

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
ĐỒ ÁN TỐT NGHIỆP

Trang web nhắn tin trực tuyến

Đỗ Thái Sơn

son.dt187273@sis.hust.edu.vn

Ngành Công nghệ thông tin

Giảng viên hướng dẫn: TS. Hoàng Văn Hiệp

Chữ kí GVHD

Khoa: Công nghệ thông tin

Trường: Công nghệ thông tin và Truyền thông

HÀ NỘI, 08/2022

LỜI CẢM ƠN

Đầu tiên, em xin cảm ơn thầy TS. Hoàng Văn Hiệp là người đã trực tiếp hướng dẫn và giúp đỡ em hoàn thiện đồ án tốt nghiệp này.

Tiếp theo, em muốn gửi lời cảm ơn đến nhà trường, các giảng viên cùng các bạn học cùng lớp đã tận tình hướng dẫn, giúp đỡ trong những năm học qua.

Cuối cùng con xin cảm ơn gia đình đã luôn là nguồn động lực để con tiến bước trên con đường đại học và ra ngoài xã hội khởi đầu với đồ án tốt nghiệp này.

i

TÓM TẮT NỘI DUNG ĐỒ ÁN

Ngày nay ứng dụng công nghệ thông tin vào việc tin học hóa được xem là một trong những yếu tố quyết định trong hoạt động của các chính phủ, tổ chức, cũng như các công ty; nó đóng vai trò hết sức quan trọng, có thể tạo ra những bước đột phá mạnh mẽ. Với cuộc cách mạng khoa học công nghệ 4.0, ứng dụng nhắn tin đang chiếm lĩnh các kênh thông tin, giải trí, công việc, giao lưu không chỉ của của giới trẻ. Ứng dụng nhắn tin là một trang web hay nền tảng trực tuyến với nhiều hình thức, tính năng, giúp mọi người dễ dàng kết nối với nhau ở bất cứ nơi nào. Ở đó, không chỉ có các mối quan hệ ảo của những người cùng đam mê, sở thích... mà còn có cả những mối

quan hệ công việc, đa ngành nghề. Việc sử dụng mạng xã hội mang lại nhiều lợi ích cho người dùng nói riêng và cho cộng đồng nói chung

Chính vì vậy, đề án này hướng tới một hệ thống nhắn tin trực tuyến, trong đó tập trung hơn đến quy trình soạn thảo, gửi, hiển thị tin nhắn. Để thực hiện được điều đó, hệ thống sẽ được xây dựng sử dụng nền tảng Web để phát triển, bao gồm những chức năng cơ bản và tập trung vào quy trình xử lý tin nhắn.

Đóng góp chính của đề án này là xây dựng một trang web nhắn tin được sử dụng với mục đích cụ thể cho công việc, học tập,... với đầy đủ các chức năng yêu cầu cơ bản của một ứng dụng nhắn tin trực tuyến như: kết bạn, nhắn tin 2 người, tạo nhóm nhắn nhiều người, tin nhắn nháp, tương tác tin nhắn, theo dõi luồng tin nhắn, thông báo,... Ứng dụng nhắn tin của em hướng tới mọi nhóm người dùng với các mục đích sử dụng khác nhau. Ứng dụng nhắn tin được xây dựng cho mục đích giải trí, công việc, chia sẻ thông tin, kiến thức; giao lưu, gắn kết mọi người với nhau.

ii

MỤC LỤC

CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI 1

1. Đặt vấn đề 1
2. Mục tiêu và phạm vi đề tài 2
3. Định hướng giải pháp 3
4. Bố cục đề án 3

CHƯƠNG 2. KHẢO SÁT VÀ PHÂN TÍCH YÊU CẦU 5

1. Tổng quan 5
2. Khảo sát hiện trạng 5
3. Tổng quan chức năng 7
 1. Biểu đồ use case tổng quát 7
 2. Biểu đồ use case phân rõ tìm việc làm 8
 3. Biểu đồ use case phân rõ quản lý đơn ứng tuyển 9
 4. Biểu đồ use case phân rõ tìm ứng viên 9
 5. Biểu đồ use case phân rõ quản lý việc làm 10
 6. Biểu đồ use case phân rõ quản lý nhân viên 10
 7. Biểu đồ use case phân rõ xem danh sách đơn ứng tuyển đang tham

gia 11

8. Quy trình nghiệp vụ 11
4. Đặc tả chức năng 13
 1. Đặc tả use case “Đăng nhập” 13
 2. Đặc tả use case “Ứng viên ứng tuyển” 14
 3. Đặc tả use case “Nhân viên nhận hồ sơ” 15
 4. Đặc tả use case “Nhân viên quản lý danh sách việc làm” 15
5. Yêu cầu phi chức năng 17
 1. Tính an toàn 17
 2. Tính bảo mật 17
 3. Giao diện 17
 4. Một số đặc tính cần có của sản phẩm 17
 5. Hiệu năng 17
 6. Các yêu cầu khác 17
6. Kết chương 18

CHƯƠNG 3. CÔNG NGHỆ SỬ DỤNG 19

1. Tổng quan 19
2. Công nghệ lưu trữ dữ liệu 19
 1. Hệ quản trị cơ sở dữ liệu quan hệ 19
 2. Hệ quản trị cơ sở dữ liệu MySQL 20
3. Ngôn ngữ lập trình và framework sử dụng 21
 1. Lập trình hướng đối tượng 21
 2. Ngôn ngữ lập trình: Ruby 23
 3. Framework sử dụng: Ruby on Rails 24
4. Công cụ xử lý tác vụ ngầm 25
 1. Khái niệm tác vụ ngầm 25
 2. Công cụ xử lý tác vụ ngầm: Sidekiq 26
 3. Công nghệ lưu trữ các tác vụ ngầm: Redis 27
5. Kết chương 28

CHƯƠNG 4. PHÂN TÍCH, THIẾT KẾ VÀ XÂY DỰNG HỆ THỐNG 29

1. Tổng quan 29
2. Thiết kế kiến trúc 29
 1. Lựa chọn kiến trúc phần mềm 29

iv

2. Thiết kế tổng quan. 32
3. Thiết kế chi tiết gói 33
3. Thiết kế chi tiết 36
 1. Thiết kế giao diện 36
 2. Thiết kế lớp 38
 3. Thiết kế cơ sở dữ liệu 41
4. Xây dựng ứng dụng 45
 1. Thư viện và công cụ sử dụng 45
 2. Kết quả đạt được 46
 3. Minh họa các chức năng chính 47
5. Kiểm thử 48
 1. Kiểm thử chức năng đăng nhập 49
 2. Kiểm thử chức năng cập nhật thông tin việc làm 49
 3. Kiểm thử chức năng nộp đơn ứng tuyển cho ứng viên 50
 4. Tổng kết phần kiểm thử chức năng 50
6. Triển khai 51
7. Kết chương 51

CHƯƠNG 5. CÁC GIẢI PHÁP VÀ ĐÓNG GÓP NỔI BẬT 52

1. Tổng quan 52
2. Hỗ trợ quá trình ứng tuyển 52
 1. Giới thiệu vấn đề 52
 2. Giải pháp xây dựng 52
 3. Kết quả thu được 57
3. Kết chương 59

CHƯƠNG 6. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 60

1. Kết luận 60

2. Hướng phát triển 61

TÀI LIỆU THAM KHẢO 63

PHỤ LỤC 65

A. ĐẶC TẢ USE CASE 65

1. Đặc tả use case “Quản lý quản lý danh sách nhân viên” 65

2. Đặc tả use case “Đăng ký tài khoản ứng viên” 66

DANH MỤC HÌNH VẼ

| | | |
|-------------|---|----|
| Hình 2.1 | Biểu đồ usecase tổng quát | 7 |
| | Use case phân rã tìm việc làm | 8 |
| Hình 2.2 | | |
| | Use case phân rã quản lý đơn ứng tuyển | 9 |
| Hình 2.3 | | |
| | Use case phân rã tìm ứng viên | 9 |
| Hình 2.4 | | |
| | Use case phân rã quản lý việc làm | 10 |
| Hình 2.5 | | |
| | Use case phân rã quản lý nhân viên | 10 |
| Hình 2.6 | | |
| | Use case phân rã xem danh sách đơn ứng tuyển đang tham gia | 11 |
| Hình 2.7 | Biểu đồ hoạt động cho quy trình ứng tuyển | 12 |
| Hình 2.8 | | |
| | Logo của MySQL1 | 20 |
| Hình 3.1 | | |
| | Logo của ngôn ngữ lập trình Ruby2 | 23 |
| Hình 3.2 | | |
| | Logo của Ruby on Rails3 | 24 |
| Hình 3.3 | | |
| | Logo của Sidekiq4 | 26 |
| Hình 3.4 | | |

| | | |
|----------|---|----|
| Hình 2.1 | Biểu đồ usecase tổng quát | 7 |
| | Logo của Redis5 | 27 |
| Hình 3.5 | Mô hình kiến trúc MVC | 29 |
| Hình 4.1 | Biểu đồ gói tổng quan | 32 |
| Hình 4.2 | Biểu đồ gói chi tiết cho gói View | 33 |
| Hình 4.3 | Biểu đồ gói Model chi tiết | 34 |
| Hình 4.4 | Biểu đồ gói Model chi tiết | 35 |
| Hình 4.5 | Minh họa giao diện trang chủ phần tìm kiếm | 37 |
| Hình 4.6 | Minh họa giao diện trang việc làm | 37 |
| Hình 4.7 | Minh họa giao diện trang biểu mẫu | 38 |
| Hình 4.8 | Biểu đồ lớp thiết kế chi tiết cho lớp Employee và LoginCon- | |
| Hình 4.9 | | |

troller 38

Hình 4.10 Biểu đồ trình tự mô tả use case đăng nhập đối với nhân viên . 39
 Hình 4.11 Biểu đồ lớp thiết kế chi tiết cho lớp Apply và AppliesController 40
 Hình 4.12 Biểu đồ trình tự mô tả use case nộp đơn ứng tuyển 41

Hình 4.13 Sơ đồ thực thể liên kết 42

Hình 4.14 Thiết kế cơ sở dữ liệu cho hệ thống 43

Hình 4.15 Hình ảnh trang chủ với danh sách việc làm theo chủ đề 47

Hình 4.16 Hình ảnh trang việc làm 48

Hình 4.17 Hình ảnh trang chỉnh sửa thông tin việc làm 48

Hình 5.1 Sơ đồ trạng thái Giai đoạn mở đầu của đơn ứng tuyển 53

Hình 5.2 Sơ đồ trạng thái Giai đoạn sơ tuyển của đơn ứng tuyển 54

Hình 5.3 Sơ đồ trạng thái Giai đoạn phỏng vấn của đơn ứng tuyển .

