

**ĐẠI HỌC BÁCH KHOA HÀ NỘI**

**ĐỒ ÁN TỐT NGHIỆP**

**Thiết kế xây dựng nền tảng đăng và tìm việc làm**

**Vũ Bá Lượng**

luong.vb194109@sis.hust.edu.vn

**Ngành Công nghệ thông tin và truyền thông**

**Giảng viên hướng dẫn:** TS. Phạm Quang Dũng

\_\_\_\_\_

Chữ ký GVHD

**Khoa:** Khoa học máy tính

**Trường:** Công nghệ Thông tin và Truyền thông

**HÀ NỘI, 12/2023**

# LỜI CẢM ƠN

Để hoàn thành đồ án của mình, em xin gửi lời cảm ơn chân thành và sâu sắc nhất đến thầy Phạm Quang Dũng, người đã tận tình hướng dẫn em trong suốt quá trình thực hiện đồ án của mình. Dưới sự hướng dẫn của thầy, em không chỉ có được những kiến thức quý báu mà còn thấm nhuần đạo đức làm việc nghiêm túc và hiệu quả. Những phẩm chất này rất cần thiết cho việc học tập và sự nghiệp trong tương lai của em.

Em xin chân thành cảm ơn các thầy cô trong bộ môn Khoa học máy tính đã nhiệt tình truyền đạt những kiến thức quý báu cho em trong suốt 4 năm đại học. Những kiến thức này không chỉ giúp em thành công trong đồ án này mà còn là tài sản quý giá để em tự tin theo đuổi thành công.

Em cũng xin gửi lời cảm ơn đến gia đình, bạn bè, người thân và đặc biệt là các bạn cùng lớp đã hỗ trợ tận tình cho em trong suốt những năm sinh viên.

Em chúc tất cả mọi người sức khỏe và nhiệt huyết trong cả cuộc sống cá nhân và sự nghiệp.

# TÓM TẮT NỘI DUNG ĐỒ ÁN

Ngày nay với sự phát triển mạnh mẽ của nền kinh tế hội nhập đã tạo ra cho chúng ta rất nhiều những cơ hội việc làm khác nhau. Tuy vậy những năm gần đây nguồn lực lao động ở nước ta rất dồi dào và rất rẻ song tỷ lệ thất nghiệp vẫn còn khá cao. Vấn đề đặt ra là làm thế nào để kết nối người lao động đến được với những cơ hội việc làm đó.

Để giải quyết các vấn đề mà người tìm việc đang tìm kiếm cơ hội phù hợp hoặc các công ty đang tìm kiếm nhân viên tiềm năng, các trang web tìm kiếm việc làm đã thu hút được sự chú ý đáng kể. Những nền tảng như vậy tạo ra một môi trường nơi ứng viên và nhà tuyển dụng có thể kết nối dễ dàng, nhanh chóng và thuận tiện, từ đó tăng triển vọng việc làm. Hơn nữa, các trang web tìm kiếm việc làm này cung cấp các tính năng bổ sung như mẫu CV, đề xuất công việc và hướng dẫn tìm kiếm việc làm.

Hiểu được thực trạng đó, chúng tôi quyết định thiết kế và xây dựng một phần mềm ứng dụng web kết nối nhà tuyển dụng và ứng viên cho phép tạo và tìm kiếm việc làm.

Sinh viên thực hiện  
(Ký và ghi rõ họ tên)

## MỤC LỤC

|  |          |
|--|----------|
| <b>CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI.....</b>                          | <b>1</b> |
| 1.1 Đặt vấn đề.....  | 1        |
| 1.2 Mục tiêu và phạm vi đề tài.....                              | 2        |
| 1.3 Định hướng giải pháp.....                                    | 2        |
| 1.4 Bố cục đồ án .....   | 3        |
| <b>CHƯƠNG 2. KHẢO SÁT VÀ PHÂN TÍCH YÊU CẦU.....</b>              | <b>4</b> |
| 2.1 Khảo sát hiện trạng .....                                    | 4        |
| 2.2 Tổng quan chức năng .....                                    | 4        |
| 2.2.1 Biểu đồ use case tổng quát .....                           | 6        |
| 2.2.2 Biểu đồ use case phân rã của quản trị viên hệ thống .....  | 7        |
| 2.2.3 Biểu đồ use case phân rã của ứng viên .....                | 8        |
| 2.2.4 Biểu đồ use case phân rã của nhà tuyển dụng .....          | 9        |
| 2.3 Đặc tả chức năng .....                                       | 10       |
| 2.3.1 Đặc tả use case Đăng nhập.....                             | 10       |
| 2.3.2 Đặc tả use case Đăng ký .....                              | 13       |
| 2.3.3 Đặc tả use case Tìm kiếm .....                             | 15       |
| 2.3.4 Đặc tả use case Xem thông tin việc làm.....                | 17       |
| 2.3.5 Đặc tả use case Ứng tuyển việc làm .....                   | 19       |
| 2.3.6 Đặc tả use case Cập nhật thông tin cá nhân.....            | 21       |
| 2.3.7 Đặc tả use case Quản lý đơn ứng tuyển.....                 | 23       |
| 2.3.8 Đặc tả use case Cập nhật CV cá nhân.....                   | 25       |
| 2.3.9 Đặc tả use case Quản lý ứng tuyển của nhà tuyển dụng ..... | 28       |
| 2.3.10 Đặc tả use case Quản lý việc làm .....                    | 31       |
| 2.3.11 Đặc tả use case Liên hệ ứng viên .....                    | 34       |

|  |           |
|--|-----------|
| 2.3.12 Đặc tả use case Quản lý danh mục .....        | 35        |
| 2.3.13 Đặc tả use case Quản lý quảng cáo.....        | 37        |
| 2.3.14 Đặc tả use case Quản lý tài khoản .....       | 39        |
| 2.4 Yêu cầu phi chức năng .....                      | 41        |
| <b>CHƯƠNG 3. CÔNG NGHỆ SỬ DỤNG.....</b>              | <b>43</b> |
| 3.1 Hệ quản trị cơ sở dữ liệu MySQL.....             | 43        |
| 3.1.1 Giới thiệu .....                               | 43        |
| 3.1.2 Đặc trưng .....                                | 43        |
| 3.1.3 Ưu điểm của MySQL .....                        | 43        |
| 3.2 Tổng quan về Java.....                           | 45        |
| 3.2.1 Giới thiệu .....                               | 45        |
| 3.2.2 Ứng dụng của Java .....                        | 46        |
| 3.2.3 Đặc điểm của Java.....                         | 46        |
| 3.2.4 Java Platforms .....                           | 46        |
| 3.2.5 Tiêu chuẩn của môi trường Java điển hình ..... | 46        |
| 3.3 Tổng quan về Spring Boot .....                   | 47        |
| 3.3.1 Những tính năng nổi bật của Spring Boot .....  | 47        |
| 3.3.2 Một số tính năng đặc biệt.....                 | 47        |
| 3.3.3 Các tính năng quan trọng của Spring Boot.....  | 47        |
| 3.4 Tổng quan về React.....                          | 48        |
| 3.4.1 Khái niệm chính.....                           | 48        |
| 3.4.2 Ưu điểm của React .....                        | 48        |
| 3.5 Tổng quan về HTML và CSS .....                   | 49        |
| 3.5.1 HTML (Hypertext Markup Language) .....         | 49        |
| 3.5.2 CSS (Cascading Style Sheets) .....             | 49        |
| 3.5.3 Hợp nhất .....                                 | 50        |

|   |           |
|---|-----------|
| 3.5.4 Thực tiễn.....  | 50        |
| 3.5.5 Sự phát triển .....                                       | 50        |
| 3.5.6 Ứng dụng .....  | 50        |
| <b>CHƯƠNG 4. THIẾT KẾ, TRIỂN KHAI VÀ ĐÁNH GIÁ HỆ THỐNG ....</b> | <b>52</b> |
| 4.1 Thiết kế kiến trúc.....                                     | 52        |
| 4.1.1 Lựa chọn kiến trúc phần mềm .....                         | 52        |
| 4.1.2 Thiết kế tổng quan.....                                   | 53        |
| 4.1.3 Thiết kế chi tiết gói .....                               | 54        |
| 4.2 Thiết kế chi tiết.....                                      | 57        |
| 4.2.1 Thiết kế giao diện người dùng .....                       | 57        |
| 4.2.2 Thiết kế database .....                                   | 63        |
| 4.3 Xây dựng ứng dụng.....                                      | 67        |
| 4.3.1 Thư viện và công cụ sử dụng .....                         | 67        |
| 4.3.2 Minh họa các chức năng chính .....                        | 67        |
| 4.4 Triển khai .....  | 69        |
| <b>CHƯƠNG 5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN .....</b>             | <b>70</b> |
| 5.1 Kết luận .....  | 70        |
| 5.2 Hướng phát triển.....                                       | 70        |
| <b>TÀI LIỆU THAM KHẢO.....</b>                                  | <b>72</b> |



## DANH MỤC HÌNH VẼ

|           |  |    |
|-----------|--|----|
| Hình 2.1  | Biểu đồ use case tổng quát . . . . .   | 6  |
| Hình 2.2  | Biểu đồ use case phân rã của quản trị viên hệ thống . . . . .                | 7  |
| Hình 2.3  | Biểu đồ use case phân rã của ứng viên . . . . .                              | 8  |
| Hình 2.4  | Biểu đồ use case phân rã của nhà tuyển dụng . . . . .                        | 9  |
| Hình 2.5  | Biểu đồ hoạt động của usecase Đăng nhập . . . . .                            | 11 |
| Hình 2.6  | Biểu đồ tuần tự của usecase Đăng nhập . . . . .                              | 12 |
| Hình 2.7  | Biểu đồ hoạt động của usecase Đăng ký . . . . .                              | 14 |
| Hình 2.8  | Biểu đồ tuần tự của usecase Đăng ký . . . . .                                | 15 |
| Hình 2.9  | Biểu đồ hoạt động của usecase Tìm kiếm . . . . .                             | 16 |
| Hình 2.10 | Biểu đồ tuần tự của usecase Tìm kiếm . . . . .                               | 17 |
| Hình 2.11 | Biểu đồ hoạt động của usecase Xem thông tin việc làm . . . . .               | 18 |
| Hình 2.12 | Biểu đồ tuần tự của usecase Xem thông tin việc làm . . . . .                 | 18 |
| Hình 2.13 | Biểu đồ hoạt động của usecase Ứng tuyển việc làm . . . . .                   | 20 |
| Hình 2.14 | Biểu đồ tuần tự của usecase Ứng tuyển việc làm . . . . .                     | 21 |
| Hình 2.15 | Biểu đồ hoạt động của usecase Cập nhật thông tin cá nhân . . . . .           | 22 |
| Hình 2.16 | Biểu đồ tuần tự của usecase Cập nhật thông tin cá nhân . . . . .             | 23 |
| Hình 2.17 | Biểu đồ hoạt động của usecase Quản lý đơn ứng tuyển . . . . .                | 24 |
| Hình 2.18 | Biểu đồ tuần tự của usecase Quản lý đơn ứng tuyển . . . . .                  | 25 |
| Hình 2.19 | Biểu đồ hoạt động của usecase Cập nhật CV cá nhân . . . . .                  | 26 |
| Hình 2.20 | Biểu đồ tuần tự của usecase Cập nhật CV cá nhân . . . . .                    | 27 |
| Hình 2.21 | Biểu đồ hoạt động của usecase Quản lý ứng tuyển của nhà tuyển dụng . . . . . | 29 |
| Hình 2.22 | Biểu đồ tuần tự của usecase Quản lý ứng tuyển của nhà tuyển dụng . . . . .   | 30 |
| Hình 2.23 | Biểu đồ hoạt động của usecase Quản lý việc làm . . . . .                     | 32 |
| Hình 2.24 | Biểu đồ tuần tự của usecase Quản lý việc làm . . . . .                       | 33 |
| Hình 2.25 | Biểu đồ hoạt động của usecase Liên hệ ứng viên . . . . .                     | 34 |
| Hình 2.26 | Biểu đồ tuần tự của usecase Liên hệ ứng viên . . . . .                       | 35 |
| Hình 2.27 | Biểu đồ hoạt động của usecase Quản lý danh mục . . . . .                     | 36 |
| Hình 2.28 | Biểu đồ tuần tự của usecase Quản lý danh mục . . . . .                       | 37 |
| Hình 2.29 | Biểu đồ hoạt động của usecase Quản lý quảng cáo . . . . .                    | 38 |
| Hình 2.30 | Biểu đồ tuần tự của usecase Quản lý quảng cáo . . . . .                      | 39 |
| Hình 2.31 | Biểu đồ hoạt động của usecase Quản lý tài khoản . . . . .                    | 40 |
| Hình 2.32 | Biểu đồ tuần tự của usecase Quản lý tài khoản . . . . .                      | 40 |

|           |  |    |
|-----------|--|----|
| Hình 4.1  | Biểu đồ gói trong Spring Boot . . . . .        | 54 |
| Hình 4.2  | Biểu đồ chi tiết gói user . . . . .            | 55 |
| Hình 4.3  | Biểu đồ chi tiết gói của admin . . . . .       | 56 |
| Hình 4.4  | Biểu đồ chi tiết gói của ứng viên . . . . .    | 56 |
| Hình 4.5  | Biểu đồ lớp của nhà tuyển dụng . . . . .       | 57 |
| Hình 4.6  | Nút đăng nhập và đăng ký . . . . .             | 58 |
| Hình 4.7  | Phần header của trang web . . . . .            | 59 |
| Hình 4.8  | Thông báo đăng nhập thành công . . . . .       | 59 |
| Hình 4.9  | Thông báo đăng nhập thất bại . . . . .         | 59 |
| Hình 4.10 | Đăng nhập dành cho ứng viên . . . . .          | 60 |
| Hình 4.11 | Đăng nhập dành cho Nhà tuyển dụng . . . . .    | 61 |
| Hình 4.12 | Thanh điều hướng của quản trị viên . . . . .   | 62 |
| Hình 4.13 | Thiết kế database . . . . .                    | 63 |
| Hình 4.14 | Chức năng lọc công việc . . . . .              | 68 |
| Hình 4.15 | Chức năng lọc công việc và ứng tuyển . . . . . | 68 |
| Hình 4.16 | Chức năng tạo CV . . . . .                     | 69 |
| Hình 4.17 | Chức năng xem trước CV . . . . .               | 69 |

## **DANH MỤC BẢNG BIỂU**

|           |   |    |
|-----------|---|----|
| Bảng 2.1  | Use case của quản trị viên hệ thống . . . . .   | 7  |
| Bảng 2.2  | Use case của ứng viên . . . . .                 | 8  |
| Bảng 2.3  | Use case của nhà tuyển dụng . . . . .           | 9  |
| <br>      |   |    |
| Bảng 4.1  | User . . . . .                                  | 63 |
| Bảng 4.2  | Company . . . . .                               | 64 |
| Bảng 4.3  | Recruitment . . . . .                           | 64 |
| Bảng 4.4  | Curriculum Vitae . . . . .                      | 65 |
| Bảng 4.5  | Advertisement . . . . .                         | 65 |
| Bảng 4.6  | Jobseeker . . . . .                             | 65 |
| Bảng 4.7  | Job . . . . .                                   | 66 |
| Bảng 4.8  | Recruiter . . . . .                             | 66 |
| Bảng 4.9  | Category . . . . .                              | 67 |
| Bảng 4.10 | Roles . . . . .                                 | 67 |
| Bảng 4.11 | User Roles . . . . .                            | 67 |
| Bảng 4.12 | Danh sách thư viện và công cụ sử dụng . . . . . | 67 |

## **DANH MỤC THUẬT NGỮ VÀ TỪ VIẾT TẮT**

| <b>Thuật ngữ</b> | <b>Ý nghĩa</b>  |
|------------------|---|
| API              | Giao diện lập trình ứng dụng<br>(Application Programming Interface)       |
| CV               | Sơ yếu lý lịch (Curriculum Vitae)   |
| EUD              | Phát triển ứng dụng người dùng<br>cuối(End-User Development)              |
| GWT              | Công cụ lập trình Javascript bằng Java<br>của Google (Google Web Toolkit) |
| HTML             | Ngôn ngữ đánh dấu siêu văn bản<br>(HyperText Markup Language)             |

# CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI

## 1.1 Đặt vấn đề

Công nghệ Internet đóng vai trò quan trọng trong cuộc sống hiện đại. Ngày nay, mọi thông tin đều được truyền qua Internet với tốc độ nhanh và ngày càng đa dạng. Thông tin việc làm cũng là một trong số đó. Người tìm việc và nhà tuyển dụng trong thời đại thông tin có thể tìm thấy điểm chung trong danh sách việc làm.

Một trong những yếu tố quan trọng giúp tìm việc làm trực tuyến trở thành một xu hướng gần đây bởi vì thông tin công việc được cung cấp miễn phí. Người tìm việc trực tuyến có thể dễ dàng tìm được việc làm phù hợp trên các trang tìm việc làm trực tuyến mà không cần phải tốn bất kỳ khoản tiền nào.

Ngoài ra, hình thức tuyển dụng qua các trang tìm việc trực tuyến được các nhà tuyển dụng ưa chuộng do tốc độ lan truyền thông tin nhanh và giá cả tiết kiệm tối đa. Nhà tuyển dụng có thể đưa thông tin của họ đến với nhiều ứng viên hơn thông qua Internet.

Tìm việc trực tuyến giúp ích rất nhiều cho nhà tuyển dụng và đặc biệt là người tìm việc. Vì vậy, chúng tôi đã chọn đề tài “Thiết kế xây dựng nền tảng đăng và tìm việc làm”. Mục đích đề tài này là tạo một ứng dụng web để:

- Trở thành kênh thông tin điện tử hiện đại, dễ tiếp cận với những thông tin đa dạng, trung thực, kịp thời về các tin tuyển dụng, việc làm tại thị trường Việt Nam.

- Giúp các tổ chức, cá nhân, nhà đầu tư trong và ngoài nước có nhu cầu giới thiệu việc làm, tuyển dụng hoặc xin việc có thể dễ dàng tìm, đăng, tuyển dụng hoặc đăng thông tin việc làm và tra cứu thông tin ở tất cả các lĩnh vực để lựa chọn và thực hiện quyết định đúng đắn cho mình.

- Giúp nhà tuyển dụng, trung tâm giới thiệu việc làm và doanh nghiệp kinh doanh có thêm lựa chọn các công cụ tuyển dụng hiện đại, nhanh chóng đưa việc làm đến với số lượng lớn khách hàng, giảm chi phí, tạo kết nối, chia sẻ thông tin, trao đổi với khách hàng, phát huy tính chuyên nghiệp và nâng cao năng lực cạnh tranh của doanh nghiệp.

- Giúp cơ quan quản lý nhà nước nắm bắt kịp thời thông tin về thị trường lao động để ban hành cơ chế, chính sách phù hợp, hoàn thiện hệ thống pháp luật, tạo hành lang pháp lý nhằm nâng cao tính minh bạch, hiệu quả, từng bước hoàn thiện cơ cấu thị trường lao động, góp phần đảm bảo sự phát triển bền vững. phát triển thị trường lao động Việt Nam

## 1.2 Mục tiêu và phạm vi đề tài

Xu hướng tìm kiếm việc làm qua Internet đã phổ biến ở nhiều nước từ nhiều năm nay. Tại Việt Nam, các trang web giới thiệu việc làm mới đang nở rộ trong thời gian gần đây. Không chỉ các công ty tư nhân, cơ quan quản lý nhà nước về thị trường tuyển dụng cũng rất quan tâm đến mô hình này, không chỉ các công ty tư nhân kinh doanh và cung cấp dịch vụ này. Bộ Giáo dục và Đào tạo cho biết, trong vòng 5 năm tới, nhiều tỉnh, thành phố ở Việt Nam sẽ cấp giấy chứng nhận quyền đăng ký dịch vụ tuyển dụng nhân sự. Như vậy, tính hợp pháp của công ty và nhà tuyển dụng sẽ hiện lên đầy đủ chỉ sau vài cú click chuột. Rõ ràng, đây là cơ hội rất lớn cho các trang tìm việc làm và khả năng bùng nổ việc làm trên Internet sắp bước vào thời kỳ phát triển mới.

Tuy nhiên, hiện nay hầu hết các trang web được lập ra để quảng cáo các công ty mà các nhà đầu tư tự thành lập để quảng cáo sản phẩm, thương hiệu của chính mình. Vì vậy, tính cạnh tranh thấp, thiếu minh bạch ảnh hưởng rất lớn đến quyền lợi của đa số ứng viên có nhu cầu tìm hiểu thông tin việc làm.

Trong đồ án này, chúng tôi đã thiết kế và xây dựng một phần mềm ứng dụng web kết nối nhà tuyển dụng và ứng viên cho phép tạo và tìm kiếm việc làm.

- Giới thiệu thông tin về đơn vị tuyển dụng, từng công ty, doanh nghiệp.
- Thông tin về công việc mới, công việc được quan tâm nhất.
- Thiết lập chức năng tìm kiếm và thống kê.
- Tính năng này cho phép khách hàng kết nối trực tuyến giữa những người tìm việc và các nhà tuyển dụng.
- Chức năng liên hệ với quản trị viên, cô vân khách hàng

## 1.3 Định hướng giải pháp

Để giải quyết các mục tiêu được nêu trong phần 1.2, giải pháp đề xuất cho ứng dụng tìm kiếm việc làm tuân theo trình tự sau:

(i) Cách tiếp cận ban đầu liên quan đến việc tận dụng công nghệ Spring Boot MVC và Reactjs để phát triển ứng dụng. Spring Boot MVC cung cấp backend framework mạnh mẽ và hiệu quả, trong khi Reactjs cung cấp giao diện người dùng tương tác, đảm bảo trải nghiệm người dùng liền mạch.

(ii) Đề tài của chúng tôi bao gồm việc tạo ra một nền tảng dựa trên web thân thiện với người dùng, nơi người tìm việc có thể tìm kiếm cơ hội việc làm dựa trên nhiều tiêu chí khác nhau như vị trí, ngành và mức độ kinh nghiệm. Ngoài ra, nhà tuyển dụng có thể đăng danh sách việc làm và quản lý hồ sơ nộp của ứng viên, hợp

lý hóa quy trình tuyển dụng. Ứng dụng cũng sẽ tích hợp xác thực người dùng và quyền truy cập dựa trên vai trò để đảm bảo an toàn và quyền riêng tư dữ liệu.

(iii) Đóng góp chính của đề tài này nằm ở việc tích hợp thành công Spring Boot MVC và Reactjs để tạo ra một ứng dụng tìm kiếm việc làm giàu tính năng. Thành tựu quan trọng của ứng dụng là khả năng cung cấp trải nghiệm tìm kiếm việc làm nhanh chóng và hiệu quả cho người dùng, giảm thời gian và công sức cần thiết để tìm cơ hội việc làm phù hợp. Hơn nữa, giao diện liền mạch và thời gian phản hồi nhanh của hệ thống sẽ nâng cao sự hài lòng và tương tác của người dùng.

Bằng cách kết hợp các điểm mạnh của Spring Boot MVC và Reactjs, giải pháp được đề xuất nhằm mục đích cung cấp một nền tảng tìm kiếm việc làm toàn diện và lấy người dùng làm trung tâm, cung cấp hỗ trợ có giá trị cho cả người tìm việc và nhà tuyển dụng.

### 1.4 BỘ CỤC ĐỒ ÁN

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

- Chương 2: KHẢO SÁT VÀ PHÂN TÍCH YÊU CẦU - Chương này tập trung thu thập và tìm hiểu nhu cầu, mong đợi của các bên liên quan đến đồ án. Nó phác thảo phương pháp thu thập các yêu cầu và phân tích dữ liệu thu thập được để xác định các thành phần chính của đồ án.
- Chương 3: CÔNG NGHỆ SỬ DỤNG - Chương này giới thiệu các công nghệ được lựa chọn cho đồ án, giải thích tính phù hợp và ưu điểm của chúng đối với giải pháp đề xuất.
- Chương 4: THIẾT KẾ, TRIỂN KHAI VÀ ĐÁNH GIÁ HỆ THỐNG - Chương này trình bày cách triển khai thực tế của giải pháp, bao gồm kiến trúc hệ thống, thiết kế cơ sở dữ liệu và giao diện người dùng. Nó bao gồm các thử nghiệm và đánh giá để xác nhận chức năng và hiệu suất của hệ thống.
- Chương 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN - Chương cuối cùng này tóm tắt những phát hiện và đóng góp chính của đồ án, thảo luận về những hạn chế và đưa ra những nhận xét mang tính kết luận về kết quả chung của đồ án. Nó cũng có thể bao gồm các đề xuất cho công việc hoặc cải tiến trong tương lai.

## CHƯƠNG 2. KHẢO SÁT VÀ PHÂN TÍCH YÊU CẦU

### 2.1 Khảo sát hiện trạng

Là một phần của quá trình phát triển nền tảng tìm kiếm việc làm, chúng tôi đã tiến hành khảo sát hiện trạng để đánh giá trạng thái hiện tại của phần mềm và các tính năng của phần mềm. Ngoài ra, chúng tôi đã tìm kiếm phản hồi từ người dùng để hiểu sở thích và thói quen của họ. Báo cáo này phác thảo những phát hiện chính và các tính năng phần mềm quan trọng cần được phát triển thêm. Kết quả khảo sát hiện trạng:

- **Cải thiện giao diện người dùng:** Thiết kế giao diện người dùng hiện tại được cho là có chức năng nhưng thiếu bối cảnh trực quan và hiện đại. Những người tham gia đề xuất triển khai giao diện sạch hơn, thân thiện với người dùng hơn với điều hướng trực quan để cải thiện trải nghiệm tổng thể của người dùng.
- **Lọc danh sách công việc:** Người dùng bày tỏ nhu cầu về các tùy chọn lọc nâng cao, chẳng hạn như vị trí, loại công việc, mức lương và mức kinh nghiệm, để thu hẹp tìm kiếm việc làm của họ và tìm các cơ hội liên quan hiệu quả hơn.
- **Đề xuất công việc được cá nhân hóa:** Nhiều người dùng mong muốn một hệ thống đề xuất công việc được cá nhân hóa có thể đề xuất danh sách công việc phù hợp dựa trên kỹ năng, kinh nghiệm và sở thích của họ, giúp họ khám phá những cơ hội mà họ có thể đã bỏ lỡ.
- **Hồ sơ người dùng:** Người tham gia yêu cầu khả năng tạo hồ sơ người dùng toàn diện với sơ yếu lý lịch và kỹ năng được cập nhật, giúp nhà tuyển dụng dễ dàng tìm được ứng viên phù hợp hơn.
- **Hệ thống theo dõi đơn đăng ký:** Người tìm việc muốn có một hệ thống theo dõi đơn đăng ký tích hợp cho phép họ theo dõi các công việc họ đã nộp đơn, trạng thái đơn đăng ký của họ và thông tin liên lạc với nhà tuyển dụng.

