

**HỆ THỐNG QUẢN LÝ TUYỂN DỤNG BẰNG SPRING BOOT**

Họ tên: Đinh Nguyên Chung

MSSV: 21127891

Giảng Viên: Võ Văn Hải

Mục Lục

[1. Giới thiệu 3](#_Toc184988779)

[2. Phân tích yêu cầu 3](#_Toc184988780)

[Yêu cầu chức năng: 3](#_Toc184988781)

[Quản lý hồ sơ ứng viên: 3](#_Toc184988782)

[Quản lý công việc: 3](#_Toc184988783)

[Quản lý người dùng: 4](#_Toc184988784)

[Yêu cầu phi chức năng: 4](#_Toc184988785)

[3. Thiết kế hệ thống 4](#_Toc184988786)

[Kiến trúc hệ thống: 4](#_Toc184988787)

[Sơ đồ ERD: 5](#_Toc184988788)

[4. Triển khai 5](#_Toc184988789)

[Cấu trúc thư mục dự án: 5](#_Toc184988790)

[Các API chính: 6](#_Toc184988791)

[Cơ sở dữ liệu: 6](#_Toc184988792)

[5. Kết quả 7](#_Toc184988793)

[Kết quả đạt được: 7](#_Toc184988794)

# 1. Giới thiệu

**Tên dự án:** Hệ thống Quản lý Tuyển dụng

**Lý do thực hiện:** Trong xu hướng số hóa và tự động hóa quy trình làm việc, nhiều doanh nghiệp gắp khó khăn trong việc quản lý các quy trình tuyển dụng. Việc sử dụng hệ thống quản lý tuyển dụng giúp tối ưu hóa quy trình từ khâu tiếp nhận hồ sơ, lọc ứng viên, đến theo dõi quá trình phỏng vấn và tuyển chọn. Dự án này nhằm tạo ra một hệ thống giúp doanh nghiệp quản lý hiệu quả, nhanh chóng và minh bạch.

**Mục tiêu:**

* Xây dựng hệ thống quản lý tuyển dụng bao gồm quản lý hồ sơ ứng viên, quản lý công việc.
* Tối ưu hóa quy trình tuyển dụng và giảm thiểu thời gian xử lý.
* Cung cấp giao diện người dùng dễ sử dụng.

**Công cụ và công nghệ sử dụng:**

Java, Spring Boot, MariaDB, IntelliJ IDEA, Bootstrap.

# 2. Phân tích yêu cầu

## Yêu cầu chức năng:

### Quản lý hồ sơ ứng viên:

* + Thêm, sửa, xóa và xem danh sách ứng viên.
  + Lọc ứng viên theo trình độ học vấn, kinh nghiệm, vị trí mong muốn.

### Quản lý công việc:

* + Thêm, sửa, xóa công việc tuyển dụng.

### Quản lý người dùng:

* + Phân quyền người dùng: Quản trị và nhân viên.

## Yêu cầu phi chức năng:

* Hệ thống phải an toàn và đảm bảo tính bảo mật.
* Tối ưu hóa tốc độ xử lý.
* Giao diện người dùng dễ sử dụng và thân thiện.

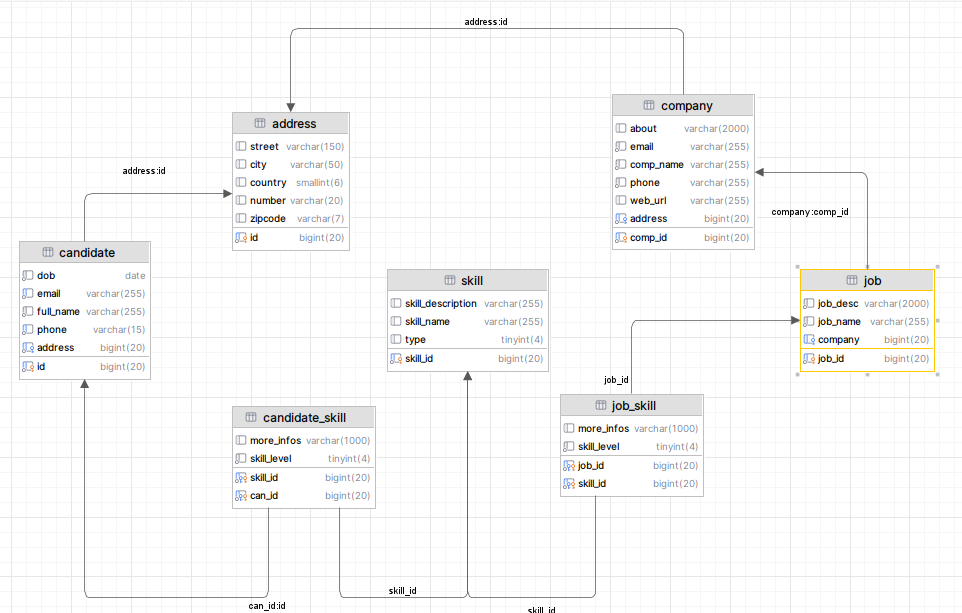
# 3. Thiết kế hệ thống

## Kiến trúc hệ thống:

Hệ thống sử dụng mô hình MVC:

* **Model:** Các entity như Candidate, Job, User.
* **View:** Giao diện người dùng được xây dựng bằng HTML, CSS, Bootstrap.
* **Controller:** Xử lý các request HTTP và giao tiếp giữa Model và View.

## Sơ đồ ERD:

****

Minh họạ các quy trình:

* Ứng viên tìm bài tuyển dụng và thêm hồ sơ.
* Quản trị phê duyệt và lọc hồ sơ.
* Nhân viên cập nhật trạng thái tuyển dụng.

# 4. Triển khai

## Cấu trúc thư mục dự án:

* src/main/java: Chứa các file Java.
* src/main/resources: Các file cấu hình (đặc biệt là application.properties).
* templates: Chứa các file giao diện JSP/HTML.

## Các API chính:

1. **Candidate Management:**
   * **GET** /candidates: Lấy danh sách ứng viên.
   * **POST** /candidates: Thêm mới ứng viên.
   * **PUT** /candidates/{id}: Cập nhật thông tin ứng viên.
   * **DELETE** /candidates/{id}: Xóa ứng viên.
2. **Job Management:**
   * **GET** /jobs: Lấy danh sách công việc.
   * **POST** /jobs: Thêm mới công việc.

## Cơ sở dữ liệu:

* **Bảng Candidate:**
  + Trường: id, fullname, email, dod, phone, address.
* **Bảng Job:**
  + Trường: id, name, description, company.
* **Bảng address:** 
  + Trường: id, street, city, country, number, zipcodes.
* **Bảng company:**
  + Trường: id, about, name, email,phone, url, address.
* **Bảng skill:**
  + Trường: name, description, type.

# 5. Kết quả

## Kết quả đạt được:

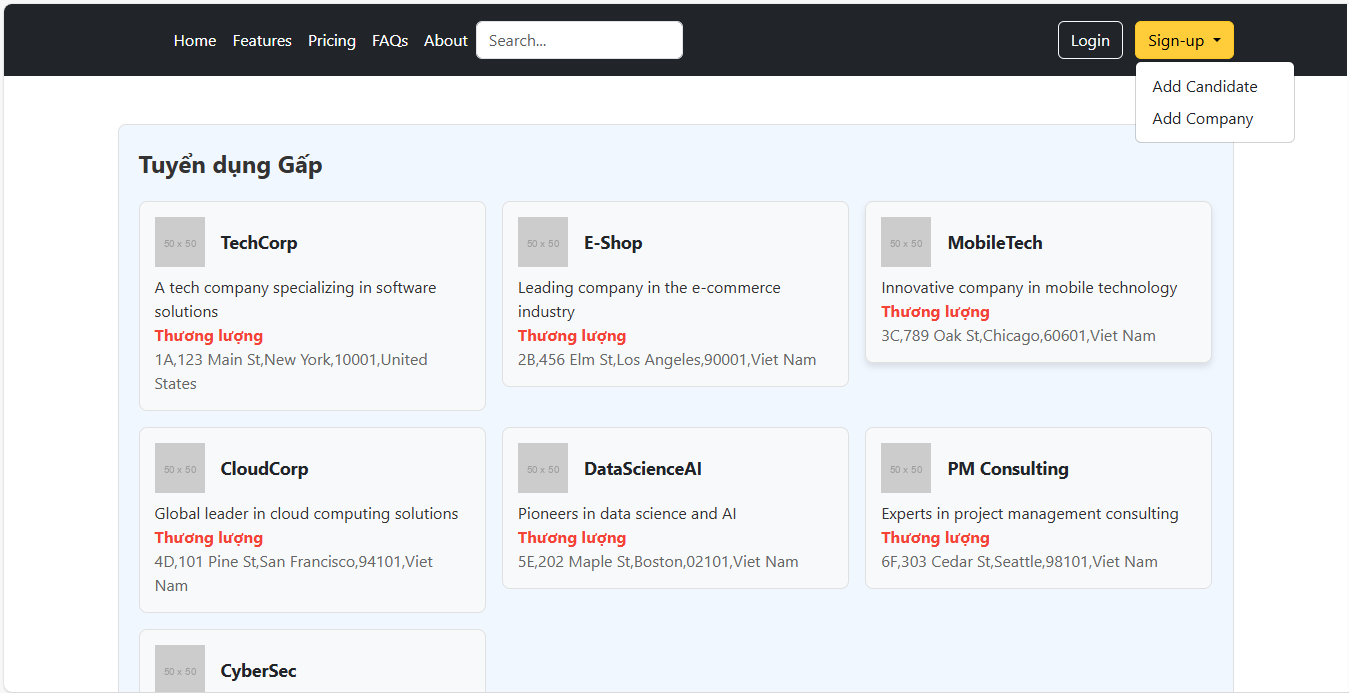
* Hệ thống đã hoàn thiện các chức năng quản lý tuyển dụng.
* Giao diện thân thiện, cho phép người dùng thực hiện các tác vụ như thêm ứng viên, thêm bài tuyển dụng.
* Tối ưu thời gian xử lý và lưu trữ dữ liệu.