. . . 55 Hình 5.4 Sơ đồ trạng thái Giai đoạn chấp nhận của đơn ứng
tuyển . . . 56 Hình 5.5 Giao diện thay đổi nội dung biểu mẫu đơn
ứng tuyển 58

Hình 5.6 Giao diện cập nhật đơn ứng tuyển của ứng viên tại trạng
thái

"Bổ sung hồ sơ" 58

Hình 5.7 Giao diện đơn ứng tuyển sau khi cập nhật và có thông báo
cho ứng viên 59

viii

DANH MỤC BẢNG BIỂU

| | | |
|-------------|--|----|
| Bảng 2.1 | Đặc tả use case đăng nhập | 13 |
| Bảng 2.2 | Đặc tả use case ứng viên ứng tuyển | 14 |
| Bảng 2.3 | Đặc tả use case nhân viên nhận hồ sơ | 15 |
| Bảng 2.4 | Đặc tả use case nhân viên quản lý danh sách việc làm | 16 |
| Bảng 4.1 | Đặc tả thông tin về màn hình | 36 |
| Bảng 4.2 | Danh sách thư viên và công cụ sử dụng | 45 |
| Bảng 4.3 | Thông tin thống kê về ứng dụng | 47 |
| Bảng A.1 | Đặc tả use case nhân viên quản lý danh sách nhân viên | 65 |
| Bảng A.2 | Đặc tả use case ứng viên đăng ký tài khoản | 66 |

DANH MỤC THUẬT NGỮ VÀ TỪ VIẾT TẮT

| Viết tắt | Tên tiếng Anh | Tên tiếng Việt |
|---------------------|----------------------------|--|
| | Internet | Mạng máy tính toàn cầu |
| | Framework | Phần mã nguồn được viết sẵn thành một khung dùng để làm thư viện |
| | E-mail | Thư điện tử |
| CV | Curriculum Vitae | Hồ sơ ứng tuyển |
| | Use case | Cả sử dụng |
| UC | Relational | Hệ quản trị cơ sở dữ liệu quan hệ |
| RDBMS | Database Management System | |
| | Structured Query Language | Ngôn ngữ truy vấn có cấu trúc |
| SQL | Object-Oriented | Lập trình hướng đối tượng |
| OOP | Programming | Đến trước sẽ được giải quyết trước |
| | First In First Out | Ngôn ngữ đánh dấu siêu văn bản |
| FIFO | HyperText Markup Language | Công nghệ thông tin |
| HTML | | |
| CNTT | | Đồ án tốt nghiệp |
| DATN | | Sinh viên |
| SV | | |

x

CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI

Đặt vấn đề

Vấn đề chia sẻ trao đổi thông tin, tài liệu từ trước đến nay luôn là một vấn đề quan trọng có ảnh hưởng lớn đến sự phát triển từ các đơn vị lớn như tập đoàn, các cơ quan hành chính nhà nước tới các đơn vị nhỏ như mô hình kinh doanh hộ gia đình, các doanh nghiệp vừa và nhỏ. Việc làm thế nào để người lao động có thể tiếp cận được với các thông tin, tài liệu có liên quan đến bản thân, công việc của mình cũng như tự chia sẻ thông tin, tài liệu, đề xuất với người khác một cách nhanh, chính xác đã trở thành một vấn đề quan trọng cần phải giải quyết.

Ban đầu vấn đề này được giải quyết bằng các phương pháp truyền

thống như gặp mặt trực tiếp, tổ chức các cuộc họp nhóm, . . . Trong quá trình phát triển của công nghệ, các hình thức khác bắt đầu phát triển như gọi điện trực tiếp hay gửi tin nhắn sms. Tuy nhiên cách làm này vẫn còn nhiều hạn chế: chi phí cao, không hiệu quả trong việc chia sẻ thông tin hướng tài liệu phức tạp, khó khăn trong giao tiếp nhóm nhiều người

Trong thời gian gần đây, nhờ sự phát triển của nền kinh tế kèm theo sự phát triển của khoa học kỹ thuật, rất nhiều các doanh nghiệp mới được thành lập cùng với đó là bùng nổ về số lượng thông tin, dữ liệu là vô cùng lớn. Từ đó nhu cầu về việc trao đổi thông tin, thảo luận trong các dự án, làm việc nhóm trở nên vô cùng quan trọng và cần thiết đối với sự thành công của một dự án. Việc máy tính được phát triển và được sử dụng rộng rãi trong xã hội cùng với sự phổ cập của mạng Internet toàn cầu đã tạo điều kiện cho một phương pháp mới làm trung gian trao đổi, thảo luận thông tin giữa người với người trong các tổ chức, nhóm, doanh nghiệp . Vì vậy việc trao đổi thông tin trực tuyến trên mạng Internet trở thành mục tiêu cần phải hướng tới.

Việc chuyển sang trao đổi thông tin trên mạng Internet đem lại nhiều lợi ích cho con người. Tận dụng được sự phổ cập của mạng Internet hiện nay, người dùng có thể tiếp cận được các thông tin công việc, học tập liên quan tới bản thân ngay tại nhà riêng hoặc ở bất cứ đâu có kết nối mạng Internet, tiết kiệm được nhiều chi phí, công sức. Bên cạnh đó, các doanh nghiệp, tổ chức, đoàn thể có thể tạo một kênh thông tin chung, riêng biệt cho các thành viên, nhân viên trong nhóm, tổ chức của mình để các cá nhân có thể trao đổi, chia sẻ thông tin phục vụ cho mục đích riêng. Lợi ích mà kênh trao đổi chung, trực tuyến giữa nhiều người với nhau được thiết lập là vô cùng lớn. Do đó việc tạo ra môi trường trao đổi thông tin trực tuyến giữa mọi người với nhau là vấn đề cần thiết.

2. MỤC TIÊU VÀ PHẠM VI ĐỀ TÀI

Mục tiêu và phạm vi đề tài

Hiện nay, trên nền tảng Internet đã có nhiều sản phẩm, trang web hỗ trợ tìm kiếm việc làm, đăng tin tuyển dụng, môi giới việc làm giữa hai bên người tìm việc và nhà tuyển dụng. Một số trang web nổi bật như topcv.vn do Công ty cổ phần TopCV Việt Nam quản lý, có các chức năng tìm kiếm việc làm theo nhu cầu, hay các việc làm nổi bật, kèm theo đó là các chức năng phục vụ cho việc đăng tin tuyển dụng và tìm hồ sơ ứng viên theo mong muốn của bên nhà tuyển dụng. Một trang web khác là CareerLink.vn do công ty CareerLink quản lý và phát triển bao gồm các chức năng tương tự như topcv.vn nhưng trong đó nổi bật có tính năng tìm kiếm nhân sự cao cấp hướng tới

các đối tượng như các cấp lãnh đạo, chuyên môn cao thường nắm các vị trí chủ chốt trong công ty cần tuyển nhân sự.

Đặc điểm chung của các trang web này là đã làm tốt các công việc về đăng tin tuyển dụng, tìm kiếm công việc phù hợp theo các tiêu chí. Ngoài ra các bên triển khai các trang web trên cũng hợp tác với nhiều doanh nghiệp trong và ngoài nước để có được những thông tin tuyển dụng sớm và phù hợp với nhu cầu của người dùng. Các trang web đều có giao diện sáng sủa, dễ dùng, có hướng dẫn rõ ràng cho người mới sử dụng. Các tính năng được phát triển đầy đủ, trong đó phần quy trình đăng kí sử dụng cho bên nhà tuyển dụng được công khai rõ ràng. Các trang web cũng được quản lý bởi các công ty có đội ngũ nhân viên nhiều kinh nghiệm trong quản lý nhân sự giúp việc phân tích và đánh giá các nhà tuyển dụng và người tìm việc đạt hiệu quả hơn.

Tuy nhiên, điểm chung mà các sản phẩm trên đều chưa hoàn thiện hoặc chưa đầy đủ là trong quá trình nộp đơn ứng tuyển. Khi người dùng ứng tuyển, họ sẽ nộp đơn ứng tuyển trên trang tuyển dụng công việc kèm theo các giấy tờ liên quan. Tuy nhiên các trang web trên hầu như chỉ hỗ trợ đến bước này, và tại các bước sau phía nhà tuyển dụng sẽ trực tiếp trao đổi với người tìm việc thông qua các kênh liên lạc khác như e-mail, gọi điện thoại, . . .

Vì vậy, mục tiêu của đề án này sẽ hướng tới việc tạo ra một sản phẩm mới giới việc làm giữa hai bên người tìm việc và nhà tuyển dụng, trong đó xây dựng ra một sản phẩm hỗ trợ quy trình tuyển dụng hoàn chỉnh từ khi người tìm việc nộp hồ sơ cho đến khi đơn ứng tuyển đó được chấp nhận hoặc bị từ chối. Và qua đó khắc phục được hạn chế giao tiếp giữa hai bên và không cần một kênh thứ ba bên ngoài để trao đổi thêm khi chuyển trạng thái trong quá trình xét duyệt hồ sơ tuyển dụng.