Dựa trên kết quả khảo sát hiện trạng, chúng tôi nhận thấy tầm quan trọng của việc cải thiện giao diện người dùng, giới thiệu tính năng lọc công việc nâng cao và đề xuất được cá nhân hóa cũng như triển khai hệ thống theo dõi ứng dụng. Bằng cách giải quyết các tính năng chính này, chúng tôi mong muốn làm cho ứng dụng tìm kiếm việc làm của mình lấy người dùng làm trung tâm hơn và thành công hơn trong việc giúp người tìm việc tìm thấy cơ hội việc làm lý tưởng của họ.

### 2.2 Tổng quan chức năng

Đối với chức năng nghiệp vụ:

## CHƯƠNG 2. KHẢO SÁT VÀ PHÂN TÍCH YÊU CẦU

---

- Cho phép người dùng đăng ký làm ứng viên.
- Lưu trữ thông tin của tất cả các đối tượng.
- Cho phép nhiều kiểu người dùng khác nhau chỉnh sửa thông tin cá nhân khi cần thiết.
- Tạo điều kiện trao đổi thông tin lẫn nhau giữa các đối tượng.
- Hỗ trợ quản trị viên trong việc quản lý hệ thống bao gồm: quản lý thông tin của tất cả các đối tượng (ứng viên, nhà tuyển dụng, người sử dụng quyền quản trị )
- Quyền thêm hoặc xóa các đối tượng này; quản lý thông tin về lĩnh vực, ngành nghề, việc làm và các thông tin liên quan.
- Cho phép ứng viên tìm kiếm việc làm theo các tiêu chí khác nhau.
- Cho phép ứng viên tạo hồ sơ tìm việc.
- Cho phép ứng viên thông báo cho nhà tuyển dụng về công việc.
- Cho phép nhà tuyển dụng tìm kiếm hồ sơ ứng viên phù hợp với công ty mình.
- Cho phép nhà tuyển dụng quản lý thông tin ứng viên, xem hồ sơ ứng viên và liên hệ với ứng viên qua email hoặc điện thoại.

Đối với chức năng hệ thống:

- Cho phép người dùng đăng nhập vào hệ thống với các vai trò khác nhau cho từng đối tượng.
- Đảm bảo tính bảo mật thông tin người dùng.
- Phân cấp quyền cho từng đối tượng người dùng trong việc tùy chỉnh quyền truy cập vào tài nguyên hệ thống.

Đối với lợi ích người dùng:

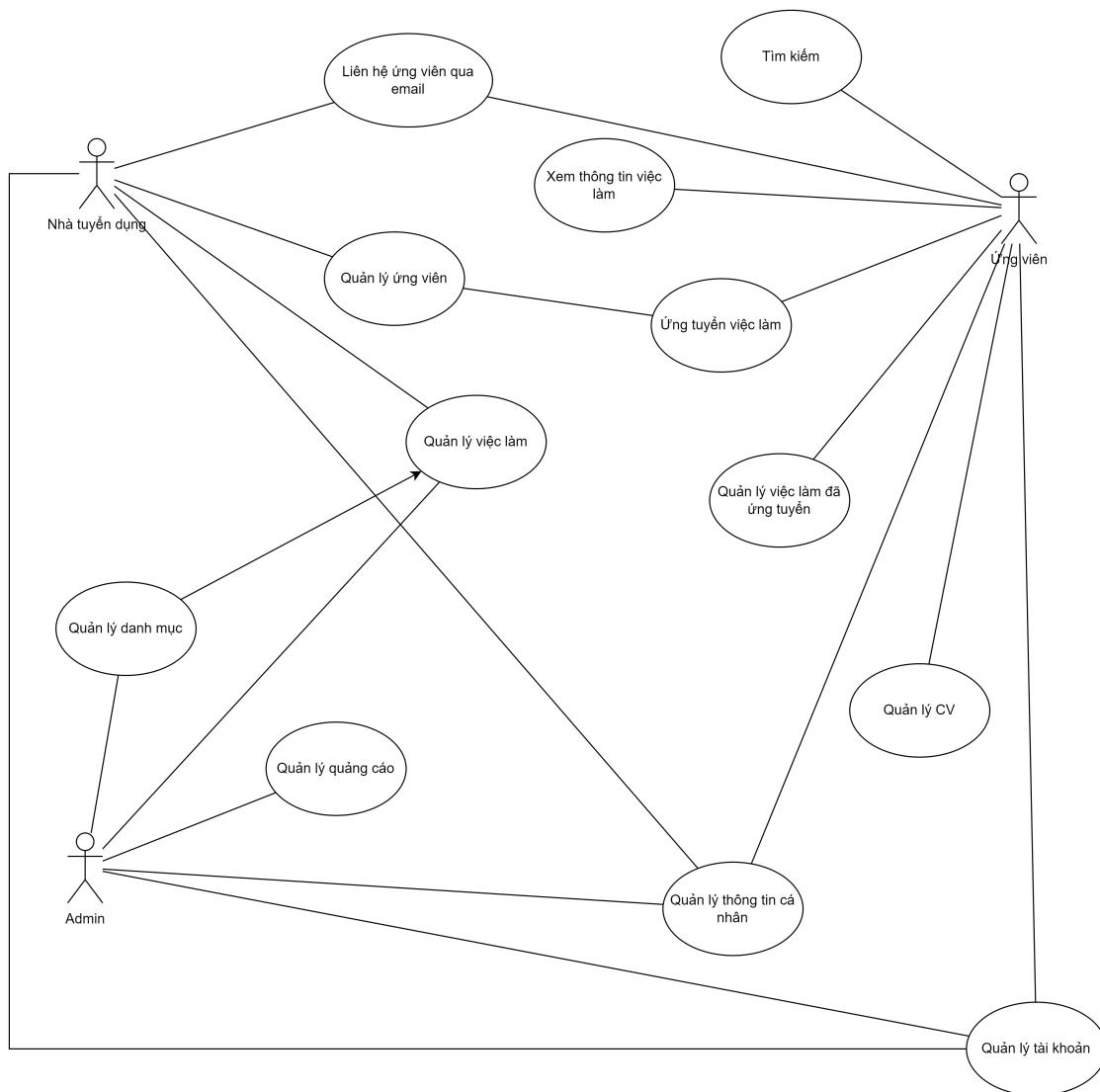
- Tính sử dụng: giao diện thân thiện, dễ sử dụng.
- Tính hiệu quả: đảm bảo truy cập nhanh vào các trang và kiểm soát lỗi tốt.
- Khả năng tương thích: Hỗ trợ tốt trên các trình duyệt phổ biến như GoogleChrome, Microsoft Edge, Cốc Cốc,...
- Tính ổn định: hệ thống phải hoạt động liên tục 24 giờ trong ngày, 7 ngày trong tuần, thời gian ngừng hoạt động không quá 5 phút.
- Hỗ trợ nhiều người dùng truy cập cùng lúc.
- Phản thông báo kết quả đăng ký cho ứng viên trong thời gian nhanh nhất.

Đối với việc phát triển dự án:

- Được thiết kế theo mô hình ba tầng(three-tier model).
- Tính mô đun
- Các thành phần của hệ thống được thiết kế động, dễ điều chỉnh.
- Khả năng tái sử dụng mã.

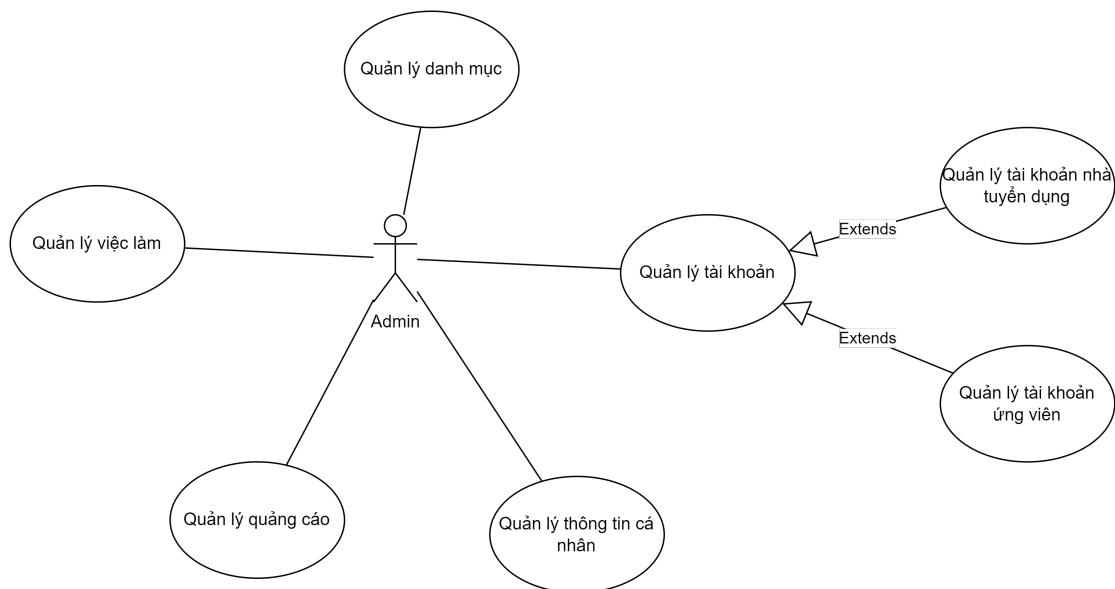
### 2.2.1 Biểu đồ use case tổng quát

Mô hình chức năng hệ thống:



Hình 2.1: Biểu đồ use case tổng quát

### 2.2.2 Biểu đồ use case phân rã của quản trị viên hệ thống

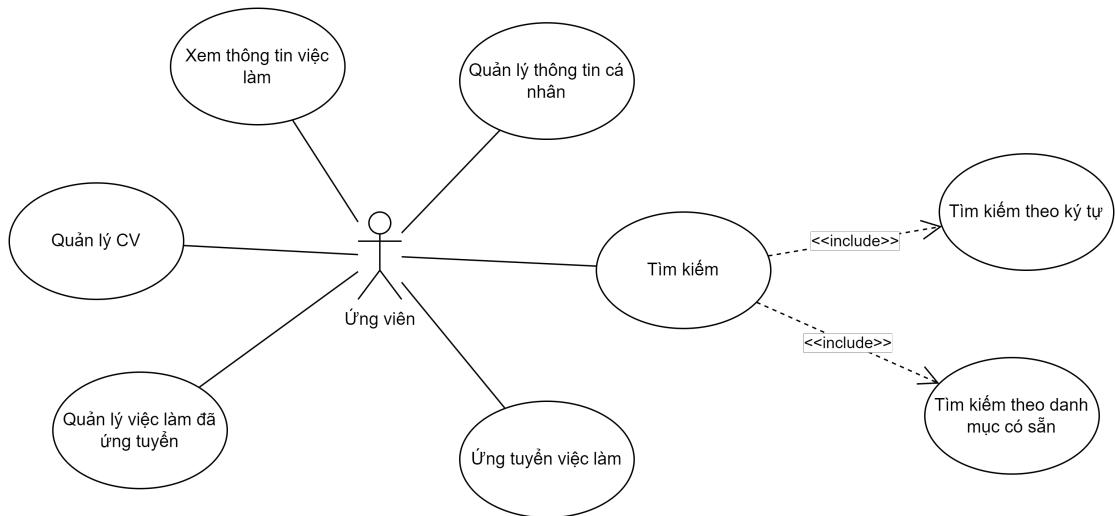


**Hình 2.2:** Biểu đồ use case phân rã của quản trị viên hệ thống

| Tác nhân | Tên use case   |
|----------|--|
| Admin    | Đăng nhập  |
|          | Cập nhật thông tin                                     |
|          | Quản lý danh mục                                       |
|          | Quản lý quảng cáo                                      |
|          | Quản lý việc làm được đăng                             |
|          | Quản lý tài khoản người dùng(ứng viên, nhà tuyển dụng) |

**Bảng 2.1:** Use case của quản trị viên hệ thống

### 2.2.3 Biểu đồ use case phân rã của ứng viên

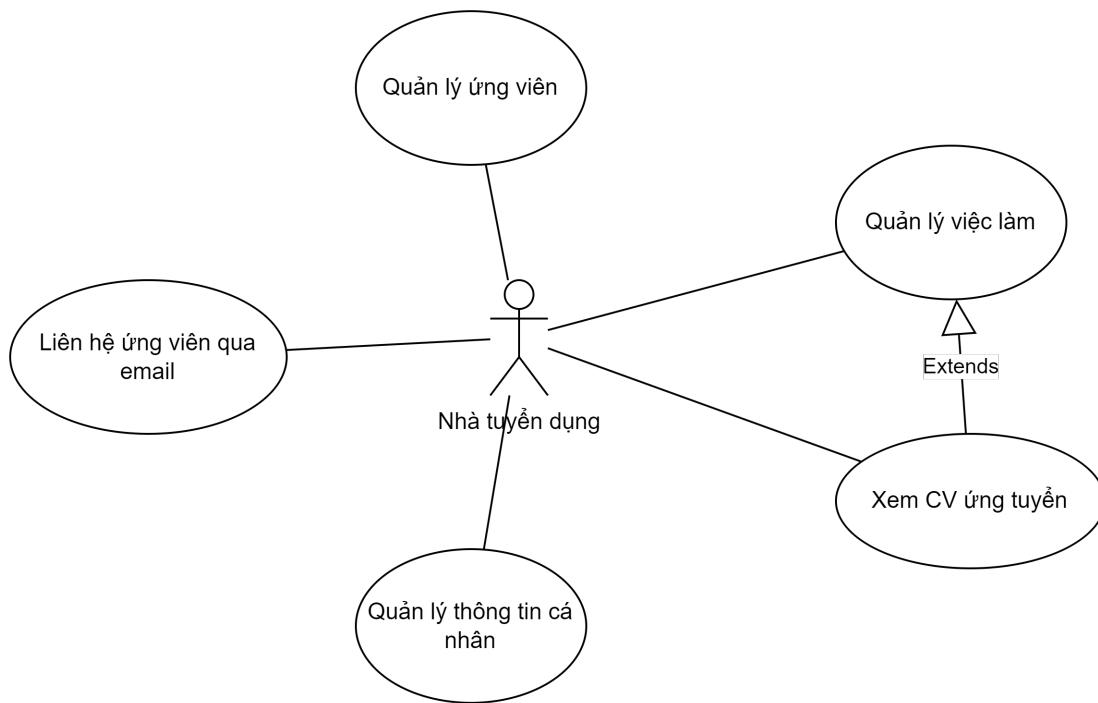


**Hình 2.3:** Biểu đồ use case phân rã của ứng viên

| Tác nhân | Tên use case                   |
|----------|--------------------------------|
| Úng viên | Đăng nhập, Đăng ký             |
|          | Tim kiếm                       |
|          | Xem thông tin việc làm         |
|          | Ung tuyển việc làm             |
|          | Cập nhật thông tin cá nhân     |
|          | Quản lý công việc đã ứng tuyển |
|          | Quản lý CV                     |

**Bảng 2.2:** Use case của ứng viên

#### 2.2.4 Biểu đồ use case phân rã của nhà tuyển dụng



**Hình 2.4:** Biểu đồ use case phân rã của nhà tuyển dụng

| Tác nhân | Tên use case               |
|----------|----------------------------|
| Ứng viên | Đăng nhập, Đăng ký         |
|          | Quản lý ứng viên           |
|          | Quản lý việc làm           |
|          | Cập nhật thông tin cá nhân |
|          | Liên hệ ứng viên qua email |

**Bảng 2.3:** Use case của nhà tuyển dụng

### 2.3 Đặc tả chức năng

#### 2.3.1 Đặc tả use case Đăng nhập

Hình 2.5 và hình 2.6 lần lượt là biểu đồ hoạt động và biểu đồ tuần tự của usecase đăng nhập trong đó:

**Tác nhân:** Admin, ứng viên, nhà tuyển dụng

**Mô tả:** Mô tả cách người dùng đăng nhập vào hệ thống.

**Tiền điều kiện:** Người dùng truy cập vào hệ thống với vai trò của mình.

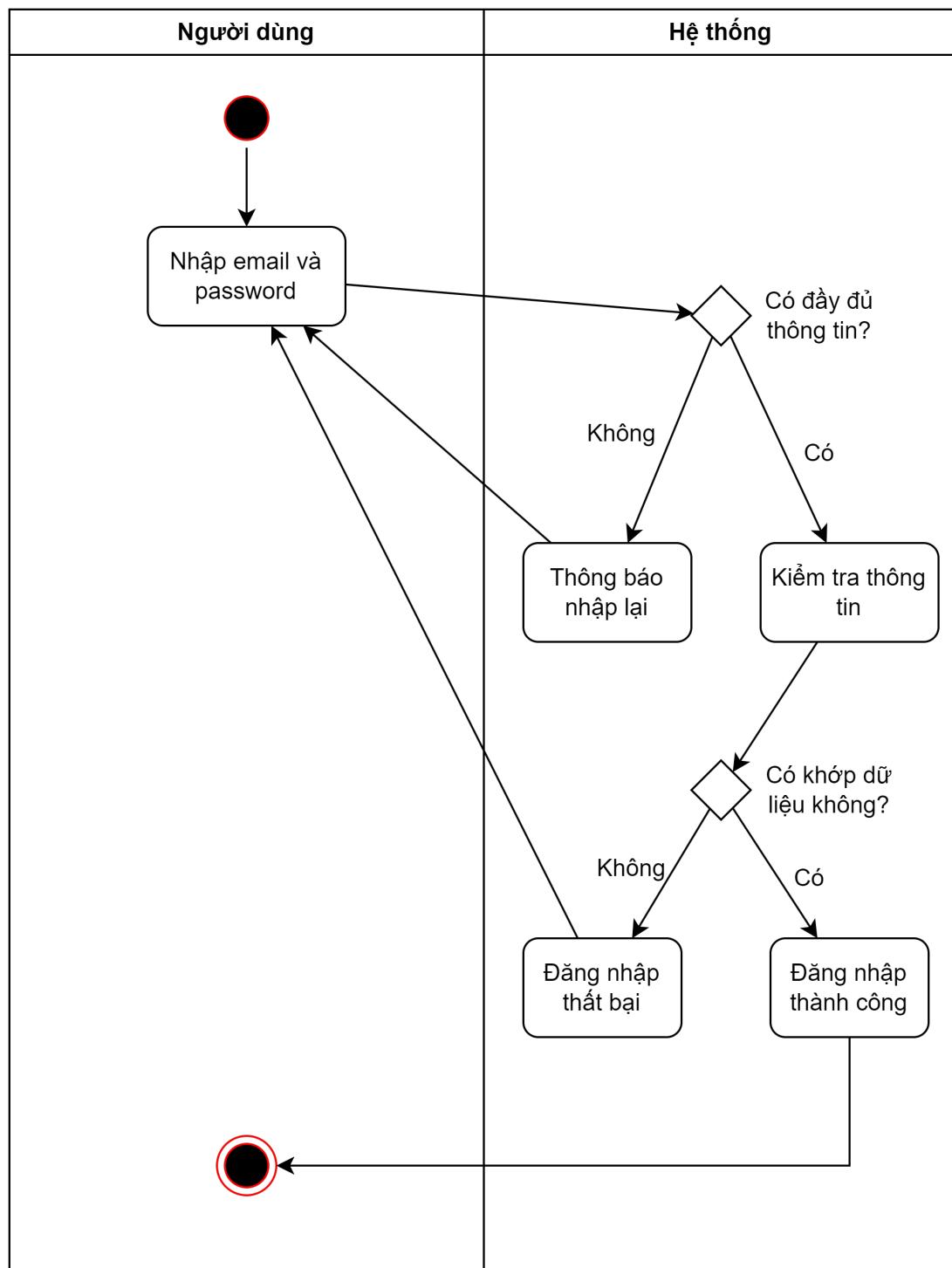
**Luồng hoạt động chính:**

1. Người dùng nhập email và mật khẩu vào hệ thống.
2. Hệ thống kiểm tra xem email và mật khẩu có tồn tại trong hệ thống hay không.  
Nếu có thì hệ thống thông báo thao tác đăng nhập đã thành công.

**Luồng hoạt động phát sinh:**

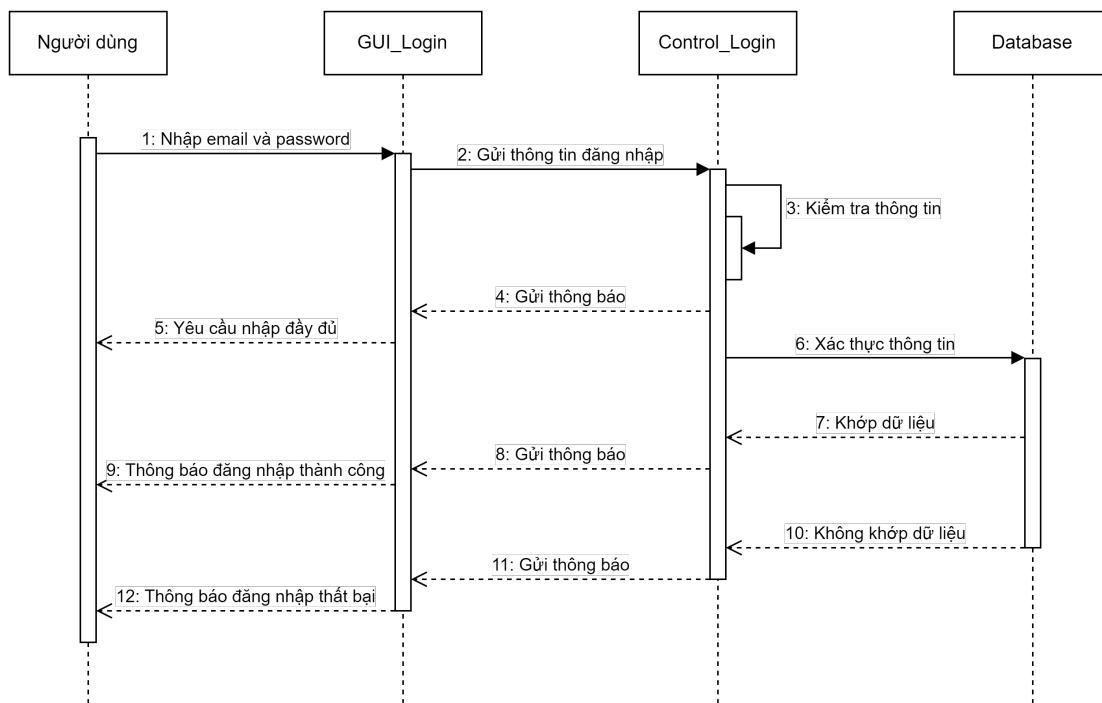
1. Nếu người dùng không nhập đủ email và mật khẩu, hệ thống yêu cầu người dùng nhập đầy đủ các thông tin trên.
2. Nếu người dùng có email và mật khẩu đã nhập không tồn tại trong hệ thống, hệ thống sẽ yêu cầu nhập lại.

**Hậu điều kiện:** Người dùng đăng nhập thành công vào hệ thống.



**Hình 2.5:** Biểu đồ hoạt động của usecase Đăng nhập

## CHƯƠNG 2. KHẢO SÁT VÀ PHÂN TÍCH YÊU CẦU



Hình 2.6: Biểu đồ tuần tự của usecase Đăng nhập

### 2.3.2 Đặc tả use case Đăng ký

Hình 2.7 và hình 2.8 lần lượt là biểu đồ hoạt động và biểu đồ tuần tự của usecase đăng ký trong đó:

**Tác nhân:** Ứng viên, nhà tuyển dụng

**Mô tả:** Mô tả cách người dùng đăng ký vào hệ thống.

**Tiền điều kiện:** Người dùng truy cập vào hệ thống với vai trò của mình.

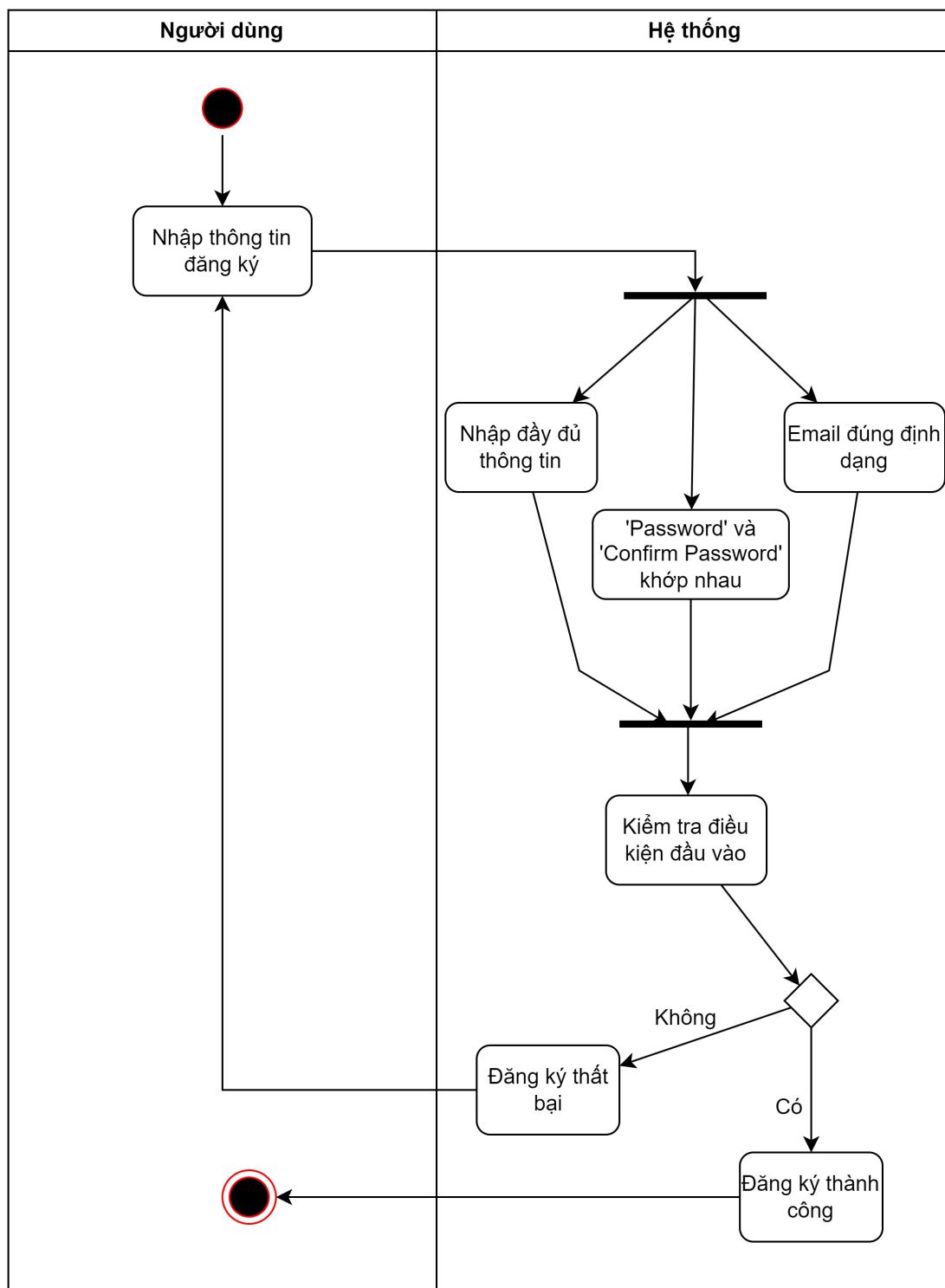
**Luồng hoạt động chính:**

1. Ứng viên và nhà tuyển dụng yêu cầu đăng ký tài khoản. Trang đăng ký sẽ mở ra. Người dùng điền một số thông tin sau: họ tên, email, mật khẩu, xác nhận mật khẩu,...
2. Nếu người dùng yêu cầu đăng ký, hệ thống sẽ kiểm tra xem mẫu đăng ký có đáp ứng các thông tin sau hay không:
  - Phải nhập đầy đủ các thông tin được yêu cầu (thông tin có dấu \*).
  - Thông tin trong 2 ô: Mật khẩu và Xác nhận mật khẩu phải khớp nhau.
  - Thông tin email phải chính xác theo mẫu: user@domain.xxx và không được trùng khớp với các tài khoản hiện có.
3. Nếu nhập thông tin đáp ứng yêu cầu trên, hệ thống sẽ tiến hành đăng ký. Thông tin của người dùng được lưu vào cơ sở dữ liệu.

