng đó đã được ADMIN duyệt qua hay chưa.



Hình 4.2: Giao diện đăng nhập.

## Đăng ký.

### Đăng ký ứng viên

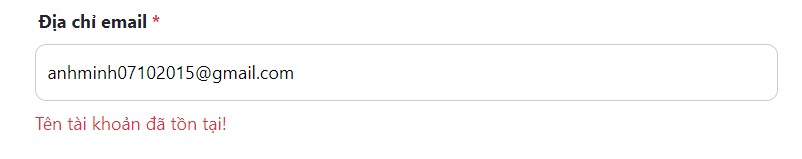
Nếu như người dùng muốn sử dụng các dịch vụ như là nộp hồ sơ, tạo đơn xin việc cho bản thân thì người dùng phải có cho mình một tài khoản ở trang web. Để tạo tài khoản người dùng chỉ cần nhấn vào chữ “*Tài khoản*” trong giao diện đăng nhập thì sẽ được chuyển hướng đến trang đăng ký tài khoản.



Hình 4.3: Giao diện đăng ký.

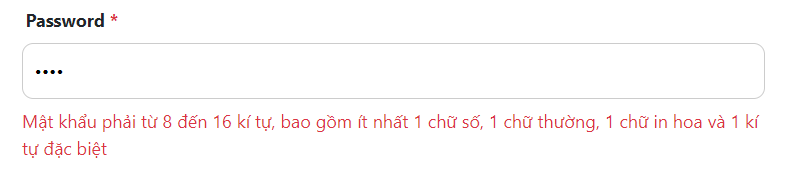
Ở đây người dùng sẽ được yêu cầu điền địa chỉ email và mật khẩu.Các thông tin đó của người dùng sẽ được gắn vào model attribute USER sau đó sử dụng phương thức POST để gửi thông tin về sever để sever tương tác với cơ sở dữ liệu để lưu các thông tin đó xuống cơ sở dữ liệu. Trang web sẽ có một số ràng buộc như sau:

* Tên đăng nhập phải định dạng địa chỉ email và không được trùng lắp với các tài khoản đã có.



Hình 4.4: Báo lỗi tên đăng nhập đã tồn tại.

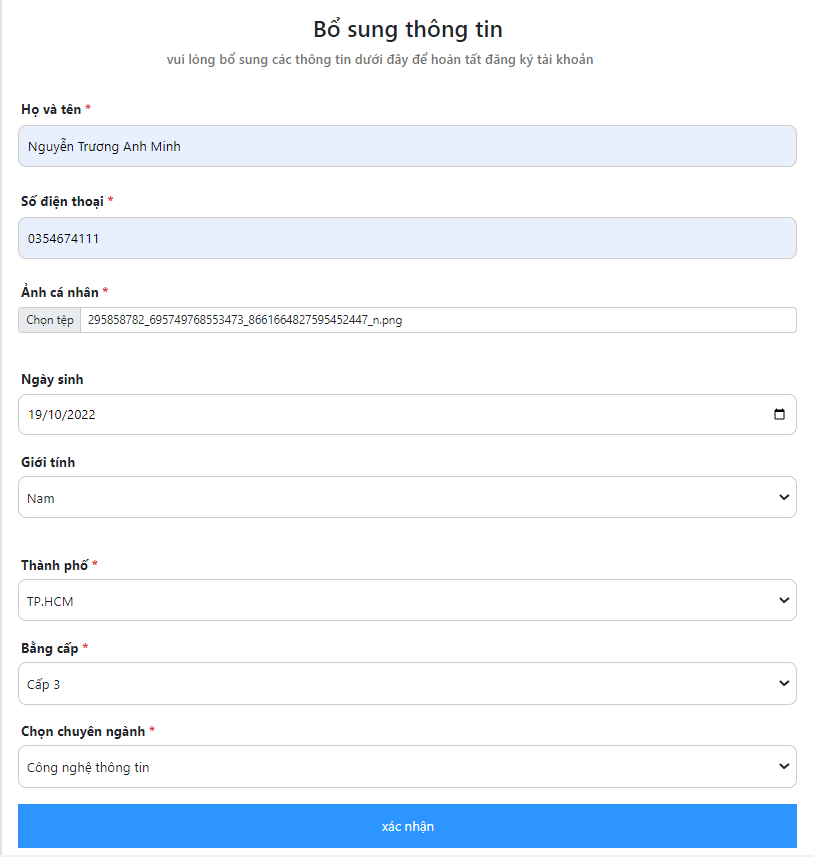
* Mật khẩu phải có từ 8 đến 16 ký tự và phải có ít nhất 1 chữ cái in hoa, 1 chữ in thường, 1 chữ số và 1 ký tự đặc biệt.



Hình 4.5: Báo lỗi mật khẩu không đúng định dạng yêu cầu.

* Mật khẩu và nhập lại mật khẩu phải trùng lắp.

Sau khi đã đăng ký thành công thì sẽ qua tiếp với trang bổ sung thông tin cho tài khoản. Ở đây người dùng bắt buộc phải điền các thông tin như là: họ và tên, số điện thoại, ảnh cá nhân, ngày sinh, giới tính, thành phố, bằng cấp, chuyên ngành của bạn. Các thông tin trên sẽ được đưa vào model attribute CANDIDATE thông qua phương thức POST để lưu các thông tin người dùng.

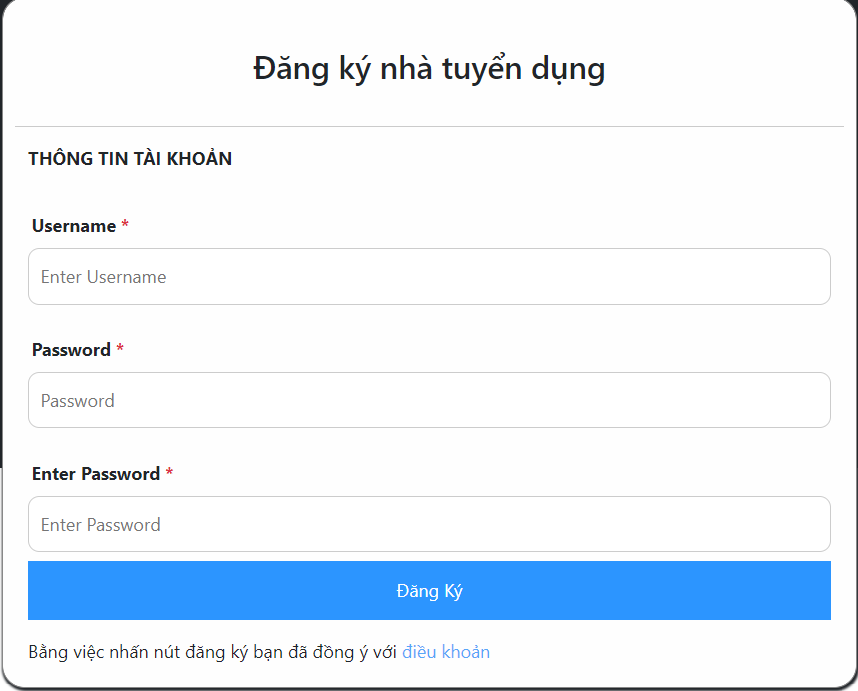


Hình 4.6: Bổ sung thông tin tài khoản ứng viên.

Sau khi đã điền đầy đủ thông tin, khi nhấn xác nhận, ảnh đại diện của bạn sẽ được mã hóa bằng encytype=“multipart/form-data”. Sau đó, hình ảnh của bạn sẽ được upload lên cloudinary và sẽ lưu link của hình ảnh xuống dưới cơ sở dữ liệu. Tài khoản của bạn đã được đăng ký thành công và sẽ có thể sử dụng để đăng nhập vào trang web.

### Đăng ký nhà tuyển dụng.

Nếu muốn đăng các tin tuyển dụng lên trên trang web thì việc đầu tiên người dùng phải làm là đăng ký nhà tuyển dụng. Việc đăng ký nhà tuyển dụng cũng tương tự như là đăng ký ứng viên, người dùng sẽ nhấn vào “*Liên hệ tuyển dụng*” trên thanh điều hướng để được chuyển tới trang đăng ký nhà tuyển dụng.

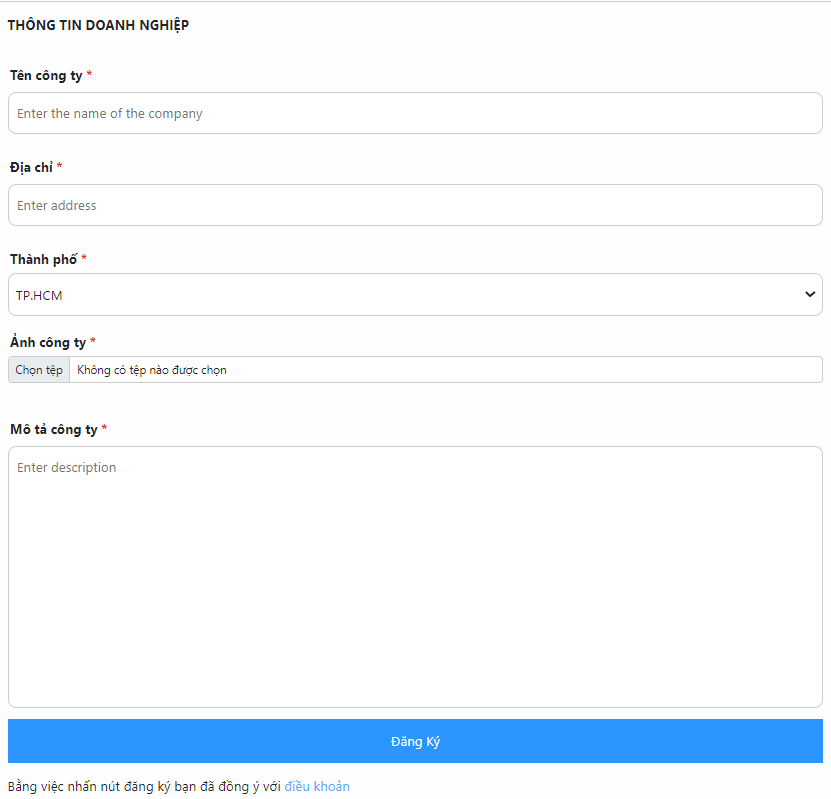


Hình 4.7: Giao diện đăng ký nhà tuyển dụng.

Tương tự như là đăng ký ứng viên thì các thông tin USERNAME hay PASSWORD đều được đưa xuống qua phương thức POST để lưu và cũng sẽ có các ràng buộc như là:

* Tên đăng nhập phải định dạng địa chỉ email và không được trùng lắp với các tài khoản đã có.
* Mật khẩu phải có từ 8 đến 16 ký tự và phải có ít nhất 1 chữ cái in hoa, 1 chữ in thường, 1 chữ số và 1 ký tự đặc biệt.
* Mật khẩu và nhập lại mật khẩu phải trùng lắp.

Sau khi đăng ký tài khoản thành công, nhà tuyển dụng sẽ được chuyển đến trang bổ sung thông tin cho nhà tuyển dụng.



Hình 4.8: Bổ sung thông tin cho nhà tuyển dụng.