Vì vậy sản phẩm được phát triển bao gồm các chức năng hỗ trợ cho quá trình người tìm việc tìm kiếm công việc phù hợp, bên nhà tuyển dụng đăng các thông tin tuyển dụng và chức năng chính sẽ tập trung vào hỗ trợ quy trình tuyển dụng và

2

3. ĐỊNH HƯỚNG GIẢI PHÁP

chuyển trạng thái đơn ứng tuyển trong quá trình xét duyệt đơn ứng tuyển. Sản phẩm được hỗ trợ thêm một vài tính năng bao gồm gợi ý công việc theo mong muốn của người tìm việc, và tìm kiếm ứng viên phù hợp cho nhà tuyển dụng đối với các công việc được đăng tin.

Định hướng giải pháp

Từ vấn đề đã nêu trên, em hướng tới việc sử dụng công nghệ web là công nghệ cốt lõi để phát triển sản phẩm trong đề án này. Các công nghệ được sử dụng bao gồm nền tảng web là nơi sản phẩm được triển khai vì mục tiêu hướng tới một sản phẩm hoạt động trên mạng Internet. Cơ sở dữ liệu quan hệ được sử dụng để đảm bảo tính toàn vẹn của dữ liệu có tính ràng buộc cao trong quá trình sử dụng và vận hành hệ thống. Cơ sở dữ liệu phi quan hệ được sử dụng để lưu trữ các dữ liệu rời rạc, số lượng lớn và cần truy vấn nhanh.

Kiến trúc được áp dụng để xây dựng hệ thống là kiến trúc Microservice để đảm bảo phân tách vai trò các thành phần trong hệ thống được triển khai một cách rõ ràng, phù hợp với các ứng dụng trên nền tảng web. Mô hình Microservice phân chia phần mềm thành nhiều service nhỏ tồn tại độc lập, điều này đồng nghĩa với việc mọi hoạt động xử lý, lưu trữ và yêu cầu dữ liệu đều riêng biệt. Những ưu điểm của mô hình Microservice:

- +) Hoạt động độc lập, linh hoạt, có tính chuyên biệt cao: Do không bị ràng buộc bởi các yêu cầu chung, nên mỗi service nhỏ có thể tự do lựa chọn công nghệ, nền tảng phù hợp
- +) Thuận tiện trong nâng cấp, mở rộng, xử lý lỗi: Việc xử lý lỗi, nâng cấp, bảo trì service hoàn toàn độc lập sẽ không làm gián đoạn quá trình vận hành của cả hệ thống. Nhờ vậy, những phiên bản mới có thể được cập nhật thường xuyên
- +) Đơn giản hóa trong quá trình kiểm thử: Với các service nhỏ, việc quản lý, tính toán, quản lý sẽ đơn giản, nhanh chóng hơn so với cả phần mềm

Từ những công nghệ đó, sản phẩm được hướng tới là một trang web nhấn tin trực tuyến cho phép người dùng chia sẻ, trao đổi, thu nhận thông tin, tài liệu từ các người dùng khác nhanh, hiệu quả. Sản phẩm cần phải đảm bảo được tính toàn vẹn dữ liệu trong quá trình vận hành.

Đóng góp chính của đề án này là tập trung vào xây dựng một sản phẩm hỗ trợ người dùng trao đổi, chia sẻ thông tin hướng văn bản nhanh, chính xác và hiệu quả nhất. Kết quả đạt được là một trang web nhấn tin trực tuyến trong đó nội dung xử lý, soạn thảo, hiển thị tin nhắn sẽ là trọng tâm chính của sản phẩm. Cùng với đó sẽ bao gồm một số tính năng cần thiết đối với một trang web nhấn tin trực tuyến

Bổ cục đề án

Phần còn lại của báo cáo đề án tốt nghiệp này được tổ chức như sau.

Chương 2 trình bày về hiện trạng hiện tại của các sản phẩm có nội dung tương tự với báo cáo đề án này, phân tích các sản phẩm đó và đưa ra phần cải tiến để giải quyết vấn đề và hạn chế liên quan tới quy trình nghiệp vụ trong sản phẩm. Tiếp theo đến phần tổng quan các yêu cầu chức năng của sản phẩm được thiết kế, trong đó trình bày các biểu đồ use case tổng quát và phân mức mô tả các chức năng chính của sản phẩm kèm theo quy trình nghiệp vụ của sản phẩm. Kế đến là phần nội dung đặc tả các chức năng chính và các yêu cầu phi chức năng cho sản phẩm được phát triển.

Trong chương 3, các công nghệ và kỹ thuật được sử dụng trong việc tạo ra sản

3

4. BỐ CỤC ĐỀ ÁN

phẩm được đưa ra và phân tích ở đây. Trong đó bao gồm công nghệ về lưu trữ dữ liệu, các công cụ dùng để phát triển sản phẩm như ngôn ngữ lập trình, các framework hỗ trợ phát triển kèm theo ưu nhược điểm của từng công nghệ và lý do lựa chọn các công nghệ này trong quá trình phát triển sản phẩm.

Kế tiếp là chương 4 đề cập tới quá trình xây dựng phần mềm. Các nội dung được đề cập trong chương này bao gồm việc thiết kế về mặt kiến trúc sản phẩm, thiết kế tổng quan các thành phần chức năng. Kế đến là nội dung thiết kế chi tiết các bộ phận của sản phẩm gồm thiết kế giao diện, thiết kế lớp và cơ sở dữ liệu. Cuối cùng đề cập tới phần xây dựng sản phẩm cùng với kiểm thử và cách thức triển khai sản phẩm.

Chương 5 nêu ra các đóng góp chính của đề tài, trọng tâm là đóng góp việc xây dựng ra tính năng hỗ trợ quy trình tuyển dụng hoàn chỉnh cho sản phẩm.

Chương kế tiếp, chương 6 nêu các kết luận trong quá trình thực hiện đề án, bao gồm các bài học kinh nghiệm, phân tích và so sánh sản phẩm đạt được. Cùng với đó là hướng phát triển tương lai cho sản phẩm.

Phần phụ lục cuối đề án tốt nghiệp trình bày thêm đặc tả của một số use case của hệ thống.

4

CHƯƠNG 2. KHẢO SÁT VÀ PHÂN TÍCH YÊU CẦU

Tổng quan

Tiếp theo chương giới thiệu về đề tài, trong chương này em sẽ trình bày chi tiết hơn về các nội dung liên quan tới khảo sát đề tài. Kế tiếp đó là nội dung tổng quan về các chức năng chính của sản phẩm

được phát triển, kèm theo quy trình nghiệp vụ quan trọng. Cuối cùng bao gồm đặc tả một số chức năng quan trọng cùng với đó là các yêu cầu phi chức năng đối với sản phẩm cần đạt được.

Khảo sát hiện trạng

Hiện nay trên nền tảng Internet đã có nhiều sản phẩm, nhiều giải pháp để giải quyết vấn đề chia sẻ, trao đổi thông tin trực tuyến. Các sản phẩm này đều đã tạo được nhiều tính năng tiện ích cho nhiều đối tượng người dùng khác nhau. Một số sản phẩm nổi bật sẽ được phân tích dưới đây.

Đầu tiên là sản phẩm Slack do Công ty Slack Technologies phát triển. Sản phẩm được khởi đầu từ công cụ tạo CV trực tuyến từ năm 2016, trải qua nhiều giai đoạn phát triển, hiện nay sản phẩm đã trở thành một công cụ hỗ trợ gắn kết giữa nhà tuyển dụng và ứng viên với nhiều tính năng, trong đó nổi bật kể đến hai kênh việc làm chất lượng cao và kênh tuyển dụng nhân sự. Trong quy chế hoạt động của TopCV [1] có nêu ra những dịch vụ mà sản phẩm này cung cấp cho người sử dụng. Tại đây, hai tính năng nổi bật được đề cập là tính năng tạo CV từ nhiều mẫu đã có sẵn cho ứng viên của hệ thống, và tính năng tìm kiếm ứng viên cho bên nhà tuyển dụng để sau đó có thể đánh giá hồ sơ và gửi lời mời ứng tuyển tới ứng viên. Một tính năng khác cho nhà tuyển dụng là tính năng đăng tin tuyển dụng, sau đó tin tuyển dụng này sẽ được đội ngũ quản trị viên kiểm tra và phê duyệt. Ứng viên có thể tìm kiếm việc làm và nộp hồ sơ có sử dụng CV đã được tạo từ trước trên hệ thống hoặc tạo mới CV. Ngoài ra hệ thống còn cung cấp dịch vụ hỗ trợ phát triển sự nghiệp cho ứng viên.

Thứ hai là sản phẩm CareerBuilder.vn do Công ty Cổ phần CareerBuilder quản lý. Cũng giống như TopCV, sản phẩm này cung cấp các chức năng tương tự như vậy. Theo quy chế hoạt động của CareerBuilder [2], quy trình hoạt động cho ứng viên cũng bao gồm tạo hồ sơ trực tuyến, nộp hồ sơ trực tuyến, quy trình hoạt động của nhà tuyển dụng cho phép đăng tin tuyển dụng, tìm kiếm ứng viên, nhận và xét duyệt hồ sơ theo quy trình.

