**BỘ CÔNG THƯƠNG**

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP TP.HCM**



**Đề tài: Cong ty tim viec**

***Sinh viên thực hiện***

1. Nguyễn Hoàng Anh - 20060251

# MỤC LỤC

[MỤC LỤC 1](#_Toc185181313)

[DANH MỤC CÁC HÌNH VẼ 3](#_Toc185181314)

[DANH MỤC CÁC BẢNG BIỂU 4](#_Toc185181315)

[CHƯƠNG 1 : GIỚI THIỆU 5](#_Toc185181316)

[1.1 Tổng quan 5](#_Toc185181317)

[1.2 Mục tiêu đề tài 5](#_Toc185181318)

[1.3 Phạm vi đề tài 5](#_Toc185181319)

[1.4 Mô tả yêu cầu chức năng 6](#_Toc185181320)

[CHƯƠNG 2 : CƠ SỞ LÝ THUYẾT 7](#_Toc185181321)

[2.1 Java JDK 21 7](#_Toc185181322)

[2.2 Kafka 8](#_Toc185181323)

[2.3 Redis 10](#_Toc185181324)

[2.4 Kiến trúc microservices 11](#_Toc185181325)

[CHƯƠNG 3 : PHÂN TÍCH VÀ THIẾT KẾ 12](#_Toc185181326)

[3.1 Kiến trúc 12](#_Toc185181327)

[3.2 Phân tích yêu cầu bằng UML 12](#_Toc185181328)

[3.2.1 Usecase tổng quát 12](#_Toc185181329)

[3.2.2 Danh sách tác nhân và mô tả 12](#_Toc185181330)

[3.2.3 Danh sách các tình huống hoạt động (Use cases) 13](#_Toc185181331)

[3.2.4 Tình huống hoạt động 14](#_Toc185181332)

[3.2.4.1 UC01\_Đăng ký 14](#_Toc185181333)

[3.2.4.2 UC02\_Đăng nhập 15](#_Toc185181334)

[3.3 Class diagram 17](#_Toc185181335)

[3.3.1 Entity diagram 18](#_Toc185181336)

[3.3.2 Database diagram 19](#_Toc185181337)

[3.4 Deployment diagram 22](#_Toc185181338)

[CHƯƠNG 4 : HIỆN THỰC 23](#_Toc185181339)

[4.1 Cấu hình phần cứng, phần mềm 23](#_Toc185181340)

[4.2 Giao diện của hệ thống 23](#_Toc185181341)

[4.3 Kế hoạch và hiện thực kiểm thử hệ thống 25](#_Toc185181342)

[4.3.1 Kế hoạch kiểm thử 25](#_Toc185181343)

[4.3.2 Kiểm thử hệ thống 25](#_Toc185181344)

[CHƯƠNG 5 : KẾT LUẬN 28](#_Toc185181345)

[5.1 Kết quả đạt được 28](#_Toc185181346)

[5.2 Hướng phát triển 28](#_Toc185181347)

[TÀI LIỆU THAM KHẢO 29](#_Toc185181348)

# DANH MỤC CÁC HÌNH VẼ

[*11. Usecase tổng quát* 13](#_Toc185172814)

# DANH MỤC CÁC BẢNG BIỂU

[*Bảng 3.1 Danh sách tác nhân và mô tả* 14](#_Toc185172863)

[*Bảng 3.2 Danh sách các tình huống trong hệ thống* 14](#_Toc185172864)

[*4.1 Phần cứng thực hiện kiểm thử* 26](#_Toc185172865)

[*4.2 Phần mềm thực hiện kiểm thử* 26](#_Toc185172866)

[*5.1 Kịch bản kiểm thử.* 26](#_Toc185172867)

[*6. Bảng kết quả kiểm thử.* 27](#_Toc185172868)

# : GIỚI THIỆU

## Tổng quan

## Mục tiêu đề tài

**Hỗ trợ công ty đăng tin tuyển dụng**: Các công ty có thể dễ dàng đăng tải các tin tuyển dụng, bao gồm yêu cầu về kỹ năng, giúp ứng viên biết rõ các yêu cầu của công việc.