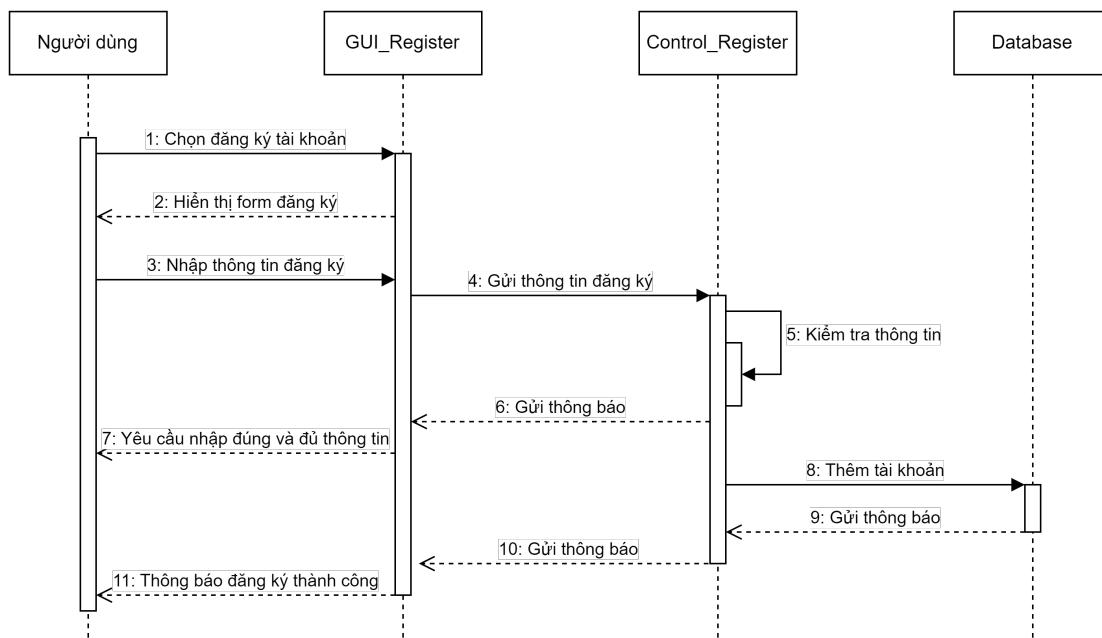
**Luồng hoạt động phát sinh:**

1. Nếu người dùng nhập thông tin không đầy đủ sẽ hiển thị thông báo lỗi "Chưa nhập đủ thông tin"
2. Nếu người dùng nhập mật khẩu khác nhau vào trường Mật khẩu và Xác nhận mật khẩu, thông báo lỗi "Mật khẩu và Xác nhận mật khẩu không khớp" sẽ được hiển thị.

**Hậu điều kiện:** Người dùng đăng ký tài khoản thành công



**Hình 2.7:** Biểu đồ hoạt động của usecase Đăng ký



**Hình 2.8:** Biểu đồ tuần tự của usecase Đăng ký

### 2.3.3 Đặc tả use case Tìm kiếm

Hình 2.9 và hình 2.10 lần lượt là biểu đồ hoạt động và biểu đồ tuần tự của usecase tìm kiếm trong đó:

**Tác nhân:** Người dùng

**Mô tả:** Mô tả cách người dùng tìm kiếm việc làm phù hợp.

**Tiền điều kiện:** Người dùng truy cập vào hệ thống.

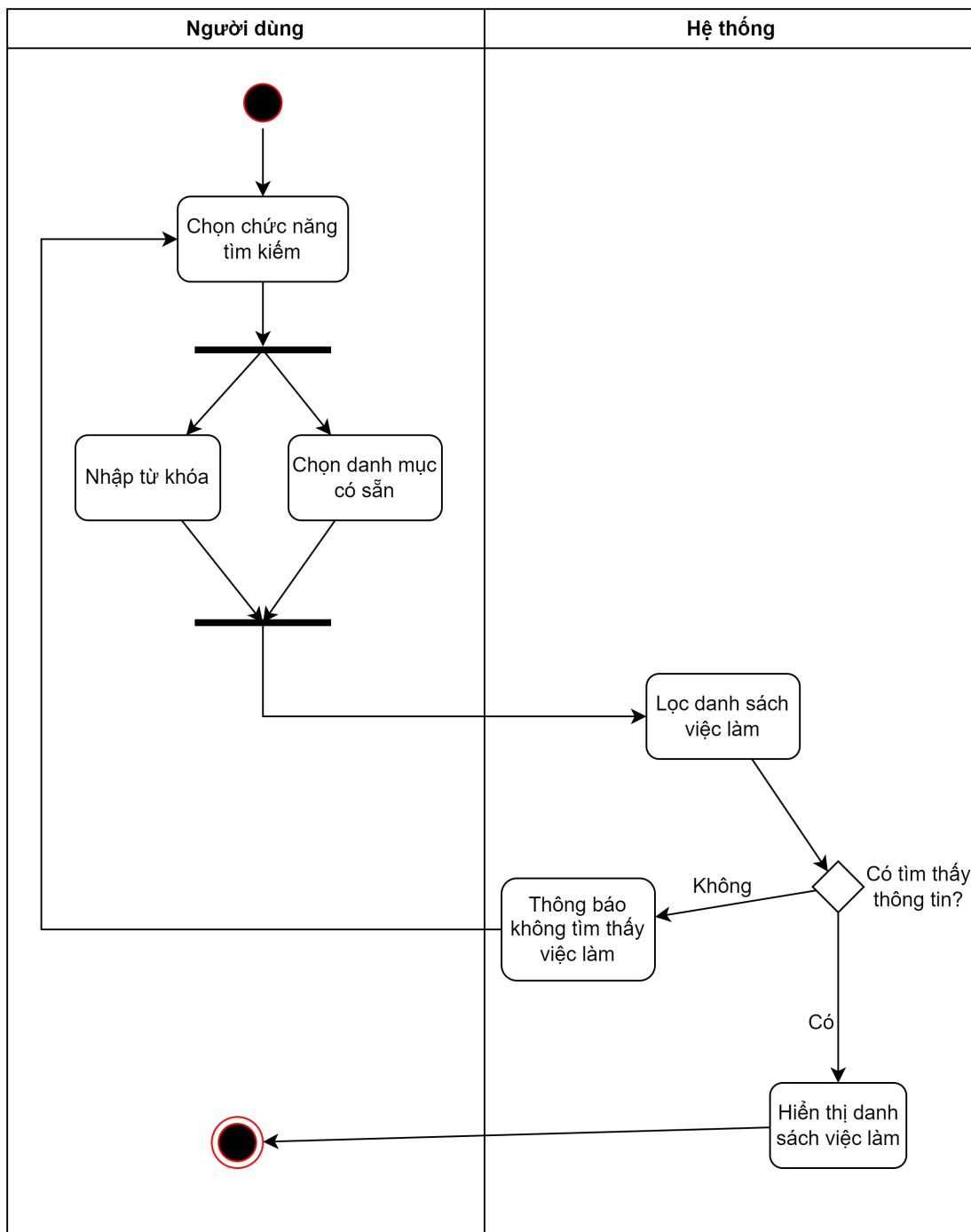
**Luồng hoạt động chính:**

- Người dùng lựa chọn tìm kiếm theo từ khóa hoặc danh mục có sẵn.
- Hệ thống hiển thị kết quả trùng khớp

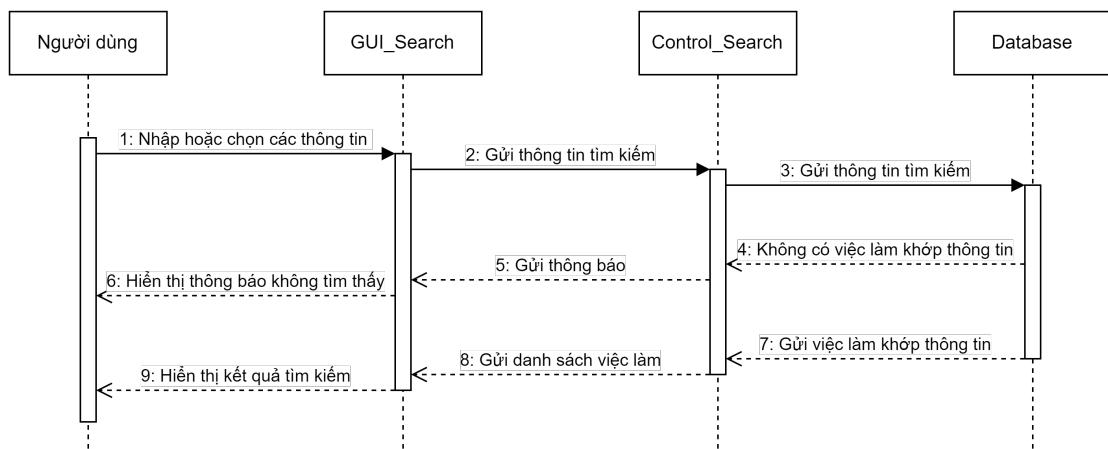
**Luồng hoạt động phát sinh:**

- Nếu không tìm thấy dữ liệu, hệ thống hiển thị thông báo không tìm thấy việc làm nào.

**Hậu điều kiện:** Người dùng thấy danh sách việc làm trùng khớp với tìm kiếm.



**Hình 2.9:** Biểu đồ hoạt động của usecase Tìm kiếm



**Hình 2.10:** Biểu đồ tuần tự của usecase Tìm kiếm

### 2.3.4 ĐẶC TẢ USE CASE XEM THÔNG TIN VIỆC LÀM

Hình 2.11 và hình 2.12 lần lượt là biểu đồ hoạt động và biểu đồ tuần tự của usecase xem thông tin việc làm trong đó:

**Tác nhân:** Người dùng

**Mô tả:** Mô tả cách người dùng xem thông tin việc làm.

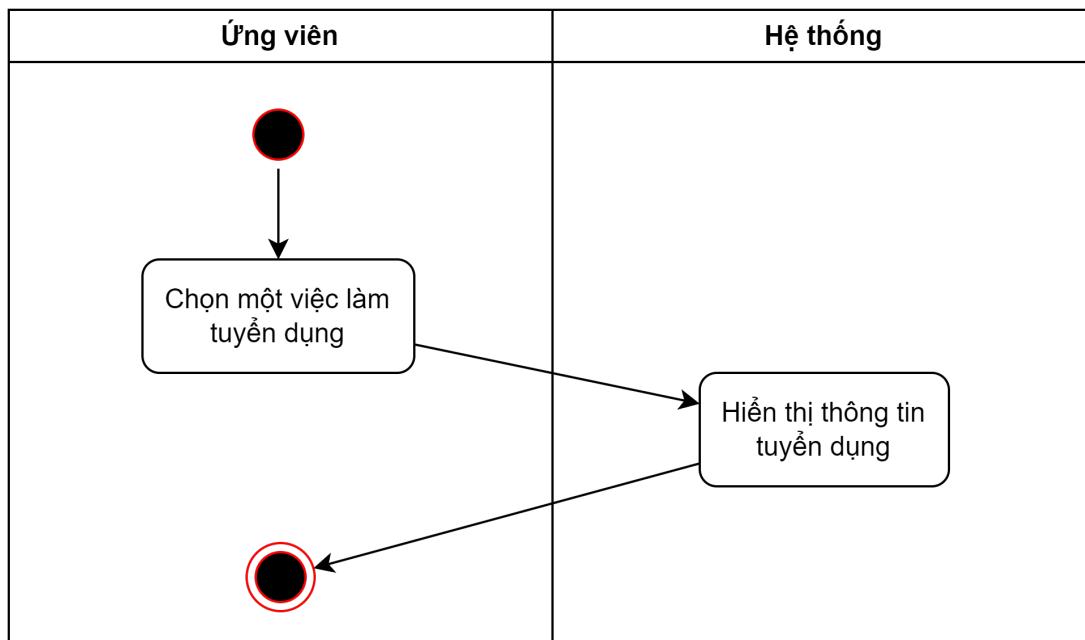
**Tiền điều kiện:** Người dùng truy cập vào hệ thống và tìm kiếm việc làm.

**Luồng hoạt động chính:**

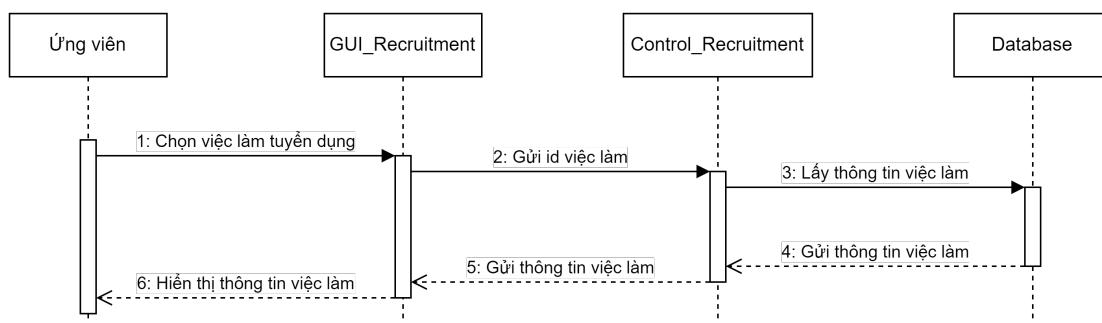
1. Người dùng chọn thông tin việc làm muốn xem.
2. Hệ thống hiển thị thông tin

**Luồng hoạt động phát sinh:** Không

**Hậu điều kiện:** Người dùng xem được thông tin việc làm



**Hình 2.11:** Biểu đồ hoạt động của usecase Xem thông tin việc làm



**Hình 2.12:** Biểu đồ tuần tự của usecase Xem thông tin việc làm

### 2.3.5 Đặc tả use case **Ứng tuyển việc làm**

Hình 2.13 và hình 2.14 lần lượt là biểu đồ hoạt động và biểu đồ tuần tự của usecase ứng tuyển việc làm trong đó:

**Tác nhân:** Ứng viên

**Mô tả:** Mô tả quá trình ứng tuyển việc làm của ứng viên

**Tiền điều kiện:** Ứng viên đăng nhập vào hệ thống.

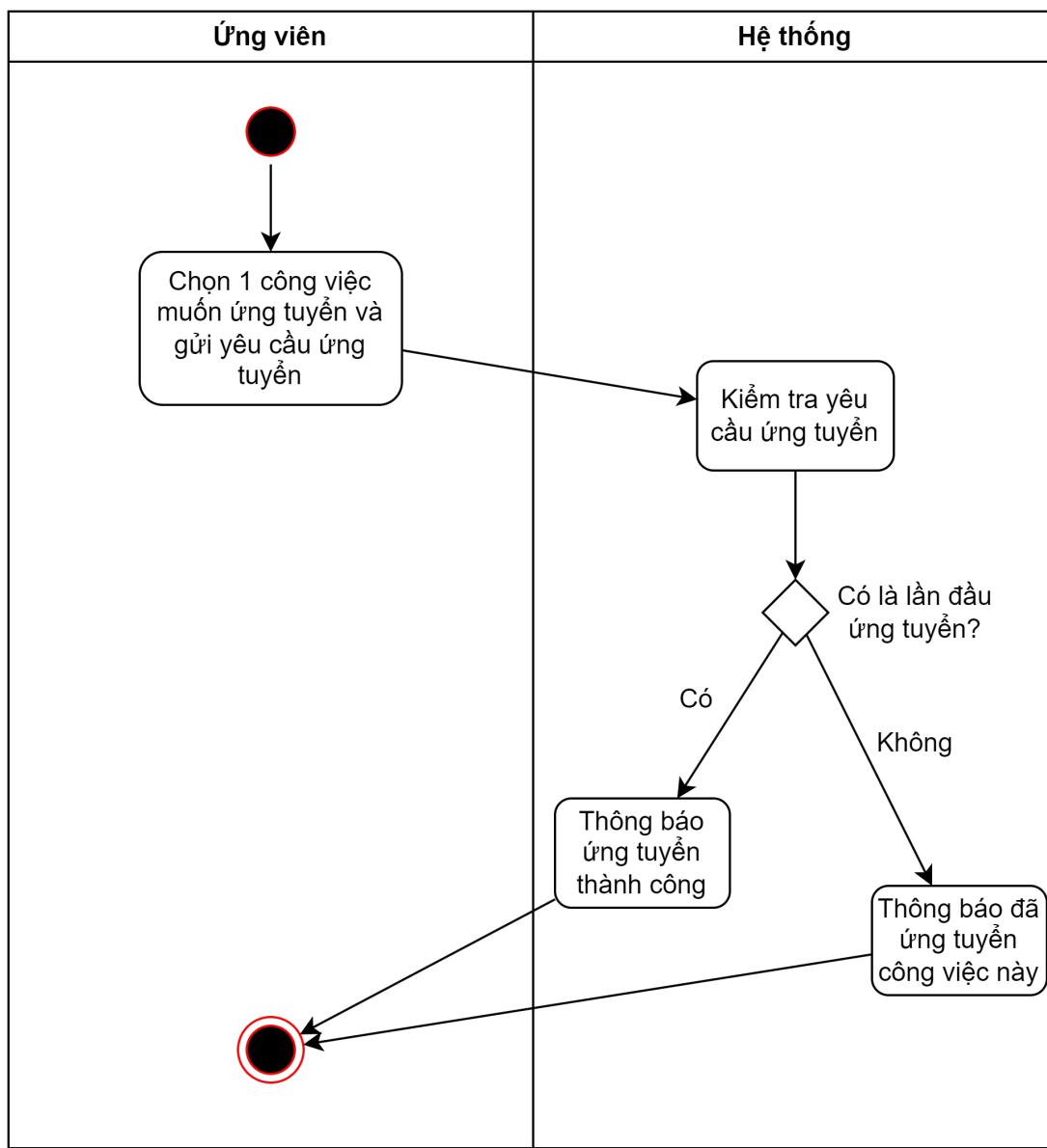
**Luồng hoạt động chính:**

1. Sau khi chọn được công việc phù hợp, ứng viên nhấp vào nút “Ứng tuyển”.
2. Thông báo thông tin hồ sơ sẽ được gửi đến nhà tuyển dụng và nhà tuyển dụng có thể xem hồ sơ.

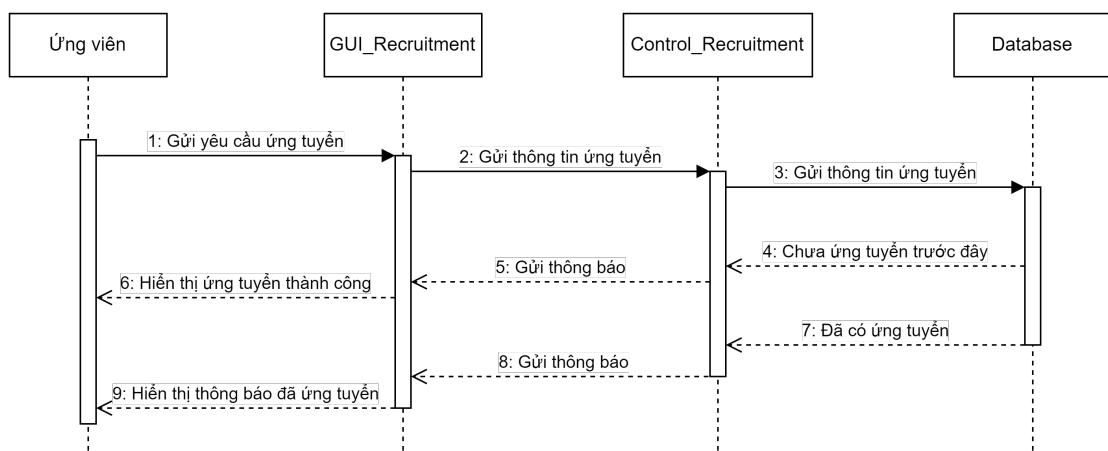
**Luồng hoạt động phát sinh:**

1. Nếu công việc đã được ứng tuyển, hệ thống sẽ thông báo đã ứng tuyển.

**Hậu điều kiện:** Ứng viên ứng tuyển thành công



**Hình 2.13:** Biểu đồ hoạt động của usecase **Ứng tuyển việc làm**



**Hình 2.14:** Biểu đồ tuần tự của usecase **Ứng tuyển việc làm**

### 2.3.6 Đặc tả use case Cập nhật thông tin cá nhân

Hình 2.15 và hình 2.16 lần lượt là biểu đồ hoạt động và biểu đồ tuần tự của usecase cập nhật thông tin cá nhân trong đó:

**Tác nhân:** Úng viên, nhà tuyển dụng

**Mô tả:** Mô tả quá trình cập nhật thông tin của người dùng

**Tiền điều kiện:** Úng viên, nhà tuyển dụng đăng nhập vào hệ thống.

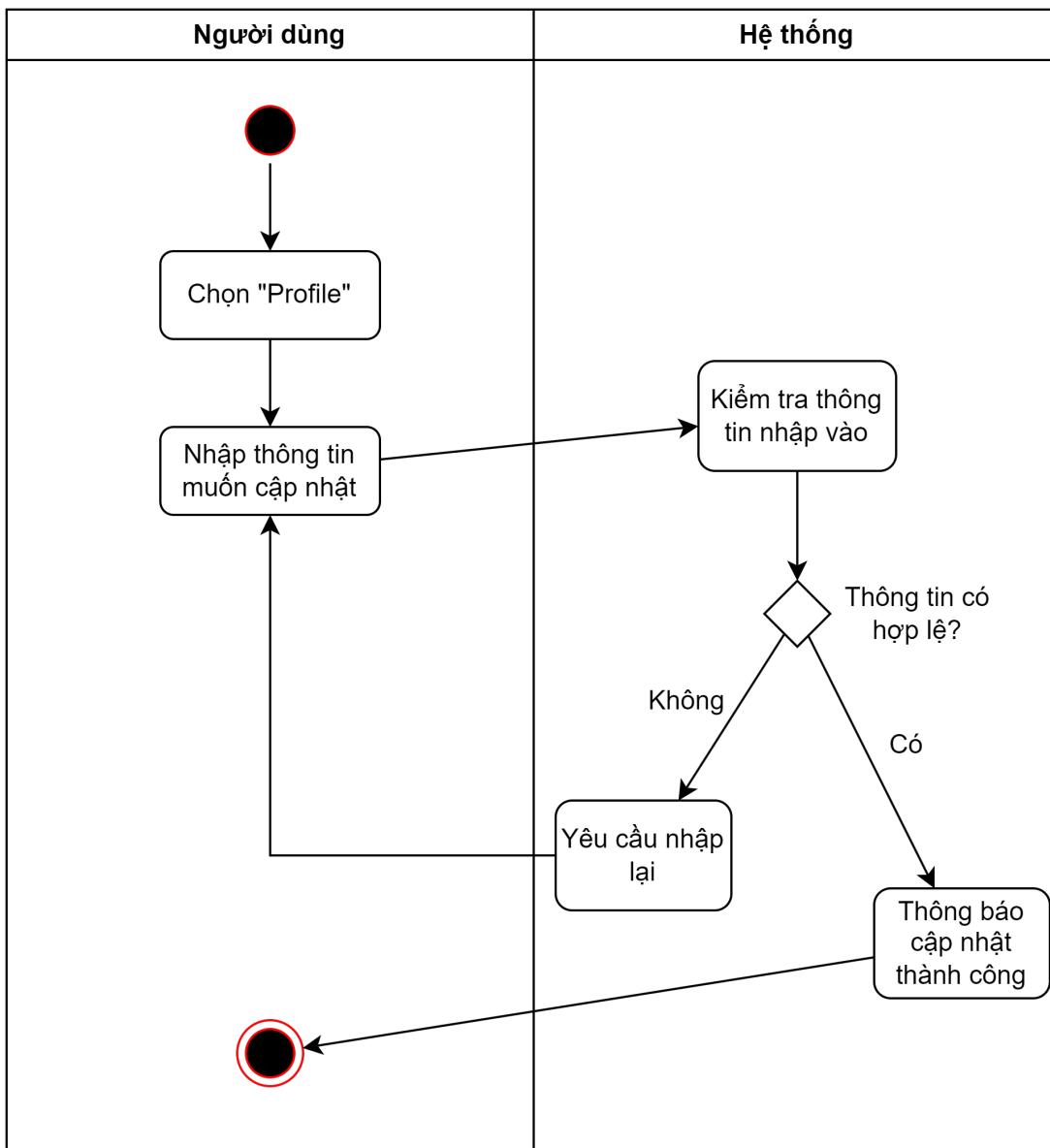
**Luồng hoạt động chính:**

1. Người dùng chọn mục "Profile".
2. Hệ thống sẽ chuyển đến trang quản lý hồ sơ.
3. Người dùng nhập thông tin và nhấn "Lưu".
4. Nếu các thông tin trên được cập nhật đầy đủ thì hệ thống thông báo thành công. Người dùng có thể xem lại hồ sơ hoàn chỉnh.

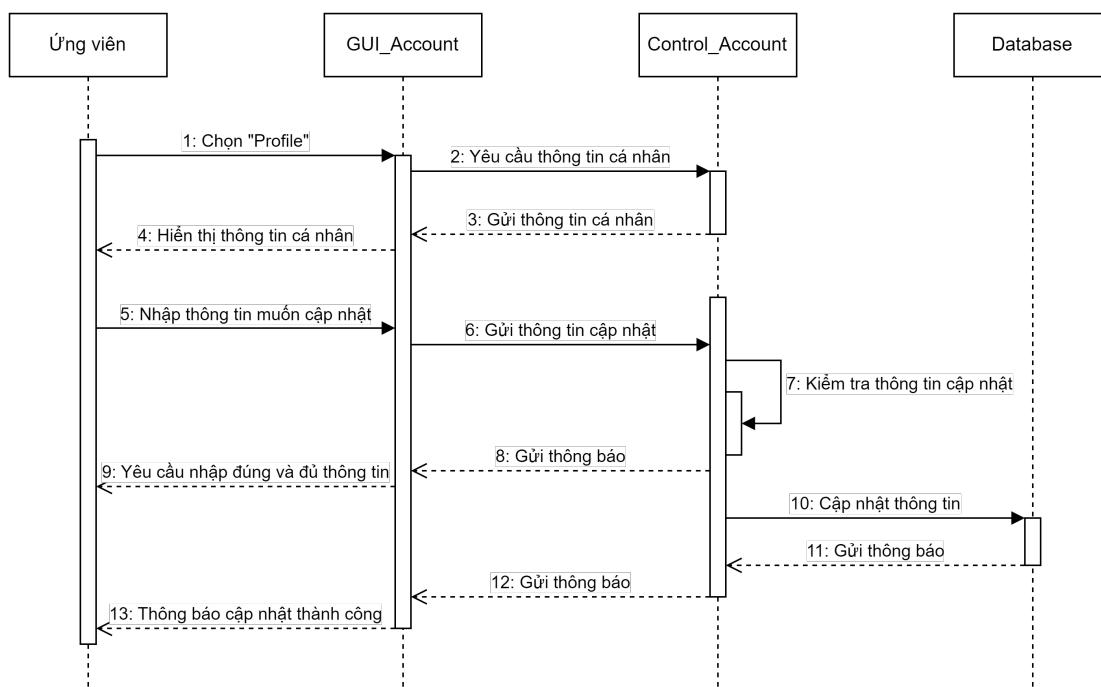
**Luồng hoạt động phát sinh:**

1. Nếu người dùng không điền các thông tin có dấu (\*), hệ thống sẽ yêu cầu người dùng nhập đầy đủ các thông tin.

**Hậu điều kiện:** Người dùng cập nhật thông tin cá nhân thành công



**Hình 2.15:** Biểu đồ hoạt động của usecase Cập nhật thông tin cá nhân



**Hình 2.16:** Biểu đồ đồ tuần tự của usecase Cập nhật thông tin cá nhân

### 2.3.7 Đặc tả use case Quản lý đơn ứng tuyển

Hình 2.17 và hình 2.18 lần lượt là biểu đồ hoạt động và biểu đồ đồ tuần tự của usecase quản lý đơn ứng tuyển trong đó:

**Tác nhân:** Úng viên

**Mô tả:** Mô tả quá trình quản lý đơn ứng tuyển của ứng viên

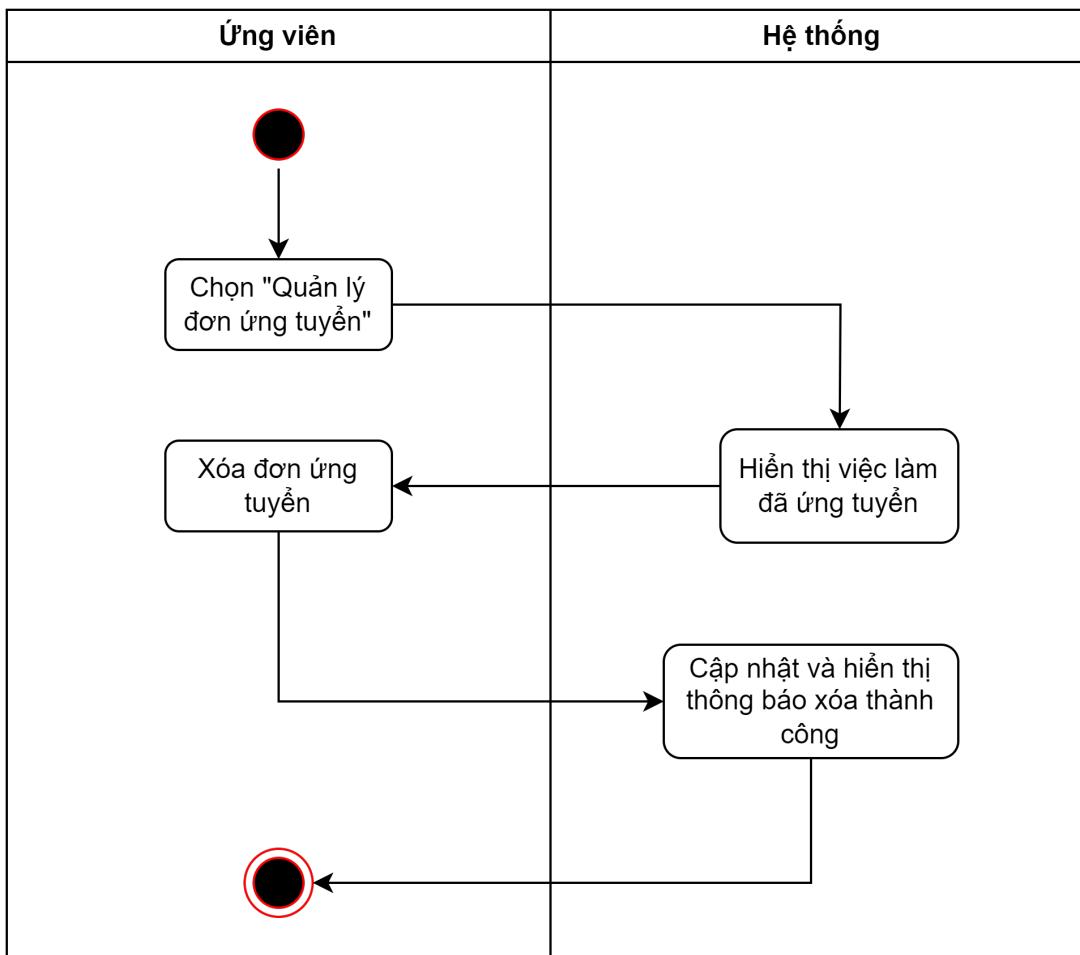
**Tiền điều kiện:** Úng viên đăng nhập vào hệ thống.

**Luồng hoạt động chính:**

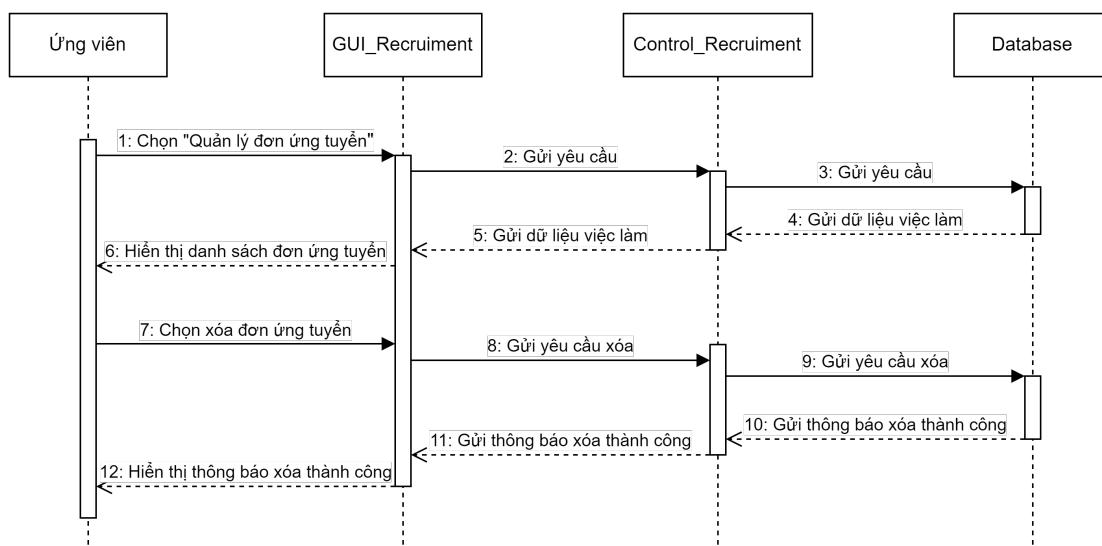
1. Úng viên chọn trang “Quản lý đơn ứng tuyển”.
2. Hệ thống chuyển trang và hiển thị các công việc đã đăng ký.
3. Úng viên có thể xem hoặc xóa công việc đã ứng tuyển nếu muốn.
4. Hệ thống cập nhật và hiển thị dữ liệu.

**Luồng hoạt động phát sinh:** Không

**Hậu điều kiện:** Úng viên xem hoặc xóa đơn đơn ứng tuyển thành công



**Hình 2.17:** Biểu đồ hoạt động của usecase Quản lý đơn ứng tuyển



**Hình 2.18:** Biểu đồ tuần tự của usecase Quản lý đơn ứng tuyển

### 2.3.8 Đặc tả use case Cập nhật CV cá nhân

Hình 2.19 và hình 2.20 lần lượt là biểu đồ hoạt động và biểu đồ tuần tự của usecase cập nhật CV cá nhân trong đó:

**Tác nhân:** Úng viên

**Mô tả:** Mô tả quá trình tạo CV cá nhân của ứng viên

**Tiền điều kiện:** Úng viên đăng nhập vào hệ thống.

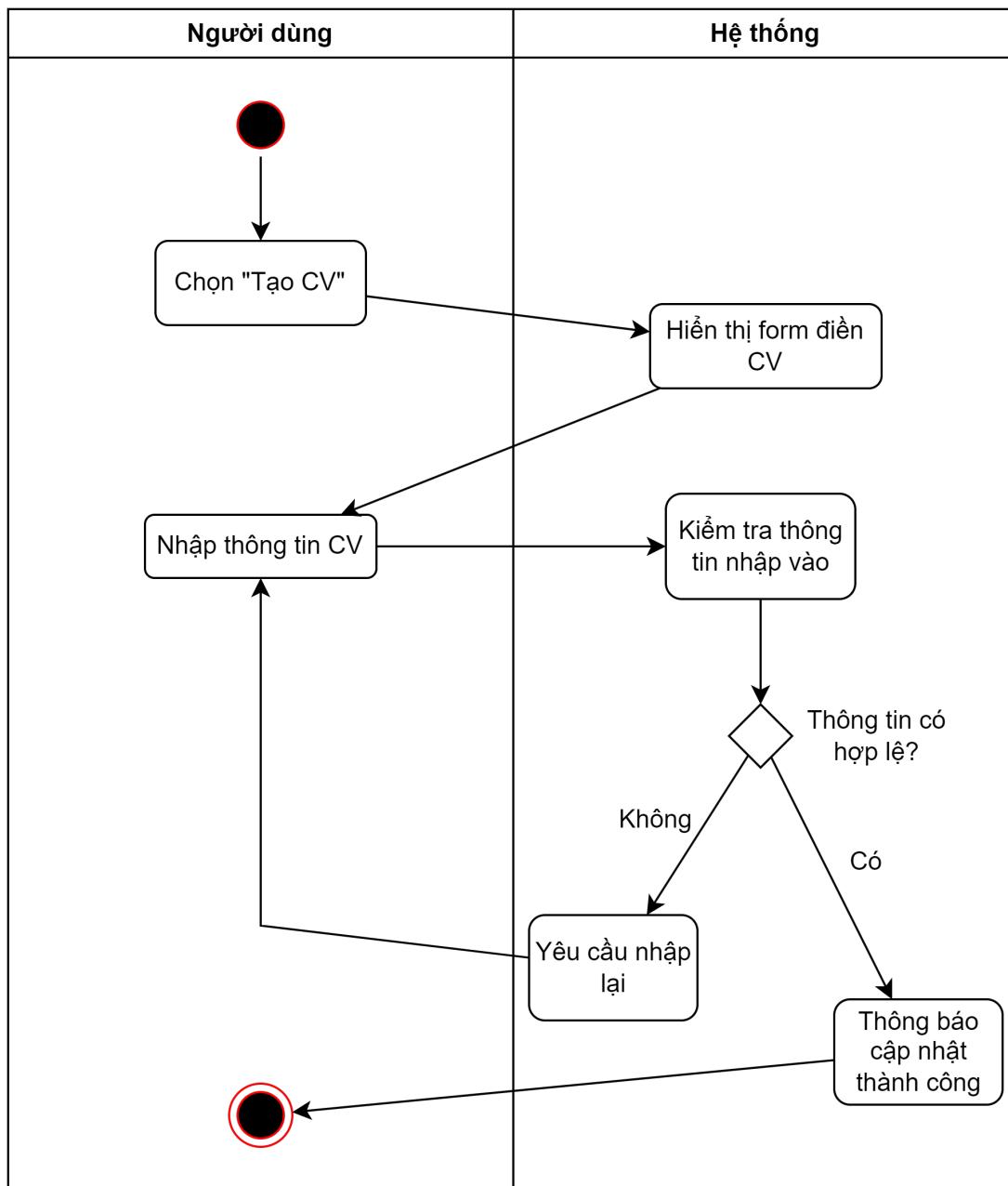
**Luồng hoạt động chính:**

1. Ứng viên chọn “Tạo CV”.
2. Hệ thống chuyển trang và hiển thị mẫu điền CV.
3. Ứng viên nhập thông tin CV đầy đủ và nhấn “Lưu”.
4. Nếu các thông tin trên được cập nhật đầy đủ thì hệ thống thông báo tạo CV thành công

**Luồng hoạt động phát sinh:**

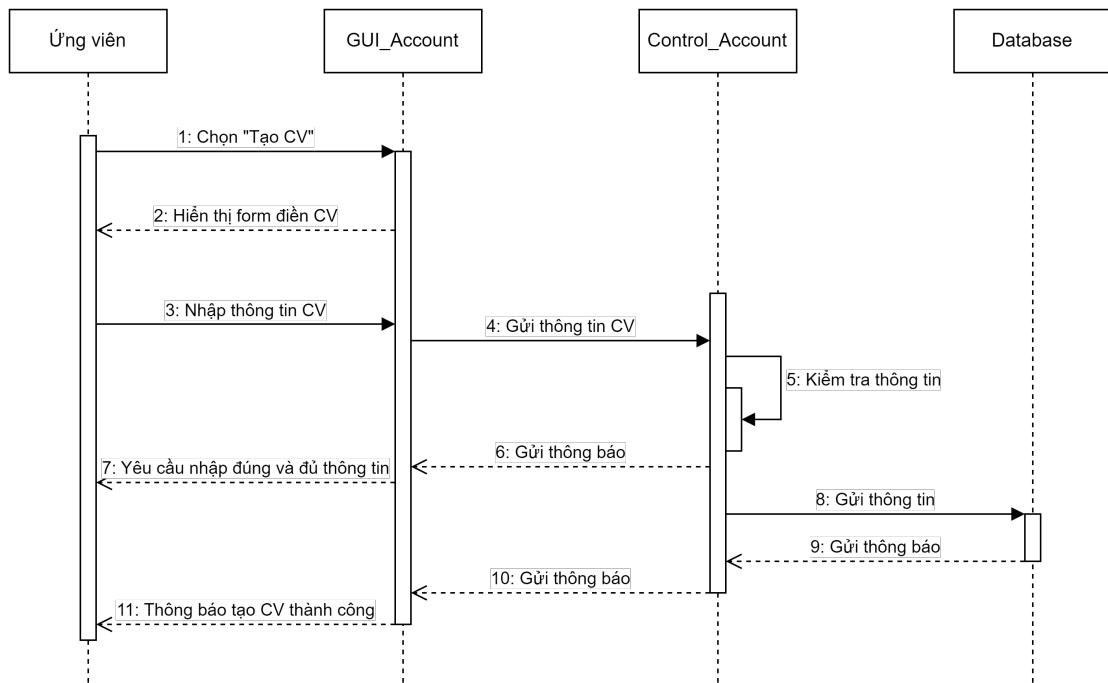
1. Nếu ứng viên không điền các thông tin có dấu (\*) hoặc không thêm ảnh, hệ thống sẽ yêu cầu ứng viên nhập đầy đủ các thông tin.

**Hậu điều kiện:** Ứng viên tạo CV thành công



**Hình 2.19:** Biểu đồ hoạt động của usecase Cập nhật CV cá nhân

## CHƯƠNG 2. KHẢO SÁT VÀ PHÂN TÍCH YÊU CẦU



**Hình 2.20:** Biểu đồ tuần tự của usecase Cập nhật CV cá nhân

### 2.3.9 Đặc tả use case Quản lý ứng tuyển của nhà tuyển dụng

Hình 2.21 và hình 2.22 lần lượt là biểu đồ hoạt động và biểu đồ tuần tự của usecase quản lý ứng tuyển của nhà tuyển dụng trong đó:

**Tác nhân:** Nhà tuyển dụng

**Mô tả:** Mô tả quá trình quản lý ứng tuyển của nhà tuyển dụng

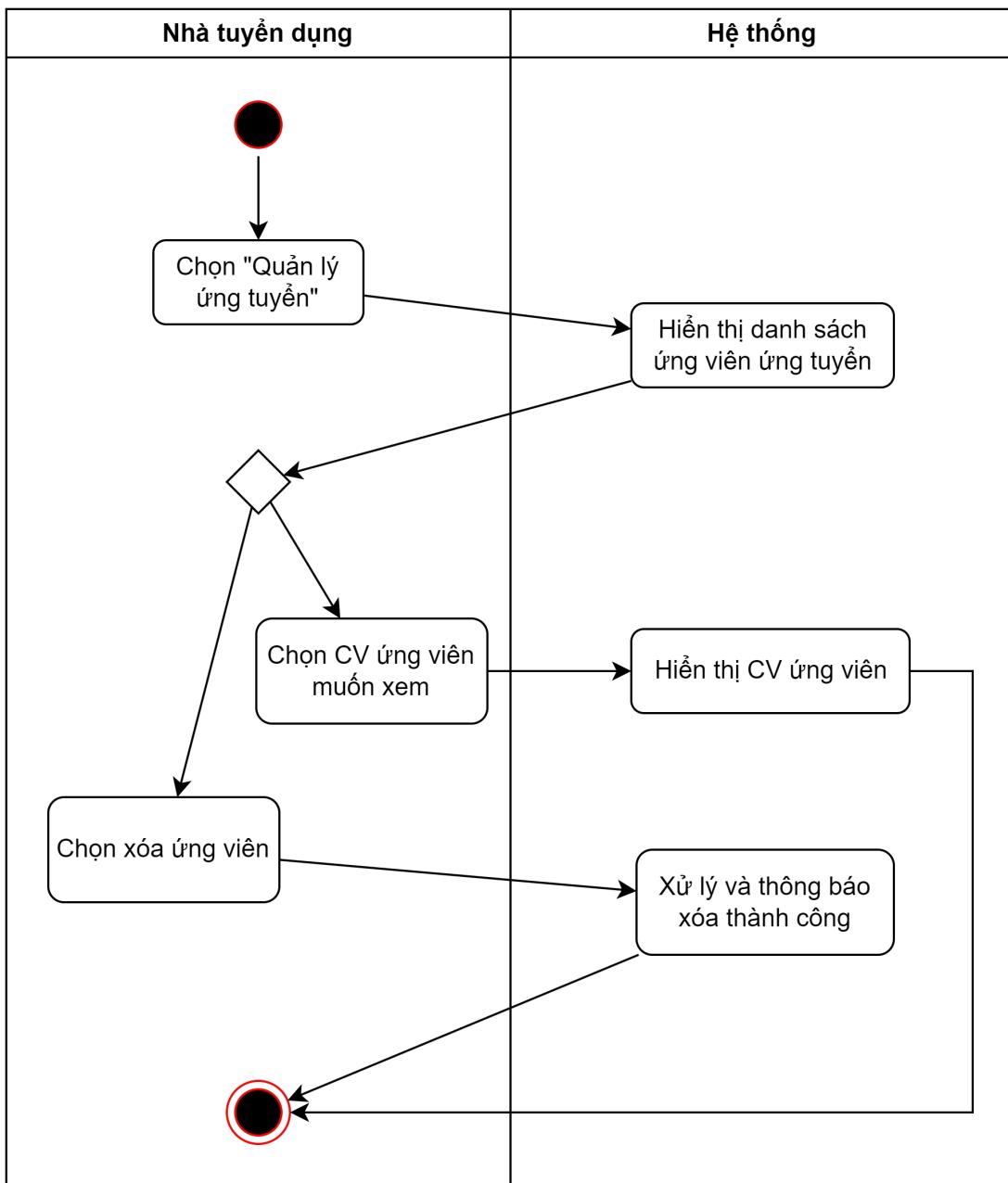
**Tiền điều kiện:** Nhà tuyển dụng đăng nhập vào hệ thống.

**Luồng hoạt động chính:**

1. Nhà tuyển dụng chọn "Quản lý đơn ứng tuyển"
2. Hệ thống hiển thị danh sách ứng viên đã nộp hồ sơ.
3. Nhà tuyển dụng lựa chọn xem CV ứng viên họ muốn.
4. Hệ thống hiển thị CV của ứng viên.
5. Nhà tuyển dụng chọn xóa ứng viên.
6. Cập nhật và hiển thị lại danh sách ứng viên.

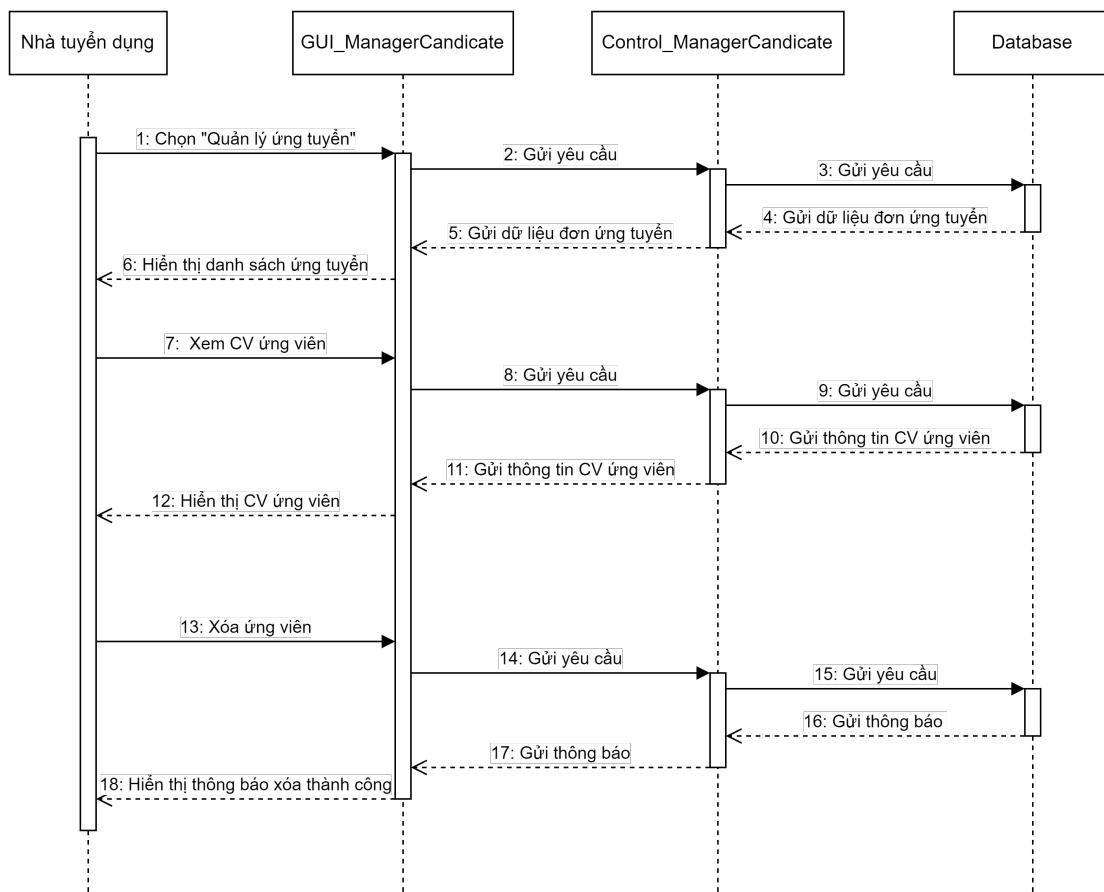
**Luồng hoạt động phát sinh:** Không

**Hậu điều kiện:** Nhà tuyển dụng xem hoặc xóa ứng viên thành công



**Hình 2.21:** Biểu đồ hoạt động của usecase Quản lý ứng tuyển của nhà tuyển dụng

## CHƯƠNG 2. KHẢO SÁT VÀ PHÂN TÍCH YÊU CẦU



**Hình 2.22:** Biểu đồ tuần tự của usecase Quản lý ứng tuyển của nhà tuyển dụng

### 2.3.10 Đặc tả use case Quản lý việc làm

Hình 2.23 và hình 2.24 lần lượt là biểu đồ hoạt động và biểu đồ tuần tự của usecase quản lý việc làm trong đó:

**Tác nhân:** Nhà tuyển dụng

**Mô tả:** Mô tả quá trình quản lý công việc của nhà tuyển dụng

**Tiền điều kiện:** Nhà tuyển dụng đăng nhập vào hệ thống.

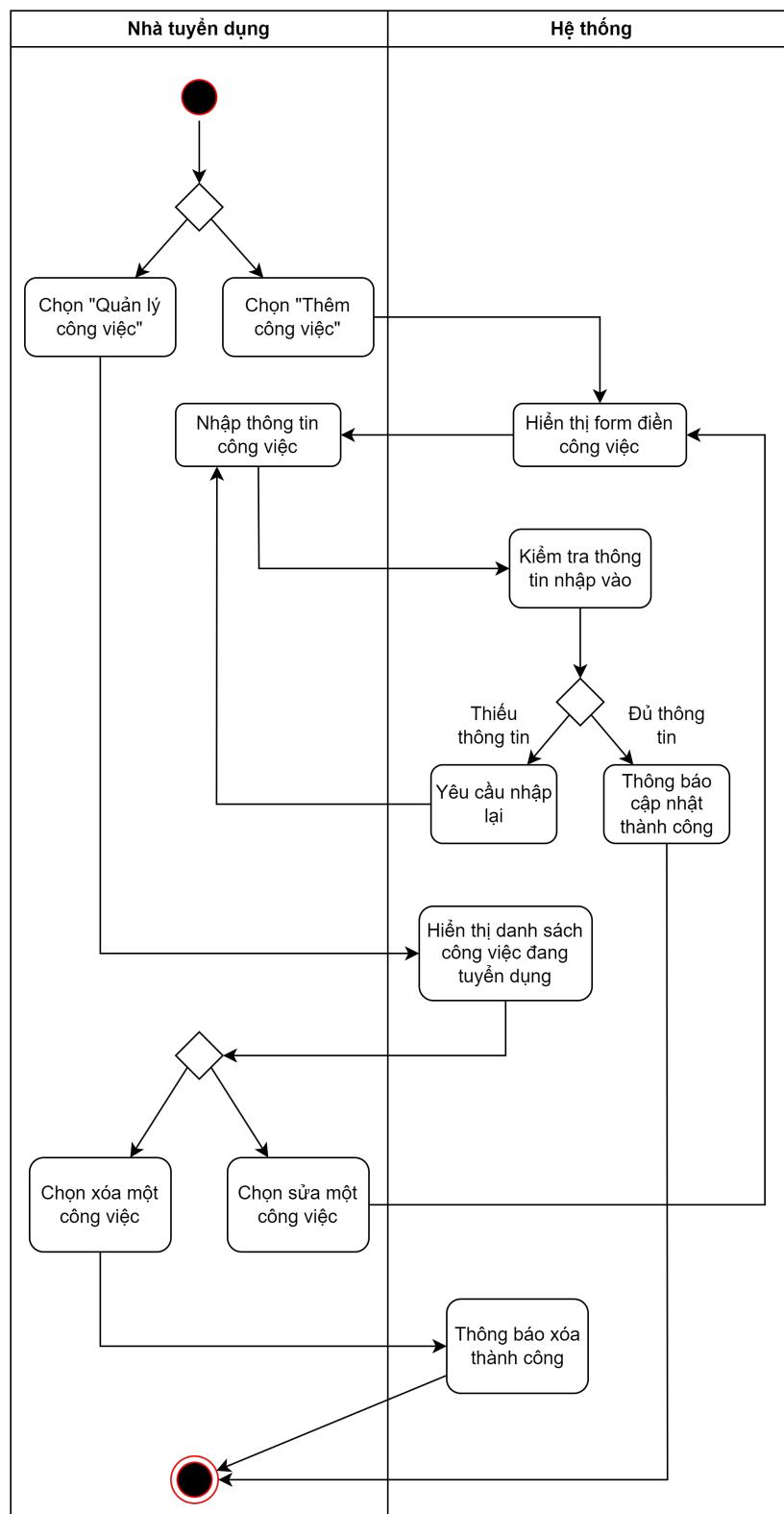
**Luồng hoạt động chính:**

1. Nhà tuyển dụng chọn “Thêm công việc”.
2. Hệ thống hiển thị trang đăng tuyển việc làm.
3. Nhà tuyển dụng nhập đầy đủ thông tin tuyển dụng
4. Hệ thống cập nhật dữ liệu và hiển thị công việc mới được thêm vào.
5. Để sửa lại tin tuyển dụng, nhà tuyển chọn công việc muốn thay đổi.
6. Hệ thống hiển thị thông tin về công việc đó.
7. Sau khi thay đổi thông tin công việc, nhà tuyển dụng chọn “Submit”.
8. Hệ thống cập nhật dữ liệu và hiển thị nội dung đã sửa.
9. Nếu nhà tuyển dụng muốn xóa tin tuyển dụng thì chọn nút “Xóa”.
10. Hệ thống thực thi và cập nhật dữ liệu.