Nhà tuyển dụng bắt buộc phải điền đầy đủ các thông tin trên. Sau đó các thông tin đó sẽ được lưu vào model COMPANY và sẽ được gửi xuống sever để lưu vào cơ sở dữ liệu.

Khi đăng ký thành công tài khoản của nhà tuyển dụng vẫn cần được ADMIN duyệt qua khi đó thuộc tính isChecked sẽ bằng 0 và sau khi được duyệt sẽ thiết lập thành 1 để nhà tuyển dụng đăng nhập vào và đăng các tin tuyển dụng.

## Cập nhật thông tin người dùng.

Chức năng này hổ trợ người dùng trong việc điều chỉnh tất thông tin cá nhân của mình bao gồm cả ảnh đại diện. Tất cả các thông tin của người dùng sẽ được truyền qua thương thức POST và được ép về kiểu đối tượng CANDIDATE khi đó sẽ tìm đối tượng theo ID của đối tượng đó để tiến hành chỉnh sửa đúng đối tượng cần chỉnh sửa. Sau đó tất cả dữ liệu mới sẽ được lưu xuống cơ sở dữ liệu.

@Override

public boolean updateCandidate(Candidate c) {

Session session = this.sessionFactory.getObject().getCurrentSession(); //lấy session

try {

Candidate c1 = this.getCandidateByID(c.getId()); //tìm đối tượng CANDIDATE theo ID

c1.setCity(c.getCity()); //tiến hành cập nhật thông tin

c1.setDateOfBirth(c.getDateOfBirth());

c1.setPhone(c.getPhone());

c1.setFullname(c.getFullname());

c1.setSex(c.getSex());

c1.setDegree(c.getDegree());

c1.setMajor(c.getMajor());

session.update(c1); //cập nhật thông tin xuống database

return true;

} catch (Exception ex) {

ex.printStackTrace();

return false;

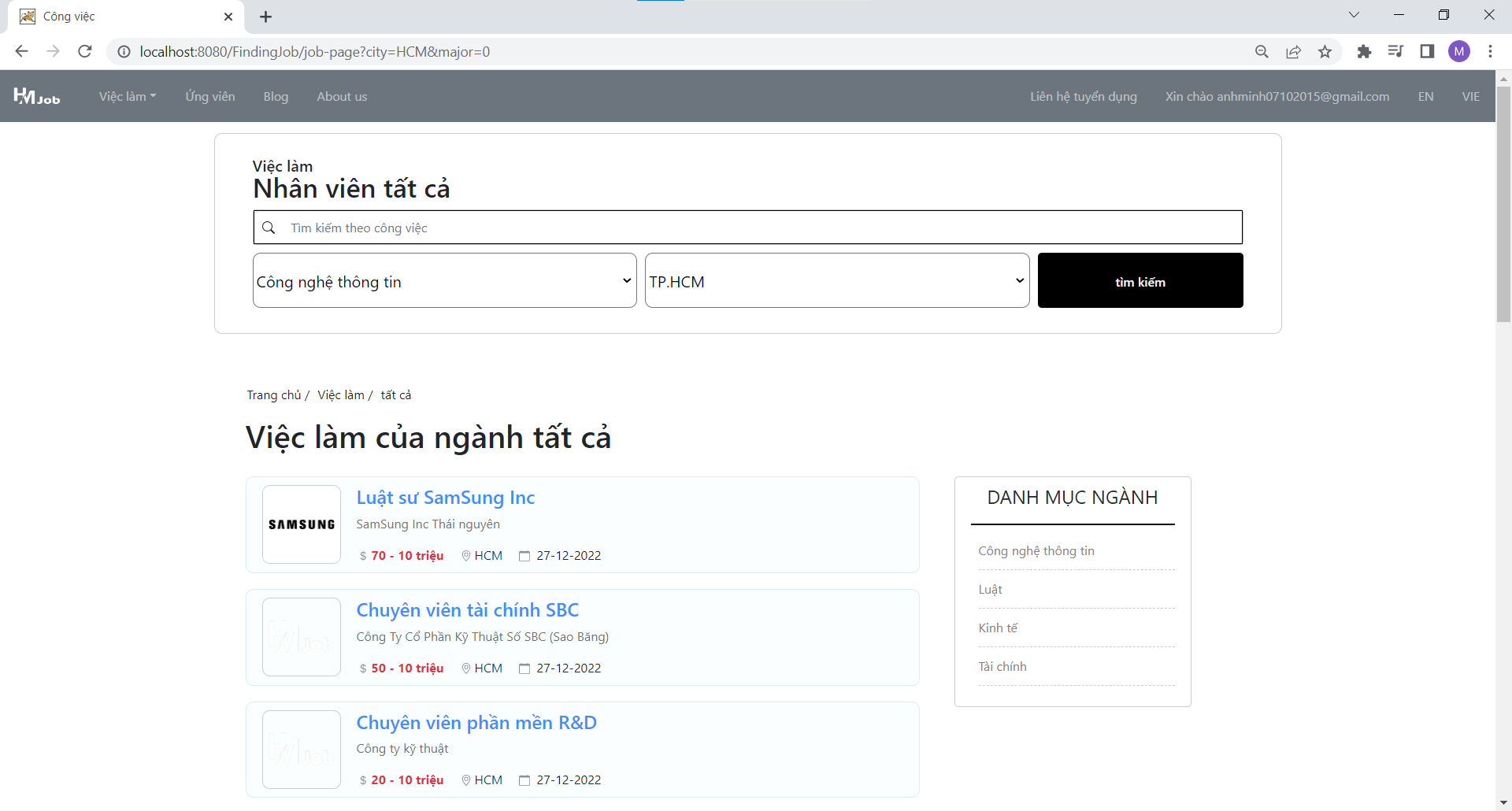
}

}

## Danh sách việc làm.

Đối với người dùng muốn tìm kiếm việc làm thì đây là chức năng dành cho họ, người dùng sẽ nhấn vào “*Việc làm”* trên thanh điều hướng để vào trang tìm kiếm việc làm.

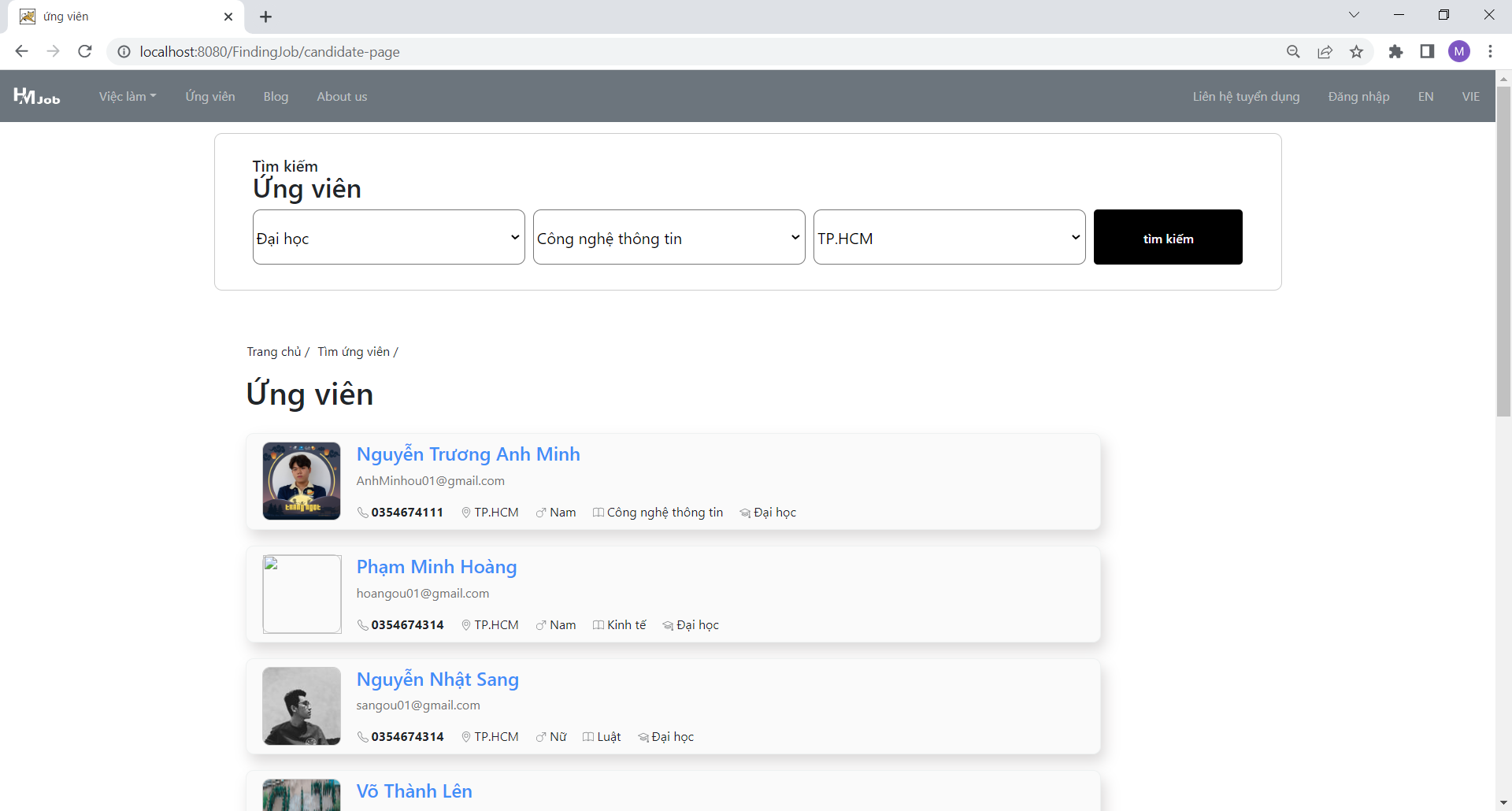
Ở đây sẽ bao gồm 1 thanh tìm kiếm, 2 ô để chọn ngành nghề và thành phố bạn muốn tìm. Sau đó khi nhấn tìm kiếm tất cả các thông tin ở trên sẽ được truyền lên url bằng request param và được bắt bằng phương thức GET để controller xử lý và lấy các dữ liệu công việc theo yêu cầu của bạn.



Hình 4.9: Giao diện trang tìm kiếm việc làm.

## Danh sách ứng viên.

Tương tự như danh sách việc làm, để đi đến trang tìm kiếm ứng viên bạn cũng sẽ nhấn vào “*Ứng viên*” trên thanh điều hướng.

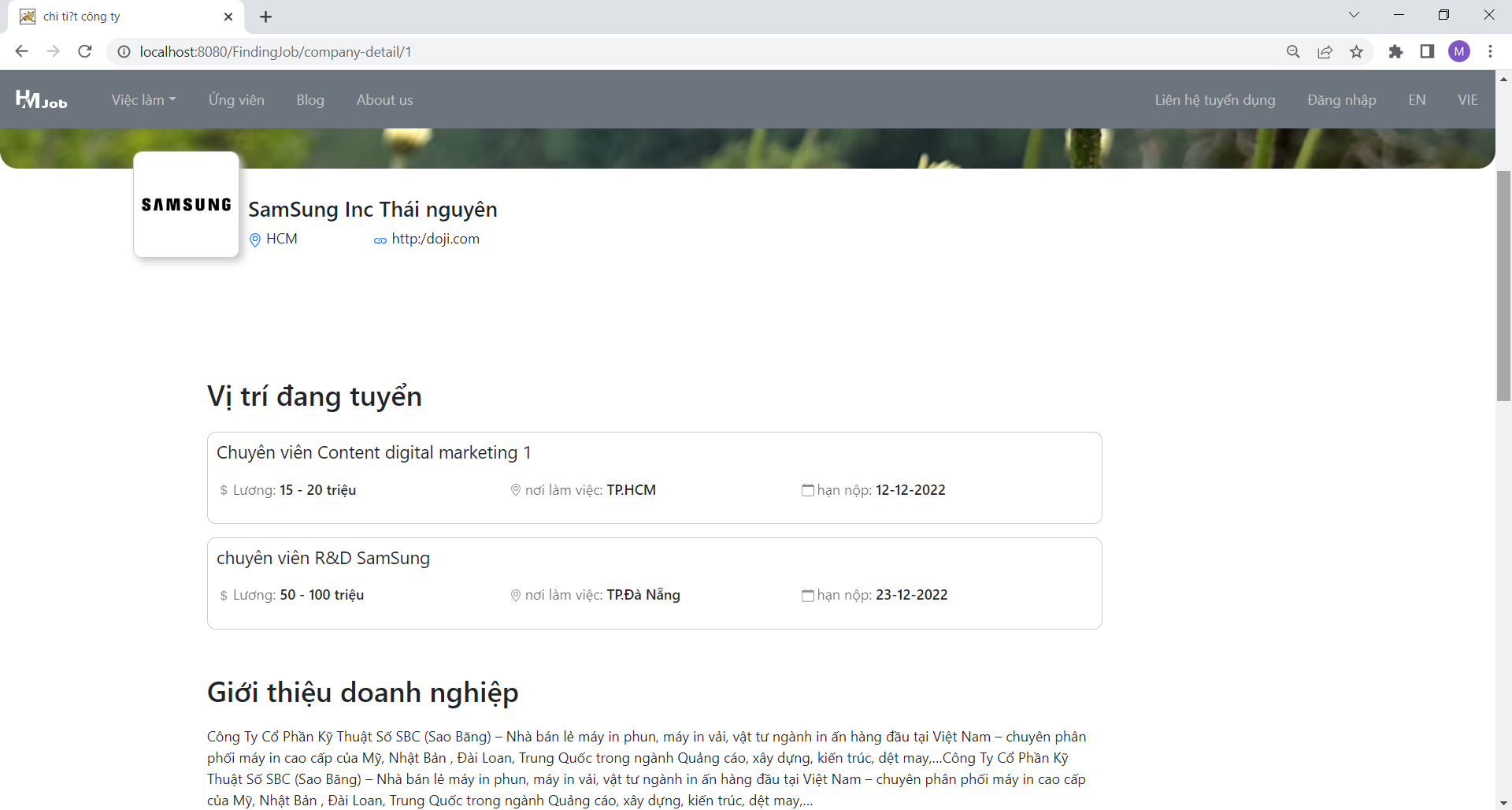


Hình 4.10: Giao diện trang tìm kiếm ứng viên.

Ở đây sẽ cung cấp cho bạn 3 bộ lọc là: bằng cấp, ngành nghề và thành phố để các nhà tuyển dụng có thể tìm kiếm ứng viên theo các tiêu chí trên.

## Thông tin công ty.

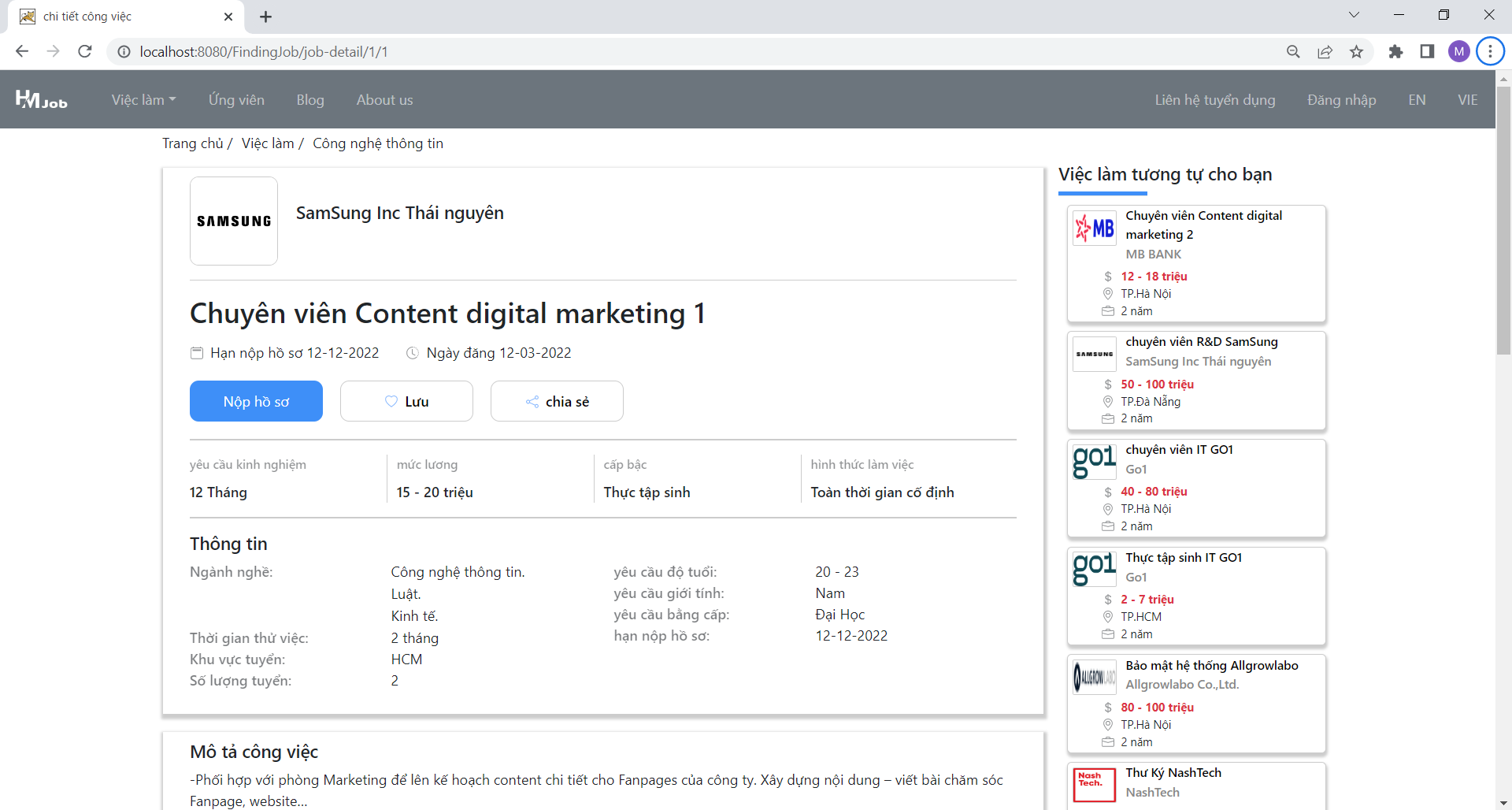
Mỗi công ty sẽ có một trang thông tin riêng cho mình, bao gồm các thông tin giới thiệu về công ty, các công việc mà công ty đang tuyển dụng và các đánh giá bình luận về công ty đó.



Hình 4.11: Giao diện trang thông tin của công ty.

## Thông tin tuyển dụng.

Khi nhấn vào một tin tuyển dụng bất kì bạn sẽ được đưa đến trang thông tin tuyển dụng của tin đó. Ở đây sẽ hiển thị cho bạn toàn bộ thông tin về tin tuyển dụng đó như là: tên công việc, vị trí, địa điểm, tiền lương, yêu cầu, mô tả, ngày kết thúc, … và sẽ hiển thị một vài thông tin cơ bản của công ty đăng tin tuyển dụng đó.



Hình 4.12: Giao diện trang thông tin tuyển dụng.

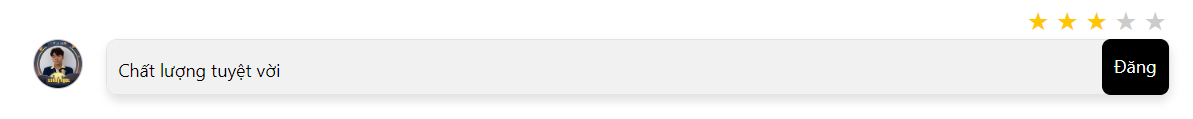
## Bình luận.

Mỗi công ty đều có một chỗ để các ứng viên có thể bình luận về công ty đó, ở đây sẽ bao gồm nội dung bình luận, số sao đánh giá, ngày giờ đã bình luận, và sẽ hiển thị toàn bộ các đánh giá của những người khác về công ty. Tất cả bình luận sẽ đều được sắp xếp theo ngày đăng tải bình luận đó.



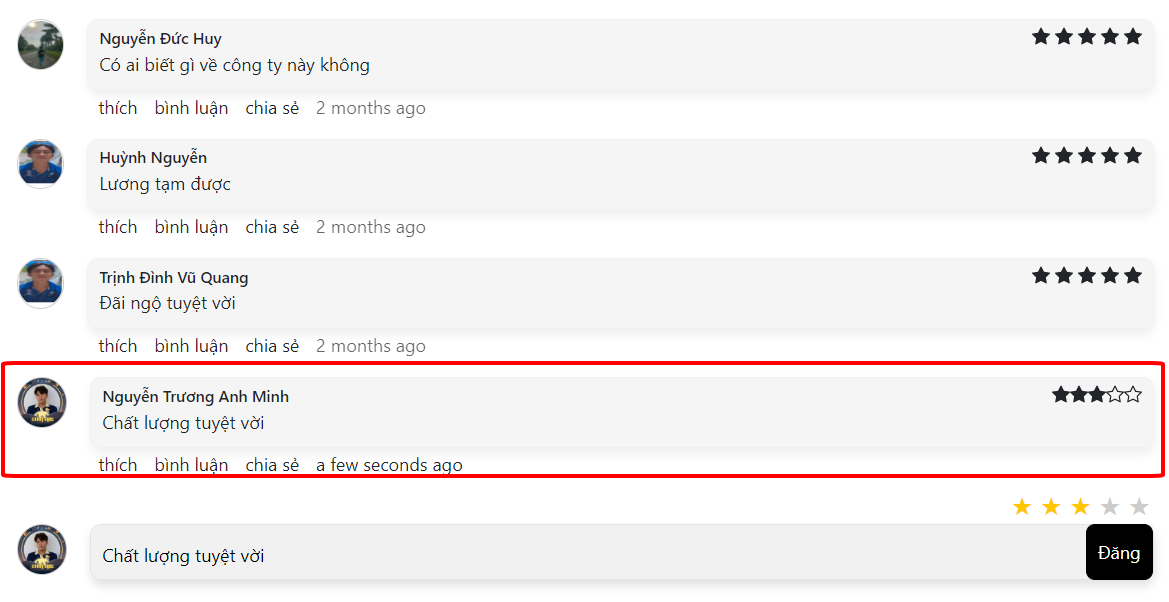
Hình 4.13: Giao diện bình luận.

