|  |  |
| --- | --- |
|  |  |
| NGUYỄN HẢI TRIỀU | **BỘ CÔNG THƯƠNG**  **TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  **---------------------------------------** |
|  |
| **ĐỒ ÁN TỐT NGHIỆP**  **NGÀNH CÔNG NGHỆ THÔNG TIN** |
|  |
| **XÂY DỰNG WEBSITE ĐĂNG TIN TUYỂN DỤNG VIỆC LÀM DỰA TRÊN REACTJS, NESTJS VÀ MONGODB** |
|  |
|  |
| **CBHD :** **TS. Đặng Trọng Hợp** |
| CÔNG NGHỆ THÔNG TIN | **Sinh viên : Nguyễn Hải Triều** |
| **Mã sinh viên: 2020604195** |
|  |
|  |
|  |
| **Hà Nội - 2024** |
|  |

|  |  |
| --- | --- |
|  |  |
| NGUYỄN HẢI TRIỀU | **BỘ CÔNG THƯƠNG**  **TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  **---------------------------------------** |
|  |
| **ĐỒ ÁN TỐT NGHIỆP**  **NGÀNH CÔNG NGHỆ THÔNG TIN** |
|  |
| **XÂY DỰNG WEBSITE ĐĂNG TIN TUYỂN DỤNG VIỆC LÀM DỰA TRÊN REACTJS, NESTJS VÀ MONGODB** |
|  |
|  |
| **CBHD :** **TS. Đặng Trọng Hợp** |
| CÔNG NGHỆ THÔNG TIN | **Sinh viên : Nguyễn Hải Triều** |
| **Mã sinh viên: 2020604195** |
|  |
|  |
|  |
| **Hà Nội - 2024** |
|  |

# LỜI CẢM ƠN

Sau khoảng thời gian học tập tại trường và được tham gia các hoạt động phát triển phần mềm được sự truyền đạt kiến thức, lời khuyên trong công việc và giúp đỡ tận tình của quý thầy cô, em đã có cơ hội ra ngoà i thực tập ở công ty, được tiếp xúc với môi trường làm việc thực tế, tiếp xúc với các công cụ quản lý dự án hiện đại,… Em xin chân thành cảm ơn các thầy cô, giảng viên trường Đại học Công Nghiệp Hà Nội nói chung và khoa Công nghệ thông tin nói riêng - những người đã ân cần giảng dạy, truyền đạt kiến thức và hướng dẫn em trong quá trình học tập tại trường.

Bên cạnh đó, em cũng xin chân thành cảm ơn sự hướng dẫn và chỉ bảo tận tình của thầy Đặng Trọng Hợp – người đã luôn quan tâm, chỉ bảo và tạo điều kiện cho em trong suốt quá trình thực tập tốt nghiệp và quá trình thực hiện đề tài. Các định hướng và sự hiểu biết về khoa học của thầy chính là tiền đề để em có thể hoàn thành được đề tài này một cách tốt nhất.

Tuy còn một số hạn chế nhất định về chuyên môn và thời gian nên bài báo cáo và ứng dụng còn nhiều hạn chế. Kính mong quý thầy cô góp ý để đề tài của em hoàn thiện hơn.

Em xin chân thành cảm ơn!

Sinh viên thực hiện

Nguyễn Hải Triều

# MỤC LỤC

[LỜI CẢM ƠN 1](#_Toc166571882)

[MỤC LỤC 2](#_Toc166571883)

[DANH MỤC CÁC KÍ HIỆU VÀ CÁC CHỮ VIẾT TẮT 5](#_Toc166571884)

[DANH MỤC CÁC BẢNG 6](#_Toc166571885)

[DANH SÁCH CÁC HÌNH VẼ, ĐỒ THỊ 7](#_Toc166571886)

[CHƯƠNG 1: TỔNG QUAN 9](#_Toc166571887)

[1.1 Lý do chọn đề tài 9](#_Toc166571888)

[1.2 Mục đích 10](#_Toc166571889)

[1.3 Đối tượng và phạm vi nghiên cứu 10](#_Toc166571890)

[1.4 Ý nghĩa khoa học và thực tiễn của đề tài 10](#_Toc166571891)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT 12](#_Toc166571892)

[2.1 Mô tả bài toán 12](#_Toc166571893)

[2.2 Mục đích dự án 12](#_Toc166571894)

[2.2.1 Đối với nhà tuyển dụng doanh nghiệp 12](#_Toc166571895)

[2.2.2 Đối với người tìm kiếm việc làm 12](#_Toc166571896)

[2.3 Các giải pháp công nghệ 12](#_Toc166571897)

[2.3.1 Antd Design 12](#_Toc166571898)

[2.3.2 ReactJS 14](#_Toc166571899)

[2.3.3 NestJS 20](#_Toc166571900)

[2.3.4 MongoDB 23](#_Toc166571901)

[CHƯƠNG 3: Phân tích và thiết kế hệ thống 25](#_Toc166571902)

[3.1 Tiến trình hoàn thành dự án 25](#_Toc166571903)

[3.2 Khảo sát 26](#_Toc166571904)

[3.3 Thu nhập thông tin 27](#_Toc166571905)

[3.3.1 Nhu cầu doanh nghiệp 27](#_Toc166571906)

[3.3.2 Nhu cầu người dùng 28](#_Toc166571907)

[3.4 Phân tích thiết kế hệ thống 30](#_Toc166571908)

[3.4.1 Use case tổng quát 30](#_Toc166571909)

[3.1.2 Use case ứng viên 31](#_Toc166571910)

[3.1.3 Use case quản trị 32](#_Toc166571911)

[3.1.4 Use case HR 33](#_Toc166571912)

[3.5 Xây dựng cơ sở dữ liệu 34](#_Toc166571913)

[CHƯƠNG 4: KẾT QUẢ THỰC HIỆN 34](#_Toc166571914)

[4.1 Môi trường cài đặt và công cụ 34](#_Toc166571915)

[4.1.1 Visual Studio Code 34](#_Toc166571916)

[4.1.2 MongoDB Compass 34](#_Toc166571917)

[4.2 Giao diện người dùng 35](#_Toc166571918)

[4.2.1 Trang chủ 35](#_Toc166571919)

[4.2.2 Xem việc làm 36](#_Toc166571920)

[4.2.3 Xem chi tiết việc làm 36](#_Toc166571921)

[4.2.4 Xem công ty 37](#_Toc166571922)

[4.2.5 Xem chi tiết công ty 37](#_Toc166571923)

[4.2.6 Xem lịch sử rải CV 38](#_Toc166571924)

[4.2.7 Nhận job thông qua Gmail 38](#_Toc166571925)

[4.2.8 Cập nhật thông tin cá nhân 39](#_Toc166571926)

[4.2.9 Nộp CV 39](#_Toc166571927)

[4.2.10 Đăng ký tài khoản 40](#_Toc166571928)

[4.2.11 Đăng nhập 40](#_Toc166571929)

[4.3 Giao diện Admin 41](#_Toc166571930)

[4.3.1 Đăng nhập Admin 41](#_Toc166571931)

[4.3.3 Quản lý tài khoản 41](#_Toc166571932)

[4.3.4 Thống kê 42](#_Toc166571933)

[4.3.5 Quản lý việc làm 43](#_Toc166571934)

[4.3.6 Quản lý CV ứng viên 44](#_Toc166571935)

[4.3.7 Quản lý quyền hạn người dùng 44](#_Toc166571936)

[4.3.8 Quản lý Role 45](#_Toc166571937)

[4.4 Giao diện HR 46](#_Toc166571938)

[4.4.1 Đăng nhập HR 46](#_Toc166571939)

[4.4.2 Quản lý CV ứng viên 47](#_Toc166571940)

[4.4.3 Quản lý việc làm 48](#_Toc166571941)

[KẾT LUẬN 49](#_Toc166571942)

[TÀI LIỆU THAM KHẢO 50](#_Toc166571943)

# DANH MỤC CÁC KÍ HIỆU VÀ CÁC CHỮ VIẾT TẮT

|  |  |
| --- | --- |
| Từ viết tắt | Giải thích |
| HR | Human Resource |
| CV | Resume |
| Antd | Ant Design |
| HTML | Hypertext Markup Language |
| CSS | Cascading Style Sheets |
| UED | User Experience Design |
| IOS | iPhone Operating System |
| DOM | Document Object Model |
| UI | User Interface |
| SEO | Search Engine Optimization |
| JSX | JavaScript XML |
| DI | Dependencies Injection |
| HTTP | Hypertext Transfer Protocol |
| NoSQL | Not Only SQL |
| JSON | JavaScript Object Notation |
| RDBMS | Relational Database Management System |
| RAM | Random Access Memory |
| VS | Visual Studio |
| GUI | Graphical User Interface |
| SQL | Structured Query Language |
| DB | Database |

# DANH MỤC CÁC BẢNG

[Bảng 1: Bảng so sánh ReactJS và Angular 16](#_Toc166571720)

[Bảng 2: Bảng so sánh ReactJS và VueJS 17](#_Toc166571721)

[Bảng 3: Bảng so sánh NoSQL và SQL 22](#_Toc166571722)

[Bảng 4: Bảng tiến trình hoàn thành dự án 23](#_Toc166571723)

# DANH SÁCH CÁC HÌNH VẼ, ĐỒ THỊ

[Hinh 1:Tìm hiểu về Antd Design 13](#_Toc166572073)

[Hinh 2: Tìm hiểu về thư viện ReactJS 15](#_Toc166572074)

[Hinh 3:Biểu đồ mức độ tải xuống của ReactJS 17](#_Toc166572075)

[Hinh 4: So sánh ReactJs và Angular 18](#_Toc166572076)

[Hinh 5: So sảnh ReactJS và VueJS 19](#_Toc166572077)

[Hinh 6: Tìm hiểu về Framework NestJS 20](#_Toc166572078)

[Hinh 7: Sơ dồ hoạt động của NestJS 21](#_Toc166572079)

[Hinh 8: Hình ảnh minh họa độ nối tiếng của NestJS 22](#_Toc166572080)

[Hinh 9: Tìm hiểu về NoSQL MongoDB 23](#_Toc166572081)

[Hinh 10: Sơ đồ use case tổng quát 30](#_Toc166572082)

[Hinh 11: Sơ đồ use case ứng viên 31](#_Toc166572083)

[Hinh 12: Sơ đồ use case người quản trị 32](#_Toc166572084)

[Hinh 13: Sơ đồ use case HR 33](#_Toc166572085)

[Hinh 14: Biều đồ thiết kế cơ sở dữ liệu 34](#_Toc166572086)

[Hinh 15: Giao diện trang chủ 36](#_Toc166572087)

[Hinh 16: Xem danh sách các công việc 37](#_Toc166572088)

[Hinh 17: Xem chi tiết công việc 37](#_Toc166572089)

[Hinh 18: Xem danh sach các công ty 38](#_Toc166572090)

[Hinh 19: Xem chi tiết công ty 38](#_Toc166572091)

[Hinh 20: Xem lịch sử ứng tuyển 39](#_Toc166572092)

[Hinh 21: Nhận việc làm qua Gmail 39](#_Toc166572093)

[Hinh 22: Cập nhật thông tin cá nhân 40](#_Toc166572094)

[Hinh 23: Ứng tuyển việc làm 40](#_Toc166572095)

[Hinh 24: Giao diện đăng kí tài khoản 41](#_Toc166572096)

[Hinh 25: Giao diện đăng nhập tài khoản 41](#_Toc166572097)

[Hinh 26: Giao diện đăng nhập người quản trị 42](#_Toc166572098)

[Hinh 27: Quản lý người dùng 42](#_Toc166572099)

[Hinh 28: Thêm mới người dùng 43](#_Toc166572100)

[Hinh 29: Thống kê 43](#_Toc166572101)

[Hinh 30: Quản lý việc làm 44](#_Toc166572102)

[Hinh 31: Thêm mới việc làm 44](#_Toc166572103)

[Hinh 32: Quản lý CV ứng viên 45](#_Toc166572104)

[Hinh 33: Quản lý quyền hạn người dùng 45](#_Toc166572105)

[Hinh 34: Thêm mới quyền hạn người dùng 46](#_Toc166572106)

[Hinh 35: Quản lý quyền 46](#_Toc166572107)

[Hinh 36: Thêm mối quyền 47](#_Toc166572108)

[Hinh 37: Giao diện đăng nhập HR 47](#_Toc166572109)

[Hinh 38: Quản lý CV ứng viên 48](#_Toc166572110)

[Hinh 39: Cập nhật trạng thái CV ứng viên 48](#_Toc166572111)

[Hinh 40: Quản lý việc làm 49](#_Toc166572112)

[Hinh 41: Thêm mới việc làm 49](#_Toc166572113)

# CHƯƠNG 1: TỔNG QUAN

## Lý do chọn đề tài

Ngày nay, các công nghệ tiên tiến phát triển ngày càng mạnh mẽ và được ứng dụng ngày càng nhiều vào các lĩnh vực kinh tế, sản xuất cũng như đời sống của con người. Công nghệ thông tin đã được áp dụng rất nhiều vào đời sống hàng ngày của chúng ta. Nhờ đó, các công việc được thực hiện một cách nhanh chóng, dễ dàng và mang lại hiệu quả rất cao.

Việc ra đời website đăng tin tuyển dụng việc làm trực tuyến mang lại giá trị rất lớn cho các doanh nghiệp nó rủ ngắn thời gian để các doanh nghiệp có thể tiếp cận được cấc ứng viên tiềm năng. Với một website trực tuyến, cá nhân hoặc doanh nghiệp có thể tiếp cận một cách dễ dàng. Ứng viên có thể tìm kiếm việc làm một cách nhanh chóng và ứng tuyển vào các doanh nghiệp mà ứng viên có đủ kĩ năng.

Một website đăng tin tuyển dụng việc làm mang lại hiệu quả rất cao cho cả doanh nghiệp và cá nhân.Doanh nghiệp có thể đăng tin tuyển dụng việc làm một cách dễ dàng, ứng việc có thế tìm kiếm việc làm phù hợp với kĩ năng của bản thân nhanh chóng. Ngoài ra, việc đăng tin tuyển dụng trực tuyến giúp các doanh nghiệp tiếp kiệm chi phí cho mỗi lần tuyển dụng nhân sự, ứng viên có thể tìm kiếm và ứng tuyển công việc nhanh.

Một website đăng tịn tuyển dụng còn giúp doanh nghiệp giới thiệu về công ty và còn là cơ hội để các ứng viên tìm việc làm có thể dễ dàng tiếp cận với các công việc mà các công ty đang tuyển dụng. Doanh nghiệp có thể xây dựng hình ảnh thương hiệu chuyên nghiệp, quy mô doanh nghiệp nhằm thu hút các ứng viên ứng tuyển vào doanh nghiệp.

Tóm lại, website đăng tịn tuyển dụng việc làm giúp các doanh nghiệp loại bỏ các bước không cần thiết trong việc tuyển dụng nhân sư cũng tang chất lượng tuyển dụng khi có thể chọn lọc các ứng viên online, giúp các ứng viên có thể tìm kiếm và ứng tuyển vào các vị trí công việc một các nhanh chóng.

## Mục đích

Việc xây dựng website đăng tin tuyển dụng việc làm cấc nhà tuyển dụng có thể đăng tin tuyển dụng việc làm một cách nhanh chóng và hiệu quả. Ngoài ra, website đăng tin tuyển dụng việc làm cũng là nơi để các ứng viên tìm việc làm có thể tìm kiếm được nhưng công việc phù hợp với kĩ năng của bản thân để ứng tuyển vào công việc đó.

Website này cũng giúp em áp dụng những kiến thức học tập được trên trường vào thực tế. Từ đó sẽ tích lũy thêm nhiều kinh nghiệm cho bản thân.

## Đối tượng và phạm vi nghiên cứu

* Đối tượng nghiên cứu: Những người có nhu cầu tìm kiếm việc làm và các doanh nghiệp muốn tìm kiếm nguồn nhân lực cho doanh nghiệp…
* Phạm vi nghiên cứu: Xây dựng một website đăng tin tuyển dụng giúp các ứng viên có thể xem thông tin các công việc có trên website và lựa chọn các công ty phù hợp với kĩ năng của bản thân để ứng tuyển. Người nhân sự thuộc doanh nghiệp có thể tiếp nhận các đơn ứng tuyển của các ứng viên và quản lý đơn xin việc và đăng tin tuyển dụng các công việc của công ty họ.

## Ý nghĩa khoa học và thực tiễn của đề tài

Ý nghĩa khoa học: Việc thực hiện đề tài giúp đóng góp vào việc hiểu rõ hơn về quy trình xây dựng, triển khai và vận hành website đăng tin tuyển dụng việc làm một cách nhanh chóng, bao gồm cả các yếu tố kỹ thuật, thiết kế giao diện, trải nghiệm người dùng, quản lý dữ liệu và giao dịch, bảo mật thông tin, quản lý đơn ứng tuyển việc làm, quản lý các công việc tuyển dụng, tích hợp việc trao đổi trực tuyến giũa nhà tuyển dụng và các ứng viên ứng tuyển vào doanh nghiệp. Ngoài ra, nó giúp tìm hiểu và phân tích các xu hướng việc việc . Nghiên cứu này có thể cung cấp thông tin giá trị về nhu cầu, kí năng của mỗi ứng việc cần có để có thể ứng tuyển vào công việc của doanh nghiệp.

Ý nghĩa thực tiễn: Tạo ra một công cụ hiệu quả giúp cá nhân tiếp cận có thể tiếp cận được các công việc của họ và các kĩ năng doanh nghiệp cần có để có thể ứng tuyển vào vị trí công việc , giúp doanh nghiệp có thể đăng tin tuyển dụng các công việc, việc làm một cách nhanh chóng. Nghiên cứu này có thể cung cấp những kiến thức, kỹ năng và kinh nghiệm cần thiết để cá nhân hoặc doanh nghiệp xây dựng và quản lý website đăng tin tuyển dụng việc làm một cách chuyên nghiệp, từ đó tăng cường khả năng cạnh tranh và phát triển các nền tảng đăng tin tuyển dụng việc làm. Nó giúp nâng cao chất lượng tuyển dụng của doanh nghiệp , từ đó giúp doanh nghiệp có thẻ tiếp cận các ứng viên tiềm năng cho vị trí tuyển dụng, giúp người tìm kiếm việc làm có một nói để tìm kiếm các công việc phù hợp với kĩ năng.

# CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

## 2.1 Mô tả bài toán

Xây dựng website đăng tin tuyển dụng việc làm giúp việc quản lý đăng tin tuyển dụng thuộc doanh ngiệp cũng như người quản lý có thể quản lý công việc họ đăng tuyển, các CV các ứng viên ứng tuyển một cách thuận tiện, nhanh chóng, dễ dàng. Giúp cho người tìm việc làm có dễ dạng lựa chọn công việc cho bản thân một cách dễ dàng mà không cần phải dành thời gian để đến doanh nghiệp, giúp tiết kiệm được thời gian, công sức. Ngoài ra, website đăng tin tuyển dụng cũng giúp các doanh nghiệp có thể tiếp cận với nhiều nhà ứng tuyển nhằm nâng cao chất lượng tuyển dụng.

## 2.2 Mục đích dự án

### 2.2.1 Đối với nhà tuyển dụng doanh nghiệp

Việc xây dựng website đăng tin tuyển dụng việc làm giúp doanh nghiệp có thể mở rộng thị trường, tiếp cận nhiều ứng viên ứng tuyển nhằm năng cao chất lượng tuyển dụng, quản lý dễ dàng công việc và đơn xin việc của nhiều ứng viên.

### 2.2.2 Đối với người tìm kiếm việc làm

Việc xây dựng website đăng tin tuyển dụng việc làm giúp các người tìm kiếm việc làm dễ dàng có thể tim kiếm các công việc và ứng tuyển vào vị trí đó, tiết kiệm thời gian và công sức để có thể tìm kiếm được một công việc phù hợp với bản thân.

## 2.3 Các giải pháp công nghệ

### 2.3.1 Antd Design

Hiểu một cách đơn giản, Antd Design là một Framework bao gồm 3 thành phần cơ bản đó là: HTML, CSS và JavaScript. Framework này được sử dụng để phát triển trang web chuẩn theo Responsive và sử dụng các component có sắn mà Antd design đã viết sẵn. Sử dụng Antd Design giúp việc thiết kế web tiết kiệm nhiều thời gian và đơn giản hơn.



Hinh 1:Tìm hiểu về Antd Design

Trong quá trình phát triển các ứng dụng trên máy tính, có nhiều thông số thiết kế và hệ thống xử lý khác nhau rất phức tạp, điều này gây khó khăn gấp đôi cho các nhà thiết kế và lập trình viên, đồng thời sẽ làm giảm hiệu quả cho việc phát triển này. Sau khi có cho mình các dự án lớn và sơ lược lại, đội ngũ Ant UED đã cho ra mắt ngôn ngữ thiết kế Ant Design dành cho giao diện của ứng dụng, nhằm mục đích nhất quán các thông số giao diện người dùng trên nền tảng dự án nội bộ, giảm chi phí không cần thiết của việc triển khai ý tưởng và cung cấp nhiều tài nguyên cho việc thiết kế và lập trình front-end.