**Tối ưu hóa quy trình tuyển dụng: Phát triển hệ thống hỗ trợ công ty đăng tin tuyển dụng trên nhiều nền tảng khác nhau, đồng thời theo dõi và phân tích kết quả để điều chỉnh chiến lược tuyển dụng.**

**Tăng cường tiếp cận ứng viên tiềm năng: Đảm bảo công ty có thể tiếp cận đối tượng ứng viên đa dạng thông qua việc đăng tin tuyển dụng trên các kênh thông dụng, mạng xã hội và các trang web việc làm uy tín.**

## Phạm vi đề tài

Ứng dụng được sử dụng bởi quản trị viên hệ thống, công ty tuyển dụng và ứng viên tìm việc.

Ứng dụng hỗ trợ cho nền tảng web mobile và web app.

Mục dích ứng dụng:

+ Quản trị viên: có thể quản lý các kỹ năng cho một hệ thống.

+ Ứng viên: tiện lợi trong việc tìm kiếm công việc phù hợp với bản thân.

+ Công ty: tìm được ứng viên phù hợp với vị trí mô tả từng công việc.

## Mô tả yêu cầu chức năng

**Yêu cầu chức năng:**

Công ty đăng nhập vào hệ thống.

Công ty điền thông tin mô tả công việc và các kỹ năng yêu cầu.

Hệ thống lưu thông tin và hiển thị tin tuyển dụng cho ứng viên.

Công ty có thể chỉnh sửa hoặc xóa tin tuyển dụng nếu cần.

Ứng viên đăng nhập vào hệ thống.

Ứng viên điền kỹ năng của mình vào hồ sơ và các thông tin.

**Yêu cầu phi chức năng:**

Đảm bảo hoạt động được trên mọi thiết bị.

Thời gian cập nhật dữ liệu dưới 2 giây.

Thời gian hiển thị các danh sách phải dưới 5 giây.

Các thông tin quan trọng phải được hiển thị to và rõ để nhận biết.

# : CƠ SỞ LÝ THUYẾT

## Spring Boot

Spring Boot là một framework Java được sử dụng để xây dựng các ứng dụng và dịch vụ web dễ dàng và nhanh chóng. Nền tảng cung cấp các cấu hình mặc định cho một số thư viện và bộ công cụ hỗ trợ xây dựng, triển khai, quản lý ứng dụng Spring-based.



Công nghệ được ứng dụng rộng rãi

Cách Spring Boot hoạt động nhằm tối ưu hóa quy trình phát triển ứng dụng Java. Điều này sẽ giúp nhà phát triển tập trung vào việc xây dựng tính năng chính của ứng dụng mà không cần phải lo lắng về cấu hình phức tạp.

## Java JDK 17

**Vào tháng 9 năm 2021, "Java Platform, SE 17 Development Kit (JDK 17)" hay Java 17 đã được phát hành, bao gồm các tính năng mới, bao gồm các cải tiến trên nhiều lĩnh vực chức năng. Trong bản phát hành này, Java tiếp tục quá trình phát triển của mình.**

* **Sau đây là một số cập nhật:**
* **Các lớp học kín được chuẩn hóa.**
* **Ngữ nghĩa dấu phẩy động luôn nghiêm ngặt đã được khôi phục.**
* **Các câu lệnh chuyển đổi được làm phong phú hơn với khả năng khớp mẫu tốt hơn.**
* **Máy tạo số giả ngẫu nhiên đã được nâng cấp.**
* **Hỗ trợ cho các bộ lọc hủy tuần tự hóa theo ngữ cảnh cụ thể và được chọn động.**
* **Cơ chế kích hoạt RMI đã bị xóa.**
* **Security Manager và Applet API nổi tiếng đã lỗi thời và sẽ bị xóa/lên lịch xóa.**



*Hình 1 JDK 17*

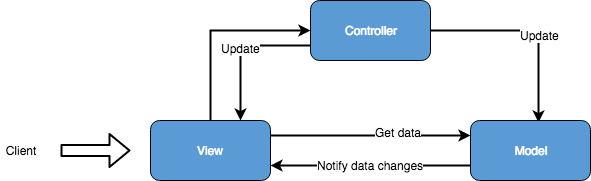
## MariaDB

MariaDB là một trong những hệ quản trị cơ sở dữ liệu phổ biến nhất trên thế giới, MariaDB được tạo bởi các nhà phát triển MySQL ban đầu và được bảo đảm bản chất vẫn luôn là mã nguồn mở.



API và giao thức của MariaDB tương thích với API của MySQL, ngoài việc kế thừa các chức năng cốt lõi của MySQL thì MariaDB cung cấp thêm nhiều tính năng cải tiến về cơ chế lưu trữ, tối ưu tốc độ.

## Kiến trúc MVC



MVC (Model-View-Controller) là một mẫu kiến trúc phần mềm được sử dụng rộng rãi trong việc phát triển ứng dụng để tạo ra các giao diện người dùng trực quan và có khả năng tương tác cao. MVC chia một ứng dụng thành 3 phần chính và mỗi phần có một vai trò riêng biệt:

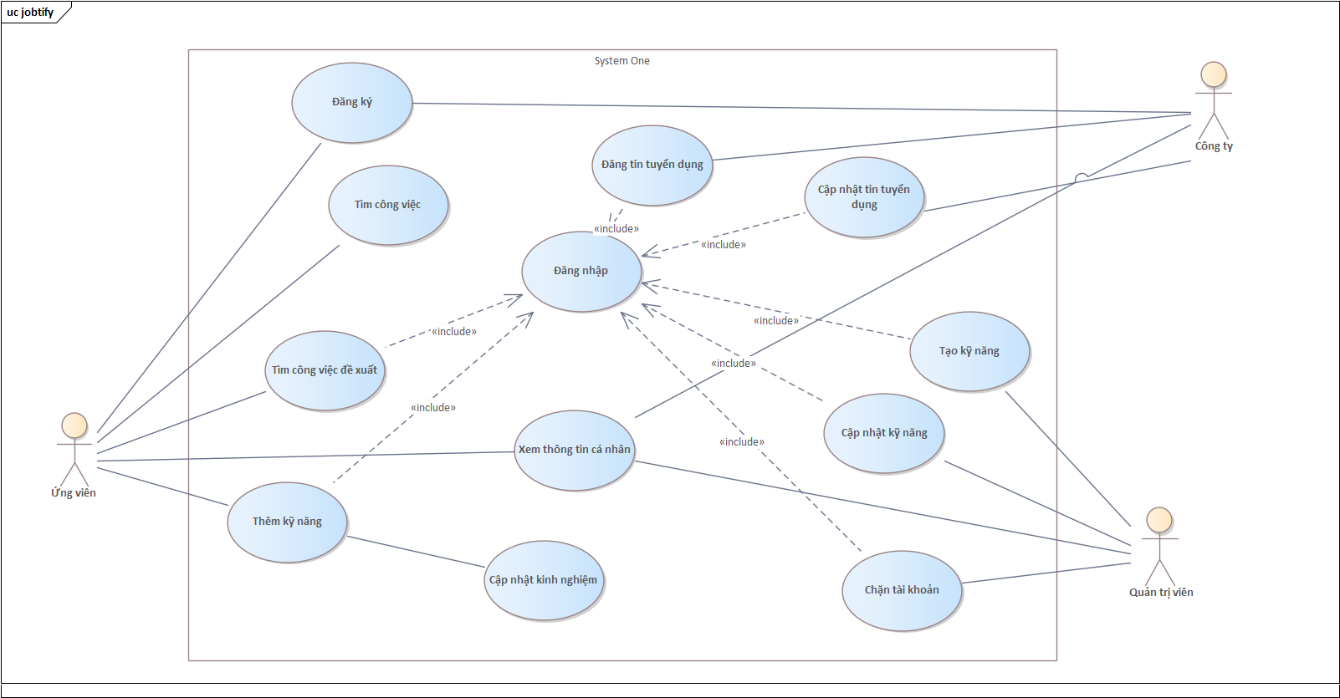
* Model đại diện cho dữ liệu và quy tắc nghiệp vụ của ứng dụng.
* View chịu trách nhiệm hiển thị dữ liệu cho người dùng một cách trực quan và tương tác.
* Controller đóng vai trò là cầu nối giữa Model và View, xử lý các yêu cầu từ người dùng và cập nhật giao diện tương ứng.

