**TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI**

**TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

**ĐỒ ÁN TỐT NGHIỆP**

**Xây dựng hệ thống mạng xã hội tìm kiếm việc làm**

**NGUYỄN ĐỨC THÀNH**

thanh.nd183991@sis.hust.edu.vn

**Ngành: Công nghệ thông tin**

**Chuyên ngành: Công nghệ thông tin**

|  |  |
| --- | --- |
| **Giảng viên hướng dẫn:** | ThS. Nguyễn Tiến Thành |
| **Bộ môn:** | Công nghệ phần mềm |
| **Trường:** | Trường Công nghệ thông tin và Truyền thông |
| **HÀ NỘI, 08/2022** | |

# Lời cam kết

Họ và tên sinh viên: Nguyễn Đức Thành. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

Điện thoại liên lạc: 0386851140. . . . . . . . . . . . . . . Email:thanh.nd183991@sis.hust.edu.vn

Lớp: Công nghệ thông tin 01 –K63 .. . . . . . . . . . . Hệ đào tạo: Chính quy . . . . . . . . . . . . . .

Tôi –*Nguyễn Đức Thành*– cam kết Đồ án Tốt nghiệp (ĐATN) là công trình nghiên cứu của bản thân tôi dưới sự hướng dẫn của Thạc sỹ Nguyễn Tiến Thành. Các kết quả nêu trong ĐATN là trung thực, là thành quả của riêng tôi, không sao chép theo bất kỳ công trình nào khác. Tất cả những tham khảo trong ĐATN – bao gồm hình ảnh, bảng biểu, số liệu, và các câu từ trích dẫn – đều được ghi rõ ràng và đầy đủ nguồn gốc trong danh mục tài liệu tham khảo. Tôi xin hoàn toàn chịu trách nhiệm với dù chỉ một sao chép vi phạm quy chế của nhà trường.

|  |  |
| --- | --- |
|  | *Hà Nội, ngày 08 tháng 08 năm 2022*  Tác giả Đồ án tốt nghiệp  Thành  Nguyễn Đức Thành |

# Lời cảm ơn

Trước hết em xin chân thành cảm ơn thầy Thạc Sỹ Nghuyễn Tiến Thành- Bộ môn Khoa học máy tính – Trường Công nghệ Thông tin và Truyền thông – Trường Đại học Bách khoa Hà Nội, người đã hướng dẫn em rất nhiều trong suốt quá trình tìm hiểu làm đồ án tốt nghiệp. Đồng thời, em xin cám ơn các thầy cô trong trường đã cung cấp cho em những kiến thức cơ bản, cần thiết để em hoàn thành đồ án tốt nghiệp này.

Em xin gửi lời cảm ơn đến các thành viên lớp Công nghệ thông tin IT3-01, những người bạn đã luôn ở bên cạnh động viên, tạo điều kiện thuận lợi giúp em hoàn thành tốt học phần.

Sau cùng, em xin gửi lời cảm ơn đến gia đình, bạn bè đã tạo mọi điều kiện để em xây dựng thành công học phần này.

Mặc dù bản thân em đã cố gắng nhưng do thời gian, kiến thức và kinh nghiệm có hạn nên sản phẩm và báo cáo của em còn có thể có nhiều thiếu xót…Vì vậy, em rất mong nhận được sự thông cảm và nhận xét từ thầy để sản phẩm cũng như báo cáo của em được hoàn thiện hơn.

# Tóm tắt

Ngày nay, các ứng dụng công nghệ thông tin đã được áp dụng trong mọi lĩnh vực nghiên cứu khoa học kĩ thuật cũng như trong đời sống, máy tính trở thành công cụ trợ giúp đắc lực cho con người trong lưu trữ, phân tích và xử lý thông tin. Đối với lĩnh vực tuyển dụng việc làm, áp dụng công nghệ thông tin mang lại những lợi ích thiết thực hơn bao giờ hết. Trước đây, khi hạ tầng công nghệ chưa phát triển, mọi người thường tìm kiếm công việc qua các mẩu tin trên báo. Người tim việc cũng có thể dựa vào các công ty môi giới việc làm nhưng cũng chỉ đáp ứng một phần nhỏ thị trường lao động. Những cách này thường tốn thời gian, công sức với cả người tuyển dụng lẫn người tìm việc.

Áp dụng công nghệ thông tin trong việc tìm kiếm việc làm đã gần như xóa bỏ những khuyết điểm của những hình thức tìm kiếm cũ. Tuy nhiên, các nội dụng tuyển dụng được đăng khắp mọi nơi trên Internet: mạng xã hội, trang website của công ty, mẩu tin tuyển việc trên các website blog… Điều này gây khó khăn cho người lao động khi tìm kiếm nguồn thông tin xác thực và cản trở nhà tuyển dụng tiếp cận các ứng viên tiềm năng.

Để giải quyết hạn chế nêu trên, em chọn đề tài cho đồ án tốt nghiệp của mình là “Xây dựng hệ thống mạng xã hội tìm kiếm việc làm”. Hệ thống được phát triển nhằm tạo môi trường giao tiếp hiệu quả giữa người lao động và nhà tuyển dụng.

Hệ thống được phát triển theo mô hình thiết kế client-server. Bên phía client, hệ thống sử dụng thư viện ReactJS để xây dựng giao diện cho người dùng, phía server hệ thống sử dụng NodeJS và cơ sở dữ liệu phi quan hệ NoSQL MongoDB. Hệ thống đã đi vào thử nghiệm và bước đầu hoạt động ổn định trên mạng LAN.

# Mục lục

[Lời cam kết ii](#_Toc107433565)

[Lời cảm ơn iii](#_Toc107433566)

[Tóm tắt iv](#_Toc107433567)

[Mục lục v](#_Toc107433568)

[Danh mục hình vẽ viii](#_Toc107433569)

[Danh mục bảng ix](#_Toc107433570)

[Danh mục công thức x](#_Toc107433571)

[Danh mục các từ viết tắt xi](#_Toc107433572)

[Danh mục thuật ngữ xii](#_Toc107433573)

[Chương 1 Giới thiệu đề tài 1](#_Toc107433574)

[1.1 Đặt vấn đề 1](#_Toc107433575)

[1.2 Mục tiêu và phạm vi đề tài 1](#_Toc107433576)

[1.3 Định hướng giải pháp 2](#_Toc107433577)

[1.4 Bố cục đồ án 3](#_Toc107433578)

[Chương 2 Khảo sát và phân tích yêu cầu 4](#_Toc107433579)

[2.1 Khảo sát hiện trạng 4](#_Toc107433580)

[2.2 Tổng quan chức năng 6](#_Toc107433581)

[2.2.1 Biểu đồ use case tổng quan 6](#_Toc107433582)

[2.2.2 Biểu đồ use case phân rã “Quản lý người dùng” 10](#_Toc107433583)

[2.2.3 Biểu đồ use case phân rã “Quản lý bài đăng” 11](#_Toc107433584)

[2.2.4 Biểu đồ use case phân rã “Quản lý công ty” 12](#_Toc107433585)

[2.2.5 Biểu đồ use case phân rã “Quản lý tin tuyển dụng” 13](#_Toc107433586)

[2.2.6 Biểu đồ use case phân rã “Quản lý kỹ năng” 14](#_Toc107433587)

[2.2.7 Biểu đồ use case phân rã “Theo dõi” 15](#_Toc107433588)

[2.2.8 Biểu đồ use case phân rã “Tìm kiếm” 16](#_Toc107433589)

[2.2.9 Biểu đồ use case phân rã “Quản lý công ty của tôi” 17](#_Toc107433590)

[2.2.10 Quy trình nghiệp vụ 18](#_Toc107433591)

[2.3 Đặc tả chức năng 22](#_Toc107433592)

[2.3.1 Đặc tả use case “Quên mật khẩu” 22](#_Toc107433593)

[2.3.2 Đặc tả use case “Theo dõi người dùng” 24](#_Toc107433594)

[2.3.3 Đặc tả use case “Cập nhật thông tin” 25](#_Toc107433595)

[2.3.4 Đặc tả usecase “Đăng bài” 27](#_Toc107433596)

[2.3.5 Đặc tả use case “Trao đổi” 29](#_Toc107433597)

[2.3.6 Đặc tả use case “Ứng tuyển công việc” 29](#_Toc107433598)

[2.3.7 Đặc tả use case “Đăng tin tuyển dụng” 31](#_Toc107433599)

[2.4 Yêu cầu phi chức năng 33](#_Toc107433600)

[2.4.1 Yêu cầu bảo mật 33](#_Toc107433601)

[2.4.2 Yêu cầu vận hành 33](#_Toc107433602)

[2.4.3 Yêu cầu giao diện 34](#_Toc107433603)

[Chương 3 Công nghệ sử dụng 35](#_Toc107433604)

[3.1 Ngôn ngữ lập trình Javascript 35](#_Toc107433605)

[3.2 Công nghệ lập trình giao diện ReactJS 35](#_Toc107433606)

[3.3 Công nghệ lập trình máy chủ NodeJS 37](#_Toc107433607)

[3.4 Công nghệ JWT 38](#_Toc107433608)

[3.5 Công nghệ lập trình thời gian thực SocketIO 38](#_Toc107433609)

[3.6 Cơ sở dữ liệu MongoDB 39](#_Toc107433610)

[Chương 4 Phát triển và triển khai ứng dụng 40](#_Toc107433611)

[4.1 Thiết kế kiến trúc 40](#_Toc107433612)

[4.1.1 Lựa chọn kiến trúc phần mềm 40](#_Toc107433613)

[4.1.2 Thiết kế tổng quan 42](#_Toc107433614)

[4.1.3 Thiết kế chi tiết gói 45](#_Toc107433615)

[4.2 Thiết kế chi tiết 45](#_Toc107433616)

[4.2.1 Thiết kế giao diện 45](#_Toc107433617)

[4.2.2 Thiết kế lớp 45](#_Toc107433618)

[4.2.3 Thiết kế cơ sở dữ liệu 45](#_Toc107433619)

[4.3 Xây dựng ứng dụng 45](#_Toc107433620)

[4.3.1 Thư viện và công cụ sử dụng 45](#_Toc107433621)

[4.3.2 Kết quả đạt được 46](#_Toc107433622)

[4.3.3 Minh hoạ các chức năng chính 46](#_Toc107433623)

[4.4 Kiểm thử 46](#_Toc107433624)

[4.5 Triển khai 46](#_Toc107433625)

[Chương 5 Các giải pháp và đóng góp nổi bật 47](#_Toc107433626)

[Chương 6 Kết luận và hướng phát triển 48](#_Toc107433627)

[6.1 Kết luận 48](#_Toc107433628)

[6.2 Hướng phát triển 49](#_Toc107433629)

[Tài liệu tham khảo 50](#_Toc107433630)

[Phụ lục 1](#_Toc107433631)

# Danh mục hình vẽ

[**Hình 1** Sơ đồ Usecase tổng quan 6](#_Toc107433547)

[**Hình 2** Phân rã Usecase “Quản lý người dùng” 10](#_Toc107433548)

[**Hình 3** Phân rã Usecase “Quản lý bài đăng” 11](#_Toc107433549)

[**Hình 4** Phân rã Usecase “Quản lý công ty” 12](#_Toc107433550)

[**Hình 5** Phân rã Usecase “Quản lý tin tuyển dụng” 13](#_Toc107433551)

[**Hình 6** Phân rã Usecase “Quản lý kỹ năng” 14](#_Toc107433552)

[**Hình 7** Phân rã Usecase “Theo dõi” 15](#_Toc107433553)

[**Hình 8** Phân rã Usecase “Tìm kiếm” 16](#_Toc107433554)

[**Hình 9** Phân rã Usecase “Quản lý công ty của tôi” 17](#_Toc107433555)

[**Hình 10** Biểu đồ hoạt động đăng tin tuyển dụng 20](#_Toc107433556)

[**Hình 11** Biều đồ hoạt động thay đổi chủ của tin tuyển dụng 22](#_Toc107433557)

[**Hình 12** Cấu trúc, thành phần của ReactJS 36](#_Toc107433558)

[**Hình 13** Mô hình MVC 41](#_Toc107433559)

[**Hình 14** Mô hình MVVM 41](#_Toc107433560)

[**Hình 15** Biều đồ phụ thuộc gói Client 42](#_Toc107433561)

[**Hình 16** Tổng quan gói server 44](#_Toc107433562)

# Danh mục bảng

[**Bảng 1** Quy trình đăng tin tuyển dụng 19](#_Toc107061571)

[**Bảng 2** Quy trình thay đổi chủ của tin tuyển dụng 21](#_Toc107061572)

[**Bảng 3** Đặc tả usecase “Quên mật khẩu” 24](#_Toc107061573)

[**Bảng 4** Đặc tả use case “Theo dõi người dùng” 25](#_Toc107061574)

[**Bảng 5** Đặc tả use case “Cập nhật thông tin” 26](#_Toc107061575)

[**Bảng 6** Đặc tả use case “Đăng bài” 28](#_Toc107061576)

[**Bảng 7** Đặc tả use case “Trao đổi” 29](#_Toc107061577)

[**Bảng 8** Đặc tả use case “Ứng tuyển công việc” 30](#_Toc107061578)

[**Bảng 9** Đặc tả use case “Đăng tin tuyển dụng” 32](#_Toc107061579)

[**Bảng 16** Danh sách thư viện và công cụ sử dụng 45](#_Toc107061580)

# Danh mục công thức

# Danh mục các từ viết tắt

|  |  |
| --- | --- |
| API | Application Programming Interface  Giao diện lập trình ứng dụng |
| **IT** | Information Technology  Công nghệ thông tin |
| **UI** | User Interface  Giao diện người dùng |
| **HTML** | HyperText Markup Language  Ngôn ngữ đánh dấu siêu văn bản |
| **CNTT** | Công nghệ thông tin |
| **CSS** | Cascading Style Sheets  Là ngôn ngữ định dạng văn bản |
| **SV** | Sinh viên |
| **JWT** | JSON Web Tokens  Chuỗi xác thực người dùng |
| **SSR** | SERVER-SIDE RENDERING  Cơ chế hiển thị trang website |

# Danh mục thuật ngữ

|  |  |
| --- | --- |
| Browser | Trình duyệt |
| **Cache memory** | Bộ nhớ đệm |
| **Compiler** | Trình biên dịch |
| **Javascript** | Là ngôn ngữ lập trình |
| **ReactJS** | Một thư viện viết bằng **javascript**, dùng để **xây dựng giao diện người dùng**. |
| **MUI** | Thư viện hỗ trợ ReactJS xây dựng giao diện. |
| **NodeJS** | Là nền tảng được xây dựng dựa trên “V8 Javascript engine”(giúp phân tích và thông dịch javascript thành mã máy) được viết bằng C++ |
| **MongoDB** | Cơ sơ dữ liệu phi quan hệ |
| **Server** | Là một máy chủ hay máy phục vụ là một phần của máy tính phần cứng hoặc phần mềm mà cung cấp chức năng cho các chương trình khác. |
| **JWT** | Giao thức xác thực người dùng trên internet |
| **Redux** | Công cụ quản lý trạng thái của giao diện |
| **RSA** | Một thuật toán mật mã hóa khóa công khai. |
| **MVC** | Mô hình kiến trúc ứng dụng |
| **MVVM** | Mô hình kiến trúc ứng dụng |
| **Client** | Máy khách |
| **JSON** | JavaScript Object Notation  Một định dạng trao đổi dữ liệu nhẹ. |
| **base64** | Phương thức chuyển đổi dạng mã hóa 2 chiều từ binary sang string để có thể gửi đi được trong network một cách dễ dàng |
| **Public key** | Khóa công khai khi mã hóa |
| **Private key** | Khóa bí mật |

# Giới thiệu đề tài

## Đặt vấn đề

Trong thời đại công nghiệp hóa, hiện đại hóa ngày nay, cuộc sống của con người ngày càng thay đổi trên nhiều lĩnh vực. Đối với lĩnh vực tìm kiếm việc làm, trước đây người lao động tiếp cận việc làm chủ yếu thông qua các tờ rơi, các mẩu tin nhỏ trên các tờ báo, hay hiện đại hơn là thông qua các công ty môi giới tuyển dụng. Nhưng đối với một xã hội công nghệ phát triển mạnh mẽ như hiện nay, các hình thức đó đã không phù hợp và ít xuất hiện.

Thay vào đó, việc áp dụng công nghệ thông tin vào trong lĩnh vực tuyển dụng được phổ biến ngày càng rộng rãi. Đăng tin tuyển dụng việc làm có thể dưới nhiều hình thức khác nhau trên Internet: mạng xã hội, trang website của công ty tuyển dụng. Đối với các bạn sinh viên, cầu nối giữa nhà trường và doanh nghiệp giúp cho sinh viên tiếp cận việc làm thông qua các buổi trò chuyện online trong thời đại dịch bệnh hiện nay. Những hình thức đó đã khắc phục được những hạn chế của những hình thức trước kia, nhưng vẫn còn rào cản đối với người lao động và nhà tuyển dụng khi tiếp cận với số lượng thông tin nhiều và khó xác thực.

Hiểu được điều đó, em chọn đề tài “Xây dựng mạng xã hội tìm kiếm việc làm” mục đích nhằm tạo ra môi trường giao tiếp hiệu quả giữa người lao động và nhà tuyển dụng. Hệ thống giúp cho họ có thể “tìm đến nhau” một cách dễ dàng và nhanh nhất.