Thứ ba là sản phẩm CareerLink.vn do Công ty Trách nhiệm hữu hạn CareerLink quản lý. Cũng giống như hai sản phẩm kể trên, CareerLink.vn cung cấp dịch vụ

2. KHẢO SÁT HIỆN TRẠNG

tuyển dụng trên website với các tính năng tương tự. Ngoài ra sản phẩm còn cung cấp dịch vụ tìm kiếm và cung cấp nhân sự cao cấp cho các nhà tuyển dụng cần tìm những vị trí nhân sự chủ chốt, có vai trò quan trọng, hoặc yêu cầu cần có chuyên môn cao, nhiều kinh

nghiệm. [3]

Thông qua khảo sát một vài hệ thống nổi bật trên thị trường về đăng tin tuyển dụng và môi giới việc làm, em đã thấy được các hệ thống đều có điểm mạnh khi hỗ trợ cho các ứng viên tạo hồ sơ một cách nhanh chóng, hiệu quả. Hệ thống tìm kiếm mạnh mẽ, áp dụng một số công nghệ hiện đại như trí tuệ nhân tạo để nâng cao chất lượng tìm kiếm. Bên phía nhà tuyển dụng được cung cấp các tính năng đăng tin tuyển dụng, tìm kiếm ứng viên, và tìm kiếm ứng viên chất lượng cao.

Tuy nhiên các hệ thống trên đều có một điểm chung là khi ứng viên ứng tuyển và nhà tuyển dụng nhận hồ sơ, họ sẽ trao đổi trực tiếp với nhau để tiến hành tiếp các nội dung trong quy trình tuyển dụng. Các hệ thống trên chỉ hỗ trợ một vài bước đầu trong quy trình tuyển dụng và thường không có một quy trình tuyển dụng đầy đủ.

Chính vì vậy sản phẩm em hướng tới trong đồ án này sẽ cải thiện để hỗ trợ được một quy trình tuyển dụng hoàn chỉnh từ khi bắt đầu nộp hồ sơ cho đến khi đơn ứng tuyển được chấp nhận hoặc bị từ chối.

Từ những khảo sát trên đây, em nhận thấy các tính năng quan trọng cần được phát triển trong sản phẩm bao gồm các tính năng được nêu dưới đây.

- Tính năng đăng nhập/đăng xuất, đây là tính năng bắt buộc phải có để phân quyền người sử dụng, đồng thời cung cấp các tính năng cho từng đối tượng cụ thể.
- Các tính năng liên quan đến đăng tin việc làm, bao gồm các tính năng thêm/sửa/xóa để tạo ra danh sách các việc làm có nhu cầu tuyển dụng.
- Các tính năng tìm kiếm, bao gồm tìm kiếm việc làm (đối với người tìm việc) và tìm kiếm ứng viên (đối với nhà tuyển dụng) để giúp dễ dàng hơn trong việc lọc các đối tượng quan tâm với người sử dụng.
- Quan trọng nhất là các tính năng liên quan tới hỗ trợ quy trình tuyển dụng, bao gồm nộp đơn ứng tuyển, nhận đơn, chuyển trạng thái đơn trong quá trình tuyển dụng, để tạo ra một hệ thống hỗ trợ quy trình tuyển dụng đầy đủ từ đầu đến cuối trong quá trình sử dụng.

6

3. TỔNG QUAN CHỨC NĂNG



Hình 2.1: Biểu đồ usecase tổng quát

3.

Tổng quan chức năng

1.

Biểu đồ use case tổng quát

Hệ thống chia đối tượng sử dụng thành hai loại người dùng chính. Loại I là phía người tìm việc, được chia nhỏ thành người dùng có tài khoản và khách vãng lai có khả năng tìm kiếm việc làm và ứng tuyển vào công việc mong muốn. Loại II là phía nhà tuyển dụng, bao gồm nhân viên quản lý và nhân viên bình thường có quyền liên quan tới đăng tin tuyển dụng và xét duyệt hồ sơ ứng tuyển của các ứng viên.

Hình 2.1 ở đây là biểu đồ use case tổng quát mô tả chức năng phần mềm. Trong đó đối tượng sử dụng được chia ra làm hai nhóm chính bao gồm phía người tìm việc và phía nhà tuyển dụng. Đối với mỗi nhóm, các đối tượng tiếp tục được chia nhỏ ra và biểu đồ mô tả các chức năng từng loại đối tượng có thể thực hiện.

Đầu tiên về phía người tìm việc. Đối với khách vãng lai, họ có thể sử dụng hệ thống để tìm kiếm việc làm theo các tiêu chí phù hợp với nhu cầu.

Kế tiếp là đến các ứng viên. Sự khác biệt đối với khách vãng lai là ứng viên được yêu cầu phải đăng ký tài khoản để có thể sử dụng. Ngoài chức năng cơ bản tìm kiếm việc làm giống như bên khách

vãng lai, các ứng viên có thể ứng tuyển nộp hồ sơ vào công việc mình mong muốn, và theo dõi được các đơn ứng tuyển bản thân đã nộp.

Chuyển sang phía nhà tuyển dụng. Đầu tiên đối với nhân viên, giống với ứng viên, họ được yêu cầu phải có tài khoản đăng nhập với tư cách nhân viên để có thể sử dụng các chức năng đúng với yêu cầu. Nhân viên có khả năng nhận đơn ứng

7

3. TỔNG QUAN CHỨC NĂNG

tuyển, thực hiện quy trình ứng tuyển hay còn được gọi là quản lý đơn ứng tuyển. Nhân viên cần phải theo dõi được các đơn ứng tuyển họ đang phụ trách. Ngoài ra nhân viên có thể tìm ứng viên phù hợp với nhu cầu và mời ứng viên đó ứng tuyển một công việc từ phía nhà tuyển dụng.

Cuối cùng là đối với quản lý. Quản lý cũng có các khả năng giống với nhân viên. Ngoài ra quản lý còn được bổ sung thêm các chức năng liên quan tới quản lý danh sách các công việc, cập nhật nội dung, thêm mới công việc. Quản lý cũng đồng thời có khả năng quản lý các nhân viên của nhà tuyển dụng tham gia vào hệ thống này.

Biểu đồ use case phân rõ tìm việc làm



Hình 2.2: Use case phân rõ tìm việc làm

Hình 2.2 này mô tả use case phân rõ tìm việc làm cho phía người tìm việc. Trong đó sau khi tìm việc làm theo một số tiêu chí, người tìm việc có thể xem chi tiết nội dung việc làm đó. Tiếp đến người tìm việc có thể ứng tuyển vào công việc này nếu như có nhu cầu, và điều đó yêu cầu người tìm việc cần phải đăng nhập với tư cách ứng viên để có thể sử dụng tính năng này.

8

3. TỔNG QUAN CHỨC NĂNG

1.

Biểu đồ use case phân rã quản lý đơn ứng tuyển



Hình 2.3: Use case phân rã quản lý đơn ứng tuyển

Trong hình 2.3, use case phân rã quản lý đơn ứng tuyển được mô tả là chức năng phía bên nhà ứng tuyển. Trong đó đơn ứng tuyển sẽ được gắn với công việc cụ thể. Khi đó bên phía nhà ứng tuyển có thể xem chi tiết các đơn này, nhận đơn ứng tuyển. Sau đó bên nhà ứng tuyển có thể chuyển trạng thái của các đơn ứng tuyển họ đang phụ trách.

Biểu đồ use case phân rã tìm ứng viên



Hình 2.4: Use case phân rã tìm ứng viên

Hình 2.4 tại đây minh họa cho use case phân rã tìm ứng viên bên phía nhà tuyển dụng. Trong đó nhà tuyển dụng có thể tìm kiếm các

ứng viên đã đăng kí trên hệ thống theo một số tiêu chí. Tiếp đến có thể xem thông tin ứng viên và mời tuyển dụng cho một công việc nếu cảm thấy ứng viên đó phù hợp với nhu cầu của bên nhà tuyển dụng.

9

3. TỔNG QUAN CHỨC NĂNG

1.

Biểu đồ use case phân rã quản lý việc làm



Hình 2.5: Use case phân rã quản lý việc làm

Hình 2.5 mô tả cho use case quản lý việc làm. Trong use case này, người dùng cần phải đăng nhập với tư cách quản lý để có thể thực hiện các thao tác. Các thao tác liên quan đến việc làm bao gồm thêm việc làm mới, cập nhật thông tin việc làm và xóa việc làm khỏi cơ sở dữ liệu.

Biểu đồ use case phân rã quản lý nhân viên



Hình 2.6: Use case phân rã quản lý nhân viên

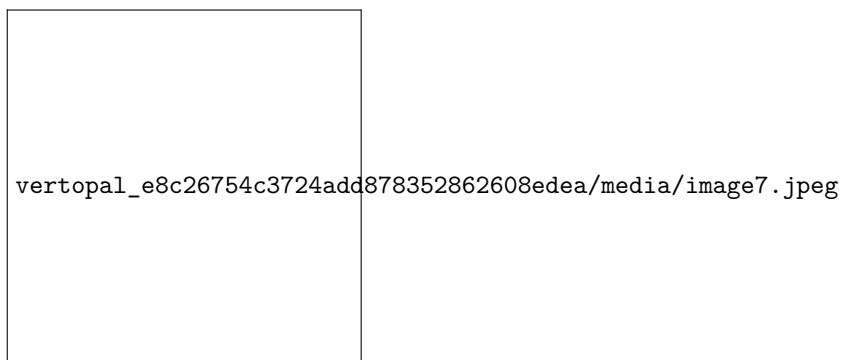
Use case phân rã quản lý nhân viên được minh họa trong hình 2.6. Tại đây người dùng cần phải đăng nhập với tư cách quản lý để có thể sử dụng chức năng này. Khi đó quản lý có thể thêm nhân viên mới, xem thông tin của nhân viên trong cùng bên tuyển dụng và xóa nhân viên khỏi cơ sở dữ liệu khi có nhu cầu.

10

3. TỔNG QUAN CHỨC NĂNG

1.