Để có thể bình luận thì việc đầu tiên bạn sẽ phải đăng nhập. Nếu bạn chưa đăng nhập và ấn vào ô bình luận thì trang web sẽ điều hướng bạn đến trang đăng nhập.



Hình 4.14: Giao diện ô nhập bình luận.

Sau khi nhấn đăng thì toàn bộ thông tin bình luận của bạn sẽ được chuyển về kiểu dữ liệu JSON để lưu xuống database và giao diện sẽ hiển thị bình luận của bạn ngay sau đó.



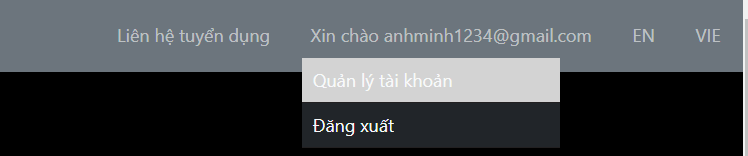
Hình 4.15: Bình luận của bạn vừa được đăng.

## Đăng tin tuyển dụng.

Một chức năng vô cùng quan trọng để các nhà tuyển dụng có thể tìm kiếm ứng viên dựa theo các tiêu chí của họ thì ở đây website của chúng tôi sẽ cung cấp cho họ chức năng để đăng các tin tuyển dụng cho mình.

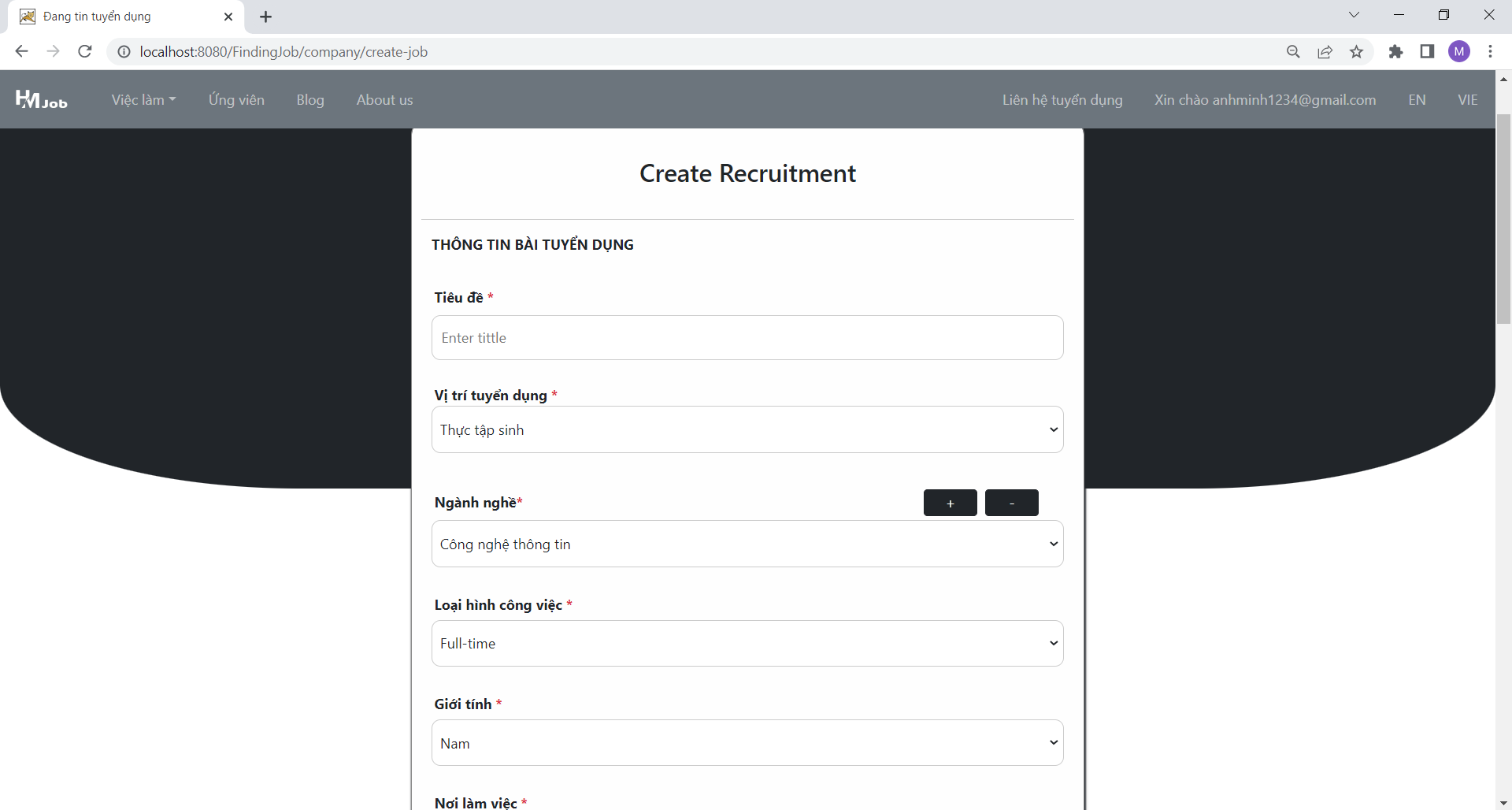
Trước hết thì nhà tuyển dụng phải tạo cho mình một tài khoản công ty và phải được các admin duyệt cho phép sử dụng.

Sau đó vô phần quản lý bằng cách rê chuột lên địa chỉ email của mình trên thanh điều hướng và chọn “*Quản lý tài khoản*”.



Hình 4.16: Quản lý tài khoản trên thanh điều hướng

Sau đó nhấn vào nút “Thêm công việc” trong trang quản lý để được điều hướng đến trang tạo tin tuyển dụng. Hoặc là truy cập theo đường dẫn <http://localhost:8080/FindingJob/company/create-job>.



Hình 4.17: Giao diện đăng tin tuyển dụng

Ở đây, bạn phải nhập toàn bộ các thông tin yêu cầu như là: tiêu đề, vị trí tuyển dụng, ngành nghề, loại hình công việc, giới tính, các yêu cầu về công việc, … Sau đó nhấn nút đăng tin thì toàn bộ thông tin sẽ được lưu vào model attribute JOB và sẽ được gửi xuống sever qua phương thức POST và sẽ được xử lý lưu vào cơ sở dữ liệu.

Tuy nhiên, tin tuyển dụng của bạn sẽ không được đăng ngay mà phải chờ các admin của trang web duyệt các tin tuyển dụng đó.

## Tạo đơn xin việc cho ứng viên.

Với mỗi ứng viên, để gây ấn tượng đối với nhà tuyển dụng thì trước hết phải có một CV thật là ấn tượng. Ở trang tuyển dụng H&M sẽ có những mẫu CV để hổ trợ bạn tạo một CV cho chính bản thân mình.

Trước hết, để có thể tạo được CV thì bạn phải tạo một tài khoản ứng viên. Sau đó đăng nhập vào tài khoản và nhấn vào “*Quản lý tài khoản*” trên thanh điều hướng và bạn sẽ được chuyển đến trang quản lý sau đó nhấn vào nút “*Thêm CV*” để đi đến trang tạo CV.

Ở đây sẽ bao gồm những thông tin như là: thông tin cá nhân của người dùng, mục tiêu nghề nghiệp, kinh nghiệm làm việc, học vấn, đánh giá tin học, tiếng anh của bản thân, kỹ năng, sở thích, …



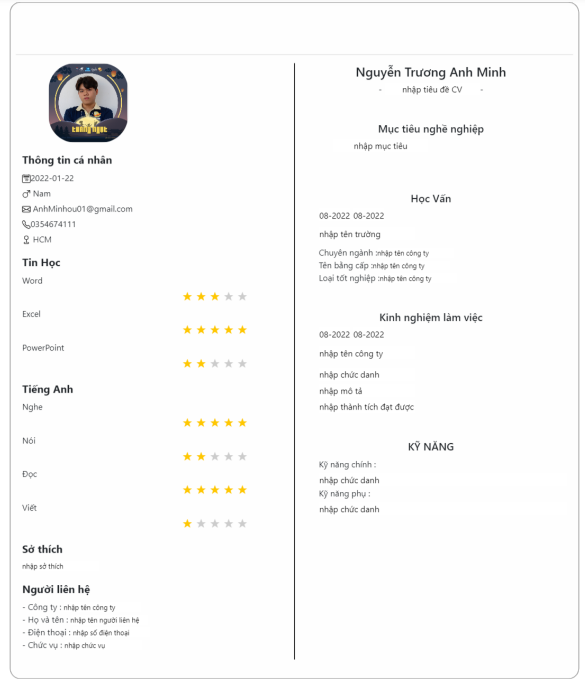
Hình 4.18: Giao diện trang tạo CV.

Sau khi điền đầy đủ thông tin, bạn nhấn vào nút “*Lưu CV*” thì trang web sẽ chụp ảnh cái CV của bạn lại và xuất ra cho bạn một bản PDF tải về máy và hình ảnh đó cũng sẽ được đưa lên *Cloudinary* và sẽ lưu đường link của ảnh đó xuống cơ sở dữ liệu.



Hình 4.19: CV của bạn đã được tải xuống dưới dạng PDF.

Bạn có thể sử dụng file pdf đó để nộp CV cho bản thân mình. Và có thể dùng những cái CV đã được lưu cho từng người để nộp đơn cho các tin tuyển dụng đang có trên hệ thống của trang.



Hình 4.20: Hình ảnh CV của bạn đã được lưu.