## Mục tiêu và phạm vi đề tài

Mục tiêu của đề tài là tạo ra một nơi giúp người lao động và nhà tuyển dụng tìm kiếm nhau dễ dàng, hiệu quả và đáng tin cậy.

Đối với quản trị viên, hệ thống cung cấp các chức năng giúp thống kê xét duyệt công việc, công ty, nhận thông báo từ người dùng…

Đối với quản lý công ty, hệ thống cung cấp các chức năng cho phép người quản lý giám sát các nhà tuyển dụng, việc làm của công ty…

Đối với nhà tuyển dụng của công ty, hệ thống cung cấp các chức năng giúp nhà tuyển dụng đăng tin tạo việc làm, tạo các bài viết giúp quảng bá công ty…

Đối với người lao động (tìm việc), hệ thống cung cấp các chức năng giúp người lao động tìm kiếm các công việc phù hợp nhất đối với bản thân dựa trên kỹ năng có ở trong hồ sơ của họ.

Đồng thời, hệ thống có cung cấp dịch vụ trao đổi và thông báo để giúp các thành viên trong mạng lưới giao tiếp với nhau một cách trực tiếp, dễ dàng và hiệu quả.

## Định hướng giải pháp

Hệ thống được phát triển dựa trên giao diện đồ họa người dùng cho phép người dùng hoàn thành tác vụ với tốc độ nhanh hơn so với các thiết bị sử dụng giao diện cũ hơn. Hệ thống sẽ tránh hoàn toàn việc sử dụng giao diện dòng lệnh, bởi vì, nó khó ghi nhớ câu lệnh và phải nhập câu lệnh văn bản. Ưu điểm của việc sử GUI bởi có các phần tử đồ họa tương tác với các chức năng giống nhau giúp người dùng dễ dàng ghi nhớ và nắm bắt thông tin trực quan, sinh động.

Hệ thống sẽ được phát triển dựa trên nền tảng website, một ứng dụng có sẵn và được tích hợp trên mọi loại thiết bị điện tử hiện nay. Hệ thống được xây dựng dựa trên mô hình client-server, một mô hình được sử dụng rộng rãi bởi vì các ưu điểm của nó: tốc độ truy cập nhanh, khả năng mở rộng cao, đáng tin cậy, an toàn, và hoạt động với bất kỳ loại ứng dụng nào… Hệ thống sẽ bao gồm hai phần riêng biệt: front-end, back-end được giao tiếp thông qua giao thức JWT để xác định danh tích người dùng.

FrontEnd thực hiện nhiệm vụ hiển thị giao diện cho người dùng, dữ liệu từ server (back-end) được lấy theo giao thức RESTful API. Xây dựng giao diện người dùng sử dụng công nghệ ReactJS, một thư viện được sử dụng rộng rãi bởi các lợi ích của nó đem lại: tương thích với nhiều thư viện khác, dễ tổ chức, linh hoạt tối ưu hiệu năng cho tốc độ xử lý dữ liệu.

BackEnd chịu trách nhiệm cho việc cung cấp dịch vụ (như cơ sở dữ liệu, in ấn, truyền file, hệ thống…) cho các clients. Hệ thống sử dụng ngôn ngữ lập trình là Javascript. cùng với framework Express là một web application framework, nó cung cấp cho chúng những rất nhiều tính năng mạnh mẽ trên nền tảng web, dựa trên nền tảng NodeJS. NodeJS là một nền tảng được xây dựng trên V8 JavaScript Engine – trình thông dịch thực thi mã JavaScript, giúp xây dựng các ứng dụng web một cách đơn giản và dễ dàng mở rộng.

Hệ thống sử dụng hệ cơ sở quản trị dữ liệu phi quan hệ (NoSql) cụ thể là MongoDB. Đó là một hệ quản trị cơ sở giúp xây dựng một ứng dụng có khả năng thay đổi linh hoạt, hiệu năng cao, lưu được nhiều kiểu dữ liệu…

## Bố cục đồ án

Phần còn lại của báo cáo đồ án tốt nghiệp này được tổ chức như sau.

Chương 2 trình bày về Khảo sát và phân tích yêu cầu của bài toán được phân dựa trên các tác nhân của hệ thống, đồng thời cũng nêu lên những yêu cầu phi chức năng của hệ thống.

Trong Chương 3, sau khi đã phân tích yêu cầu bài toán từ chương 2, em bắt đầu chọn lựa những công nghệ sử dụng phù hợp với đặc tả của bài toán trên 3 yếu tố: front-end, back-end và cơ sở dữ liệu

Ở Chương 4, (chưa viết)

Trong Chương 5, (chưa viết)

Chương 6 nói về kết luận những gì đã đạt được trong học phần này và hạn chế cần phải giải quyết trong hệ thống. Từ đó, nêu lên hướng phát triển tiếp theo của hệ thống.

# ­­Khảo sát và phân tích yêu cầu

## Khảo sát hiện trạng

Trong bối cảnh toàn cầu hóa, và xu thế hội nhập, dưới sự tác động mạnh mẽ của cuộc cách mạng khoa học công nghệ, đặc biệt là cuộc cách mạng công nghiệp 4.0 đang phát triển mạnh mẽ, thì vấn đề việc làm trở thành vấn đề quan trọng, với nhiều vấn đề đặt ra. Trước kia, khi công nghệ thông tin chưa được áp dụng rộng rãi trong lĩnh vực tìm kiếm việc làm, người lao động chỉ có thể tìm việc thông qua các mẩu tin được đăng trên các mặt báo, mẩu tin được dán trên các lề đường, hay cao cấp hơn là thông qua các kênh trung gian là các công ty mua giới việc làm. Những hình thức tìm kiếm việc làm đó hiện nay đã không còn xuất hiện nhiều bởi nó khá tốn kém chi phí về mặt thời gian và công sức. Hiện nay, mọi người tìm kiếm việc làm trên các trang website được sử dụng rộng rãi và phổ biến.

Người tìm việc có thể lên trang web của chính công ty để tìm kiếm công việc phù hợp với khả năng của bản thân mình. Tuy nhiên, điều này gặp trở ngại khi người dùng mất công sức trong việc tìm đúng công ty và công việc phù hợp với khả năng của mình. Cũng có một cách khác mà mọi người hay sử dụng hiện nay đó chính là thông qua các trang mạng xã hội điểm hình là Facebook. Với đông đảo số lượng người dùng Facebook hiện nay thông qua các hội nhóm Facebook, người tim việc và nhà tuyển dụng điều có thể tìm kiếm việc làm trên đó. Tuy nhiên, việc sử dụng các trang mạng xã hội vẫn còn nhiều bất cập, hạn chế, các trang mạng xã hội này mục đích chính là để giúp người dùng giao lưu, kết nối, hệ thống đó không chú trọng vào việc xây dựng hệ thống giúp tối ưu hóa việc tìm kiếm việc làm. Chính vì vậy, nó thiếu cơ chế phân loại việc làm, tìm kiếm ứng viên, lưu công việc… dẫn đến việc người dùng phải tìm kiếm một cách khá vất vả.

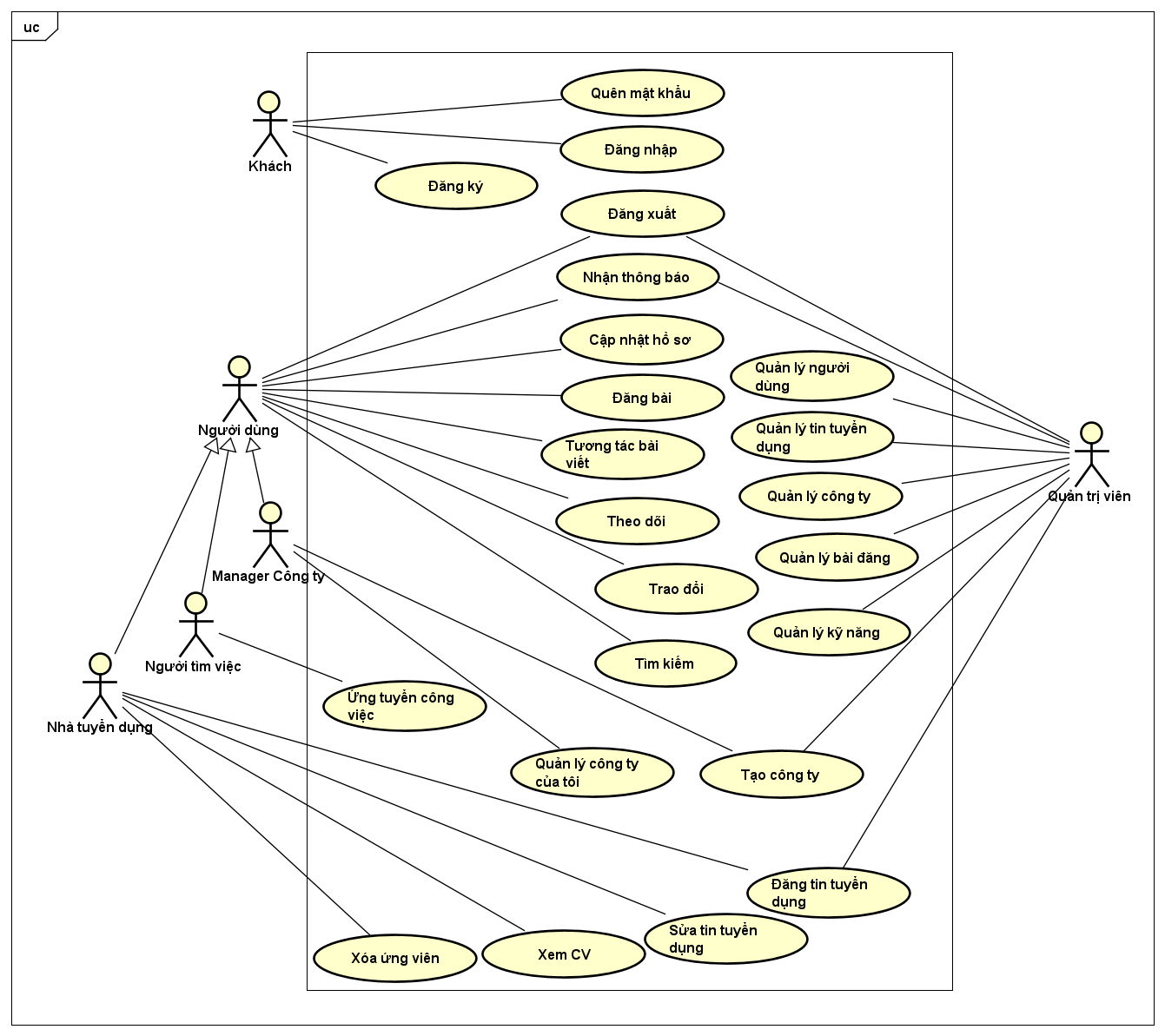
Cũng có một cách khác đó chính là vào các trang tìm kiếm việc làm như: topcv, ITViec, timviecnhanh.com… Các hệ thống website này cung cấp một cách cung cấp đầy đủ các chức năng giúp nhà tuyển dụng đăng tải các công việc của mình, cùng với đó là việc phân chia các công việc theo kỹ năng công ty một cách hợp lý giúp người lao động dễ dàng tìm kiếm các công việc mà mình mong muốn. Tuy nhiên, các hệ thống vẫn chưa cung cấp đầy đủ các thông tin hữu ích về công ty: lượt theo dõi, tổng quan về công ty, hoạt động gần đấy của công ty,… điều này gây khó khắn cho người lao động khi phải chuyển tab để tìm hiểu các thông tin về công ty nơi mình ứng tuyển công việc. Hơn nữa, đối với những hệ thống như trang website timviecnhanh.com còn hạn chế trong việc giúp cho người dùng thể hiện CV của mình. Từ đó, gây khó khăn cho nhà tuyển dụng khi tìm kiếm các ứng viên tiềm năng cho công việc của mình.

Chính vì những mặt hạn chế đó, hệ thống tìm kiếm việc làm như LinkedIn ra đời giúp cải thiện đáng kể hiệu quả tìm kiếm việc làm cho người lao động. Trong hệ thống LinkedIn, người dùng được cung cấp đa dạng các tính năng: có thể theo dõi hoạt động của các công ty mình muốn, gợi ý việc làm dựa theo kỹ năng của người dùng, đưa ra ứng viên tiềm năng cho nhà tuyển dụng,… Tuy nhiên, hệ thống vẫn chưa phân biệt rạch ròi trong việc trao đổi giữa người dùng và công việc hay xử lý các vấn đề phát sinh trong khi nhà tuyển dụng gặp sự cố.

Dựa trên những điều phân tích ở trên, ta có thể thấy, trên thị trường hiện nay có rất nhiều các kênh thông tin tìm kiếm việc làm mà ở đó người dùng có thể tìm kiếm công việc phù hợp với năng lực của bản thân mình. Tuy nhiên như đã phân tích, các hệ thống đó ít nhiều còn tồn tại các mặt hạn chế đối với người sử dụng. Từ đó, em nhận thấy việc xây dựng một hệ thống tạo môi trường tốt cho việc tương tác giữa người lao động và nhà tuyển dụng là cần thiết. Do đó, em chọn đề tài là “Xây dựng hệ thống mạng xã hội tìm kiếm việc làm” nhằm tìm cách khắc phục một số hạn chế đã nêu ra và đảm bảo các ưu điểm của các hệ thống hiện có.

## Tổng quan chức năng

### Biểu đồ use case tổng quan



**Hình 1** Sơ đồ Usecase tổng quan

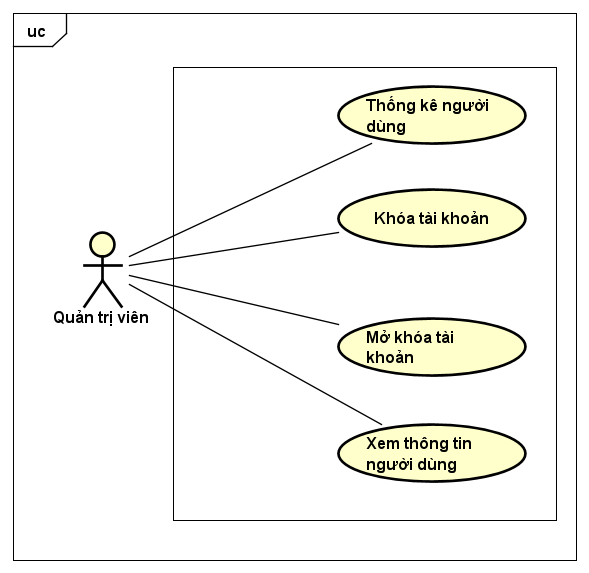
Danh sách tác nhân:

|  |  |  |
| --- | --- | --- |
| STT | Tên tác nhân | Mô tả tác nhân |
| 1 | Khách | Là người chưa đăng nhập vào hệ thống các chức năng đăng nhập, đăng ký để trở thành người dùng của hệ thống, khách cũng có chức năng quên mật khẩu giúp thay đổi mật khẩu khi học quên được gửi qua email mà họ đã đăng ký với hệ thống từ trước |
| 2 | Người dùng | Là người sử dụng hệ thống, có thể thực hiện các hoạt động như: cập nhật thông tin hồ sơ cá nhân, đăng bài, tương tác bài viết, trao đổi, theo dõi, tìm kiếm, nhận thông báo,… |
| 3 | Manager công ty | Người quản lý là người dùng của hệ thống được cung cấp thêm các chức năng như: tạo công ty, quản lý công ty của họ. |
| 4 | Người tìm việc | Người tìm việc là người dùng của hệ thống được cung cấp thêm các chức năng như: ứng tuyển công việc, trao đổi công việc,… |
| 5 | Nhà tuyển dụng | Nhà tuyển dụng là người dùng của hệ thống được cung cấp thêm các chức năng như: đăng tin tuyển dụng, xem CV của ứng viên, xóa ứng viên,… |
| 6 | Quản trị viên | Là người điều hành hệ thống được cung cấp các chức năng: nhận thông báo, quản lý người dùng, quản lý công việc, quản lý công ty, quản lý kỹ năng, quản lý bài đăng. |