Biểu đồ use case phân rã xem danh sách đơn ứng tuyển đang tham gia



Hình 2.7: Use case phân rã xem danh sách đơn ứng tuyển đang tham gia

Hình 2.7 mô tả use case quản lý danh sách đơn ứng tuyển đang tham

gia. Để sử dụng use case này, người dùng cần phải đăng nhập với bất cứ tư cách gì trong ứng viên với bên nhà ứng tuyển. Trong đó người dùng có thể xem chi tiết các đơn ứng tuyển bản thân tham gia và có khả năng cập nhật, hoặc chuyển trạng thái đơn nếu như được cho phép

Quy trình nghiệp vụ

Trong hệ thống có nhiều quy trình nghiệp vụ dẫn tới các hoạt động của hệ thống, trong đó quy trình quan trọng nhất là quy trình ứng tuyển vào một công việc. Quy trình này có xuất phát điểm từ khi ứng viên nộp hồ sơ ứng tuyển cho đến kết thúc khi đơn ứng tuyển được chấp nhận, hoặc đơn ứng tuyển bị từ chối. Mô tả của quy trình được biểu diễn trong biểu đồ hoạt động dưới đây (biểu diễn trên hình 2.8).

11

3. TỔNG QUAN CHỨC NĂNG



Hình 2.8: Biểu đồ hoạt động cho quy trình ứng tuyển

Trong biểu đồ trên, mô tả quy trình ứng tuyển với sự tham gia của hai đối tượng là ứng viên và nhân viên, xoay quanh đơn ứng tuyển là đối tượng được tác động trực tiếp trong quy trình. Bắt đầu từ ứng viên nộp đơn ứng tuyển, trải qua các giai đoạn chính bao gồm giai đoạn sơ tuyển, kế tiếp là giai đoạn phỏng vấn, và cuối cùng là giai đoạn chấp nhận/từ chối hồ sơ. Trên biểu đồ đã lược bỏ phần trong bất kì giai đoạn nào, nếu như ứng viên không còn nhu cầu hoặc nhân viên đánh giá ứng viên không đạt các tiêu chí mong muốn thì đơn ứng tuyển có thể bị từ chối ngay lập tức.

4. ĐẶC TẢ CHỨC NĂNG

4.

Đặc tả chức năng

1.

Đặc tả use case “Đăng nhập”

Bảng 2.1 dưới đây đặc tả cho use case đăng nhập, đối tượng áp dụng bao gồm mọi người sử dụng hệ thống đã đăng kí tài khoản.

| Mã Use case | UCDN | Tên use case | Đăng nhập |
|--------------------------------|------------------------------|--|--|
| Tác nhân Tiên điều kiện | Ứng viên, quản lý, nhân viên | | |
| | Không | | |
| | STT | Thực hiện bởi | Hành động |
| Luồng sự kiện chính | 1. 2. 3. 4. | Người dùng Hệ thống Người dùng Hệ thống | Chọn tính năng đăng nhập Hiển thị giao diện đăng nhập Nhập thông tin tên tài khoản, mật khẩu, lựa chọn đăng nhập Thông báo đăng nhập thành công |
| Luồng sự kiện thay thế | 4a. 4b. | Hệ thống Hệ thống | Thông báo lỗi nếu người dùng nhập sai tài khoản hoặc mật khẩu Thông báo lỗi nếu người dùng chưa nhập đủ thông tin tài khoản hoặc mật khẩu |
| Hậu điều kiện | Không | | |

Bảng 2.1: Đặc tả use case đăng nhập

13

4. ĐẶC TẢ CHỨC NĂNG

1.

Đặc tả use case “Ứng viên ứng tuyển”

Bảng 2.2 dưới đây đặc tả cho use case ứng viên ứng tuyển, đối tượng áp dụng bao gồm ứng viên đã đăng kí tài khoản và đã đăng nhập vào hệ thống.

| Mã Use case | UC001 | Tên use case | Ứng tuyển |
|-------------------------------------|-----------------------------------|---------------------|-----------|
| Tác nhận Tiền điều kiện | Ứng viên Ứng viên đã đăng nhập | | |
| | STT | Thực hiện bởi | Hành động |

| Mã Use case | | UC001 | Tên use case | Ứng tuyển |
|------------------------------------|-----|-------|-----------------------|-----------|
| Luồng sự kiện chính | 1. | Ứng | Tìm kiếm về thông | |
| | 2. | viên | tin công việc muốn | |
| | 3. | Hệ | ứng | |
| | 4. | thống | tuyển | |
| | 5. | Ứng | Hiển thị danh sách | |
| | 6. | viên | công việc phù hợp với | |
| | 7. | Hệ | kết quả tìm kiếm | |
| | 8. | thống | Lựa chọn xem chi tiết | |
| Luồng sự kiện thay thế | | Ứng | công việc muốn ứng | |
| | | viên | tuyển | |
| | | Hệ | Hiển thị trang thông | |
| | | thống | tin chi tiết về công | |
| | | Ứng | việc đó | |
| | | viên | Lựa chọn chức năng | |
| | | Hệ | ứng tuyển | |
| | | thống | Hiển thị giao diện | |
| | | Ứng | nộp đơn ứng tuyển | |
| | | viên | vào công việc | |
| | | Hệ | Điền các thông tin | |
| | | thống | được yêu cầu vào đơn | |
| | | Ứng | ứng tuyển, sau đó | |
| | | viên | nhấn nút xác nhận | |
| | | Hệ | ứng tuyển Thông báo | |
| | | thống | ứng viên ứng tuyển | |
| | | Ứng | thành công | |
| | | viên | Kết thúc use case nếu | |
| | 8a. | thống | ứng viên xác nhận | |
| | 8b. | Hệ | không ứng tuyển | |
| | 8c. | thống | Thông báo lỗi nếu | |
| | 8d. | Hệ | như đơn ứng tuyển có | |
| | | thống | các nội dung không | |
| | | Hệ | hợp lệ | |
| | | thống | Thông báo lỗi nếu | |
| | | | như đơn ứng tuyển có | |
| | | | những trường bất | |
| | | | buộc chưa được điền | |
| | | | Thông báo lỗi nếu | |
| | | | như ứng viên đã đang | |
| | | | ứng tuyển công việc | |
| | | | này | |

| | | | |
|----------------------|-------|---------------------|-----------|
| Mã Use case | UC001 | Tên use case | Ứng tuyển |
| Hậu điều kiện | Không | | |

Bảng 2.2: Đặc tả use case ứng viên ứng tuyển

14

4. ĐẶC TẢ CHỨC NĂNG

1.

Đặc tả use case “Nhân viên nhận hồ sơ”

Bảng 2.3 dưới đây đặc tả cho use case nhân viên nhận hồ sơ, đối tượng áp dụng bao gồm nhân viên đã đăng kí tài khoản và đã đăng nhập vào hệ thống, với đối tượng hồ sơ là đơn ứng tuyển vào một việc làm đang tuyển dụng trong công ty của nhân viên thuộc về.

| | | | |
|-----------------------|------------------------|----------------------|--|
| Mã Use case | UC002 | Tên use case | Nhận đơn ứng tuyển |
| Tác nhân | Nhân viên | | |
| Tiền điều kiện | Nhân viên đã đăng nhập | | |
| | | Hành động | |
| | STT | Thực hiện bởi | |
| Luồng sự kiện chính | 1. | Nhân viên | Truy cập vào trang việc làm, lựa chọn đơn ứng tuyển muốn phụ trách |
| | 2. | Hệ thống | Hiển thị nội dung chi tiết đơn ứng tuyển |
| | 3. | Nhân viên | Xác nhận phụ trách đơn ứng tuyển |
| | 4. | Hệ thống | Cập nhật đơn ứng tuyển và thông báo nhận phụ trách thành công |

| Mã Use case | | Tên use case | Nhận đơn ứng tuyển |
|------------------------|------------|----------------------|--|
| | UC002 | | |
| Luồng sự kiện thay thế | 4a. 4b. | Hệ thống Hệ thống | Kết thúc use case nếu nhân viên không nhận đơn phụ trách Thông báo lỗi nếu đã có nhân viên khác phụ trách đơn |
| Hậu điều kiện | Không | | |

Bảng 2.3: Đặc tả use case nhân viên nhận hồ sơ

Đặc tả use case “Nhân viên quản lý danh sách việc làm”

Bảng 2.4 dưới đây đặc tả cho use case nhân viên quản lý việc làm, đối tượng áp dụng bao gồm quản lý đã đăng kí tài khoản và đã đăng nhập vào hệ thống, các đối tượng hướng tới là các việc làm đăng tin tuyển dụng của công ty mà quản lý đó là nhân viên và có vai trò phụ trách.

15

4. ĐẶC TẢ CHỨC NĂNG

| Mã Use case | UC003 | Tên use case | Quản lý việc làm |
|--|---------------------------------|--------------|------------------|
| Tác nhân Tiên điều kiện | Quản lý Quản lý đã đăng nhập | | |
| Mã: UC0031 - Thêm việc làm: Mã: UC0032 - Xóa việc làm: Mã: UC0033 - Cập nhật việc làm: | | Hành động | |
| STT | Thực hiện bởi | | |

| Mã Use case | | UC003 | Tên use case | Quản lý việc làm |
|------------------------------|----|-------|--------------------|---|
| | | | | Chọn việc làm muốn cập nhật, vào xem chi tiết |
| Luồng sự kiện chính | 1. | Quản | | Hiển thị trang nội dung chi tiết của việc làm Lựa chọn tính năng cập nhật nội dung việc làm |
| | 2. | lý | | Hiển thị biểu mẫu cập nhật nội dung |
| | 3. | Hệ | | Tiến hành chỉnh sửa nội dung và xác nhận |
| | 4. | thống | | Cập nhật nội dung việc làm và thông báo cập nhật thành công |
| | 5. | Quản | | Kết thúc use case nếu quản lý xác nhận |
| | 6. | lý | | không lựa chọn không cập nhật |
| | | Hệ | | Thông báo lỗi nếu như thông tin cập nhật không hợp lệ |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | |
| | | Quản | | |
| | | lý | | |
| | | Hệ | | |
| | | thống | | |
| | | Hệ | | |
| | | thống | | </ |

pháp lưu trữ sao lưu hoặc lưu trữ lịch sử chỉnh sửa dữ liệu trong quá trình vận hành để khôi phục được dữ liệu.