Ant Design đặc biệt được tạo ra cho các ứng dụng máy tính nội bộ, cam kết trải nghiệm của người dùng và các nhà thiết kế sản phẩm. Những người thiết kế giao diện và thiết kế trải nghiệm cho người dùng được gọi chung là thiết kế sản phẩm, Ant Design sẽ xóa bỏ ranh giới giữa việc quản lý sản phẩm, thiết kế tương tác, thiết kế hình ảnh, phát triển giao diện và quản trị dữ liệu người dùng. Nhờ việc tận dụng sự đồng nhất về đặc điểm kỹ thuật, Ant Design đơn giản hóa việc thiết kế và lập trình cho những người đang thực hiện Project, điều này sẽ giúp họ có thêm nhiều kinh nghiệm và tăng hiệu quả phát triển giao diện của sản phẩm và ứng dụng.

Ant hiện đang có hơn 51k star trên Github (Nguồn: Tham khảo số lượt tải trên GitHub)

Có thể coi Ant Design cho React là tập hợp của hầu hết các thư viện về React. Nó đáp ứng được hầu hết các yêu cầu của project của bạn mà ban không phải cài thêm bất cứ thư viện nào nữa. Dưới đây là danh sách các component mà nó cung cấp:

* General: Button, Icon
* Layout: Grid, Layout
* Navigation: Affix, Breadcrumb, Dropdown, Menu, Pagination, Steps
* Data Entry: AutoComplete, Checkbox, Cascader, DatePicker, Form, InputNumber, Input, Mention, Rate, Radio, Switch, Slider, Select, TreeSelect, Transfer, TimePicker, Upload
* Data Display: Avatar, Badge, Collapse, Carousel, Card, Calendar, List, Popover, Tree, Tooltip, Timeline, Tag, Tabs, Table
* Feedback: Alert, Drawer, Modal, Message, Notification, Progress, Popconfirm, Spin, Skeleton
* Other: Anchor, BackTop, Divider, LocaleProvider

### 2.3.2 ReactJS

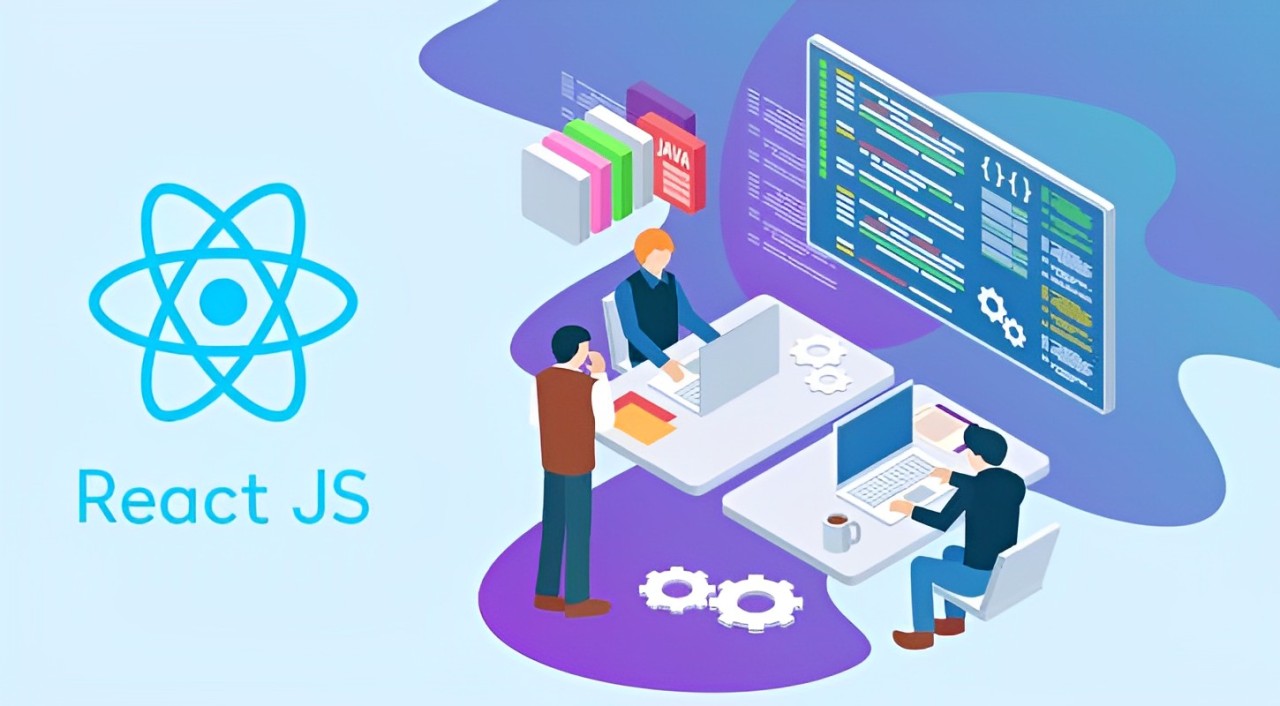
#### 2.3.2.1 Reactjs là gì?

ReactJS được phát triển bởi Facebook và được giới thiệu lần đầu tiên vào năm 2011. Ban đầu, ReactJS được phát triển để xây dựng giao diện người dùng trên trang web Facebook, nhằm cải thiện tốc độ và hiệu suất của ứng dụng web.

Tuy nhiên, ReactJS không được công bố cho cộng đồng phát triển cho đến năm 2013, khi Facebook công bố mã nguồn mở của nó và giới thiệu cho cộng đồng lập trình viên. Từ đó, ReactJS nhanh chóng trở thành một trong những thư viện phổ biến nhất để phát triển các ứng dụng web động.

Sau đó, vào năm 2015, Facebook giới thiệu phiên bản React Native, một framework phát triển ứng dụng di động sử dụng ReactJS. React Native cho phép các nhà phát triển xây dựng các ứng dụng di động cho cả iOS và Android sử dụng cùng một mã nguồn, tương tự như ReactJS trên web.

Hiện nay, ReactJS đã trở thành một trong những thư viện phát triển web phổ biến nhất, được sử dụng rộng rãi bởi các công ty lớn và nhỏ trên toàn thế giới. Facebook cũng tiếp tục đầu tư phát triển và nâng cấp ReactJS để đáp ứng nhu cầu của cộng đồng phát triển.



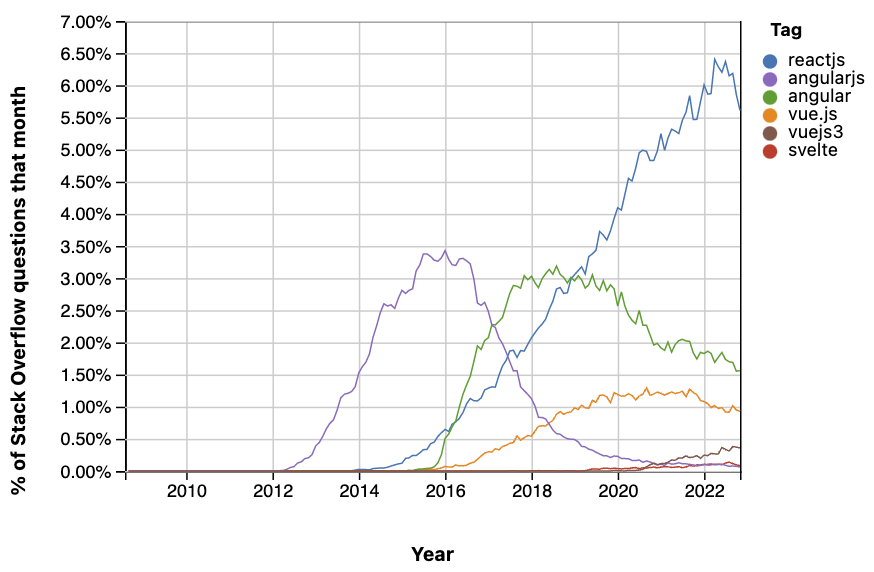
Hinh 2: Tìm hiểu về thư viện ReactJS

#### 2.3.2.3 Các tính năng nổi bật

ReactJS có nhiều tính năng hữu ích cho việc phát triển ứng dụng web, bao gồm:

* Components: ReactJS cho phép phát triển ứng dụng web theo mô hình component. Các component là các phần tử UI độc lập có thể được tái sử dụng trong nhiều phần khác nhau của ứng dụng.
* Virtual DOM: ReactJS sử dụng Virtual DOM để tối ưu hóa hiệu suất của ứng dụng. Virtual DOM là một bản sao của DOM được lưu trữ trong bộ nhớ và được cập nhật một cách nhanh chóng khi có thay đổi, giúp tăng tốc độ và hiệu suất của ứng dụng.
* JSX: JSX là một ngôn ngữ lập trình phân biệt được sử dụng trong ReactJS để mô tả các thành phần UI. JSX kết hợp HTML và JavaScript, giúp cho việc viết mã dễ hiểu và dễ bảo trì hơn.
* State và Props: ReactJS cho phép quản lý trạng thái của các thành phần UI thông qua State và Props. State là trạng thái của một thành phần được quản lý bởi nó chính, trong khi Props là các giá trị được truyền vào từ bên ngoài để tùy chỉnh hoặc điều khiển hành vi của một thành phần.
* Hỗ trợ tốt cho SEO: ReactJS hỗ trợ tốt cho việc tối ưu hóa SEO. Với các thư viện như React Helmet, các nhà phát triển có thể quản lý các phần tử meta và title cho từng trang web, giúp tăng khả năng tìm kiếm và tăng cường trải nghiệm người dùng.
* Hỗ trợ đa nền tảng: ReactJS không chỉ được sử dụng để phát triển ứng dụng web, mà còn được sử dụng để phát triển ứng dụng di động với React Native. Sử dụng React Native, các nhà phát triển có thể xây dựng ứng dụng di động cho cả iOS và Android sử dụng cùng một mã nguồn.
* Redux: Redux là một thư viện quản lý trạng thái cho các ứng dụng ReactJS. Nó giúp quản lý trạng thái của ứng dụng một cách chính xác và dễ dàng, đồng thời giúp tăng tính linh hoạt và khả năng mở rộng của ứng dụng.

#### 2.3.2.4 So sánh ReactJS với các Framework nổi tiếng hiện này

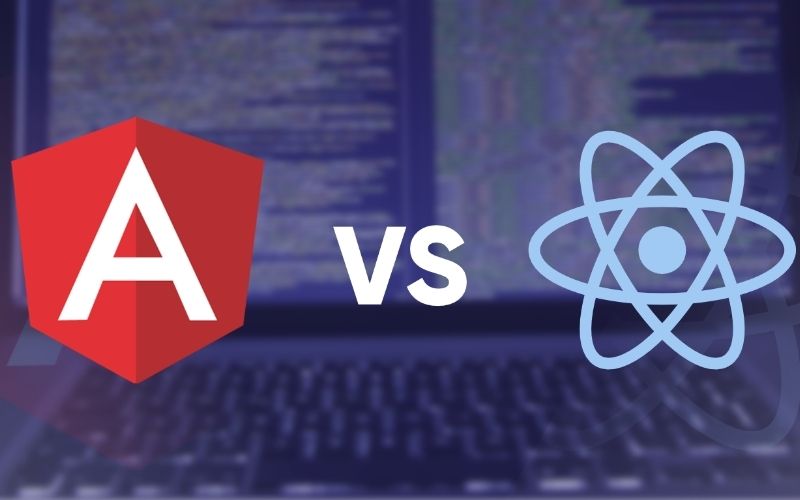


Nguồn:Cuộc khảo sát trên Stackoverflow về độ phổ biến của ReactJS

Hinh 3:Biểu đồ mức độ tải xuống của ReactJS

Tình đến năm 2022 theo bình trọn của rất nhiều lập trình viên trên thế giới thì ReactJS đang là thư viên có lượt tải và sử dụng cao nhất.

##### **2.3.2.4.1 ReactJS và Angular**



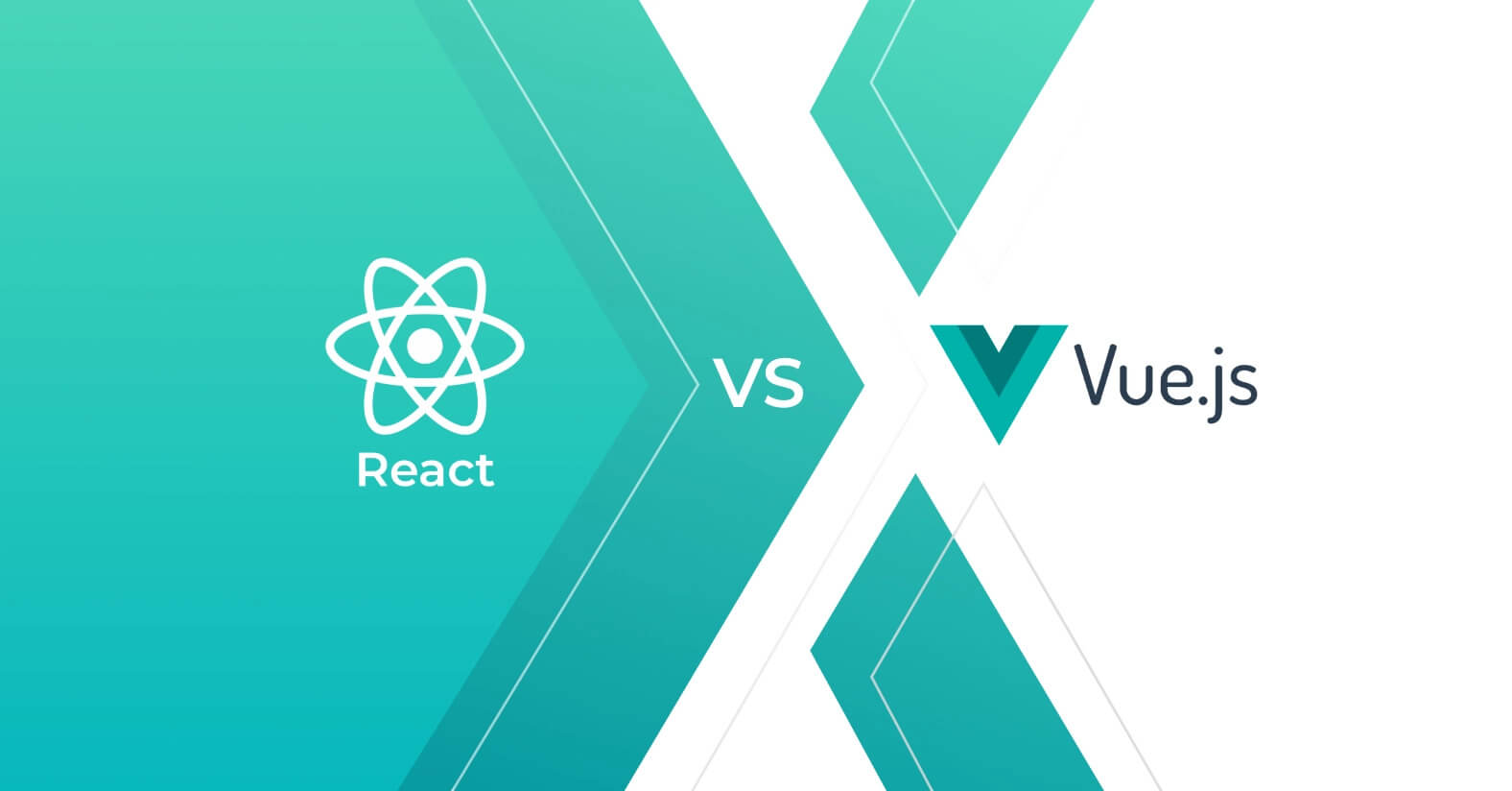
Hinh 4: So sánh ReactJs và Angular

|  |  |  |
| --- | --- | --- |
|  | ReactJS | Angular |
| Người sáng lập | Jordan Walke | Misko Hevery |
| Năm ra mắt | 2013 | 2009 |
| DOM | DOM ảo | DOM thật |
| Kích thước ứng dụng | Tướng đối nhỏ | Tương đối nhỏ |
| Hiệu suất | Cao | Cao |
| Chi phí | Mã nguồn mở | Mã nguồn mở |
| Mục đích | Tạo các ứng dụng web với dữ liệu thường xuyên thay đổi | Tạo các ứng dụng web có hiệu suất cao |
| Dữ liệu | One-way binding | Two-way binding |
| Mức độ phổ biến | Rất cao | Trung bình |

Nguồn: Cuộc thảo luận trên Stackoverflow

Bảng 1: Bảng so sánh ReactJS và Angular

##### **22.3.2.4.2 ReactJS và VueJS**



Hinh 5: So sảnh ReactJS và VueJS

|  |  |  |
| --- | --- | --- |
|  | ReactJS | VueJS |
| Người sáng lập | Jordan Walke | Evan You |
| Năm ra mắt | 2013 | 2014 |
| DOM | DOM ảo | DOM ảo |
| Kích thước ứng dụng | Tướng đối nhỏ | Tương đối nhỏ |
| Hiệu suất | Cao | Cao |
| Chi phí | Mã nguồn mở | Mã nguồn mở |
| Mục đích | Tạo các ứng dụng web với dữ liệu thường xuyên thay đổi | Tạo các ứng dụng web có hiệu suất cao |
| Dữ liệu | One-way binding | Two-way binding |
| Mức độ phổ biến | Rất cao | Trung bình |

Nguồn: Cuộc thảo luận trên Stackoverflow

Bảng 2: Bảng so sánh ReactJS và VueJS

### 2.3.3 NestJS

#### 2.3.3.1 NestJS là gì?

NestJS là một framework mã nguồn mở để phát triển ứng dụng server-side (backend applications) bằng ngôn ngữ TypeScript hoặc JavaScript. Nó được xây dựng trên cơ sở của Node.js và sử dụng các khái niệm từ TypeScript để tạo ra một môi trường phát triển hiện đại và mạnh mẽ cho việc xây dựng các ứng dụng web và API.

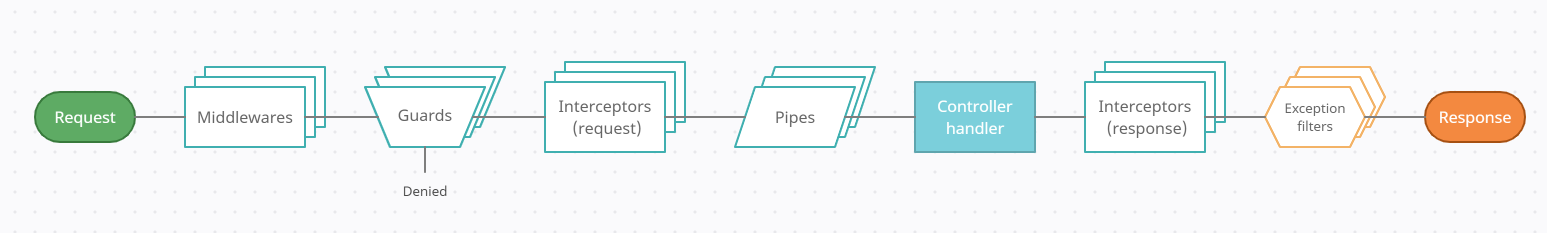
Mục tiêu chính của NestJS là cung cấp một cấu trúc ứng dụng rõ ràng và dễ quản lý, giúp tăng tính bảo trì và sự tổ chức trong mã nguồn. Để đạt được điều này, NestJS triển khai mô hình kiến trúc lõi (core architecture) dựa trên các nguyên tắc của Angular, đặc biệt là sử dụng Dependency Injection (DI) và Modules (Các module).