Mô tả các usecase chính

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Mã Usecase | Tên Usecase | Mô tả Usecase | Tác nhân tương tác |
| 1 | UC01 | Đăng nhập | Cho phép khách có tài khoản đăng nhập trở thành người dùng của hệ thống | Khách |
| 2 | UC02 | Đăng ký | Cho phép khách đăng ký vào hệ thống | Khách |
| 3 | UC03 | Quên mật khẩu | Cho phép khách lấy lại tài khoản thông qua email mà khách đã đăng ký với hệ thống từ trước đó | Khách |
| 4 | UC04 | Đăng xuất | Đưa tác nhân thoát khỏi hệ thống | Quản trị viên, người dùng |
| 5 | UC05 | Nhận thông báo | Cho phép hệ thống thông báo những sự kiện mà tác nhân quan tâm theo thời gian thực được phân loại theo loại: theo dõi người dùng, tạo công ty,… | Quản trị viên, người dùng |
| 6 | UC06 | Quản lý người dùng | Cho phép quản trị viên có thể xem thông tin về tình trạng của người dùng, thống kê số lượng người dùng đã đăng ký vào hệ thống | Quản trị viên |
| 7 | UC07 | Quản lý bài đăng | Cho phép quản trị viên theo dõi tình trạng bài đăng lượt tương tác và loại bài đăng của người dùng. | Quản trị viên |
| 8 | UC08 | Quản lý công ty | Cho phép quản trị viên thống kê, vô hiệu các công ty có trong hệ thống | Quản trị viên |
| 9 | UC09 | Quản lý tin tuyển dụng | Cho phép quản trị viên thay đổi trạng thái của tin tuyển dụng, thống kê số lượng tin tuyển dụng, các thông tin như số lượng người ứng tuyển, số lượng người theo dõi tin tuyển dụng | Quản trị viên |
| 10 | UC10 | Quản lý kỹ năng | Cho phép quản trị viên thống kê kỹ năng được xây dựng từ người dùng trong hệ thống | Quản trị viên |
| 11 | UC11 | Cập nhật thông tin | Cho phép người dùng cập nhật thông tin hồ sơ cá nhân: kỹ năng, học vấn, thông tin tổng quan… | Người dùng |
| 12 | UC12 | Đăng bài | Cho phép người dùng đăng tải các hoat động gần đây của bản thân, công ty. | Người dùng |
| 13 | UC13 | Tương tác bài viết | Cho phép người dùng tương tác với bài đăng. | Người dùng |
| 14 | UC14 | Theo dõi | Cho phép người dùng theo dõi: người dùng khác, công ty, công việc, | Người dùng |
| 15 | UC15 | Trao đổi | Cho phép trao đổi giữa người dùng với người dùng hoặc người dùng với công việc | Người dùng |
| 16 | UC16 | Tìm kiếm | Cho phép người dùng tìm kiếm: ngươì dừng khác, công ty, công việc. | Người dùng |
| 17 | UC17 | Tạo công ty | Cho phép quản lý tạo công ty của riêng mình và phải được người quản trị viên kiểm duyệt. | Manager công ty, quản trị viên |
| 18 | UC18 | Quản lý công ty của tôi | Cho phép quản lý của công ty quản lý công ty của họ | Manager công ty |
| 19 | UC19 | Ứng tuyển công việc | Cho phép người tìm việc ứng tuyển công việc. | Người tìm việc |
| 20 | UC20 | Đăng tin tuyển dụng | Cho phép nhà tuyển dụng tạo tin tuyển dụng và phải được chấp nhận bởi quản trị viên. | Nhà tuyển dụng, quản trị viên. |
| 21 | UC21 | Xem CV | Cho phép nhà tuyển dụng xem CV của ứng viên. | Nhà tuyển dụng |
| 22 | UC22 | Xóa ứng viên | Cho phép nhà tuyển dụng xóa ứng viên không phù hợp | Nhà tuyển dụng |
| 23 | UC23 | Sửa tin tuyển dụng | Cho phép nhà tuyển dụng sửa tin tuyển dụng của mình trong khi công việc bị vô hiệu. | Nhà tuyển dụng |

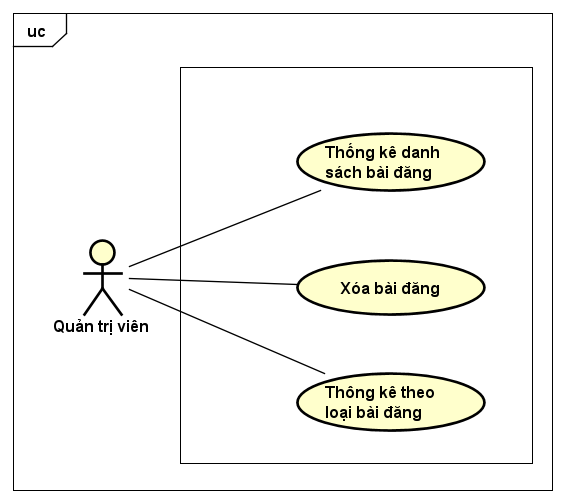
### Biểu đồ use case phân rã “Quản lý người dùng”



**Hình 2** Phân rã Usecase “Quản lý người dùng”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Mã Usecase | Tên UseCase | Mô tả Usecase | Tác nhân tương tác |
| 1 | UC06-1 | Thống kê người dùng | Thông kế những người dùng hiện có trong hệ thống | Quản trị viên |
| 2 | UC06-2 | Khóa tài khoản | Khóa tài khoản của người dùng | Quản trị viên |
| 3 | UC06-3 | Mở khóa tài khoản | Mở khóa tài khoản của người dùng | Quản trị viên |
| 4 | UC06-4 | Xem thông tin người dùng | Xem thông tin cá nhân của người dùng: hồ sơ, hoạt động gần đây của người dùng. | Quản trị viên |

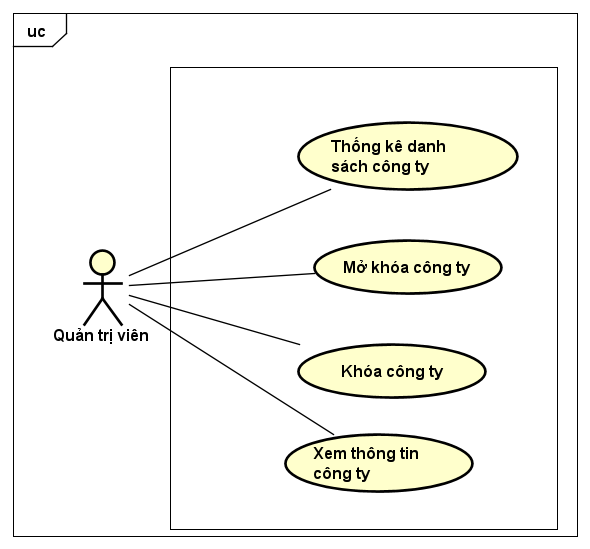
### Biểu đồ use case phân rã “Quản lý bài đăng”



**Hình 3** Phân rã Usecase “Quản lý bài đăng”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Mã Usecase | Tên UseCase | Mô tả Usecase | Tác nhân tương tác |
| 1 | UC07-1 | Thống kê danh sách bài đăng | Thống kê bài đăng hiện có: tên, số lượng ảnh,…và lọc bài đăng | Quản trị viên |
| 2 | UC07-2 | Xóa bài đăng | Xóa bài đăng, lượt thích, bình luận của bài đăng | Quản trị viên |
| 3 | UC07-3 | Thống kê theo loại bài đăng | Thống kê theo loại: công ty, cá nhân. | Quản trị viên |

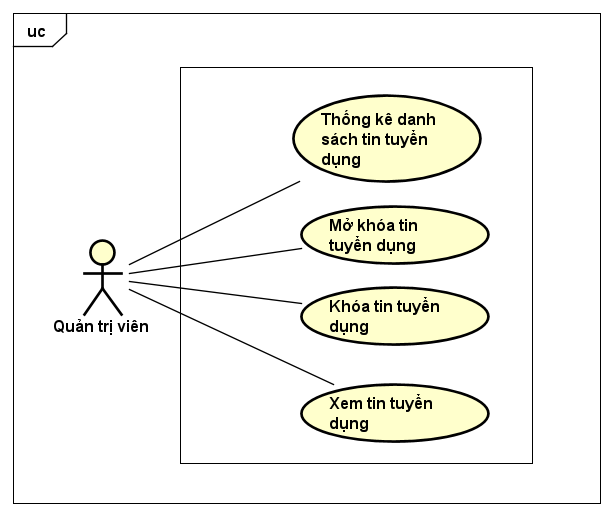
### Biểu đồ use case phân rã “Quản lý công ty”



**Hình 4** Phân rã Usecase “Quản lý công ty”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Mã Usecase | Tên UseCase | Mô tả Usecase | Tác nhân tương tác |
| 1 | UC08-1 | Thống kê danh sách công ty | Thống kê danh sách công ty có trong hệ thống | Quản trị viên |
| 2 | UC08-2 | Mở khóa công ty | Mở khóa công ty, công việc của công ty, phòng trao đổi của nhà tuyển dụng công ty. | Quản trị viên |
| 3 | UC08-3 | Khóa công ty | Khóa công ty, công việc của công ty, phòng trao đổi của nhà tuyển dụng công ty | Quản trị viên |
| 4 | UC08-4 | Xem thông tin công ty | Xem mô tả chi tiết về công ty | Quản trị viên |

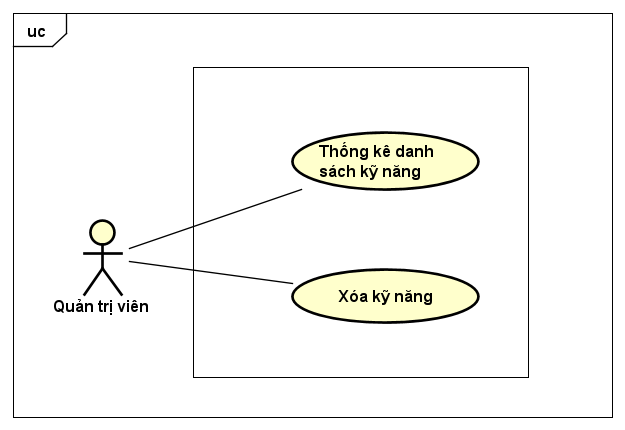
### Biểu đồ use case phân rã “Quản lý tin tuyển dụng”



**Hình 5** Phân rã Usecase “Quản lý tin tuyển dụng”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Mã Usecase | Tên UseCase | Mô tả Usecase | Tác nhân tương tác |
| 1 | UC09-1 | Thống kê danh sách tin tuyển dụng | Thống kê danh sách tin tuyển dụng có trong hệ thống: theo tên, số lượng người theo dõi,… | Quản trị viên |
| 2 | UC09-2 | Mở khóa tin tuyển dụng | Mở khóa tin tuyển dụng cho người tìm việc ứng tuyển | Quản trị viên |
| 3 | UC09-3 | Khóa tin tuyển dụng | Khóa tin tuyển dụng không cho người tìm việc ứng tuyển | Quản trị viên |

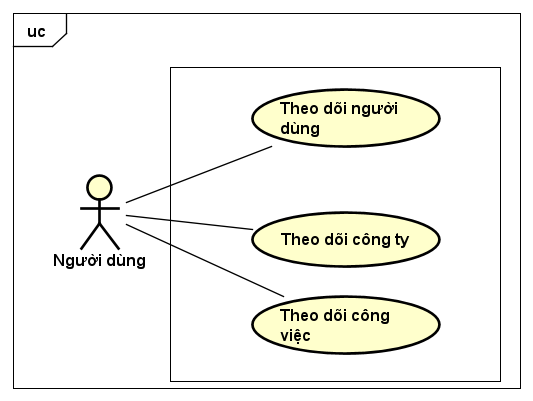
### Biểu đồ use case phân rã “Quản lý kỹ năng”



**Hình 6** Phân rã Usecase “Quản lý kỹ năng”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Mã Usecase | Tên UseCase | Mô tả Usecase | Tác nhân tương tác |
| 1 | UC10-1 | Thống kê danh sách kỹ năng | Thống kê danh sách kỹ năng có trong hệ thống: tên, số lượng công việc, số lượng người dùng,… | Quản trị viên |
| 2 | UC10-2 | Xóa kỹ năng | Xóa kỹ năng của người dùng, công việc liên quan | Quản trị viên |

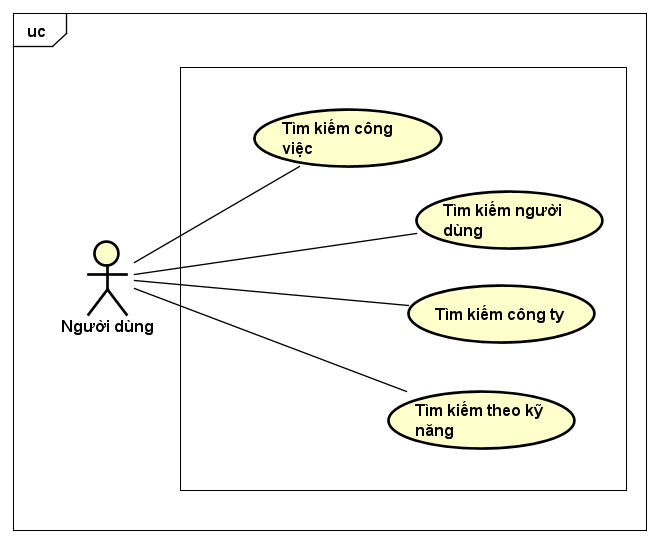
### Biểu đồ use case phân rã “Theo dõi”



**Hình 7** Phân rã Usecase “Theo dõi”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Mã Usecase | Tên UseCase | Mô tả Usecase | Tác nhân tương tác |
| 1 | UC14-1 | :Theo dõi người dùng | Theo dõi người dùng để nhận thông tin mới nhất của họ | Người dùng |
| 2 | UC14-2 | Theo dõi công ty | Theo dõi công ty để nhận thông tin mới nhất của họ | Người dùng |
| 3 | UC14-3 | Theo dõi công việc | Theo dõi công việc để nhận thông tin, lưu công việc. | Người dùng |

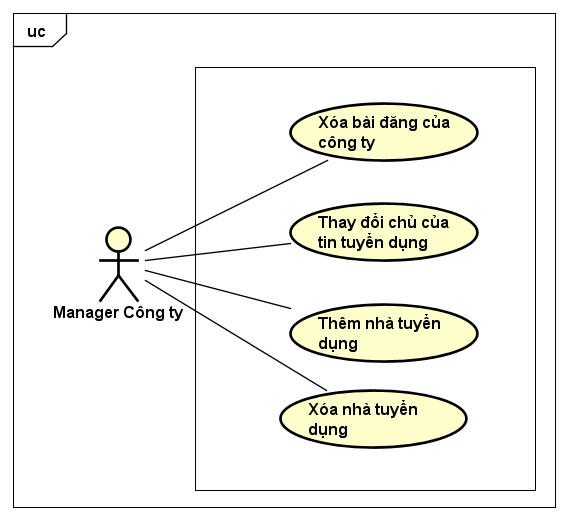
### Biểu đồ use case phân rã “Tìm kiếm”



**Hình 8** Phân rã Usecase “Tìm kiếm”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Mã Usecase | Tên UseCase | Mô tả Usecase | Tác nhân tương tác |
| 1 | UC16-1 | Tìm kiếm công việc | Tìm kiếm công việc theo từ khóa hoặc theo đề xuất dựa trên hồ sơ người dùng | Người dùng |
| 2 | UC16-2 | Tìm kiếm người dùng | Tìm kiếm người dùng dựa theo từ khóa hoặc dựa trên hồ sơ của người dùng | Người dùng |
| 3 | UC16-3 | Tìm kiếm công ty | Tìm kiếm công ty dựa theo từ khóa hoặc dựa trên hồ sơ của người dùng | Người dùng |
| 4 | UC16-4 | Tìm kiếm theo kỹ năng | Tìm kiếm theo kỹ năng các người dùng, công ty, công việc phù hợp với kỹ năng | Người dùng |

### Biểu đồ use case phân rã “Quản lý công ty của tôi”



**Hình 9** Phân rã Usecase “Quản lý công ty của tôi”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Mã Usecase | Tên UseCase | Mô tả Usecase | Tác nhân tương tác |
| 1 | UC18-1 | Xóa bài đăng của công ty | Xóa bài đăng không phù hợp hay sai nội dung định hướng của công ty | Manager công ty |
| 2 | UC18-2 | Thay đổi chủ của tin tuyển dụng | Thay đổi quyền sở hữu của tin tuyển dụng thông báo đến các bên liên quan. | Manager công ty |
| 3 | UC18-3 | Thêm nhà tuyển dụng | Thêm nhà tuyển dụng cho công ty | Manager công ty |
| 4 | UC18-4 | Xóa nhà tuyển dụng | Xóa nhà tuyển dụng ra khỏi công ty lúc này họ sẽ trở thành người tìm việc | Manager công ty |