**Luồng hoạt động phát sinh:**

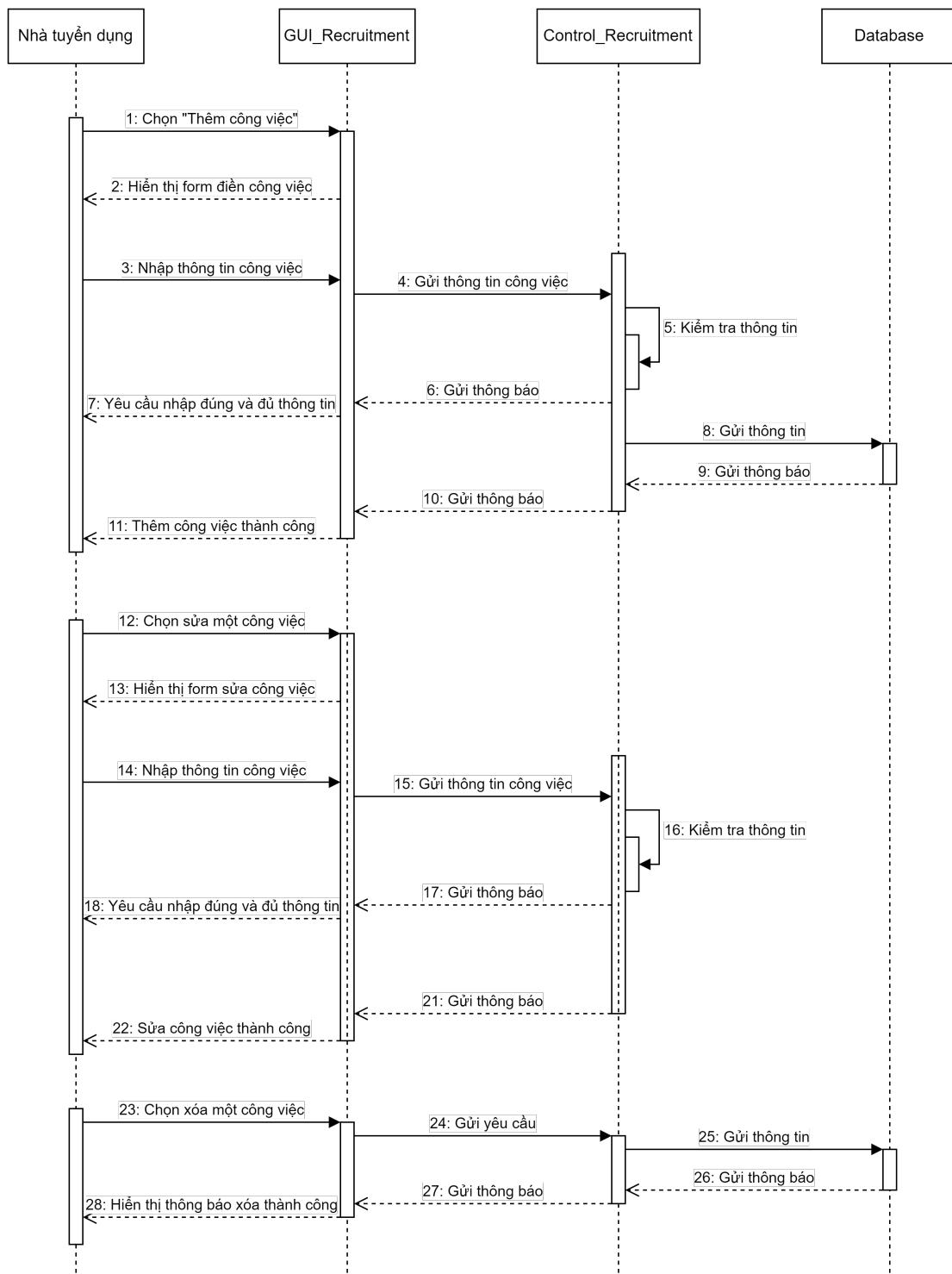
1. Nếu nhà tuyển dụng không nhập đầy đủ thông tin vào các trường có dấu (\*) thì hệ thống yêu cầu nhập đầy đủ thông tin.
2. Nhà tuyển dụng có thể quản lý tất cả các công việc đã hết hạn.

**Hậu điều kiện:** Nhà tuyển dụng thêm, sửa, xóa công việc thành công



**Hình 2.23:** Biểu đồ hoạt động của usecase Quản lý việc làm

## CHƯƠNG 2. KHẢO SÁT VÀ PHÂN TÍCH YÊU CẦU



**Hình 2.24:** Biểu đồ tuần tự của usecase Quản lý việc làm

### 2.3.11 Đặc tả use case Liên hệ ứng viên

Hình 2.25 và hình 2.26 lần lượt là biểu đồ hoạt động và biểu đồ tuần tự của usecase liên hệ ứng viên trong đó:

**Tác nhân:** Nhà tuyển dụng

**Mô tả:** Mô tả quá trình liên hệ ứng viên.

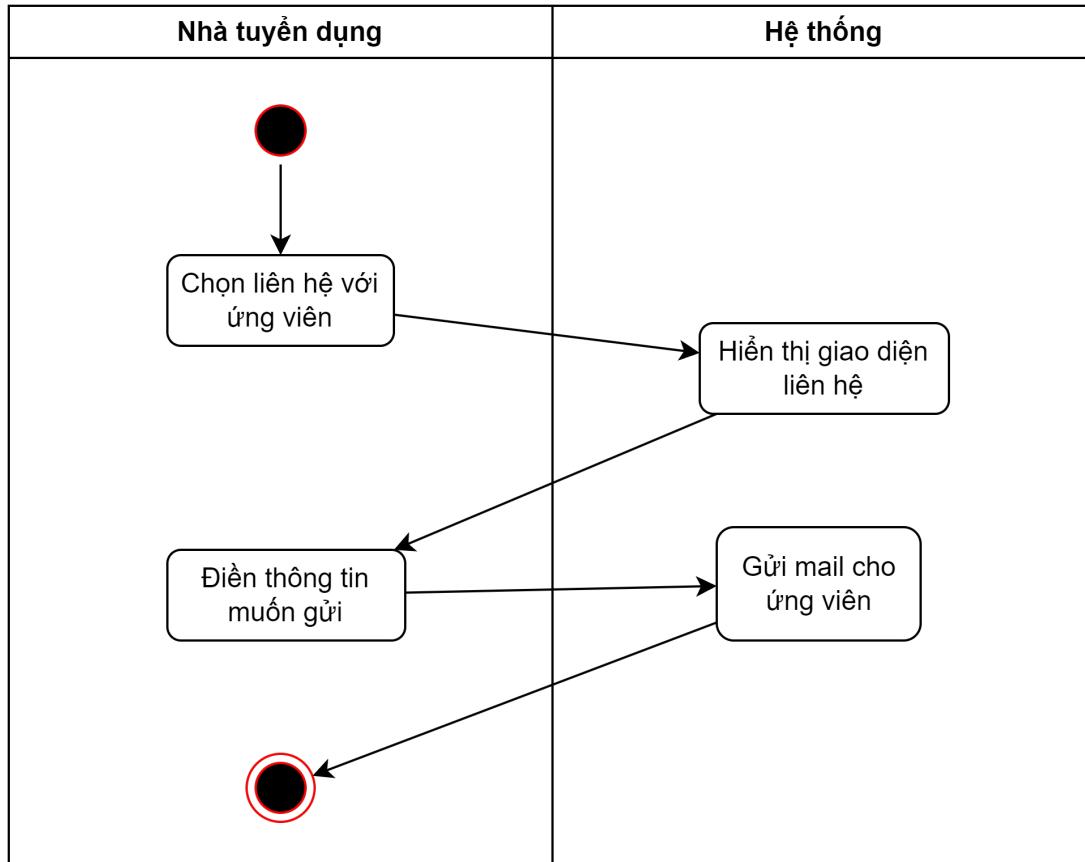
**Tiền điều kiện:** Nhà tuyển dụng đăng nhập vào hệ thống.

**Luồng hoạt động chính:**

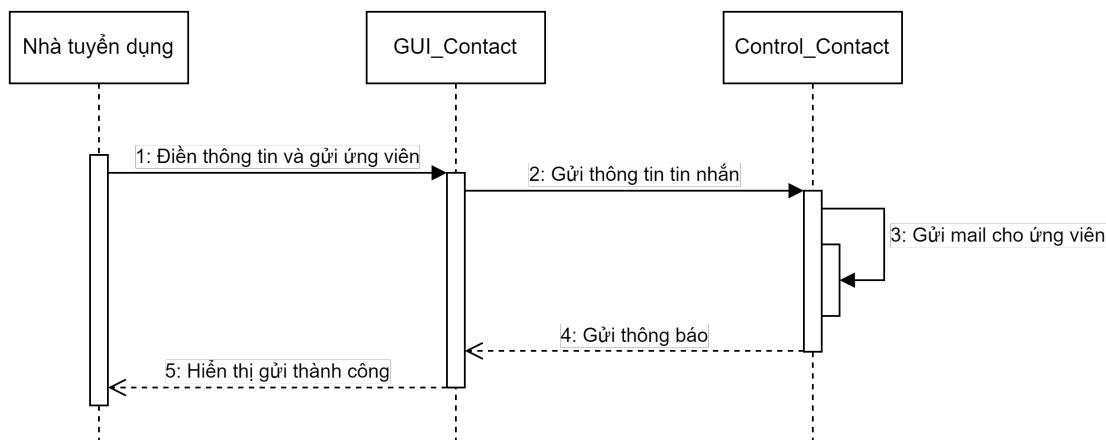
1. Nhà tuyển dụng tìm hồ sơ ứng viên phù hợp với công ty mình và nhấn vào nút “liên hệ ứng viên”.
2. Hệ thống chuyển đến trang liên hệ ứng viên.
3. Nhà tuyển dụng điền thông tin và gửi
4. Nội dung sẽ được gửi tới email của ứng viên.

**Luồng hoạt động phát sinh:** Không

**Hậu điều kiện:** Nhà tuyển gửi mail cho ứng viên thành công



**Hình 2.25:** Biểu đồ hoạt động của usecase Liên hệ ứng viên



**Hình 2.26:** Biểu đồ tuần tự của usecase Liên hệ ứng viên

### 2.3.12 Đặc tả use case Quản lý danh mục

Hình 2.27 và hình 2.28 lần lượt là biểu đồ hoạt động và biểu đồ tuần tự của usecase quản lý danh mục trong đó:

**Tác nhân:** Quản trị viên

**Mô tả:** Mô tả quá trình quản lý danh mục gồm thêm sửa xóa của quản trị viên

**Tiền điều kiện:** Quản trị viên đăng nhập vào hệ thống.

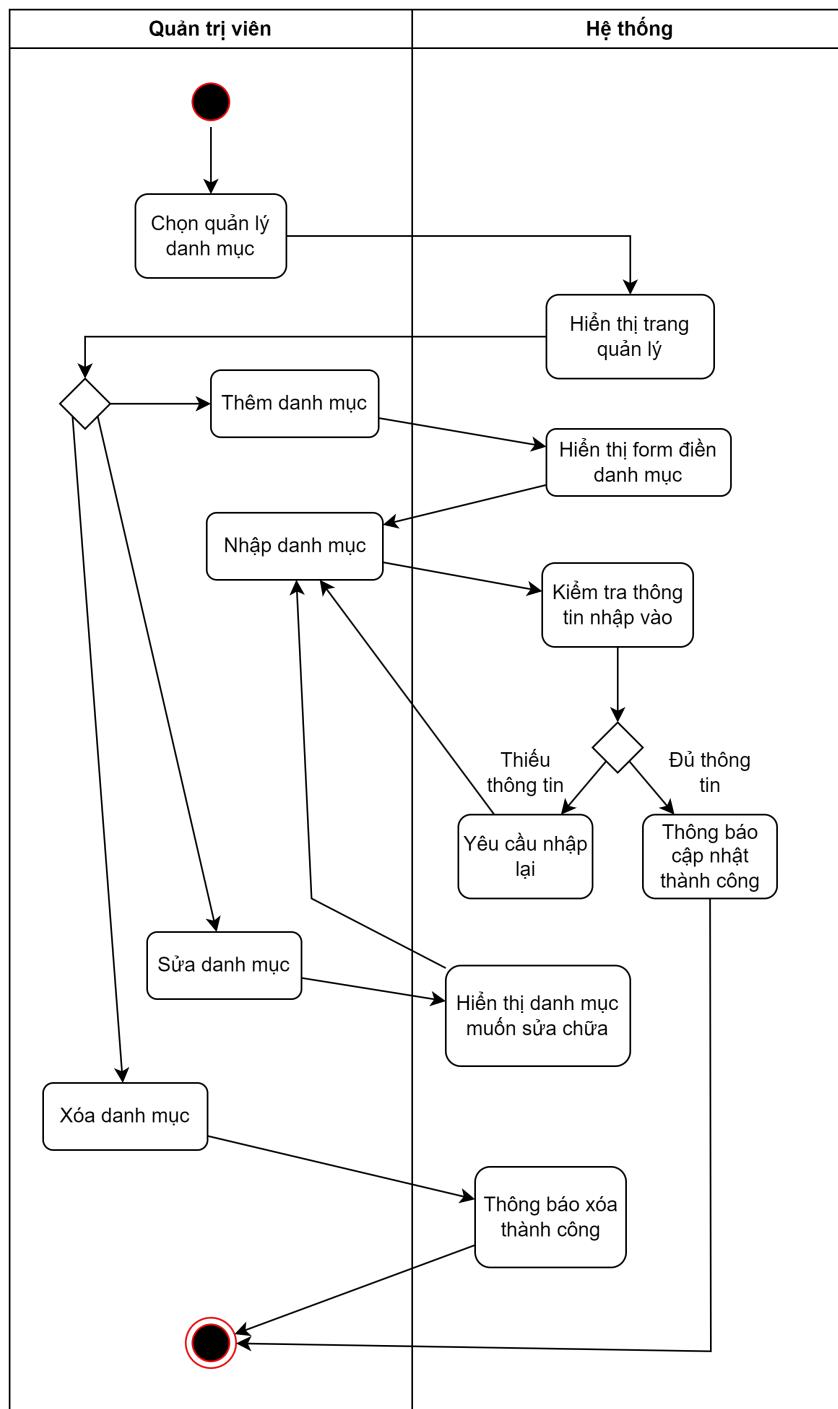
**Luồng hoạt động chính:**

1. Quản trị viên chọn quản lý danh mục.
2. Hệ thống hiển thị danh sách danh mục.
3. Quản trị viên thực hiện thêm danh mục.
4. Hệ thống thêm dữ liệu mới được bổ sung.
5. Quản trị viên chọn danh mục muốn sửa và sửa thông tin danh mục.
6. Hệ thống cập nhật dữ liệu mới.
7. Quản trị viên chọn danh mục muốn xóa và xóa danh mục.
8. Hệ thống xóa danh mục và hiển thị danh sách danh mục.

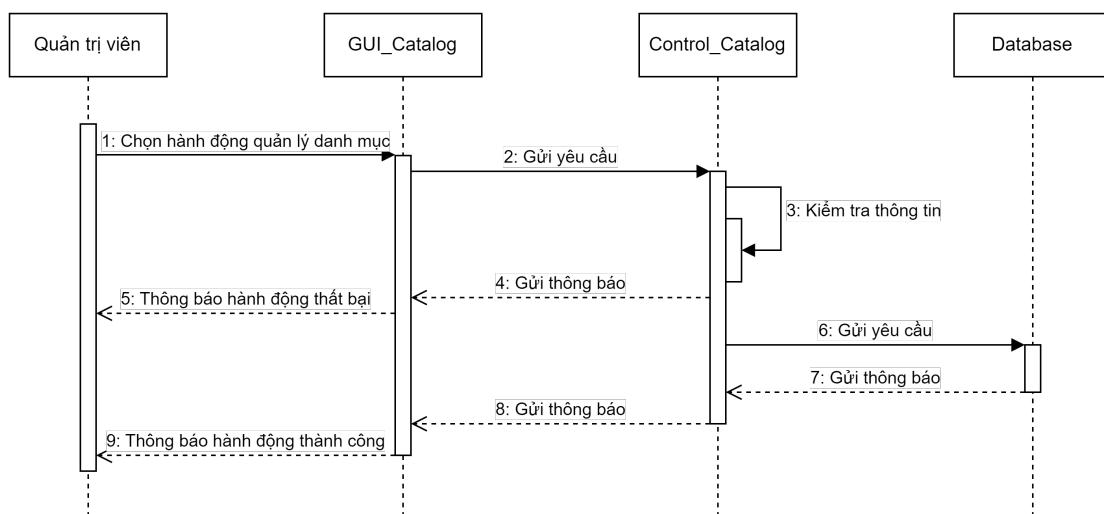
**Luồng hoạt động phát sinh:**

1. Nếu quản trị viên không nhập đầy đủ thông tin vào các trường có dấu (\*) thì hệ thống yêu cầu nhập đầy đủ thông tin.
2. Nếu danh mục đã tồn tại, hệ thống sẽ thông báo cho quản trị viên nhập tên khác.

**Hậu điều kiện:** Quản trị viên thêm, sửa, xóa danh mục thành công.



**Hình 2.27:** Biểu đồ hoạt động của usecase Quản lý danh mục



**Hình 2.28:** Biểu đồ tuần tự của usecase Quản lý danh mục

### 2.3.13 Đặc tả use case Quản lý quảng cáo

Hình 2.29 và hình 2.30 lần lượt là biểu đồ hoạt động và biểu đồ tuần tự của usecase quản lý quảng cáo trong đó:

**Tác nhân:** Quản trị viên

**Mô tả:** Mô tả quá trình quản lý quảng cáo gồm thêm sửa xóa của quản trị viên

**Tiền điều kiện:** Quản trị viên đăng nhập vào hệ thống.

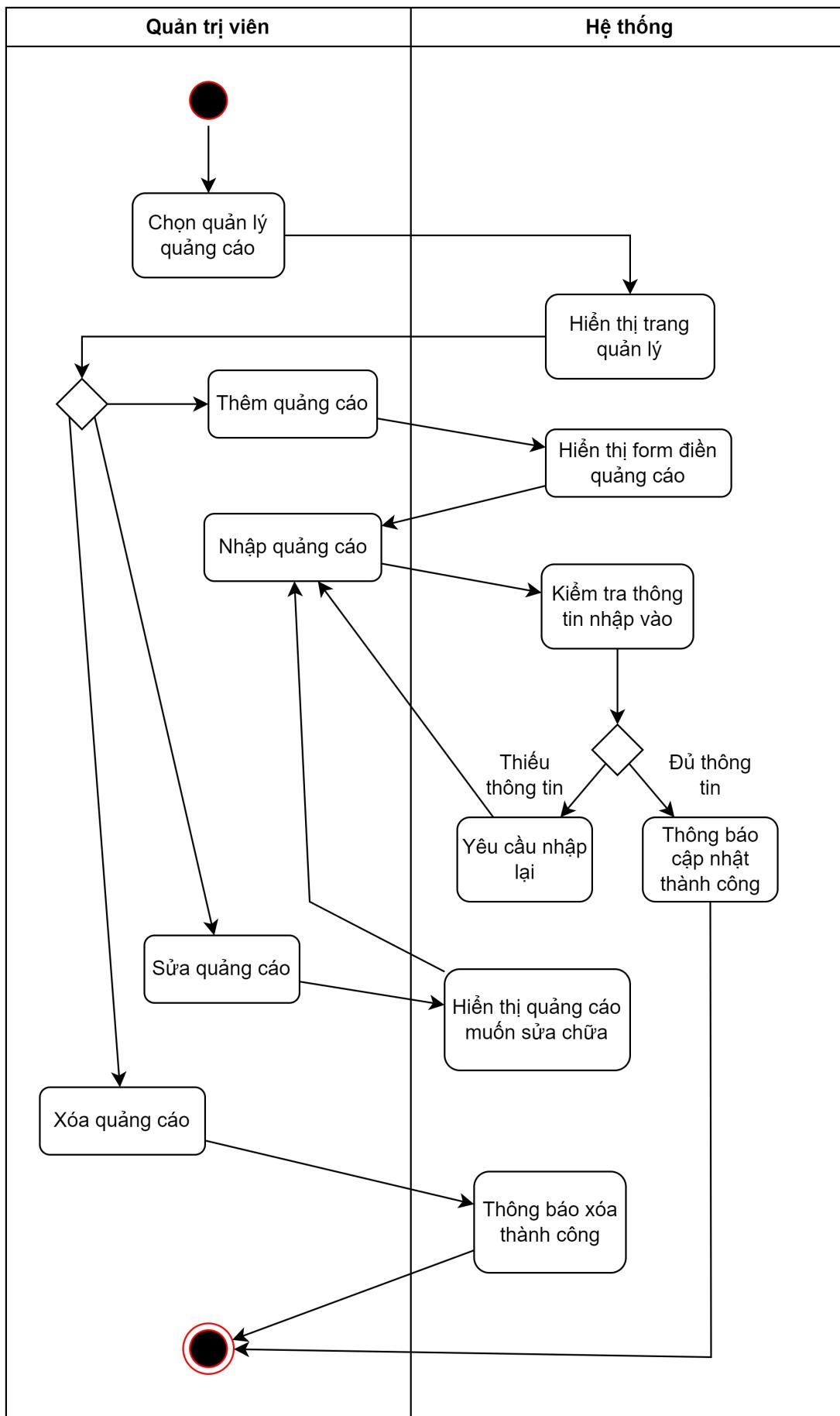
**Luồng hoạt động chính:**

- Quản trị viên chọn quản lý quảng cáo.
- Hệ thống hiển thị danh sách quảng cáo.
- Quản trị viên thực hiện thêm quảng cáo.
- Hệ thống thêm dữ liệu mới được bổ sung.
- Quản trị viên chọn danh mục muốn sửa và sửa thông tin quảng cáo.
- Hệ thống cập nhật dữ liệu mới.
- Quản trị viên chọn quảng cáo muốn xóa và xóa quảng cáo.
- Hệ thống xóa quảng cáo và hiển thị danh sách quảng cáo.

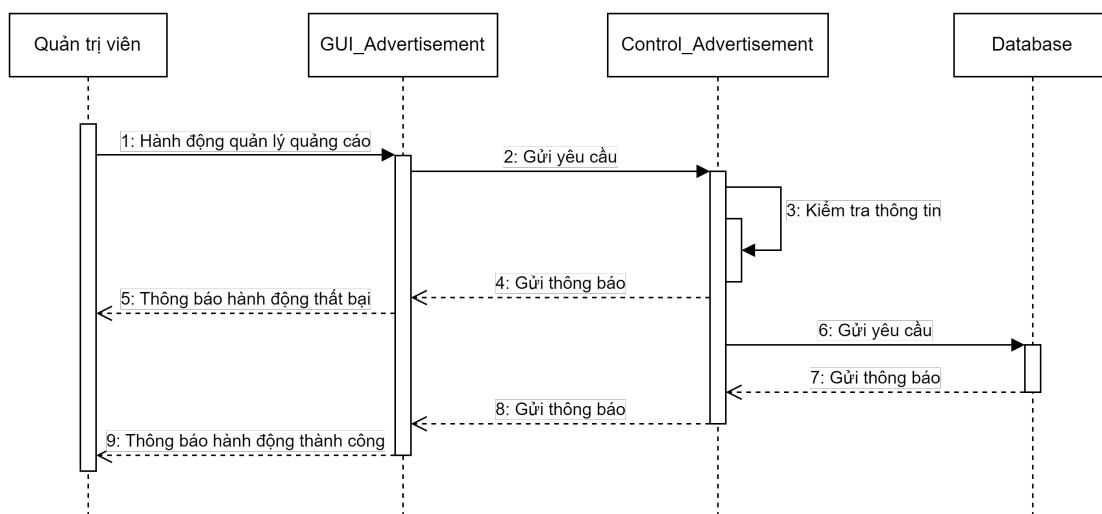
**Luồng hoạt động phát sinh:**

- Nếu quản trị viên không nhập đầy đủ thông tin vào các trường có dấu (\*) thì hệ thống yêu cầu nhập đầy đủ thông tin.