Tính bảo mật

Cần phải đảm bảo tính bảo mật, đảm bảo đối tượng sử dụng chỉ được truy cập những phần dữ liệu được phép. Đảm bảo phân quyền đúng đối tượng và có cơ chế bảo vệ những truy cập trái phép đến các tài nguyên.

Giao diện

Giao diện cần phải được xây dựng phù hợp với người sử dụng. Đảm bảo các yếu tố dễ dàng, thân thiện với người dùng. Các phần của giao diện phải đồng bộ, thống nhất, đơn giản, dễ tiếp cận, dễ sử dụng, có hướng dẫn chi tiết.

Một số đặc tính cần có của sản phẩm

Đầu tiên là về tính sẵn có. Hệ thống luôn phải sẵn sàng hoạt động kể từ khi triển khai để đảm bảo quá trình sử dụng của người dùng ít khi hoặc không bao giờ gặp vấn đề về gián đoạn.

Tiếp đến là tính khả dụng, hệ thống cần phải hoạt động ổn định, thực hiện các yêu cầu từ người dùng một cách nhanh chóng, chính xác, đảm bảo được các quá trình tìm kiếm, xử lý, hiển thị cho người dùng một cách hiệu quả.

Kế đến là tính đúng đắn, hệ thống cần đảm bảo các quá trình hoạt động sẽ luôn đảm bảo được toàn vẹn dữ liệu, các kết quả thu được luôn phải được đảm bảo tính chính xác trong các hành vi thực hiện.

Hiệu năng

Cơ sở dữ liệu cần được tối ưu, tránh dư thừa trong quá trình lưu trữ dữ liệu. Các kết quả truy vấn dữ liệu không được vượt quá 5 giây. Đảm bảo phần tài nguyên còn trống đạt tối thiểu 20%. Có khả năng đáp ứng truy cập cho đồng thời đến đơn vị nghìn người.

Các yêu cầu khác

Dữ liệu cần được lưu trữ lâu dài, trong trường hợp phần dữ liệu đã lâu không được sử dụng sẽ được chuyển sang lưu trữ ở một kho dữ liệu khác để tránh quá tải đến nơi lưu trữ dữ liệu hiện tại, vừa đảm bảo sao lưu dữ liệu, vừa để sử dụng cơ sở

dữ liệu lưu trữ thường ngày hiệu quả. Khi cần sẽ truy cập vào kho dữ liệu đó để lấy ra và sử dụng.

Hệ thống được tiến hành bảo trì định kì theo chu kì 3 tháng.

Kết chương

Chương này đã phân tích chi tiết về khảo sát hiện trạng của bài toán đang xét tại thời điểm hiện tại, các giải pháp đã được triển khai cùng với đó là những hạn chế còn gặp, từ đó đề xuất ra các cải tiến. Tiếp đến là triển khai phân tích thiết kế về các chức năng cần có trong sản phẩm được phát triển, kèm theo đó là các yêu cầu về nghiệp vụ, yêu cầu chức năng và phi chức năng cho sản phẩm.

Từ những phân tích trên đây, em tiến hành sử dụng các công nghệ và giải pháp được trình bày trong chương tiếp theo – Chương 3.

18

CHƯƠNG 3. CÔNG NGHỆ SỬ DỤNG

Tổng quan

Trong chương trước, em đã đề cập tới các nội dung về phân tích và khảo sát bài toán được giải quyết, kèm theo đó là những giải pháp thiết kế để giải quyết bài toán cùng với phân tích nghiệp vụ.

Vì vậy trong chương này, em sẽ giới thiệu những công nghệ, phần mềm được sử dụng để phát triển và giải quyết bài toán trong đồ án này cùng với những nguyên do lựa chọn chúng để giải quyết bài toán hiệu quả.

2.

Công nghệ lưu trữ dữ liệu

1.

Hệ quản trị cơ sở dữ liệu quan hệ

Hệ quản trị cơ sở dữ liệu quan hệ (Relational Database Management System – RDBMS) là một hệ thống quản trị cơ sở dữ liệu dựa trên nền tảng mô hình quản hệ được phát biểu và giới thiệu bởi Edgar F. Codd vào năm 1970. [4]

Mô hình dữ liệu quan hệ có đặc trưng các dữ liệu được tổ chức biểu diễn dưới dạng bảng. Mỗi bảng biểu diễn các nội dung của một đối tượng hay một quan hệ nào đó. Mỗi bảng được chia ra thành các hàng và cột biểu diễn các giá trị, và là đơn vị lưu trữ dữ liệu.

Mô hình quan hệ có một số khái niệm cơ bản để xác định và biểu diễn dữ liệu. Đầu tiên là thuộc tính, thuộc tính mô tả một tính chất, hay đặc trưng của đối tượng, được biểu diễn trong bảng dưới dạng cột và thuộc tính yêu cầu tên kèm kiểu giá trị cần lưu trữ. Tiếp đến là quan hệ, hay lược đồ, bao gồm một tập hợp các thuộc tính có liên quan và tên của quan hệ xác lập trên các thuộc tính đó. Kế tiếp là bộ giá trị, hay bản ghi, là các thông tin của một đối tượng trong quan hệ kể trên, được biểu diễn dưới dạng dòng trong bảng, kèm theo tên của từng thuộc tính trong bảng. Tiếp đến là một thành phần quan trọng trong mô hình quan hệ, đó là khóa. Khóa được định nghĩa là tập tối thiểu các thuộc tính để xác định duy nhất một bộ giá trị, điều này có nghĩa là các bộ giá trị sẽ phân biệt nhau thông qua khóa và không có hai bộ khác nhau nào có cùng khóa. Ngoài ra còn có một số phép toán để thực hiện trên mô hình quan hệ này, một số phép toán có thể kể đến như phép hợp, phép giao, phép lấy tích chập, . . .

Mô hình có nhiều ưu điểm khi dựa trên lý thuyết tập hợp để xây dựng ra các bảng, đồng thời tạo khả năng xây dựng được các truy vấn phong phú, phức tạp dựa trên các quan hệ được định nghĩa trong mô hình. Các quan hệ được định nghĩa có mối quan hệ rõ ràng, đảm bảo được các yếu tố về dữ liệu khi thiết kế tốt bao gồm

2. CÔNG NGHỆ LƯU TRỮ DỮ LIỆU

chống dữ thừa dữ liệu, đảm bảo toàn vẹn dữ liệu, . . . Tuy nhiên mô hình cũng mang nhiều tính hạn chế. Do biểu diễn rõ ràng và chặt chẽ nên cấu trúc dữ liệu không linh hoạt, khó khăn khi thay đổi sau khi đã thiết lập xong. Ngoài ra các truy vấn cần phải có ngữ nghĩa rõ ràng, chính xác, không thể hoạt động với các truy vấn mập mờ, không chính xác, không sử dụng ngôn ngữ được định nghĩa qua các phép toán.

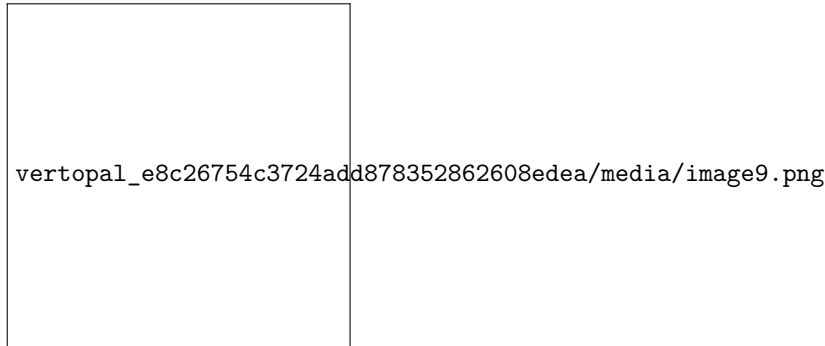
Hệ quản trị cơ sở dữ liệu quan hệ là hệ quản trị cơ sở dữ liệu được xây dựng trên mô hình quan hệ, trong đó cung cấp cơ sở dữ liệu để lưu trữ dữ liệu theo mô hình quan hệ, cung cấp khả năng truy vấn vào dữ liệu một cách nhanh chóng, tối ưu và hiệu quả. Hệ quản trị này cung cấp khả năng thực hiện các truy vấn tới dữ liệu thông qua ngôn ngữ SQL (Structured Query Language – Ngôn ngữ truy vấn có cấu trúc) với khả năng tạo ra và thực hiện các truy vấn phức tạp. Đồng thời do được xây dựng trên nền mô hình quan hệ nên hệ quản trị cơ sở dữ liệu quan hệ cũng mang những ưu và nhược điểm tương tự với mô hình quan hệ.

Một số hệ quản trị cơ sở dữ liệu quan hệ nổi tiếng thường được sử dụng là Microsoft SQLServer, Oracle, IBM DB2, MySQL và Microsoft Access.

Từ những ưu nhược điểm của hệ quản trị cơ sở dữ liệu quan hệ, từ các yêu cầu liên quan tới lưu trữ, xử lý dữ liệu và bảo mật, an toàn,

tính toàn vẹn dữ liệu, em lựa chọn hệ quản trị cơ sở dữ liệu quan hệ làm hệ thống để quản lý dữ liệu của sản phẩm.