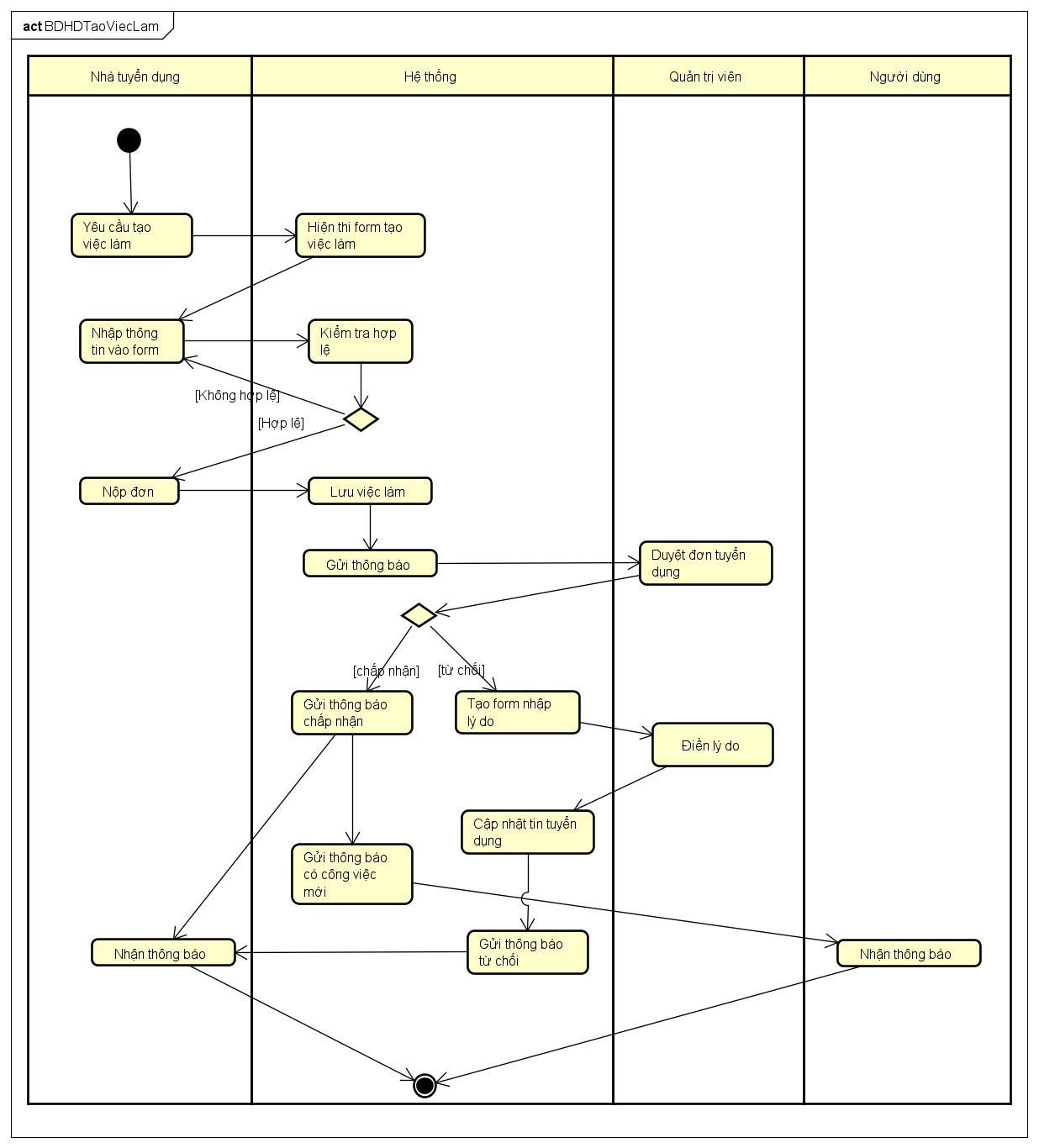
### Quy trình nghiệp vụ

1. Quy trình nghiệp vụ đăng tin tuyển dụng

Quy trình đăng tin tuyển dụng bao gồm các nhóm người dùng tham gia là nhà tuyển dụng, quản trị viên, và người dùng để đảm bảo dữ liệu việc làm được kiểm soát một cách chặt chẽ, chính xác và để cho mọi người có thể nắm bắt được thông tin tuyển dụng một cách nhanh nhất đầy đủ và chính xác nhất.

|  |  |
| --- | --- |
|  | Đăng tin tuyển dụng |
| **Mục tiêu quy trình** | Đăng tin tuyển dụng việc làm lên hệ thống |
| **Phạm vi quy trình** | Nhà tuyển dụng, quản trị viên, người dùng |
| **Đầu vào quy trình** | * Thông tin tuyển dụng * Thông tin người đăng tin tuyển dụng |
| **Luồng quy trình** | * Nhà tuyển dụng chọn chức năng đăng tin tuyển dụng * Hệ thống hiển thị giao diện đăng tin tuyển dụng * Nhà tuyển dụng nhập các thông tin cần thiết cho việc làm * Hệ thống xác thực đầy đủ thông tin, nếu thiếu, hiện thị lỗi và yêu cầu người dùng sửa lại * Việc làm được đăng lên hệ thống với trạng thái bị vô hiệu hóa, hệ thống sẽ gửi thông báo đến quản trị viên để yêu cầu xác nhận tin tuyển dụng. * Quản trị viên xác nhận hoặc từ chối tin tuyển dụng sau khi kiểm tra tính đúng đắn của tin * Trong trường hợp chấp nhận tin tuyển dụng, hệ thống sẽ gửi thông báo đến nhà tuyển dụng thông báo tin tuyển dụng đã được xác nhận đồng thời gửi đến các người dùng đăng theo dõi công ty có tin tuyển dụng đó. Trong trường hợp thất bại , quản trị viên viết lý do, sau đó, hệ thống gửi thông báo từ chối đến nhà tuyển dụng |
| **Đầu ra quy trình** | Gửi thông báo xác nhận hoặc từ chối tin tuyển dụng |
| **Quy trình đặc biệt** | Không có |

**Bảng 1** Quy trình đăng tin tuyển dụng



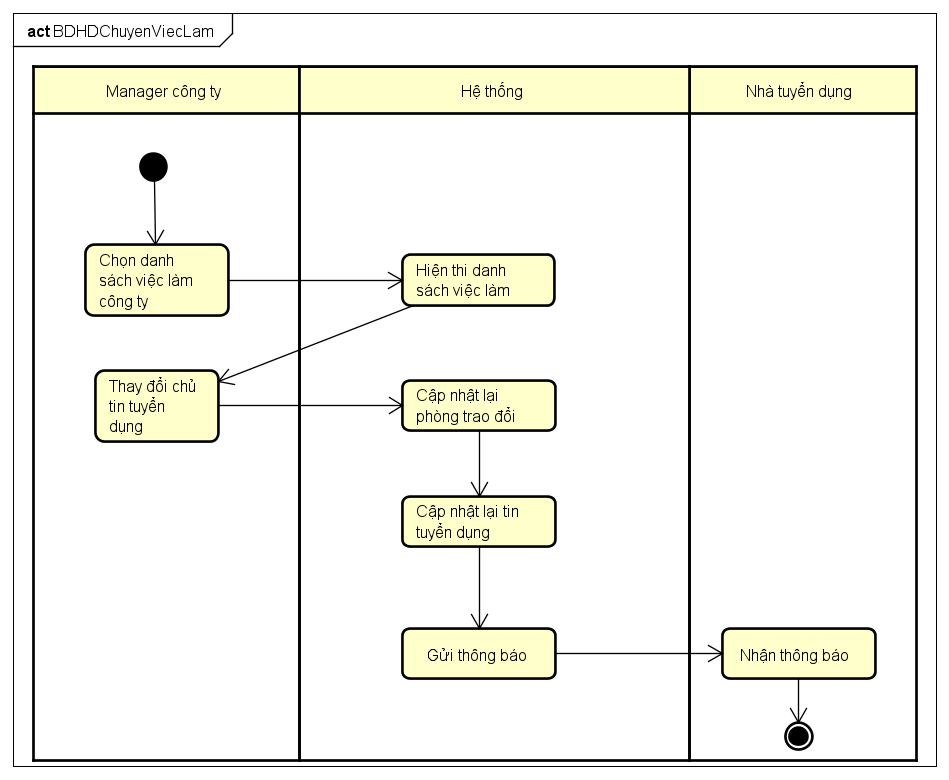
**Hình 10** Biểu đồ hoạt động đăng tin tuyển dụng

1. Quy trình nghiệp vụ “Thay đổi chủ của tin tuyển dụng”

Quy trình thay đổi chủ của tin tuyển dụng bao gồm các nhóm người dùng tham gia là nhà tuyển dụng, manager công ty, và người dùng giúp cho công ty ứng phó được với các tình huống không may khi nhà tuyển dụng của công ty gặp sự cố.

|  |  |
| --- | --- |
|  | Thay đổi chủ của tin tuyển dụng |
| **Mục tiêu quy trình** | Chuyển công việc từ nhà tuyển dụng này đến nhà tuyển dụng khác thuộc cùng một công ty do manager công ty thực hiện |
| **Phạm vi quy trình** | Nhà tuyển dụng, manager công ty |
| **Đầu vào quy trình** | * Tin tuyển dụng * Nhà tuyển dụng cũ * Nhà tuyển dụng mới |
| **Luồng quy trình** | * Manager công ty chọn xem danh sách công việc của công ty * Hiện thị danh sách việc làm của công ty * Manager công ty thực hiện chuyển giao công việc * Hệ thống cập nhật lại phòng trao đổi của công việc đó * Hệ thống cập nhật lại thông tin chủ của tin tuyển dụng * Hệ thống gửi thông báo đến câc nhà tuyển dụng liên quan |
| **Đầu ra quy trình** | Gửi thông báo thay đổi nhà tuyển dụng thành công |
| **Quy trình đặc biệt** | Không có |

**Bảng 2** Quy trình thay đổi chủ của tin tuyển dụng



**Hình 11** Biều đồ hoạt động thay đổi chủ của tin tuyển dụng

## Đặc tả chức năng

### Đặc tả use case “Quên mật khẩu”

|  |  |  |  |
| --- | --- | --- | --- |
| Mã Use case | UC03 | Tên Use case | Quên mật khẩu |
| Tác nhân | Khách | | |
| Tiền điều kiện | Đã vào trang quên mật khẩu | | |
| Luồng sự kiện chính  (Thành công) | |  |  |  | | --- | --- | --- | | STT | Thực hiện bởi | Hành động | | 1 | Khách | Nhập email cần lấy mật khẩu (\*) | | 2 | Khách | Gửi yêu cầu lấy mật khẩu | | 3 | Hệ thống | Kiểm tra thông tin có đã nhập đúng trường dữ liệu hay chưa | | 4 | Hệ thống | Kiểm tra tính hợp lệ | | 5 | Hệ thống | Gửi email reset mật khẩu | | 6 | Khách | Click đường đãn được gửi qua email | | 7 | Hệ thống | Hiển thị form đặt lại mật khẩu | | 8 | Khách | Điền biểu mẫu đặt lại mật khẩu (\*\*) | | 9 | Khách | Nộp biều mẫu | | 10 | Hệ thống | Kiểm tra thông tin có đã nhập đúng trường dữ liệu hay chưa | | 11 | Hệ thống | Kiểm tra tính hợp lệ | | 12 | Hệ thống | Thực hiện đặt lại mật khẩu cho khách, xóa token | | 13 | Hệ thống | Thông báo đặt lại mật khẩu thành công | | | |
| Luồng sự kiện thay thế | |  |  |  | | --- | --- | --- | | STT | Thực hiện bởi | Hành động | | 3a | Hệ thống | Thông báo lỗi: cần nhập đầy đủ cá trường thông tin bắt buộc | | 4a | Hệ thống | Thông báo lỗi email không tồn tại | | 5a | Hệ thống | Thông báo lỗi khi gửi email | | 10a | Hệ thống | Thông báo nhập thiếu hoặc sai trường dữ liệu | | 11a | Hệ thống | Báo token hết hạn | | | |
| Hậu điều kiện | Không có | | |

**Bảng 3** Đặc tả usecase “Quên mật khẩu”

Dữ liệu đầu vào (\*):

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Trường dữ liệu | Mô tả | Bắt buộc | Điều kiện hợp lệ | Ví dụ |
| 1 | Email | Email của tài khoản | Có | Định dạng email | [alice@gmai.com](mailto:alice@gmai.com) |

Dữ liệu đầu vào (\*\*)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Trường dữ liệu | Mô tả | Bắt buộc | Điều kiện hợp lệ | Ví dụ |
| 1 | Password | Mật khẩu đặt lại | Có | Từ 6 đến 20 ký tự | [aclie12312](mailto:alice@gmai.com) |
| 2 | Confirmpassword | Kiểm tra lại mật khẩu | Có | Đúng với trường password | [aclie12312](mailto:alice@gmai.com) |

### Đặc tả use case “Theo dõi người dùng”

|  |  |  |  |
| --- | --- | --- | --- |
| Mã Use case | UC14-1 | Tên Use case | Theo dõi người dùng |
| Tác nhân | Người dùng | | |
| Tiền điều kiện | Hiện ở trang profile của người dùng muốn theo dõi | | |
| Luồng sự kiện chính  (Thành công) | |  |  |  |  | | --- | --- | --- | --- | | STT | Thực hiện bởi | Hành động | | | 1. | Ngưởi dùng | | Click nút “Theo dõi” người dùng | | 2. | Hệ thống | | Kiểm tra người dùng hợp lệ | | 3. | Hệ thống | | Thông báo theo dõi người dùng thành công | | 4. | Hệ thống | | Gửi thông báo đến người dùng được theo dõi | | | |
| Luồng sự kiện thay thế | |  |  |  | | --- | --- | --- | | STT | Thực hiện bởi | Hành động | | 2a. | Hệ thống | Người dùng hiện đang bị khóa | | 3a. | Hệ thống | Bỏ theo dõi người dùng | | 3b | Hệ thống | Đã theo dõi người dùng thì chuyển đến phòng trao đổi | | | |
| Hậu điều kiện | Cập nhật lại Feed của người theo dõi | | |

**Bảng 4** Đặc tả use case “Theo dõi người dùng”

### Đặc tả use case “Cập nhật thông tin”

|  |  |  |  |
| --- | --- | --- | --- |
| Mã Use case | UC11 | Tên Use case | Cập nhật thông tin |
| Tác nhân | Người dùng | | |
| Tiền điều kiện | Đang ở trong profile của bản thân | | |
| Luồng sự kiện chính  (Thành công) | |  |  |  |  | | --- | --- | --- | --- | | STT | Thực hiện bởi | Hành động | | | 1. | Người dùng | | Chọn mục cần cập nhật | | 2. | Hệ thống | | Hiện thị giao diện cập nhật thông tin | | 4. | Người dùng | | Nhập thông tin cần cập nhật (\*) | | 5. | Người dùng | | Nhấn nút “Nộp” | | 6. | Hệ thống | | Kiểm tra tính hợp lệ | | . | Hệ thống | | Cập nhật lại trang profile của người dùng | | | |
| Luồng sự kiện thay thế | |  |  |  | | --- | --- | --- | | STT | Thực hiện bởi | Hành động | | 6a. | Hệ thống | thông báo lỗi: Hiển thị thông báo các trường thông tin còn thiếu hoặc không hợp lệ | | | |
| Hậu điều kiện | Không có | | |

**Bảng 5** Đặc tả use case “Cập nhật thông tin”

Dữ liệu đầu vào (\*\*)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Trường dữ liệu | Mô tả | Bắt buộc | Điều kiện hợp lệ | Ví dụ |
| 1 | name | Tên người dùng được hiển trị trên hệ thống | Có | Phải là duy nhất trong hệ thống | [phongvu](mailto:alice@gmai.com) |
| 2 | email | Email của người dùng | Có | Phải là duy nhất trong hệ thống | [alice@gmail.com](mailto:alice@gmail.com) |
| 3 | avatar | ảnh đại diện của người dùng | Không | Chuỗi string | http://localhost:3000/ abc.png |
| 4 | about | Giới thiệu thông quan về bản thân mình | Không | Chuỗi tring ít hơn 1000 ký tự | Hello 123 |
| 5 | educations | Trình độ văn hóa mà người dùng đạt được | Không | Là một mảng các trường đào tạo của người dùng | [{school: “HUST”, degree: “Cu Nhan”,  field\_of\_style: “IT”,  start\_date: 08/2018,  end\_date: 08/2022}] |
| 6 | skills | Kỹ năng mà người dùng có | Không | Một mảng các kỹ năng | [ “C#”, “Java”] |
| 7 | experience | Kinh nhiệm của người dùng | Không | Chuỗi string | Tôi có 5 năm kinh nghiệp về C# |
| 8 | languages | Ngoại ngữ của người dùng | Không | Chuỗi string | English, Viẹtnamese |
| 9 | social | Đường liên kết đến các trang website của người dùng | Không | Object | {youtube: <http://localhost>, facebook: <http://localhost>,} |

### Đặc tả usecase “Đăng bài”

|  |  |  |  |
| --- | --- | --- | --- |
| Mã Use case | UC12 | Tên Use case | Đăng bài |
| Tác nhân | Người dùng | | |
| Tiền điều kiện | Đang thao tác với hệ thống | | |
| Luồng sự kiện chính  (Thành công) | |  |  |  |  | | --- | --- | --- | --- | | STT | Thực hiện bởi | Hành động | | | 1. | Người dùng | | Nhấn nút đăng bài | | 2 | Hệ thống | | Hiển thị giao diện đăng bài | | 3. | Người dùng | | Điền đầy đủ các trường thông tin (\*) | | 4. | Người dùng | | Nhấn nút “Nộp” | | 5. | Hệ thống | | Kiểm tra xem các trường thông tin nhập vào đã hợp lệ hay chưa | | 6. | Hệ thống | | Lưu bài đăng vào hệ thống | | 7. | Hệ thống | | Hiển thị thông báo tạo bài đăng thành công | | 8 | Hệ thống | | Gửi thông báo đến các người dùng quan tâm | | | |
| Luồng sự kiện thay thế | |  |  |  | | --- | --- | --- | | STT | Thực hiện bởi | Hành động | | 5a. | Hệ thống | thông báo lỗi: Hiển thị thông báo các trường thông tin còn thiếu hoặc không hợp lệ | | 6a. | Hệ thống | thông báo lỗi: Lỗi upload ảnh khi bài đăng có file đính kèm | | 6b | Hệ thống | thông báo lỗi: Lỗi tạo bài đăng khi không có đủ quyền dể tạo đối với bài đăng công ty | | | |
| Hậu điều kiện | Cập nhật lại danh sách bài đăng | | |

**Bảng 6** Đặc tả use case “Đăng bài”

Dữ liệu đầu vào (\*):

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Trường dữ liệu | Mô tả | Bắt buộc | Điều kiện hợp lệ | Ví dụ |
| 1 | content | Nội dung của bài viết | Có | Chuỗi string | Viết gì đó |
| 2 | images | Hình ảnh minh hoạt của bài viết | Không | Đường dẫn đến ảnh | <http://abc.com> |
| 3 | videos | Video của bài viết | Không | Chứa 1 file video | http://abc.com |