**Hậu điều kiện:** Quản trị viên thêm, sửa, xóa quảng cáo thành công.



**Hình 2.29:** Biểu đồ hoạt động của usecase Quản lý quảng cáo



Hình 2.30: Biểu đồ tuần tự của usecase Quản lý quảng cáo

### 2.3.14 Đặc tả use case Quản lý tài khoản

Hình 2.31 và hình 2.32 lần lượt là biểu đồ hoạt động và biểu đồ tuần tự của usecase quản lý tài khoản trong đó:

**Tác nhân:** Quản trị viên

**Mô tả:** Mô tả quá trình quản lý tài khoản gồm xem hoặc khóa của quản trị viên

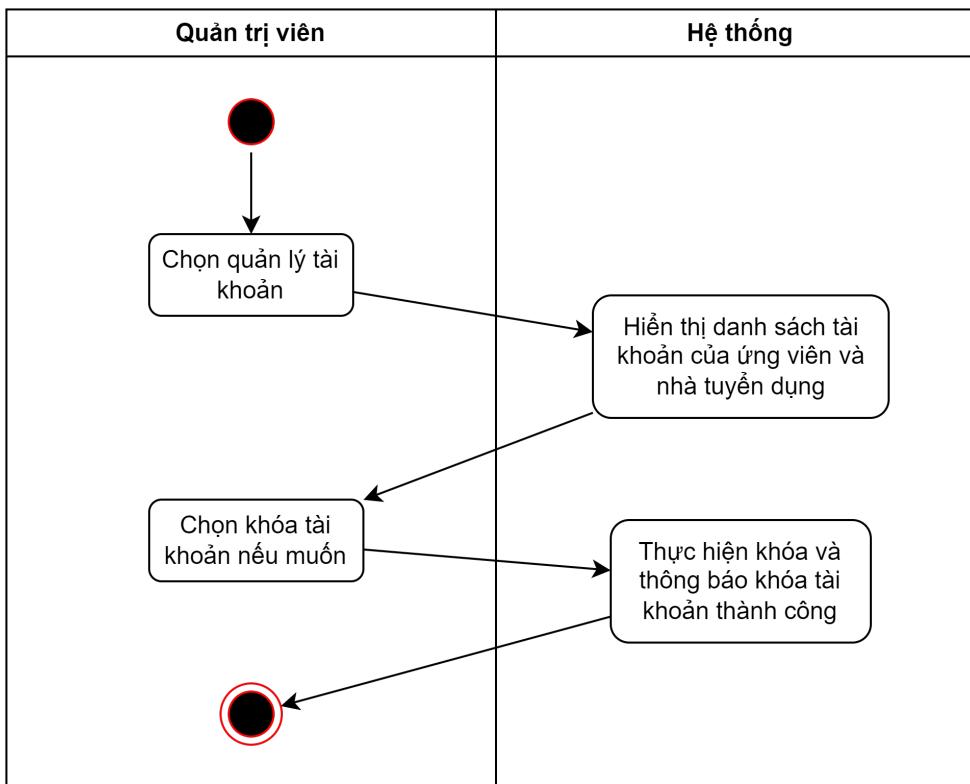
**Tiền điều kiện:** Quản trị viên đăng nhập vào hệ thống.

**Luồng hoạt động chính:**

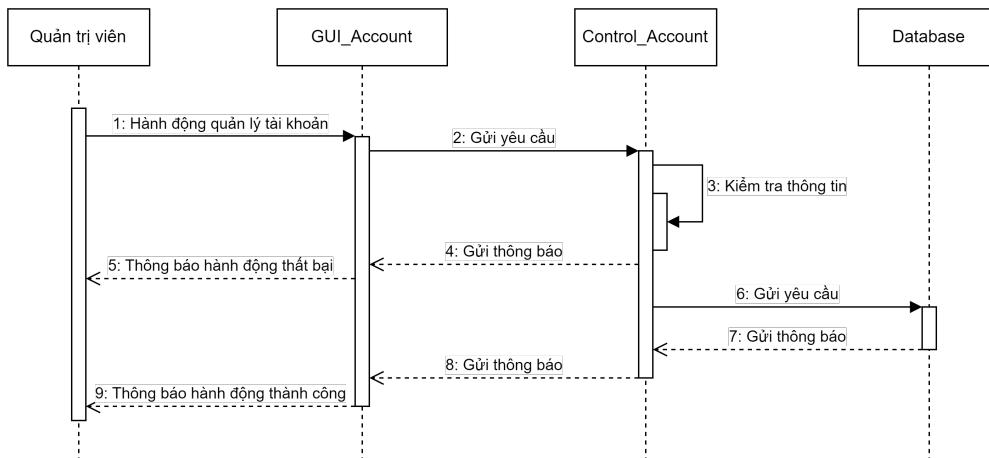
- Quản trị viên chọn quản lý tài khoản.
- Hệ thống vào trang quản lý tài khoản và hiển thị thông tin tài khoản của ứng viên, nhà tuyển dụng.
- Quản trị viên có thể xem hoặc khóa tài khoản của ứng viên và nhà tuyển dụng.
- Hệ thống cập nhật dữ liệu.

**Luồng hoạt động phát sinh:** Không

**Hậu điều kiện:** Quản trị viên xem hoặc khóa tài khoản thành công.



**Hình 2.31:** Biểu đồ hoạt động của usecase Quản lý tài khoản



**Hình 2.32:** Biểu đồ tuần tự của usecase Quản lý tài khoản

### 2.4 Yêu cầu phi chức năng

#### Dễ sử dụng:

- Giao diện thân thiện với người dùng: Giao diện người dùng của nền tảng phải trực quan, dễ sử dụng cho cả ứng viên và nhà tuyển dụng. Nó phải tuân theo các nguyên tắc trong thiết kế UI/UX để đảm bảo trải nghiệm người dùng liền mạch.
- Khả năng tiếp cận: Nền tảng phải tuân thủ các hướng dẫn về khả năng tiếp cận, đảm bảo rằng nó có thể được sử dụng bởi những người khuyết tật. Nó cần hỗ trợ đọc màn hình, điều hướng bằng bàn phím và các công cụ khả năng tiếp cận khác để phục vụ mọi đối tượng người dùng đa dạng.

#### Dễ bảo trì:

- Tính mô-đun và tài liệu hóa: Cơ sở mã nguồn của cả phần backend và frontend phải có tính mô-đun và được ghi chép đầy đủ. Điều này sẽ tạo điều kiện dễ dàng cho việc bảo trì và cải tiến trong tương lai, giúp các nhà phát triển hiểu và sửa đổi mã nguồn dễ dàng hơn.
- Ghi nhật ký và giám sát lỗi: Cả phía máy chủ (backend) và phía giao diện người dùng (frontend) đều cần tích hợp cơ chế ghi nhật ký lỗi và giám sát mạnh mẽ. Điều này sẽ cho phép xác định và giải quyết vấn đề nhanh chóng, đồng thời theo dõi hiệu suất hệ thống và việc sử dụng tài nguyên.

#### Cơ sở dữ liệu:

- Bảo mật dữ liệu: Cơ sở dữ liệu MySQL sẽ triển khai các cơ chế mã hóa phù hợp để đảm bảo an ninh và tính bảo mật của thông tin người dùng, bao gồm chi tiết cá nhân và dữ liệu ứng dụng. Việc truy cập vào dữ liệu nhạy cảm nên được giới hạn và kiểm soát.
- Sao lưu và khôi phục dữ liệu: Việc thực hiện sao lưu cơ sở dữ liệu định kỳ và tự động là cần thiết để ngăn chặn mất dữ liệu trong trường hợp các sự cố phần cứng hoặc các sự kiện không dự kiến khác. Ngoài ra, cần có kế hoạch khôi phục dữ liệu để khôi phục hệ thống về trạng thái ổn định cuối cùng trong trường hợp dữ liệu bị hỏng hoặc bị lỗi.

#### Công nghệ sử dụng:

- Tính tương thích nền tảng: Nền tảng sẽ được phát triển bằng cách sử dụng các công nghệ đảm bảo tính tương thích với các trình duyệt web hàng đầu, hệ điều hành, đồng thời cung cấp trải nghiệm người dùng nhất quán trên các nền tảng khác nhau.

## CHƯƠNG 2. KHẢO SÁT VÀ PHÂN TÍCH YÊU CẦU

---

Những yêu cầu phi chức năng này bổ sung cho những yêu cầu chức năng của nền tảng tìm kiếm việc làm, góp phần vào tổng thể thành công của nó trong việc đáp ứng nhu cầu của người dùng và đảm bảo tính bền vững, đáng tin cậy và thân thiện với người dùng lâu dài.

## CHƯƠNG 3. CÔNG NGHỆ SỬ DỤNG

### 3.1 Hệ quản trị cơ sở dữ liệu MySQL

#### 3.1.1 Giới thiệu

MySQL [1] là một hệ quản trị cơ sở dữ liệu mạnh mẽ được sử dụng để tổ chức và lưu trữ thông tin một cách có cấu trúc. Nó cho phép truy cập dữ liệu nhanh chóng và thuận tiện, điều này đã khiến nó trở nên phổ biến trong cộng đồng phát triển phần mềm. Với việc là một hệ thống mã nguồn mở, MySQL được sử dụng rộng rãi cho nhiều ứng dụng và hỗ trợ nhiều ngôn ngữ lập trình. Độ ổn định và hiệu suất của nó khiến MySQL trở thành sự lựa chọn ưa thích cho phát triển ứng dụng.

#### 3.1.2 Đặc trưng

MySQL hoạt động như một phần mềm quản lý cơ sở dữ liệu, tương tự như SQL Server của Microsoft. Nó quản lý dữ liệu thông qua các cơ sở dữ liệu, mỗi cơ sở dữ liệu chứa các bảng liên quan với dữ liệu cụ thể. MySQL triển khai một cơ chế quản lý người dùng, giúp người dùng có thể xử lý một hoặc nhiều cơ sở dữ liệu đồng thời. Để truy cập, cần sử dụng tên người dùng và mật khẩu với các quyền thích hợp. Khả năng tương thích với môi trường Java nâng cao khả năng của MySQL, cho phép nhà phát triển viết và chỉnh sửa mã MySQL bằng cách sử dụng các IDE Java một cách hiệu quả.

#### 3.1.3 Ưu điểm của MySQL

Ra mắt hoàn toàn miễn phí, hệ quản trị cơ sở dữ liệu MySQL mang lại nhiều lợi ích thực tế, được áp dụng trong nhiều dự án của các công ty công nghệ, trong đó có các công ty lập trình như Viettel, Misa,... Hãy cùng tìm hiểu những lợi ích của MySQL ngay bên dưới:

- **Hiệu suất cao:** Hầu hết, hệ quản trị cơ sở dữ liệu MySQL rất phổ biến đối với các nhà phát triển trong quá trình phát triển ứng dụng. Và họ đánh giá cao hiệu suất của MySQL. Với kiến trúc công cụ lưu trữ, MySQL dành riêng cho các ứng dụng chuyên biệt, đặc biệt là các trang web có dung lượng lớn phục vụ hàng triệu khách hàng. Hoặc đối với các hệ thống xử lý giao dịch tốc độ cao, MySQL có thể đáp ứng được khả năng xử lý khắt khe của mọi hệ thống. Đặc biệt, với các tiện ích tải tốc độ cao, cơ chế xử lý tiên tiến, MySQL cung cấp tất cả các tính năng cần thiết, khiến nó trở thành giải pháp hoàn hảo ngay cả đối với các hệ thống kinh doanh đòi hỏi khắt khe nhất hiện nay.
- **Hỗ trợ giao dịch mạnh mẽ:** Hệ quản trị cơ sở dữ liệu MySQL không chỉ mang lại hiệu suất cao mà còn cung cấp một trong những công cụ giao dịch

cơ sở dữ liệu tốt nhất trên thị trường hiện nay. Tính năng này bao gồm khóa cấp độ luồng không hạn chế, hỗ trợ giao dịch ACID hoàn chỉnh, khả năng giao dịch được phân loại và hỗ trợ giao dịch đa dạng mà không làm cản trở người đọc và ngược lại. Với MySQL, dữ liệu sẽ được đảm bảo trong suốt thời gian hiệu lực của máy chủ. Các cấp độ giao dịch độc lập sẽ được chuyên môn hóa, nếu phát hiện ngay lỗi bê tắc.

- **Tốc độ nhanh:** Đánh giá chung của các nhà phát triển là hệ thống quản lý cơ sở dữ liệu MySQL là cơ sở dữ liệu nhanh nhất. Đây là nơi để các website trao đổi dữ liệu thường xuyên vì nó có công cụ xử lý tốc độ cao. Khả năng chèn dữ liệu cực nhanh và hỗ trợ mạnh mẽ các chức năng chuyên dụng cho website. Các tính năng này cũng được sử dụng cho môi trường lưu trữ dữ liệu mà hệ thống quản trị tăng cường lên hàng terabyte cho các máy chủ đơn lẻ.
- **Dễ Sử Dụng:** Ngoài việc MySQL nổi tiếng với tốc độ cao và tính ổn định, nó thực sự là một hệ quản trị cơ sở dữ liệu rất đơn giản, rất dễ sử dụng. Nó ít phức tạp hơn trong việc cài đặt và quản trị so với các hệ thống lớn. Đặc biệt, nó có thể hoạt động trên tất cả các hệ điều hành.
- **Hỗ trợ Ngôn ngữ truy vấn:** Hệ quản trị cơ sở dữ liệu MySQL là ngôn ngữ được ưa chuộng cho tất cả các hệ thống cơ sở dữ liệu hiện đại. Người dùng có thể truy cập đầy đủ MySQL bằng cách sử dụng các ứng dụng hỗ trợ ODBC (một giao thức truyền thông cơ sở dữ liệu được phát triển bởi Microsoft). Nhiều khách hàng có thể truy cập máy chủ đồng thời. Đặc biệt, khách hàng có thể sử dụng nhiều cơ sở dữ liệu cùng một lúc. Bạn cũng có thể truy cập MySQL một cách tương tác thông qua nhiều giao diện để nhập truy vấn và xem kết quả, chẳng hạn như các dòng lệnh yêu cầu của khách hàng, trình duyệt web,...
- **Bảo mật và kết nối Cao:** Điều quan trọng nhất đối với mọi doanh nghiệp là bảo mật dữ liệu tuyệt đối, và hệ thống quản lý cơ sở dữ liệu MySQL tích hợp các tính năng bảo mật tuyệt đối. MySQL được kết nối mạng đầy đủ và cơ sở dữ liệu có thể được truy cập từ mọi nơi trên internet. Bạn có thể chia sẻ dữ liệu của mình với bất kỳ ai, bất cứ khi nào và ở bất kỳ đâu bạn muốn. Tuy nhiên, MySQL kiểm soát quyền truy cập để những người không nên xem dữ liệu của bạn sẽ không thể xem nó. Với xác nhận quyền truy cập cơ sở dữ liệu, MySQL được trang bị các kỹ thuật mạnh mẽ. Chỉ những người dùng được xác minh mới có thể truy cập vào cơ sở dữ liệu. Ngoài ra, SSH và SSL cũng được hỗ trợ để đảm bảo một kết nối an toàn và bảo mật. Các tiện ích sao lưu và khôi phục do MySQL cung cấp, các hệ thống quản lý cơ sở dữ liệu và các nhà cung cấp

phần mềm bên thứ ba cho phép sao lưu logic và vật lý cũng như khôi phục đầy đủ một lần.

- **Chuyển đổi cao:** MySQL có thể hoạt động trên tất cả các hệ điều hành, chạy trên mọi thứ từ máy tính cá nhân đến máy chủ. Hệ quản trị cơ sở dữ liệu MySQL cung cấp nhiều tính năng linh hoạt. Nó có khả năng xử lý các ứng dụng được nhúng sâu với dung lượng chỉ 1MB, đồng thời có thể chạy các kho dữ liệu lớn lên đến terabytes thông tin. Tính chất mã nguồn mở của MySQL cho phép tùy chỉnh theo ý muốn để thêm các yêu cầu phù hợp vào máy chủ cơ sở dữ liệu.
- **Mã nguồn mở miễn phí:** Mỗi quan ngại của nhiều doanh nghiệp khi gặp khó khăn trong việc giao toàn bộ phần mềm cho mã nguồn mở. Bởi vì khó khăn trong việc tìm kiếm hỗ trợ chuyên nghiệp hoặc bảo mật để phục vụ một cách chuyên nghiệp. Nhưng vấn đề này sẽ biến mất nếu sử dụng hệ quản trị cơ sở dữ liệu MySQL. Với MySQL, mọi cam kết đều rất rõ ràng, mọi sự cố đều được bồi thường bởi MySQL. Bạn có thể tìm thấy các nguồn lực sẵn có mà hệ thống quản lý này hỗ trợ. Cộng đồng MySQL rất trách nhiệm. Họ thường trả lời các câu hỏi trên danh sách gửi thư chỉ trong vài phút. Nếu xảy ra lỗi, các nhà phát triển sẽ cung cấp cho bạn bản vá nhanh nhất. Và bản vá đó sẽ ngay lập tức có sẵn trên internet.
- **Chi phí sở hữu thấp:** Hệ thống quản trị cơ sở dữ liệu MySQL là miễn phí cho hầu hết các sử dụng trong một tổ chức. Do đó, việc sử dụng MySQL cho các dự án, doanh nghiệp mới nhận ra rằng sự tiết kiệm chi phí là rất đáng kể. Người dùng MySQL cũng không cần phải dành nhiều thời gian để khắc phục sự cố hoặc gián đoạn hoạt động.

Không gì hoàn hảo hơn khi doanh nghiệp của bạn có được sự hỗ trợ đắc lực từ hệ thống quản trị cơ sở dữ liệu MySQL. Cơ hội phát triển sẽ trong tầm tay bạn, đơn giản, dễ dàng và hiệu quả cao.

### 3.2 Tổng quan về Java

#### 3.2.1 Giới thiệu

Java [2] là một ngôn ngữ lập trình được phát triển bởi Sun Microsystems vào năm 1995. Nó là một ngôn ngữ lập trình hướng đối tượng và là người kế nhiệm trực tiếp của C/C++. Ban đầu có tên là "Oak," ngôn ngữ sau đó được đổi tên thành "Java" do vấn đề về thương hiệu và trở thành tên chính thức cho Ngôn ngữ Lập trình Java. Java được đặt tên theo một hòn đảo ở Indonesia, nổi tiếng với cà phê Peet, một đồ uống phổ biến của các kỹ sư của Sun.

### 3.2.2 Ứng dụng của Java

Java được sử dụng trong nhiều ứng dụng, bao gồm:

- Phát triển ứng dụng cho các thiết bị điện tử thông minh và các ứng dụng kinh doanh quy mô lớn.
- Tạo các trang web có nội dung động (web applet) và cải thiện chức năng của máy chủ.
- Xây dựng nhiều loại ứng dụng, bao gồm cơ sở dữ liệu, mạng, internet, viễn thông và giải trí.

### 3.2.3 Đặc điểm của Java

Java sở hữu những đặc điểm cơ bản sau:

- Đơn giản và quen thuộc: Java kế thừa những đặc điểm từ C/C++ nhưng lược bỏ những tính năng khó nhất của C++ để dễ sử dụng.
- Java là ngôn ngữ lập trình hướng đối tượng.
- Mạnh mẽ và an toàn, có cơ chế thu gom rác tự động
- Không phụ thuộc kiến trúc, độc lập nền tảng và di động.
- Hiệu suất cao.
- Java là ngôn ngữ vừa biên dịch vừa thông dịch.
- Khả năng đa nhiệm, cho phép nhiều tiến trình diễn ra đồng thời.

### 3.2.4 Java Platforms

Java Platforms bao gồm ba thành phần chính:

- Máy ảo Java (Java VM): Một máy ảo để thực thi các chương trình Java.
- Giao diện lập trình ứng dụng Java (Java API).
- Bộ công cụ phát triển Java (JDK): Bao gồm trình biên dịch, trình thông dịch, trình trợ giúp, trình soạn thảo tài liệu và thư viện chuẩn

### 3.2.5 Tiêu chuẩn của môi trường Java điển hình

Các chương trình Java thường trải qua năm giai đoạn chính:

- Trình soạn thảo: Lập trình viên viết chương trình được lưu ở định dạng .java.
- Trình biên dịch: Các chương trình được biên dịch thành mã byte (định dạng .class), cho phép tương thích đa nền tảng.
- Trình nạp lớp: Đọc các tệp .class chứa mã byte và lưu chúng vào bộ nhớ.
- Trình xác minh mã byte: Đảm bảo rằng mã byte hợp lệ và tuân thủ các tiêu

chuẩn bảo mật Java.

- Trình thông dịch: Biên dịch mã byte thành mã máy để thực thi.

### 3.3 Tổng quan về Spring Boot

Spring Boot [3] là một extension của Spring Framework giúp đơn giản hóa các bước cấu hình cần thiết trong các ứng dụng Spring truyền thống. Đây là một dự án được phát triển bằng ngôn ngữ Java trong hệ sinh thái Spring framework.

#### 3.3.1 Những tính năng nổi bật của Spring Boot

Spring Boot cung cấp một số tính năng nổi bật, bao gồm:

- Tạo các ứng dụng Spring độc lập.
- Nhúng trực tiếp Tomcat, Jetty hoặc Undertow mà không cần triển khai tệp WAR.
- Tự động chạy cấu hình Spring khi cần thiết.
- Loại bỏ nhu cầu về mã cấu hình và cấu hình XML.

#### 3.3.2 Một số tính năng đặc biệt

Spring Boot cung cấp các tính năng đặc biệt để tối ưu hóa quá trình phát triển và tăng năng suất trong quá trình lập trình. Một số tính năng này bao gồm:

- Đơn giản hóa và thắt chặt cấu hình XML trong Spring.
- Giảm thời gian lập trình đến mức tối thiểu.
- Giúp người dùng có kiến thức lập trình hạn chế có thể xây dựng ứng dụng.

#### 3.3.3 Các tính năng quan trọng của Spring Boot

Spring Boot cung cấp một số tính năng cần thiết như:

- SpringApplication: Một cách thuận tiện để chạy các ứng dụng Spring là chỉ cần gọi phương thức run() và Spring Boot sẽ lo phần còn lại, yêu cầu các cấu hình tối thiểu.
- Externalized Configuration: Cho phép tạo các ứng dụng có thể chạy trên các môi trường khác nhau bằng cách cấu hình bên ngoài.
- Profiles: Cho phép gán các cấu hình khác nhau cho các môi trường khác nhau để quản lý dễ dàng hơn.
- Logging: Quản lý toàn bộ chức năng nhật ký nội bộ theo mặc định.

Ngoài những tính năng này, Spring Boot còn cung cấp các tính năng khác như Bảo mật, Phát triển ứng dụng Web, Công nghệ SQL, Gửi email, Xác thực và Gọi các dịch vụ REST bằng RestTemplate/WebClient, cùng nhiều tính năng khác.

### 3.4 Tổng quan về React

React [4] là một thư viện JavaScript được phát triển bởi Facebook để xây dựng giao diện người dùng. Nó cho phép nhà phát triển tạo ra các thành phần UI có thể tái sử dụng và quản lý trạng thái của ứng dụng một cách hiệu quả. React tuân theo khái niệm về DOM ảo, giúp nó cập nhật một cách hiệu quả chỉ những phần cần thiết của DOM thực sự, từ đó cải thiện hiệu suất.

#### 3.4.1 Khái niệm chính

React được xây dựng trên một số khái niệm quan trọng, giúp nó trở thành một thư viện mạnh mẽ và phổ biến cho việc xây dựng giao diện người dùng.

- **Components:** Ứng dụng React được xây dựng bằng cách sử dụng các thành phần, đó là những phần giao diện người dùng độc lập và có thể tái sử dụng. Các thành phần có thể được kết hợp để tạo ra giao diện người dùng phức tạp.
- **Virtual DOM:** React duy trì sự thể hiện ảo của DOM thực trong bộ nhớ, cho phép nó thực hiện các cập nhật hiệu quả và giảm thiểu thao tác trực tiếp lên DOM thực.
- **Unidirectional Data Flow:** React tuân theo luồng dữ liệu một chiều (unidirectional data flow), trong đó dữ liệu truyền từ các thành phần cha đến các thành phần con. Điều này giúp dễ hiểu hơn về cách dữ liệu thay đổi trong ứng dụng.
- **JSX (JavaScript XML):** JSX là một cú pháp mở rộng cho JavaScript, cho phép các nhà phát triển viết mã giống HTML bên trong JavaScript, giúp mô tả các thành phần giao diện người dùng một cách dễ dàng hơn.

#### 3.4.2 Ưu điểm của React

- **Khả năng sử dụng lại thành phần:** Với kiến trúc dựa trên thành phần của React, các nhà phát triển có thể tạo các thành phần có thể tái sử dụng, giảm trùng lặp mã và cải thiện khả năng bảo trì.
- **Hiệu quả của Virtual DOM:** Khái niệm Virtual DOM cho phép React giảm thiểu các thao tác DOM, mang lại hiệu suất tốt hơn và hiển thị nhanh hơn.
- **Hệ sinh thái lớn:** React có một cộng đồng lớn và tích cực, cung cấp một hệ sinh thái phong phú của thư viện, công cụ và các tiện ích mở rộng để tăng cường năng suất phát triển.
- **Cú pháp khai báo:** Cú pháp khai báo của React cho phép các nhà phát triển mô tả trạng thái giao diện người dùng mong muốn và React tự động xử lý các cập nhật cơ bản.

- **Hỗ trợ từ Facebook và cộng đồng:** Được phát triển và duy trì bởi Facebook, React hưởng lợi từ các bản cập nhật và sự hỗ trợ đều đặn, và cộng đồng lớn đảm bảo tài liệu và nguồn lực phong phú.

### 3.5 Tổng quan về HTML và CSS

HTML (Hypertext Markup Language) và CSS (Cascading Style Sheets) là những công nghệ cơ bản được sử dụng trong phát triển web. HTML cung cấp cấu trúc và nội dung của trang web, trong khi CSS chịu trách nhiệm trình bày và bố cục.

#### 3.5.1 HTML (Hypertext Markup Language)

HTML [5] là ngôn ngữ đánh dấu được sử dụng để tạo cấu trúc và nội dung của các trang web. Nó bao gồm nhiều phần tử khác nhau, mỗi phần tử được biểu thị bằng các thẻ, xác định các phần khác nhau của trang web. Các phần tử HTML có thể bao gồm tiêu đề, đoạn văn, hình ảnh, liên kết, danh sách, bảng,...