Hinh 6: Tìm hiểu về Framework NestJS

#### 2.3.3.2 Cấu trúc NestJS

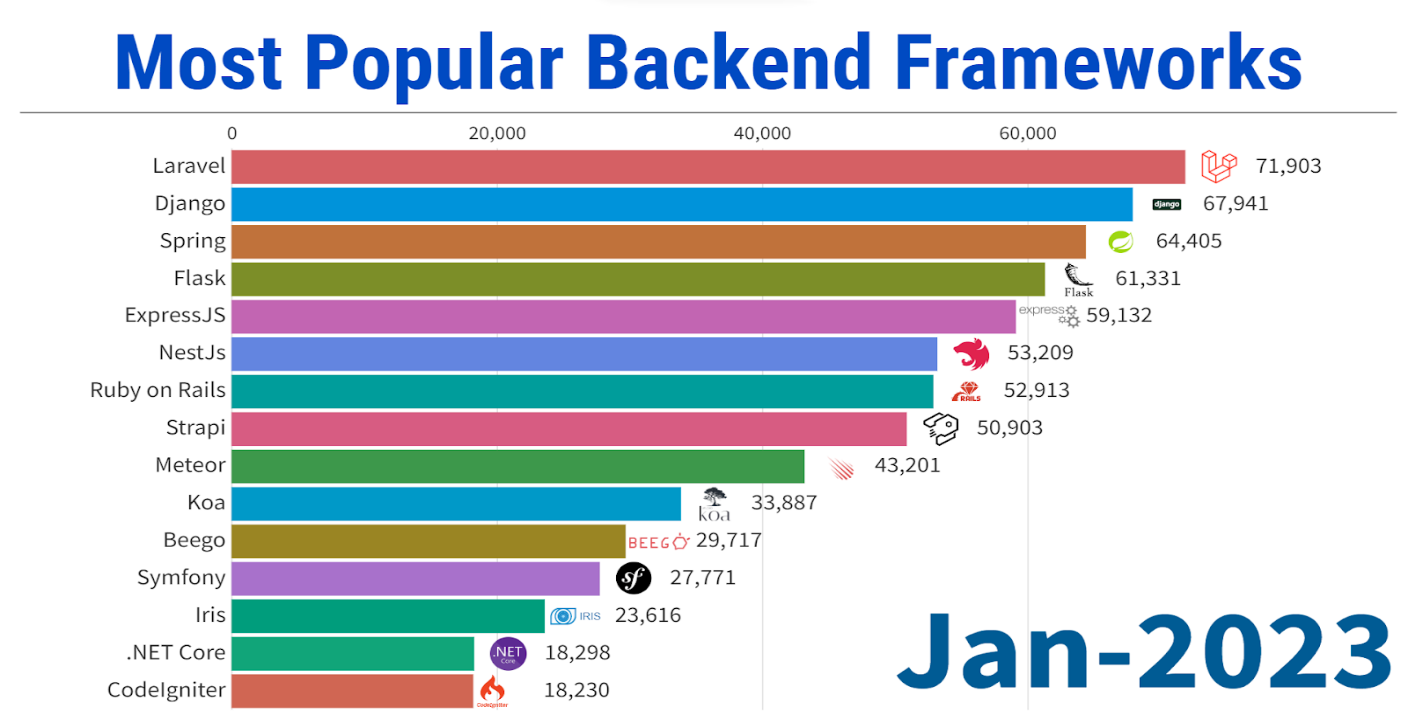
* Cấu trúc của NestJS được xây dựng dựa trên mô hình kiến trúc lõi (core architecture) giúp tạo ra một ứng dụng server-side (backend application) rõ ràng, dễ quản lý và dễ mở rộng. Cấu trúc NestJS thường được tổ chức thành các phần chính sau:
* Module (Các module): Module là một phần cơ bản trong cấu trúc NestJS. Mỗi ứng dụng NestJS bao gồm ít nhất một module gốc (root module) và có thể có nhiều module con. Module là nơi tổ chức các thành phần của ứng dụng như Controllers, Providers và các thành phần khác. Mỗi module đại diện cho một phần chức năng cụ thể của ứng dụng.
* Controller (Bộ điều khiển): Controllers là thành phần chịu trách nhiệm xử lý các yêu cầu HTTP từ phía client và trả về kết quả tương ứng. Controllers là nơi xử lý các request và trả về các response. Các phương thức của controller được chú thích (decorated) bằng các decorator như `@Get()`, `@Post()`, `@Put()`, v.v., để chỉ định các route và phương thức HTTP tương ứng.
* Provider (Các nhà cung cấp): Providers là thành phần chịu trách nhiệm cung cấp các dịch vụ cho ứng dụng. Đây có thể là các service, repository, logger, v.v. Providers sử dụng dependency injection để chèn vào các thành phần khác và có thể được sử dụng bởi các controllers hoặc các providers khác.



Hinh 7: Sơ dồ hoạt động của NestJS

* Middleware (Trung gian): Middleware là các hàm xử lý mà NestJS sử dụng để xử lý các yêu cầu HTTP trước khi chúng đến các route xử lý chính. Middleware có thể được sử dụng để thực hiện các thao tác chung như xác thực, ghi log, xử lý lỗi, v.v.
* Filter (Bộ lọc): Filters được sử dụng để xử lý các exception (ngoại lệ) xảy ra trong ứng dụng. Filters cho phép bạn xử lý và thay đổi response trước khi gửi về client khi có exception xảy ra.
* Guard (Bảo vệ): Guards được sử dụng để kiểm tra xem một yêu cầu có thể được xử lý hoặc không. Guards cho phép bạn thực hiện các kiểm tra xác thực hoặc kiểm tra quyền trước khi xử lý một yêu cầu.
* Interceptor (Bộ chặn): Interceptors là các hàm xử lý mà NestJS sử dụng để chặn và thay đổi response trước khi nó được gửi về client. Interceptors có thể được sử dụng để thực hiện các thao tác chung trên response trước khi nó đi ra ngoài.
* Exception (Ngoại lệ): Exception handling (xử lý ngoại lệ) là một phần quan trọng của cấu trúc NestJS. Exception handling cho phép bạn xử lý các exception xảy ra trong ứng dụng và trả về các thông báo lỗi thích hợp cho client.

#### 2.3.3.3 So sánh NestJS với các Framework khác



Nguồn:Cuộc khảo sat trên Github năm 2023

Hinh 8: Hình ảnh minh họa độ nối tiếng của NestJS

Tính đến Tháng 1 năm 2023 dựa trên bình chọn trên GitHub NestJS đang có độ phổ biến và lượt tải rất cao.

### 2.3.4 MongoDB

#### 2.3.4.1 Lịch sử ra đời?

MongoDB lần đầu ra đời bởi MongoDB Inc., tại thời điểm đó là thế hệ 10, vào tháng Mười năm 2007, nó là một phần của sản phẩm PaaS (Platform as a Service) tương tự như Windows Azure và Google App Engine. Sau đó nó đã được chuyển thành nguồn mở từ năm 2009.

MongoDB đã trở thành một trong những NoSQL database nổi trội nhất bấy giờ, được dùng làm backend cho rất nhiều website như eBay, SourceForge và The New York Times.

#### 2.3.4.2 MongoDB là gì?



Hinh 9: Tìm hiểu về NoSQL MongoDB

MongoDB là một hệ quản trị cơ sở dữ liệu mã nguồn mở, là CSDL thuộc NoSql và được hàng triệu người sử dụng.

MongoDB là một database hướng tài liệu (document), các dữ liệu được lưu trữ trong document kiểu JSON thay vì dạng bảng như CSDL quan hệ nên truy vấn sẽ rất nhanh.

Với CSDL quan hệ chúng ta có khái niệm bảng, các cơ sở dữ liệu quan hệ (như MySQL hay SQL Server...) sử dụng các bảng để lưu dữ liệu thì với MongoDB chúng ta sẽ dùng khái niệm là collection thay vì bảng

So với RDBMS thì trong MongoDB collection ứng với table, còn document sẽ ứng với row, MongoDB sẽ dùng các document thay cho row trong RDBMS.

Các collection trong MongoDB được cấu trúc rất linh hoạt, cho phép các dữ liệu lưu trữ không cần tuân theo một cấu trúc nhất định.

Thông tin liên quan được lưu trữ cùng nhau để truy cập truy vấn nhanh thông qua ngôn ngữ truy vấn MongoDB.

#### 2.3.4.3 So sánh NoSQL và SQL

|  |  |  |
| --- | --- | --- |
| Tiêu chí | SQL | NoSQL |
| Mô hình dữ liệu | Quan hệ (Relational) | Phi quan hệ(Non-Relational) |
| Ngôn ngữ truy vấn | Sử dụng SQL | Sử dụng các ngôn ngữ truy vấn khác nhau tùy thuộc vào cơ sở dữ liệu |
| Khả năng mở rộng | Chủ yếu mở rộng theo chiểu dọc | Chủ yếu mở rộng theo chiều ngang |
| Hiệu suất | Hiệu suất có thể bị ảnh hưởng nếu xử lý khối lượng dữ liệu lớn | Hiệu suất xử lý cao khi xử lý khối lượng dữ liệu lớn |
| Ứng dụng | Trong các dự án yêu cầu tính cấu trúc cao | Trong các dự án không yêu càu cấu trúc cao |
| Các cớ sở dư liệu nổi tiếng | MySQL,PostgreSQL,SQL Server,Oracle | MongoDB,Cassandra |

Nguồn: Tham khảo câu trả lời trên Github và Stackoverflow

Bảng 3: Bảng so sánh NoSQL và SQL

# CHƯƠNG 3: Phân tích và thiết kế hệ thống

## 3.1 Tiến trình hoàn thành dự án

Bảng 4: Bảng tiến trình hoàn thành dự án

|  |  |
| --- | --- |
| **Tiến trình** | **Mô tả** |
| Khởi tạo dự án | Tìm hiểu các tài liệu liên quan đến dự án.  Tìm hiểu cách thức hoạt động của việc bán hàng và giao  hàng trên mạng.  Xem xét tính khả thi của dự án, các khía cạnh liên quan  đến dự án. |
| Lập kế hoạch | Mô tả tổng quát quá trình thực hiện dự án.  Ràng buộc dự án với các điều kiện chủ quan và khách  quan. Xác định tính khả thi của dự án.  Xác định rủi ro và các vấn đề phát sinh trong quá trình  xây dựng dự án có thể gặp.  Lập kế hoạch hệ thống mới. |
| Thu thập yêu cầu | Thu thập yêu cầu của khách hàng đối với hệ thống một  cách chi tiết và chính xác.  Khảo sát, tìm hiểu các hệ thống đã có thể đưa ra các yêu  cầu, các chức năng cho hệ thống của mình. |
| Phân tích và kiểm tra  yêu cầu | Phân tích các yêu cầu chức năng và phi chức năng.  Kiểm tra tính khả thi của yêu cầu.  Bổ sung các yêu cầu còn thiếu sót. |
| Phân tích | Mô hình hóa các yêu cầu hệ thống và đặc tả các chức  năng. |
| Thiết kế | Thiết kế cơ sở dữ liệu. Thiết kế giao diện. |
| Kiểm thử | Kiểm thử cho từng giai đoạn của dự án từ lúc bắt đầu đến lúc kết thúc dự án. |
| Cài đặt | Xây dựng website cho kế hoạch đã đặt ra.  Viết tài liệu hướng dẫn sử dụng hệ thống người dùng. |

## 3.2 Khảo sát

* Sử nhiều cách khác nhau để khảo sát nhu cầu của doanh nghiệp và người dùng như:

1. **Khảo sát trực tuyến (Online Surveys):**

* Sử dụng các công cụ như Google Forms, SurveyMonkey để tạo và phân phối bảng khảo sát trực tuyến.
* Tiện lợi cho việc thu thập dữ liệu từ một lượng lớn người tham gia.
* Dễ dàng quản lý và phân tích kết quả.

1. **Phỏng vấn (Interviews):**

* Phỏng vấn cung cấp cơ hội cho những cuộc trò chuyện sâu sắc và cụ thể với các đối tượng nghiên cứu.
* Phù hợp khi cần hiểu sâu về quan điểm, trải nghiệm và ý kiến của đối tượng.
* Phỏng vấn có thể được thực hiện cá nhân, qua điện thoại hoặc trực tuyến.

1. **Nhóm thảo luận (Focus Groups):**

* Tập trung một nhóm nhỏ người tham gia để thảo luận về một chủ đề cụ thể.
* Được sử dụng để hiểu sâu về những quan điểm, ý kiến và suy nghĩ của một nhóm người dùng.
* Có thể tạo ra những cơ hội tương tác và phát triển ý tưởng mới.

1. **Quan sát (Observation):**

* Quan sát trực tiếp hành vi, hoạt động và tương tác của đối tượng nghiên cứu trong một môi trường cụ thể.
* Phù hợp khi cần hiểu rõ hành vi tự nhiên của người dùng hoặc khi dữ liệu bảng khảo sát có thể không phản ánh đúng thực tế.

1. **Phân tích dữ liệu sẵn có (Secondary Data Analysis):**

* Sử dụng dữ liệu đã được thu thập trước đó từ các nguồn khác nhau như báo cáo thị trường, nghiên cứu trước đó, hoặc cơ sở dữ liệu công cộng.

## 3.3 Thu nhập thông tin

### 3.3.1 Nhu cầu doanh nghiệp

* Hệ thống quản lý website đăng tin tuyển dụng trực tuyến của ứng viên tại website:
* Khi các ứng viên có nhu cầu tìm việc thì người dùng sẽ trực tiếp truy cập vào website để tham khảo các job mà các công ty đã đăng tin tuyển dụng lên website. Thông tin mỗi job: tên job,tên công ty, hình hảnh, kĩ năng,mức lương…. Sau khi lựa chọn được jobs phù hợp với bản thân ứng viên có thể nộp CV vào vi trí công viêc.Người dung phải bắt buộc đăng nhập mới có thể ứng tuyển và nộp.Ứng viên có thể đăng nhập qua 3 phương thức khác nhau là tài khoản có sắn hoặc đăng nhập bằng tài khoản Google or Facebook.Ứng viên có thể Xem lại được lịch sử các vị trí mà ứng viên đã ứng tuyển…
* Hệ thống sẽ tự động gửi mail thông báo tự động các job vào sáng chủ nhật hàng tuần mà các ứng viên quan tâm khi các ứng viên đăng kí việc tìm job qua Gmail thông qua kĩ năng.
* Các chức năng của quản trị viên:
* Quản trị viên có quyền hạn cao nhất, thực hiện được tất cả thao tác trênweb. Trang quản trị sẽ hiển thị những thông tin theo sự phân quyền sử dụng khác nhau được quản lý bởi thông tin sau: tên đăng nhập, mật khẩu.
* Quản lý công ty: xem danh sách danh sách công ty, thêm, sửa, xóa danh mục.
* Quản lý sản người dùng: xem danh sách sản phẩm, thêm, sửa, xóa sản phẩm.
* Quản lý jobs: xem danh sách thành viên, thêm, sửa, xóa thành viên
* Quản lý CV: xem, cập nhật trạng thái CV
* Quản lý role: xem, thêm, sửa, xóa (chỉ với role admin)
* Quản lý Permission: xem, thêm, sửa, xóa (chỉ với role admin)
* Đối với quyền hạn HR:
* Quản lý CV: chỉ có thể xem, cập nhật tráng thái của CV (review,reject…)
* Quản lý jobs: xem, thêm ,xóa, sửa.

### 3.3.2 Nhu cầu người dùng

* Trải nghiệm người dùng thuận tiện: Người dùng mong đợi một trải nghiệm trực tuyến mượt mà và dễ sử dụng khi tìm kiếm và nộp đơn ứng tuyển cho các vị trí công việc.
* Tìm kiếm và lọc công việc: Các công cụ tìm kiếm và lọc công việc phải được thiết kế một cách thông minh để người dùng có thể dễ dàng tìm kiếm và lựa chọn các vị trí phù hợp với nhu cầu và kỹ năng của họ.
* Tính bảo mật và quyền riêng tư: Người dùng quan tâm đến việc thông tin cá nhân của họ được bảo vệ an toàn khi sử dụng website, đặc biệt là khi đăng nhập và nộp đơn ứng tuyển.
* Đa dạng phương tiện đăng nhập: Các lựa chọn đăng nhập thông qua tài khoản có sẵn hoặc tài khoản Google và Facebook cung cấp sự linh hoạt và tiện lợi cho người dùng.
* Thông báo về việc tìm kiếm công việc: Hệ thống gửi mail thông báo tự động về các vị trí công việc phù hợp hàng tuần giúp người dùng không bỏ lỡ các cơ hội việc làm mới.
* Quản lý và xem lại lịch sử ứng tuyển: Người dùng cần có khả năng xem lại lịch sử các vị trí mà họ đã ứng tuyển để theo dõi trạng thái và quản lý thông tin cá nhân của mình.

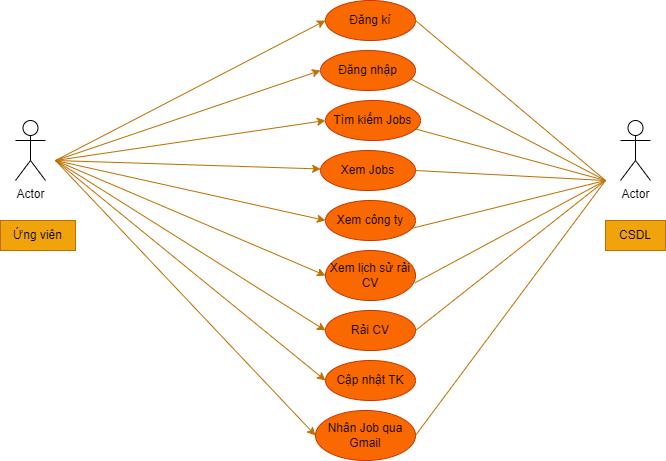
## 3.4 Phân tích thiết kế hệ thống

### 3.4.1 Use case tổng quát



Hinh 10: Sơ đồ use case tổng quát

### 3.1.2 Use case ứng viên



Hinh 11: Sơ đồ use case ứng viên

* Đăng ký: Use case này cho phép ứng viên đăng ký tài khoản của hệ thống khi chưa có tài khoản.
* Đăng nhập: Use case này cho phép ứng viên đăng nhập vào hệ thống khi đã có tài khoản.
* Xem Jobs: Use case này cho phép ứng viên xem các công việc các công ty đang tuyển dung.
* Xem công ty: Use case này cho phép ứng viên xem chi tiết về công ty mà ứng viên quan tâm.
* Tìm kiếm jobs: Use case này cho phép ứng viên tìm kiếm jobs theo tên job, skills, địa chỉ.
* Xem lịch sử rải CV: Use case này cho phép ứng viên xem lại lịch sử các CV mà ứng viên đã ứng tuyển
* Rải CV: Use case này cho phép ứng viên nộp CV vào các công việc phù hợp với bản thân.
* Cập nhật tài khoản: Use case này cho phép ứng viên cập nhật lại thông tin của ứng viên.
* Nhận Job qua Gmail: Use case này cho phép ứng viên nhận jobs qua các skills mà ứng viên đă đăng kí.

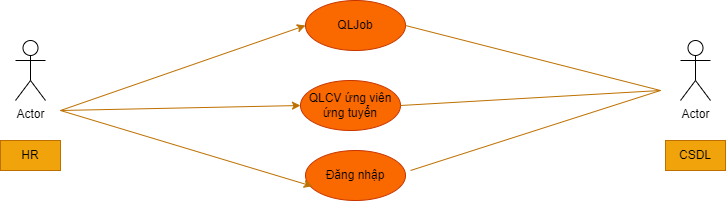
### 3.1.3 Use case quản trị



Hinh 12: Sơ đồ use case người quản trị

* Quản lý công ty: Use case này cho phép người quản trị xem, thêm, sửa, xóa công ty.
* Quản lý Người dùng: Use case này cho phép người quản trị xem, thêm, sửa, xóa user hệ thống.
* Quản lý Jobs: Use case này cho phép người quản trị xem, thêm, sửa, xóa các jobs mà đă đăng tuyển.
* Quản lý CV: Use case này cho phép người quản trị xem, thêm, sửa, xóa CV ứng viên ứng tuyển.
* Quản lý Roles: Use case này cho phép người quản trị xem, thêm, sửa, xóa các role của hệ thống.
* Thống kê: Use case này cho phép người quản trị xem được tổng số CV ứng tuyển,số công ty, số tài khoản…
* Quản lý Permission: Use case này cho phép người quản trị xem, thê, sửa, xóa các quyền của người dung hệ thống
* Đăng nhập: Use case này cho phép người quản trị đăng nhập vào hệ thống

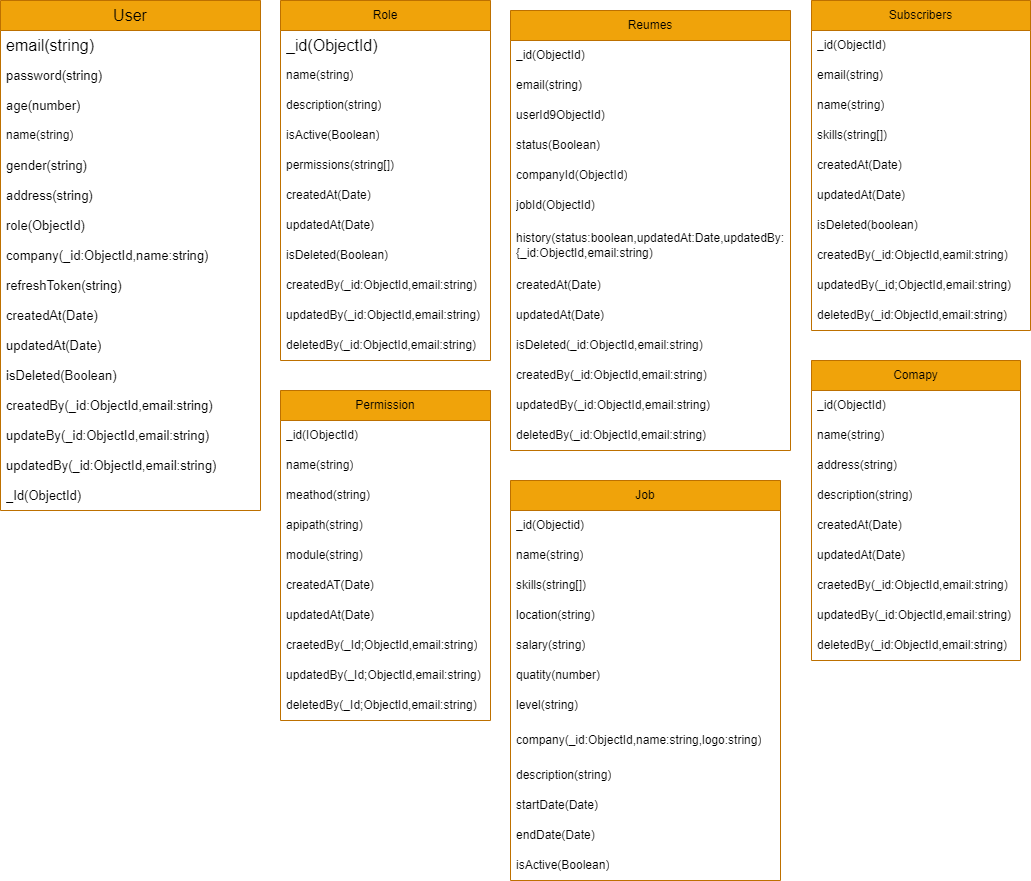
### 3.1.4 Use case HR



Hinh 13: Sơ đồ use case HR

* QL Jobs:Use case này cho phép HR có thể xem, thêm, sửa, xóa jobs
* QL CV ứng viên ứng tuyển: Use case này cho phép HR xem,cập nhật tình trạng của CV mà ứng viên ứng tuyển.