### Đặc tả use case “Trao đổi”

|  |  |  |  |
| --- | --- | --- | --- |
| Mã Use case | UC15 | Tên Use case | Trao đổi |
| Tác nhân | Người dùng | | |
| Tiền điều kiện | Đã ở trang profile của người dùng cần trao đổi | | |
| Luồng sự kiện chính  (Thành công) | |  |  |  | | --- | --- | --- | | STT | Thực hiện bởi | Hành động | | 1. | Người dùng | Nhấn nút “trao đổi” | | 2. | Hệ thống | Kiểm tra người dùng hợp lệ | | 3. | Hệ thống | Gửi thông báo kết nối đến người được trao đổi | | 4 | Hệ thống | Thông báo chờ người dùng khác duyệt trao đổi | | 5 | Hệ thống | Thông báo chấp nhận hoặc từ chối trao đổi | | | |
| Luồng sự kiện thay thế | |  |  |  | | --- | --- | --- | | STT | Thực hiện bởi | Hành động | | 2a. | Hệ thống | Báo không tìm thấy người dùng | | 3a | Hệ thống | Chuyển hướng đến phòng đã được mở giữa hai người dùng. | | | |
| Hậu điều kiện | Không | | |

**Bảng 7** Đặc tả use case “Trao đổi”

### Đặc tả use case “Ứng tuyển công việc”

|  |  |  |  |
| --- | --- | --- | --- |
| Mã Use case | UC19 | Tên Use case | Ứng tuyển công việc |
| Tác nhân | Người tìm việc | | |
| Tiền điều kiện | Đang ở trang xem chi tiết công việc hoặc tìm kiếm, đề xuất công việc | | |
| Luồng sự kiện chính  (Thành công) | |  |  |  | | --- | --- | --- | | STT | Thực hiện bởi | Hành động | | 1. | Người tìm việc | Nhấn nút ứng tuyển công việc | | 2. | Hệ thống | Hiển thị giao diện ứng tuyển công việc | | 3. | Người tìm việc | Nhập các trường theo yêu cầu (\*) | | 4. | Người tìm việc | Nhấn nút nộp | | 5. | Hệ thống | Kiểm tra tính hợp lệ các trường đã điền | | 6 | Hệ thống | Kiểm tra tính hợp lệ của đơn ứng tuyển | | 7 | Hệ thống | Thông báo ứng tuyển thành công | | 8 | Hệ thống | Gửi thông báo đến nhà tuyển dụng chịu trách nhiệm cho công việc | | | |
| Luồng sự kiện thay thế | |  |  |  |  | | --- | --- | --- | --- | | STT | Thực hiện bởi | | Hành động | | 5a. | Hệ thống | thông báo lỗi: nhập thiếu hoặc sai định dạng các trường | | | 6a | Hệ thống | Thông báo lỗi: quá hạn ứng tuyển hoặc đã ứng tuyển rồi | | | 7a | Hệ thống | Lỗi upload file ứng tuyển | | | | |
| Hậu điều kiện | Cập nhật danh sách đã ứng tuyển | | |

**Bảng 8** Đặc tả use case “Ứng tuyển công việc”

Dữ liệu đầu vào (\*):

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Trường dữ liệu | Mô tả | Bắt buộc | Điều kiện hợp lệ | Ví dụ |
| 1 | email | Email hiện tại người tìm việc đang sử dụng | Có | Định dạng email | [Alize@gmail.com](mailto:Alize@gmail.com) |
| 2 | phone | Số điên thoại hiện tại người tìm việc sử dụng | Có | Định dạng 10 chữ số | [1234567890](http://abc.com) |
| 3 | CV | Đơn ứng tuyển của ứng viên | Có | Đường dẫn | http://abc.com |

### Đặc tả use case “Đăng tin tuyển dụng”

|  |  |  |  |
| --- | --- | --- | --- |
| Mã Use case | UC20 | Tên Use case | Đăng tin tuyển dụng |
| Tác nhân | Nhà tuyển dụng | | |
| Tiền điều kiện | Đang ở trang chủ của hệ thống | | |
| Luồng sự kiện chính  (Thành công) | |  |  |  | | --- | --- | --- | | STT | Thực hiện bởi | Hành động | | 1. | Nhà tuyển dụng | Nhấn nút tạo “Việc làm” | | 2. | Hệ thống | Hiển thị giao diện tạo việc làm | | 3. | Nhà tuyển dụng | Nhập các trường theo yêu cầu (\*) | | 4. | Nhà tuyển dụng | Nhấn nút nộp | | 5. | Hệ thống | Kiểm tra tính hợp lệ các trường đã điền | | 6 | Hệ thống | Kiểm tra tính hợp lệ của tin tuyển dụng | | 7 | Hệ thống | Thông báo chờ quản trị viên duyệt | | 8 | Hệ thống | Gửi thông báo đến quản trị viên có việc là vừa tạo | | | |
| Luồng sự kiện thay thế | |  |  |  |  | | --- | --- | --- | --- | | STT | Thực hiện bởi | | Hành động | | 5a. | Hệ thống | thông báo lỗi: nhập thiếu hoặc sai định dạng các trường | | | 6a | Hệ thống | Thông báo lỗi: sai hạn chót, công ty bị khóa,… | | | | |
| Hậu điều kiện | Không | | |

**Bảng 9** Đặc tả use case “Đăng tin tuyển dụng”

Dữ liệu đầu vào (\*):

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Trường dữ liệu | Mô tả | Bắt buộc | Điều kiện hợp lệ | Ví dụ |
| 1 | title | Nhan đề của tin tuyển dụng | Có | Chuỗi từ 3 đến 200 ký tự | [lạlafjl](mailto:Alize@gmail.com) |
| 2 | salary | Lương của công việc | Có | Định dạng số | [2000000](http://abc.com) |
| 3 | job\_type | Kiểu công việc | Có | Một trong 3 giá trị:fulltime, parttime, freelance | fulltime |
| 4 | numbers | Số lượng tuyển | Có | Chữ số | 10 |
| 5 | responsibilities | Mô tả trách nhiệm công việc | Có | Ít nhất 3 ký tự | Thực hiện phân tích nghiệp vụ các hệ thống, vẽ biểu đồ |
| 6 | requirements | Yêu cầu của công việc | Có | Ít nhất 10 ký tự | Biết một ngôn ngữ lập trình |
| 7 | location | Địa điểm làm việc | Có | Chuỗi | Hà nội |
| 8 | experience | Kinh nghiệp đã trải qua | Có | Chuỗi | 1-3 năm kinh nghiệp |
| 9 | rank | Chức vụ | Có | Chuỗi | Intern |
| 10 | other | Thông tin khác | Không | Chuỗi | Yêu cầu biết chơi đàn |
| 11 | deadline | Hạn chót ứng tuyển | Có | Ngày | 10/11/2022 |
| 12 | categories | Kỹ năng của công việc | Có | Mảng kỹ năng | [Java, C#] |

## Yêu cầu phi chức năng

### Yêu cầu bảo mật

Hệ thống phải được vận hành một cách chính xác, đáng tin cậy. Các tác nhân trong hệ thống được phân quyền rõ ràng đảm bảo mỗi đối tượng chỉ được thực hiện phần việc mà mình có quyền hạn. Hệ thống có khả năng ngăn chặn phòng ngừa tấn công từ bên ngoài để bảo đảm dữ liệu người dùng. Mật khẩu lưu trong cơ sở dữ liệu phải được mã hóa, các thông tin cá nhân phải được đảm bảo tính riêng tư.

### Yêu cầu vận hành

Hệ thống được xây dựng dựa trên nền tảng ứng dụng website, chính vì vậy, hệ thống cần đẩm bảo chạy tốt trên bất cứ ứng dụng website nào (Google Chorm, Cốc Cốc,…). Hệ thống cần phải hoạt động ổn định 24/7, hoạt động 500 giờ liên tục không lỗi. Hệ thống có thể hoạt động trở lại bình thường trong vòng 2 giờ sau khi xảy ra lỗi. Thời gian đáp ứng tối đa của hệ thống (nếu không được nêu rõ) là 2 giây lúc bình thường hoặc 5 giây lúc cao điểm. Hệ thống hỗ trợ 1000 người truy xuất vào cơ sở dữ liệu cùng lúc.

### Yêu cầu giao diện

Hệ thống có giao diện đồ họa, đảm bảo dễ sử dụng, thân thiện với sử dụng. Bố cục hệ thống cần được phân chia rõ ràng, giúp người sử dụng hiểu và thao tác dễ dàng. Thiết kế giao diện cần có độ tương phản giữa màu nền và màu chữ, nội dung bảng biều, thanh điều hướng chú trọng sử dụng các hình vẽ và những từ ngữ cô đọc, dễ hiều, giúp người sử dụng nắm bắt nhanh nhất ý nghĩa. Ngôn ngữ trong hệ thống được kiểm soát, xét duyệt bởi quản trị viên đảm bảo tính lịch sự, văn phong rõ ràng, không chứa những từ ngữ phản cảm, gây kích động.

# Công nghệ sử dụng

## Ngôn ngữ lập trình Javascript

Javascript viết tắt là JS được bắt đầu phát triền từ năn 1995 tại Netscape Communications được phát triển bởi Brendan Eich. JavaScript là một trình thông dịch ngôn ngữ lập trình chính thức được nhúng bên trong trình duyệt web, làm cho website của bạn trở lên sinh động, lôi cuốn người xem. Javascript được hoạt động khi HTML và CSS phân tích hoàn tất. Khi điều này được thực hiện, JavaScript được thực thi theo thứ tự chúng được tìm thấy trên trang web: các biến và hàm được xác định, các lời gọi hàm được thực thi, trình xử lý sự kiện được kích hoạt,... Các hoạt động này dẫn đến việc DOM được cập nhật bởi JavaScript và được hiển thị ngay lập tức bởi trình duyệt.

Tuy nhiên vào năm 2009, với sự xuất hiện của Node.js, Javascript giờ đây không những có thể biên dịch và chạy ở browser, mà nó còn có thể chạy được ở phía server-side thông qua trình biên dịch là Node.js.

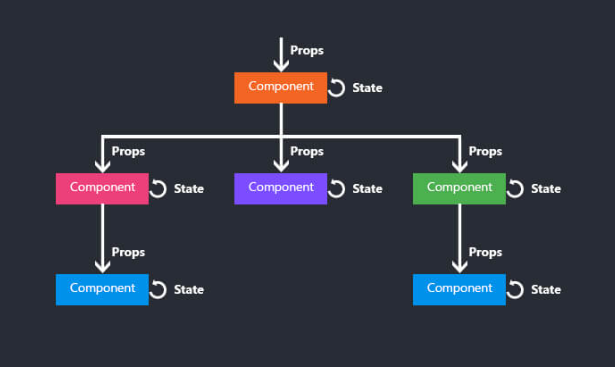
Cho đến ngày nay, JS đã trở thành ngôn ngữ phổ biến, có một cộng đồng phát triển to lớn nhờ các ưu điểm của nó đem lại.JS ở phía client giúp chung ta xử lý nhiều việc như: validation, filter và pagination… Từ đó, phía server sẽ tiết kiệm được lượng lớn lưu lượng khi client giao tiếp với server. JS được đánh giá là ngôn ngữ dữ học nhất hiện nay. JS hoạt động được trên nhiều trình duyệt, nền tảng, mã JS nhanh hơn và nhẹ hơn các ngôn ngữ khác.

Tuy nhiên, ngôn ngữ nào cũng có những ưu điểm và hạn chế của nó, JS không ngoại lệ. JS dễ bị khai thác, có thể được dùng để thực thi mã độc trên máy tính của người dùng.

## Công nghệ lập trình giao diện ReactJS

React (hay ReactJS, React.js) là một thư viện Javascript mã nguồn mở để xây dựng các thành phần giao diện có thể tái sử dụng. Nó khuyến khích việc tạo ra các thành phần giao diện người dùng có thể tái sử dụng, hiển thị dữ liệu thay đổi theo thời gian. Rất nhiều người sử dụng React làm V trong MVC. React trừu tượng hóa DOM khỏi bạn, cung cấp mô hình lập trình đơn giản hơn và hiệu suất tốt hơn. React cũng có thể hiển thị trên máy chủ bằng Node và nó có thể cung cấp năng lượng cho các ứng dụng gốc bằng React Native. React triển khai luồng dữ liệu phản ứng một chiều, điều này giúp giảm bớt phần soạn sẵn và dễ lý luận hơn so với ràng buộc dữ liệu truyền thống.

React được xây dựng xung quanh các component, chứ không dùng template như các framework khác. Trong React, chúng ta xây dựng trang web sử dụng những thành phần (component) nhỏ. Chúng ta có thể tái sử dụng một component ở nhiều nơi, với các trạng thái hoặc các thuộc tính khác nhau, trong một component lại có thể chứa thành phần khác. Mỗi component trong React có một trạng thái riêng, có thể thay đổi, và React sẽ thực hiện cập nhật component dựa trên những thay đổi của trạng thái. Mọi thứ React đều là component. Chúng giúp bảo trì mã code khi làm việc với các dự án lớn. Một react component đơn giản chỉ cần một method render. Có rất nhiều methods khả dụng khác, nhưng render là method chủ đạo. Có hai khái niệm cơ bản trong React là: state và props.



**Hình 12** Cấu trúc, thành phần của ReactJS

Props: giúp các component tương tác với nhau, component nhận input gọi là props, và trả thuộc tính mô tả những gì component con sẽ render. Prop là bất biến. State: thể hiện trạng thái của ứng dụng, khi state thay đồi thì component đồng thời render lại để cập nhật UI. Về cơ bản, việc xây dựng một ứng dụng MVC phía client với giàng buộc dữ liệu 2 chiều (2 way data-binding) là khá đơn giản.

Tuy nhiên nếu dự án ngày càng mở rộng, nhiều tính năng hơn, làm cho việc bảo trì dự án gặp khó khăn, đồng thời hiệu năng cũng bị giảm. Bạn cũng có thể giải quyết vấn đề đó bằng các thư viện khác như Backbone.js hay Angular.js, tuy nhiên bạn sẽ thấy các hạn chế của chúng khi dự án của bạn ngày càng lớn. React ra đời sau AngularJS, nó sinh ra để dành cho các ứng dụng lớn dễ dàng quản lý và mở rộng. Mục tiêu chính của React là nhanh, đơn giản, hiệu năng cao và dễ dàng mở rộng.

Trong hệ thống này, React sẽ giúp phân tách các thành phần giao diện để dễ quản lý tổ chức và tái sử dụng. Cùng với đó, là một bộ các thư viện bổ trợ giúp xây dựng hệ thống tối ưu trải nghiệm người dùng.

Đầu tiên, thư viện Redux (cụ thể là Redux toolkit) để quản lý các state trong hệ thống. Như đã đề cập ở trên, React sử dụng props để chia sẻ state giữa các component với nhau là khá phức tạp và rắc rối. Redux sẽ giải quyết một bài toán khá là quan trọng đó là chia sẻ state. Cách Redux hoạt động rất đơn giản. Có một "store" trung tâm chứa toàn bộ trạng thái của ứng dụng. Mỗi thành phần có thể truy cập trạng thái được lưu trữ mà không phải gửi từ thành phần này sang thành phần khác. Có ba phần xây dựng: actions, store, and reducers. Actions nói một cách đơn giản, action là sự kiện. Chúng là cách duy nhất bạn có thể gửi dữ liệu từ ứng dụng của mình đến "store" Redux. Reducers là các hàm thuần túy lấy trạng thái hiện tại của ứng dụng, thực hiện một hành động và trả về trạng thái mới. Store giữ trạng thái ứng dụng. Chỉ có một "store" trong bất kỳ ứng dụng Redux nào.

Thứ hai, hệ thống sử dụng thư viện React-query giúp dễ dàng tìm nạp, lưu vào bộ nhớ đệm, đồng bộ hóa và cập nhật trạng thái máy chủ. React-query giúp chúng ta tạo ra các hook giúp chúng ta tránh lặp code, dễ bảo trì, tăng trải nhiệm người sử dụng và tiết kiệm băng thông khi giao tiếp với máy chủ.

## Công nghệ lập trình máy chủ NodeJS

NodeJS là một nền tảng được xây dựng trên V8 JavaScript Engine – trình thông dịch thực thi mã JavaScript, giúp xây dựng các ứng dụng web một cách đơn giản và dễ dàng mở rộng. NodeJS được phát triển bởi Ryan Dahl vào năm 2009 và có thể chạy trên nhiều hệ điều hành khác nhau: OS X, Microsoft Windows, Linux. Hiện nay, ngôn ngữ lập trình phía máy chủ có thể sử dụng các ngôn ngữ C++, C#, Python nhưng NodeJS đem lại những ưu diểm. NodeJS được viết bằng JavaScript, vì thế, lập trình viên sẽ tốn ít công sức hơn khi làm quen với NodeJS nếu nền tảng lập trình phía client sẵn có. NodeJS có một công đồng lớn, chính vì vậy, nếu bạn cần hỗ trợ gì về NodeJS, sẽ nhanh chóng có người hỗ trợ bạn. Không chỉ vậy, NodeJS còn được đánh giá có tốc độ xử lý nhanh, nhờ cơ chế xử lý bất đồng độ (non-blocking), NodeJS có thể xử lý hàng ngàn kết nối cùng lúc mà không gặp bất cứ khó khăn nào, dễ dàng mở rộng, phát huy hiệu quả với các ứng dụng thời gian thực như các ứng dụng nhắn tin, các mạng xã hội.