# : PHÂN TÍCH VÀ THIẾT KẾ

## Kiến trúc

## Phân tích yêu cầu bằng UML

### Usecase tổng quát



*11. Usecase tổng quát*

### Danh sách tác nhân và mô tả

Danh sách tác nhân và mô tả xem ở Bảng 3-1.

|  |  |
| --- | --- |
| Tác nhân | Mô tả tác nhân |
| Quản trị viên | Thực hiện quản lý các kỹ năng cơ bản và người dùng hệ thống. |
| Ứng viên | Xem thông tin công việc, công việc gợi ý được các công ty đăng bài tuyển dụng, quản lý thông tin kỹ năng cá nhân, địa chỉ. |
| Công ty | Đăng bài tuyển dụng, các yêu cầu kỹ năng cho một công việc. |

*Bảng 3.1 Danh sách tác nhân và mô tả*

### Danh sách các tình huống hoạt động (Use cases)

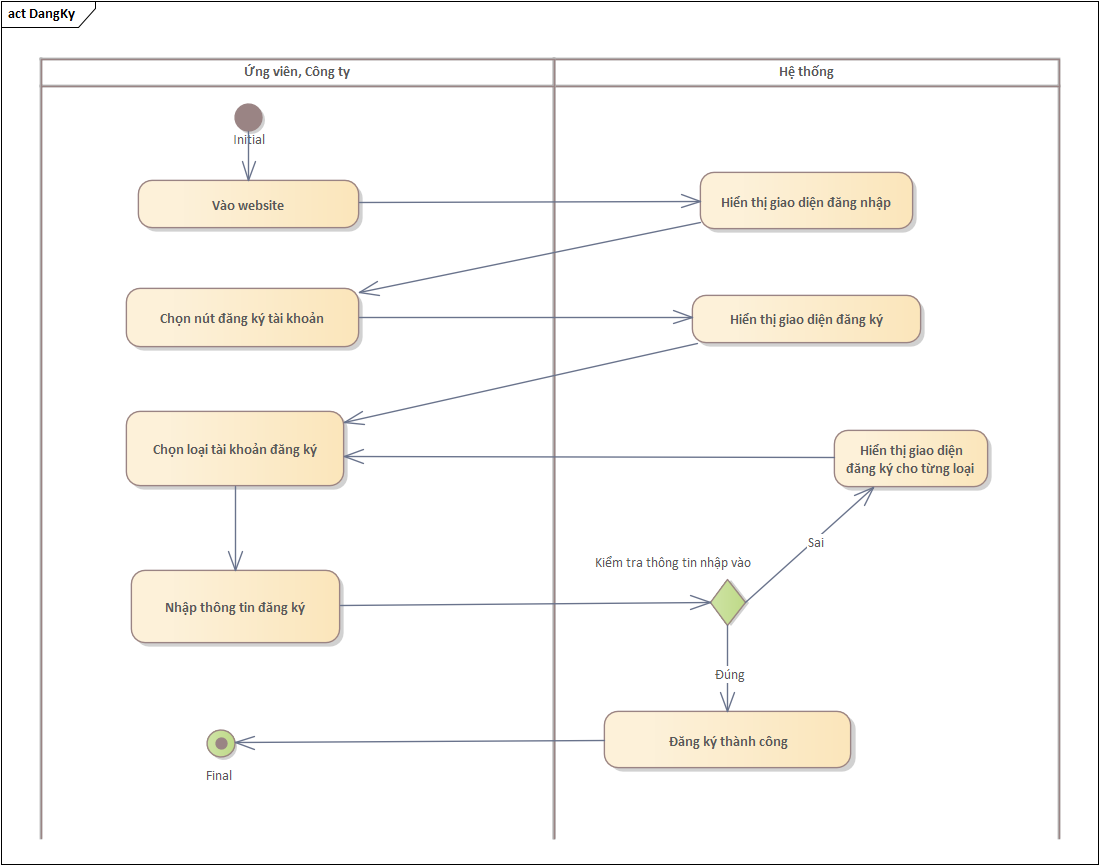
|  |  |
| --- | --- |
| ID | Tên Use case |
| UC01 | Đăng ký |
| UC02 | Đăng nhập |
| UC03 | Quên mật khẩu |
| UC04 | Đăng xuất |
| UC05 | Đăng tin tuyển dụng |
| UC06 | Cập nhật tin tuyển dụng |
| UC07 | Tạo kỹ năng cơ bản |

*Bảng 3.2 Danh sách các tình huống trong hệ thống*

### Tình huống hoạt động

#### UC01\_Đăng ký

|  |  |
| --- | --- |
| * **Tên use case**: UC01\_Đăng ký | |
| * **Mô tả sơ lược**: Đăng ký tài khoản của khách hàng trước khi sử dụng các dịch vụ của RecyclePro | |
| * **Actor chính**: Khách hàng | |
| * **Actor phụ**: Không có | |
| * **Tiền điều kiện (Pre-condition):** Chưa có tài khoản sử dụng | |
| * **Hậu điều kiện (Post-condition):** Tài khoản đăng ký thành công sẽ được lưu vào cơ sở dữ liệu. | |
| * **Luồng sự kiện chính (main flow):** | |
| **Khách hàng** | **Hệ thống** |
| 1. Vào trang chủ | 1. Hiển thị giao diện đăng nhập. |
| 1. Chọn nút đăng ký tài khoản. | 1. Hiển thị giao diện đăng ký tài khoản. |
| 1. Nhập các thông tin: tên người dùng(username), địa chỉ mail (email), mật khẩu(password) và nhập lại mật khẩu (confirm password), loại tài khoản(type), địa chỉ (address), địa chỉ website (webURL) |  |
| 1. Nhấn vào nút đăng ký (signup) | 1. Kiểm tra các thông tin nhập vào. Nếu hợp lệ thông báo thành công. |
| 1. Xác nhận |  |
| * **Luồng sự kiện thay thế (alternate flow):** | |
|  | 7.1. Nếu không hợp lệ thông báo trường nhập vào nào bị lỗi. |
| 7.2. Người dùng nhập lại thông tin. |  |
| 7.3. Quay lại bước 7. |  |