## Nộp đơn xin việc cho nhà tuyển dụng.

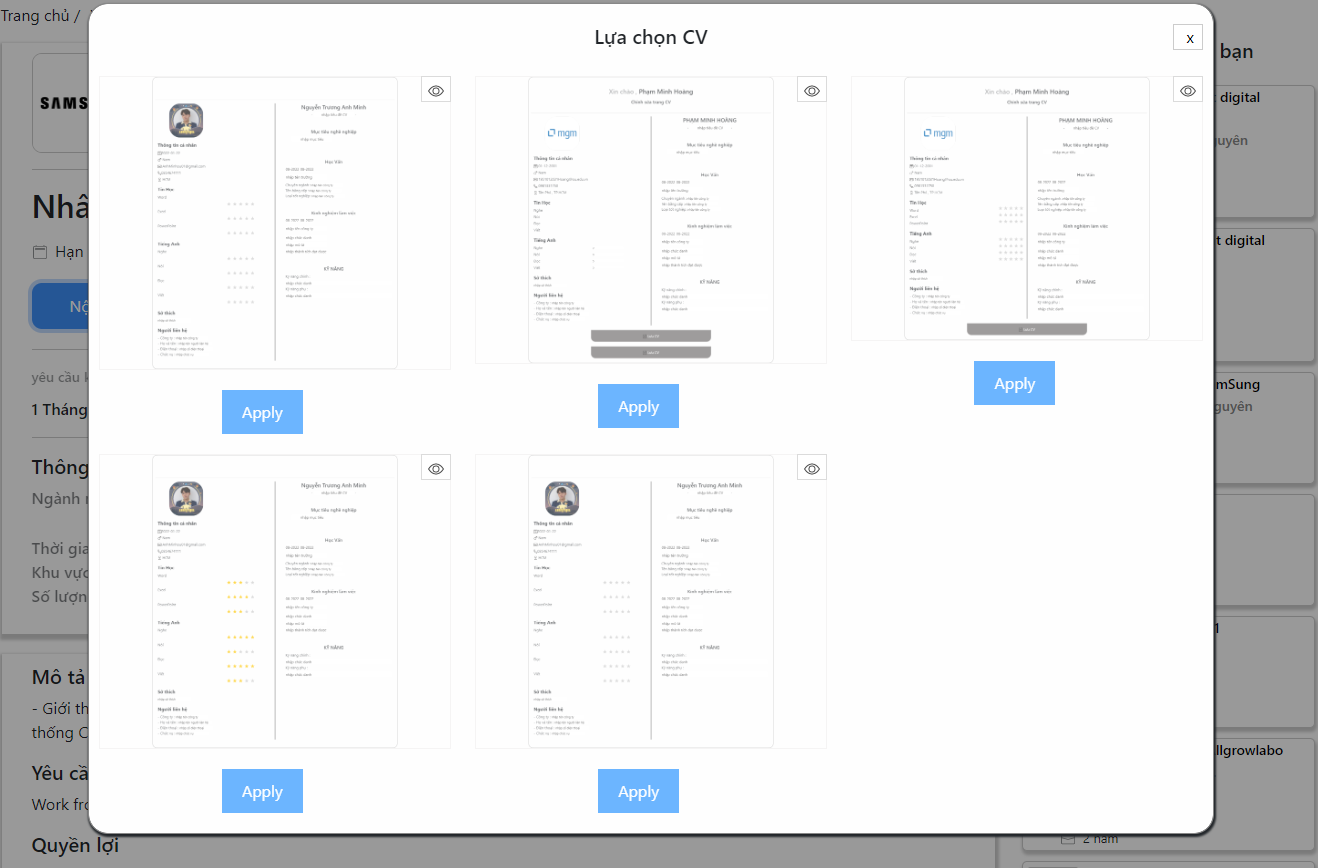
Sau khi đã tạo riêng cho mình một CV thật ấn tượng, bạn có thể sử dụng nó để nộp cho nhà tuyển dụng để ứng tuyển các vị trí tương ứng.

Trước hết bạn sẽ vô trang của một tin tuyển dụng bất kì, sau đó sẽ có nút “*Nộp hồ sơ*” để chọn một CV của mình để nộp hồ sơ.



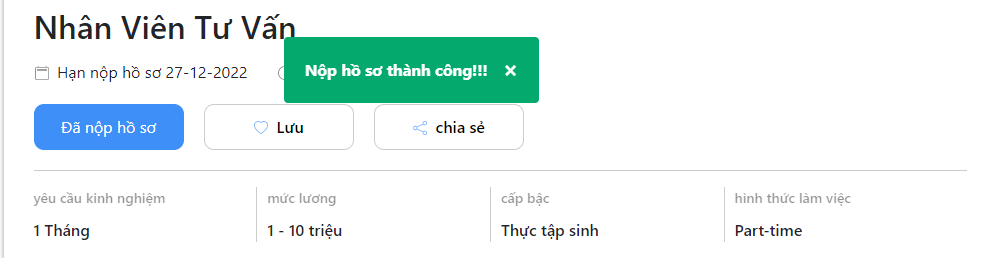
Hình 4.21: Giao diện nộp hồ sơ.

Sau khi ấn vào nút nộp hồ sơ, hệ thống sẽ hiển thị toàn bộ các CV của bạn đã từng tạo, nếu chưa tạo thì hãy tạo ngay cho mình một CV.



Hình 4.22: Giao diện lựa chọn CV.

Sau khi nhấn vào nút “*Apply*” thì CV đó sẽ được gửi đến nhà tuyển dụng và bạn sẽ chỉ để nộp hồ sơ một lần duy nhất cho một công việc.



Hình 4.23: Giao diện nộp CV thành công.

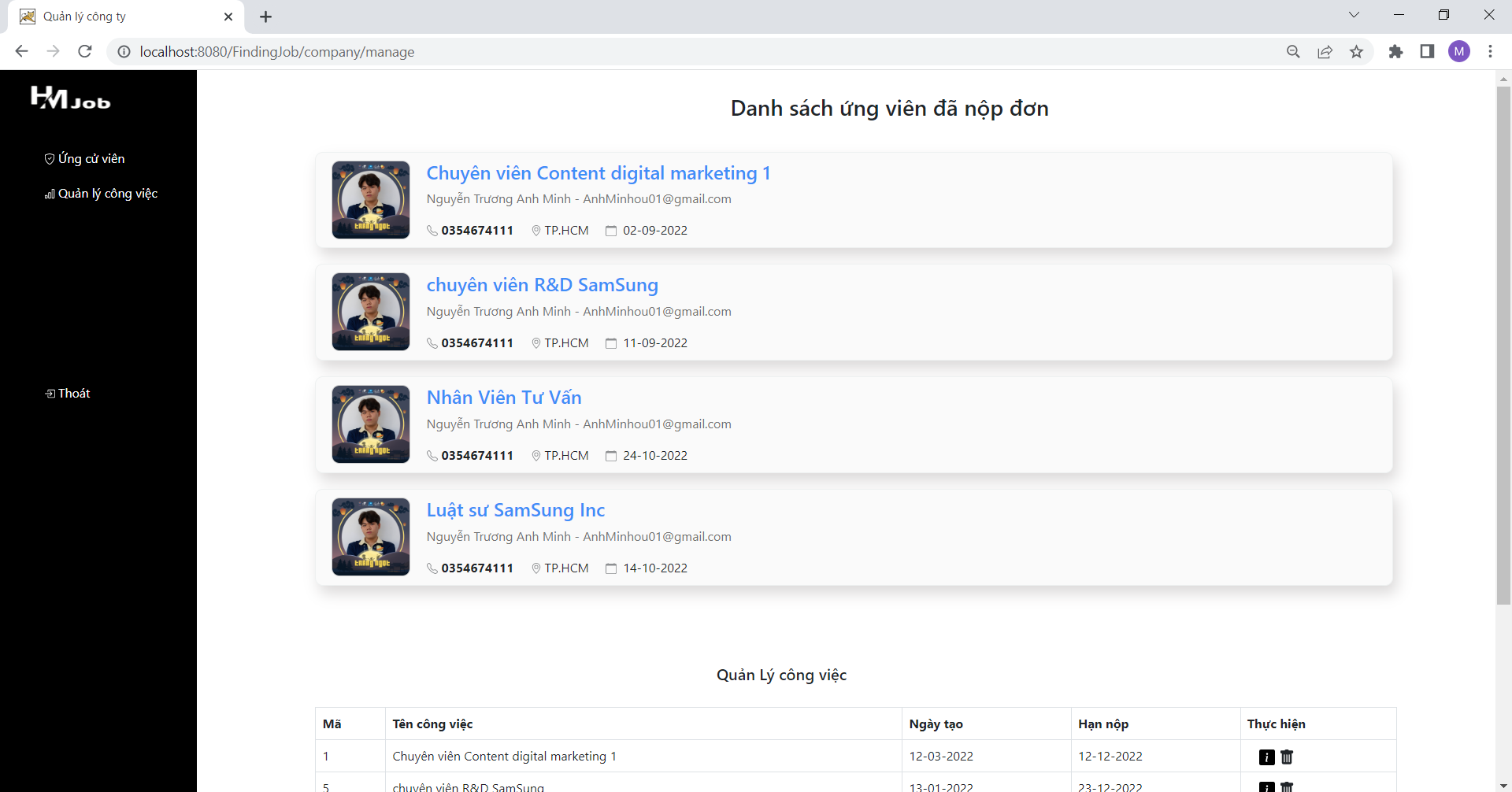
## Quản lý.

Chức năng quản lý sẽ bao gồm 3 cấp độ tương ứng với 3 đối tượng USER là: ứng viên, công ty và admin.

### Quản lý của công ty.

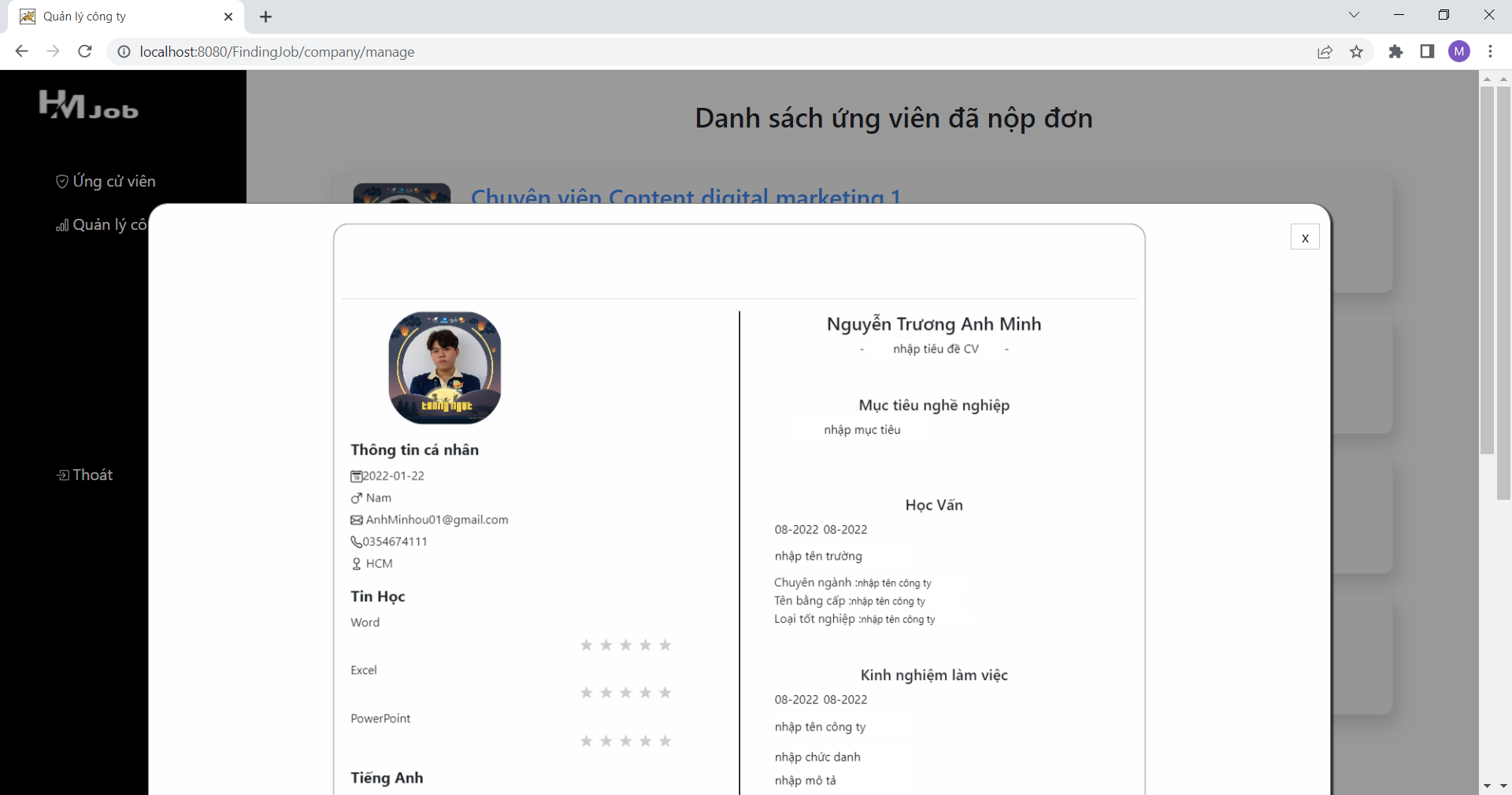
Chức năng quản lý của công ty sẽ bao gồm:

* + - Quản lý công việc.
    - Quản lý các CV của ứng viên đã nộp.