Tuy nhiên, bên cạnh đó, Nodejs gây hao tốn tài nguyên và thời gian. Nodejs được viết bằng C++ và JavaScript nên khi xử lý cần phải trải qua một quá trình biên dịch. Nếu bạn cần xử lý những ứng dụng tốn tài nguyên CPU thì không nên sử dụng Nodejs. NodeJS phù hợp cho việc phát triển các ứng dụng mới, tuy nhiên đối với các dự án lớn thì không phải là sự lựa chọn tối ưu.

Với đồ án tốt nghiệp, em sử dụng NodeJS và ExpressJS là một framework được xây dựng trên nền tảng của NodeJS. ExpressJS cung cấp các tính năng mạnh mẽ để phát triển web hoặc mobile. ExpressJS hỗ trợ các method HTTP và midleware tạo ra API vô cùng mạnh mẽ và dễ sử dụng. hiết lập các lớp trung gian để trả về các HTTP request. Define router cho phép sử dụng với các hành động khác nhau dựa trên phương thức HTTP và URL.Cho phép trả về các trang HTML dựa vào các tham số.

## Công nghệ JWT

Hệ thống sử dụng JWT giúp phân quyền người dùng, nhằm để phân biệt yêu cầu gửi từ client đến server xác thực quyền và bảo mật các tài nguyên của hệ thống. JSON Web Token (JWT) là một chuẩn mở (RFC 7519) định nghĩa một cách nhỏ gọn và khép kín để truyền một cách an toàn thông tin giữa các bên dưới dạng đối tượng JSON. Thông tin này có thể được xác minh và đáng tin cậy vì nó có chứa chữ ký số. JWTs có thể được ký bằng một thuật toán bí mật (với thuật toán HMAC) hoặc một public / private key sử dụng mã hoá RSA. JWT là sự kết hợp (bởi dấu .) một Object Header dưới định dạng JSON được encode base64, một payload object dưới định dạng JSON được encode base64 và một Signature cho URI cũng được mã hóa base64.

## Công nghệ lập trình thời gian thực SocketIO

Trong các hệ thống hiện đại ngày nay, việc giao tiếp từ máy chủ đến máy khách là điều bắt buộc, trong trường hợp bạn cần máy khách(client) hay máy chủ(server) có thể nhận sự thay đổi bên kia thì bạn có thể giải quyết bằng cách sử dụng : AJAX, long-polling, short-polling, & HTML 5 server-sent events... Tuy nhiên, sử dụng các cách này có độ trễ lớn và gây tốn tài nguyên. Bởi vậy, SocketIO ra đời cho phép xây dựng để xử lý việc giao tiếp giữa server và client ngay lập tức và chiếm ít tài nguyên nhất.

Trong SocketIO, các kết nối sẽ được thiết lập với WebSocket, cung cấp một kênh giao tiếp chi phí thấp giữa máy chủ và máy khách. Trong trường hợp kết nối WebSocket không thể thực hiện được, nó sẽ trở lại trạng thái bỏ phiếu dài HTTP. Nếu kết nối bị mất, máy khách sẽ tự động cố gắng kết nối lại. SocketIO mở rộng quy mô đến nhiều máy chủ và gửi sự kiện đến tất cả các máy khách được kết nối một cách dễ dàng.

Trong đồ án, hệ thống có sử dụng SocketIO để realtime cho các chức năng: tương tác với bài viết, trao đổi giữa người dùng, nhận thông báo giữa các người dùng… Việc sử dụng SocketIO đem lại nhiều hiệu quả, giúp người sử dụng dễ dàng nắm bắt các thông tin nhanh chóng, dễ dàng và thuận tiện nhất.

## Cơ sở dữ liệu MongoDB

Hệ thống sử dụng MongoDB - một hệ quản trị cơ sở dữ liệu thích hợp cho các ứng dụng mạng xã hội, bán hàng. Vì vậy, MongoDB sẽ tránh cấu trúc table-based của relational database để thích ứng với các tài liệu như JSON có một schema rất linh hoạt gọi là BSON. MongoDB sử dụng lưu trữ dữ liệu dưới dạng Document JSON nên mỗi một collection sẽ các các kích cỡ và các document khác nhau. Các dữ liệu được lưu trữ trong document kiểu JSON nên truy vấn sẽ rất nhanh.

Trong MongoDB có hai khái niệm cơ bản là: Collection và Document. Một document trong MongoDB có thể hiểu như một bản ghi record trong cơ sở dữ liệu quan hệ. Tuy nhiên có sự khác biệt giữa hai loại khái niệm này là một document không nhất thiết phải có cấu trúc cố định, còn record cần cấu trúc dữ liệu phải giống hệt nhau giữa các bản ghi. Collection trong MongoDB tương đương như một bảng trong kiểu cơ sở dữ liệu quan hệ. Một Collection gồm nhiều documents ở trong nó và mỗi Collection phải có kiểu cấu trúc riêng của nó.

Do MongoDB sử dụng lưu trữ dữ liệu dưới dạng Document JSON nên mỗi một collection sẽ có các kích cỡ và các document khác nhau, linh hoạt trong việc lưu trữ dữ liệu. Dữ liệu trong MongoDB không có sự ràng buộc lẫn nhau, không có join như trong RDBMS nên khi insert, xóa hay update nó không cần phải mất thời gian kiểm tra xem có thỏa mãn các ràng buộc dữ liệu như trong RDBMS tạo ra hiệu suất truy vấn cao hơn so với RDBMS.

Trong MongoDB có hai khái niệm cần nắm rõ: Normalization và Denormalization. Normalization là kỹ thuật chia dữ liệu thành nhiều bảng để giảm bớt sự dư thừa và không nhất quán của dữ liệu và để đạt được tính toàn vẹn của dữ liệu. Mặt khác, Denormalization là kỹ thuật kết hợp dữ liệu vào một bảng duy nhất để làm cho việc truy xuất dữ liệu nhanh hơn. Normalization hỗ trợ tốt cho việc chèn, xóa, sửa dữ liệu, Denormalization lại hỗ trợ tốt cho việc đọc, truy vấn dữ liệu. MongoDB giúp bạn có thể thực hiện cả 2 điều này. Chính vì vậy, phân tích nghiệp vụ hệ thống và sử dụng phù hợp một trong hai hình thức tổ chức dữ liệu này (Normalization hay Denormalization) sẽ giúp cho hệ thống có hiệu suất cao, tốc độ thực thi nhanh.

Tuy nhiên, MongoDB cũng có mặt hạn chế. Đó là cần RAM lớn do dữ liệu được caching. Dữ liệu trong MongoDB không bị ràng buộc như RDBMS nhưng người sử dụng lưu ý cẩn thận mọi thao tác để không xảy ra các kết quả ngoài ý muốn gây ảnh hưởng đến dữ liệu.

# Phát triển và triển khai ứng dụng

## Thiết kế kiến trúc

### Lựa chọn kiến trúc phần mềm

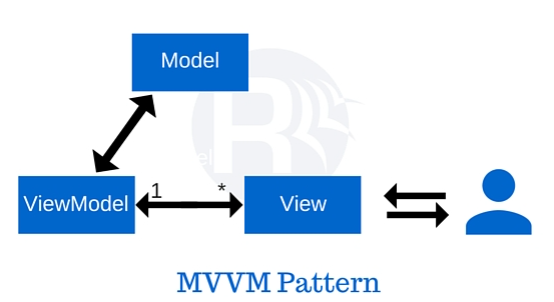
Hệ thống được triển khai trên nền tảng ứng dụng website, dựa trên kiến trúc client/server theo mô hình 3 lớp MVVM. Đề hiểu MVVM là gì, trước hết, ta cần tìm hiểu MVC là gì? vì MVVM là dẫn xuất của MVC.

MVC từ viết tắt bởi 3 từ Model – View – Controller. Đây là mô hình thiết kế sử dụng trong kỹ thuật phần mềm. Mô hình source code thành 3 phần, tương ứng mỗi từ. Mỗi từ tương ứng với một hoạt động tách biệt trong một mô hình. Model (M): là bộ phận có chức năng lưu trữ toàn bộ dữ liệu của ứng dụng. Bộ phận này là một cầu nối giữa 2 thành phần bên dưới là View và Controller. Model thể hiện dưới hình thức là một cơ sở dữ liệu hoặc có khi chỉ đơn giản là một file XML bình thường. Model thể hiện rõ các thao tác với cơ sở dữ liệu như cho phép xem, truy xuất, xử lý dữ liệu… View (V): đây là phần giao diện (theme) dành cho người sử dụng. Đó là nơi mà người dùng có thể lấy được thông tin dữ liệu của MVC thông qua các thao tác truy vấn như tìm kiếm, lọc, hoặc sử dụng thông qua các website. Thông thường, các ứng dụng web sử dụng MVC View như một phần của hệ thống, nơi các thành phần HTML được tạo ra. Bên cạnh đó, View cũng có chức năng ghi nhận hoạt động của người dùng để tương tác với Controller. Tuy nhiên, View không có mối quan hệ trực tiếp với Controller, cũng không được lấy dữ liệu từ Controller mà chỉ hiển thị yêu cầu chuyển cho Controller mà thôi. Ví dụ: Nút “delete” được tạo bởi View khi người dùng nhấn vào nút đó sẽ có một hành động trong Controller. Controller (C): bộ phận có nhiệm vụ xử lý các yêu cầu người dùng đưa đến thông qua view. Từ đó, C đưa ra dữ liệu phù hợp với người dùng. Bên cạnh đó, Controller còn có chức năng kết nối với model.



**Hình 13** Mô hình MVC

Đối với MVVM, MVVM hỗ trợ cơ chế two-way binding giữa View và ViewModel. Điều này cho phép tự động đồng bộ các thay đổi giữa 2 thành phần. Nhìn chung, ViewModel tối ưu sử dụng pattern Observer để thông báo sự thay đổi trong ViewModel sang Model. View: Tương tự như trong mô hình MVC, View là phần giao diện của ứng dụng để hiển thị dữ liệu và nhận tương tác của người dùng. Một điểm khác biệt so với các ứng dụng truyền thống là View trong mô hình này tích cực hơn. Nó có khả năng thực hiện các hành vi và phản hồi lại người dùng thông qua tính năng binding, command. Model: Cũng tương tự như trong mô hình MVC. Model là các đối tượng giúp truy xuất và thao tác trên dữ liệu thực sự. ViewModel: Lớp trung gian giữa View và Model. ViewModel có thể được xem là thành phần thay thế cho Controller trong mô hình MVC. Nó chứa các mã lệnh cần thiết để thực hiện data binding, command. ViewModel không hề biết gì về View, một ViewModel có thể được sử dụng cho nhiều View (one-to-many). ViewModel sử dụng Observer design pattern để liên lạc với View (thường được gọi là binding data, có thể là 1 chiều hoặc 2 chiều tùy nhu cầu ứng dụng).

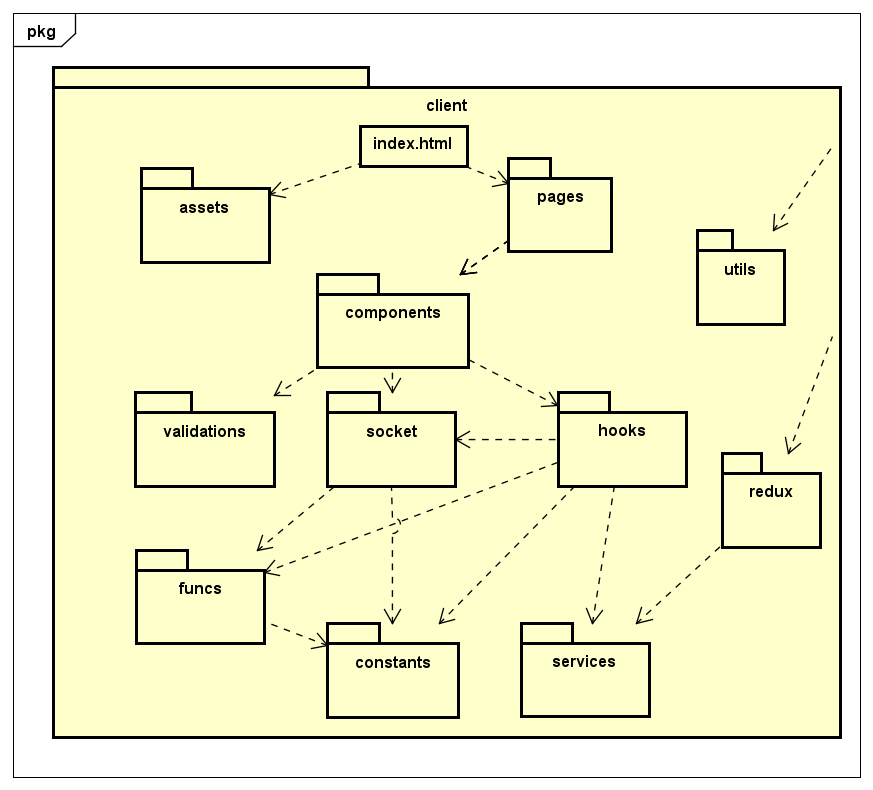


**Hình 14** Mô hình MVVM

Trong hệ thống này, phần V (View) sử dụng ReactJS chia ứng dụng thành pages (các trang) được tách riêng đảm bảo tính coupling, V (View) được có thể lọc, cache giao tiếp với service một cách tích cực và hiệu quả. Một page (trang) được chia thành các component (thành phần) để dễ tái sử dụng. Phần VM (ViewModel) bao gồm các gói routes, controllers, sockets thực hiện việc tiếp nhận dữ liệu và phản hồi dữ liệu cho người dùng. Phần M (Model) nằm trong gói models, servers cung cấp các phương thức CRUD được gọi bởi tầng VM (ViewModel).

### Thiết kế tổng quan

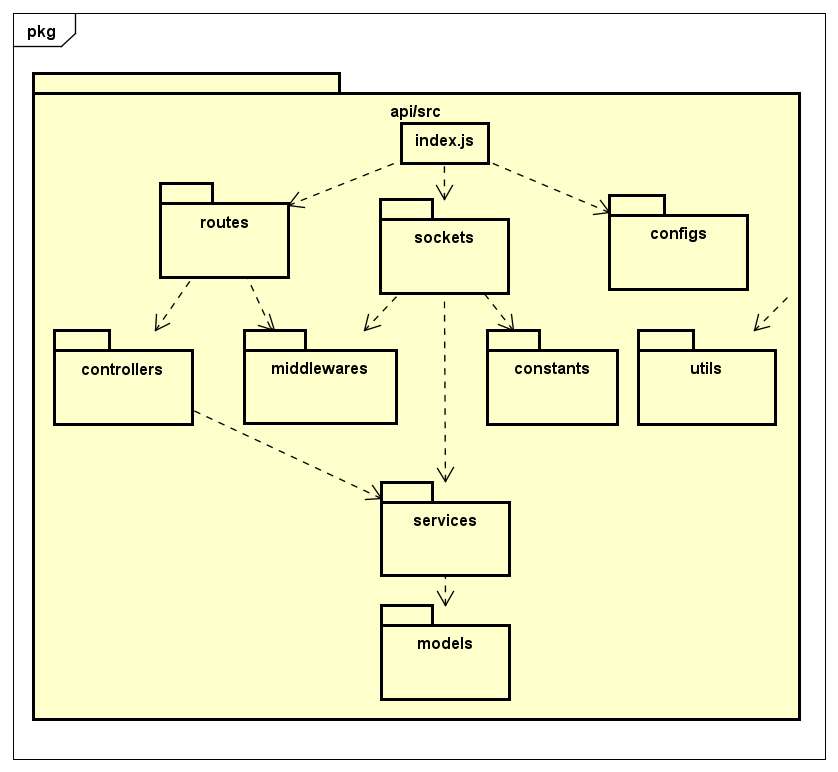
##### Biều đồ phụ thuộc gói client



**Hình 15** Biều đồ phụ thuộc gói Client

|  |  |  |
| --- | --- | --- |
| STT | Gói | Mô tả |
|  | assets | Chứa file logo của hệ thống |
|  | pages | Chứa các trang của hệ thống |
|  | components | Chứa các thành phần của hệ thống có thể tái sử dụng |
|  | validations | Chứa các file xác thực form của hệ thống |
|  | socket | Chứa các file đảm nhận tạo, nhận, sự kiện từ socketIO của hệ thống |
|  | hooks | Chứa file lưu trữ các state của components giúp tái sử dụng, dễ bảo trì hệ thống |
|  | funcs | Xứ lý các dữ liệu từ socketIO nhận được |
|  | constants | Chứa các hằng số của socketIO và hệ thống |
|  | services | Có nhiệm vụ gọi các truyền nhận dữ liệu với server theo giao thức RESTful API |
|  | redux | Chứa các state chung của hệ thống |
|  | utils | Chứa các tiện ích: định dạng ngày, định dạng số, media… được sử dụng trong toàn bộ hệ thống |