## 3.5 Xây dựng cơ sở dữ liệu



Hinh 14: Biều đồ thiết kế cơ sở dữ liệu

# CHƯƠNG 4: KẾT QUẢ THỰC HIỆN

## 4.1 Môi trường cài đặt và công cụ

### 4.1.1 Visual Studio Code

Visual Studio Code (VS Code) là một trình soạn thảo mã nguồn mở gọn nhẹ nhưng có khả năng vận hành mạnh mẽ trên 3 nền tảng là Windows, Linux và macOS được phát triển bởi Microsoft. Nó hỗ trợ cho JavaScript, Node.js và TypeScript, cũng như cung cấp một hệ sinh thái mở rộng vô cùng phong phú cho nhiều ngôn ngữ lập trình khác.

  Có thể nói, VS Code là sự kết hợp giữa một trình soạn thảo mã nguồn với những công cụ phát triển mạnh mẽ như Git, Debug hay Syntax Highlighter.

Ngoài ra, Visual Studio Code cho phép người dùng tự chỉnh giao diện nhằm tạo sự thoải mãi rất mang đến cho người dùng.

### 4.1.2 MongoDB Compass

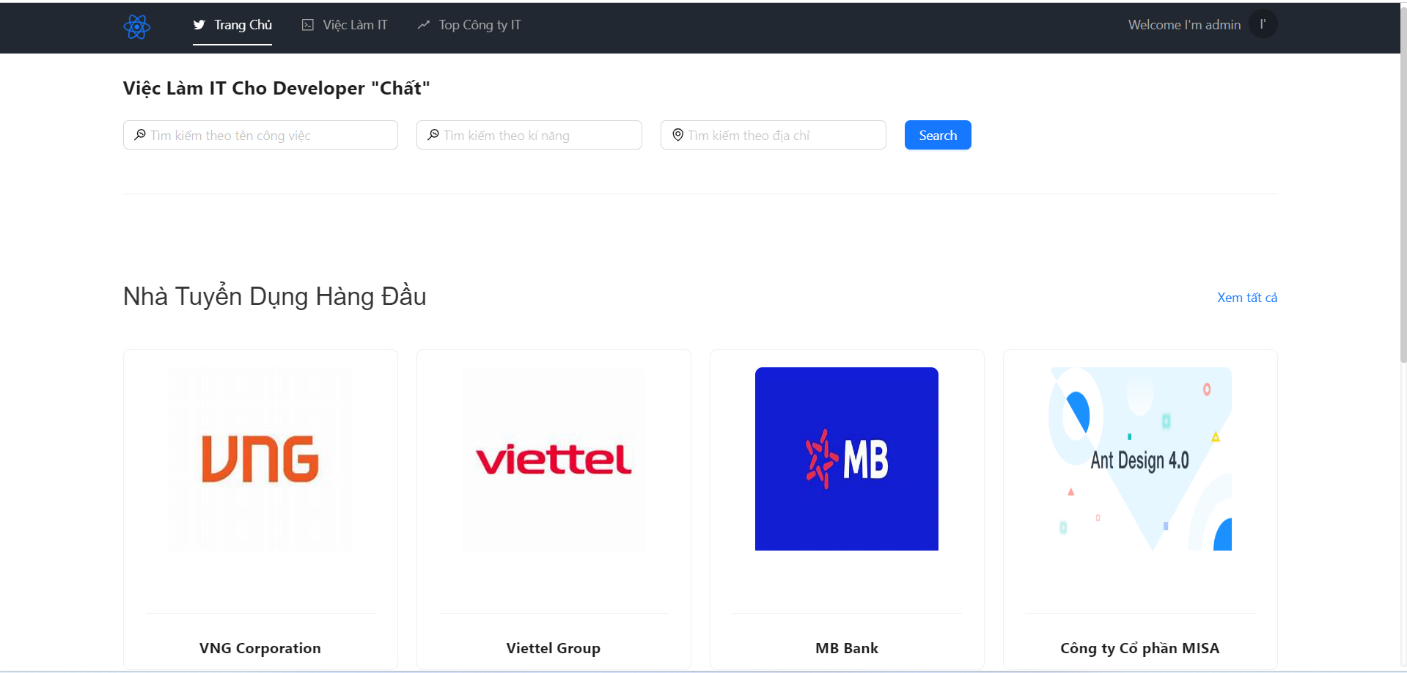
MongoDB Compass là một GUI cho MongoDB. Nó còn được gọi là MongoDB GUI. MongoDB cho phép người dùng phân tích nội dung dữ liệu được lưu trữ của họ mà không cần biết trước về cú pháp truy vấn MongoDB. Khi chúng tôi tìm hiểu khám phá dữ liệu của mình trong môi trường trực quan, chúng tôi có thể sử dụng Compass GUI để tối ưu hóa hiệu suất, quản lý chỉ mục và triển khai xác thực Document.

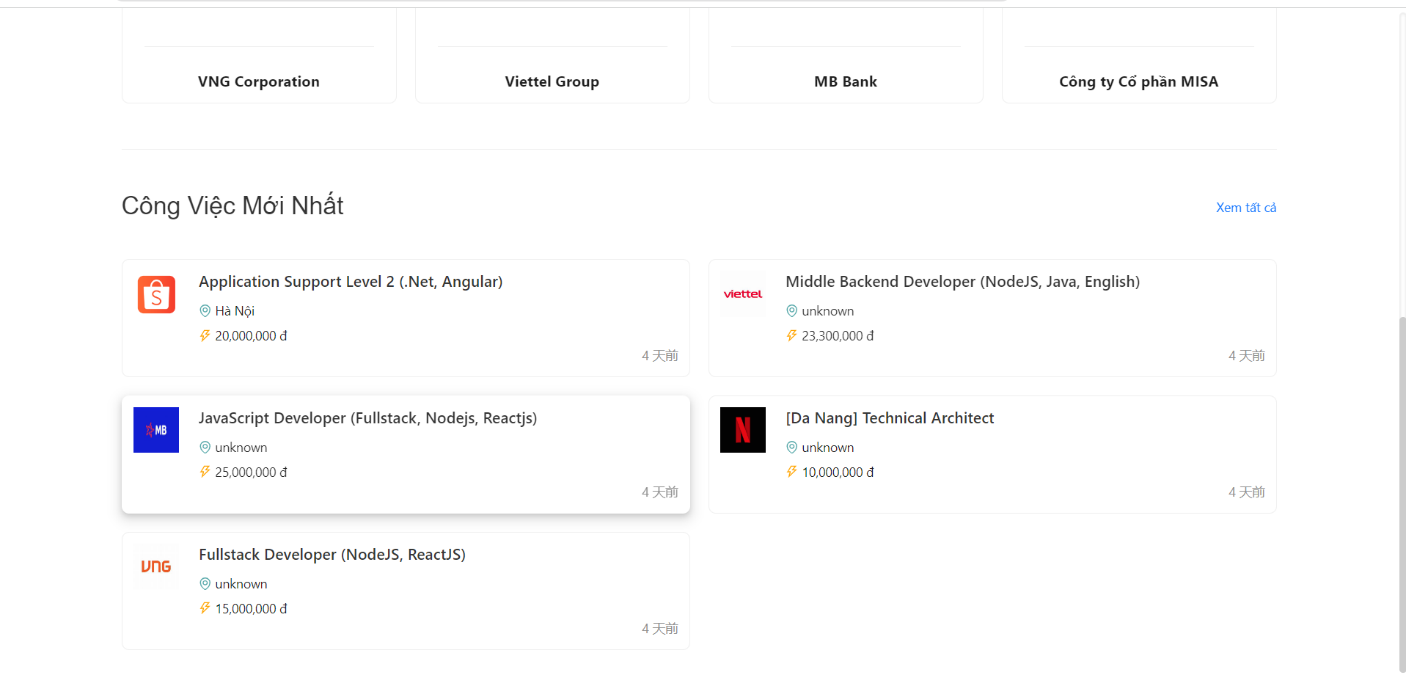
MongoDB GUI có sẵn trong bốn phiên bản sau:

* Compass Community: Phiên bản này được thiết kế để phát triển với MongoDB và bao gồm một tập hợp con các tính năng của Compass.
* Compass: Nó được phát hành dưới dạng phiên bản đầy đủ của MongoDB
* Compass. Nó bao gồm tất cả các tính năng và khả năng mà MongoDB cung cấp.
* Compass Randomly: Nó bị giới hạn ở hoạt động đọc chỉ khi tất cả các khả năng cập nhật và xóa bị loại bỏ.
* Compass Isolated: Phiên bản Isolated của Compass MongoDB không bắt đầu bất kỳ yêu cầu mạng nào ngoại trừ máy chủ MongoDB mà MongoDB GUI kết nối. Nó được thiết kế để sử dụng trong các môi trường an toàn cao.