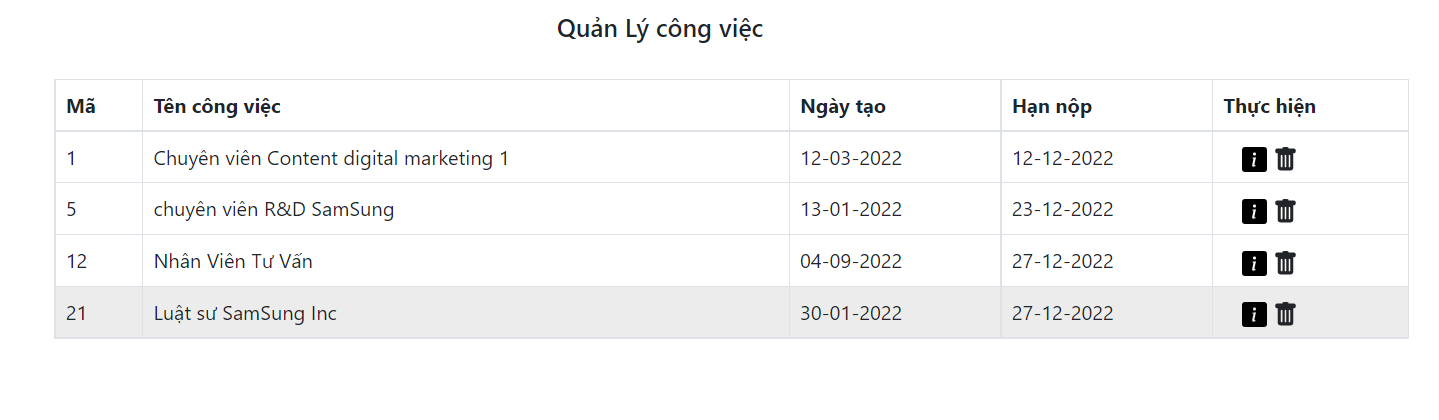
Hình 4.24: Giao diện chức năng quản lý của công ty.

Ở đây khi bạn nhấn vào thông tin ứng viên bất kỳ thì sẽ hiển thị lên toàn bộ CV của người đó.

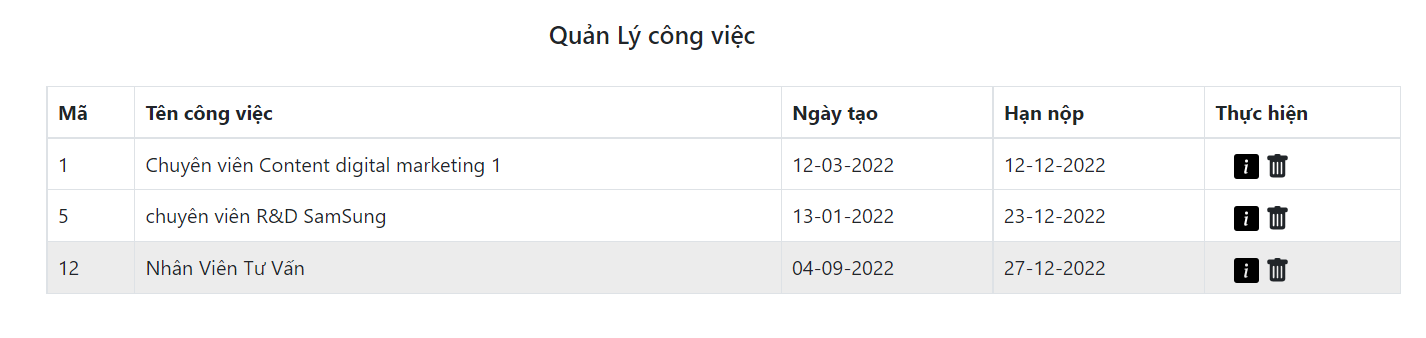


Hình 4.25: Khi nhấn vào thông tin ứng viên bất kỳ.

Ngoài ra ở đây bạn có thể xóa các công việc của bạn đã tạo bằng cách nhấn vào nút thùng rác. Khi đó client sẽ gửi ID của công việc đó dưới dạng dữ liệu JSON truyền về cho sever để tiến hành xóa công việc đó. Khi đó dòng công việc đó sẽ tự động biến mất.



Hình 4.26: Hình ảnh công việc trước khi xóa.

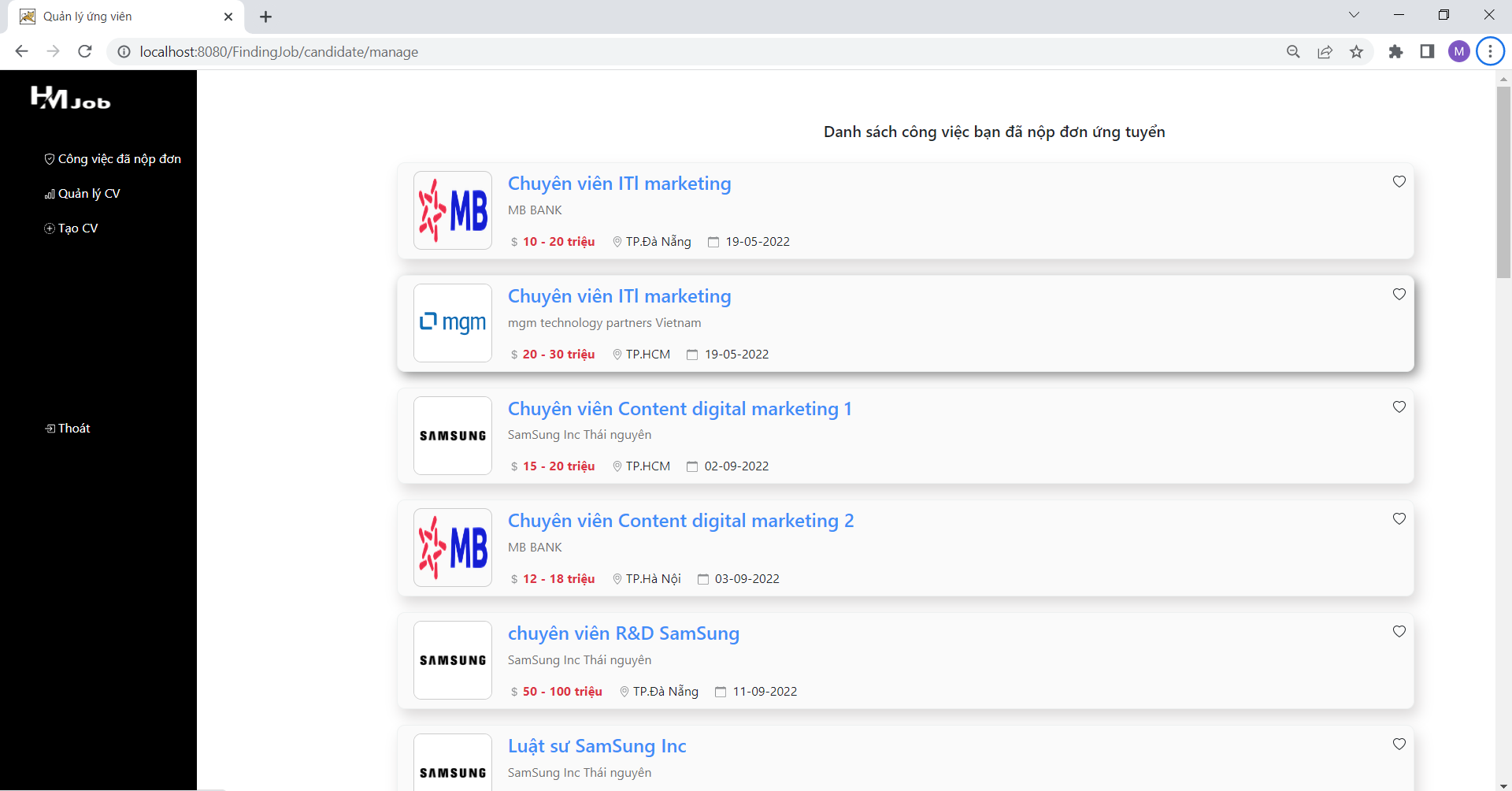


Hình 4.27: Hình ảnh công việc sau khi xóa.

### Quản lý của ứng viên.

Chức năng quản lý của ứng viên sẽ bao gồm:

* + - Hiển thị danh sách công việc bạn đã nộp đơn ứng tuyển.
    - Quản lý CV.



Hình 4.28: Giao diện trang quản lý ứng viên.

Khi ấn vào 1 công việc bất kỳ trên danh sách việc bạn đã ứng tuyến thì trang sẽ được điều hướng đến trang thông tin công việc đó.

### Quản lý của admin.

Quản lý trang của admin sẽ bao gồm các công việc như:

* Kiểm duyệt công ty.

Khi các nhà tuyển dụng đăng ký tài khoản thì tài khoản đó sẽ phải chờ admin duyệt thì đây sẽ hiển thị các công ty đang chờ duyệt. Khi ấn vào nút duyệt sẽ gửi mã ID của công ty đó dưới dạng dữ liệu JSON về cho sever để xử lý cập nhật thuộc tính isChecked của Company thành 1.



Hình 4.29: Chức năng kiểm duyệt công ty.

* Kiểm duyệt tin tuyển dụng.