- **Thẻ (Tags):** HTML dựa trên hệ thống các thẻ, được sử dụng để định nghĩa và bao quanh các phần tử khác nhau của một trang web. Thẻ được viết giữa các dấu ngoặc nhọn (< >).
- **Phần tử và Thuộc tính:** Các phần tử HTML được tạo ra bằng cách sử dụng các thẻ, và một số phần tử có thể có các thuộc tính cung cấp thông tin bổ sung hoặc sửa đổi hành vi của chúng.
- **Thành phần ngữ nghĩa (Semantic Elements):** HTML5 giới thiệu các thành phần ngữ nghĩa như <header>, <nav>, <main>, <article>, <section>, <footer>,... giúp mang ý nghĩa đến nội dung và cải thiện khả năng truy cập và tối ưu hóa cho công cụ tìm kiếm.

HTML được sử dụng để cấu trúc nội dung của các trang web và ứng dụng. Nó là xương sống của mọi trang web và rất cần thiết để tạo ra một giao diện người dùng đầy đủ chức năng và dễ tiếp cận.

#### 3.5.2 CSS (Cascading Style Sheets)

CSS [6] là một ngôn ngữ thiết kế được sử dụng để kiểm soát cách trình bày và bố cục của tài liệu HTML. Nó cho phép các nhà phát triển xác định cách hiển thị các phần tử HTML trên màn hình, giúp kiểm soát màu sắc, phông chữ, kích thước, lề và các khía cạnh trực quan khác của trang web.

- **Bộ chọn (Selectors):** CSS sử dụng các bộ chọn để chọn các phần tử HTML và áp dụng các kiểu cho chúng. Bộ chọn có thể chọn các phần tử dựa trên tên thẻ, lớp, ID, hoặc các thuộc tính khác.
- **Thuộc tính và Giá trị:** Kiểu CSS được xác định bằng cách sử dụng các cặp

thuộc tính-giá trị. Mỗi thuộc tính đại diện cho một khía cạnh trực quan và giá trị của nó chỉ định cách tạo kiểu cho phần tử.

- **Luồng và Độ chính xác (Cascade and Specificity):** CSS tuân theo mô hình luồng, trong đó các kiểu có thể được kế thừa từ các phần tử cha và bị đè bởi các quy tắc cụ thể hơn. Độ chính xác xác định quy tắc kiểu nào được ưu tiên khi có nhiều quy tắc áp dụng cho cùng một phần tử.

CSS được sử dụng để kiểm soát bố cục và giao diện của trang web, cho phép các nhà phát triển tạo ra các thiết kế nhất quán và hấp dẫn trực quan trên các thiết bị và kích thước màn hình khác nhau.

### 3.5.3 Hợp nhất

HTML và CSS hoạt động cùng nhau để tạo ra các trang web. HTML cung cấp nội dung và cấu trúc, trong khi CSS xác định cách nội dung đó nên được trình bày và kiểu dáng. Bằng cách kết hợp hai công nghệ này, nhà phát triển có thể xây dựng các trang web và ứng dụng tương tác và hấp dẫn mỹ thuật.

### 3.5.4 Thực tiễn

Phát triển web hiện đại thường liên quan đến việc sử dụng các framework CSS (ví dụ: Bootstrap, Foundation) và các trình biên dịch trước ( preprocessors ) (ví dụ: Sass, Less) để tối ưu hóa quá trình phát triển và đảm bảo thiết kế linh hoạt và thân thiện.

### 3.5.5 Sự phát triển

Cả HTML và CSS đã phát triển qua nhiều năm, với HTML5 và CSS3 là các phiên bản mới nhất. Các tính năng và cải tiến mới tiếp tục được giới thiệu để nâng cao khả năng và linh hoạt của phát triển web.

Nhìn chung, HTML và CSS tạo thành xương sống của phát triển web và việc nắm vững các công nghệ này là điều cần thiết để tạo ra các trang web và ứng dụng hiệu quả và hấp dẫn về mặt hình ảnh.

### 3.5.6 Ứng dụng

React thường được sử dụng cho nhiều loại ứng dụng, bao gồm:

- **Single-Page Applications (SPAs):** React là lựa chọn lý tưởng để xây dựng các SPAs, nơi một trang HTML duy nhất được cập nhật động dựa trên tương tác của người dùng.
- **Giao diện người dùng động:** Các ứng dụng với giao diện người dùng phức tạp và tương tác, như bảng điều khiển và trực quan hóa dữ liệu, có thể hưởng lợi từ kiến trúc dựa trên thành phần của React.

- **Ứng dụng thời gian thực:** Quá trình render hiệu quả của React làm cho nó phù hợp cho các ứng dụng thời gian thực như hệ thống trò chuyện và các công cụ hợp tác.

Tính linh hoạt và hiệu suất của React khiến nó trở thành sự lựa chọn phổ biến đối với những nhà phát triển khi xây dựng giao diện người dùng hiện đại và tương tác cho ứng dụng web và di động.

## **CHƯƠNG 4. THIẾT KẾ, TRIỂN KHAI VÀ ĐÁNH GIÁ HỆ THỐNG**

### **4.1 Thiết kế kiến trúc**

#### **4.1.1 Lựa chọn kiến trúc phần mềm**

Model-View-Controller (MVC) là một mẫu kiến trúc phổ biến được sử dụng trong phát triển phần mềm để tổ chức cấu trúc và hành vi của một ứng dụng. Nó tách ứng dụng thành ba thành phần được kết nối với nhau: Model, View và Controller, mỗi thành phần có trách nhiệm riêng.

Đồ án hiện tại áp dụng mô hình thiết kế Model-View-Controller (MVC) để tổ chức mã nguồn của mình. Trong đồ án này, React được sử dụng như là phần View, chịu trách nhiệm xử lý giao diện người dùng và hiển thị dữ liệu cho người dùng. Ngược lại, SpringBoot đóng vai trò là phần Controller và Model, xử lý logic nghiệp vụ, truy xuất dữ liệu và cập nhật cơ sở dữ liệu.

##### **a, Model (Spring Boot)**

Model đại diện cho dữ liệu và logic nghiệp vụ của ứng dụng. Trong đồ án này, Spring Boot được sử dụng làm Model, chịu trách nhiệm quản lý việc truy xuất dữ liệu từ cơ sở dữ liệu, triển khai các hoạt động logic nghiệp vụ và cập nhật dữ liệu khi cần thiết.

- Quản lý dữ liệu và trạng thái của ứng dụng.
- Triển khai logic nghiệp vụ và xác thực dữ liệu.
- Thông báo cho View (React) và Controller về những thay đổi trong dữ liệu.

##### **b, View (React)**

React đóng vai trò là phần View trong đồ án này và chịu trách nhiệm cho tầng trình bày của ứng dụng. Nó đại diện cho giao diện người dùng và hiển thị dữ liệu được cung cấp bởi Model (Spring Boot) đến người dùng.

- Hiển thị dữ liệu được cung cấp bởi Model đến người dùng.
- Cung cấp giao diện tương tác thân thiện với người dùng.
- Xử lý đầu vào của người dùng và chuyển tiếp nó đến Controller (Spring Boot).

##### **c, Controller (Spring Boot)**

Controller đóng vai trò là một trung gian giữa Model (Spring Boot) và View (React). Nó nhận đầu vào từ người dùng từ View, xử lý nó và cập nhật Model tương ứng. Nó cũng cập nhật View (React) với dữ liệu mới nhất từ Model.

- Nhận và xử lý đầu vào từ người dùng từ View (React).

- Cập nhật Model (Spring Boot) dựa trên các hành động của người dùng.
- Cập nhật View (React) với dữ liệu mới nhất từ Model.

### d, Luồng tương tác

Mô hình thiết kế MVC tuân theo một luồng tương tác cụ thể, điều này cũng áp dụng cho đồ án này:

- Người dùng tương tác với View (React), cung cấp đầu vào.
- View (React) chuyển tiếp đầu vào của người dùng đến Controller (Spring Boot).
- Controller (Spring Boot) xử lý đầu vào và cập nhật Model tương ứng.
- Model (Spring Boot) thông báo cho View (React) và Controller về các thay đổi trong dữ liệu.
- View (React) truy xuất dữ liệu đã cập nhật từ Model (Spring Boot) và hiển thị nó đến người dùng.

### e, Ưu điểm của MVC

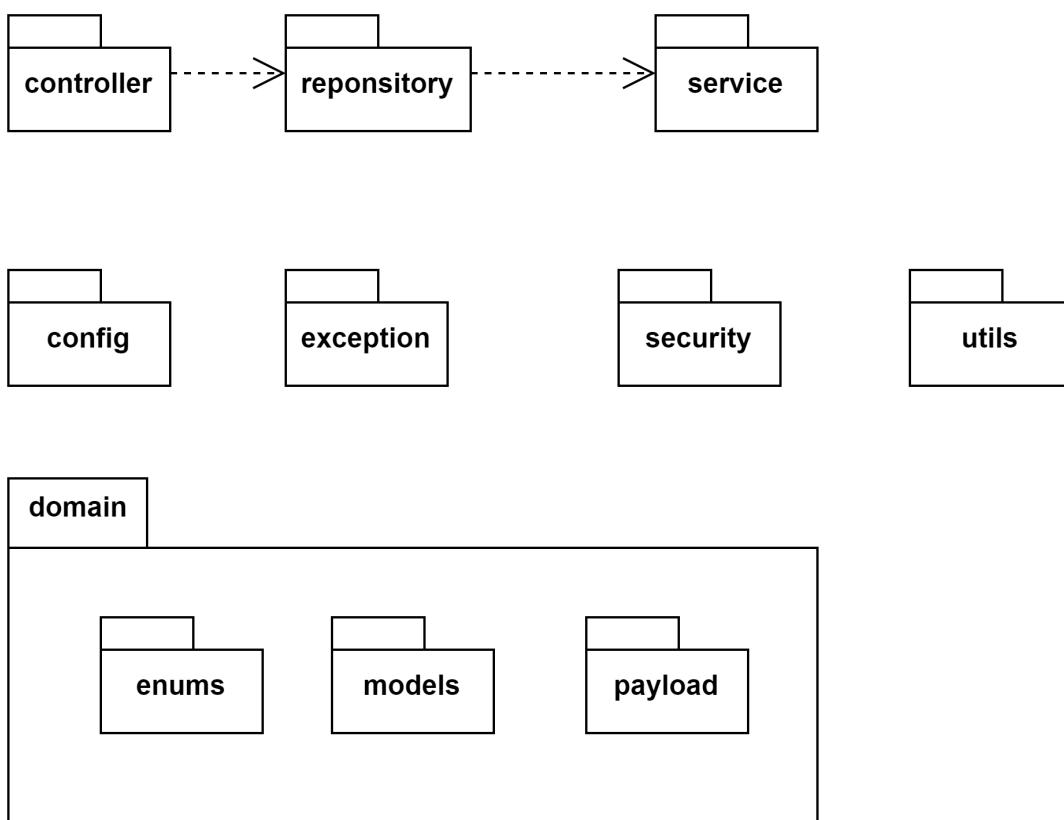
- Tách biệt các thành phần, làm cho ứng dụng dễ dàng bảo trì và mở rộng.
- Khả năng tái sử dụng các thành phần, vì mỗi thành phần có thể được phát triển độc lập.
- Cải thiện tổ chức mã và khả năng đọc.
- Hỗ trợ nhiều View và Controller, tạo điều kiện cho giao diện người dùng linh hoạt.

MVC được rộng rãi sử dụng trong các framework phát triển web như Ruby on Rails, Django, Spring và Laravel. Nó giúp nhà phát triển xây dựng ứng dụng web có khả năng mở rộng, có cấu trúc và dễ bảo trì.

Việc sử dụng mô hình MVC trong đồ án này, với React làm View và Spring Boot làm Controller và Model, cho phép có một mã nguồn mã nguồn cấu trúc và tổ chức tốt, đảm bảo sự phát triển và bảo trì hiệu quả của ứng dụng.

#### 4.1.2 Thiết kế tổng quan

Dưới đây là sơ đồ gói UML cho Java và Spring Framework:



**Hình 4.1:** Biểu đồ gói trong Spring Boot

Các gói controllers, services, repositories, models, config, exceptions, domains, security, và utils đều là phần của ứng dụng và được tổ chức dưới gói gốc.

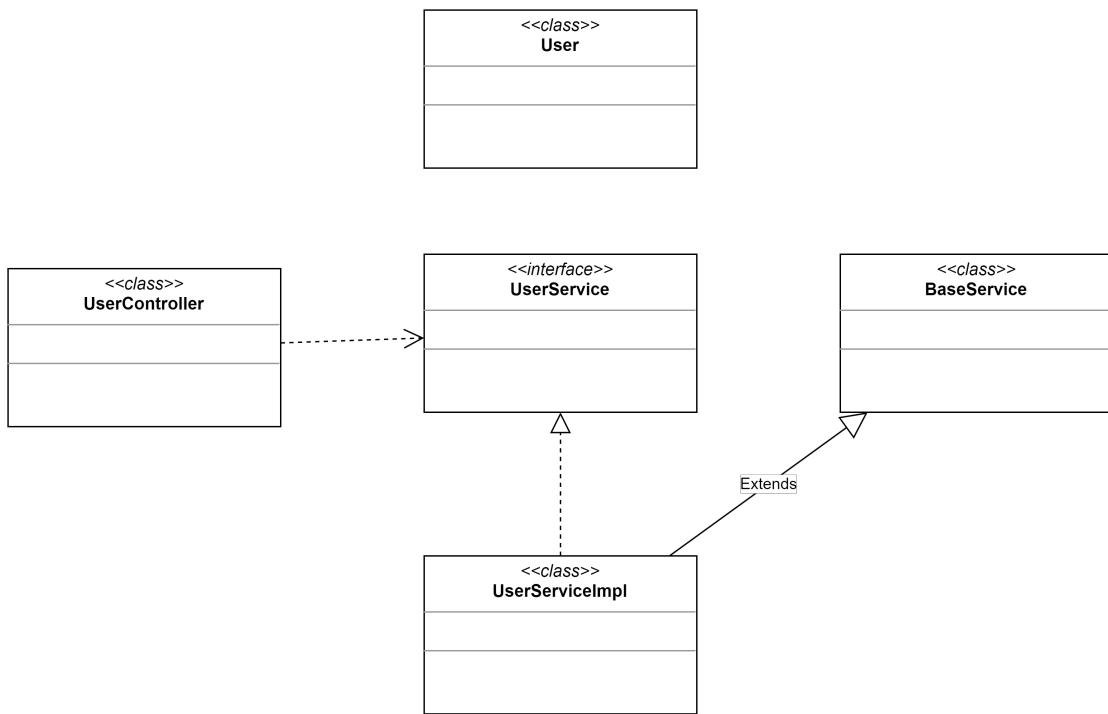
Gói controllers phụ thuộc vào gói services vì controllers thường gọi các phương thức của services để xử lý các yêu cầu HTTP.

Gói services phụ thuộc vào gói repositories, vì services thường tương tác với repositories để truy cập dữ liệu từ cơ sở dữ liệu.

Các gói exceptions, domains, security, và utils là các gói độc lập có thể được sử dụng trong các lớp khác nhau của ứng dụng, nhưng chúng vẫn là một phần của cấu trúc tổng thể.

#### 4.1.3 Thiết kế chi tiết gói

Spring Boot cung cấp một cơ chế "tiêm phụ thuộc" mạnh mẽ thông qua Spring Framework. "Tiêm phụ thuộc" cho phép chúng ta quản lý mối quan hệ giữa các lớp và các phụ thuộc của chúng một cách linh hoạt và không liên kết chặt chẽ. Bằng cách sử dụng các chú thích như @Autowired, @Component, @Service, @Repository,..., chúng ta có thể để Spring xử lý việc tạo và kết nối các bean (đối tượng) dựa trên các phụ thuộc của chúng. Do đó, dưới đây là một số biểu đồ mô tả mối quan hệ giữa các lớp:

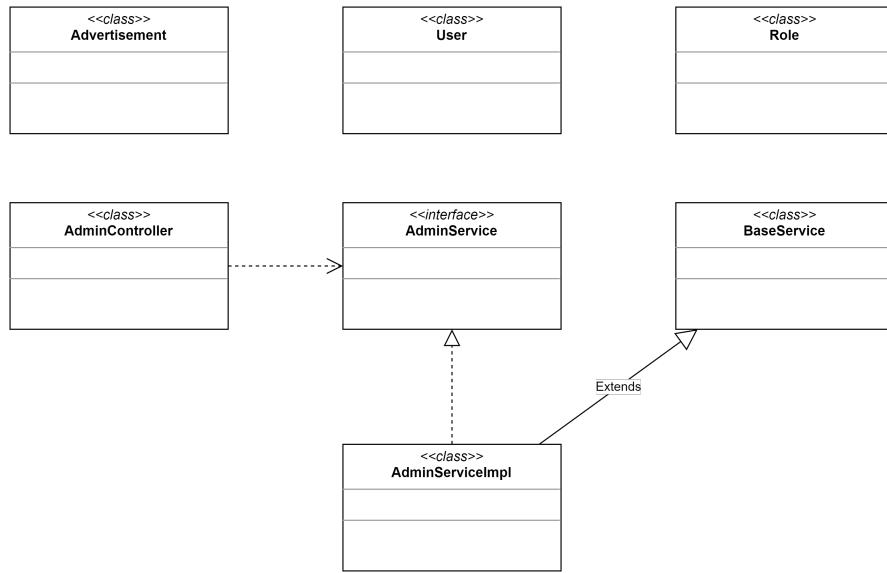


**Hình 4.2:** Biểu đồ chi tiết gói user

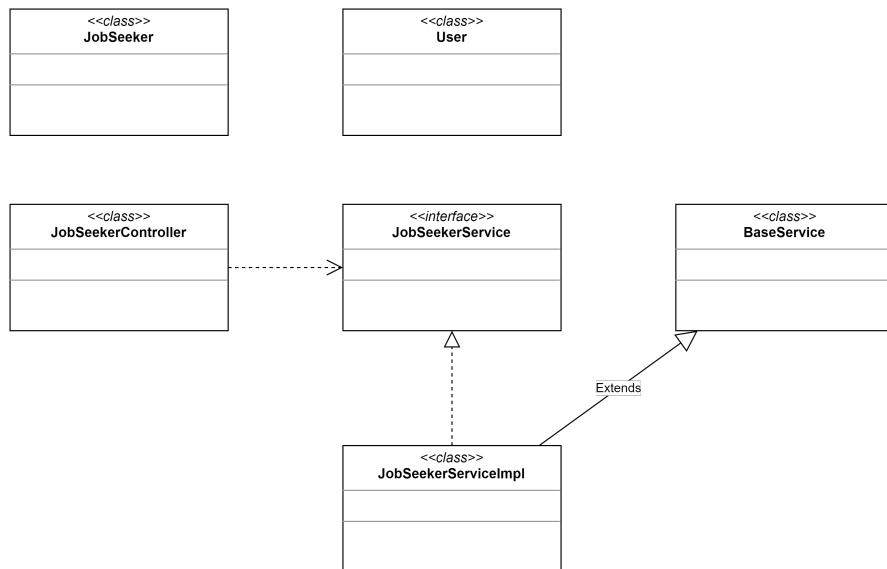
Với cài đặt này, khi một yêu cầu gửi đến UserController, nó sẽ ủy quyền logic nghiệp vụ cho UserServiceImpl có thể sử dụng các phương thức từ BaseService hoặc tương tác với các đối tượng truy cập dữ liệu để thực hiện các thao tác cần thiết. Luồng điều khiển tuân theo mẫu Spring Boot MVC tiêu chuẩn, trong đó controllers xử lý yêu cầu, services xử lý logic nghiệp vụ, và repositories xử lý truy cập dữ liệu.

Bằng cách sử dụng kiến trúc phân lớp này, bạn có thể tách biệt giữa các quan tâm và khả năng bảo trì cao trong ứng dụng Spring Boot của mình. Controller tiếp tục tập trung vào xử lý yêu cầu và phản hồi, service xử lý logic kinh doanh, và mã truy cập dữ liệu nằm trong repositories hoặc DAOs. Tương tự, chúng tôi có:

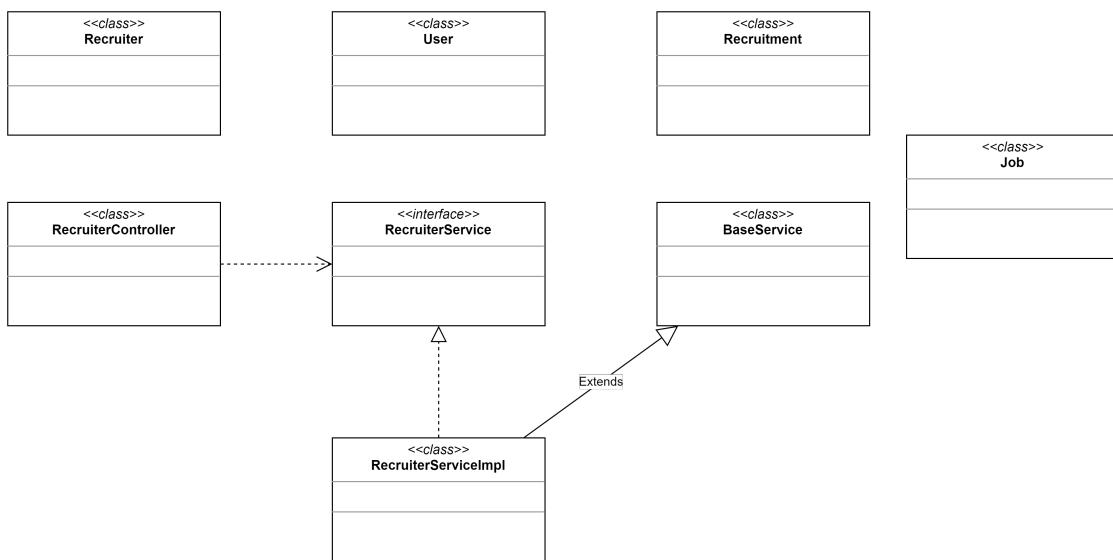
## CHƯƠNG 4. THIẾT KẾ, TRIỂN KHAI VÀ ĐÁNH GIÁ HỆ THỐNG



**Hình 4.3:** Biểu đồ chi tiết gói của admin



**Hình 4.4:** Biểu đồ chi tiết gói của ứng viên



**Hình 4.5:** Biểu đồ lớp của nhà tuyển dụng

## 4.2 Thiết kế chi tiết

### 4.2.1 Thiết kế giao diện người dùng

#### a, Giới thiệu

Thiết kế giao diện người dùng (UI) là một khía cạnh quan trọng trong ứng dụng này vì nó ảnh hưởng trực tiếp đến trải nghiệm và khả năng sử dụng của người dùng. Phần này nhằm mục đích mô tả các khía cạnh chính của thiết kế giao diện người dùng, bao gồm thông số kỹ thuật của màn hình mục tiêu, nguyên tắc thiết kế và minh họa trực quan về giao diện cho các chức năng thiết yếu.

#### b, Thông số màn hình mục tiêu

- Độ phân giải màn hình:** Ứng dụng này hướng đến nhiều loại thiết bị và để đảm bảo khả năng tương thích, chúng tôi đã thiết kế giao diện người dùng để hỗ trợ độ phân giải màn hình tối thiểu là 1280x720 pixel.
- Kích thước màn hình:** Giao diện người dùng được tối ưu hóa cho nhiều kích thước màn hình khác nhau.
- Hỗ trợ màu sắc:** Để nâng cao sức hấp dẫn trực quan và phù hợp với đối tượng rộng hơn, thiết kế giao diện người dùng của ứng dụng hỗ trợ tối thiểu 16 triệu màu.

#### c, Nguyên tắc thiết kế và tiêu chuẩn hóa

Để tạo trải nghiệm người dùng nhất quán và trực quan, chúng tôi đã thiết lập các nguyên tắc thiết kế và nguyên tắc tiêu chuẩn hóa cho các thành phần giao diện người dùng khác nhau. Những nguyên tắc này bao gồm:

- Thiết kế nút:** Tất cả các nút trong ứng dụng đều tuân theo một ngôn ngữ thiết

kết nối nhất, đảm bảo sự rõ ràng và dễ dàng nhận biết các yếu tố tương tác. Các nút có kích thước phù hợp, có nhãn rõ ràng và phản hồi trực quan khi tương tác.

- **Điều khiển:** Giao diện người dùng của chúng tôi kết hợp các điều khiển trực quan, chẳng hạn như thanh trượt, nút chuyển đổi và menu thả xuống, tuân thủ các nguyên tắc thiết kế dành riêng cho nền tảng. Điều này đảm bảo trải nghiệm người dùng quen thuộc trên các hệ điều hành khác nhau.
- **Hiển thị phản hồi:** Để cung cấp cho người dùng những phản hồi có ý nghĩa, tôi đã chuẩn hóa việc hiển thị tin nhắn và thông báo. Thông báo phản hồi được trình bày theo cách không xâm phạm, sử dụng màu sắc và vị trí thích hợp để dễ nhận thấy.
- **Bảng màu:** Chúng tôi đã lựa chọn cẩn thận bảng màu phản ánh thương hiệu của ứng dụng và đảm bảo giao diện đẹp mắt. Việc lựa chọn màu sắc tuân thủ các nguyên tắc trợ năng, có tính đến các yếu tố như độ tương phản và khả năng đọc.
- **Bố cục và Điều hướng:** Bố cục và điều hướng giao diện người dùng đã được tối ưu hóa để thúc đẩy luồng liền mạch và hợp lý trong toàn bộ ứng dụng. Các phần tử chung được đặt nhất quán và điều hướng được cấu trúc theo thứ bậc để đơn giản hóa tương tác của người dùng.