##### Biều đồ phụ thuộc gói server

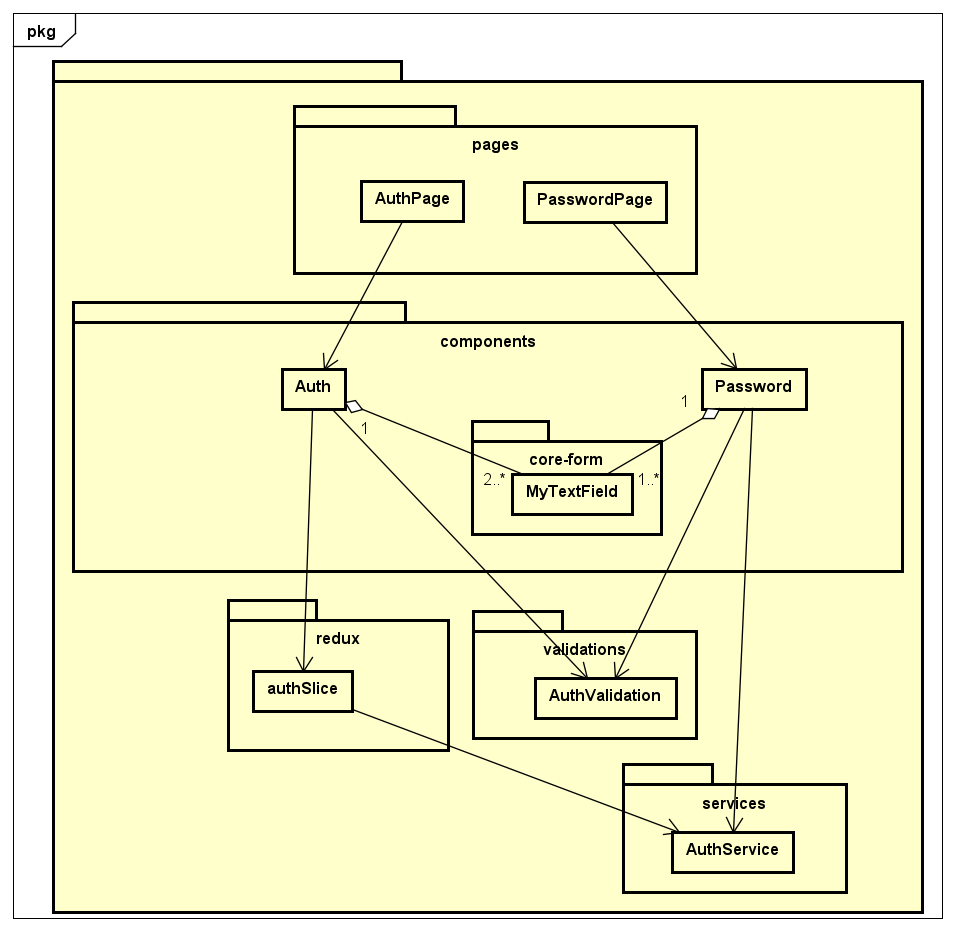


**Hình 16** Tổng quan gói Server

|  |  |  |
| --- | --- | --- |
| STT | Gói | Mô tả |
|  | routes | Xử lý phân định các yêu cầu lời gọi RESTful API đầu vào |
|  | sockets | Tạo, nhận các sự kiện của hệ thống |
|  | configs | Cấu hình của hệ thống |
|  | middlewares | Xác thực tính hợp lệ của lời gọi RESTful API, socketIO, xác định danh tính người dùng |
|  | constants | Chứa các hằng số của socketIO và hệ thống |
|  | controllers | Định dạng dữ liệu trả về, xử lý lỗi |
|  | services | Thực hiện các nghiệp vụ cho hệ thống |
|  | models | Bao gồm các thành phần cơ bản của hệ thống, với thuộc tính, các phương thức đại diện cho các bảng trong cơ sở dữ liệu. |
|  | utils | Chứa các tiện ích của hệ thống |

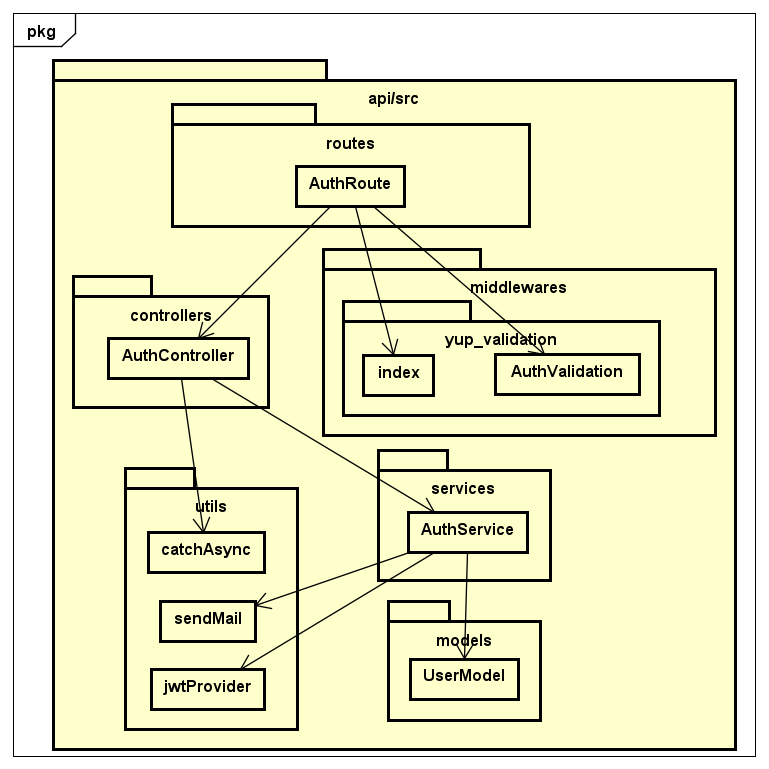
### Thiết kế chi tiết gói

##### Thiết kế chi tiết gói auth



**Hình 17** Thiết kế chi tiết gói AuthClient

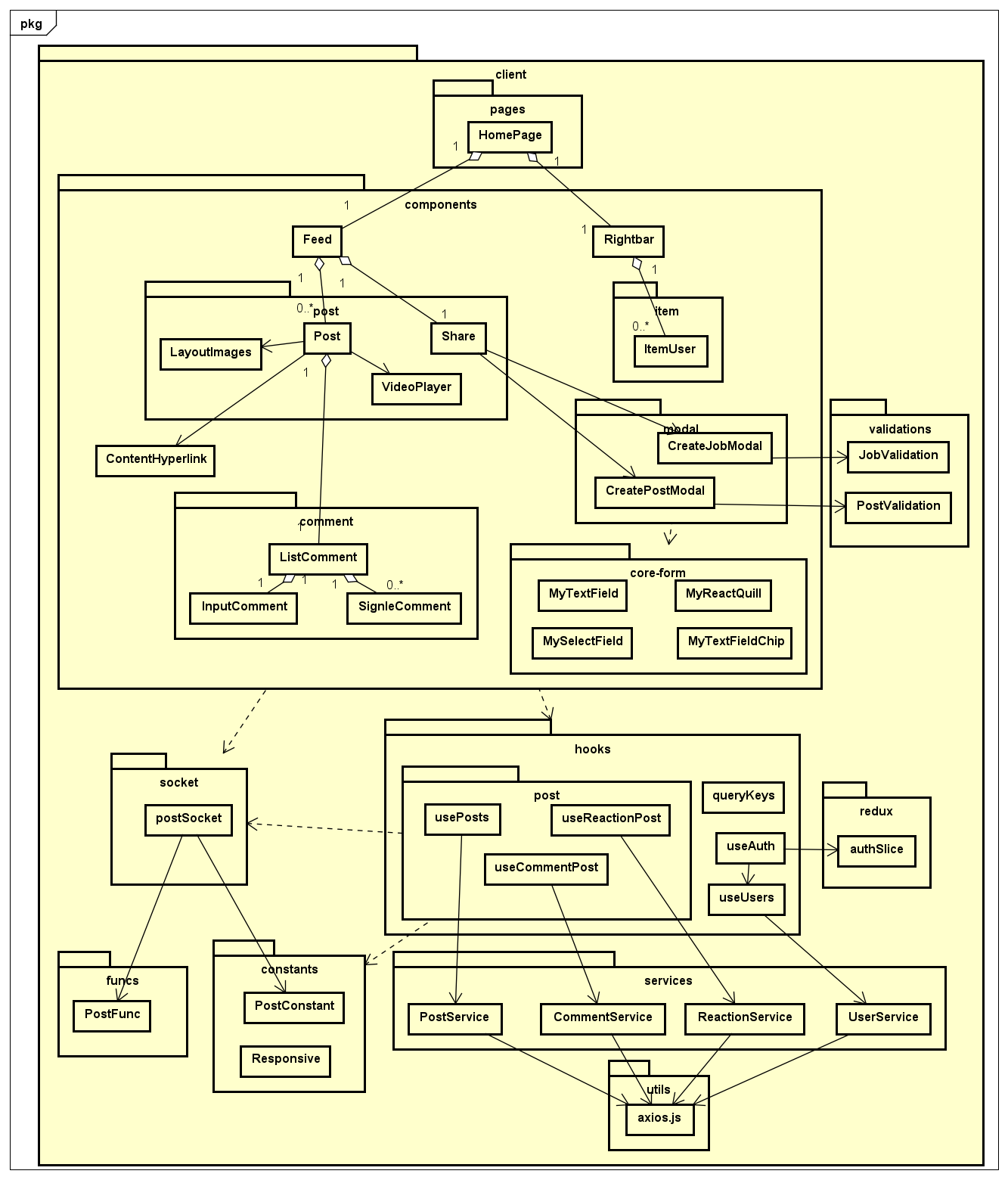
|  |  |  |
| --- | --- | --- |
| STT | Lớp | Mô tả |
|  | AuthPage | Hiện thị trang đăng nhập hoặc đăng ký |
|  | PasswordPage | Hiện thị trang quên mật khẩu hoặc trang thay đổi mật khẩu |
|  | Auth | Tạo ra mẫu đơn đăng nhập hoặc đăng ký sử dụng react-hook-form |
|  | Password | Tạo ra mẫu đơn quên mật khẩu hoặc thay đổi mật khẩu |
|  | MyTextField | Là trường input để điền vào form |
|  | authSlice | Chứa các hành động đăng nhập và đăng ký |
|  | AuthValidation | Chứa các schema giúp xác thực biểu mẫu của form |
|  | AuthService | Giao tiếp nhận dữ liệu từ server |



**Hình 18** Thiết kế chi tiết gói AuthServer

|  |  |  |
| --- | --- | --- |
| STT | Lớp | Mô tả |
|  | AuthRoute | Điều hướng các RESTful API từ client |
|  | index | Xác thực dữ liệu tới từ client: query,params, body |
|  | AuthValidation | Chứa các schema để xác thực dữ liệu từ client |
|  | AuthController | Gọi các nghiệp vụ và định dạng đầu ra của dữ liệu |
|  | catchAsync | Bắt lỗi ngoại lệ |
|  | AuthService | Thực hiện nghiệp vụ của hệ thống: đăng ký, đăng nhập, đổi mật khẩu… |
|  | sendMail | Dịch vụ gửi thư điện tử |
|  | jwtProvider | Ký mã token để xác thực người dùng |
|  | UserModel | Định dạng kiểu mẫu dữ liệu người dùng được lưu trữ trên cơ sở dữ liệu |

##### Thiết kế chi tiết trang chủ



**Hình 19** Thiết kế màn hình trang chủ

---còn tiếp----

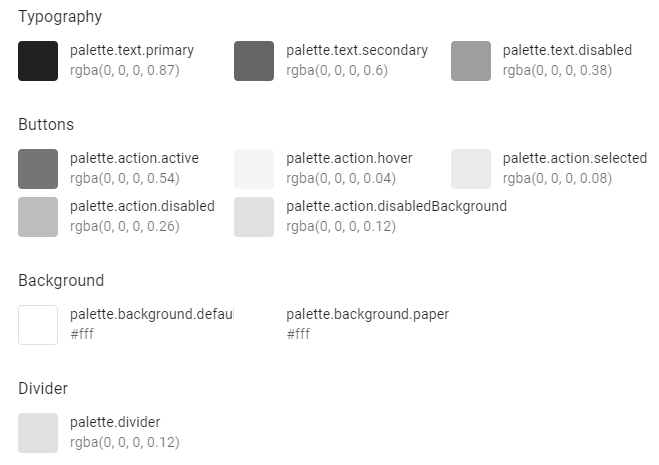
## Thiết kế chi tiết

### Thiết kế giao diện

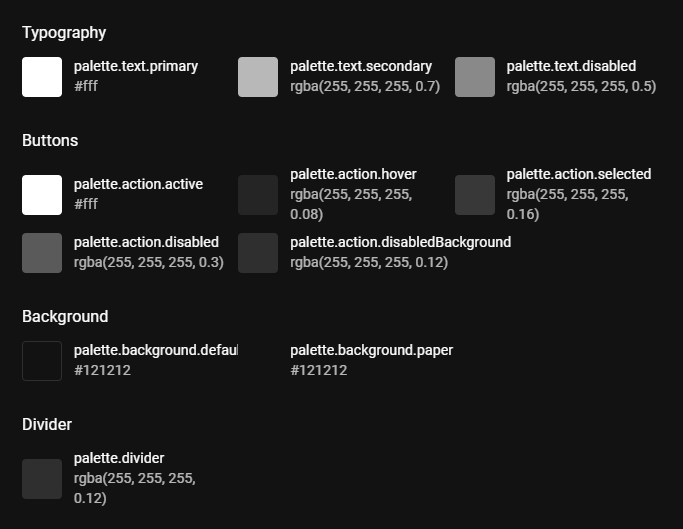
Thiết kế giao diện của người dùng phải đảm bảo tuân thủ các nguyên tắc thiết kế, đáp ứng đầy đủ yêu cầu người dùng. Hệ thống phải đảm bảo đơn giản hóa các tác vụ cho người sử dụng, trình bày bố cục thuận tiện rõ ràng nhất cho người sử dụng dễ dàng hiểu và thao tác

Giao diện hệ thống hỗ trợ tính responsive với từng kích thước pixel của màn hình thiết bị. Cụ thể, hệ thống sẽ quy định các breakpoint để hiển thị giao diện tương ứng, có 4 loại breakpoint được hệ thống hỗ trợ: xs, extra-small: 0px; sm, small: 600px; md, medium: 900px; lg, large: 1200px;

Về màu sắc, hệ thống có hai chế độ: sáng và tối, các màu sắc của hệ thống được dựa trên bảng màu của thư viện MUI – một thư viện hỗ trợ xây dựng giao diện nhanh chóng, dễ dàng tùy chỉnh.

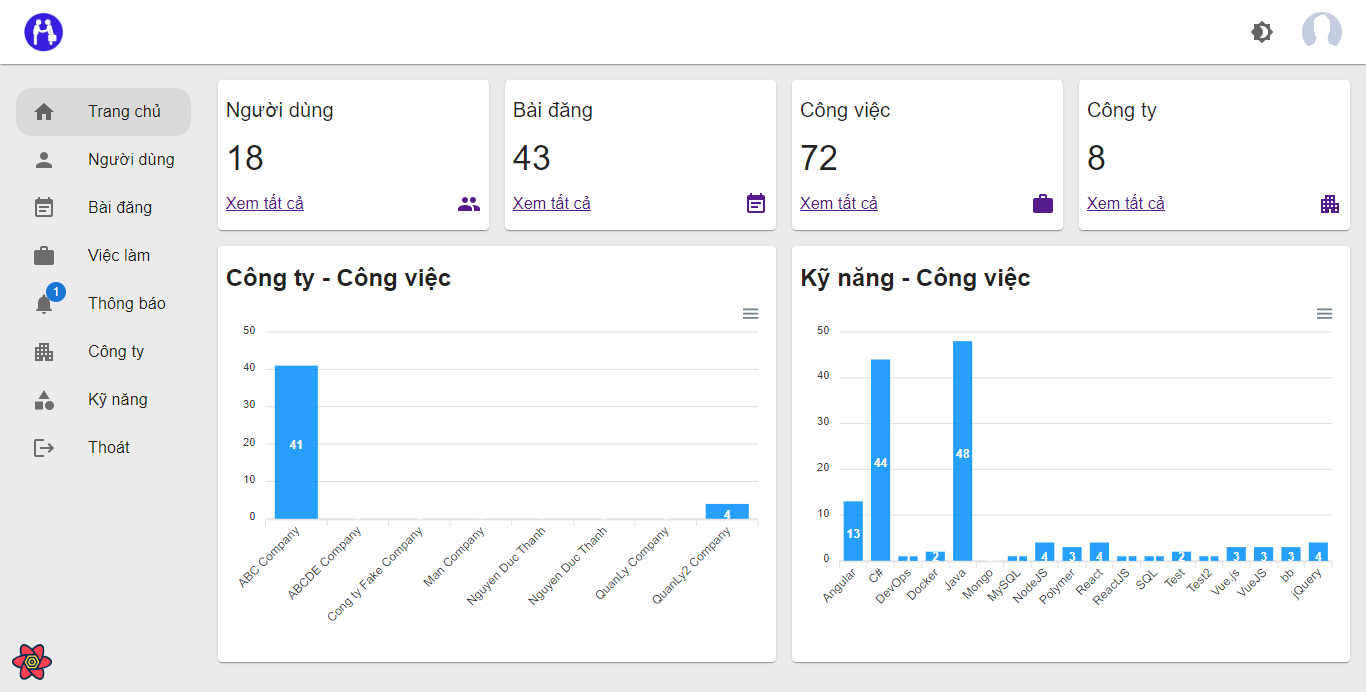


**Hình 20** Bảng màu với giao diện sáng



**Hình 21** Bảng màu với giao diện sáng

Một số giao diện minh họa cho chức năng của hệ thống



**Hình 22** Giao diện trang chủ quản trị viên

--còn tiếp---

### Thiết kế lớp

##### Lớp JobService chứa các nghiệp vụ liên quan đến tin tuyển dụng của hệ thống

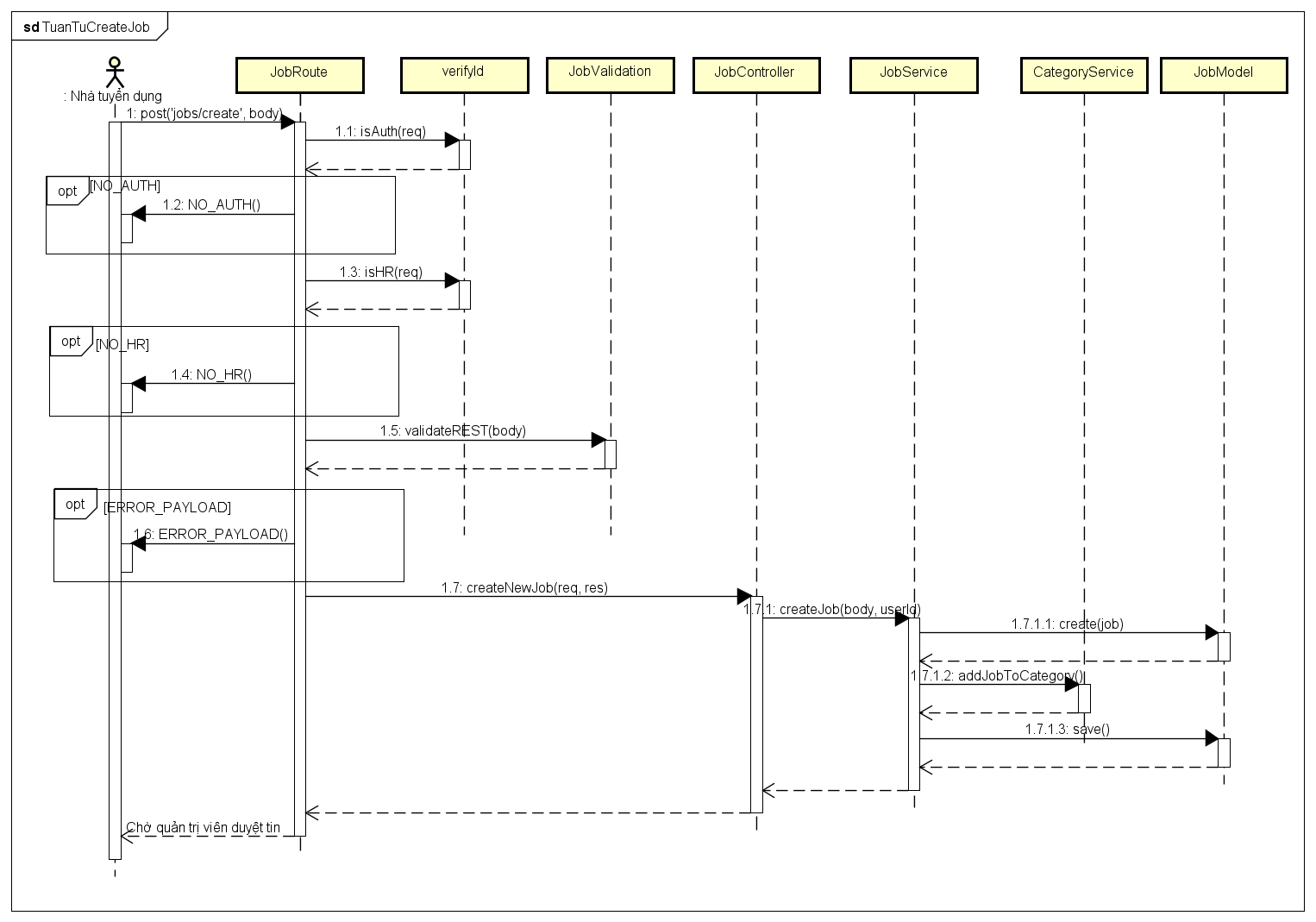
Lớp bao gồm các phương thức và thuộc tính sau đây:

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Mô tả |
|  | Category | Category | Một đối tượng của CategoryModel chứa phương thức thao tác với bảng Category |
|  | Company | Company | Chứa các phương thức thao tác với bảng Company |
|  | Job | Job | Chứa các phương thức thao tác với bảng Job |
|  | User | User | Chứa các phương thức thao tác với bảng User |
|  | convertToSlug | convertToSlug | Chuyển chuỗi thành chuỗi slug |
|  | CategoryService | CategoryService | Nghiệp vụ của category |
|  | RoomService | RoomService | Nghiệp vụ của room |

**Bảng 10** Mô tả thuộc tính của lớp JobService

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Phương thức | Kiểu trả về | Mô tả |
|  | createJob | Object | Tạo mới công việc |
|  | updateActiveJobs | Object | Cập nhật trạng thái của công việc theo companyId |
|  | updateApplicantJob | Object | Người tìm kiếm việc làm ứng tuyển công việc |
|  | deleteApplicantJob | Object | Chủ sở hữu tin tuyển dụng xóa ứng viên |
|  | updateJob | Object | Nhà tuyển dụng cập nhật công việc hoặc quản trị viên thêm lý do từ chối công việc |
|  | confirmJob | Object | Quàn trị viên từ chối hoặc chấp nhận công việc |
|  | getJobByCategoryService | Object | Tìm công việc theo tên kỹ năng |
|  | listJobRelateCurrentJob | Object | Tìm các công việc liên quan theo mã công việc |
|  | infoJob | Object | Hiện thị thông tin chi tiết công việc với từng loại tác nhân: người tìm việc, nhà tuyển dụng, quản trị viên, nhà quản lý công ty. |
|  | recommendJobByUser | Object | Gợi ý các công việc dựa vào hồ sơ kỹ năng của người dùng |
|  | viewApplyJobService | Object | Xem đơn ứng tuyển |
|  | listJobByCompanyService | Object | Danh sách công việc của công ty |
|  | listMyJobHR | Object | Danh sách công việc của nhà tuyển dụng |
|  | getJobStatsService | Object | Thống kê danh sách việc làm trong năm theo tháng |
|  | getJobStatusService | Object | Thống kê trạng thái của công việc |
|  | changeOwnerJob | Object | Nhà quản lý công ty thay đổi quyền sở hữu của tin tuyển dụng |

**Bảng 11** Mô tả phương thức của lớp JobService



**Hình 23** Biểu đồ tuần tự createJob

##### Lớp PostService chứa các nghiệp vụ lên quan đến bài đăng của hệ thống

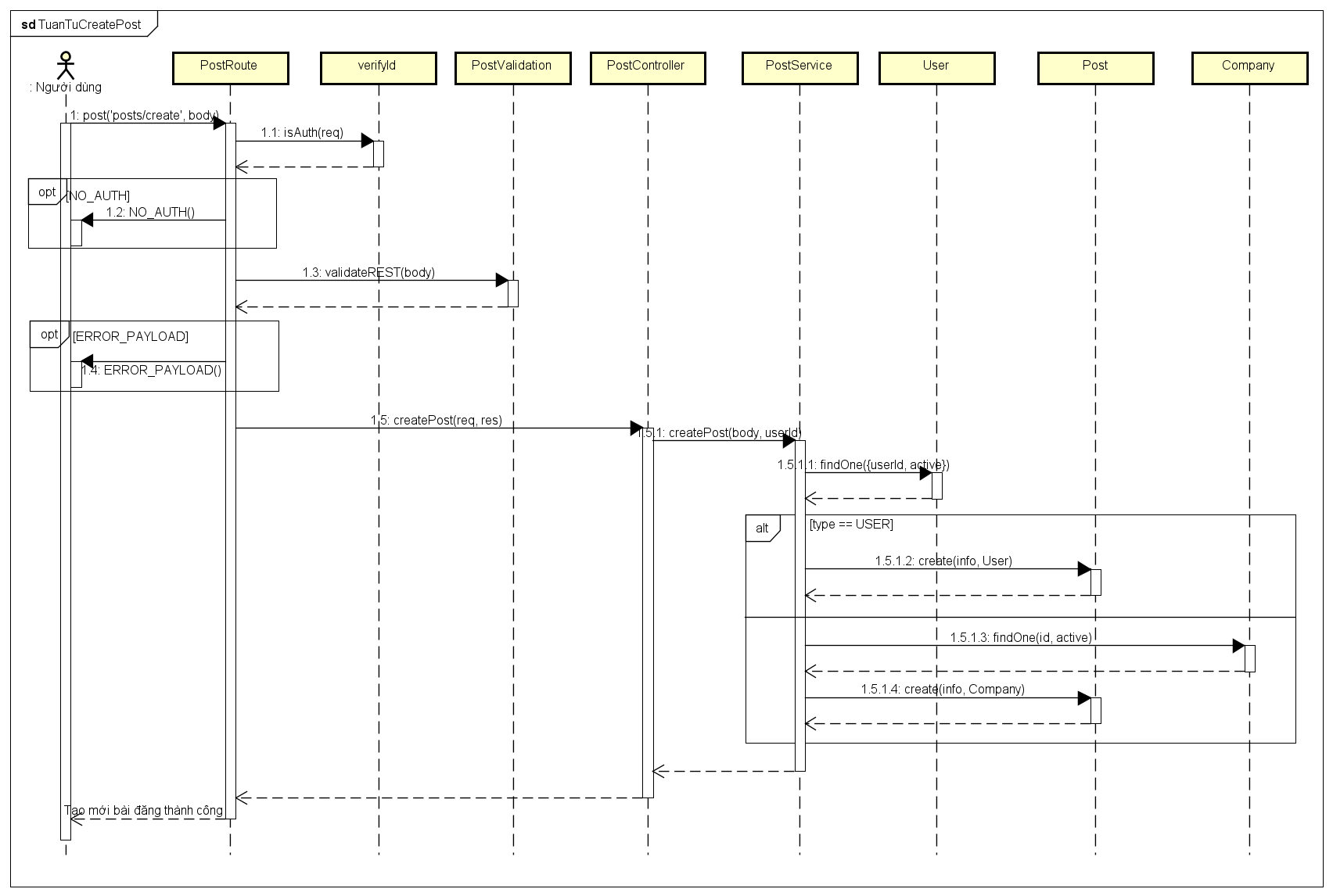
Lớp bao gồm các phương thức và thuộc tính sau đây:

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên thuộc tính | Kiểu dữ liệu | Mô tả |
|  | Post | Post | Là một đối tượng của PostModel chứa phương thức giao tiếp với bảng Post |
|  | Company | Company | Chứa các phương thức thao tác với bảng Company |
|  | Reaction | Reaction | Chứa các phương thức thao tác với bảng Reaction |
|  | Comment | Comment | Chứa các phương thức thao tác với bảng Comment |
|  | User | User | Chứa các phương thức thao tác với bảng User |

**Bảng 12** Mô tả thuộc tính của lớp PostService

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Phương thức | Kiểu trả về | Mô tả |
|  | createPost | Object | Tạo mới bài đăng kiểu cá nhân và công ty |
|  | findPost | Object | Tìm bài đăng theo mã |
|  | listPostAdmin | Object | Danh sách bài đăng với quyền quản trị viên |
|  | listPostByUser | Object | Danh sách bài đăng của người dùng lấy theo mã người dùng |
|  | listPostByCompany | Object | Danh sách bài đăng của công ty lấy theo mã công ty |
|  | listPostByFeed | Object | Danh sách bài đăng dựa trên lượt theo dõi người dùng và theo dõi công ty của người dùng |
|  | getPostStatsService | Object | Thống kê danh sách bài đăng của năm theo tháng |
|  | getPostTypeService | Object | Thống kê danh sách bài đăng dựa theo kiểu bài viết: cá nhân hoặc công ty. |
|  | deletePostService | Object | Xóa bài đăng, tương tác, bình luận của bài đăng. |

**Bảng 13** Mô tả phương thức của lớp PostService



**Hình 24** Biểu đồ tuần tự createPost

### Thiết kế cơ sở dữ liệu

## Xây dựng ứng dụng

### Thư viện và công cụ sử dụng

**Bảng 14** Danh sách thư viện và công cụ sử dụng

|  |  |  |
| --- | --- | --- |
| **Mục đích** | **Công cụ** | **Địa chỉ URL** |
| IDE lập trình | Visual Studio Code | https://code.visualstudio.com/ |
| Lập trình Front-end | React (v17.0.2) | <https://reactjs.org/> |
| Lập trình Back-end | NodeJS (v14.17.0) | <https://nodejs.org/en/> |

### Kết quả đạt được

### Minh hoạ các chức năng chính

## Kiểm thử

## Triển khai

# Các giải pháp và đóng góp nổi bật

# Kết luận và hướng phát triển

## Kết luận

Hệ thống xây dựng đã cung cấp được các chức năng cơ bản giúp nhà tuyển dụng và người lao động có một kênh thông tin có thể giao tiếp với nhau một cách hiệu quả và đáng tin cậy. Nhà tuyển dụng có thể đăng tin tuyển dụng, tìm kiếm các ứng viên tiềm năng và có thể quảng bá cho công ty của mình qua kênh thông tin chung của hệ thống,… Người lao động được cung cấp một mạng lưới các doanh nghiệp tiềm năng dựa trên việc khai báo hồ sơ cá nhân của mình từ đó tìm được các công việc phù hợp với bản thân.

Hệ thống là một mạng xã hội hướng việc làm vì vậy so với Facebook hay mạng các trang mạng xã hội khác thì hệ thống cung cấp một cách chính xác, hiệu quả các công việc cho người lao động không khiến cho họ phải tìm kiếm vất vả như các trang mạng xã hội thông thường.

Hệ thống cung cấp dịch vụ thông báo hai chiều giữa người lao động và nhà tuyển dụng giúp họ có thế tìm kiếm nhau dễ dàng hơn. Hơn nữa, các thông tin trong hệ thống đều được quản trị viên kiểm duyệt đảm bảo tính đúng đắn. Chính vì vậy, hệ thống sẽ cung cấp các thông tin hữu ích so với các trang thông tin tìm việc hiện nay như timviecnhanh.vn, vietnamworks.com,… chỉ cung cấp một chiều từ ứng viên đến nhà tuyển dụng.

Hệ thống có một mạng lưới các bảng tin giúp người tìm kiếm việc làm nắm bắt thông tin chính xác về các hoạt động của người dùng và công ty mình quan tâm. Đối với nhà tuyển dụng, bảng tin giúp cho họ nắm bắt thông tin nhanh chóng những hoạt động xung quanh các ứng viên tiềm năng. Người sử dụng không phải mất công phải “chuyển” tab hay lướt web các trang mạng xã hội khác để có thể hiểu hơn về công ty, điều này sẽ cải thiện hơn so với các trang website tìm kiếm việc làm hiện nay như: topCV, ITViec,…

Hệ thống cung cấp một cách rạch ròi về việc trao đổi giữa việc trao đổi cá nhân và công việc. Từ đó, người lao động có thể trao đổi lẫn nhau để hiểu hơn về công ty mà họ quan tâm dựa theo ý kiến chủ quan của họ. Hơn nữa, hệ thống phân chia như vậy sẽ giúp nhà quản lý dễ dàng phân chia công việc của công ty mình một cách hiệu quả, đó là điểm mạnh mà hệ thống đem lại so với các hệ thống mạng xã hội tìm kiếm việc làm hiện nay như LinkedIn.

Qua quá trình làm đồ án tốt nghiệp, em đã học được nhiều công nghệ mới, cũng như kỹ năng viết báo cáo. Việc làm đồ án tốt nghiệp giúp em vận dụng những gì đã học được xuyên suốt quá trình học đại học: phân tích hệ thống, thiết kế giao diện, thiết kế cơ sở dữ liệu, thiết kế hệ thống, kiểm thử sản phẩm,… Đó sẽ là hành trang vững chắc trong công việc sau này của em.

Do thời gian làm đồ án có hạn, cũng như hạn chế của năng lực bản thân nên hệ thống em xây dựng vẫn còn nhiều thiếu xót. Hệ thống vẫn chưa có chức năng đăng nhập bằng bằng mạng xã hội Facebook, tài khoản Google,…

## Hướng phát triển

Để chương trình có thể hoạt động hiệu quả, dữ liệu phải được lưu trữ bằng hệ quản trị cơ sở dữ liệu có tính bảo mật hơn. Hệ thống cần có sự giao tiếp song song giữa quản trị viên và người dùng một cách chặt chẽ và nhiều hơn để tối ưu trải nghiệm của người dùng.

Hệ thống cần mở rộng chức năng đăng nhập cho người dùng sử dụng thư viện PassportJS để người dùng có thể đăng nhập từ các bên thứ 3 giúp làm giảm thời gian nhập liệu cho người dùng.

Hệ thống cần cải thiện thêm tính năng đánh giá công việc thông qua các phiếu đánh giá từ chính người tìm việc của hệ thống. Để làm được điều đó, ta cần tạo thêm một Evaluate Collection trong database đươc liên kết khóa ngoại với id của công việc mà người tim việc đánh giá.

Hệ thống cần được chỉnh sửa để có thể tương tác dễ dàng qua thiết bị màn hình như: di động, iPad, iWatch. Đồng thời, hệ thống cần phải chỉnh sửa lại giao diện phối màu,bố cục… để tăng thêm trải nhiệm của người dùng.

# Tài liệu tham khảo

Trang đặc tả JWT của tổ chức auth0: <https://jwt.io/introduction>

Mẫu mô hình MVC, MVVM của tổ chức TINO: <https://wiki.tino.org/mvc-mvp-mvvm-la-gi/>

...

# Phụ lục