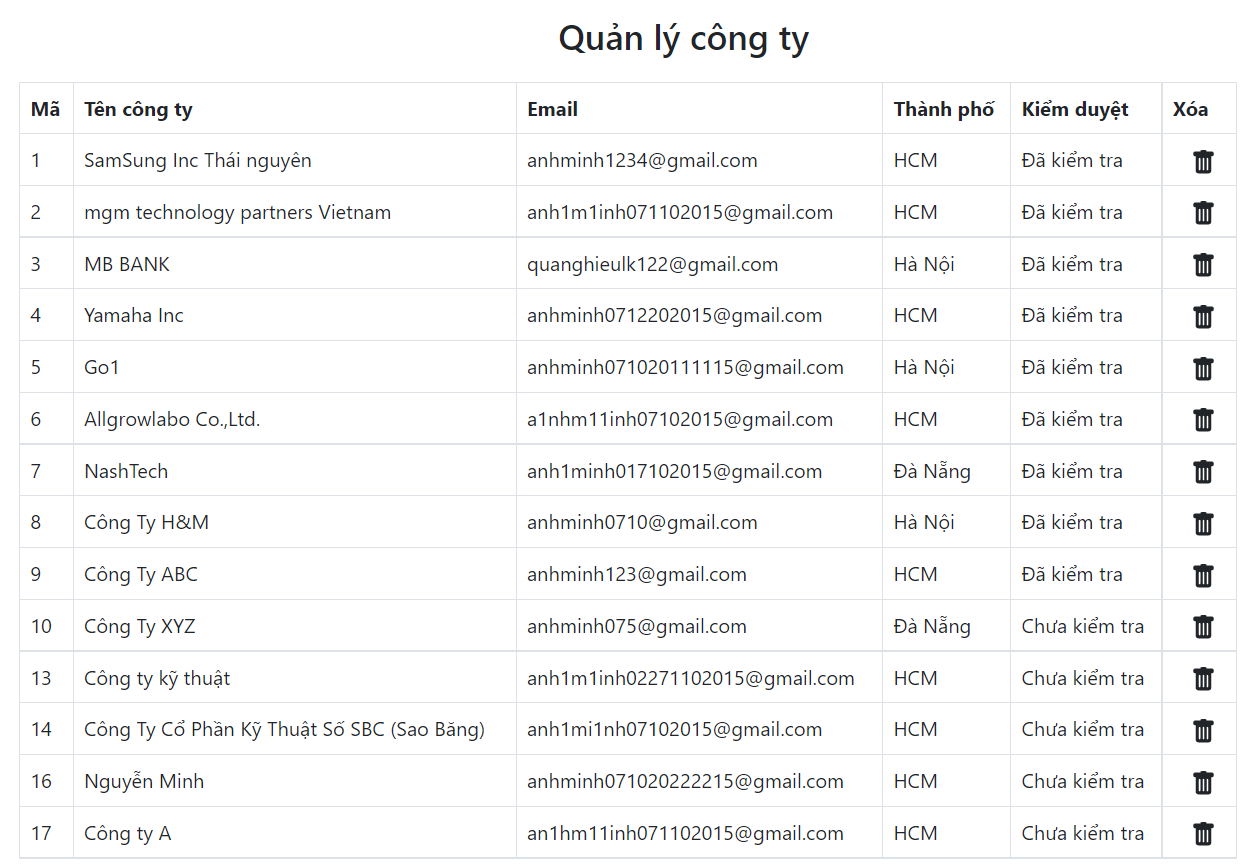
Tương tự như kiểm duyệt công ty, khi nhà tuyển dụng đăng một tin tuyển dụng thì tin đó sẽ trong trạng thái chờ duyệt. Khi ấn vào nút duyệt sẽ gửi mã ID của công việc đó dưới dạng dữ liệu JSON về cho sever để xử lý cập nhật thuộc tính isChecked của JOB thành 1.



Hình 4.30: chức năng kiểm duyệt tin tuyển dụng.

* Quản lý công ty.

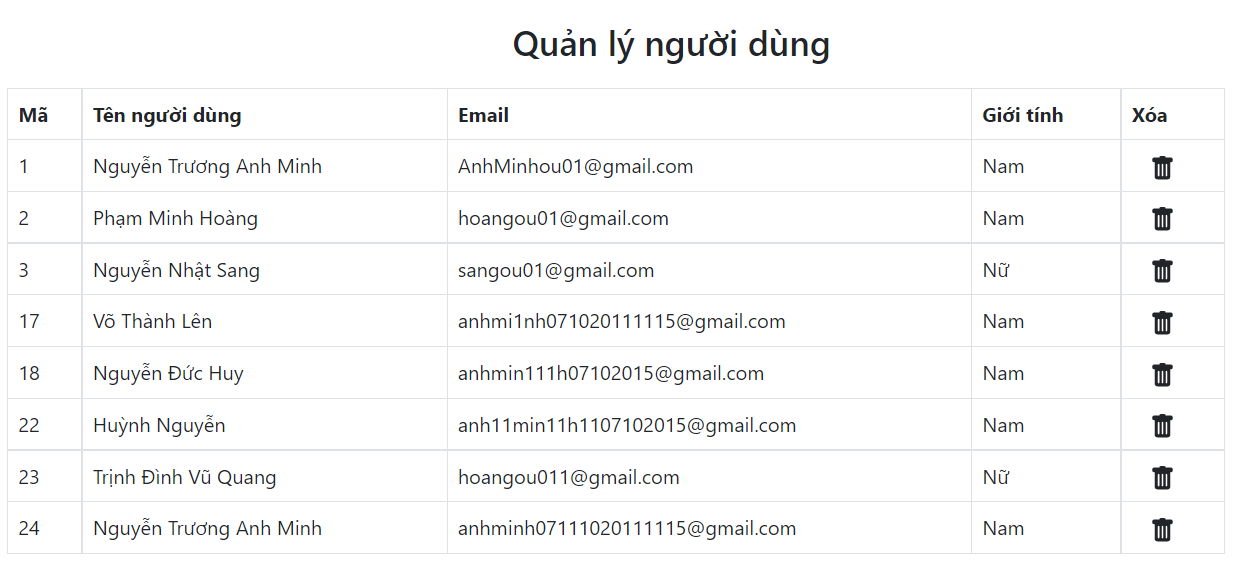
Ở đây sẽ hiển thị toàn bộ danh sách các công ty cho người quản lý để có thể thực hiện các chức năng kiểm tra hay xóa công ty đó. Hệ thống sẽ hiển thị các thông tin như tên công ty, ngày đăng ký, đã kiểm duyệt hay chưa, … Sau khi nhấn vào nút thùng rác thì thuộc tính isDeleted của công ty sẽ được đổi thành 1 và người dùng đó không thể đăng nhập vô hệ thống được nữa.



Hình 4.31: Chức năng quản lý công ty.

* Quản lý người dùng.

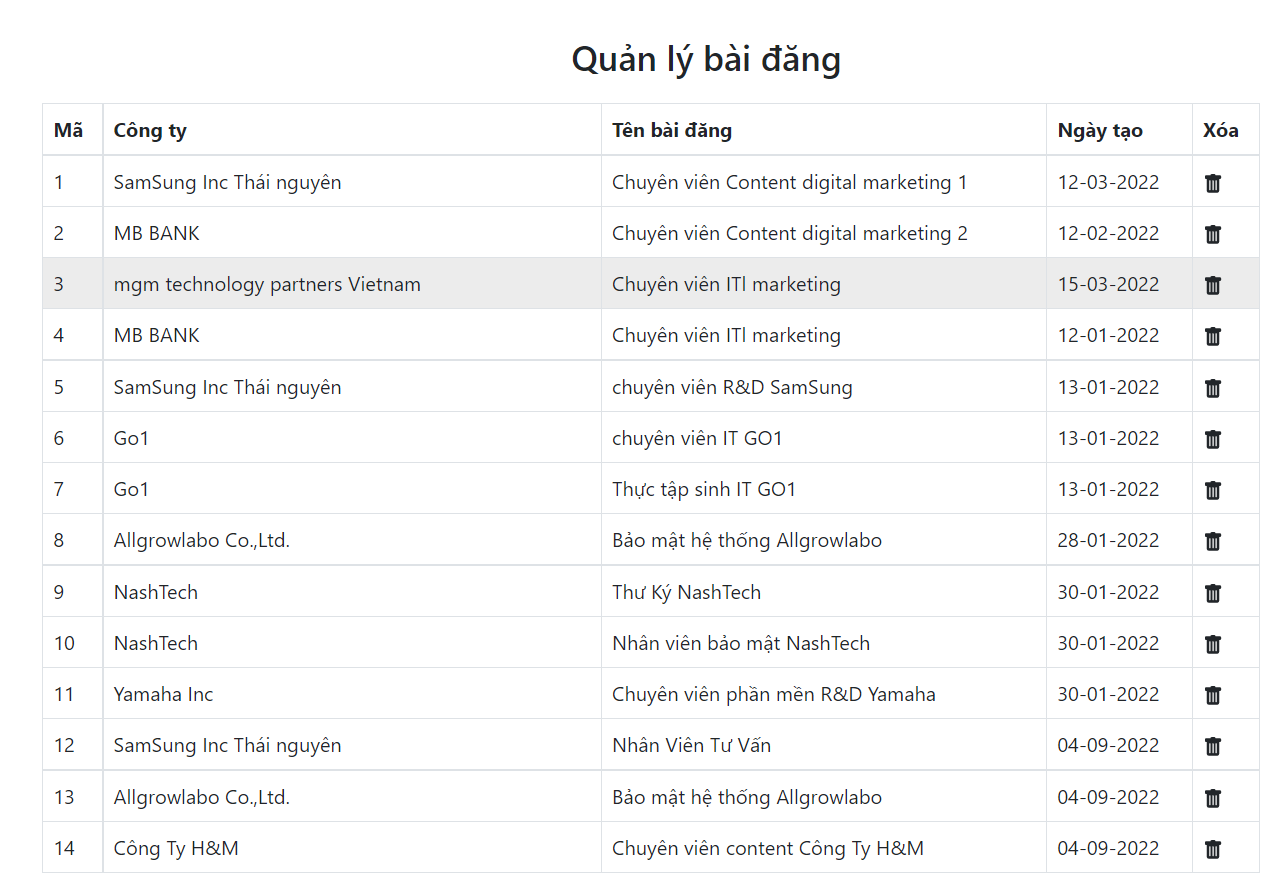
Tương tự như quản lý công ty, nó cũng sẽ hiển thị các thông tin cơ bản của người dùng như là: tên người dùng, email, giới tính, …



Hình 4.32: Chức năng quản lý người dùng.