### d, Minh họa về thiết kế giao diện chính

Để trình bày trực quan về thiết kế giao diện người dùng của tôi, tôi trình bày một loạt hình ảnh minh họa giới thiệu giao diện của một số chức năng quan trọng nhất. Các hình minh họa thể hiện cách áp dụng các nguyên tắc thiết kế và tiêu chuẩn hóa để tạo ra trải nghiệm gắn kết và thân thiện với người dùng:



Hình 4.6: Nút đăng nhập và đăng ký

Hầu hết các nút trong dự án đều có mã màu di chuột là #da2461 và giao diện có thiết kế phẳng



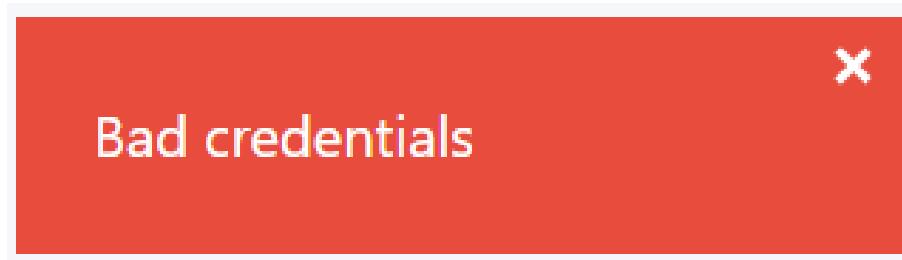
**Hình 4.7:** Phần header của trang web

Phần header chứa thanh điều hướng được cố định khi chúng ta cuộn (vị trí cố định). Nếu chúng ta đăng nhập thành công, thông báo sẽ hiện lên:



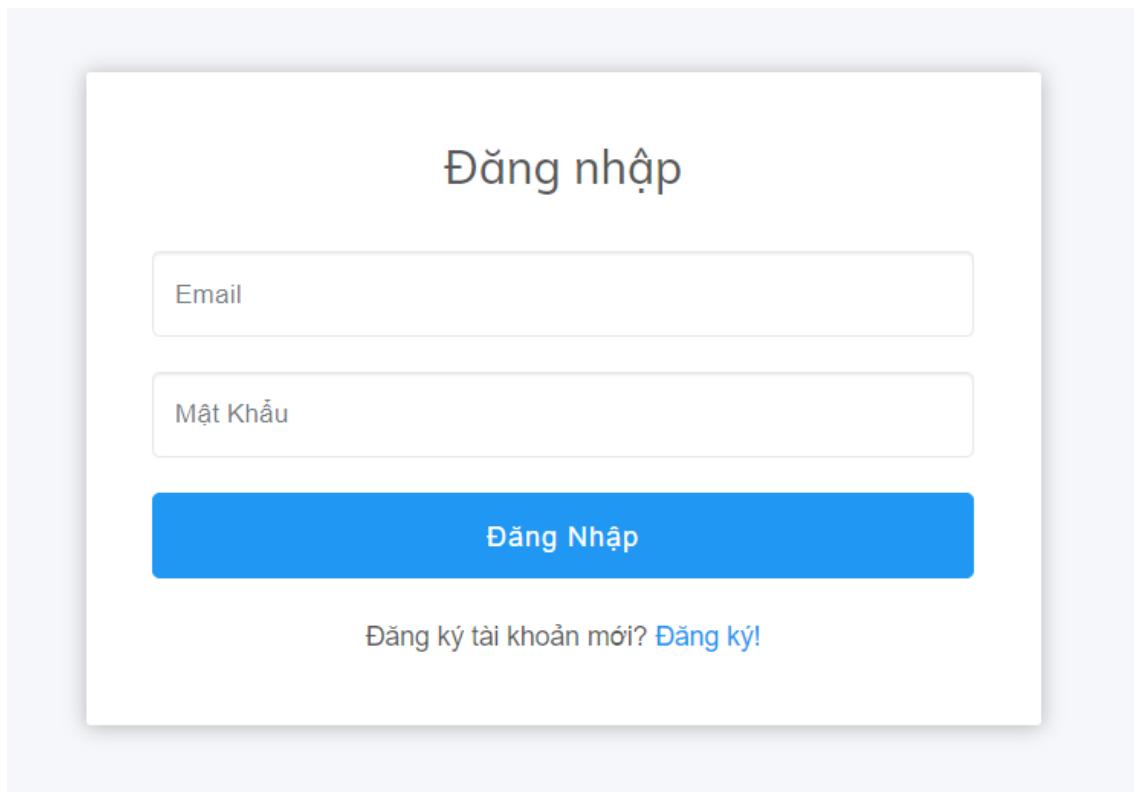
**Hình 4.8:** Thông báo đăng nhập thành công

Vì mô tả thành công nên thông báo có nền màu xanh lá cây. Ngược lại, nếu chúng ta đăng nhập thất bại, thông báo lỗi sẽ hiển thị:



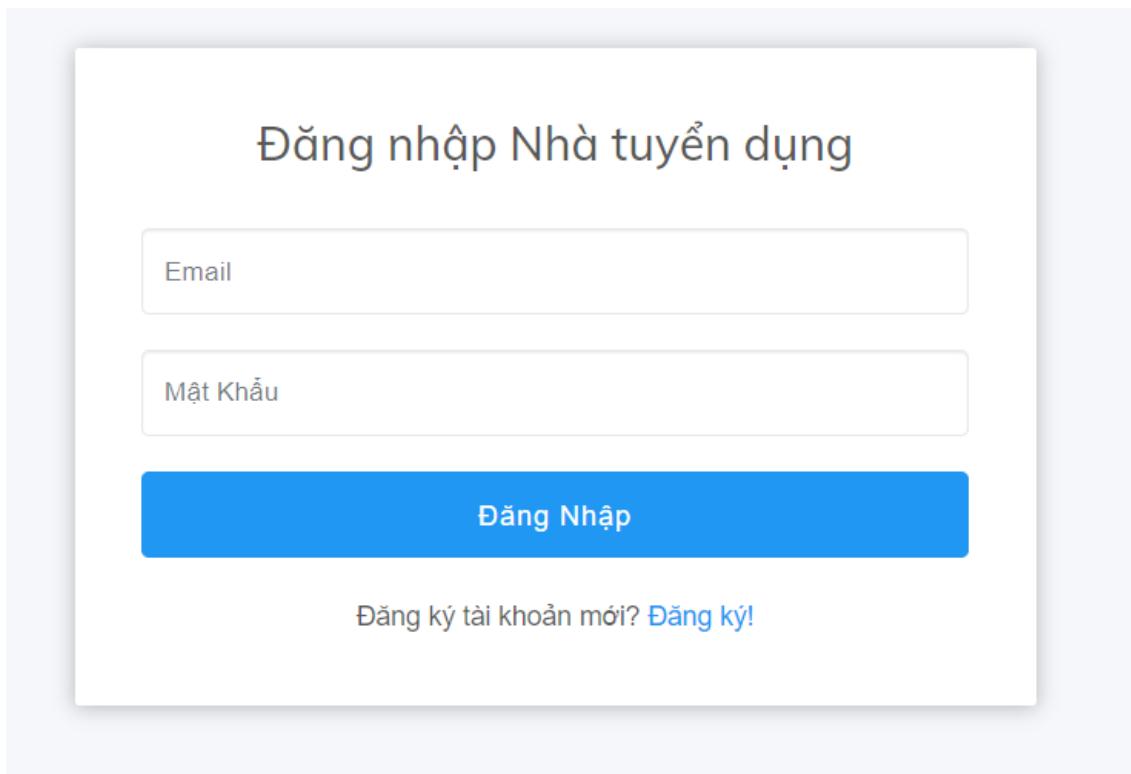
**Hình 4.9:** Thông báo đăng nhập thất bại

Trong ứng dụng này, có 2 biểu mẫu đăng nhập cho 2 vai trò khác nhau. Đối với người dùng bình thường, đây là biểu mẫu đăng nhập dành cho ứng viên và nhà tuyển dụng:



**Hình 4.10:** Đăng nhập dành cho ứng viên

Đối với người dùng là quản trị viên hoặc nhà tuyển dụng, biểu mẫu đăng nhập có cấu trúc giống nhau chỉ có các tiêu đề khác nhau:



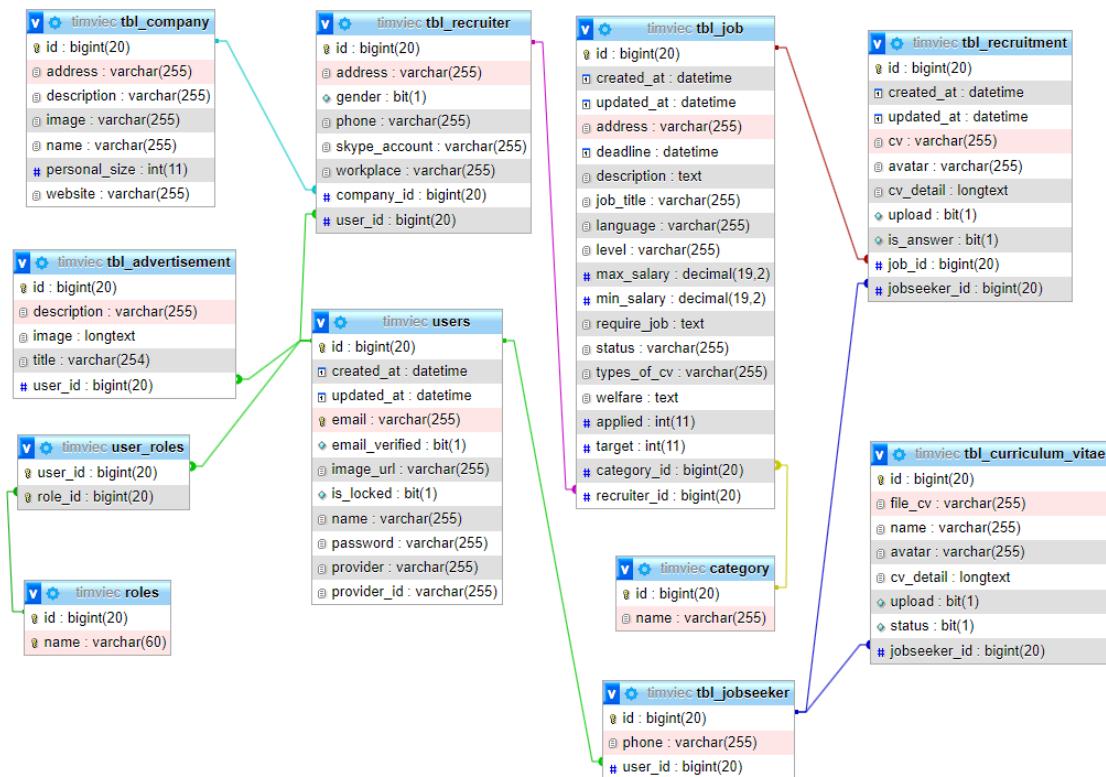
**Hình 4.11:** Đăng nhập dành cho Nhà tuyển dụng

Đối với quản trị viên, có một thanh điều hướng bên trái với các tính năng sau:



**Hình 4.12:** Thanh điều hướng của quản trị viên

### 4.2.2 Thiết kế database



Hình 4.13: Thiết kế database

Hình 4.13 mô tả sự liên kết giữa các thực thể, các bảng với nhau trong database. Các bảng từ bảng 4.1 đến bảng 4.11 lần lượt mô tả ý nghĩa và kiểu dữ liệu của các trường trong cơ sở dữ liệu.

| Trường dữ liệu | Kiểu dữ liệu | Mô tả                      |
|----------------|--------------|----------------------------|
| id             | Bigint       | id người dùng (khóa chính) |
| created_at     | datetime     | Thời gian tạo              |
| updated_at     | datetime     | Thời gian cập nhật         |
| email          | varchar      | Địa chỉ email              |
| email_verified | bit          | Trạng thái xác thực email  |
| image_url      | varchar      | Địa chỉ hình ảnh           |
| name           | varchar      | Tên người dùng             |
| password       | varchar      | Mật khẩu                   |
| provider       | varchar      | Nhà cung cấp xác thực      |
| provider_id    | varchar      | id nhà cung cấp            |
| is_locked      | bit          | Trạng thái khóa            |

Bảng 4.1: User

## CHƯƠNG 4. THIẾT KẾ, TRIỂN KHAI VÀ ĐÁNH GIÁ HỆ THỐNG

---

| Trường dữ liệu | Kiểu dữ liệu | Mô tả                   |
|----------------|--------------|-------------------------|
| id             | Bigint       | id công ty (khóa chính) |
| address        | varchar      | Địa chỉ                 |
| description    | varchar      | Mô tả                   |
| image          | varchar      | Địa chỉ hình ảnh        |
| name           | varchar      | Tên công ty             |
| personal_size  | int          | Số lượng nhân viên      |
| website        | varchar      | Trang web công ty       |

**Bảng 4.2:** Company

| Trường dữ liệu | Kiểu dữ liệu | Mô tả                     |
|----------------|--------------|---------------------------|
| id             | Bigint       | id ứng tuyển (khóa chính) |
| created_at     | varchar      | Thời gian tạo             |
| update_at      | varchar      | Thời gian cập nhật        |
| cv             | varchar      | Địa chỉ file CV           |
| avatar         | varchar      | Địa chỉ hình ảnh trong CV |
| cv_detail      | longtext     | Chi tiết CV               |
| upload         | bit          | Trạng thái CV             |
| job_id         | varchar      | Job ID                    |
| jobseeker_id   | varchar      | Jobseeker_id              |
| status         | int          | Trạng thái ứng tuyển      |

**Bảng 4.3:** Recruitment

| Trường dữ liệu | Kiểu dữ liệu | Mô tả                     |
|----------------|--------------|---------------------------|
| id             | Bigint       | id CV (khóa chính)        |
| file_cv        | varchar      | Địa chỉ file CV           |
| name           | varchar      | Tên CV                    |
| avatar         | varchar      | Địa chỉ hình ảnh trong CV |
| cv_detail      | longtext     | Chi tiết CV               |
| upload         | bit          | Loại CV(tải lên hay tạo)  |
| status         | varchar      | Trạng thái CV             |
| jobseeker_id   | varchar      | Jobseeker_id              |

**Bảng 4.4:** Curriculum Vitae

| Trường dữ liệu | Kiểu dữ liệu | Mô tả                     |
|----------------|--------------|---------------------------|
| id             | Bigint       | id quảng cáo (khóa chính) |
| image          | longtext     | Địa chỉ hình ảnh          |
| title          | varchar      | Tiêu đề quảng cáo         |
| user_id        | Bigint       | User ID                   |
| description    | varchar      | Mô tả                     |

**Bảng 4.5:** Advertisement

| Trường dữ liệu | Kiểu dữ liệu | Mô tả                    |
|----------------|--------------|--------------------------|
| id             | Bigint       | id ứng viên (khóa chính) |
| phone          | varchar      | Số điện thoại            |
| user_id        | Bigint       | User ID                  |

**Bảng 4.6:** Jobseeker

## CHƯƠNG 4. THIẾT KẾ, TRIỂN KHAI VÀ ĐÁNH GIÁ HỆ THỐNG

---

| Trường dữ liệu | Kiểu dữ liệu  | Mô tả                    |
|----------------|---------------|--------------------------|
| id             | Bigint        | id ứng viên (khóa chính) |
| created_at     | datetime      | Thời gian tạo            |
| updated_at     | datetime      | Thời gian cập nhật       |
| address        | varchar       | Địa chỉ                  |
| deadline       | datetime      | Hạn ứng tuyển            |
| description    | varchar       | Mô tả                    |
| job_title      | varchar       | Tiêu đề công việc        |
| language       | varchar       | Ngôn ngữ                 |
| level          | varchar       | Trình độ                 |
| max_salary     | decimal(19,2) | Lương tối đa             |
| min_salary     | decimal(19,2) | Lương tối thiểu          |
| require_job    | varchar       | Yêu cầu công việc        |
| status         | varchar       | Trạng thái công việc     |
| types_of_cv    | varchar       | Loại CV                  |
| welfare        | varchar       | Phúc lợi                 |
| applied        | int           | Đã ứng tuyển             |
| target         | int           | Mục tiêu                 |
| category_id    | Bigint        | Category ID              |
| recruiter_id   | Bigint        | Recruited ID             |

**Bảng 4.7:** Job

| Trường dữ liệu | Kiểu dữ liệu | Mô tả                    |
|----------------|--------------|--------------------------|
| id             | Bigint       | id ứng viên (khóa chính) |
| address        | varchar      | Địa chỉ                  |
| gender         | bit          | Giới tính                |
| phone          | varchar      | Số điện thoại            |
| skype_account  | varchar      | Tài khoản Skype          |
| workplace      | varchar      | Vị trí làm việc          |
| company_id     | Bigint       | Company ID               |
| user_id        | Bigint       | User ID                  |

**Bảng 4.8:** Recruiter

| Trường dữ liệu | Kiểu dữ liệu | Mô tả                    |
|----------------|--------------|--------------------------|
| id             | Bigint       | id danh mục (khóa chính) |
| name           | varchar      | tên danh mục             |

Bảng 4.9: Category

| Trường dữ liệu | Kiểu dữ liệu | Mô tả                   |
|----------------|--------------|-------------------------|
| id             | Bigint       | id vai trò (khóa chính) |
| name           | varchar      | Tên vai trò             |

Bảng 4.10: Roles

| Trường dữ liệu | Kiểu dữ liệu | Mô tả                |
|----------------|--------------|----------------------|
| user_id        | Bigint       | User ID (khóa chính) |
| role_id        | Bigint       | Role ID (khóa chính) |

Bảng 4.11: User Roles

### 4.3 Xây dựng ứng dụng

#### 4.3.1 Thư viện và công cụ sử dụng

| Mục đích                   | Công cụ                    | Địa chỉ URL   |
|----------------------------|----------------------------|---|
| IDE lập trình cho Backend  | IntelliJ IDEA 2022.1.2     | <a href="https://www.jetbrains.com/">https://www.jetbrains.com/</a>         |
| IDE lập trình cho Frontend | Visual studio code         | <a href="https://code.visualstudio.com/">https://code.visualstudio.com/</a> |
| Lập trình backend          | Springboot 2.7.14          | <a href="https://spring.io/">https://spring.io/</a>                         |
| Lập trình frontend         | React 17.0.0               | <a href="https://react.dev/">https://react.dev/</a>                         |
| Quản trị cơ sở dữ liệu     | Xampp control panel v3.3.0 | <a href="https://www.apachefriends.org/">https://www.apachefriends.org/</a> |

Bảng 4.12: Danh sách thư viện và công cụ sử dụng

#### 4.3.2 Minh họa các chức năng chính

Một số chức năng chính của đồ án:

## CHƯƠNG 4. THIẾT KẾ, TRIỂN KHAI VÀ ĐÁNH GIÁ HỆ THỐNG

Người dùng có thể lọc một số thuộc tính để có được công việc phù hợp.

The screenshot shows the TechHire platform interface. On the left, there is a sidebar with input fields for filtering: 'Tên công việc' (Job Title) with 'JAVA' typed in, 'Danh Mục' (Category) with a dropdown menu showing 'Chọn danh mục' (Select category) and a list of programming languages like JAVA, PYTHON, RUBY, PHP, C++, CSS, and JAVASCRIPT, 'Lương nhỏ nhất' (Lowest salary), and 'Lương cao nhất' (Highest salary). On the right, two job listings are displayed:

- Lập trình java** by **Ziggo** (ABC) in **KaengNam** at **1000\$ - 1500\$**. Status: **Senior**. Date: **Hạn: 25 - 11 - 2023**.
- Java Developer** by **Ziggo** (Viettel) in **Hà Nội** at **1000\$ - 1500\$**. Status: **Senior**. Date: **Hạn: 23 - 11 - 2023**.

**Hình 4.14:** Chức năng lọc công việc

Sau đó họ có thể xem chi tiết công việc và ứng tuyển.

The screenshot shows a detailed view of a job listing for a 'Lập trình viên Reactjs' at 'Số 1 Đại Cồ Việt, Hà Nội' with a salary range of '1000\$ - 1500\$'. The listing includes a 'Mô Tả Công Việc:' (Job Description) stating 'Phát triển frontend cho nền tảng thương mại điện tử', a 'Yêu Cầu:' (Requirements) section with 'Nắm rõ về CSS, HTML, JAVASCRIPT', a 'Phúc Lợi:' (Benefits) section with 'Có bảo hiểm đầy đủ', and a 'Công việc:' (Work) section with details: 'Địa chỉ: Số 1 Đại Cồ Việt, Hà Nội', 'Level: Junior', 'Lương: 1500\$', and 'Hạn ứng tuyển: 27 - 12 - 2023'. A large red 'Ứng Tuyển' (Apply) button is prominently displayed. To the right, there is a 'Thông tin công ty' (Company Information) section with 'Tập đoàn Công nghiệp - Viễn thông Quân đội (VIETTEL)' and the Viettel Media logo.

**Hình 4.15:** Chức năng lọc công việc và ứng tuyển

## CHƯƠNG 4. THIẾT KẾ, TRIỂN KHAI VÀ ĐÁNH GIÁ HỆ THỐNG

Các ứng viên còn có thể tạo CV ngay trực tiếp trên ứng dụng.

Dashboard

Tải lên CV

Tên CV

CV\_VuBaLuong

Họ và tên

Vũ Bá Lương

Ngày sinh

23/12/2001

Giới tính

Nam

Email

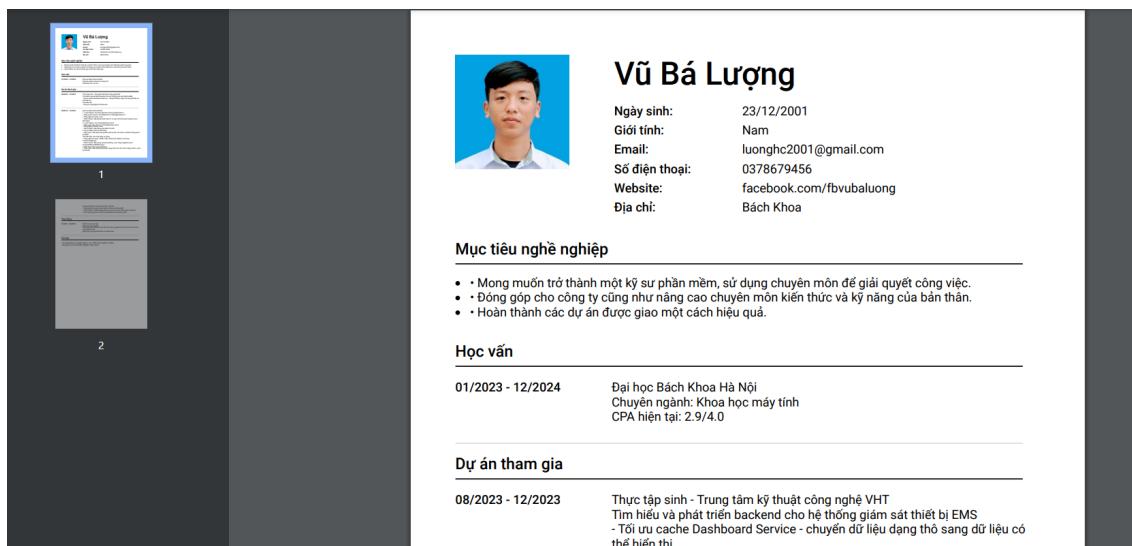
luonghc2001@gmail.com

Số điện thoại

0378679456

**Hình 4.16:** Chức năng tạo CV

Sau đó ứng viên có thể xem trước CV vừa tạo.



**Hình 4.17:** Chức năng xem trước CV

### 4.4 Triển khai

Phần mềm ứng dụng được triển khai và chạy thử trên máy tính cá nhân và đã chạy được các tính năng cơ bản đã được thiết kế ở trên.

## CHƯƠNG 5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

### 5.1 Kết luận

Trong đồ án tốt nghiệp, chúng tôi đã xây dựng và phát triển nền tảng đăng và tìm kiếm việc làm dành cho các nhà tuyển dụng và người tìm việc có thể kết nối với nhau dễ dàng hơn. Để phát triển nền tảng này chúng tôi đã áp dụng mô hình MVC sử dụng Spring Boot và ReactJs để xây dựng ứng dụng web.

Kết quả chúng tôi xây dựng được là ứng dụng web đăng và tìm kiếm việc làm. Giao diện được xây dựng khá đầy đủ, tương ứng với từng đối tượng người dùng với menu riêng, chứa đựng các khả năng có thể thực hiện được. Đối với người dùng là quản trị viên, có thể thực hiện các chức năng sau: Đăng nhập vào hệ thống (với quyền quản trị), cập nhật thông tin cá nhân, quản lý thông tin tuyển dụng, quản lý tài khoản của ứng viên và nhà tuyển dụng, quản lý quảng cáo và danh mục. Đối với các nhà tuyển dụng, các chức năng sau có thể được thực hiện: Đăng ký, đăng nhập hệ thống, quản lý thông tin tuyển dụng, quản lý ứng viên đã nộp đơn, xem hồ sơ của ứng viên, liên lạc với ứng viên qua mail, cập nhật thông tin nhà tuyển dụng. Đối với các ứng viên, các chức năng sau có thể được thực hiện: Đăng ký, đăng nhập hệ thống, tìm kiếm việc làm theo danh mục hoặc theo từ khóa, cập nhật CV của bạn, nộp đơn ứng tuyển, xem thông tin công việc, cập nhật thông tin nhà tuyển dụng.

Trong quá trình thực hiện đồ án tốt nghiệp, tôi đã rèn luyện được nhiều kiến thức và kỹ năng quan trọng, bao gồm kỹ năng nghiên cứu, tìm kiếm thông tin, quản lý thời gian và kỹ năng trình bày. Đó là những kỹ năng rất quan trọng cho quá trình phát triển sự nghiệp của tôi sau này.

### 5.2 Hướng phát triển

Chúng tôi đề xuất một số hướng phát triển cho đồ án bao gồm:

- Phát triển thêm chức năng đăng ký thành viên tuyển dụng trả phí trực tuyến.
- Mỗi hồ sơ phát triển cho nhiều ngành nghề.
- Thực hiện thêm chức năng tìm kiếm việc làm tự động phù hợp với hồ sơ ứng viên.
- Tạo thêm module đa ngôn ngữ.
- Thực hiện Bảo vệ DDoS: Để bảo vệ trang web khỏi các cuộc tấn công từ chối dịch vụ phân tán (DDoS), tích hợp các biện pháp bảo vệ DDoS. Điều này có thể bao gồm hợp tác với một dịch vụ bảo vệ DDoS đáng tin cậy hoặc triển khai các chiến lược giảm nhẹ như giới hạn tỷ lệ, lọc lưu lượng và cân bằng tải.

## CHƯƠNG 5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

- Bảo mật dữ liệu người dùng: Tăng cường bảo mật dữ liệu người dùng bằng cách sử dụng các giao thức mã hóa (ví dụ: SSL/TLS) để truyền và lưu trữ dữ liệu.

## TÀI LIỆU THAM KHẢO

- [1] *Mysql*. [Online]. Available: <https://www.mysql.com/> (visited on 01/01/2024).
- [2] Oracle, *Java*. [Online]. Available: <https://www.oracle.com/java/> (visited on 01/01/2024).
- [3] *Spring boot*. [Online]. Available: <https://spring.io/projects/spring-boot> (visited on 02/01/2024).
- [4] M. Platforms, *React*. [Online]. Available: <https://react.dev/> (visited on 02/01/2024).
- [5] W3Schools, *Html tutorial*. [Online]. Available: <https://www.w3schools.com/html/default.asp> (visited on 03/01/2024).
- [6] W3Schools, *Css tutorial*. [Online]. Available: <https://www.w3schools.com/css/default.asp> (visited on 03/01/2024).