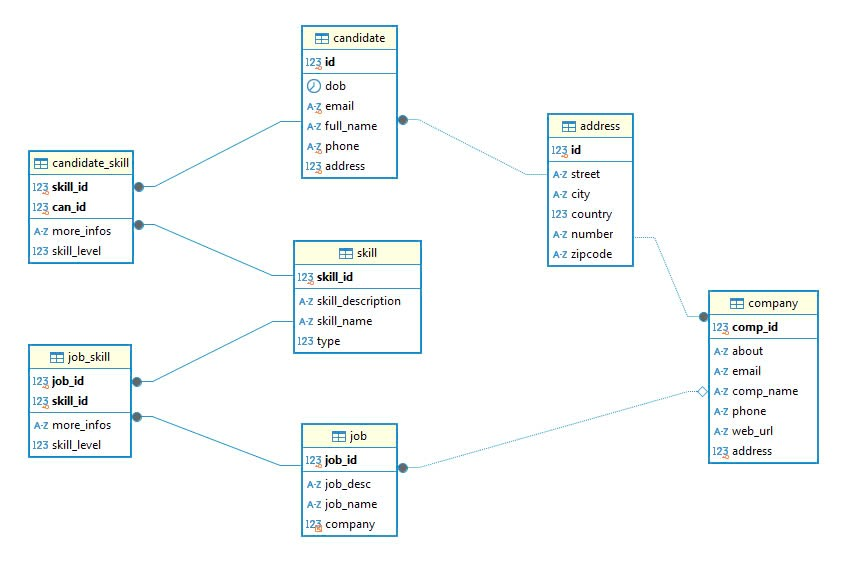
*1. Sơ đồ hoạt động đăng ký*

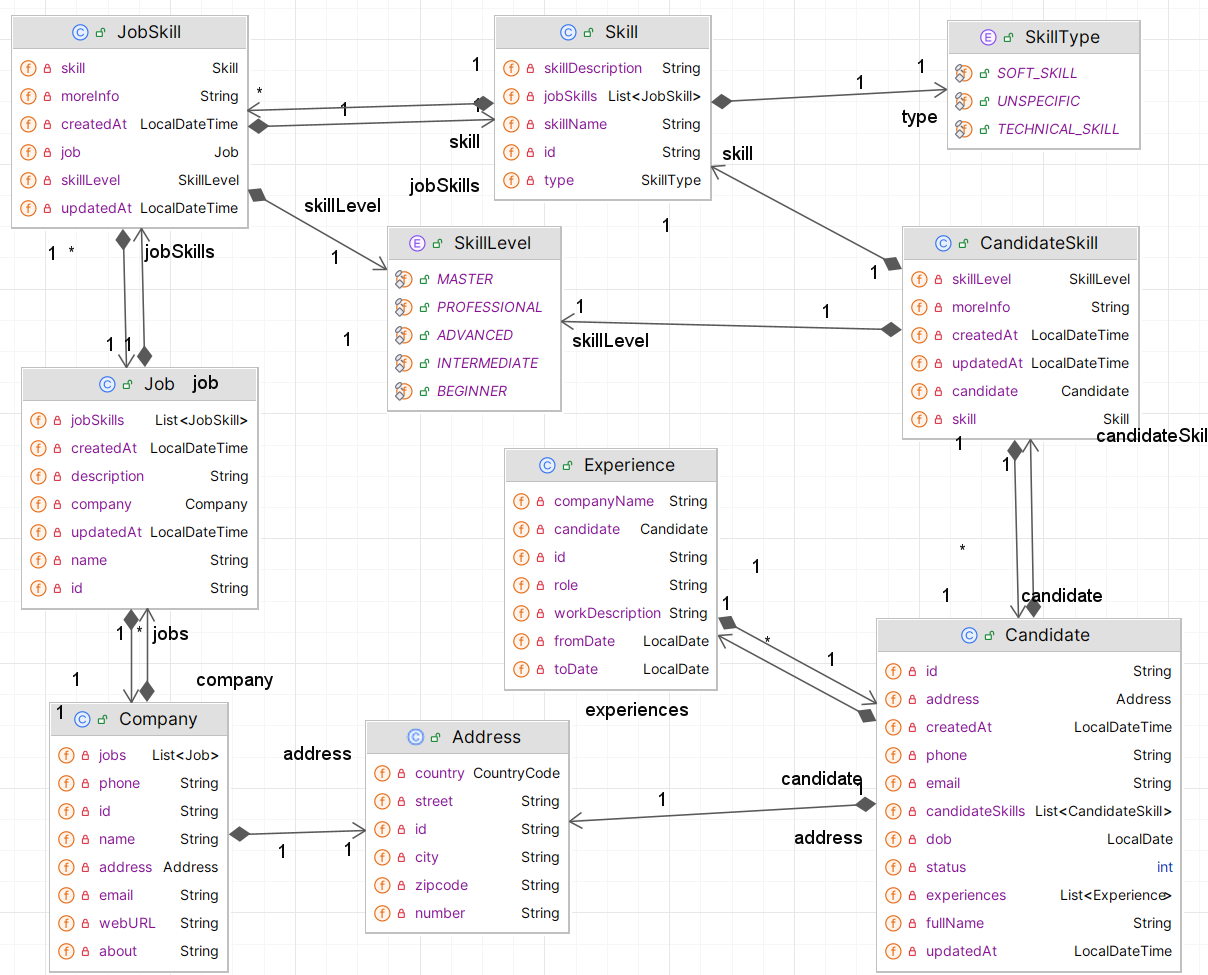
#### UC02\_Đăng nhập

|  |  |
| --- | --- |
| * **Tên use case**: UC02\_Đăng nhập | |
| * **Mô tả sơ lược**: Người dùng muốn đăng nhập vào hệ thống để sử dụng dịch vụ | |
| * **Actor chính**: Ứng viên, Công ty, Quản trị viên | |
| * **Actor phụ**: Không có | |
| * **Tiền điều kiện (Pre-condition):** Kết nối internet, người dùng đã có tài khoản | |
| * **Hậu điều kiện (Post-condition):** | |
| * **Luồng sự kiện chính (main flow):** | |
| **Khách hàng** | **Hệ thống** |
| 1. Truy cập website | 1. Hiển thị giao diện đăng nhập |
| 1. Nhập vào các thông tin: username (username), mật khẩu (password) | 1. Kiểm tra thông tin nhập vào. |
|  | 1. Kiểm tra vai trò |
|  | 1. Thông báo đăng nhập thành công |
| * **Luồng sự kiện thay thế (alternate flow):** | |
|  | 4.1. Thông báo thông tin đăng nhập không đúng. |
|  | 4.2. Quay lại bước 2 |
|  | 5.1. Thông báo vai trò không hợp lệ |
|  | 5.2. Quay lại bước 2 |