**Hạn chế:**

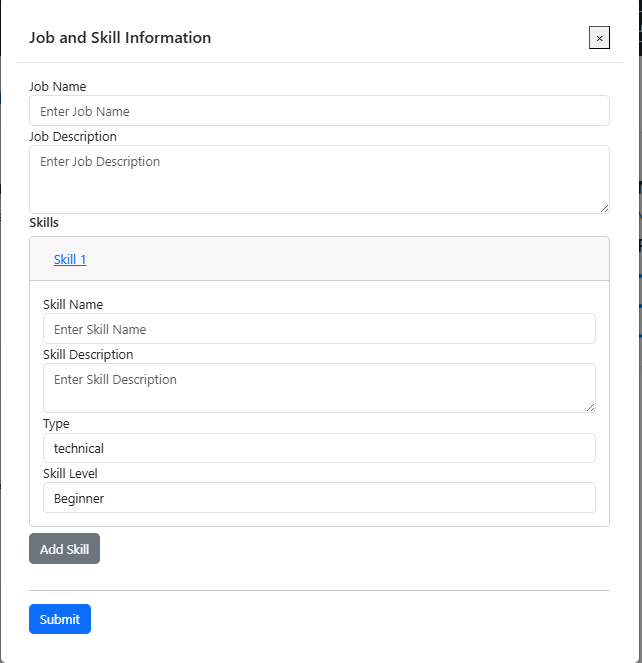
* Hệ thống chưa tích hợp email tự động cho ứng viên.
* Cần thêm các biểu đồ thống kê về quá trình tuyển dụng.

**Ảnh chụp màn hình một số giao diện chính:**

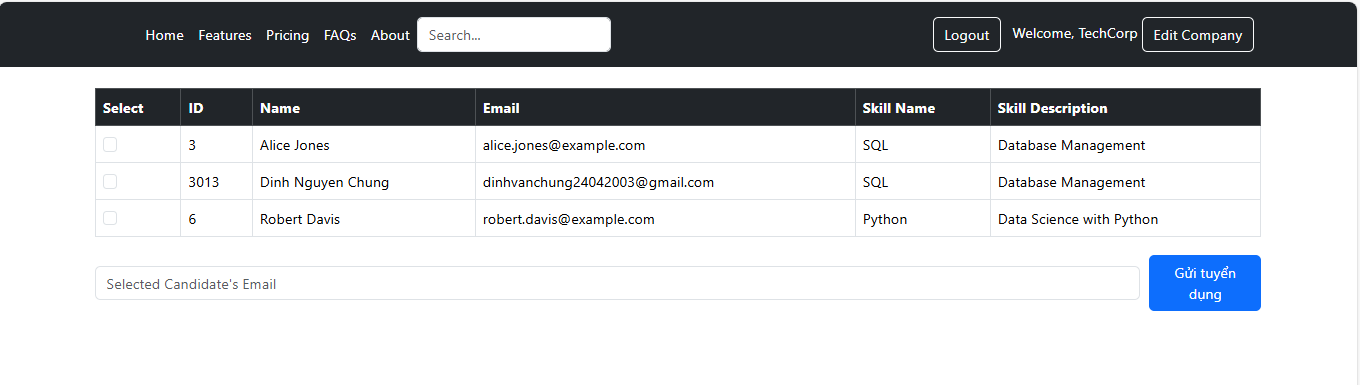
* Giao diện bài tuyển dụng.



* Giao diện thêm công việc mới.



Giao diện tìm được Candidates:



Giao diện Candidates tìm được công việc dựa vào skills:

