**MỤC LỤC**

[**LỜI CẢM ƠN** 3](#_Toc134924141)

[**DANH MỤC HÌNH ẢNH** 4](#_Toc134924142)

[**DANH MỤC BẢNG BIỂU** 5](#_Toc134924143)

[**CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI** 1](#_Toc134924144)

[**1.1.** **Lý do chọn đề tài** 1](#_Toc134924145)

[**1.2.** **Mục tiêu và phạm vi nghiên cứu** 2](#_Toc134924146)

[**CHƯƠNG 2. CƠ SỞ LÝ THUYẾT VỀ CÁC CÔNG NGHỆ SỬ DỤNG** 3](#_Toc134924147)

[**2.1.** **Javascript** 3](#_Toc134924148)

[**2.1.1.** **JavaScript là gì?** 3](#_Toc134924149)

[**2.1.2.** **JavaScript trong trình duyệt có thể làm gì?** 3](#_Toc134924150)

[**2.1.3.** **JavaScript trong trình duyệt không thể làm gì?** 4](#_Toc134924151)

[**2.1.4.** **Languages “over” JavaScript** 5](#_Toc134924152)

[**2.2.** **Typescript** 6](#_Toc134924153)

[**2.2.1.** **Typescript là gì?** 6](#_Toc134924154)

[**2.2.2.** **Đặc điểm của Typescript** 6](#_Toc134924155)

[**2.2.3.** **So sánh giữa Typescript và Javascript** 7](#_Toc134924156)

[**2.2.4.** **Vì sao chọn Typescript?** 8](#_Toc134924157)

[**2.3.** **NestJS** 8](#_Toc134924158)

[**2.3.1.** **NestJS là gì** 8](#_Toc134924159)

[**2.3.2.** **Một số đặc điểm của NestJS** 8](#_Toc134924160)

[**2.3.3.** **Ba thành phần chính của NestJs** 9](#_Toc134924161)

[**2.3.4.** **Vì sao lựa chọn NestJS cho ứng dụng Next.js** 10](#_Toc134924162)

[**2.4.** **Next.js** 10](#_Toc134924163)

[**2.4.1.** **Next.js là gì** 10](#_Toc134924164)

[**2.4.2.** **Tính năng của Next.js** 11](#_Toc134924165)

[**2.4.3.** **Next.js và SEO** 12](#_Toc134924166)

[**2.4.4.** **Vì sao sử dụng Next.js** 12](#_Toc134924167)

[**2.5.** **PostgreSQL** 13](#_Toc134924168)

[**2.5.1.** **PostgreSQL là gì** 13](#_Toc134924169)

[**2.5.2.** **Tính năng nổi bật của PostgreSQL** 14](#_Toc134924170)

[**2.5.3.** **Vì sao sử dụng PostgreSQL** 14](#_Toc134924171)

[**CHƯƠNG 3. PHÂN TÍCH THIẾT KẾ VÀ XÂY DỰNG CÔNG CỤ TÌM KIẾM ĐỊA ĐIỂM CHO SINH VIÊN** 17](#_Toc134924172)

[**3.1.** **Phân tích thiết kế hệ thống** 17](#_Toc134924173)

[**3.1.1.** **Biểu đồ hoạt động** 17](#_Toc134924174)

[**3.1.2.** **Biểu đồ ca sử dụng** 22](#_Toc134924175)

[**3.1.3.** **Biểu đồ tuần tự** 26](#_Toc134924176)

[**3.2.** **Thiết kế cơ sở dữ liệu** 39](#_Toc134924177)

[**3.2.1.** **Thiết kế cơ sở dữ liệu mức logic** 39](#_Toc134924178)

[**3.2.2.** **Thiết kế cơ sở dữ liệu mức vật lý** 39](#_Toc134924179)

[**3.3.** **Thiết kế giao diện và xây dựng ứng dụng** 43](#_Toc134924180)

[**3.3.1.** **Giao diện trang chủ** 43](#_Toc134924181)

[**3.3.2.** **Giao diện tất cả việc làm** 44](#_Toc134924182)

[**3.3.3.** **Giao diện chi tiết việc làm** 44](#_Toc134924183)

[**3.3.4.** **Giao diện đăng nhập cho ứng viên** 45](#_Toc134924184)

[**3.3.5.** **Giao diện đăng ký cho ứng viên** 45](#_Toc134924185)

[**3.3.6.** **Giao diện quản lý thông tin ứng viên đã ứng tuyển** 46](#_Toc134924186)

[**3.3.7.** **Giao diện đăng tin tuyển dụng** 46](#_Toc134924187)

[**3.3.8.** **Giao diện quản lý việc làm** 47](#_Toc134924188)

[**3.3.9.** **Giao diện hồ sơ công ty** 47](#_Toc134924189)

[**3.3.10.** **Giao diện đổi mật khẩu** 48](#_Toc134924190)

# **LỜI CẢM ƠN**

Lời đầu tiên em xin trân thành cảm ơn quý thầy cô đã giúp em thực hiện đề tài này. Đặc biệt thầy … đã tận tình hướng dẫn, giúp đỡ, chỉ bảo em trong suốt thời gian thực hiện khóa luận tốt nghiệp.

Đồng thời em xin trân trọng cảm ơn những tình cảm quý báu mà các thầy cô trong trường Đại Học Tài Nguyên Và Môi Trường Hà Nội truyền đạt cho em những những kinh nghiệm, kỹ thuật và cách thức trong việc xây dựng đề tài này.

Tuy nhiên do thời gian còn hạn chế em chưa thể phát huy hết những ý tưởng, khả năng hỗ trợ của ngôn ngữ lập trình và kỹ thuật lập trình vào đề tài này. Trong thời gian thực hiện đề tài không tránh khỏi những thiếu xót, Em mong nhận được sự đóng góp từ quý thầy cô và các bạn.

Em xin chân thành cảm ơn.

Hà Nội, ngày … tháng … năm ….

Sinh viên thực thiện

Trần Thị Hương Giang

# **DANH MỤC HÌNH ẢNH**

[**Hình 3.1. Biểu đồ hoạt động chức năng tìm kiếm việc làm** 17](file:///D:\Đồ%20án\Giang\Báo%20Cáo.docx#_Toc134924042)

[**Hình 3.2. Biểu đồ hoạt động chức năng đăng ký cho ứng viên** 18](#_Toc134924043)

[**Hình 3. 3. Biểu đồ hoạt động chức năng ứng tuyển** 19](file:///D:\Đồ%20án\Giang\Báo%20Cáo.docx#_Toc134924044)

[**Hình 3. 4. Biểu đồ hoạt động chức năng thêm/xóa/sửa tin tuyển dụng** 20](#_Toc134924045)

[**Hình 3. 5. Biểu đồ hoạt động chức năng ứng tuyển** 21](#_Toc134924046)

[**Hình 3. 6. Biểu đồ hoạt động chức năng đổi mật khẩu** 21](#_Toc134924047)

[**Hình 3. 7. Biểu đồ ca sử dụng tổng quát** 22](#_Toc134924048)

[**Hình 3. 8. Biểu đồ ca sử dụng tìm kiếm tin tuyển dụng** 22](#_Toc134924049)

[**Hình 3. 9. Biểu đồ ca sử dụng cập nhật học vấn** 23](#_Toc134924050)

[**Hình 3. 10. Biểu đồ ca sử dụng cập nhật kinh nghiệm** 23](#_Toc134924051)

[**Hình 3. 11. Biểu đồ ca sử dụng cập nhật thông tin cá nhân** 24](#_Toc134924052)

[**Hình 3. 12. Biểu đồ ca sử dụng cập nhật hồ sơ công ty** 24](#_Toc134924053)

[**Hình 3. 13. Biểu đồ ca sử dụng quản lý tin tuyển dụng** 25](#_Toc134924054)

[**Hình 3. 14. Biểu đồ ca sử dụng chấp nhận / từ chối ứng viên** 25](#_Toc134924055)

[**Hình 3. 15. Biểu đồ tuần tự đăng nhập tài khoản nhà tuyển dụng** 26](#_Toc134924056)

[**Hình 3. 16. Biểu đồ tuần tự đăng ký tài khoản** 27](#_Toc134924057)

[**Hình 3. 17. Biểu đồ tuần tự thêm học vấn** 28](#_Toc134924058)

[**Hình 3. 18. Biểu đồ tuần tự chỉnh sửa học vấn** 29](#_Toc134924059)

[**Hình 3. 19. Biểu đồ tuần tự xóa học vấn** 30](#_Toc134924060)

[**Hình 3. 20. Biểu đồ tuần tự chức năng chỉnh sửa hồ sơ** 31](#_Toc134924061)

[**Hình 3. 21. Biểu đồ tuần tự thêm kinh nghiệm làm việc** 32](#_Toc134924062)

[**Hình 3. 22. Biểu đồ tuần tự xóa kinh nghiệm làm việc** 33](#_Toc134924063)

[**Hình 3. 23. Biểu đồ tuần tự chức năng chỉnh sửa kinh nghiệm** 34](#_Toc134924064)

[**Hình 3. 24. Biểu đồ tuần tự cập nhật hồ sơ công ty** 35](#_Toc134924065)

[**Hình 3. 25. Biểu đồ tuần tự đăng tin tuyển dụng** 36](#_Toc134924066)

[**Hình 3. 26. Biểu đồ tuần tự chỉnh sửa tin tuyển dụng** 37](#_Toc134924067)

[**Hình 3. 27. Biểu đồ tuần tự xóa tin tuyển dụng** 38](#_Toc134924068)

[**Hình 3. 28. Mô hình ERD** 39](#_Toc134924069)

[**Hình 3. 29. Giao diện trang chủ** 43](#_Toc134924070)

[**Hình 3. 30. Giao diện trang tất cả việc làm** 44](#_Toc134924071)

[**Hình 3. 31. Giao diện trang chi tiết việc làm** 44](#_Toc134924072)

[**Hình 3. 32. Giao diện đang nhập cho ứng viên** 45](#_Toc134924073)

[**Hình 3. 33. Giao diện đăng ký cho ứng viên** 45](#_Toc134924074)

[**Hình 3. 34. Giao diện quản lý ứng viên 46**](#_Toc134924075)

[**Hình 3. 35. Giao diện đăng tin tuyển dụng** 46](#_Toc134924076)

[**Hình 3. 36. Giao diện quản lý tin tuyển dụng** 47](#_Toc134924077)

[**Hình 3. 37. Giao diện cập nhật hồ sơ công ty** 47](#_Toc134924078)

[**Hình 3. 38. Giao diện đổi mật khẩu** 48](#_Toc134924079)

# **DANH MỤC BẢNG BIỂU**

[**Bảng 3. 1. Bảng user** 41](#_Toc134455472)

[**Bảng 3. 2. Bảng company** 41](#_Toc134455473)

[**Bảng 3. 3. Bảng job** 42](#_Toc134455474)

[**Bảng 3. 4. Bảng education** 42](#_Toc134455475)

[**Bảng 3. 5. Bảng experience** 43](#_Toc134455476)

[**Bảng 3. 6. Bảng savejob** 43](#_Toc134455477)

[**Bảng 3. 7. Bảng application** 44](#_Toc134455478)

# **CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI**

* 1. **Lý do chọn đề tài**

Việc tìm kiếm địa điểm thực tập là một phần không thể thiếu trong quá trình học tập của sinh viên. Tuy nhiên, đôi khi việc tìm kiếm các địa điểm thực tập phù hợp có thể trở nên khó khăn và mất nhiều thời gian cho sinh viên. Để giải quyết vấn đề này, việc xây dựng một công cụ tìm kiếm địa điểm thực tập sẽ mang lại nhiều giá trị cho cả sinh viên và nhà tuyển dụng.

Đối với sinh viên, sử dụng công cụ tìm kiếm địa điểm thực tập giúp tiết kiệm thời gian tìm kiếm địa điểm thực tập. Sinh viên có thể dễ dàng tìm kiếm các địa điểm thực tập phù hợp với nhu cầu của mình một cách nhanh chóng và tiện lợi. Công cụ tìm kiếm còn giúp sinh viên có nhiều lựa chọn hơn về các địa điểm thực tập phù hợp với chuyên ngành của mình, giúp tăng cơ hội được làm việc tại những nơi có chất lượng đào tạo tốt và tiềm năng phát triển cao. Ngoài ra, sinh viên còn có cơ hội được học hỏi kinh nghiệm thực tế từ các chuyên gia và nhân viên của các doanh nghiệp và tổ chức mà mình làm việc.

Đối với nhà tuyển dụng, công cụ tìm kiếm địa điểm thực tập giúp tìm kiếm và thuê được những sinh viên có năng lực và đủ tiềm năng để làm việc tại đơn vị của họ. Nhà tuyển dụng có thể sử dụng công cụ tìm kiếm để tìm kiếm các sinh viên phù hợp với yêu cầu tuyển dụng của mình, giúp tăng khả năng tìm kiếm được ứng viên phù hợp. Bên cạnh đó, nhà tuyển dụng còn có thể giảm thời gian và chi phí tuyển dụng bằng cách sử dụng công cụ tìm kiếm để thuê được các ứng viên phù hợp.

Trên cơ sở đó, việc xây dựng một website tìm kiếm địa điểm thực tập là rất cần thiết. Công cụ tìm kiếm địa điểm thực tập cũng cung cấp thông tin về các công ty và tổ chức có nhu cầu tuyển dụng sinh viên thực tập, giúp những sinh viên chưa có kinh nghiệm tiếp cận với các cơ hội thực tập tốt hơn. Việc đưa ra thông tin về các doanh nghiệp và tổ chức cũng giúp sinh viên có thể tìm hiểu về các ngành nghề, yêu cầu tuyển dụng và khả năng phát triển trong tương lai của ngành nghề mình quan tâm.

* 1. **Mục tiêu và phạm vi nghiên cứu**

Mục tiêu của đề tài "Xây dựng website tìm kiếm địa điểm thực tập cho sinh viên" là thiết kế và triển khai một website tìm kiếm địa điểm thực tập cho sinh viên, giúp cho các sinh viên tìm kiếm và nộp đơn xin thực tập một cách dễ dàng và tiện lợi hơn. Website này sẽ giúp các sinh viên có thể tìm kiếm các địa điểm thực tập theo nhiều tiêu chí khác nhau, bao gồm ngành nghề, vị trí, khu vực, mức lương và các yêu cầu khác.

Phạm vi nghiên cứu của đề tài này sẽ tập trung vào việc thiết kế và triển khai một website tìm kiếm địa điểm thực tập cho sinh viên. Các nghiên cứu sẽ bao gồm phân tích các yêu cầu và tiêu chí của các sinh viên trong việc tìm kiếm địa điểm thực tập, phân tích các yêu cầu và tiêu chí của các doanh nghiệp và tổ chức trong việc tuyển dụng sinh viên thực tập, đánh giá các tính năng và chức năng của các website tìm kiếm địa điểm thực tập hiện có trên thị trường, và phát triển một website tìm kiếm địa điểm thực tập cho sinh viên đáp ứng được các yêu cầu và tiêu chí này. Website sẽ được thiết kế với giao diện đơn giản, thân thiện và dễ sử dụng, cung cấp cho sinh viên một trang chủ đẹp mắt và các trang tìm kiếm linh hoạt. Các doanh nghiệp và tổ chức sẽ có thể đăng tải thông tin về các địa điểm thực tập của mình, và sinh viên có thể tìm kiếm và nộp đơn xin thực tập trực tiếp trên website.

Kết quả của đề tài này sẽ là một website tìm kiếm địa điểm thực tập cho sinh viên, giúp cho các sinh viên có thể tìm kiếm và nộp đơn xin thực tập một cách dễ dàng và tiện lợi hơn, đồng thời giúp tăng khả năng tuyển dụng của các doanh nghiệp và tăng tính minh bạch trong quá trình quản lý.

# **CHƯƠNG 2. CƠ SỞ LÝ THUYẾT VỀ CÁC CÔNG NGHỆ SỬ DỤNG**

* 1. **Javascript**
     1. **JavaScript là gì?**

JavaScript là một ngôn ngữ lập trình kịch bản (scripting language) phía client-side được sử dụng để tạo các hiệu ứng động, thay đổi nội dung của trang web, kiểm tra và xử lý dữ liệu được nhập từ người dùng và thực hiện các tác vụ khác trên trình duyệt của người dùng. JavaScript là một ngôn ngữ dựa trên đối tượng (object-oriented), có cú pháp giống với các ngôn ngữ lập trình khác như C++, Java và Python.

Ngày nay, Javascript không chỉ có thể chạy trên trình duyệt mà còn có thể chạy trên máy chủ, hoặc trên bất kỳ thiết bị nào có chương trình đặc biệt được gọi là JavaScript engine.

JavaScript engine là một phần mềm chịu trách nhiệm thực thi mã JavaScript. Nó là một thành phần quan trọng của trình duyệt web, cho phép mã JavaScript được thực thi trên trang web. Một JavaScript engine thực hiện các bước sau:

1. Phân tích cú pháp (parsing): JavaScript engine phân tích cú pháp của mã JavaScript để hiểu cấu trúc của nó.
2. Biên dịch (compiling): JavaScript engine biên dịch mã JavaScript thành mã máy để có thể được thực thi.
3. Tối ưu hóa (optimizing): JavaScript engine tối ưu hóa mã máy để tăng tốc độ thực thi.
4. Thực thi (executing): JavaScript engine thực thi mã JavaScript trên trang web và kết quả sẽ được hiển thị cho người dùng.

Mỗi trình duyệt web có một JavaScript engine riêng biệt, với các đặc điểm và tính năng khác nhau. Ví dụ, trình duyệt Google Chrome sử dụng JavaScript engine V8, trong khi Firefox sử dụng SpiderMonkey và Microsoft Edge sử dụng Chakra.

* + 1. **JavaScript trong trình duyệt có thể làm gì?**

JavaScript trong trình duyệt có thể làm rất nhiều công việc khác nhau, bao gồm:

* Tương tác với HTML và CSS để thay đổi nội dung và kiểu dáng của trang web.
* Xử lý dữ liệu được nhập từ người dùng, bao gồm kiểm tra tính hợp lệ và xử lý các lỗi.
* Thực hiện các tác vụ tự động trên trang web, bao gồm tải lại nội dung, đóng và mở cửa sổ, và cuộn trang tự động.
* Tạo các hiệu ứng động, bao gồm hiệu ứng hover, hiệu ứng chuyển động và hiệu ứng tương tác.
* Thực hiện các kết nối đến các API bên ngoài, bao gồm kết nối đến cơ sở dữ liệu, lấy dữ liệu từ các trang web khác, và gửi dữ liệu đến các dịch vụ trực tuyến khác.
* Phát triển các ứng dụng web và trang web động, bao gồm các ứng dụng web đa trang và các trang web thương mại điện tử.

Tóm lại, JavaScript là một ngôn ngữ lập trình phía client-side rất mạnh mẽ và linh hoạt, cho phép phát triển các ứng dụng web và trang web động phong phú và đa dạng.

* + 1. **JavaScript trong trình duyệt không thể làm gì?**

Mặc dù JavaScript là một ngôn ngữ lập trình phía client-side rất mạnh mẽ và linh hoạt, nhưng nó cũng có một số hạn chế về khả năng thực hiện một số tác vụ. Sau đây là một số tác vụ mà JavaScript trong trình duyệt không thể thực hiện:

* Truy cập vào các tệp tin trên máy tính của người dùng: JavaScript không thể đọc hoặc ghi các tệp tin trên máy tính của người dùng mà không thông qua một yêu cầu từ người dùng.
* Thực thi các tác vụ nặng, tốn kém tài nguyên: JavaScript không thể thực hiện các tác vụ nặng, tốn kém tài nguyên như xử lý hình ảnh hoặc video với kích thước lớn.
* Truy cập vào các nguồn tài nguyên khác qua cross-origin: JavaScript trong trình duyệt không thể truy cập các nguồn tài nguyên khác nếu chúng không thuộc cùng một nguồn gốc (origin).
* Thực hiện các tác vụ bảo mật như đọc các cookie hoặc thông tin cá nhân của người dùng từ các trang web khác.
* Thực hiện các tác vụ trực tiếp trên cơ sở dữ liệu: JavaScript không thể trực tiếp thực hiện các tác vụ trên cơ sở dữ liệu mà cần thông qua một lớp trung gian như một API để kết nối với cơ sở dữ liệu.

Tóm lại, JavaScript trong trình duyệt có những giới hạn về khả năng thực hiện một số tác vụ, nhưng vẫn là một công cụ rất mạnh mẽ để phát triển các ứng dụng web và trang web động.

* + 1. **Languages “over” JavaScript**

Languages “over” JavaScript có nghĩa là các ngôn ngữ lập trình được viết trên nền tảng JavaScript, sử dụng cú pháp và cấu trúc tương tự như JavaScript nhưng được bổ sung thêm các tính năng và chức năng mới.

Một số ngôn ngữ "trên" JavaScript phổ biến hiện nay bao gồm:

* TypeScript: TypeScript là một ngôn ngữ lập trình mã nguồn mở được phát triển bởi Microsoft. Nó bổ sung tính năng kiểm tra kiểu dữ liệu và cung cấp hỗ trợ cho các tính năng mới của ECMAScript (phiên bản chuẩn của JavaScript) để giúp tăng hiệu suất và độ tin cậy của mã JavaScript.
* CoffeeScript: CoffeeScript là một ngôn ngữ lập trình mã nguồn mở, đơn giản hóa cú pháp và cung cấp các tính năng mới như hàm bất đồng bộ (asynchronous functions) và bảo mật (sandboxing). Mã CoffeeScript được biên dịch thành mã JavaScript để chạy trên các trình duyệt web.
* Dart: Dart là một ngôn ngữ lập trình được phát triển bởi Google, được thiết kế để thay thế JavaScript. Nó cung cấp các tính năng mới như kiểm tra kiểu tĩnh (static type checking) và hỗ trợ cho các tính năng mới của ECMAScript, nhưng đòi hỏi trình biên dịch riêng để chuyển đổi mã Dart thành mã JavaScript để chạy trên các trình duyệt web.

Tất cả các ngôn ngữ "trên" JavaScript này đều giúp cho các nhà phát triển tăng cường hiệu suất và độ tin cậy của mã JavaScript, tuy nhiên, chúng đều cần một quá trình biên dịch hoặc chuyển đổi thành mã JavaScript để chạy trên các trình duyệt web.

* 1. **Typescript**
     1. **Typescript là gì?**

TypeScript là một ngôn ngữ lập trình mã nguồn mở, được phát triển bởi Microsoft. TypeScript được thiết kế để cung cấp các tính năng mới và kiểm tra kiểu dữ liệu cho JavaScript, để tăng khả năng phát triển và bảo trì ứng dụng. TypeScript có cú pháp tương tự như JavaScript, nhưng cung cấp thêm các tính năng mới như kiểm tra kiểu dữ liệu tĩnh, tính năng mới của ECMAScript, tính mở rộng và tương thích ngược với JavaScript.

Mã TypeScript có thể được biên dịch sang mã JavaScript, để chạy trên trình duyệt web hoặc bất kỳ nền tảng nào hỗ trợ JavaScript. TypeScript giúp cho việc phát triển ứng dụng web trở nên dễ dàng hơn, đồng thời giúp tăng tính ổn định và độ tin cậy của mã. TypeScript được sử dụng rộng rãi trong các dự án phát triển ứng dụng web, đặc biệt là trong các dự án lớn và phức tạp.

* + 1. **Đặc điểm của Typescript**

TypeScript có các đặc điểm chính sau:

* Kiểm tra kiểu dữ liệu: TypeScript cung cấp kiểm tra kiểu dữ liệu tĩnh, cho phép kiểm tra các lỗi tiềm ẩn trong mã trước khi chạy nó. Kiểm tra kiểu dữ liệu cũng giúp cung cấp các gợi ý thông minh và phát hiện lỗi sai kiểu dữ liệu trong quá trình phát triển ứng dụng.
* Tính năng mới của ECMAScript: TypeScript hỗ trợ các tính năng mới của ECMAScript, phiên bản chuẩn của JavaScript, bao gồm async/await, iterators, generators, và module. Điều này giúp cho mã TypeScript có thể được biên dịch và chạy trên các trình duyệt web, giống như mã JavaScript.
* Tính mở rộng: TypeScript cho phép mở rộng các đối tượng và kiểu dữ liệu trong mã, giúp tăng cường khả năng tái sử dụng mã và mở rộng chức năng của mã.
* Tương thích ngược với JavaScript: TypeScript được thiết kế để tương thích ngược với JavaScript, điều này có nghĩa là mã JavaScript có thể được sử dụng trong mã TypeScript và ngược lại. Điều này giúp cho các nhà phát triển có thể chuyển đổi dần dần các ứng dụng JavaScript sang TypeScript một cách dễ dàng và tránh các lỗi không mong muốn.
  + 1. **So sánh giữa Typescript và Javascript**

|  |  |  |
| --- | --- | --- |
| Tính năng | Typescript | Javascript |
| Ra mắt lần đầu | 2012 | 1995 |
| Kiểu dữ liệu | Nhập tĩnh | Nhập động |
| Cú pháp | Tương tự Javascript, với các tính năng bổ sung | Cú pháp Javascript chuẩn |
| Khả năng tương thích | Tương thích ngược với JavaScript | Không thể chạy TypeScript trong tệp JavaScript |
| Cộng đồng | Nhỏ hơn Javascript | Lớn hơn Typescript |
| Phần mở rộng | .js | .ts |

Về các tính năng, đây là 10 điểm khác biệt đáng kể giữa JavaScript và TypeScript:

1. TypeScript có thể được gõ mạnh, trong khi JavaScript chỉ được gõ động.
2. TypeScript dễ đọc và dễ bảo trì hơn JavaScript.
3. TypeScript hỗ trợ trừu tượng hóa thông qua các giao diện , trong khi JavaScript thì không.
4. TypeScript cho phép các nhà phát triển chú thích mã bằng các bộ trang trí, trong khi JavaScript thì không.
5. TypeScript hỗ trợ khả năng mô đun hóa và sắp xếp các thành phần thông qua việc sử dụng các không gian tên, vốn không được hỗ trợ trong JavaScript.
6. TypeScript biểu cảm hơn JavaScript, thông qua việc sử dụng các yếu tố cú pháp như tham số tùy chọn và tham số được đặt tên.
7. TypeScript hỗ trợ tính năng chung và tính năng suy luận kiểu không có sẵn trong JavaScript.
8. IDE TypeScript có nhiều tính năng hơn, vì việc xây dựng các plugin và công cụ cho ngôn ngữ được nhập tĩnh sẽ dễ dàng hơn.
9. Mã TypeScript dễ gỡ lỗi hơn khi cơ sở mã mở rộng, vì lỗi loại có thể được phát hiện tại thời điểm biên dịch thay vì thời gian chạy.
10. TypeScript triển khai các tính năng bổ sung ngoài đặc tả ECMAScript hạn chế mà JavaScript tuân thủ.
    * 1. **Vì sao chọn Typescript?**

TypeScript làm cho JavaScript tốt hơn. TypeScript không phải là đối thủ cạnh tranh với JavaScript. Thay vào đó, TypeScript bổ sung cho JavaScript.

TypeScript cung cấp cho cộng đồng một cách năng động hơn, đầy đủ tính năng hơn và an toàn hơn để phát triển các ứng dụng cấp doanh nghiệp trong đó thời gian chạy mục tiêu yêu cầu JavaScript.

TypeScript không được thiết kế để thay thế JavaScript. Thay vào đó, mục đích của nó là khuyến khích sự phổ biến của các nền tảng dựa trên JavaScript bằng cách giúp viết, tích hợp, quản lý và duy trì mã dễ dàng hơn.

Các nền tảng dựa trên JavaScript như NodeJS trên máy chủ và ReactJS trên máy khách tiếp tục trở nên phổ biến. Khả năng viết mã trong TypeScript và biến nó thành JavaScript là một trong những lý do khiến tỷ lệ chấp nhận cả hai ngôn ngữ tiếp tục tăng.

* 1. **NestJS**
     1. **NestJS là gì**

NestJS là một framework Node.js được xây dựng trên TypeScript, sử dụng các khái niệm và cú pháp của Angular để xây dựng các ứng dụng back-end. Nó cung cấp một cách tiếp cận tốt hơn để xây dựng các ứng dụng back-end phức tạp, cung cấp một kiến trúc module rõ ràng, hỗ trợ dependency injection, routing và middleware. NestJS cũng hỗ trợ nhiều công nghệ như WebSockets, gRPC, GraphQL và microservices, giúp tăng tính linh hoạt và khả năng mở rộng của ứng dụng. Nó cũng cung cấp các tính năng bảo mật mạnh mẽ như xác thực, phân quyền và mã hóa.

* + 1. **Một số đặc điểm của NestJS**

Một số đặc điểm của NestJS bao gồm:

1. Dependency Injection (DI): NestJS hỗ trợ DI để giảm thiểu sự phụ thuộc giữa các phần của ứng dụng. DI giúp cho việc kiểm thử và bảo trì dễ dàng hơn.
2. Modular Architecture: NestJS cung cấp một kiến trúc module rõ ràng giúp chia ứng dụng thành các phần nhỏ hơn, dễ quản lý hơn.
3. Middleware: Middleware giúp xử lý các yêu cầu và phản hồi giữa server và client.
4. Exception Handling: NestJS cung cấp một cách tiếp cận đơn giản cho việc xử lý các ngoại lệ (exception) trong ứng dụng.
5. Validation: NestJS cung cấp một cách tiếp cận dễ dàng để xác thực và kiểm tra dữ liệu đầu vào của người dùng.
6. WebSockets: NestJS hỗ trợ WebSockets để xử lý các ứng dụng thời gian thực.
7. GraphQL: NestJS hỗ trợ GraphQL, một cách tiếp cận mới để xây dựng các API linh hoạt.
8. Microservices: NestJS hỗ trợ việc phát triển các ứng dụng phân tán và có khả năng mở rộng cao thông qua việc sử dụng microservices.

Ngoài ra, NestJS cũng cung cấp nhiều tính năng bảo mật, như xác thực, phân quyền và mã hóa, giúp đảm bảo an toàn và bảo mật cho ứng dụng.

* + 1. **Ba thành phần chính của NestJs**

NestJS có 3 thành phần chính là: controllers, providers, and modules.

Controllers: Khi nhận được yêu cầu HTTP, cơ chế định tuyến sẽ hướng yêu cầu đến bộ điều khiển thích hợp. Controller sau đó xử lý yêu cầu và trả về phản hồi thích hợp.

Providers: Nhà cung cấp là nền tảng của Nest — Providers có thể được đưa vào như một thành phần phụ thuộc để tạo mối quan hệ giữa các đối tượng khác nhau. Các lớp như dịch vụ, kho lưu trữ hoặc người trợ giúp có thể được coi là nhà cung cấp chỉ bằng cách thêm trình trang trí `@Injectable()` của Nest. Các Providers có thể được đưa vào một lớp thông qua hàm tạo và Nest sẽ xử lý việc giải quyết các phần phụ thuộc, giúp việc quản lý phần phụ thuộc trở nên cực kỳ dễ dàng.

Modules: nhằm đóng gói một tập hợp các khả năng có liên quan chặt chẽ. Mỗi ứng dụng có một mô-đun gốc mà Nest sử dụng làm điểm bắt đầu để giải quyết cấu trúc và các mối quan hệ của ứng dụng. Nên sử dụng nhiều Modules để tổ chức các thành phần ứng dụng. Một modules được xác định bằng cách sử dụng decorator `@Module()` của Nest, mô tả các thuộc tính như bộ điều khiển, nhà cung cấp và phần phụ thuộc của module.

* + 1. **Vì sao lựa chọn NestJS cho ứng dụng Next.js**

NestJS và Next.js là hai framework khác nhau, NestJS dành cho back-end và Next.js dành cho front-end. Tuy nhiên, khi phát triển một ứng dụng web hoàn chỉnh, thường cần có cả phần back-end và front-end. Việc chọn NestJS cho back-end và Next.js cho front-end sẽ có nhiều lợi ích, bao gồm:

* Đồng bộ: NestJS và Next.js đều được xây dựng trên TypeScript, cho phép việc phát triển đồng bộ giữa back-end và front-end.
* Độ linh hoạt: Sử dụng hai framework khác nhau cho back-end và front-end cho phép bạn lựa chọn các công nghệ phù hợp nhất cho mỗi phần của ứng dụng. NestJS hỗ trợ nhiều database và cho phép bạn lựa chọn các công nghệ khác nhau để xử lý các tác vụ khác nhau trong back-end, trong khi Next.js cung cấp các tính năng như Server-Side Rendering (SSR) và Static Site Generation (SSG) cho phần front-end.
* Phân tách logic: Sử dụng NestJS và Next.js cho phép bạn phân tách logic giữa back-end và front-end, làm cho mã của bạn dễ đọc hơn và dễ bảo trì hơn.
* Hỗ trợ đầy đủ: Cả NestJS và Next.js đều là các framework phổ biến, được cộng đồng hỗ trợ tích cực. Điều này đảm bảo rằng bạn có thể tìm thấy nhiều tài liệu và hỗ trợ khi phát triển ứng dụng của mình.

Tóm lại, việc chọn NestJS cho back-end và Next.js cho front-end là một lựa chọn tốt cho ứng dụng web của bạn, giúp tăng tính linh hoạt và giảm thiểu sự phụ thuộc giữa các phần của ứng dụng.

* 1. **Next.js**
     1. **Next.js là gì**

Next.js là một framework JavaScript mã nguồn mở cho phép xây dựng các ứng dụng web hiệu quả, linh hoạt và dễ dàng phát triển. Next.js được xây dựng trên nền tảng React và cho phép bạn xây dựng các ứng dụng web đơn trang (single-page applications) hoặc trang web đa trang (multi-page applications).

Next.js cung cấp các tính năng quan trọng như Server-Side Rendering (SSR), Static Site Generation (SSG), code splitting và caching, giúp tối ưu hóa tốc độ tải trang và hiệu suất của ứng dụng. Next.js cũng hỗ trợ các tính năng như Hot Module Replacement (HMR) và các công cụ phát triển để giúp việc phát triển và sửa lỗi trở nên dễ dàng hơn.

Next.js là một framework đáng tin cậy và được sử dụng rộng rãi trong cộng đồng phát triển web hiện nay. Bạn có thể sử dụng Next.js để xây dựng các ứng dụng web đa dạng, từ các trang web cá nhân đến các ứng dụng di động, thương mại điện tử và nhiều hơn nữa.

* + 1. **Tính năng của Next.js**

Next.js là một framework JavaScript mã nguồn mở, được xây dựng trên nền tảng React và Node.js, cho phép phát triển các ứng dụng web hiệu quả, linh hoạt và dễ dàng. Next.js hỗ trợ nhiều tính năng quan trọng như:

* Server-Side Rendering (SSR): Next.js cho phép bạn render các trang trên server trước khi trình duyệt của người dùng tải nó xuống. Điều này giúp cải thiện tốc độ tải trang và hiệu suất của ứng dụng web.
* Static Site Generation (SSG): Next.js cũng hỗ trợ tính năng SSG, cho phép bạn render các trang tĩnh trước đó và lưu trữ chúng trong bộ nhớ đệm. Điều này giúp cải thiện tốc độ tải trang và giảm chi phí lưu trữ.
* Automatic Code Splitting: Next.js tự động phân chia mã của bạn thành các bundle nhỏ để tối ưu hóa tốc độ tải trang.
* Dynamic Imports: Next.js hỗ trợ dynamic imports, cho phép bạn load các module cần thiết khi cần chúng. Điều này giúp giảm thời gian tải trang và cải thiện hiệu suất của ứng dụng.
* API Routes: Next.js cho phép bạn định nghĩa các API routes trên server để xử lý các yêu cầu từ phía client.
* Hot Module Replacement (HMR): Next.js cung cấp HMR, cho phép bạn chỉnh sửa mã nguồn của mình và xem kết quả ngay lập tức mà không cần tải lại trang.

Next.js là một framework đáng tin cậy và được sử dụng rộng rãi trong cộng đồng phát triển web. Nó cung cấp nhiều tính năng quan trọng và đơn giản hóa việc phát triển các ứng dụng web, giúp bạn tập trung vào việc xây dựng các tính năng quan trọng hơn là việc phải lo lắng về việc triển khai và quản lý các công nghệ.

* + 1. **Next.js và SEO**

Next.js có khả năng tối ưu hóa SEO tốt, điều này được đánh giá cao trong việc phát triển các ứng dụng web. Bạn có thể đạt được điều này nhờ vào Server-Side Rendering (SSR) và Static Site Generation (SSG) mà Next.js hỗ trợ.

Trong mô hình SSR, trình duyệt của người dùng sẽ nhận được các trang đã được render sẵn từ server trước khi nó được tải xuống, điều này giúp các công cụ tìm kiếm như Google có thể đọc được các nội dung trên trang web. Kết quả là, các trang web Next.js sẽ được tìm kiếm và đánh giá cao hơn trong kết quả tìm kiếm của Google và các công cụ tìm kiếm khác.

Ngoài ra, Next.js cũng hỗ trợ tính năng SSG, cho phép bạn render các trang tĩnh trước đó và lưu trữ chúng trong bộ nhớ đệm. Điều này giúp cải thiện tốc độ tải trang và giảm chi phí lưu trữ, đồng thời giúp các công cụ tìm kiếm dễ dàng đánh giá nội dung trên trang web. Next.js cũng cung cấp một số tính năng hỗ trợ SEO khác như tự động tạo các thẻ meta, cho phép bạn cấu hình các thuộc tính của các trang và các URL riêng lẻ, giúp bạn tối ưu hóa các trang web của mình để có thể xếp hạng cao hơn trong kết quả tìm kiếm của Google và các công cụ tìm kiếm khác.

* + 1. **Vì sao sử dụng Next.js**

Một số lý do vì sao nên sử dụng Next.js trong phát triển các ứng dụng web:

* Hỗ trợ Server-Side Rendering (SSR) và Static Site Generation (SSG): Next.js hỗ trợ cả SSR và SSG, cho phép bạn tối ưu hóa tốc độ tải trang và cải thiện trải nghiệm người dùng, đồng thời tối ưu hóa SEO cho trang web của bạn.
* Hỗ trợ TypeScript: Next.js cho phép bạn sử dụng TypeScript để phát triển các ứng dụng web, điều này giúp cho mã nguồn của bạn trở nên dễ bảo trì và dễ phát triển hơn.
* Cộng đồng phát triển đông đảo: Next.js được phát triển bởi một cộng đồng rộng lớn, điều này đảm bảo rằng bạn có thể tìm thấy nhiều tài liệu và hỗ trợ từ cộng đồng khi gặp phải vấn đề trong quá trình phát triển.
* Hỗ trợ tích hợp với các thư viện và framework phổ biến: Next.js hỗ trợ tích hợp với các thư viện và framework phổ biến như React, Redux, GraphQL, và nhiều hơn nữa, giúp cho bạn dễ dàng kết hợp các công nghệ khác vào trong ứng dụng của mình.
* Tính năng Hot Module Replacement (HMR): Next.js hỗ trợ HMR, cho phép bạn xem ngay kết quả của các thay đổi mà bạn vừa thực hiện trên trình duyệt mà không cần tải lại trang web.

Vì những lý do trên, Next.js là một lựa chọn tốt cho các nhà phát triển web muốn phát triển các ứng dụng web chuyên nghiệp và tối ưu hóa trải nghiệm người dùng, tối ưu hóa SEO, và dễ bảo trì.

* 1. **PostgreSQL**
     1. **PostgreSQL là gì**

PostgreSQL là một hệ thống quản lý cơ sở dữ liệu (Database Management System - DBMS) mã nguồn mở, được phát triển bởi PostgreSQL Global Development Group. PostgreSQL cung cấp khả năng quản lý cơ sở dữ liệu quan hệ (Relational Database Management System - RDBMS) với các tính năng như ACID (Atomicity, Consistency, Isolation, Durability) để đảm bảo tính nhất quán và độ tin cậy của dữ liệu. PostgreSQL hỗ trợ các tính năng chuyên sâu như lập trình truy vấn, xác thực và mã hóa dữ liệu, khả năng mở rộng với đa luồng, phân tán và hỗ trợ các loại dữ liệu đa dạng như JSON, XML và hình ảnh.

PostgreSQL là một trong những DBMS phổ biến nhất hiện nay và được sử dụng trong nhiều ứng dụng phần mềm và website lớn trên toàn thế giới. Nó cũng được tích hợp sẵn trong các hệ thống như Amazon Web Services (AWS) và Google Cloud Platform (GCP) để cung cấp các dịch vụ quản lý cơ sở dữ liệu.

* + 1. **Tính năng nổi bật của PostgreSQL**

PostgreSQL là một hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) có nhiều tính năng nổi bật, đáng chú ý như sau:

* ACID Compliance: PostgreSQL là một hệ thống quản lý cơ sở dữ liệu ACID-compliant (Atomicity, Consistency, Isolation, Durability), đảm bảo tính toàn vẹn, nhất quán và độ tin cậy của dữ liệu.
* Đa nền tảng: PostgreSQL có thể chạy trên nhiều nền tảng khác nhau, bao gồm các hệ điều hành như Linux, macOS và Windows.
* Hỗ trợ các kiểu dữ liệu phong phú: PostgreSQL hỗ trợ một loạt các kiểu dữ liệu phong phú như số nguyên, số thực, ký tự, ngày giờ, mảng, JSON, XML, hình ảnh, âm thanh và video.
* Truy vấn phức tạp: PostgreSQL hỗ trợ các truy vấn phức tạp và mạnh mẽ, bao gồm các toán tử, hàm, subquery và các câu lệnh SQL phức tạp khác.
* Các tính năng bảo mật: PostgreSQL cung cấp nhiều tính năng bảo mật như xác thực, mã hóa dữ liệu và giám sát bảo mật.
* Độ mở rộng: PostgreSQL có tính năng mở rộng tốt, hỗ trợ các tính năng phân tán, đa luồng và kết nối đa người dùng.
* Cập nhật thường xuyên: PostgreSQL thường xuyên cập nhật và phát triển, với các phiên bản mới được phát hành định kỳ để cải thiện hiệu suất và tính năng.
* Cộng đồng hỗ trợ: PostgreSQL có một cộng đồng lớn và sôi động, với nhiều người dùng, nhà phát triển và chuyên gia cung cấp hỗ trợ và giải đáp các câu hỏi.
  + 1. **Vì sao sử dụng PostgreSQL**

**PL/PGSQL** là ngôn ngữ thủ tục bản địa do PostgreSQL cung cấp, có các tính năng hiện đại khác nhau. Nó hỗ trợ kiểu dữ liệu JSON nhẹ và đảm bảo tính linh hoạt được bao gồm trong một gói duy nhất. Do đó, PostgreSQL hỗ trợ một số ngôn ngữ lập trình và giao thức bao gồm Perl, Ruby, Python , .Net, C/C++, Java , ODBC và Go.

**Mã nguồn mở:** Nó miễn phí và mã nguồn mở — đây là lợi ích đáng kể nhất của PostgreSQL. Nó đã được hỗ trợ bởi hơn 20 năm phát triển cộng đồng, do đó đã góp phần vào mức độ toàn vẹn cao của nó. Mã nguồn của nó có sẵn theo giấy phép nguồn mở cho phép bạn sử dụng, sửa đổi và triển khai nó theo bất kỳ cách nào bạn thấy phù hợp – mà không phải trả thêm phí.

**Cơ sở dữ liệu quan hệ đối tượng**: Các đối tượng, lớp và nạp chồng hàm được hỗ trợ trực tiếp trong PostgreSQL. Có thể mở rộng các kiểu dữ liệu để tạo các kiểu dữ liệu tùy chỉnh, do đặc điểm hướng đối tượng của chúng. Điều này đảm bảo tính linh hoạt cao cho các nhà phát triển hoạt động với các mô hình dữ liệu phức tạp yêu cầu tích hợp cơ sở dữ liệu. Kế thừa bảng là một tính năng khác được PostgreSQL hỗ trợ do các đặc điểm hướng đối tượng của nó. Bảng con có thể kế thừa các cột từ bảng cha của nó, ngoài ra còn có các cột khác mà bảng con sở hữu, làm cho nó khác với chính nó.

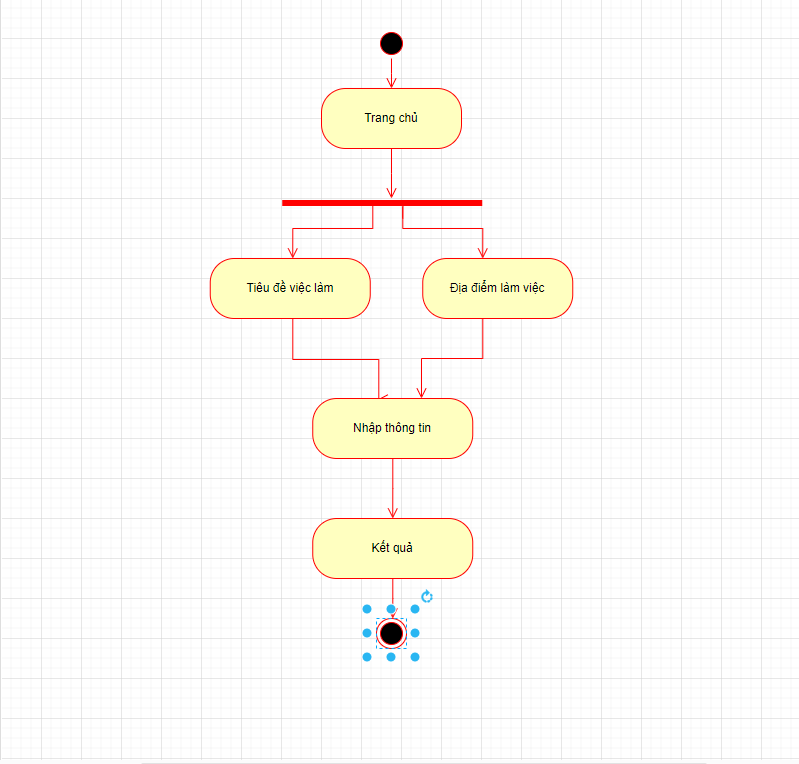
**Hiệu suất:** Thao tác ghi trong PostgreSQL có thể được thực hiện đồng thời mà không cần khóa đọc/ghi. Các chỉ mục được sử dụng để tăng tốc độ truy vấn khi xử lý một lượng lớn dữ liệu, cho phép cơ sở dữ liệu tìm thấy một hàng cụ thể mà không cần phải duyệt qua tất cả dữ liệu.

**Khả năng mở rộng:** PostgreSQL có khả năng mở rộng cao vì hoạt động của nó được định hướng theo danh mục, tức là thông tin được lưu trữ trong cơ sở dữ liệu, cột, bảng, v.v. PostgreSQL không chỉ chứa lượng thông tin tăng lên trong danh mục của nó mà còn chi tiết về kiểu dữ liệu, phương thức truy cập, chức năng, và như thế. Bạn thậm chí có thể viết mã của mình từ các ngôn ngữ lập trình khác nhau mà không cần biên dịch lại Cơ sở dữ liệu và xác định các loại dữ liệu của mình.

**Khả năng cân bằng tải**: Nó đảm bảo tính sẵn sàng cao và cân bằng tải thông qua hoạt động của máy chủ dự phòng, lập kế hoạch liên tục, chuẩn bị máy chủ chính cho máy chủ dự phòng, thiết lập máy chủ dự phòng, sao chép luồng, vị trí sao chép, sao chép theo tầng và lưu trữ liên tục ở chế độ chờ. Ngoài ra, PostgreSQL hỗ trợ sao chép đồng bộ , trong đó hai phiên bản cơ sở dữ liệu có thể chạy cùng lúc và cơ sở dữ liệu chủ được đồng bộ hóa với cơ sở dữ liệu phụ, tiếp tục đảm bảo tính sẵn sàng cao.

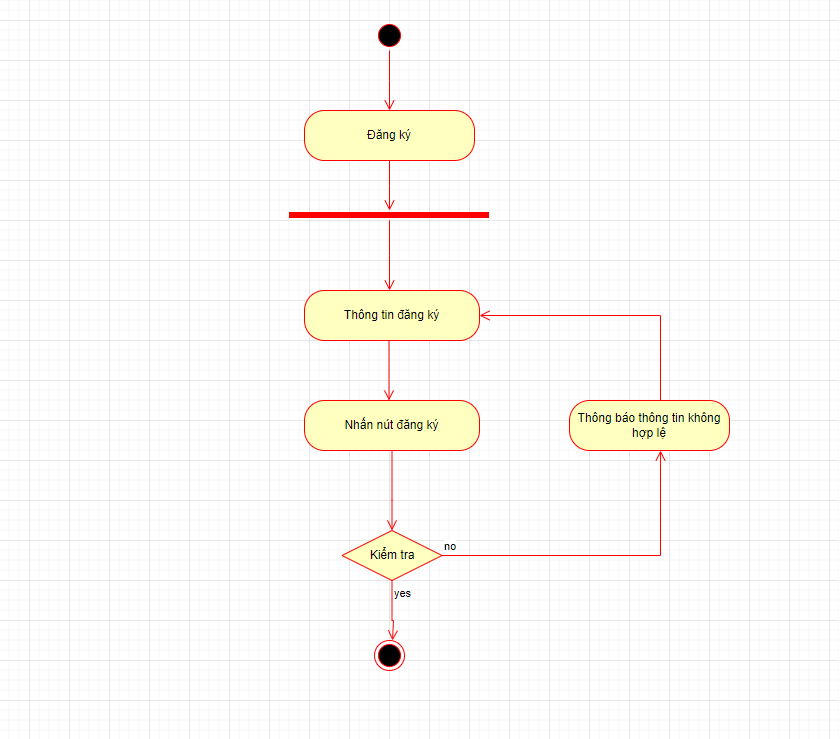
**Độ tin cậy:** Ngoài việc lưu trữ dữ liệu một cách an toàn và cho phép người dùng truy xuất dữ liệu khi yêu cầu được xử lý, nó còn được hỗ trợ bởi một cộng đồng những người đóng góp thường xuyên tìm lỗi và cố gắng cải thiện phần mềm, giúp PostgreSQL trở nên đáng tin cậy.

# **CHƯƠNG 3. PHÂN TÍCH THIẾT KẾ VÀ XÂY DỰNG CÔNG CỤ TÌM KIẾM ĐỊA ĐIỂM CHO SINH VIÊN**

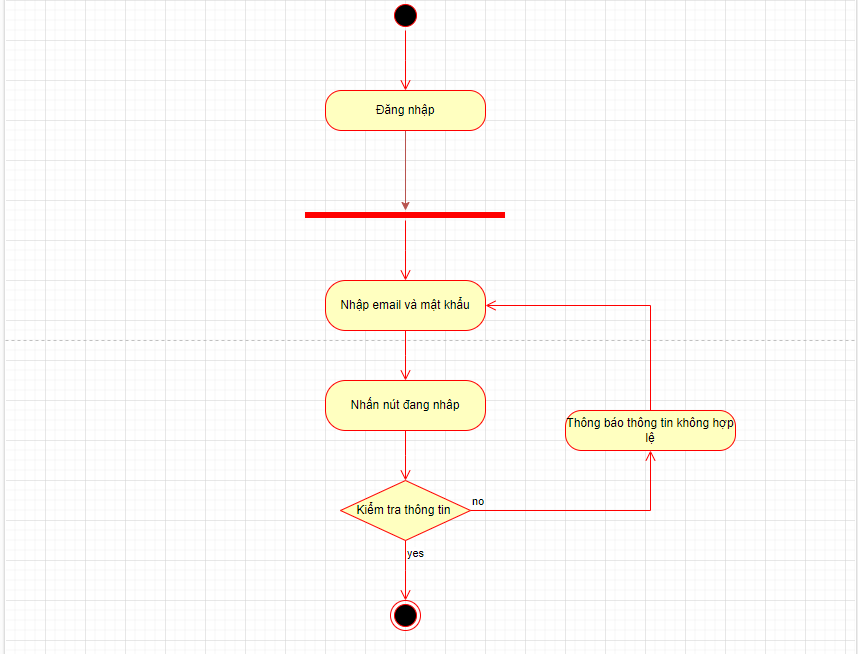
* 1. **Phân tích thiết kế hệ thống**
     1. **Biểu đồ hoạt động**
        1. **Biểu đồ hoạt động chức năng tìm kiếm việc làm**

**Hình 3.1. Biểu đồ hoạt động chức năng tìm kiếm việc làm**

* + - 1. **Biểu đồ hoạt động chức năng đăng ký cho ứng viên**

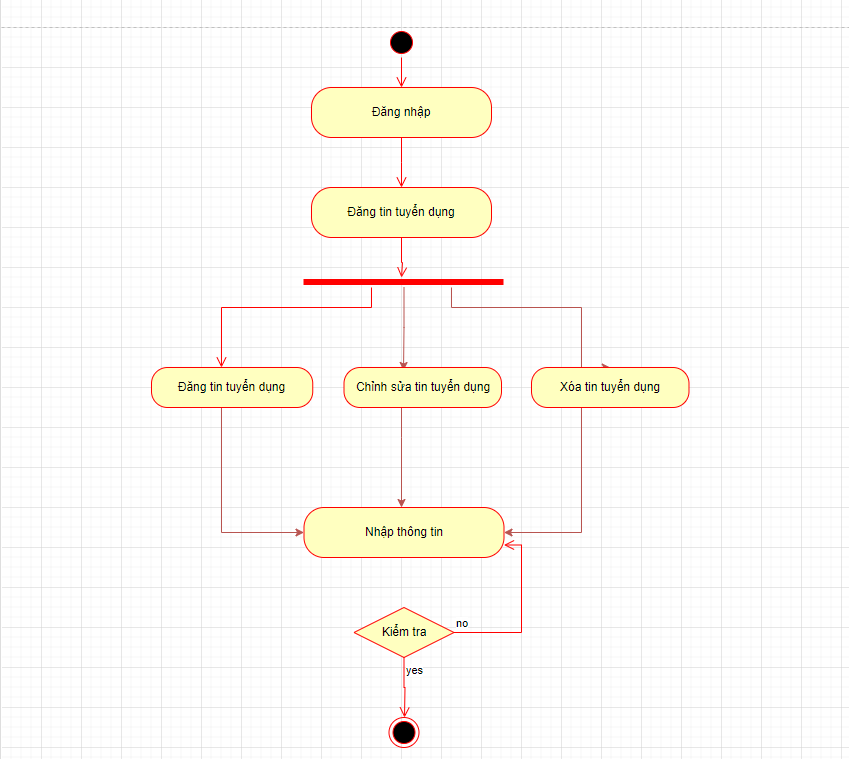
****

**Hình 3.2. Biểu đồ hoạt động chức năng đăng ký cho ứng viên**

* + - 1. **Biểu đồ hoạt động chức năng đăng nhập cho ứng viên**

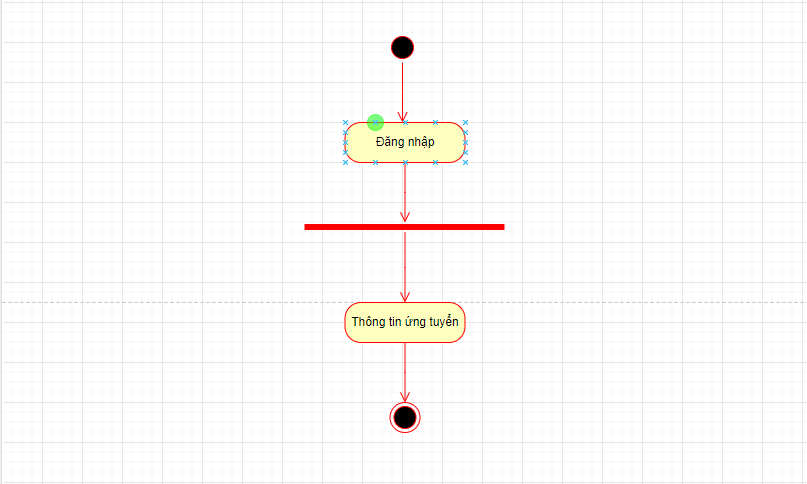
**Hình 3. 3. Biểu đồ hoạt động chức năng ứng tuyển**

* + - 1. **Biểu đồ hoạt động chức năng thêm/xóa/sửa tin tuyển dụng**

****

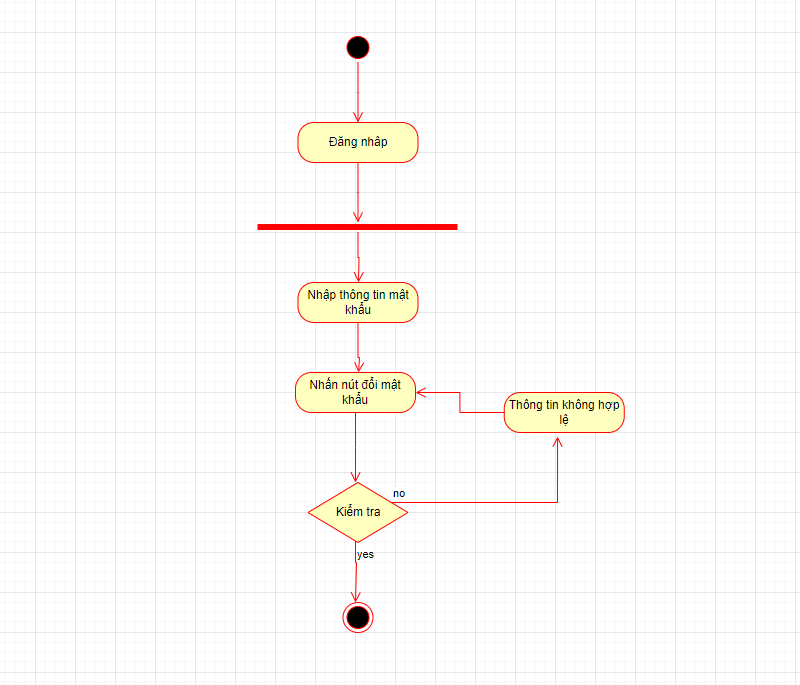
**Hình 3. 4. Biểu đồ hoạt động chức năng thêm/xóa/sửa tin tuyển dụng**

* + - 1. **Biểu đồ hoạt động chức năng ứng tuyển**



**Hình 3. 5. Biểu đồ hoạt động chức năng ứng tuyển**

* + - 1. **Biểu đồ hoạt động chức năng đổi mật khẩu**

****

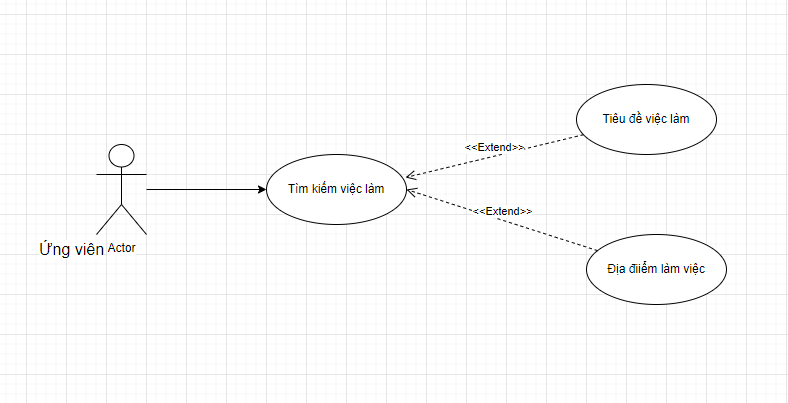
**Hình 3. 6. Biểu đồ hoạt động chức năng đổi mật khẩu**

* + 1. **Biểu đồ ca sử dụng**
       1. **Biểu đồ ca sử dụng tổng quát**

****

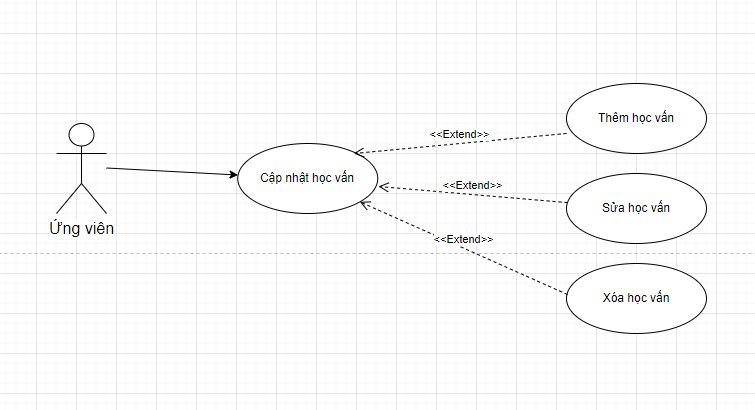
**Hình 3. 7. Biểu đồ ca sử dụng tổng quát**

* + - 1. **Biểu đồ ca sử dụng tìm kiếm tin tuyển dụng**

****

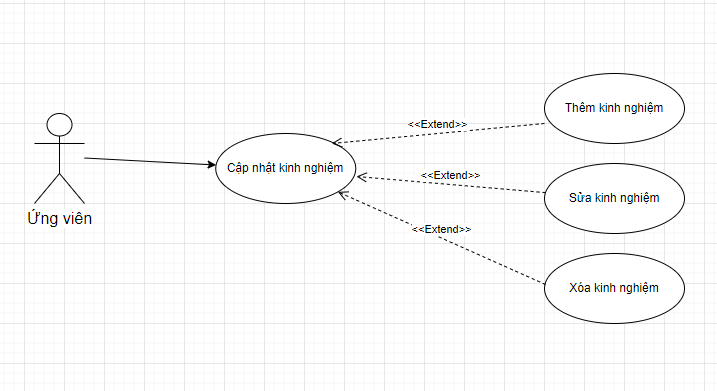
**Hình 3. 8. Biểu đồ ca sử dụng tìm kiếm tin tuyển dụng**

* + - 1. **Biểu đồ ca sử dụng cập nhật học vấn**

****

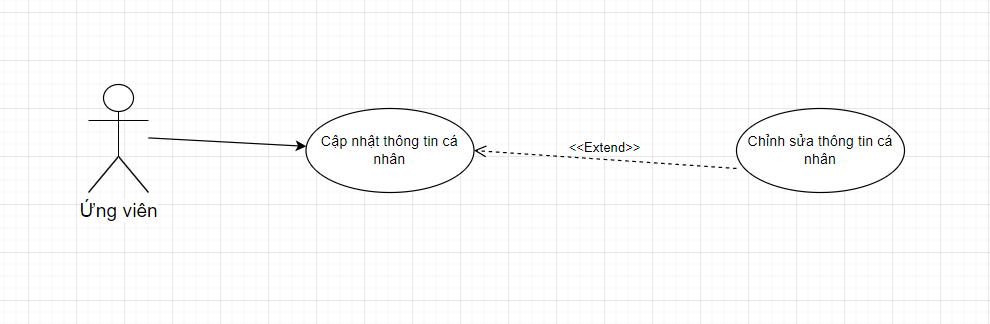
**Hình 3. 9. Biểu đồ ca sử dụng cập nhật học vấn**

* + - 1. **Biểu đồ ca sử dụng cập nhật kinh nghiệm**

****

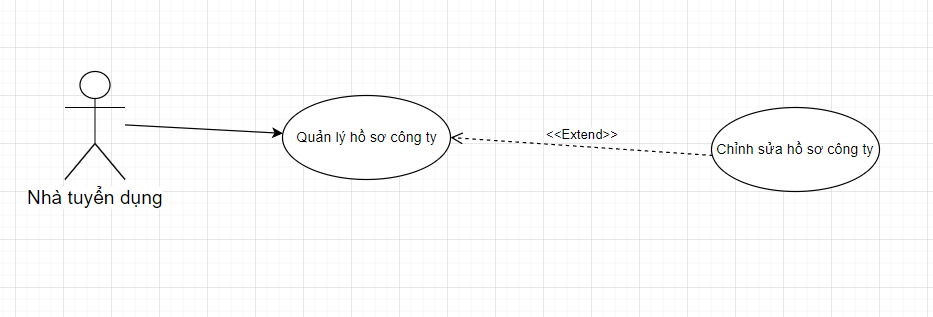
**Hình 3. 10. Biểu đồ ca sử dụng cập nhật kinh nghiệm**

* + - 1. **Biểu đồ ca sử dụng cập nhật thông tin cá nhân**

****

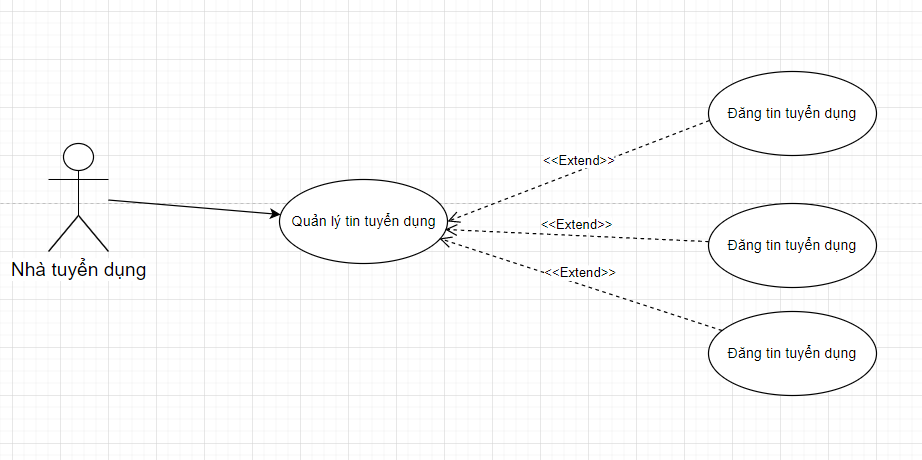
**Hình 3. 11. Biểu đồ ca sử dụng cập nhật thông tin cá nhân**

* + - 1. **Biểu đồ ca sử dụng quản lý hồ sơ công ty**

****

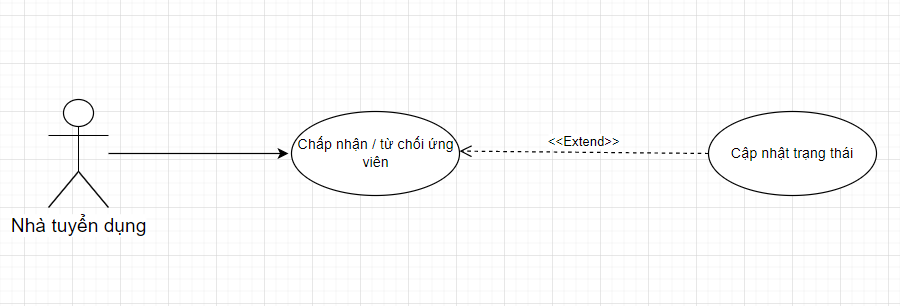
**Hình 3. 12. Biểu đồ ca sử dụng cập nhật hồ sơ công ty**

* + - 1. **Biểu đồ ca sử dụng quản lý tin tuyển dụng**

****

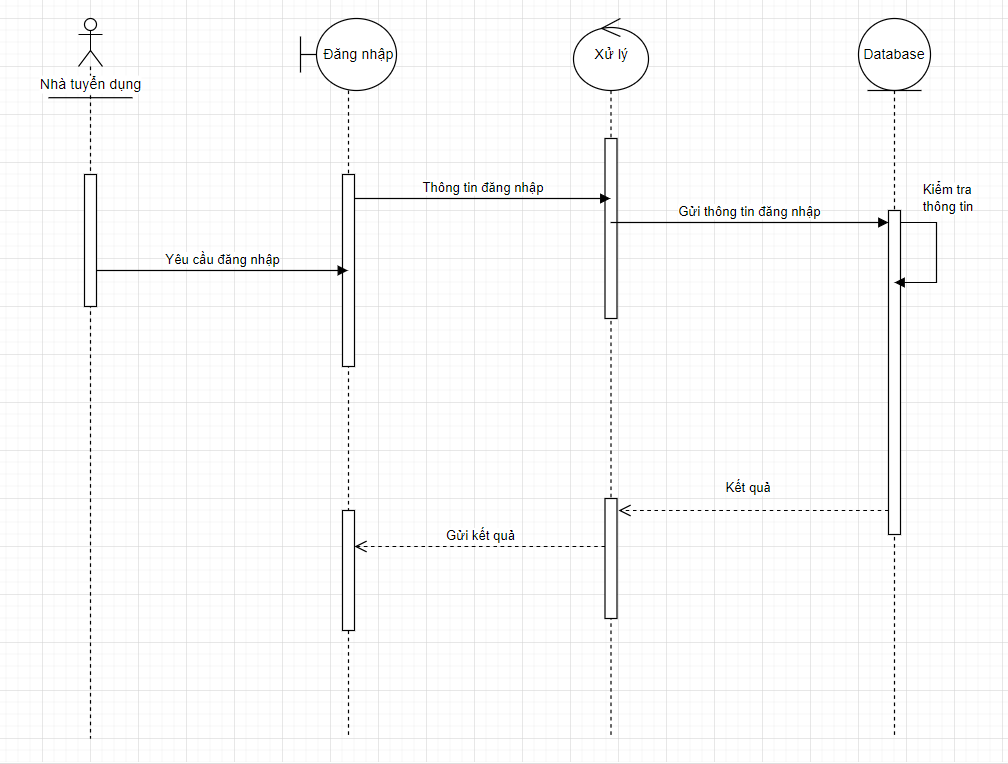
**Hình 3. 13. Biểu đồ ca sử dụng quản lý tin tuyển dụng**

* + - 1. **Biểu đồ ca sử dụng chấp nhận / từ chối ứng viên**

****

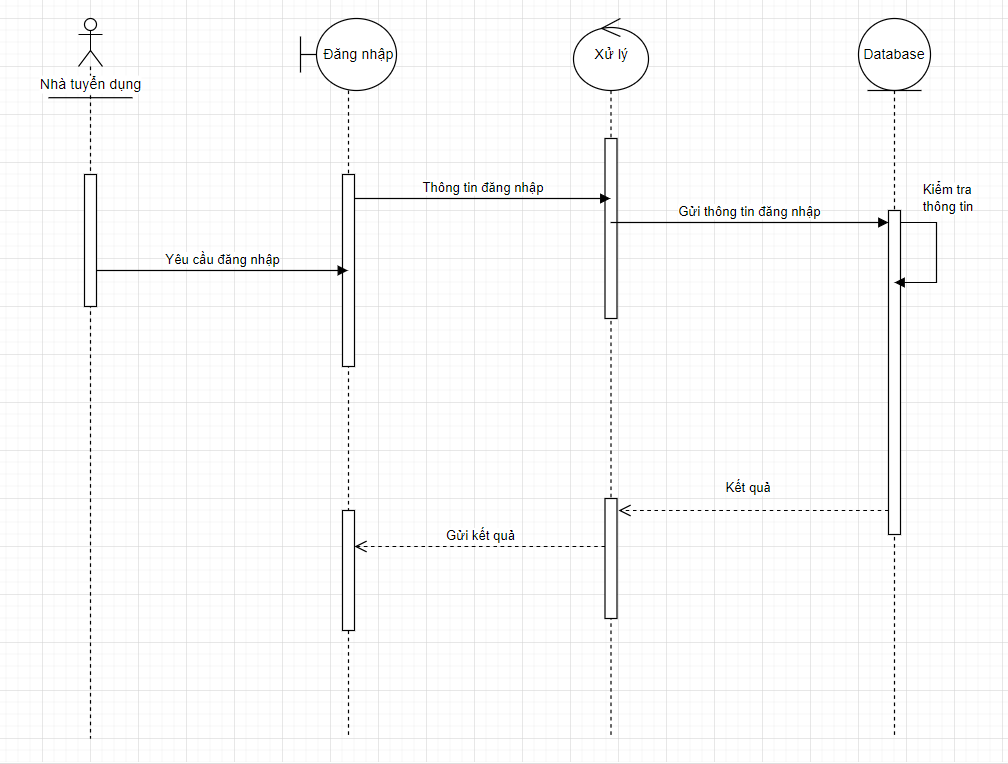
**Hình 3. 14. Biểu đồ ca sử dụng chấp nhận / từ chối ứng viên**

* + 1. **Biểu đồ tuần tự**
       1. **Biểu đồ tuần tự đăng nhập tài khoản nhà tuyển dụng**

****

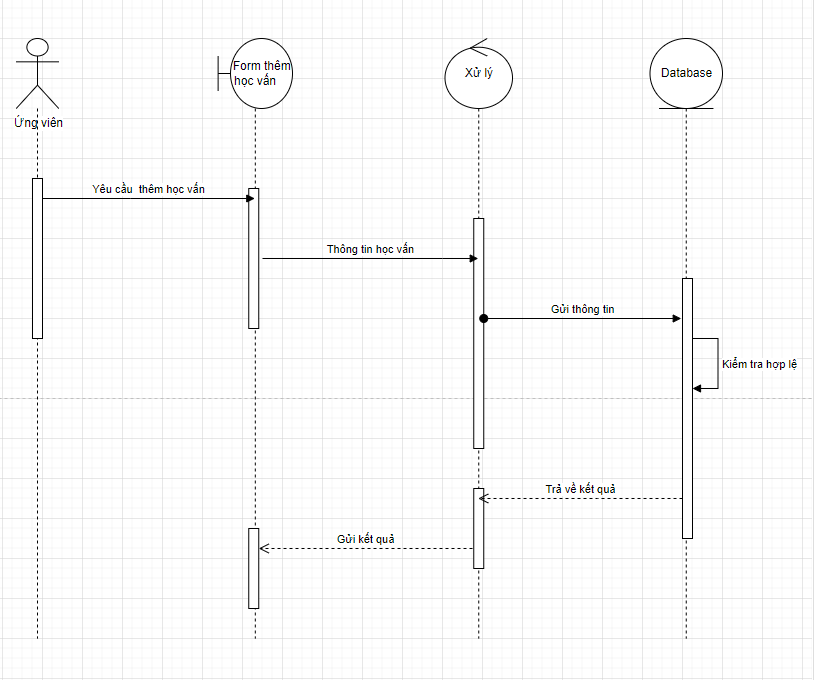
**Hình 3. 15. Biểu đồ tuần tự đăng nhập tài khoản nhà tuyển dụng**

* + - 1. **Biểu đồ tuần tự đăng ký tài khoản**

****

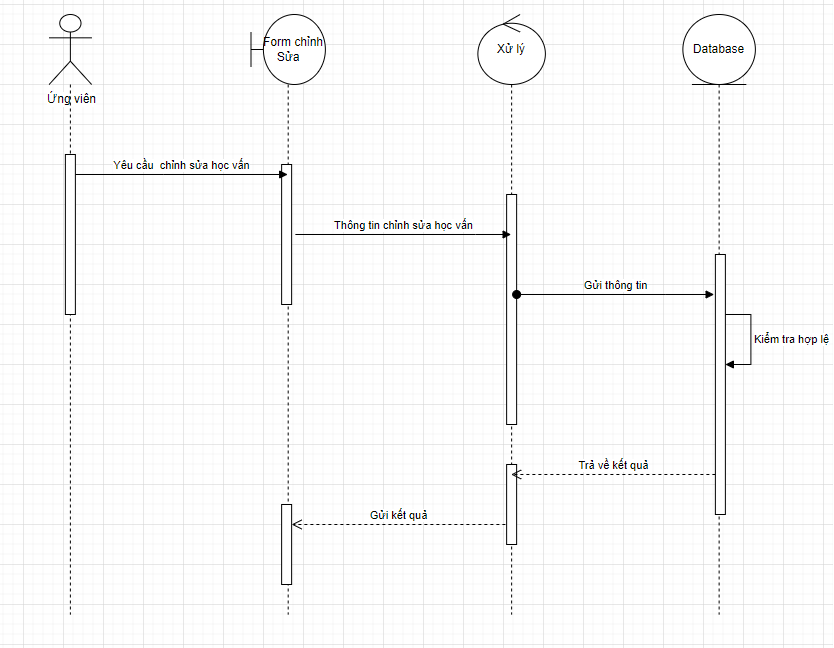
**Hình 3. 16. Biểu đồ tuần tự đăng ký tài khoản**

* + - 1. **Biểu đồ tuần tự thêm học vấn**

****

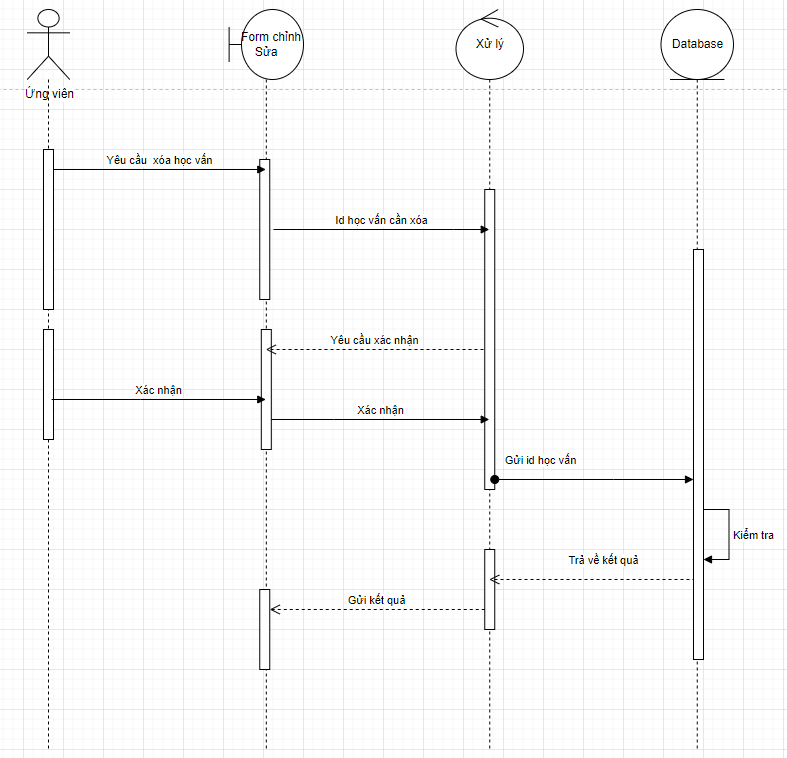
**Hình 3. 17. Biểu đồ tuần tự thêm học vấn**

* + - 1. **Biểu đồ tuần tự chỉnh sửa học vấn**

****

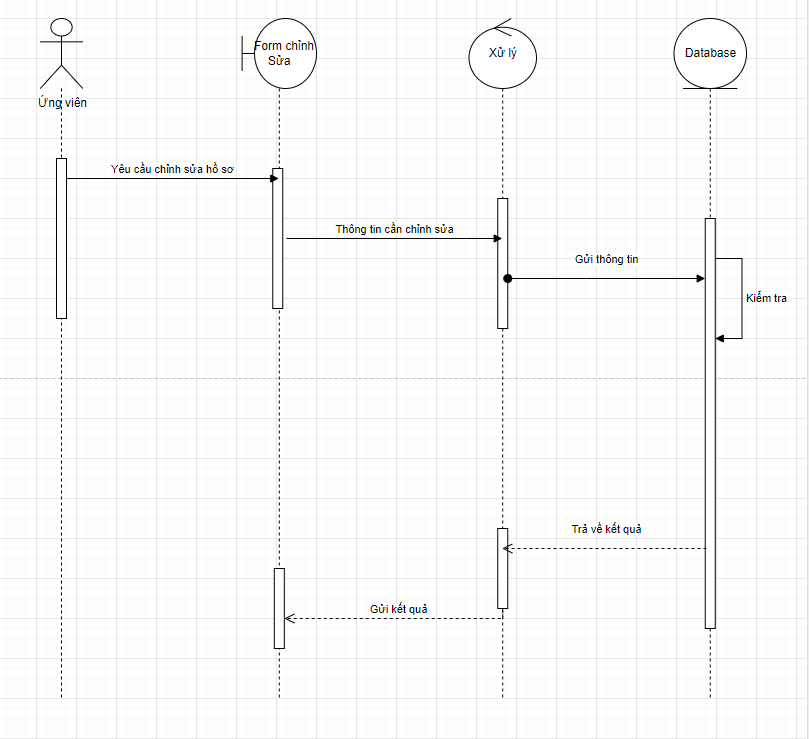
**Hình 3. 18. Biểu đồ tuần tự chỉnh sửa học vấn**

* + - 1. **Biểu đồ tuần tự xóa học vấn**

****

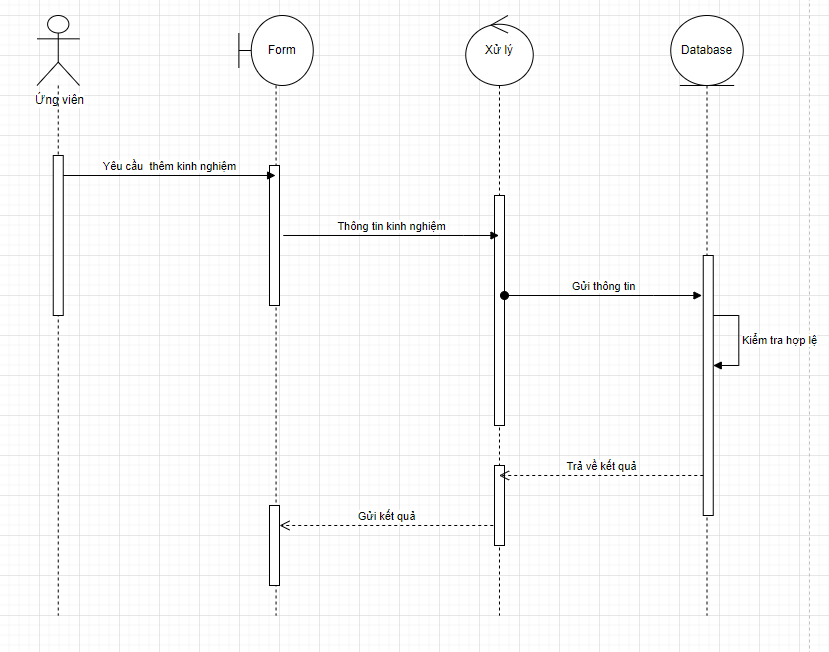
**Hình 3. 19. Biểu đồ tuần tự xóa học vấn**

* + - 1. **Biểu đồ tuần tự chức năng chỉnh sửa hồ sơ**

****

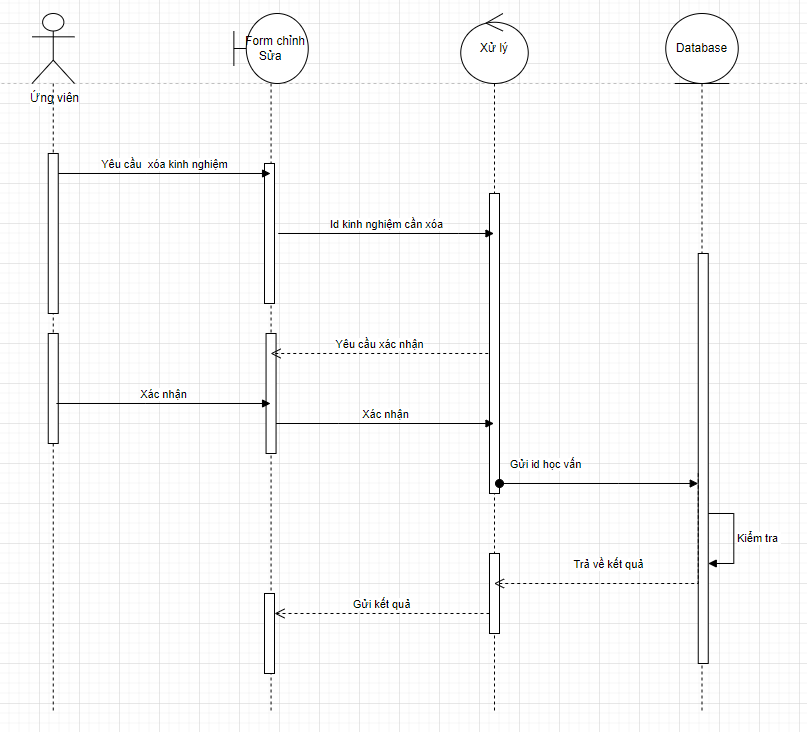
**Hình 3. 20. Biểu đồ tuần tự chức năng chỉnh sửa hồ sơ**

* + - 1. **Biểu đồ tuần tự thêm kinh nghiệm làm việc**

****

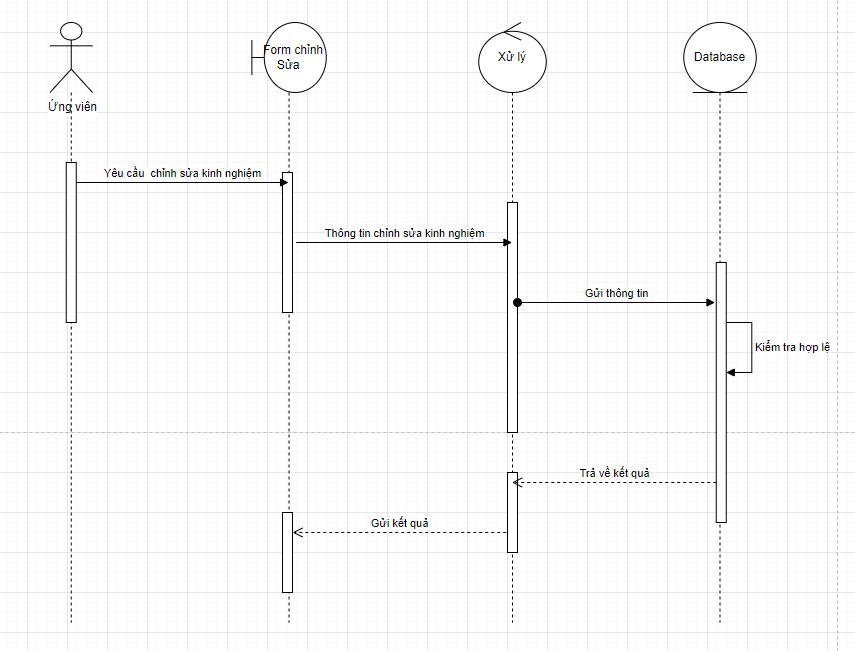
**Hình 3. 21. Biểu đồ tuần tự thêm kinh nghiệm làm việc**

* + - 1. **Biểu đồ tuần tự xóa kinh nghiệm làm việc**

****

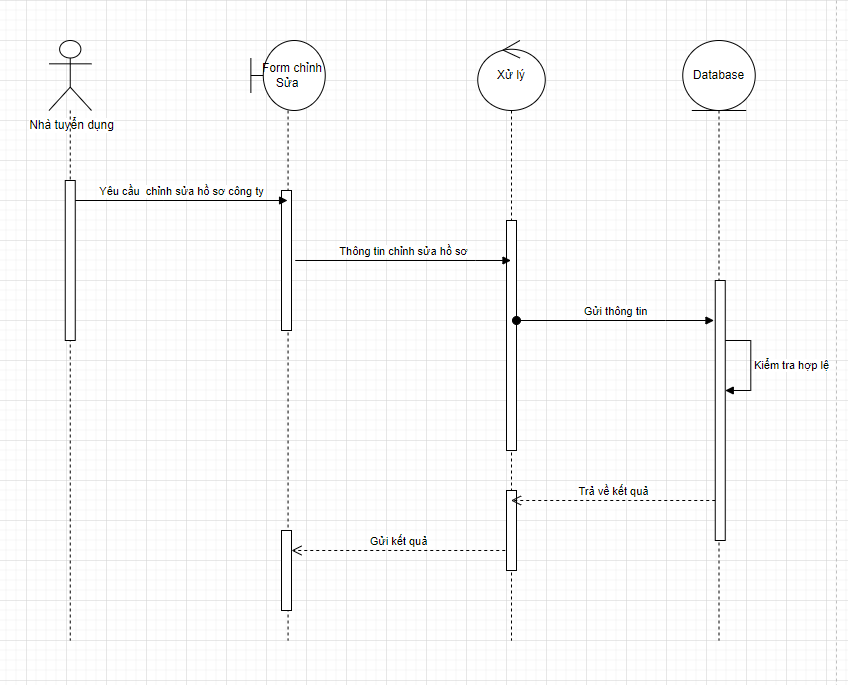
**Hình 3. 22. Biểu đồ tuần tự xóa kinh nghiệm làm việc**

* + - 1. **Biểu đồ tuần tự chức năng chỉnh sửa kinh nghiệm**

****

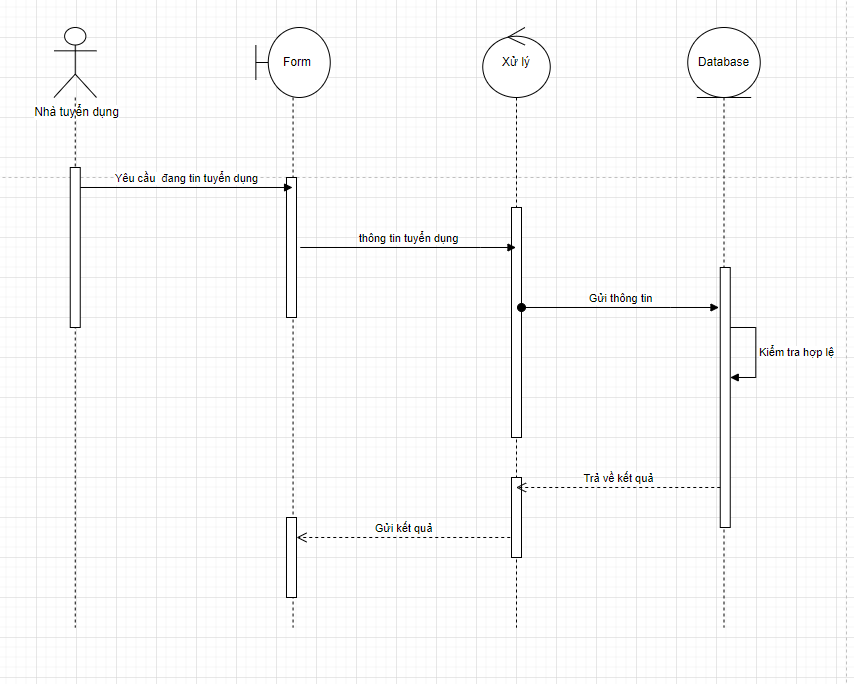
**Hình 3. 23. Biểu đồ tuần tự chức năng chỉnh sửa kinh nghiệm**

* + - 1. **Biểu đồ tuần tự cập nhật hồ sơ công ty**

****

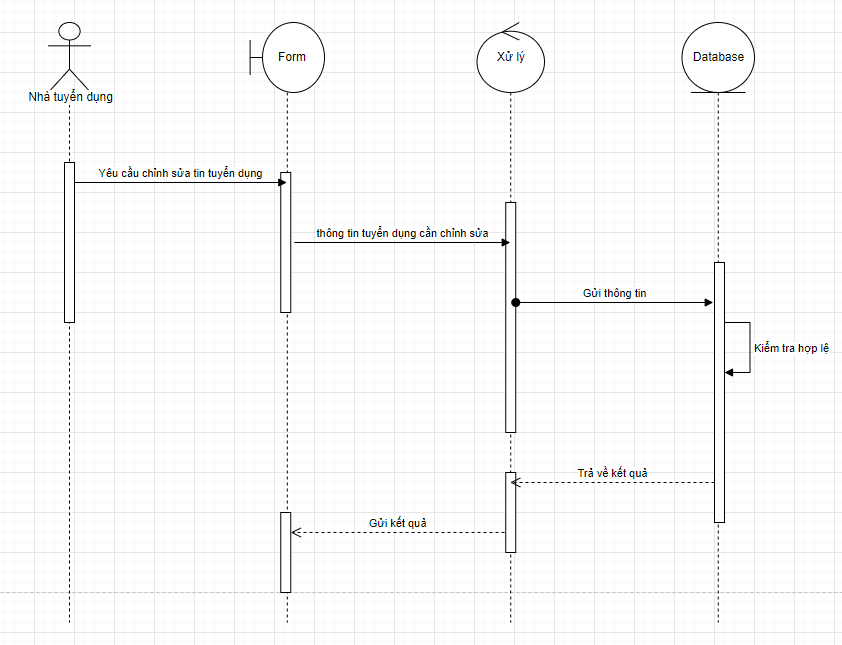
**Hình 3. 24. Biểu đồ tuần tự cập nhật hồ sơ công ty**

* + - 1. **Biểu đồ tuần tự đăng tin tuyển dụng**

****

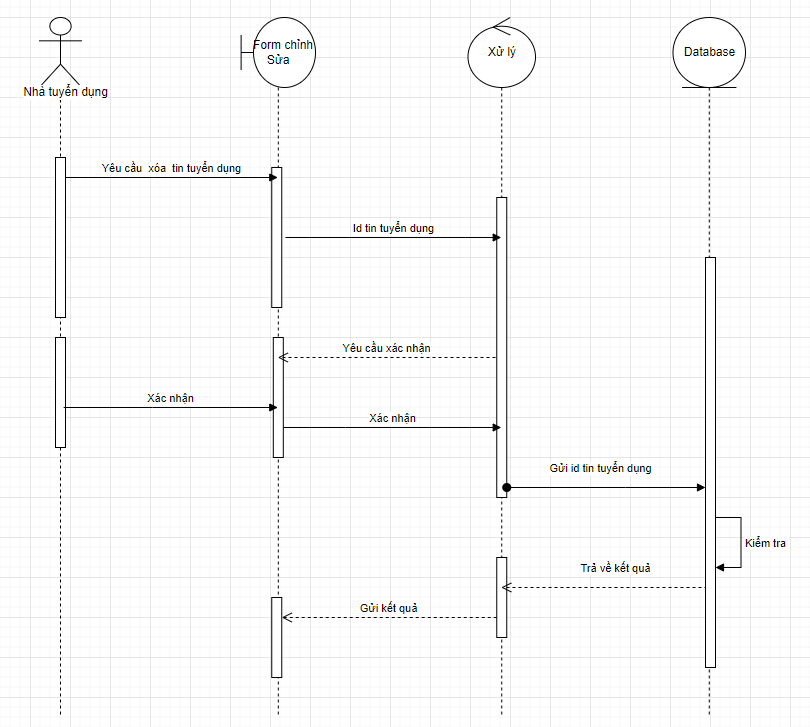
**Hình 3. 25. Biểu đồ tuần tự đăng tin tuyển dụng**

* + - 1. **Biểu đồ tuần tự chỉnh sửa tin tuyển dụng**

****

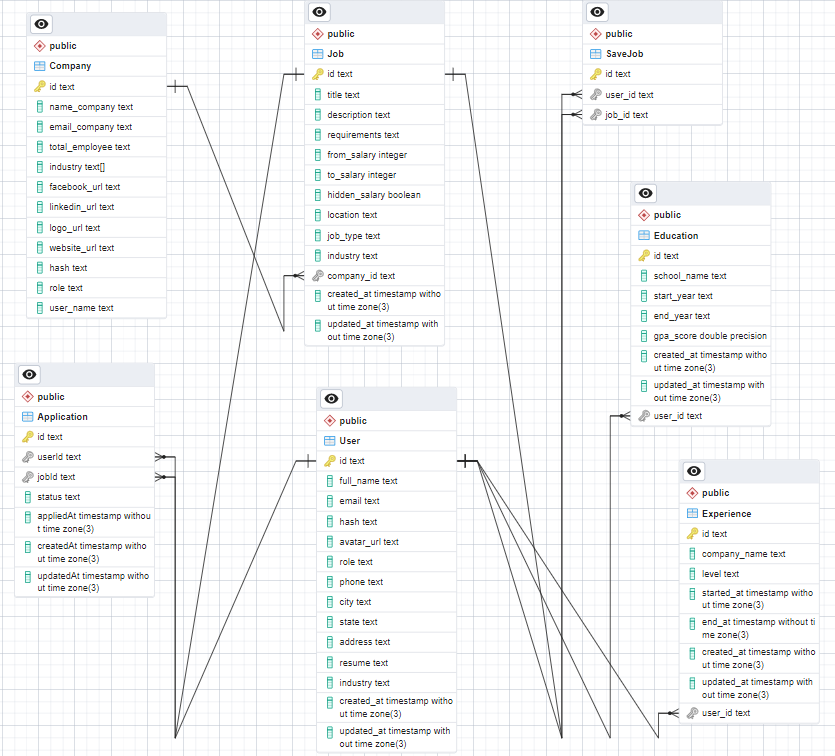
**Hình 3. 26. Biểu đồ tuần tự chỉnh sửa tin tuyển dụng**

* + - 1. **Biểu đồ tuần tự xóa tin tuyển dụng**

****

**Hình 3. 27. Biểu đồ tuần tự xóa tin tuyển dụng**

* 1. **Thiết kế cơ sở dữ liệu**
     1. **Thiết kế cơ sở dữ liệu mức logic**

****

**Hình 3. 28. Mô hình ERD**

* + 1. **Thiết kế cơ sở dữ liệu mức vật lý**
       1. **Bảng User**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Rằng buộc** | **Mô tả** |
| **1** | id | text | Primary key | Mã người dùng |
| **2** | full\_name | text |  | Họ tên người dùng |
| **3** | email | text |  | Địa chỉ email |
| **4** | hash | text |  | Mật khẩu đã mã hóa |
| **5** | avatar\_url | text |  | Đường dẫn ảnh đại diện |
| **6** | role | Text |  | Vai trò trong hệ thống |
| **7** | phone | Text |  | Điện thoại |
| **8** | city | Text |  | Tỉnh / Tp |
| **9** | state | Text |  | Quận huyện |
| **10** | address | Text |  | Địa chỉ |
| **11** | resume | Text |  | Đường dẫn cv |
| **12** | industry | Text |  | Ngành nghề |
| **13** | created\_at | timestamp |  | Ngày tạo |
| **14** | updated\_at | timestamp |  | Ngày cập nhật |

**Bảng 3. 1. Bảng user**

* + - 1. **Bảng Company**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Rằng buộc** | **Mô tả** |
| **1** | id | text | Primary key | Mã công ty |
| **2** | name\_company | text |  | Tên công ty |
| **3** | email\_company | text |  | Email công ty |
| **4** | total\_employee | text |  | Tổng số nhân viên |
| **5** | industry | text[] |  | Ngành nghề |
| **6** | facebook\_url | text |  | Đường dẫn facebook |
| **7** | linkedin\_url | text |  | Đường dẫn linkedin |
| **8** | logo\_url | text |  | Đường dẫn logo |
| **9** | website\_url | text |  | Đường dẫn website |
| **10** | hash | text |  | Mật khẩu mã hóa |
| **11** | role | text |  | Vai trò trong hệ thống |
| **12** | user\_name | text |  | Tên đang nhập |

**Bảng 3. 2. Bảng company**

* + - 1. **Bảng Job**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Rằng buộc** | **Mô tả** |
| **1** | id | Text | Primary key | Mã việc làm |
| **2** | title | text |  | Tiêu đề việc làm |
| **3** | description | text |  | Mô tả công việc |
| **4** | requirements | text |  | Yêu cầu công việc |
| **5** | from\_salary | interger |  | Lương từ |
| **6** | to\_salary | interger |  | Lương đến |
| **7** | hidden\_salary | boolean |  | Ẩn mức lương |
| **8** | location | Text |  | Địa điểm làm việc |
| **9** | job\_type | Text |  | Loại hình công việc |
| **10** | company\_id | Text | Khóa ngoại | Mã công ty |
| **11** | created\_at | timestamp |  | Ngày tạo |
| **12** | updated\_at | timestamp |  | Ngày cập nhật |

**Bảng 3. 3. Bảng job**

* + - 1. **Bảng Education**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Rằng buộc** | **Mô tả** |
| **1** | id | text | Primary key | Mã học vấn |
| **2** | school\_name | text |  | Tên trường |
| **3** | start\_year | text |  | Năm bắt đầu |
| **4** | end\_year | text |  | Năm kết thúc |
| **5** | gpa\_score | double |  | Điểm tích lũy |
| **6** | created\_at | timestamp |  | Ngày tạo |
| **7** | updated\_at | timestamp |  | Ngày cập nhật |

**Bảng 3. 4. Bảng education**

* + - 1. **Bảng Experience**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Rằng buộc** | **Mô tả** |
| **1** | id | text | Primary key | Mã học vấn |
| **2** | company\_name | text |  | Tên công ty |
| **3** | level | text |  | Cấp bậc |
| **4** | started\_at | timestamp |  | Ngày bắt đầu |
| **5** | ended\_at | timestamp |  | Nghỉ việc ngày |
| **6** | created\_at | timestamp |  | Ngày tạo |
| **7** | updated\_at | timestamp |  | Ngày cập nhật |

**Bảng 3. 5. Bảng experience**

* + - 1. **Bảng SaveJob**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Rằng buộc** | **Mô tả** |
| **1** | id | text | Primary key | Mã học vấn |
| **2** | user\_id | text | Khóa ngoại | Mã người dùng |
| **3** | job\_id | text | Khóa ngoại | Mã việc làm |

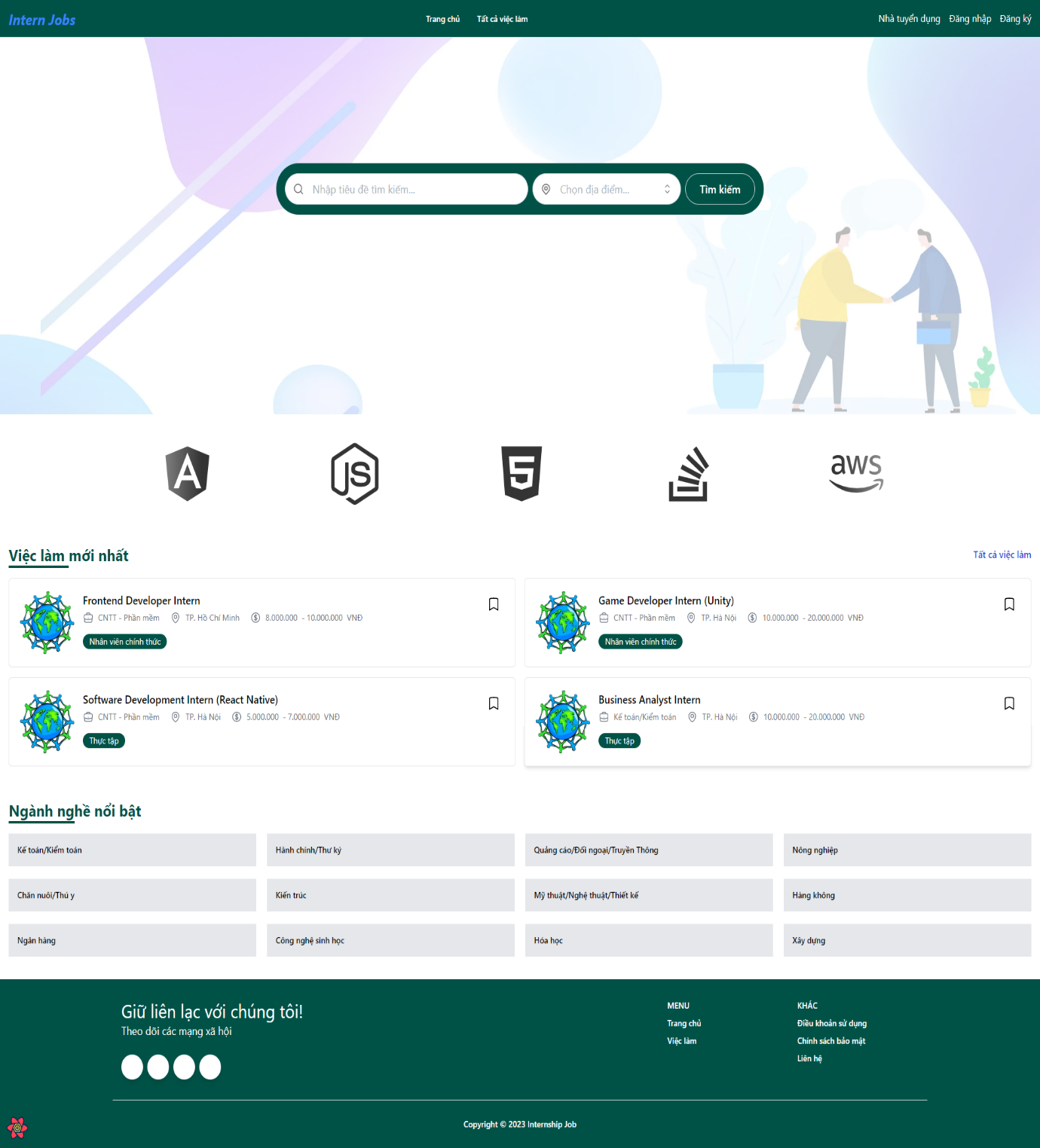
**Bảng 3. 6. Bảng savejob**

* + - 1. **Bảng Application**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên trường** | **Kiểu dữ liệu** | **Rằng buộc** | **Mô tả** |
| **1** | id | text | Primary key | Mã học vấn |
| **2** | user\_id | text | Khóa ngoại | Mã người dùng |
| **3** | job\_id | text | Khóa ngoại | Mã việc làm |
| **4** | status | text |  | Trạng thái |
| **5** | appliedAt | timestamp |  |  |
| **6** | createdAt | timestamp |  |  |
| **7** | updatedAt | timestamp |  | Ngày cập nhật |

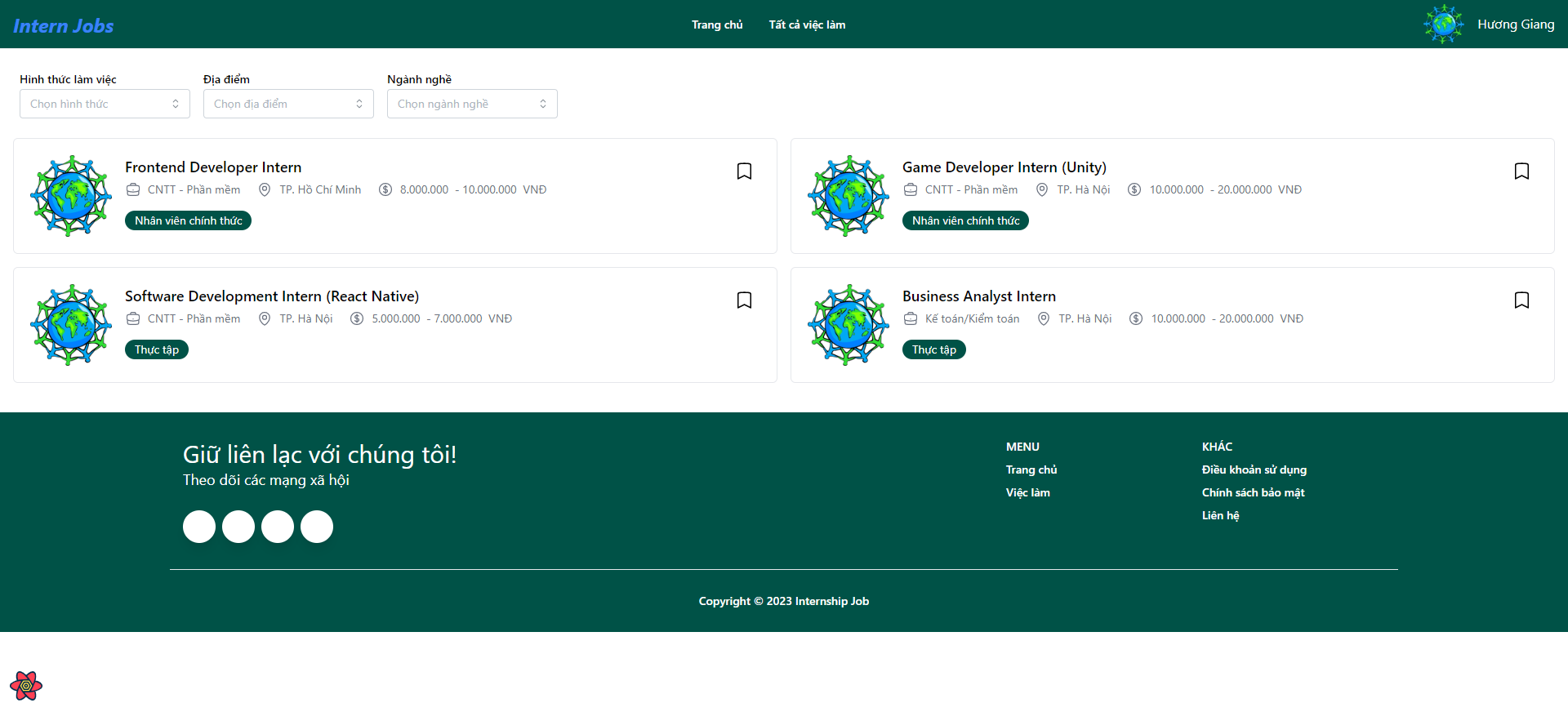
**Bảng 3. 7. Bảng application**

* 1. **Thiết kế giao diện và xây dựng ứng dụng**
     1. **Giao diện trang chủ**

****

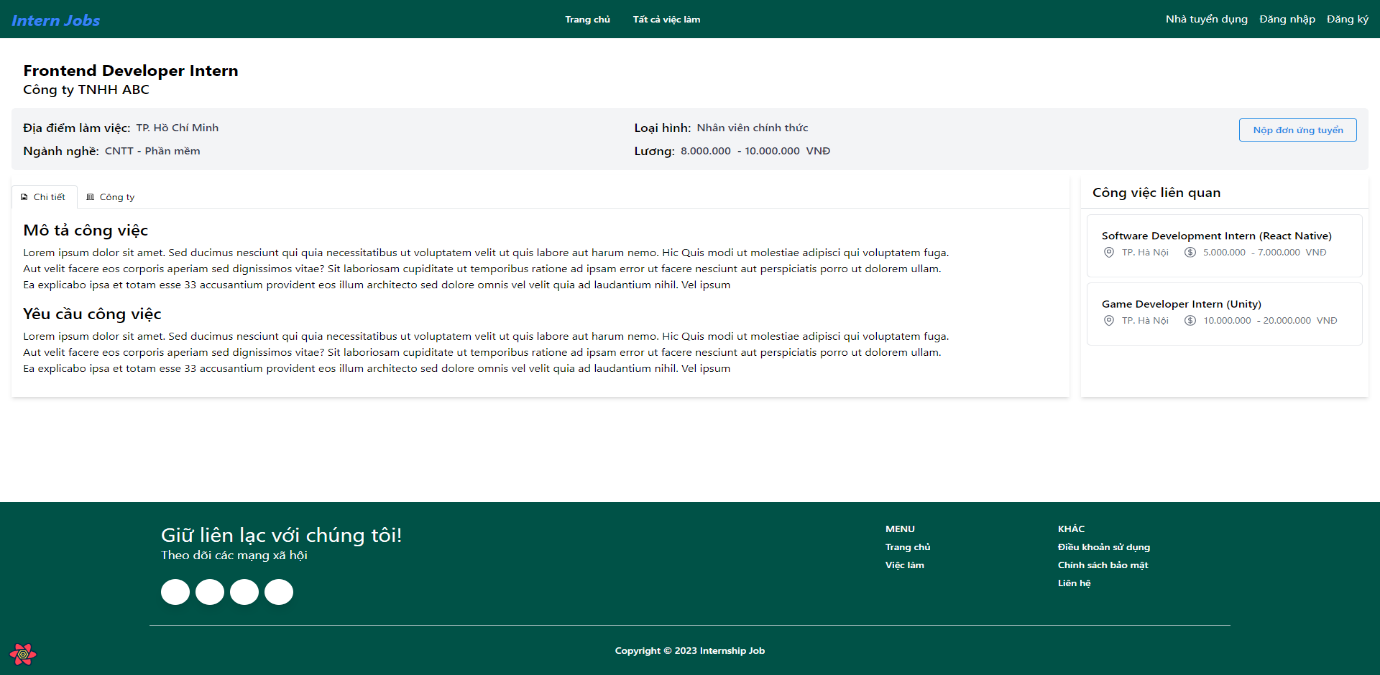
**Hình 3. 29. Giao diện trang chủ**

* + 1. **Giao diện tất cả việc làm**

****

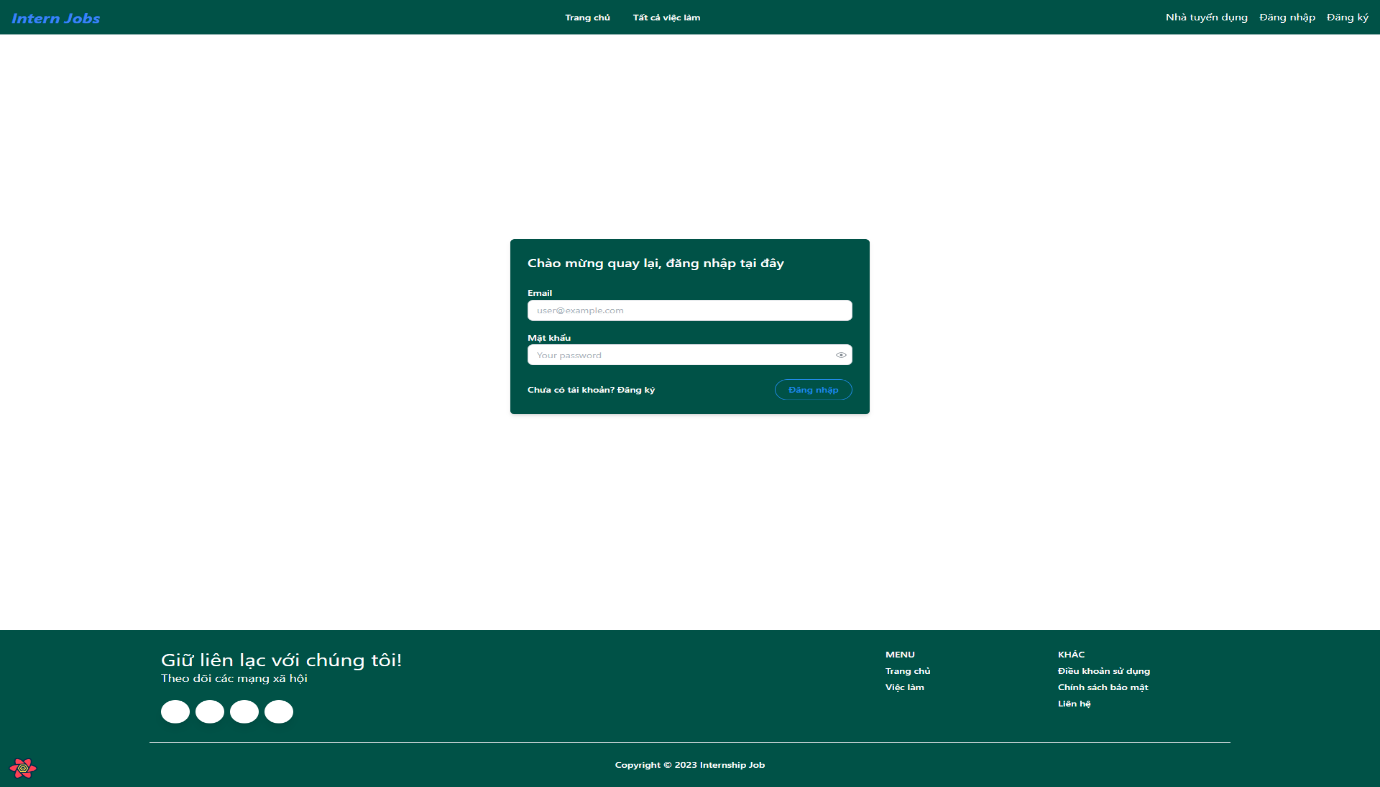
**Hình 3. 30. Giao diện trang tất cả việc làm**

* + 1. **Giao diện chi tiết việc làm**

****

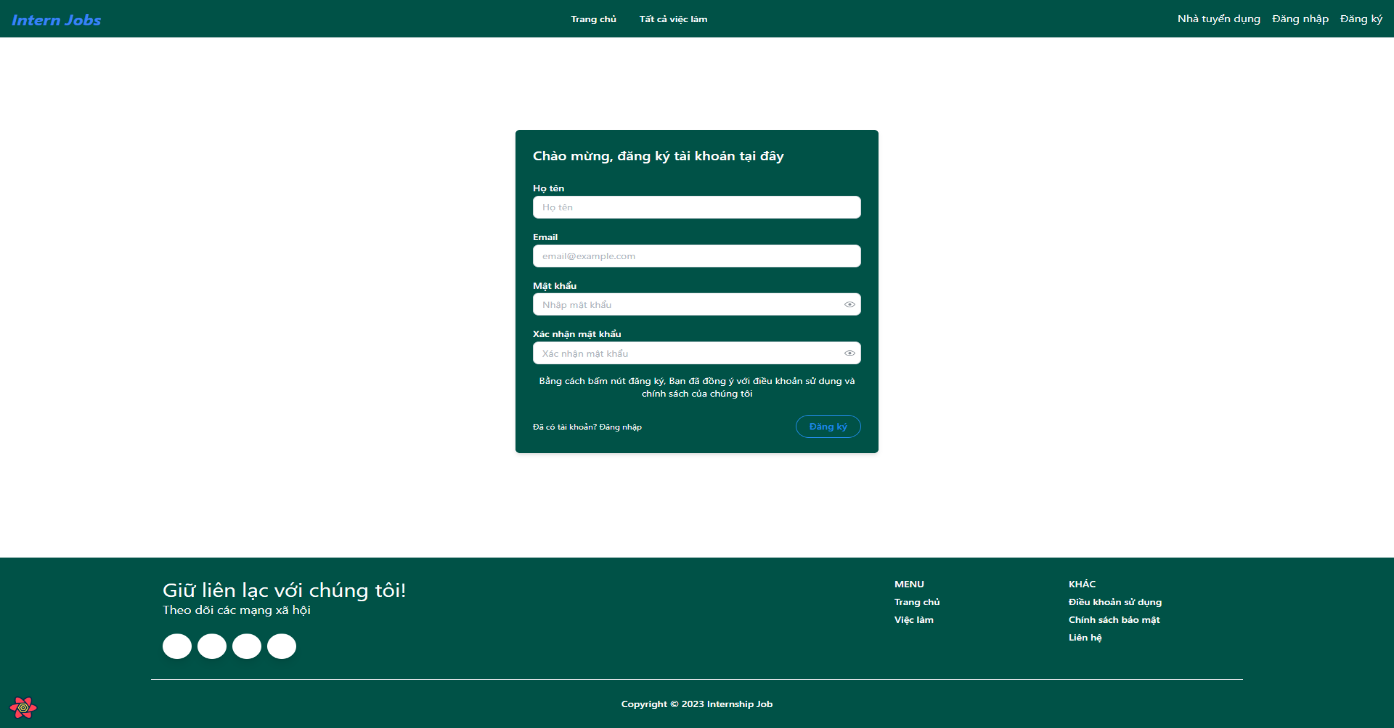
**Hình 3. 31. Giao diện trang chi tiết việc làm**

* + 1. **Giao diện đăng nhập cho ứng viên**

****

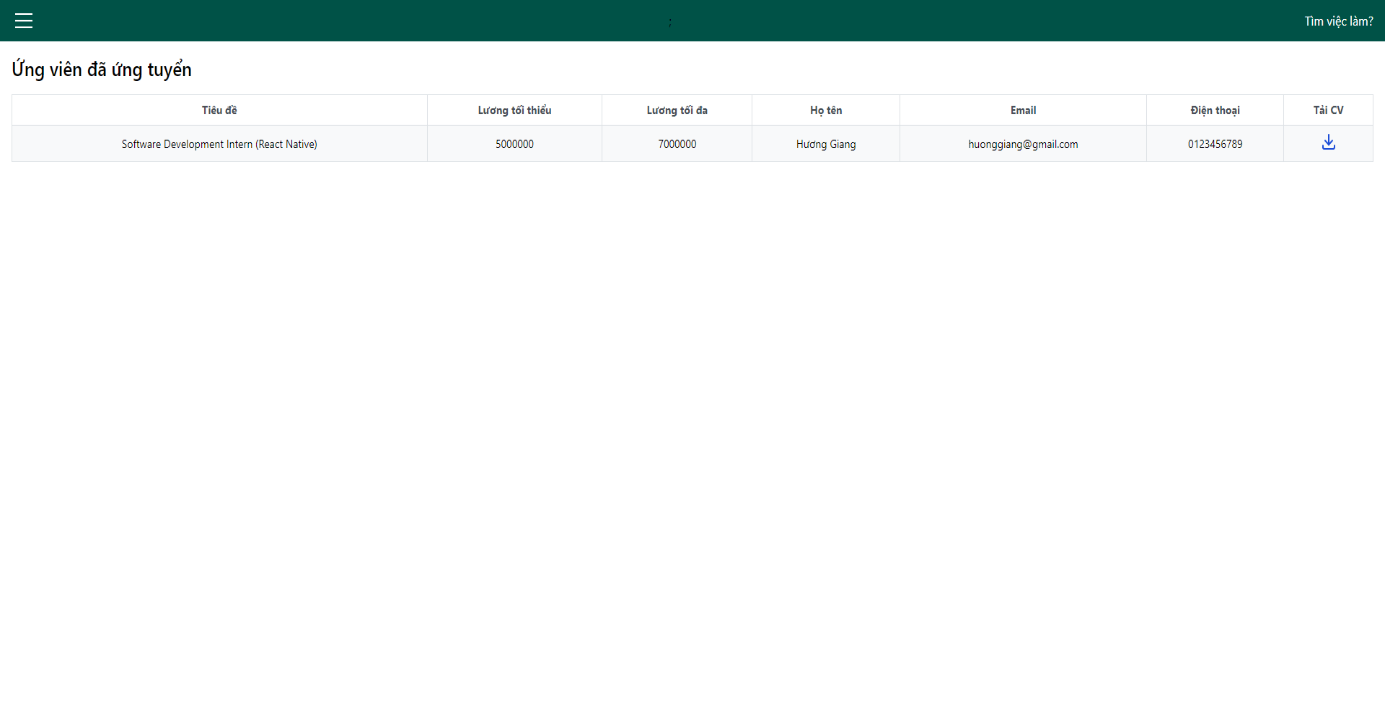
**Hình 3. 32. Giao diện đang nhập cho ứng viên**

* + 1. **Giao diện đăng ký cho ứng viên**

****

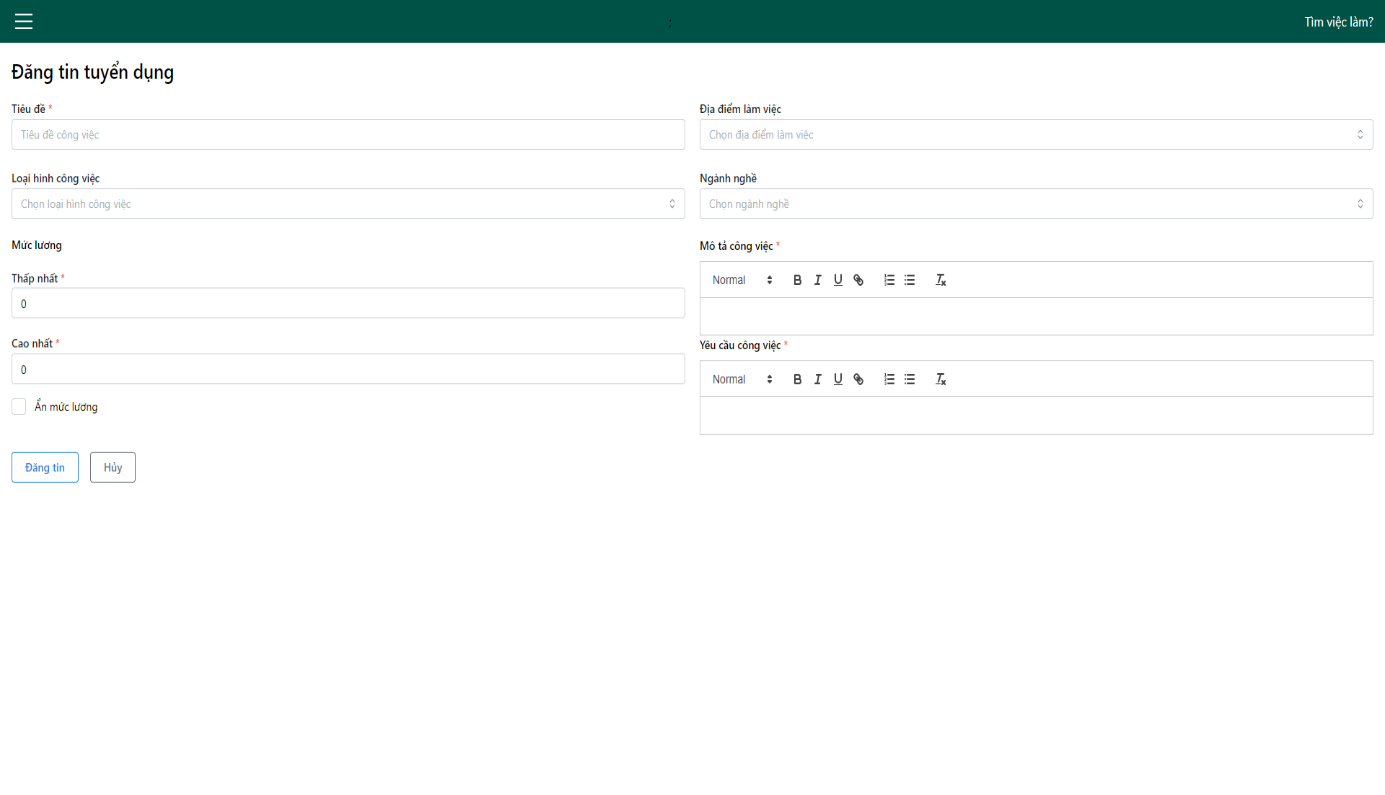
**Hình 3. 33. Giao diện đăng ký cho ứng viên**

* + 1. **Giao diện quản lý thông tin ứng viên đã ứng tuyển**

****

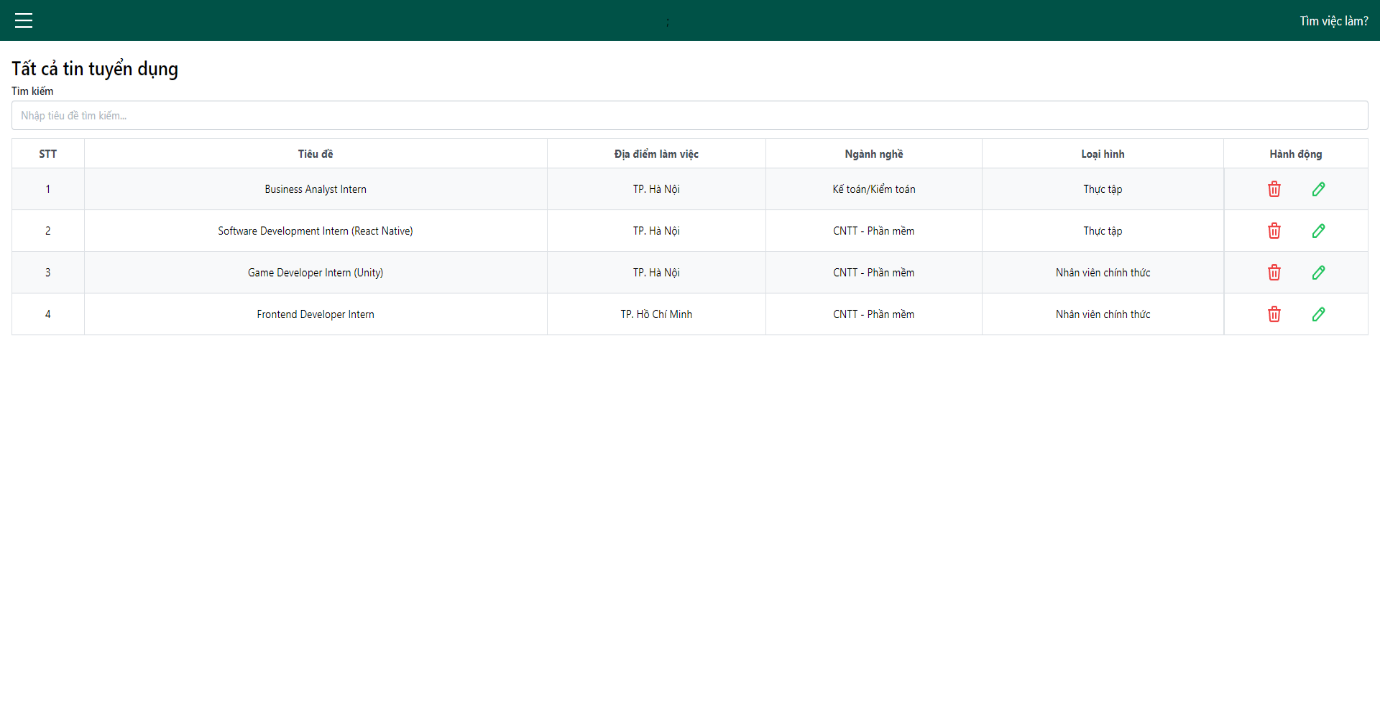
Hình 3. 34. Giao diện quản lý ứng viên

* + 1. **Giao diện đăng tin tuyển dụng**

****

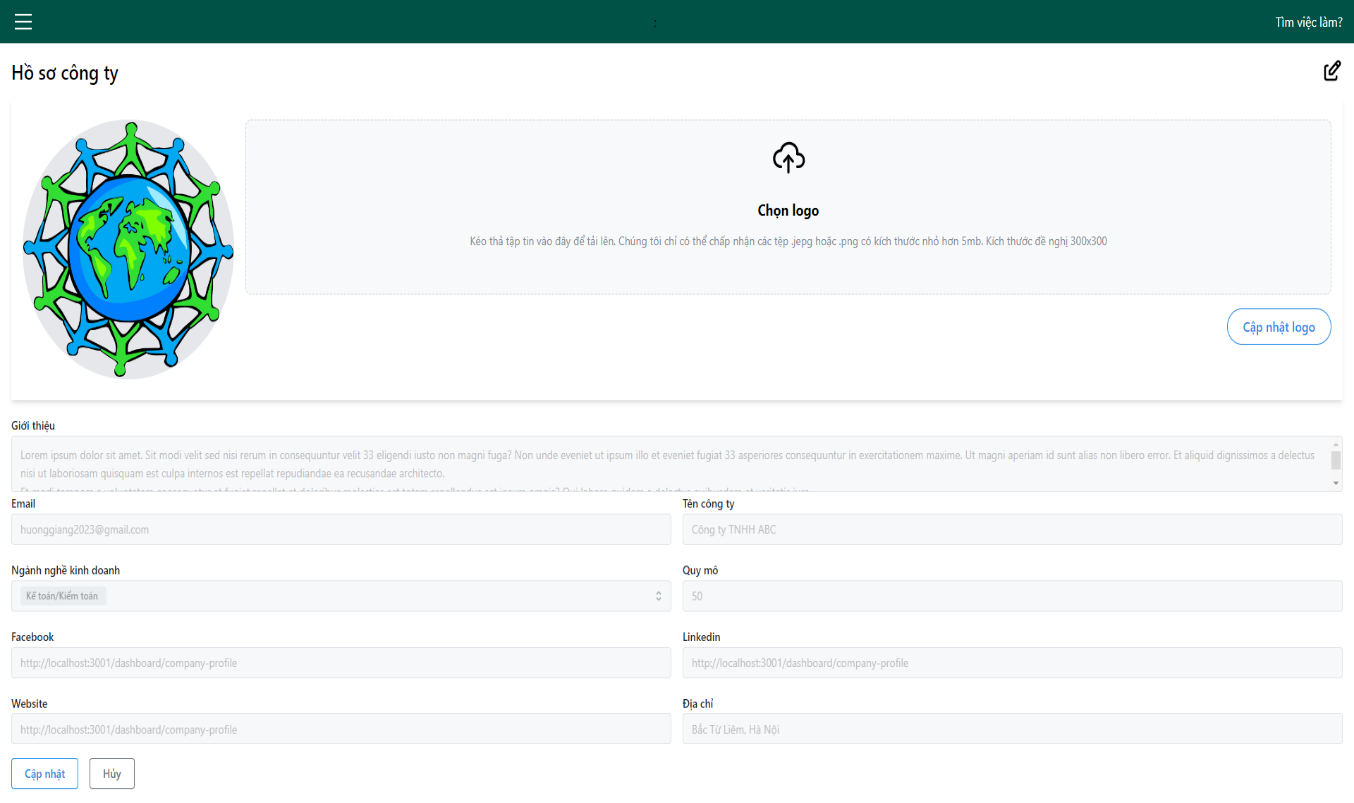
**Hình 3. 35. Giao diện đăng tin tuyển dụng**

* + 1. **Giao diện quản lý việc làm**

****

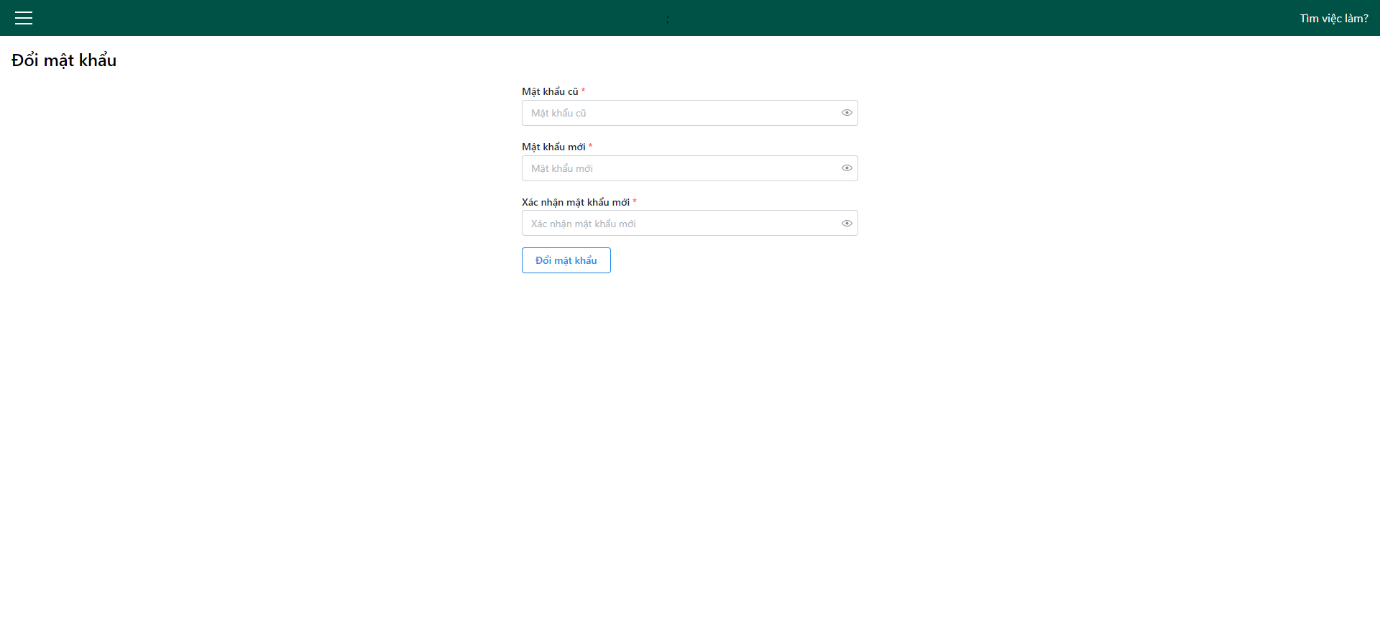
**Hình 3. 36. Giao diện quản lý tin tuyển dụng**

* + 1. **Giao diện hồ sơ công ty**

****

**Hình 3. 37. Giao diện cập nhật hồ sơ công ty**

* + 1. **Giao diện đổi mật khẩu**

****

**Hình 3. 38. Giao diện đổi mật khẩu**