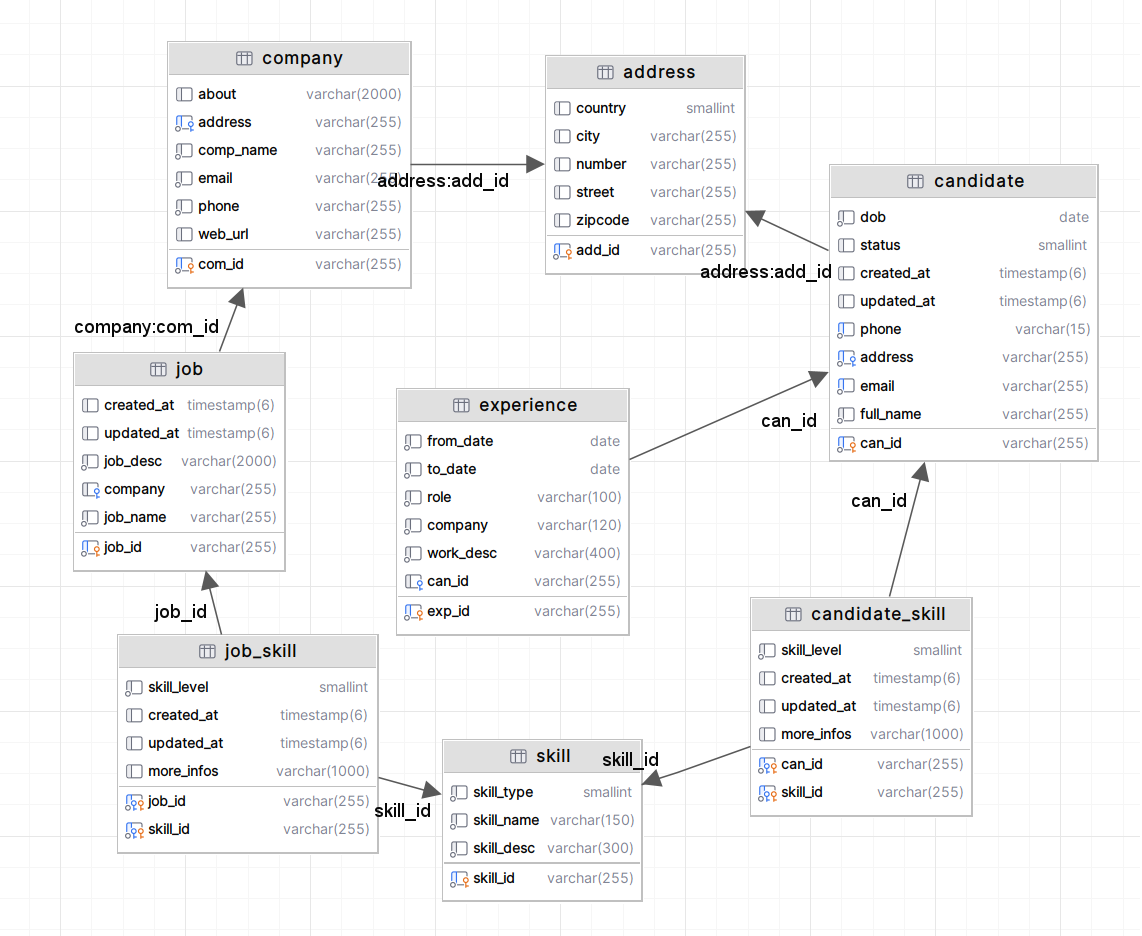
*2. Sơ đồ hoạt động đăng nhập.*

## Class diagram





### Database diagram



# : HIỆN THỰC

## Cấu hình phần cứng, phần mềm

Thiết bị sử dụng windows 10 trở lên

## Kế hoạch và hiện thực kiểm thử hệ thống

### Kế hoạch kiểm thử

*4.1 Phần mềm thực hiện kiểm thử*

|  |  |  |
| --- | --- | --- |
| **Tên phần mềm** | **Phiên bản** | **Loại** |
| IntelliJ Ultimate | 2024.2.2 | IDE cho java |
| Microsoft Windows 11 | 11 22H2 | Hệ điều hành |

# : KẾT LUẬN

## Kết quả đạt được

Giao diện đăng ký, đăng nhập

Hệ thống gợi ý công việc cho ứng viên

Cập nhật thông tin người dùng

## Hướng phát triển

Bổ sung giao diện cho ứng viên và công ty.

Bổ sung tính năng gửi email cho ứng viên khi tìm thấy công việc phù hợp.