## 4.2 Giao diện người dùng

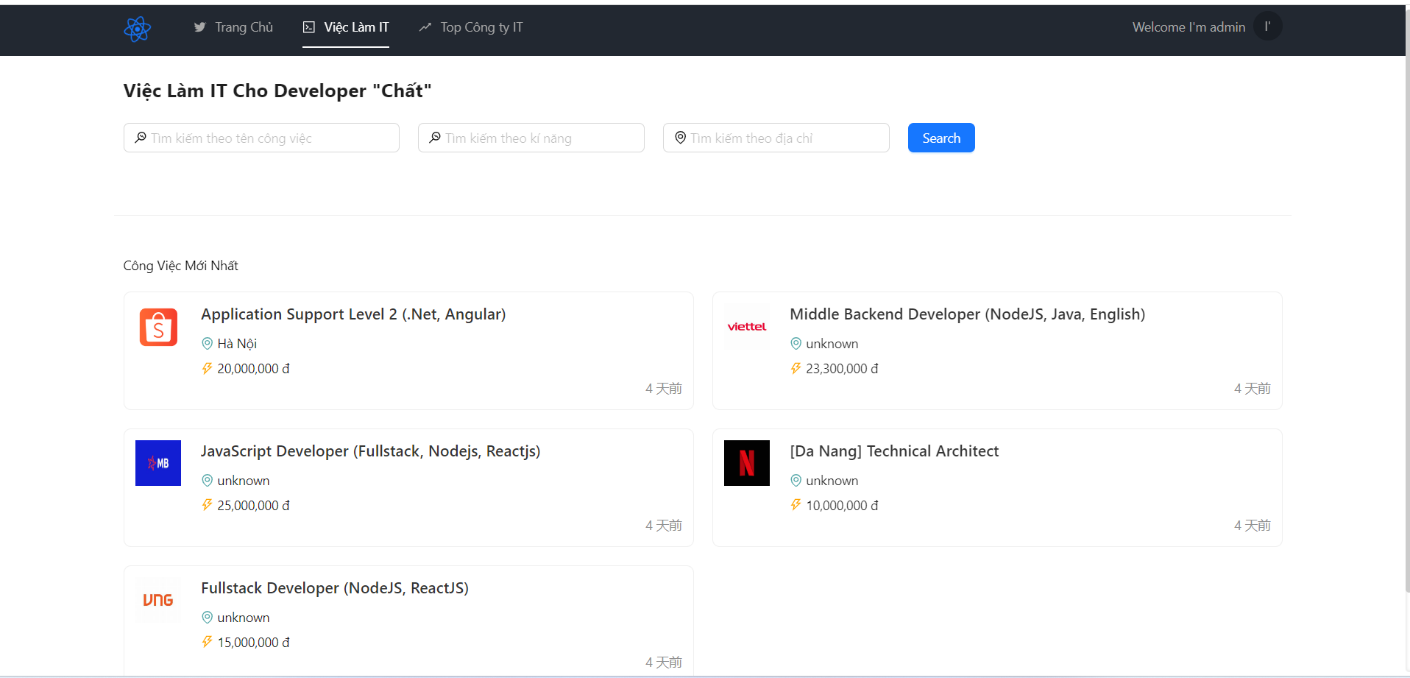
### 4.2.1 Trang chủ





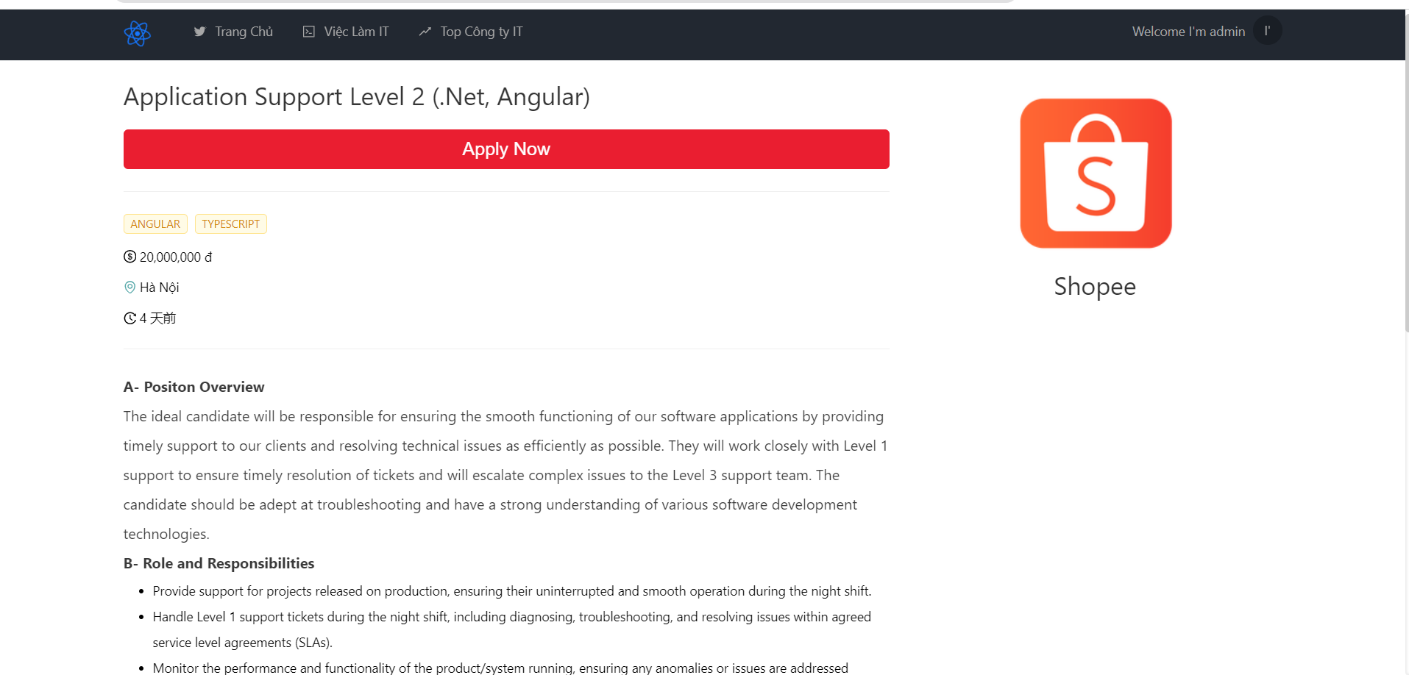
Hinh 15: Giao diện trang chủ

### 4.2.2 Xem việc làm



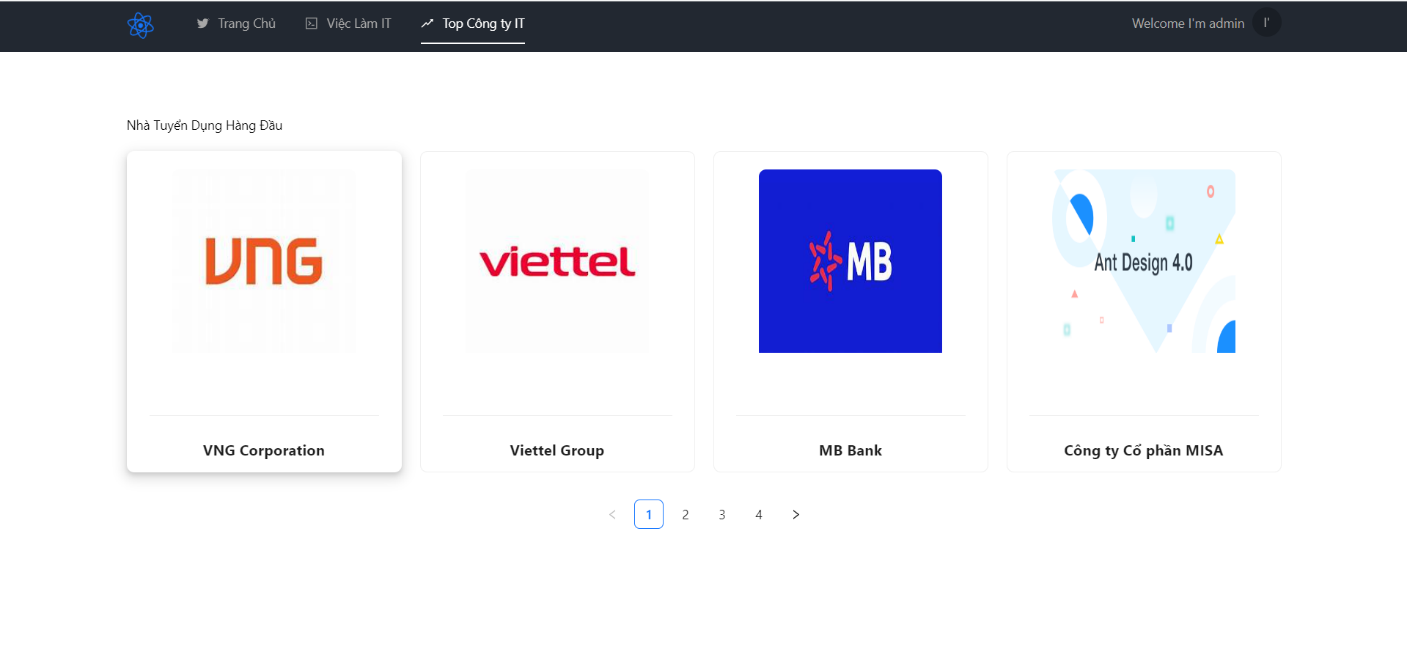
Hinh 16: Xem danh sách các công việc

### 4.2.3 Xem chi tiết việc làm



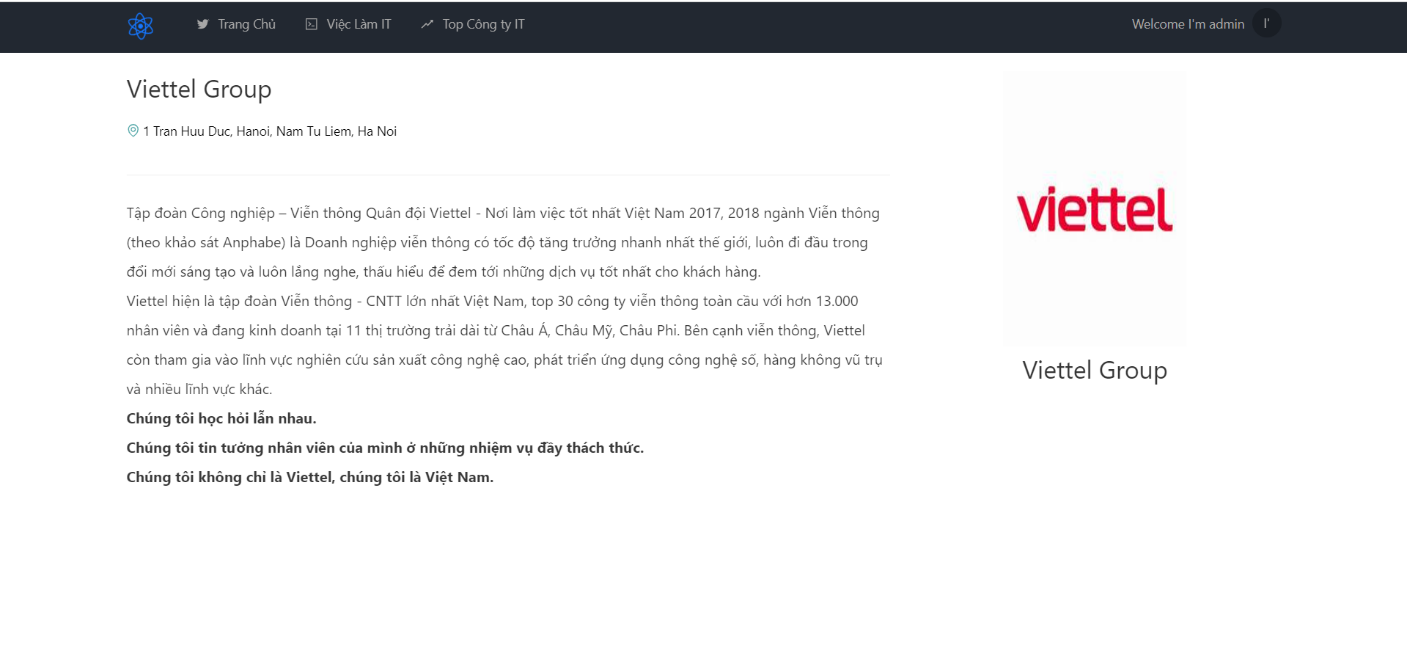
Hinh 17: Xem chi tiết công việc

### 4.2.4 Xem công ty



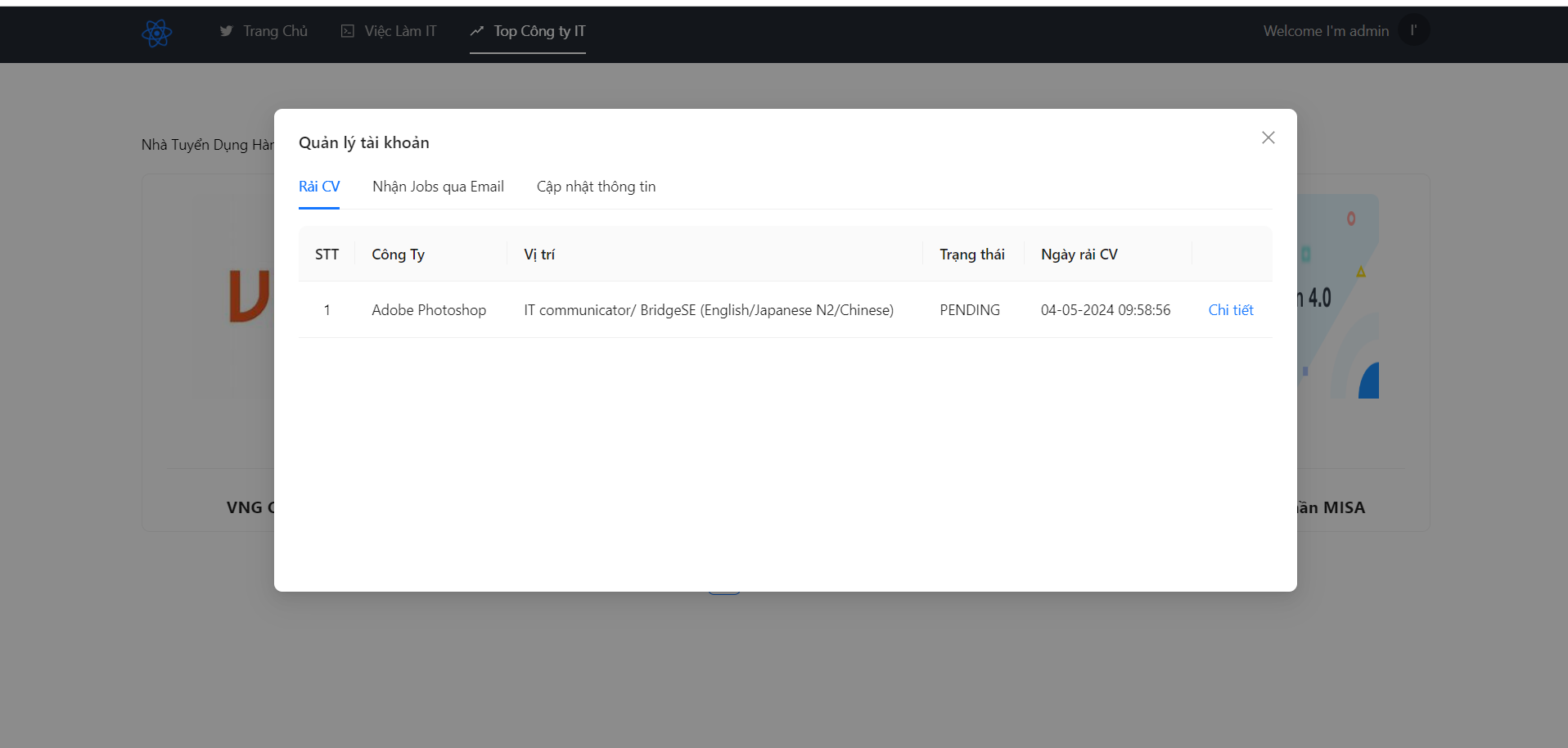
Hinh 18: Xem danh sach các công ty

### 4.2.5 Xem chi tiết công ty



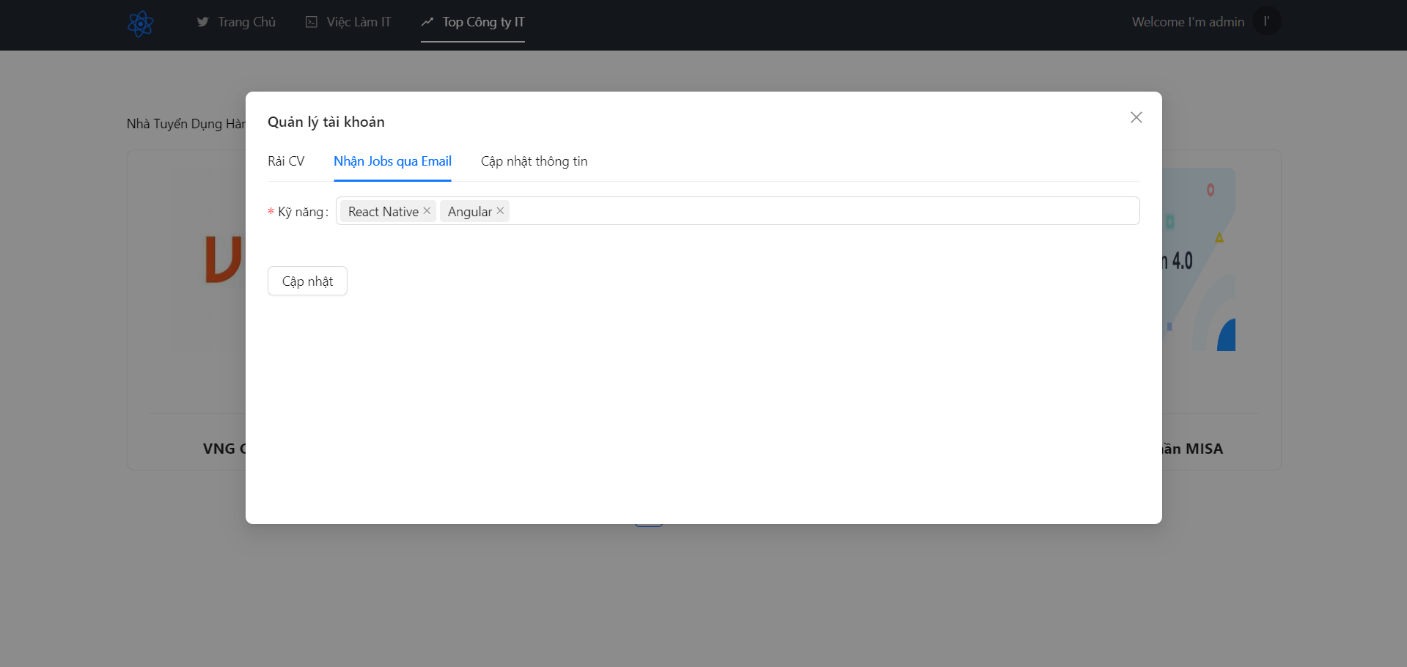
Hinh 19: Xem chi tiết công ty

### 4.2.6 Xem lịch sử rải CV



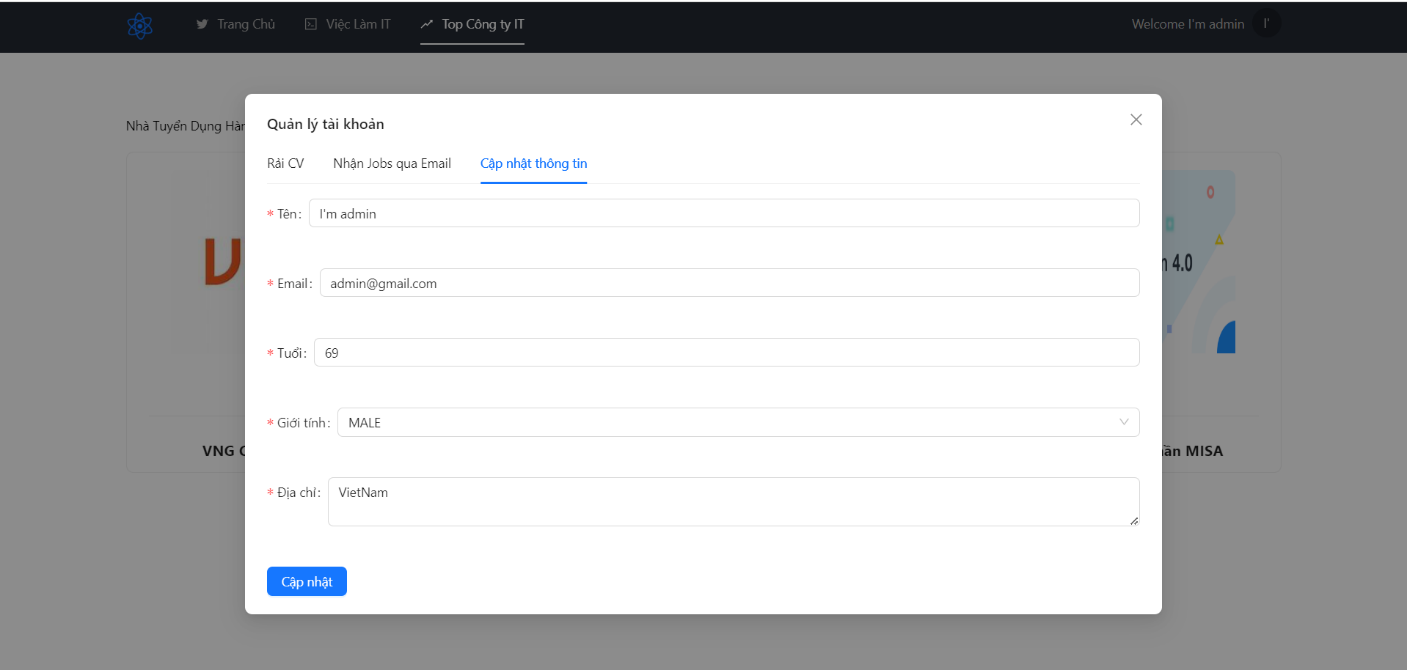
Hinh 20: Xem lịch sử ứng tuyển

### 4.2.7 Nhận job thông qua Gmail



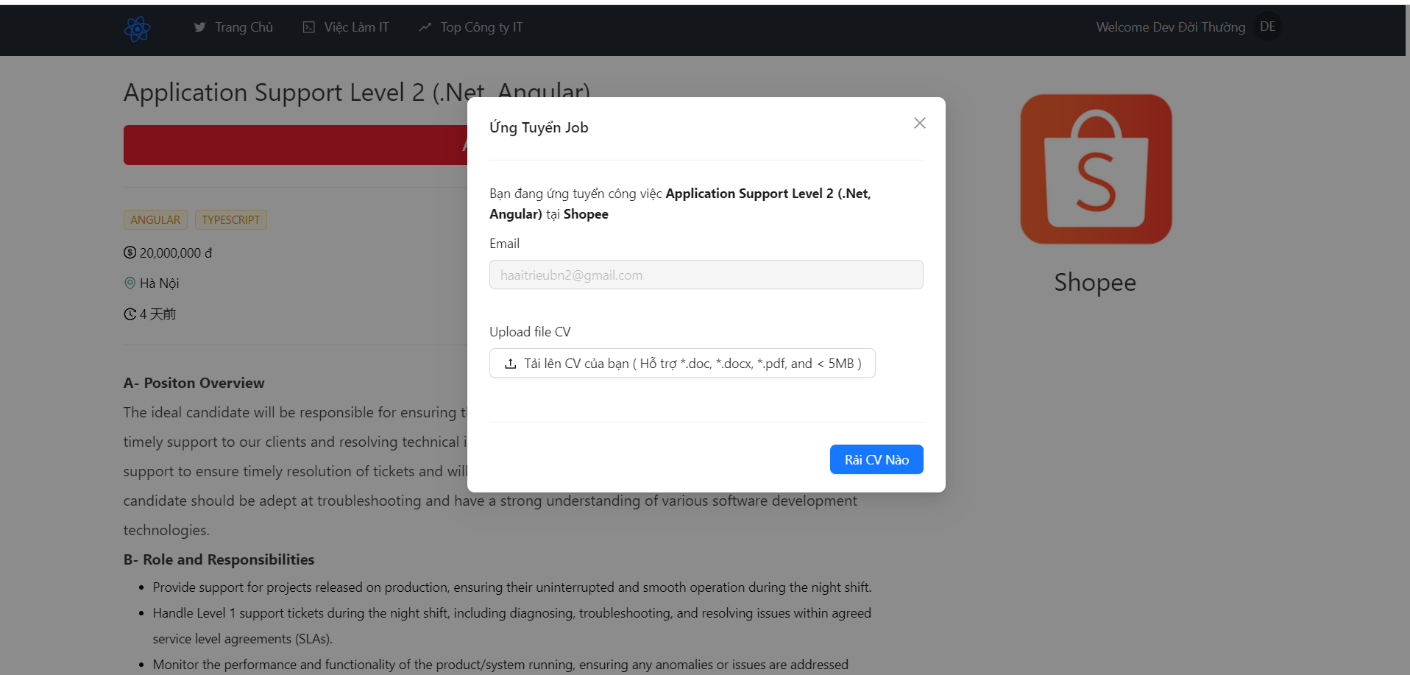
Hinh 21: Nhận việc làm qua Gmail

### 4.2.8 Cập nhật thông tin cá nhân



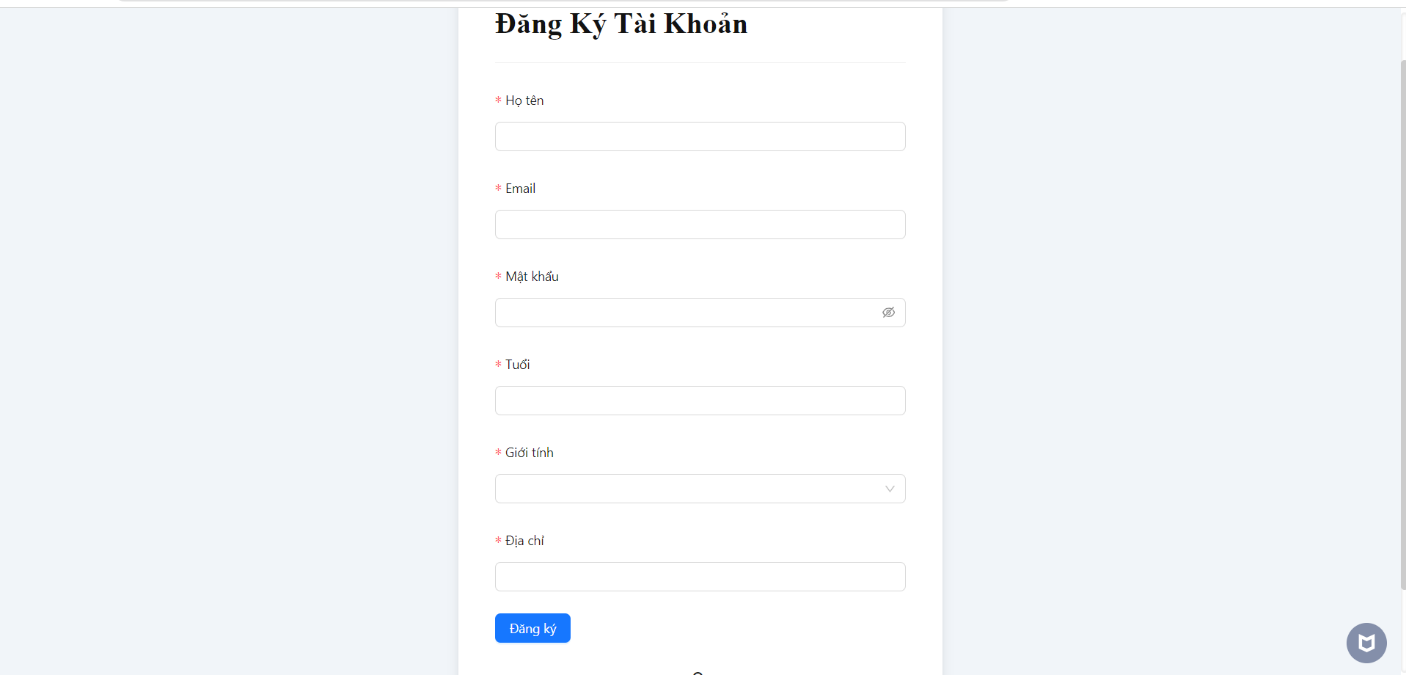
Hinh 22: Cập nhật thông tin cá nhân

### 4.2.9 Nộp CV



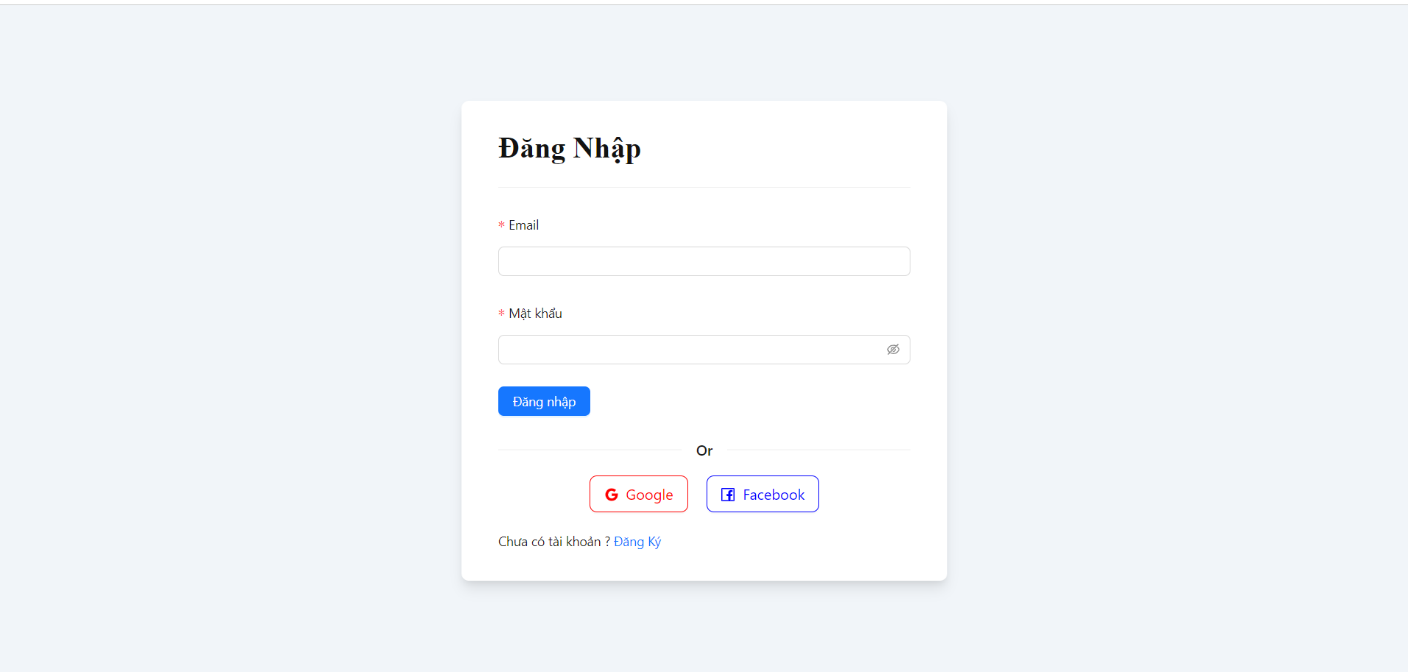
Hinh 23: Ứng tuyển việc làm

### 4.2.10 Đăng ký tài khoản



Hinh 24: Giao diện đăng kí tài khoản

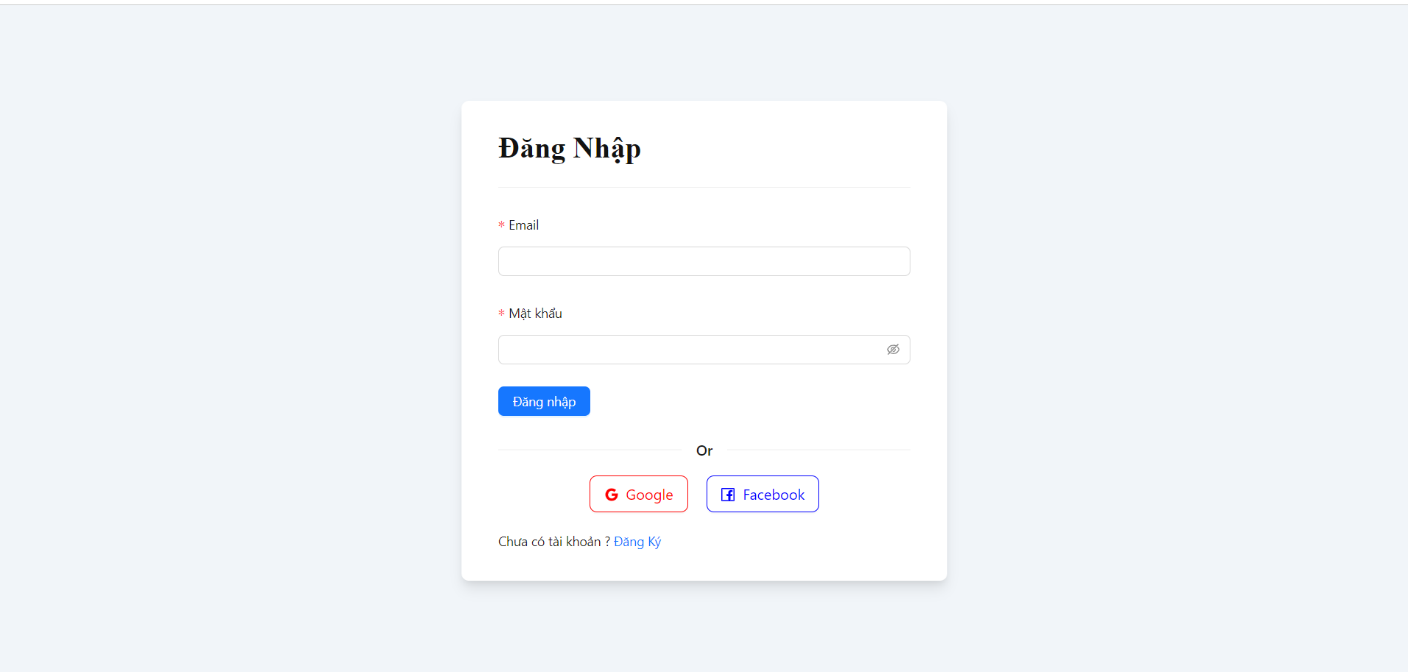
### 4.2.11 Đăng nhập



Hinh 25: Giao diện đăng nhập tài khoản

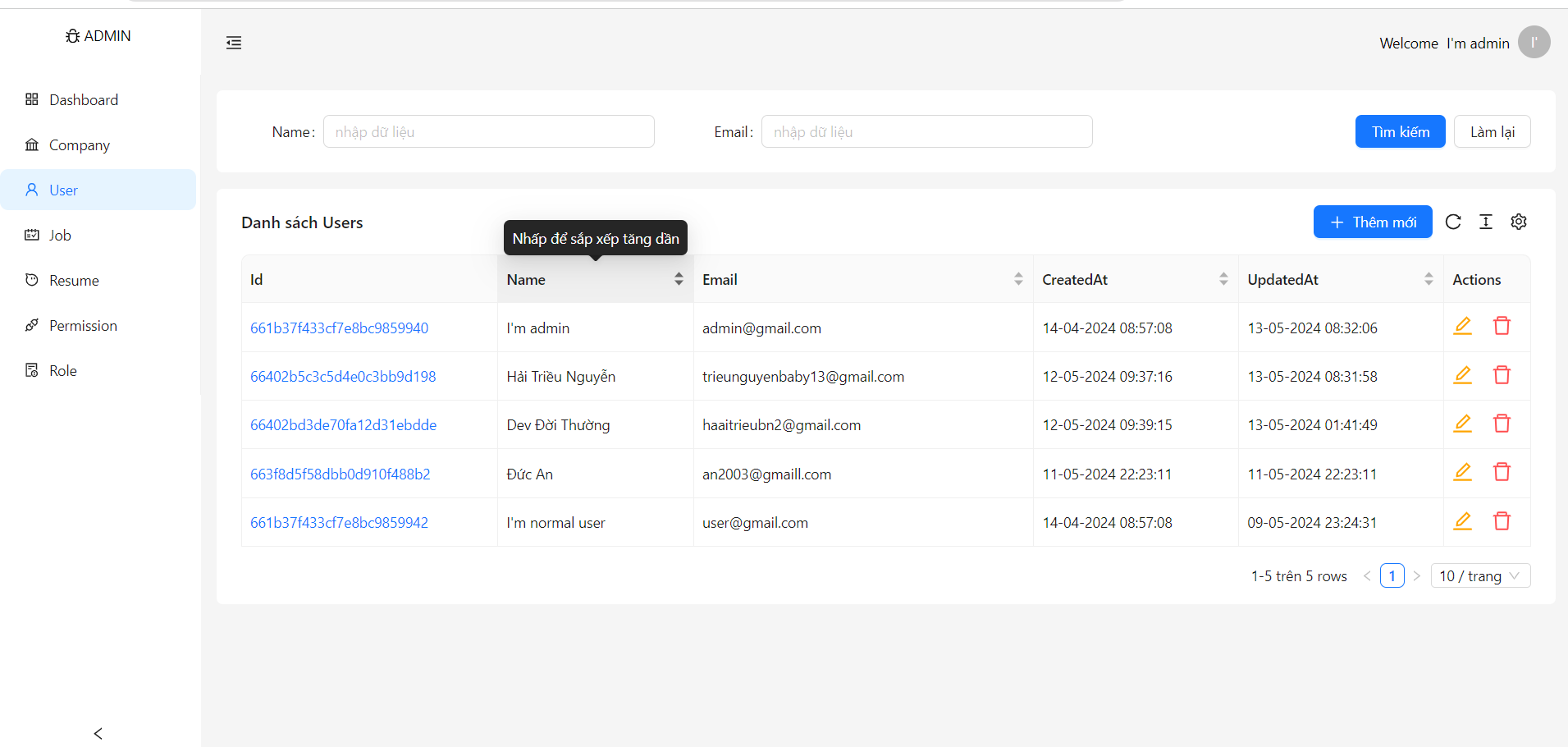
## 4.3 Giao diện Admin

### 4.3.1 Đăng nhập Admin

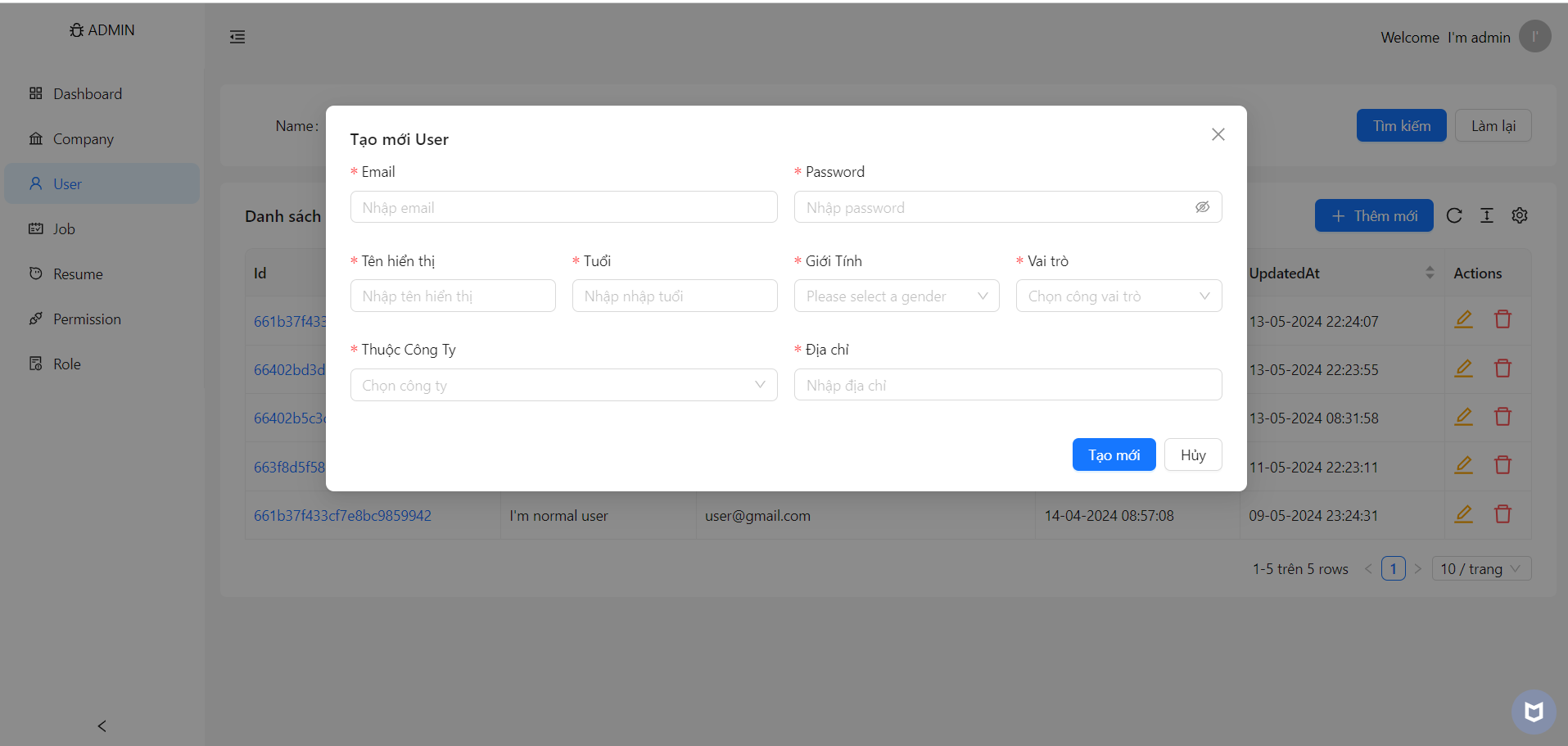


Hinh 26: Giao diện đăng nhập người quản trị

### 4.3.3 Quản lý tài khoản

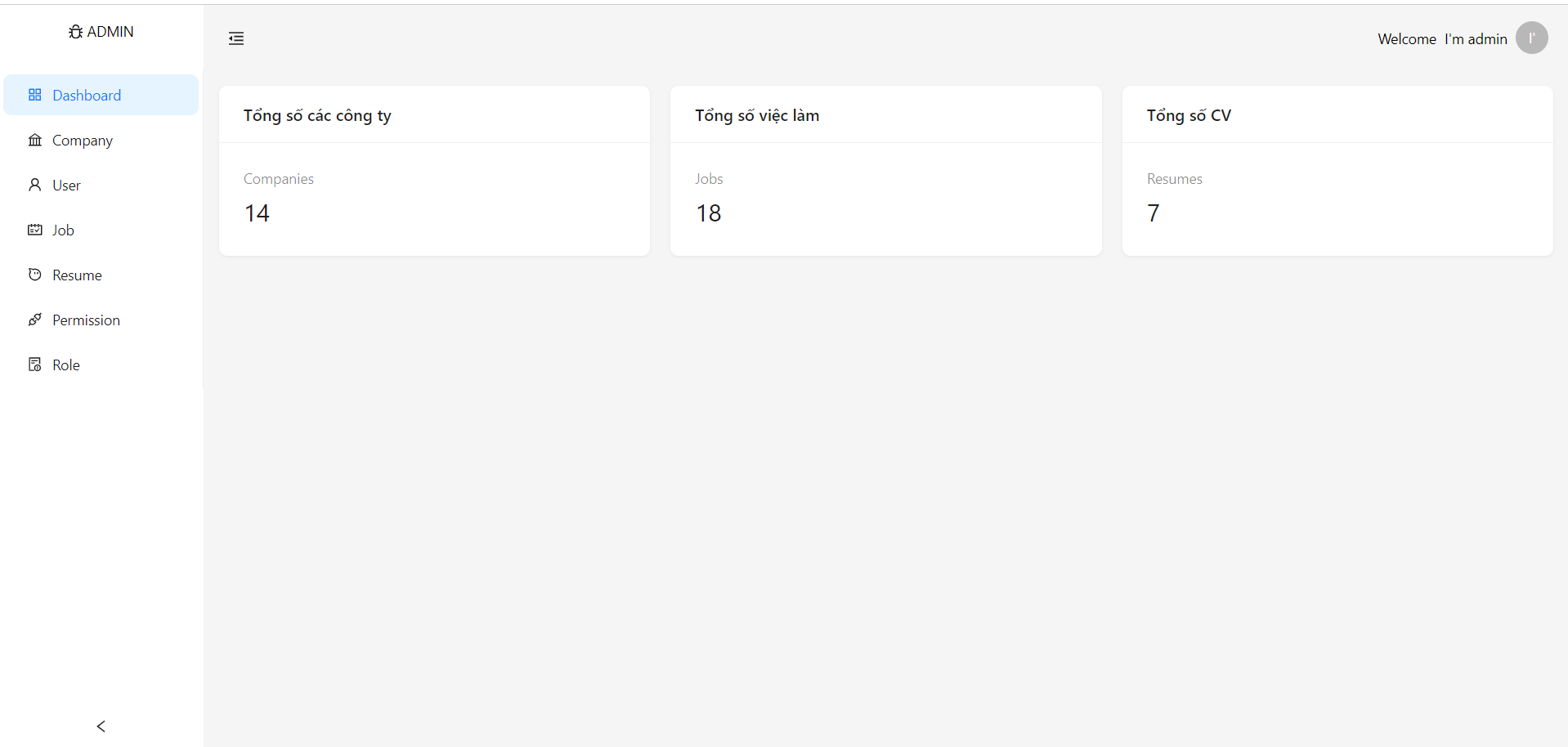


Hinh 27: Quản lý người dùng



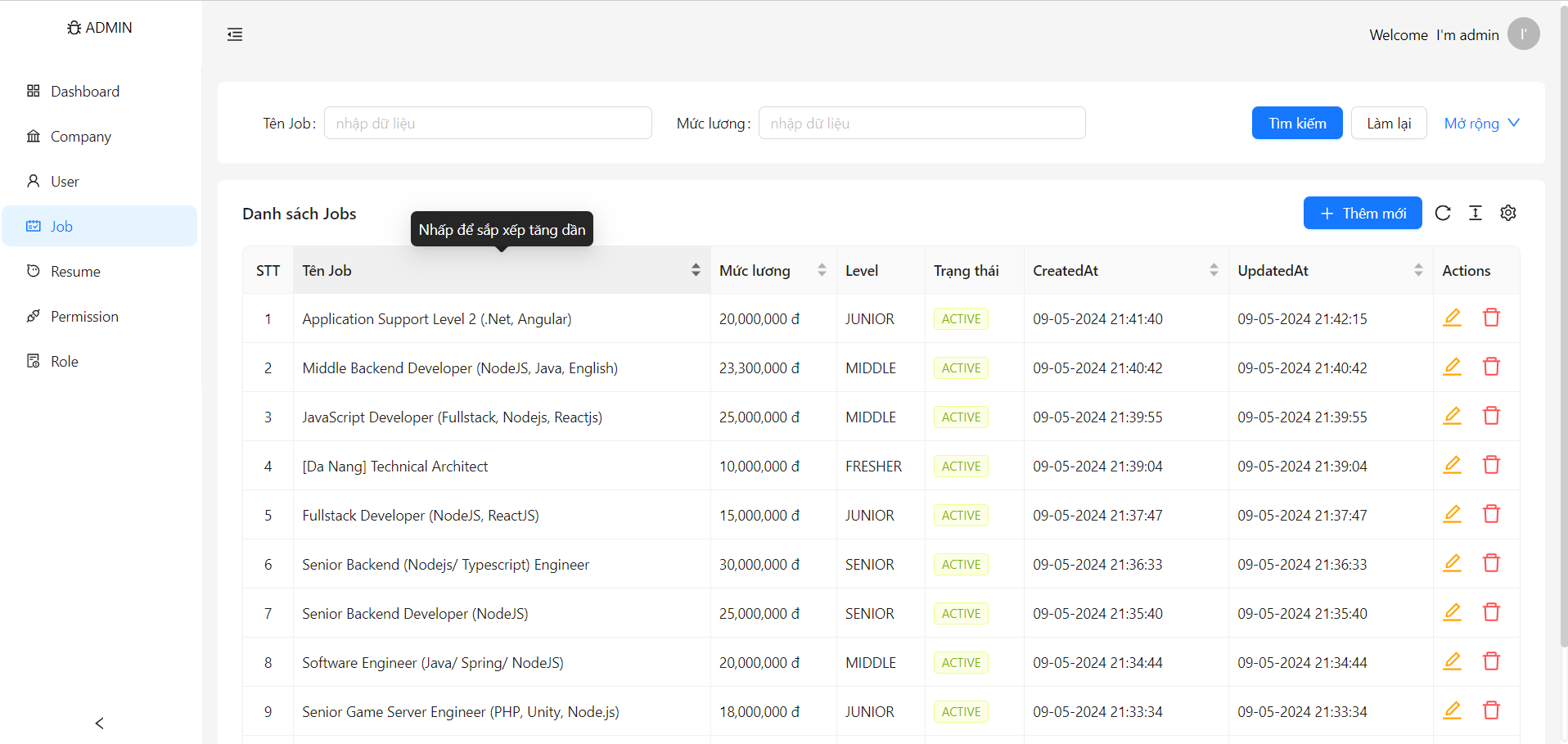
Hinh 28: Thêm mới người dùng

### 4.3.4 Thống kê

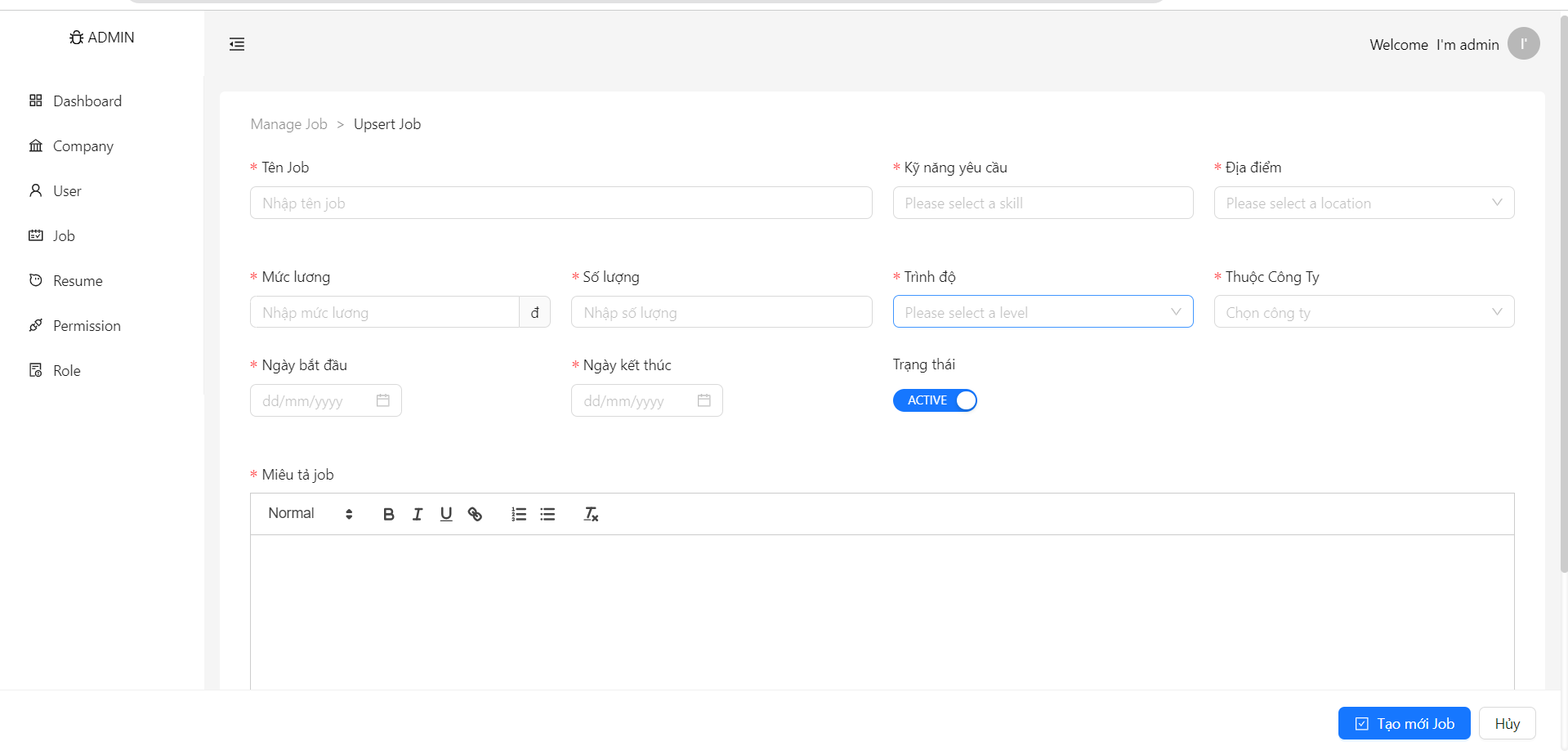


Hinh 29: Thống kê

### 4.3.5 Quản lý việc làm

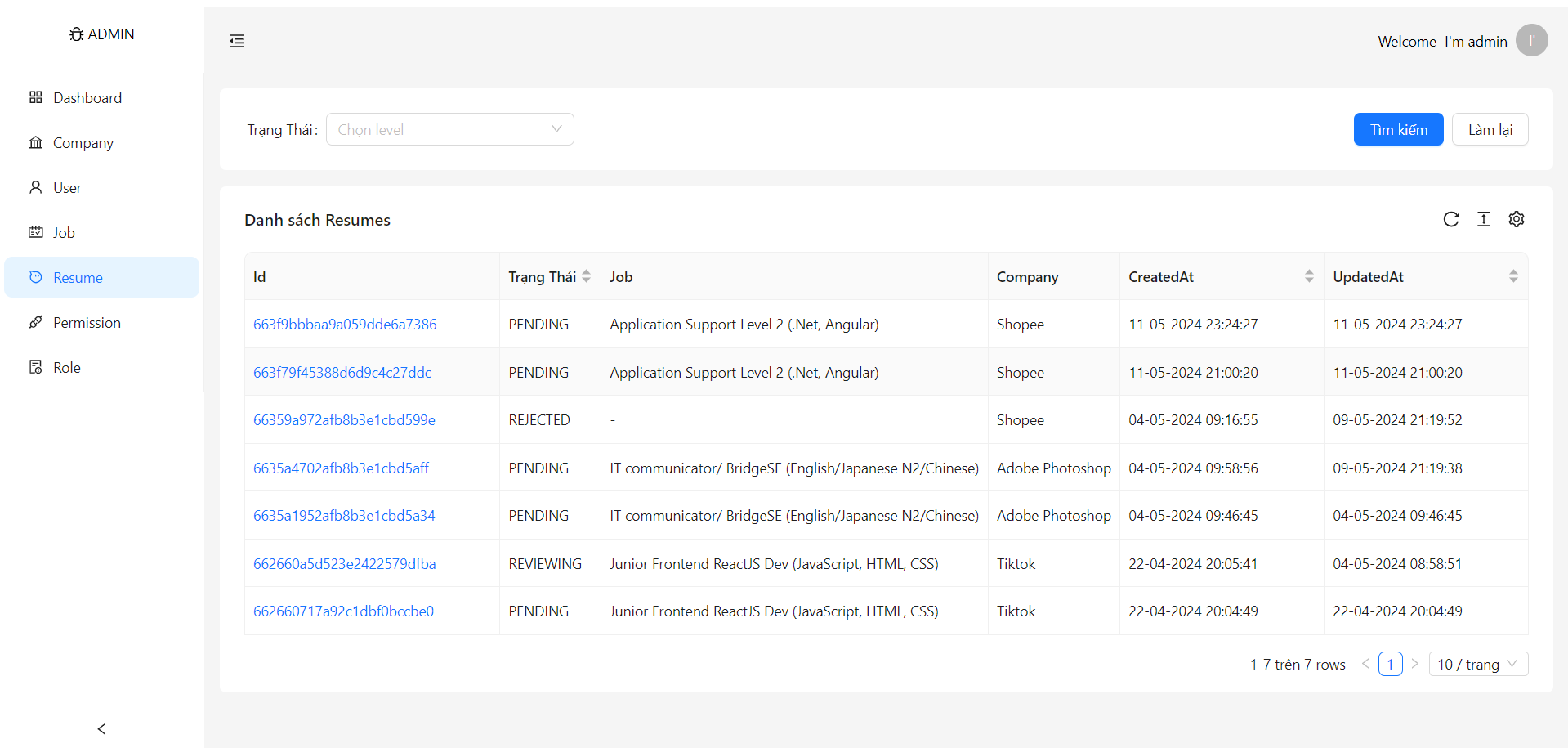


Hinh 30: Quản lý việc làm



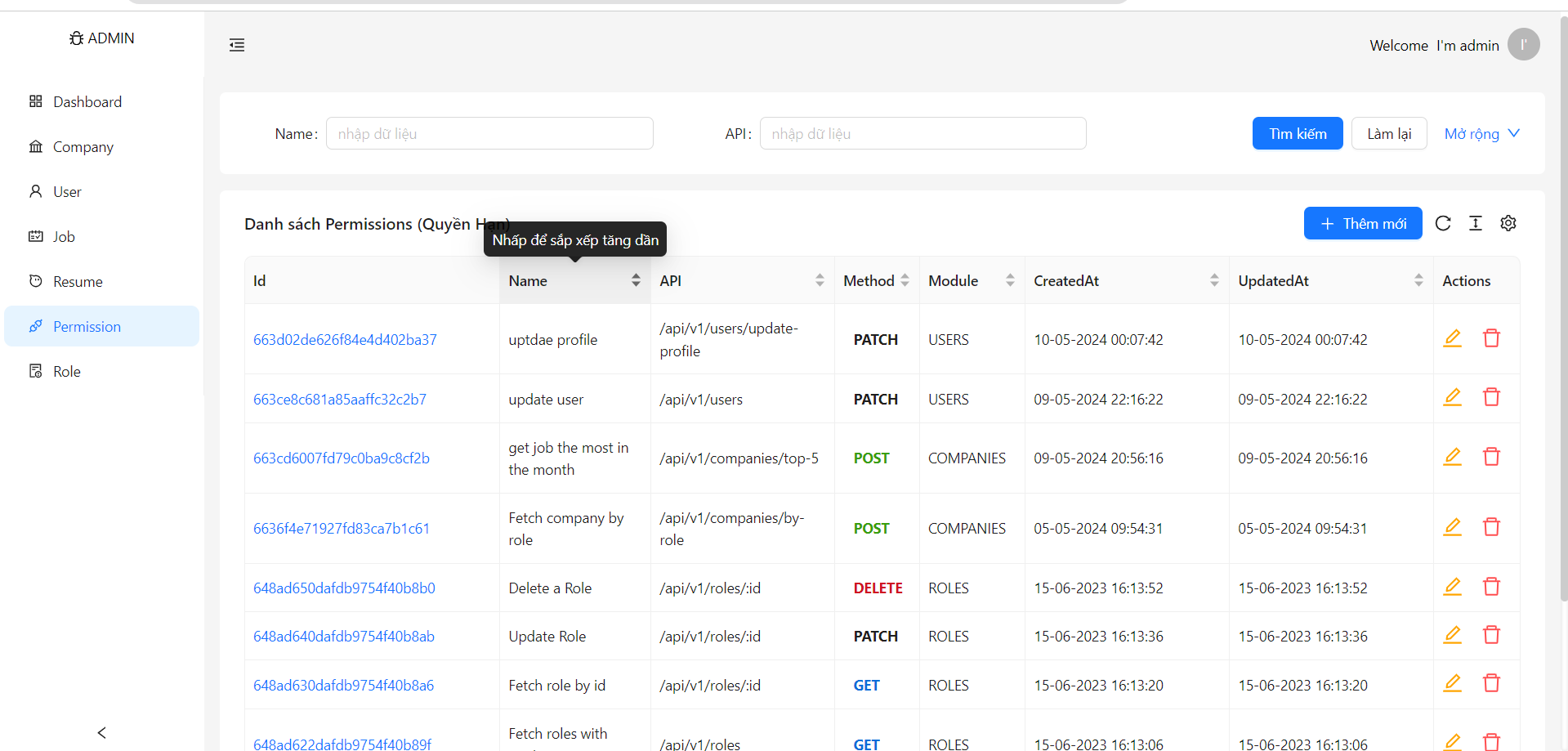
Hinh 31: Thêm mới việc làm

### 4.3.6 Quản lý CV ứng viên

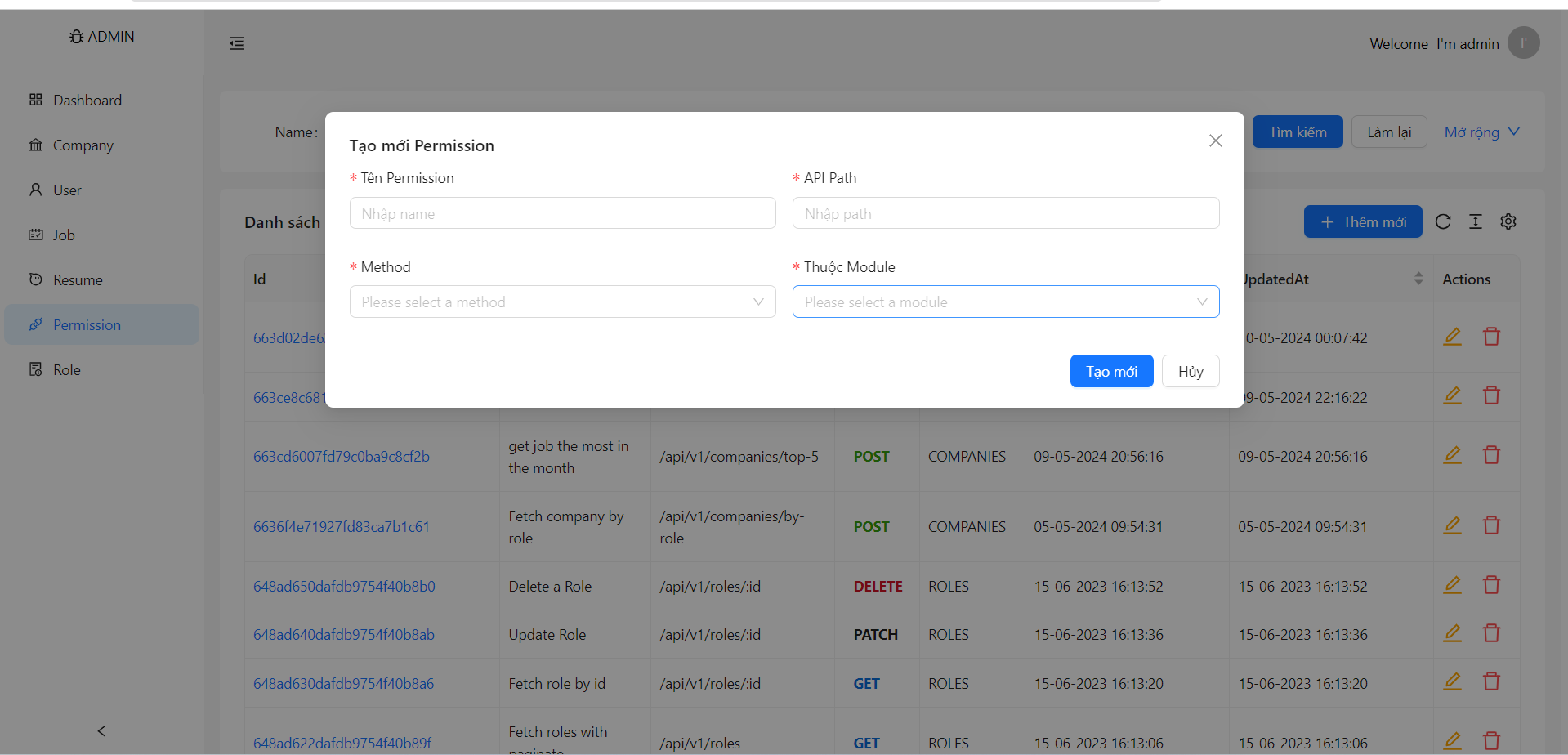


Hinh 32: Quản lý CV ứng viên

### 4.3.7 Quản lý quyền hạn người dùng

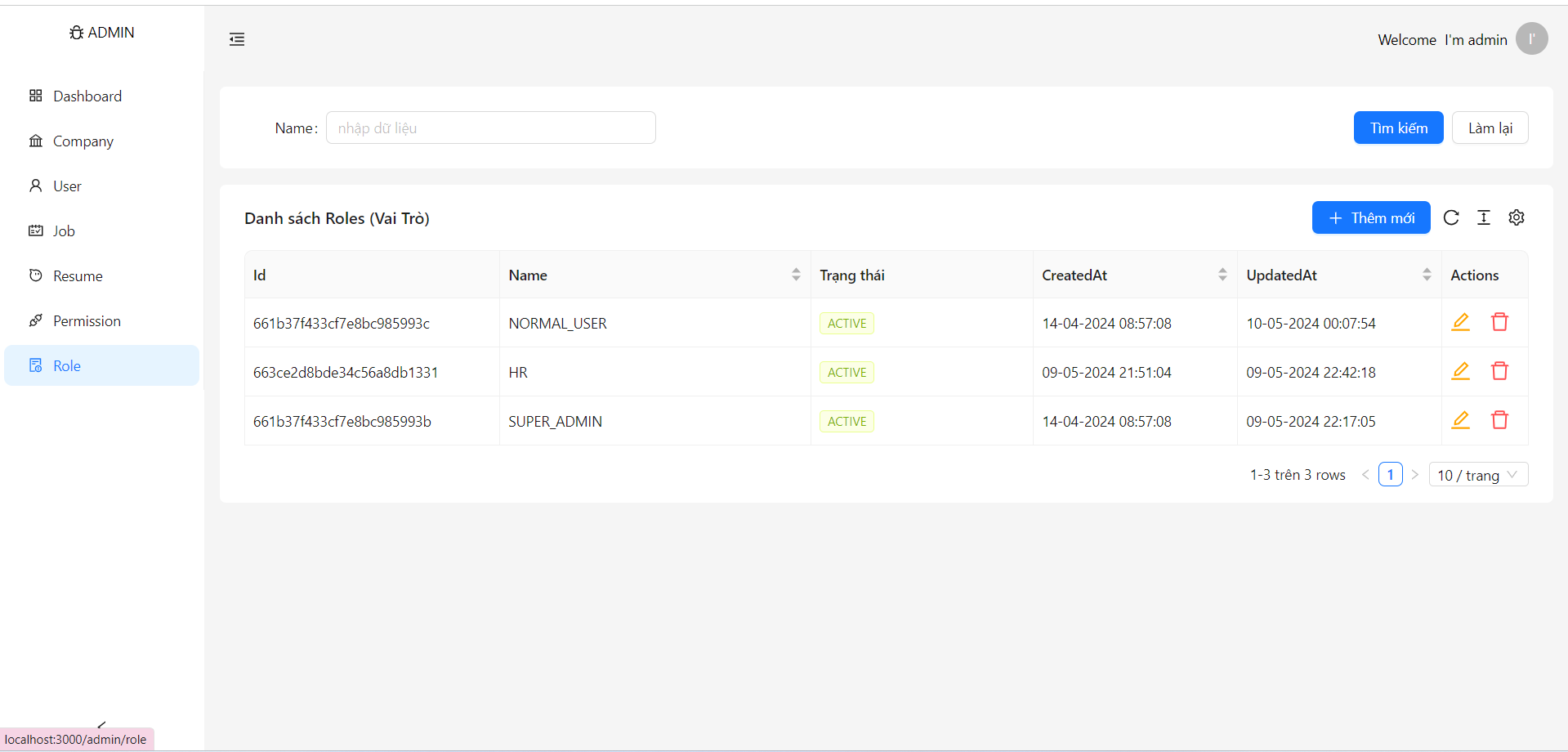


Hinh 33: Quản lý quyền hạn người dùng

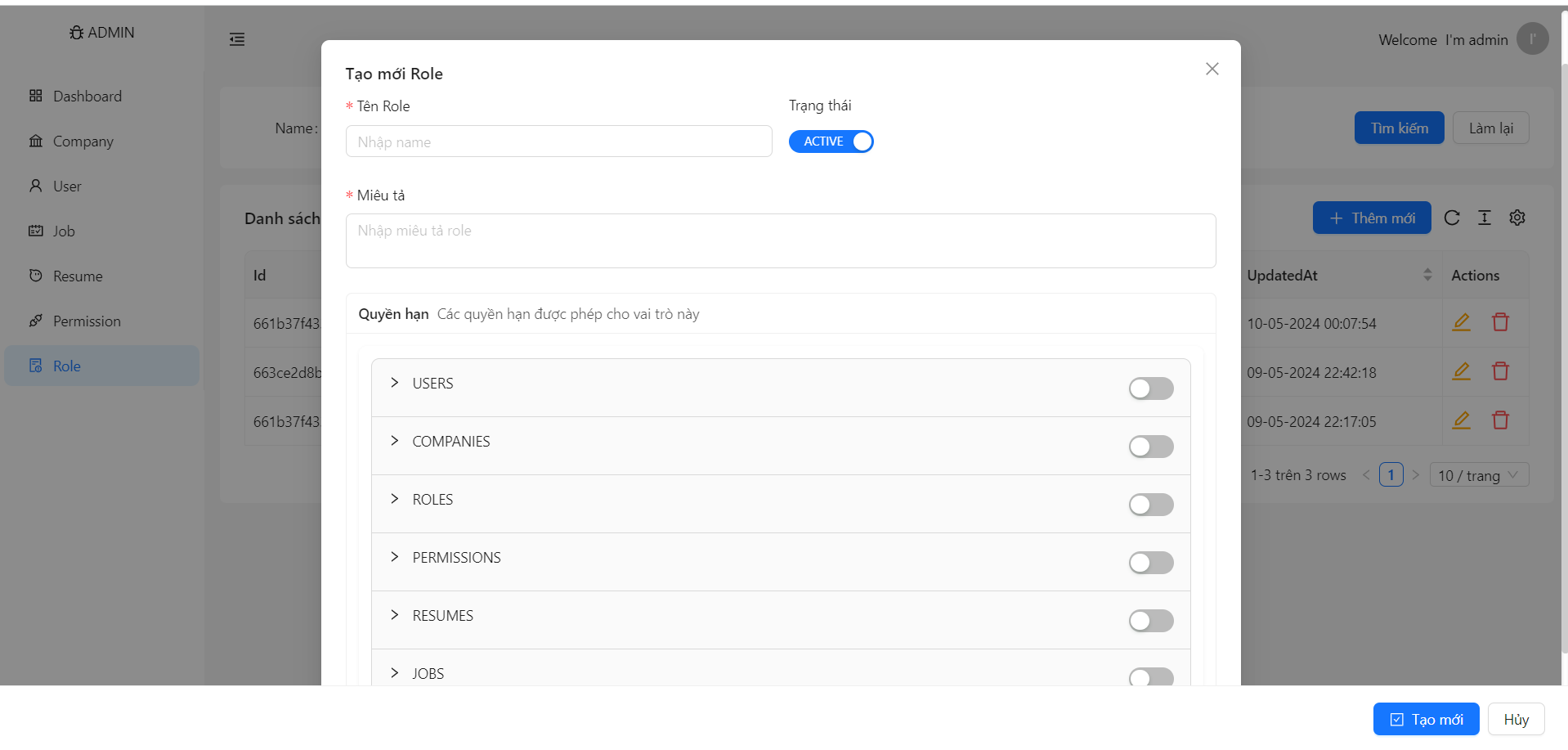


Hinh 34: Thêm mới quyền hạn người dùng

### 4.3.8 Quản lý Role



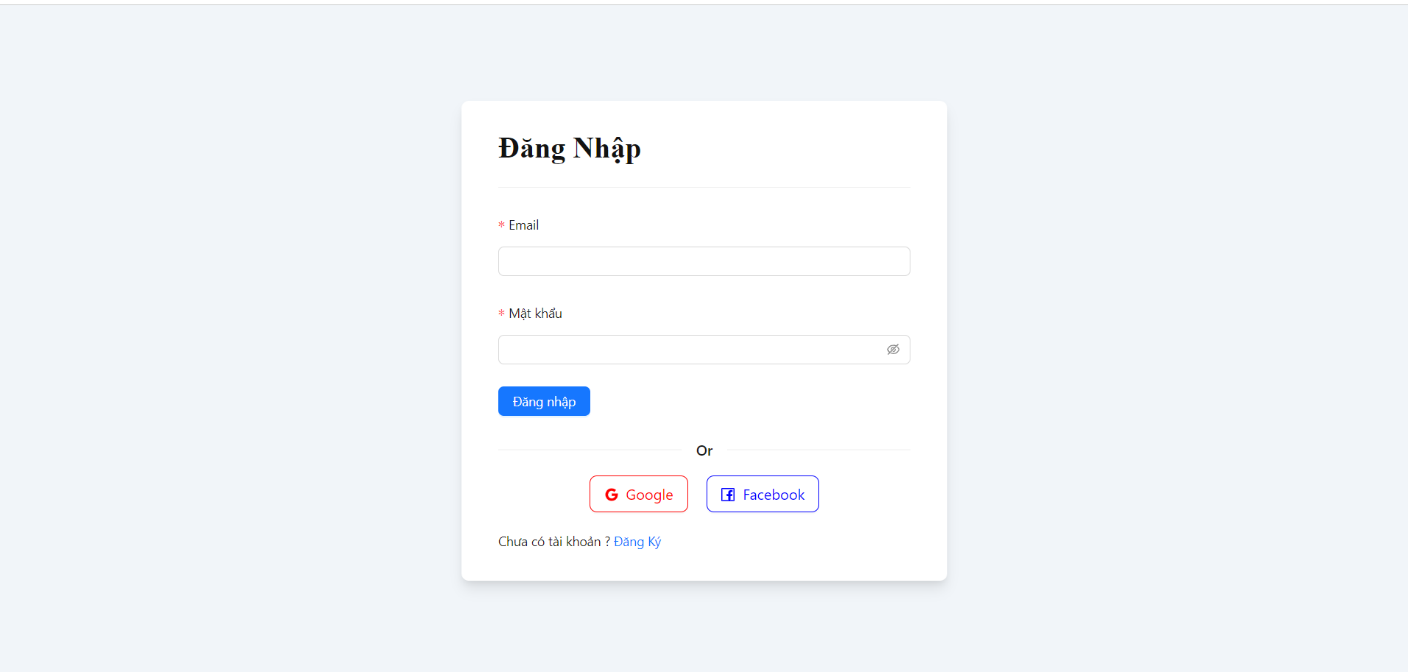
Hinh 35: Quản lý quyền



Hinh 36: Thêm mối quyền

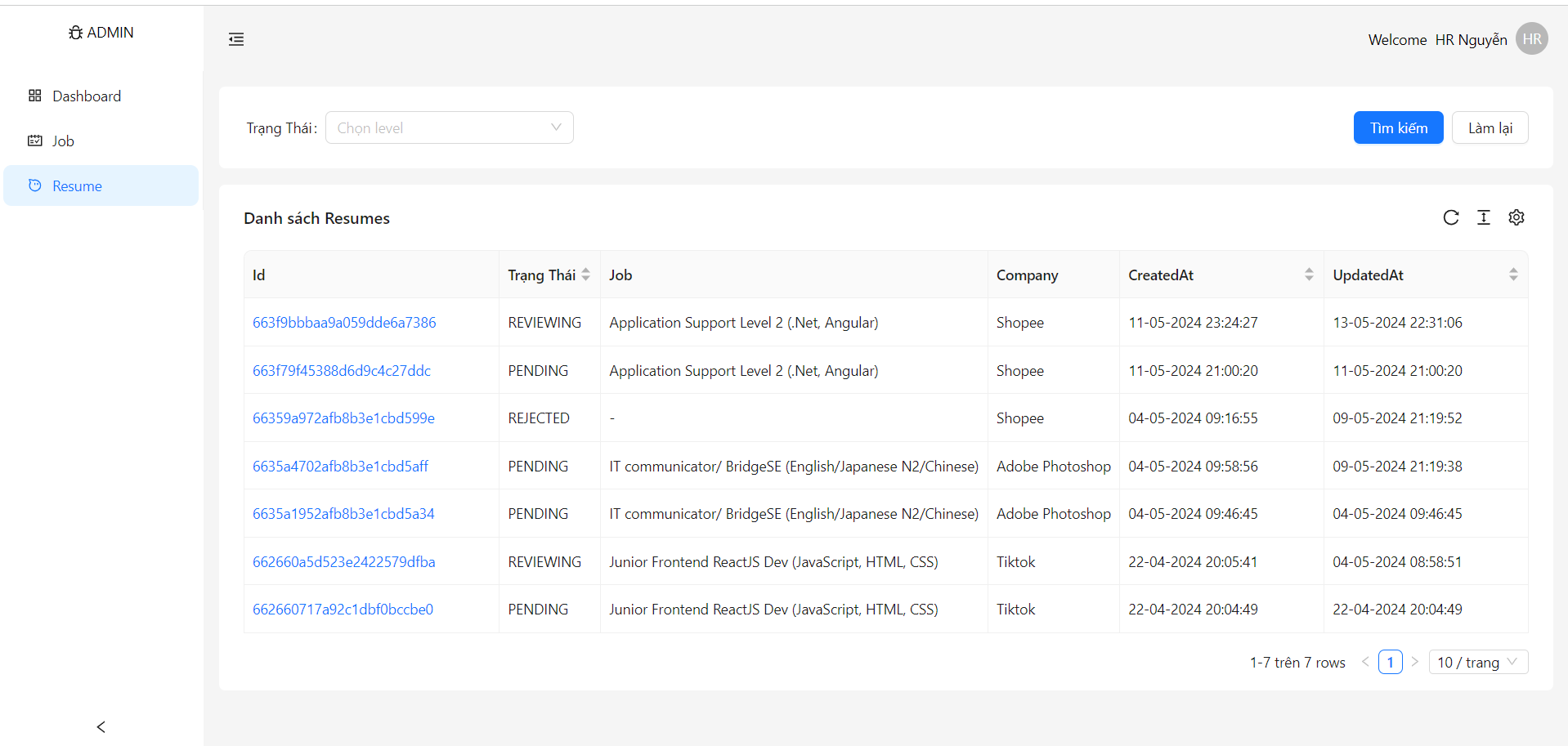
## 4.4 Giao diện HR

### 4.4.1 Đăng nhập HR

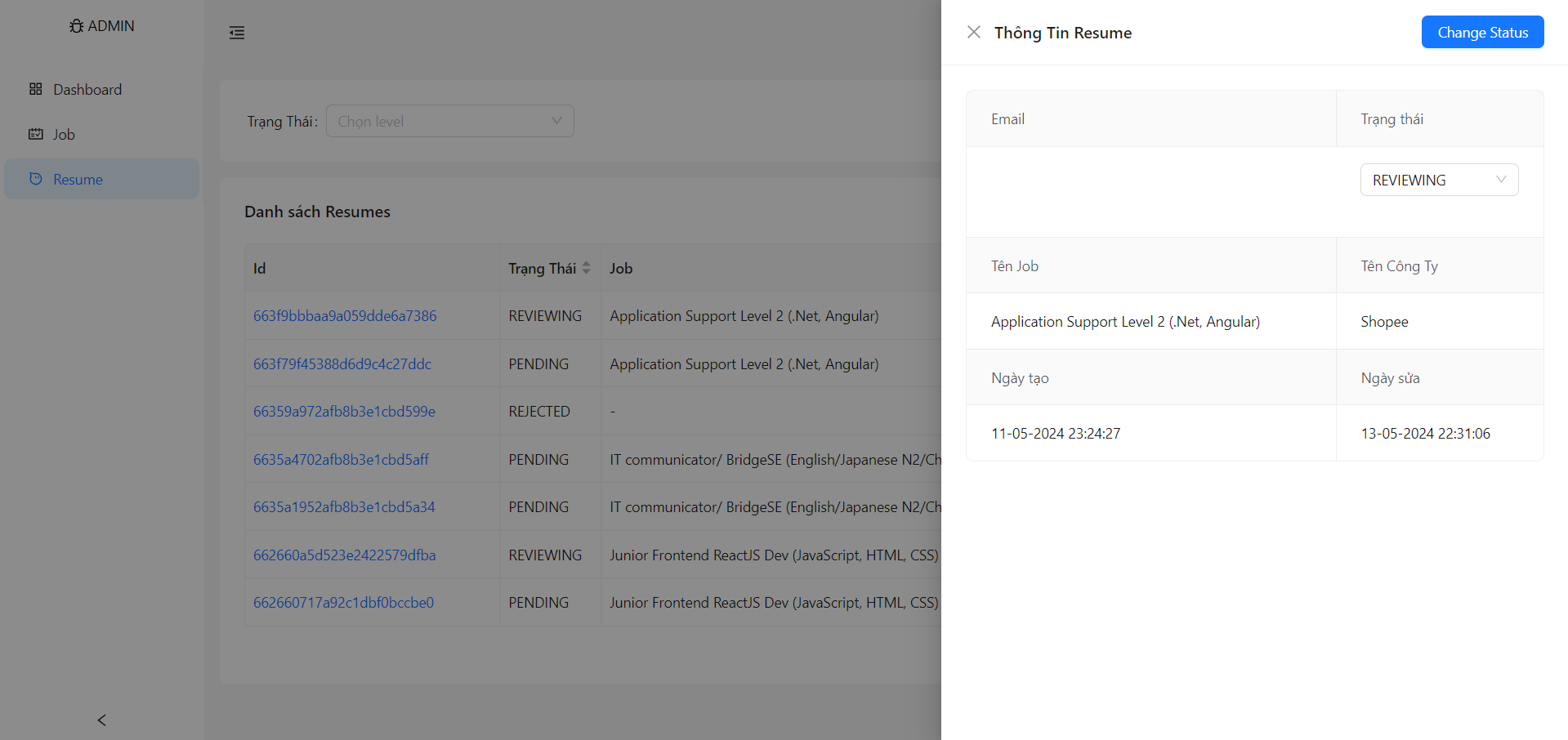


Hinh 37: Giao diện đăng nhập HR

### 4.4.2 Quản lý CV ứng viên

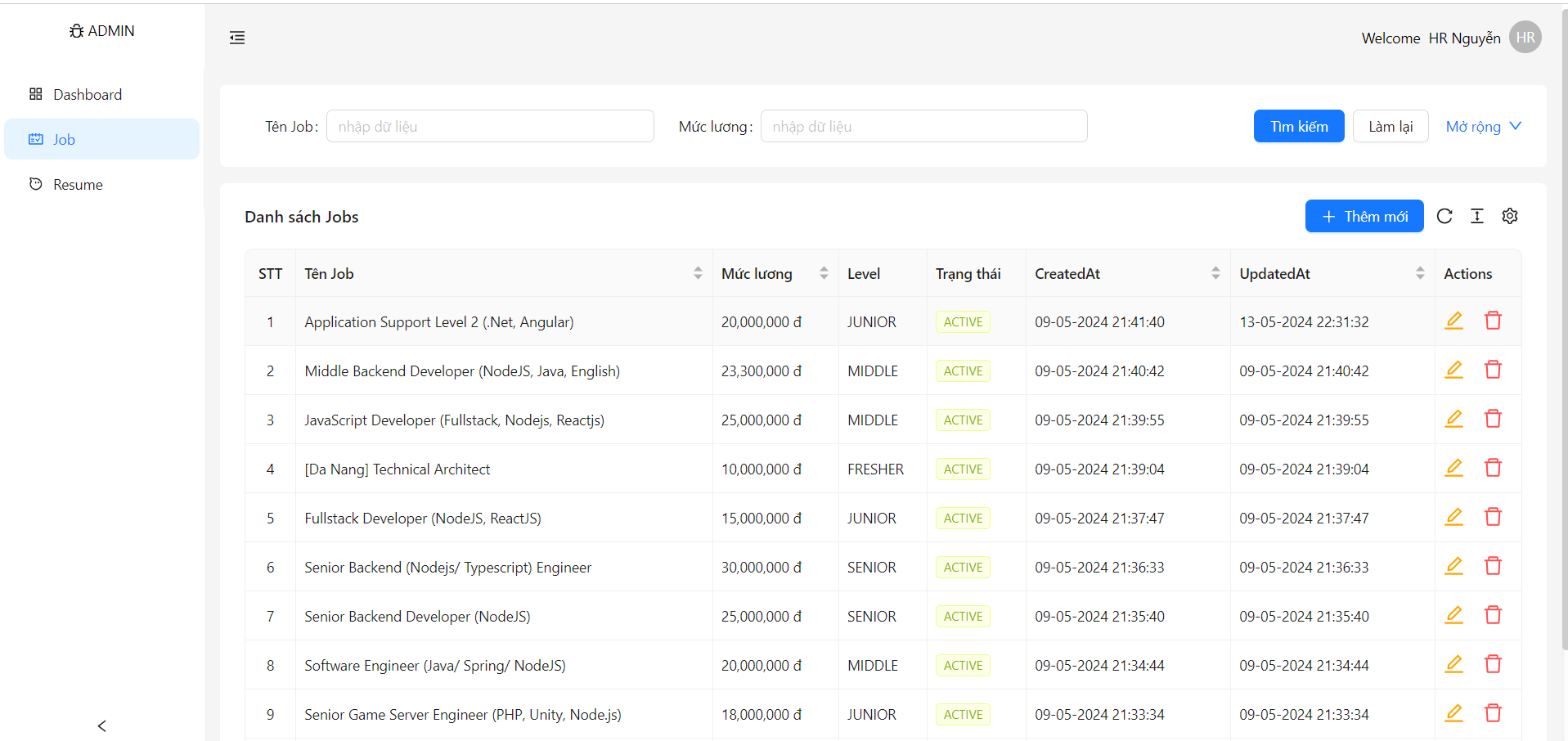


Hinh 38: Quản lý CV ứng viên

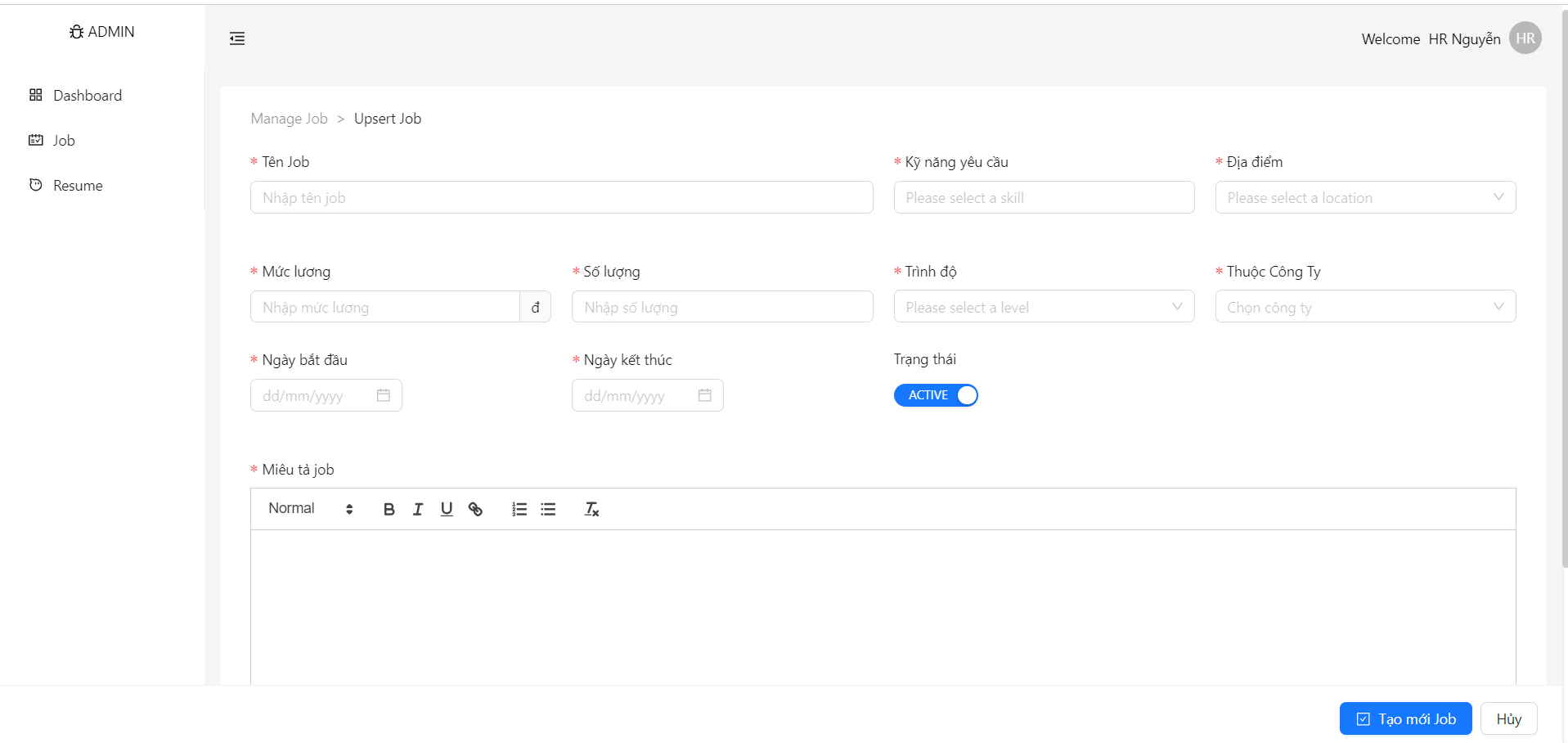


Hinh 39: Cập nhật trạng thái CV ứng viên

### 4.4.3 Quản lý việc làm



Hinh 40: Quản lý việc làm



Hinh 41: Thêm mới việc làm

# KẾT LUẬN

1. **Kết quả đạt được**

Sau một thời gian tìm hiểu cùng với sự giúp đỡ của giảng viên hướng dẫn. Em đã có thể xây dựng hoàn chỉnh cơ sở dữ liệu cho hệ thống đăng tin tuyển dụng, hiểu biết sâu rộng hơn về mô hình Client Rendering phát triển website. Đồng thời, xây dụng được giao diện, chức năng của người quản trị cũng như hoàn thiện được các giao diện, chức năng cho người dùng

Qua thời gian thực hiện đồ án, em đã biết cách thêm các thư viện để hỗ trợ việc lập trình, biết cách kết nối Visual Studio Code và sử MongoDB Compass để quản lý DB trở nên dễ dàng hơn.