Hệ quản trị cơ sở dữ liệu MySQL



Hình 3.1: Logo của MySQL¹

MySQL là hệ quản trị cơ sở dữ liệu quan hệ, được phát hành lần đầu tiên vào năm 1995 bởi nhà phát triển MySQL AB. Sau đó MySQL được mua lại bởi công ty Sun Microsystems vào năm 2008, và một lần nữa được chuyển quyền sở hữu sau khi Oracle mua lại Sun Microsystems. Cho đến hiện tại, MySQL vẫn đang nằm dưới sự

¹Nguồn: <https://www.mysql.com/>

20

3. NGÔN NGỮ LẬP TRÌNH VÀ FRAMEWORK SỬ DỤNG

quản lý của Oracle. Hình 3.1 là logo của MySQL.

MySQL là một hệ quản trị cơ sở dữ liệu mã nguồn mở, được xây dựng hoạt động theo mô hình client-server (khách – chủ). Hiện tại MySQL được xây dựng hoạt động trên nhiều hệ điều hành khác nhau như Windows, MacOS, Linux, Unix và nhiều hệ điều hành khác. Đối với ngôn ngữ lập trình, MySQL cũng đã được hỗ trợ trên nhiều ngôn ngữ khác nhau như C++, Java, PHP, Ruby, . . . Ngoài phiên bản thông thường miễn phí, MySQL còn cung cấp các phiên bản nâng cao trả phí hướng tới các nhóm đối tượng khác như các doanh nghiệp, tập đoàn.

Hiện tại, MySQL là một hệ quản trị cơ sở dữ liệu rất phổ biến vì nhiều lý do khác nhau. Được phát triển dưới dạng mã nguồn mở, MySQL rất linh hoạt cùng với việc cho phép người dùng có khả năng can thiệp và chỉnh sửa theo mong muốn, đồng thời việc cài đặt và sử dụng cũng đơn giản do có được nhiều sự hỗ trợ từ các hệ điều hành cùng với các ngôn ngữ lập trình. Một lý do khác là hiệu năng của MySQL tốt, phù hợp với để lưu trữ dữ liệu cho các hệ thống từ

đơn giản tới phức tạp với tốc độ xử lý truy vấn cao. Kèm với đó là tính năng phân quyền người sử dụng, các tính năng bảo mật và mã hóa tốt, đảm bảo đúng đối tượng mới có thể truy cập vào hệ thống, đặt tính an toàn của hệ thống lên hàng đầu. MySQL đã tồn tại và phát triển trong một thời gian dài nên cùng với các ưu điểm trên, độ phổ biến của MySQL đã lan rộng. Một vài trang web nổi tiếng cũng đã hoặc từng sử dụng MySQL để lưu trữ dữ liệu của mình như Facebook, Google.

Từ những ưu điểm kể trên, em lựa chọn MySQL là hệ quản trị cơ sở dữ liệu được sử dụng trong đồ án này.

3.

Ngôn ngữ lập trình và framework sử dụng

1.

Lập trình hướng đối tượng

Lập trình hướng đối tượng (Object-Oriented Programming – OOP) là một mô thức lập trình dựa trên nền tảng là hướng đối tượng để mô hình, thiết kế các sản phẩm sao cho chúng hoạt động giống như hoạt động thật ngoài thực tế. Mô thức lập trình này khi ra đời đã giải quyết được một số vấn đề còn tồn đọng trước đó trong giai đoạn phương pháp lập trình cấu trúc đang phổ biến. Việc lập trình cấu trúc không giải quyết tốt các bài toán phức tạp, hay không tái sử dụng được các mô đun đã lập trình trong cùng sản phẩm dẫn tới sự ra đời của lập trình hướng đối tượng.

Hướng đối tượng là một kĩ thuật, trong đó kĩ thuật này cho phép việc mô hình hóa các đối tượng trong thực tế vào trong một phần mềm để mô tả lại thế giới thực, vì vậy trọng tâm của kĩ thuật này chính là đối tượng. Đối tượng (Object) chính là khái niệm trung tâm trong lập trình hướng đối tượng, dùng để mô hình hóa lại một

21

3. NGÔN NGỮ LẬP TRÌNH VÀ FRAMEWORK SỬ DỤNG

thực thể trong hiện thực. Để mô hình hóa được thực thể, đối tượng cần phải có những thuộc tính bao gồm những thông tin, những đặc tính về thực thể đó, và đối tượng cần có phương thức (hay thông điệp) để đối tượng có thể trao đổi, giao tiếp với các đối tượng khác. Tiếp đến là khái niệm Lớp (Class) chính là tập hợp những thuộc tính, những phương thức chung của một nhóm các đối tượng có liên hệ gần với nhau. Có thể nói Lớp chính là khái quát hóa của Đối tượng.

[5]

Các tính chất chính của lập trình hướng đối tượng đem lại ưu thế trước các mô thức lập trình khác bao gồm. Tính đóng gói (encapsulation) là tính chất quy định việc trừu tượng hóa các thuộc tính, các hành vi liên quan tới nhau được đóng gói lại trong một khái niệm là lớp, sau khi đóng gói thì đối tượng có thể thực hiện các hành vi, tính năng cần thiết trong nội bộ lớp đó, và nhờ có việc này nên đảm bảo được việc chặn những truy cập trái phép từ bên ngoài, và sinh ra tính che giấu thông tin chi tiết của đối tượng với bên ngoài, bên ngoài chỉ có thể tương tác với đối tượng thông qua những phương thức mà đối tượng cung cấp công khai ra, còn bên trong xử lý như thế nào sẽ được che giấu hoàn toàn. Tính chất thứ hai được kể tới của lập trình hướng đối tượng là tính kế thừa (inheritance), đây là tính chất mô tả quan hệ phân cấp giữa các lớp, cho phép các lớp dẫn xuất được kế thừa lại các tính chất, các phương thức của lớp cơ sở để tái sử dụng lại các tính chất, phương thức đã được định nghĩa trước. Nhờ có tính chất này mà lập trình hướng đối tượng có khả năng tái sử dụng phần mềm, tái sử dụng các mô đun để làm giảm bớt dư thừa, tránh lặp trong quá trình phát triển. Tính chất kế thừa là tính đa hình (polymorphism), là một tính chất đi cùng với tính kế thừa. Tính chất này cho phép các đối tượng khác nhau có thể nhận cùng một thông điệp giống nhau nhưng được thực thi một cách khác nhau theo từng đối tượng. Để có thể dùng tính chất này, các lớp chứa các đối tượng đó cần phải kế thừa một lớp cơ sở, và từ đó ghi đè các phương thức của lớp cơ sở đó để có thể hành xử đúng như mong muốn cho từng lớp dẫn xuất này. Tính chất này đem lại khả năng tổng quát hóa cao trong quá trình thiết kế và xây dựng phần mềm.

Hiện nay có nhiều ngôn ngữ lập trình được xây dựng theo hướng lập trình hướng đối tượng và được sử dụng phổ biến trong lập trình xây dựng các sản phẩm. Một vài ngôn ngữ lập trình có hỗ trợ lập trình hướng đối tượng có thể kể ra như C++, Java, .NET, PHP, Ruby, Python, . . .

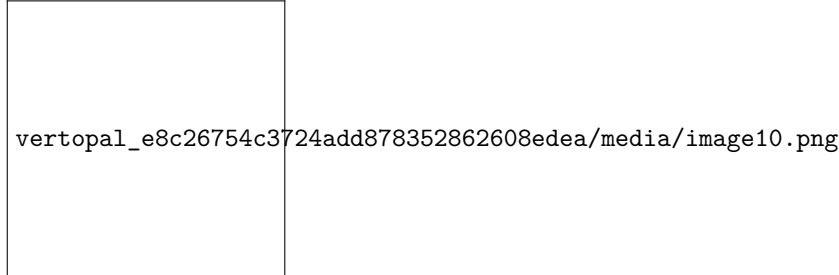
Nhờ những ưu điểm trên đây, cùng với sự phù hợp để mô hình hóa các thực thể trong thực tế trở thành các đối tượng trong sản phẩm được phát triển, em lựa chọn lập trình hướng đối tượng là phương pháp để phát triển sản phẩm trong đồ án này.

22

3. NGÔN NGỮ LẬP TRÌNH VÀ FRAMEWORK SỬ DỤNG

1.

Ngôn ngữ lập trình: Ruby



Hình 3.2: Logo của ngôn ngữ lập trình Ruby²

Ruby là ngôn ngữ lập trình hướng đối tượng được Yukihiro “Matz” Matsumoto phát triển từ năm 1993 và phát hành lần đầu năm 1995. Vào năm 2006, Ruby đã trở thành một ngôn ngữ lập trình được chấp nhận rộng rãi trên quy mô toàn cầu. Cho đến hiện tại, Ruby vẫn thường xuyên nằm trong danh sách những ngôn ngữ lập trình thông dụng nhất trên thế giới. Hình 3.2 là logo của ngôn ngữ lập trình Ruby.

Ruby được nhà phát triển kết hợp những tính năng tâm đắc từ một số ngôn ngữ lập trình khác (Perl, Smalltalk, Eiffel, Ada, và Lisp) để tạo ra một ngôn ngữ lập trình với mục tiêu ngôn ngữ đó cần phải tự nhiên, nhưng không hề đơn giản trong cấu trúc xây dựng bên trong. Sự khác biệt lớn nhất giữa Ruby và một số ngôn ngữ lập trình hướng đối tượng nổi tiếng khác như Java, Python là việc tất cả mọi thứ trong ngôn ngữ này đều là đối tượng, trong khi đó ví dụ như Java các kiểu dữ liệu cơ bản và số không phải là đối tượng, điều này cho phép Ruby đối xử với các dữ liệu trên giống như hành xử với một đối tượng thông thường. [6]

Ruby là một ngôn ngữ có tính linh hoạt cao, cho phép nhà lập trình có khả năng chỉnh sửa phù hợp với nhu cầu của bản thân. Cú pháp sáng sủa, đơn giản, linh hoạt có thể tinh chỉnh. Bản thân một đối tượng quan trọng trong lập trình là biến được Ruby phân phạm vi dựa trên quy tắc viết mã nguồn chứ không cần khai báo thêm, không cần định nghĩa trước kiểu dữ liệu mà trong quá trình sử dụng Ruby sẽ tự kiểm tra và thêm kiểu dữ liệu. Ngôn ngữ Ruby là ngôn ngữ theo phong cách thông dịch, nghĩa là Ruby cho phép lập trình tương tác được, điều đó giúp cho các quá trình về kiểm thử, rà soát lỗi trở nên dễ dàng hơn. Cùng với đó là lượng thư viện lớn để hỗ trợ ngôn ngữ này có thể tương thích với nhiều hệ điều hành, nền tảng khác nhau.