* Quản lý tin tuyển dụng

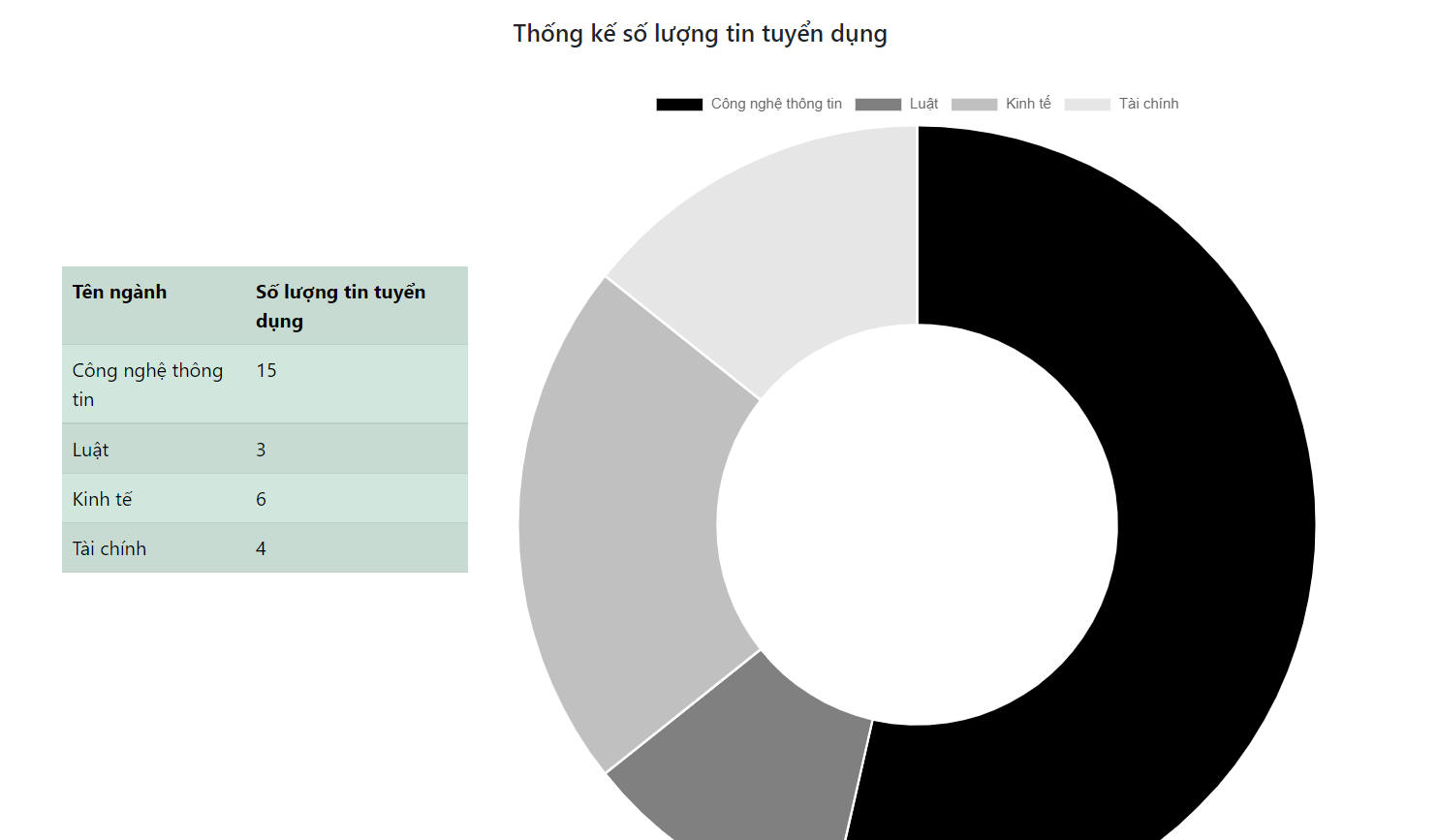
Hiển thị toàn bộ các tin tuyển dụng của tất cả các công ty. Người quản lý có thể xóa những tin tuyển dụng không phù hợp.



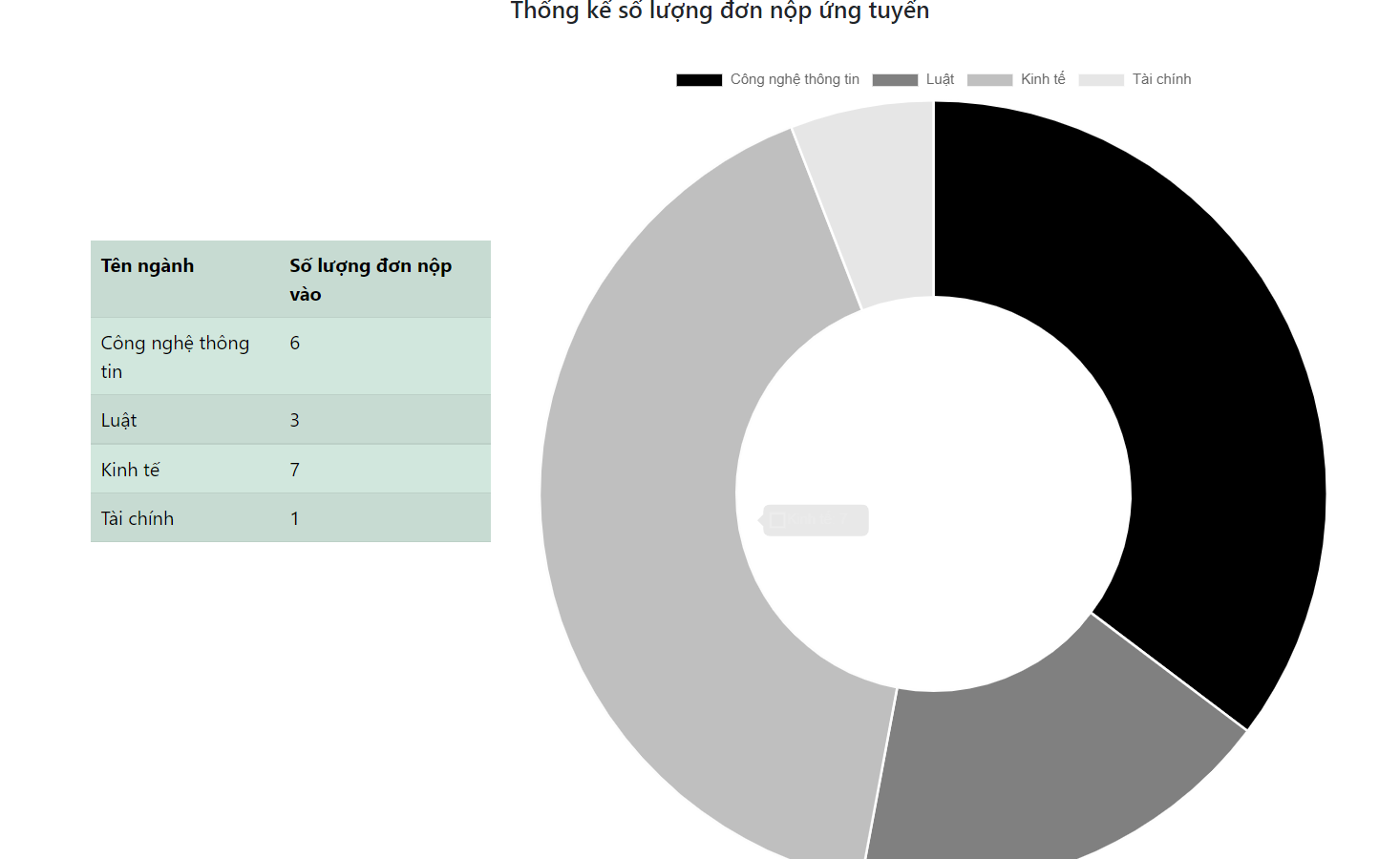
Hình 4.33: Chức năng quản lý tin tuyển dụng.

* Thông kê, báo cáo.

Trang web nào cũng sẽ có cho mình một chức năng thống kê, báo cáo riêng thì ở đây sẽ thống kê dựa trên số lượng tin tuyển dụng theo từng ngành, số lượng nộp đơn theo từng ngành theo dạng bảng và dạng biểu đồ tròn.



Hình 4.34: Thống kê số lượng tin tuyển dụng.



Hình 4.35: Thống kê số lượng nộp hồ sơ.

## Các tài khoản để sử dụng hệ thống.

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên đăng nhập | Mật khẩu | Vai trò |
| 1 | ADMIN | Anhminh123@ | Admin |
| 2 | anhminh1234@gmail.com | Anhminh123@ | Nhà tuyển dụng |
| 3 | anhminh07102015@gmail.com | Anhminh123@ | Ứng viên |

Bảng 4.1: Các tài khoản để sử dụng hệ thống.

# Tổng kết.

## Kết luận.

Qua bài đồ án này, nhóm đã rút ra được nhiều kinh nghiệm cho bản thân trong việc làm việc nhóm, phân chia công việc với nhau, quản lý thời gian của bản thân, có thêm các bài học trong việc phát triển một trang web, có thêm kiến thức về Spring MVC và quá trình xử lý dữ liệu với database, biết được thêm các phương thức HTTP, biết phát triển và triển khai một quy trình xử lý hổ trợ tuyển dụng, tìm kiếm việc làm,…

## Những vấn đề còn tồn tại.

* Mẫu CV chưa được đa dạng.
* Chưa hổ trợ tối ưu cho việc tìm kiếm việc làm.
* Tìm kiếm chưa có quá nhiều tiêu chí.
* Dữ liệu còn hạn chế để phát triển hướng marchine learning.
* Chưa có chức năng gợi ý hổ trợ người dùng.
* Chưa có chức năng nhắn tin với nhà tuyển dụng.
* Chưa có chức năng thông báo.

## Hướng phát triển.

* Làm thêm các mẫu CV cho người dùng lựa chọn.
* Thêm nhiều tiêu chí lọc tìm kiếm việc làm.
* Thêm chức năng thông báo cho nhà tuyển dụng khi có ứng viên nộp hồ sơ.
* Triển khai chức năng gợi ý, automatching.
* Triển khai chức năng nhắn tin thời gian thực.
* Thông báo cho người dùng.

TÀI LIỆU THAM KHẢO

[1] C.N.Long, “Một số khái niệm cơ bản về Spring MVC,” Cập nhật 21/03/2021 09:54. [Trực tuyến]. Địa chỉ: *https://viblo.asia/p/mot-so-khai-niem-co-ban-ve-spring-mvc-6J3Zg0PWlmB*.

[2] N.T.T.Nghĩa, “Phương thức GET và POST,” Cập nhật 25/06/2017. [Trực tuyến]. Địa chỉ: *https://viblo.asia/p/phuong-thuc-get-va-post-aWj53VBYl6m.*

[3] N.H.Nam, “Hướng dẫn Spring Security đơn giản dễ hiểu,” Cập nhật 31/05/2019. [Trực tuyến]. Địa chỉ: *https://viblo.asia/p/huong-dan-spring-security-co-ban-de-hieu-OeVKBdedlkW*.

[4] ITNavi, “Apache Tomcat là gì? Tìm hiểu tổng quan về Apache Tomcat,” Cập nhật 02/12/2020. [Trực tuyến]. Địa chỉ: *https://itnavi.com.vn/blog/apache-tomcat-la-gi*.

[5] N.Ngọc, “Sử dụng cloudinary để quản lý ảnh của bạn,” Cập nhật 28/03/2016. [Trực tuyến]. Địa chỉ: *https://viblo.asia/p/su-dung-cloudinary-de-quan-ly-anh-cho-ung-dung-cua-ban-E7bGoxggv5e2*.

[6] N.Hai, “API là gì? Những điểm bạn cần hiểu rõ,” Cập nhật 14/05/2019. [Trực tuyến]. Địa chỉ: *https://viblo.asia/p/api-la-gi-nhung-diem-ban-can-hieu-ro-gGJ59x9DlX2.*

[7] Glints Viêt Nam, “Báo cáo thị trường việc làm và thị hiếu người dùng của Glints,” Cập nhật 19/03/2022. [Trực tuyến]. Địa chỉ: *https://glints.com/vn/blog/bao-cao-thi-truong-nhu-cau-tuyen-dung-tai-glints-thang-2-2022/#.Y18y\_HZBxPY.*

[8] N.T.Hải, “JDBC là gì?,” Cập nhật 12/07/2021. [Trực tuyến]. Địa chỉ*: https://viblo.asia/p/jdbc-la-gi-khong-can-hoc-jdbc-dau-Az45bDwzZxY.*

[9] M.Duc, “JSP là gì? Tại sao nên sử dụng JSP?,” Cập nhật 04/11/2020. [Trực tuyến]. Địa chỉ: *https://laptrinhvien.net/jsp-la-gi-tai-sao-nen-su-dung-jsp*.