Tuy nhiên, do giới hạn về thời gian và còn nhiều hạn chế về mặt kinh nghiệm trong việc phát triển website đăng tin tuyển dụng thực tế nên vẫn còn một số chức năng mà website của em chưa đáp ứng được như mong muốn.

1. **Hướng phát triển**

Hướng phát triển tiếp theo của đề tài là sẽ bổ sung thêm nhiều hình thứccó thể xem được số lượt các CV đã nộp. Bên cạnh đó,nên bổ sung thêm chức năng chat với nhà tuyển dụng và thông báo về cho ứng viên biết xem CV đã nộp có phù hợp hay không. Thêm đánh giá công ty cho người dùng. Ngoài ra sẽ thay đồi một số phần của giao diện website sao cho thân thiện với người dùng hơn.

# TÀI LIỆU THAM KHẢO

[1] https://react.dev

[2 https://nestjs.com

[3] https://ant.design

[4] https://gist.github.com/tkrotoff/b1caa4c3a185629299ec234d2314e190

[5]https://www.mindxmaster.com/best-backend-frameworks-of-2023-for-web-development/#google\_vignette

[6]<https://stackoverflow.com/questions/72184517/sql-vs-nosql-explanation-for-few-points>

[7]<https://stackoverflow.com/questions/56164947/what-is-the-difference-between-vue-js-and-react-js>

[8] https://stackoverflow.com/questions/28151067/angularjs-vs-reactjs