Nhưng những ưu điểm đó cũng dẫn tới một số hạn chế của Ruby. Ngôn ngữ Ruby được hỗ trợ hoạt động với tính linh hoạt cao, vì thế

so với các ngôn ngữ thông thường như Java, ngôn ngữ Ruby chậm hơn và đó là hạn chế lớn nhất của ngôn ngữ

2Nguồn: <https://www.ruby-lang.org/en/about/logo/>

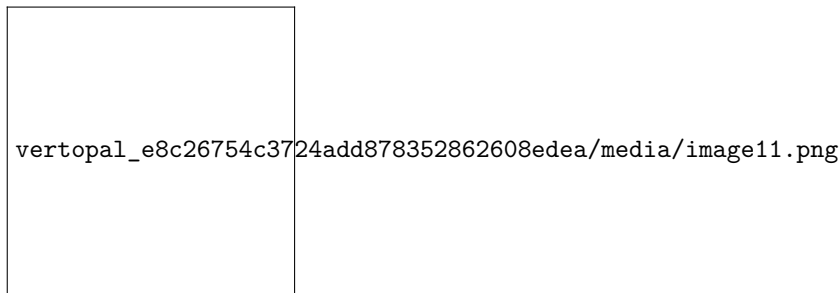
23

3. NGÔN NGỮ LẬP TRÌNH VÀ FRAMEWORK SỬ DỤNG

này. Ngoài ra việc linh hoạt trong viết mã nguồn nên khó tạo ra một quy chuẩn rõ ràng để viết mã nguồn hiệu quả mà vẫn đảm bảo tính linh động, dễ đọc, dễ hiểu.

Dựa trên những ưu điểm lợi thế hơn của Ruby so với một vài ngôn ngữ lập trình hướng đối tượng khác như Java bao gồm ưu điểm về mã nguồn linh hoạt, có nhiều thư viện xử lý các tác vụ thông thường, dễ dàng trong quá trình rà soát lỗi, kiểm thử nên em quyết định lựa chọn ngôn ngữ lập trình Ruby trở thành ngôn ngữ lập trình để phát triển sản phẩm trong đồ án này.

Framework sử dụng: Ruby on Rails



Hình 3.3: Logo của Ruby on Rails³

Ruby on Rails (thường được gọi tắt là Rails) là một framework miễn phí mã nguồn mở được sử dụng nhằm mục đích phát triển các sản phẩm chạy trên nền tảng web dựa trên ngôn ngữ lập trình Ruby được tác giả là David Heinemeier Hansson phát triển và phát hành lần đầu vào năm 2004. Cho đến hiện tại Ruby on Rails là framework phổ biến nhất để lập trình web dựa trên ngôn ngữ lập trình Ruby. Hình

3.3 là logo của framework Ruby on Rails.

Rails là một framework phát triển ứng dụng web có nhiều ưu điểm khiến cho framework này trở thành framework phổ biến nhất đối với ngôn ngữ Ruby. Rails cung cấp nhiều công cụ giúp cho người sử dụng có thể tạo ra các trang web có nhiều tính năng nhưng giảm được thời gian phát triển sản phẩm. Người sử dụng có thể dùng Rails để phát triển hoàn thiện toàn bộ ứng dụng web cả phía tương tác người

dùng với phía máy chủ xử lý hệ thống. Ngoài ra phạm vi áp dụng lớn, từ những trang web đơn giản hay những trang web với phong cách ứng dụng một trang cho đến những trang web có khối lượng xử lý lớn đều có thể sử dụng Rails để phát triển sản phẩm. Ưu điểm lớn tiếp theo là lượng thư viện của Rails lớn, hầu hết đều miễn phí, giúp cho việc tạo dựng những tính năng thông thường trở nên nhanh chóng và hiệu quả, đồng thời tận dụng được điểm mạnh của ngôn ngữ Ruby

3Nguồn: <https://rubyonrails.org/trademarks>

24

4. CÔNG CỤ XỬ LÝ TÁC VỤ NGẦM

giúp cho việc lập trình trở nên linh hoạt, dễ dàng mở rộng các thành phần trong mã nguồn. Rails còn được xây dựng bằng cách áp dụng một số mẫu thiết kế phần mềm như Convention over Configuration (CoC – ưu tiên các quy ước thay vì thiết lập) và Don't Repeat Yourself (DRY – tránh sự lặp lại, giảm dư thừa trong lập trình) khiến phần mã nguồn trở nên sáng sủa, giảm công sức triển khai. Cùng với độ phổ biến và cộng đồng lớn mạnh, Rails có được một đội ngũ hỗ trợ giải quyết các vấn đề trong việc lập trình hiệu quả. [7]

Tuy nhiên Rails cũng có nhiều nhược điểm. Các nhược điểm của Rails chủ yếu liên quan tới tốc độ. Do kế thừa từ ngôn ngữ Ruby, cùng với đó là nhiều thư viện, điều này khiến cho tốc độ hoạt động của Rails chậm hơn so với các ngôn ngữ hay framework xây dựng trang web khác. Vấn đề tiếp theo là quá trình khởi động của Rails chậm tùy thuộc vào việc dùng nhiều thư viện với nhiều mã nguồn. Một vấn đề khác là do cung cấp nhiều công cụ tiện lợi nên nhiều khi việc gỡ lỗi sẽ trở nên khó khăn do khó xác định được nguyên nhân đến từ đâu.

Dựa vào những ưu điểm trên đây, cùng với việc lựa chọn ngôn ngữ Ruby là ngôn ngữ lập trình, để xây dựng sản phẩm trên nền tảng web, em lựa chọn Ruby on Rails là framework để phát triển sản phẩm trong đồ án này.

4.

Công cụ xử lý tác vụ ngầm

1.

Khái niệm tác vụ ngầm

Tác vụ ngầm là những công việc thực thi các tác vụ hoạt động độc lập với giao diện người dùng. Đây là những công việc có thể được thực thi mà không cần có sự tương tác của người dùng. Các công

việc này thường là những công việc có mức độ quan trọng không cao, hoặc quan trọng nhưng không cần thiết phải có phản hồi ngay trong quá trình xử lý yêu cầu từ người dùng. Khi đó việc tách các công việc này ra thực hiện riêng và gửi trả kết quả lại sau khi sử dụng một tiến trình riêng để xử lý sẽ đem lại trải nghiệm người dùng tốt hơn. [8]

Có hai kiểu được sử dụng để kích hoạt một tác vụ ngầm hoạt động. Kiểu đầu tiên là Event-driven triggers (khởi tạo dựa trên sự kiện), các tác vụ kiểu này thường không biết được thời điểm chính xác sẽ được sinh ra và thực thi. Do đó các công việc này sẽ được khởi tạo sau khi xảy ra một sự kiện thuộc thể loại nào đó. Một ví dụ cho việc này là tác vụ gửi thông báo cho ứng viên trong quá trình ứng tuyển khi một đơn ứng tuyển của họ bị từ chối vì đã quá hạn duyệt. Điểm mạnh của kiểu này là có khả năng chạy song và mở rộng được.

Kiểu thứ hai là Schedule-driven triggers (khởi tạo dựa trên lịch trình), các tác vụ thuộc loại này thường đã xác định trước được thời điểm chính xác sẽ thực thi công việc này. Các công việc này sẽ được lưu trữ và kích hoạt theo lịch trình đã được đề

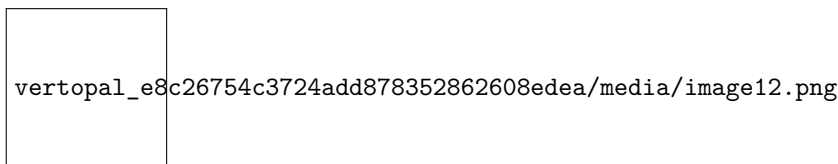
25

4. CÔNG CỤ XỬ LÝ TÁC VỤ NGẦM

ra. Một ví dụ cho việc này là báo cáo kết quả hoạt động hàng tháng của một công ty nào đó. Trái ngược với kiểu trên, do có lịch trình cố định, nên tác vụ này không thể thực hiện chạy song song và khó mở rộng được.

Vì vậy việc sử dụng các tác vụ ngầm là cần thiết trong quá trình hoạt động của hệ thống để có thể giảm đi thời gian phản hồi cho người dùng, kèm theo đó là chuyển một phần công việc không cần sự tương tác của người dùng ra một tiến trình khác để nâng cao hiệu suất hoạt động của hệ thống là cần thiết. Tuy nhiên không phải tác vụ nào cũng có thể thực hiện điều này.

Công cụ xử lý tác vụ ngầm: Sidekiq



Hình 3.4: Logo của Sidekiq⁴

Sidekiq là một công cụ được phát triển để tiến hành xử lý các tác vụ ngầm, được xây dựng bằng ngôn ngữ lập trình Ruby bởi Mike Perham được phát hành lần đầu vào năm 2012. Sidekiq là công cụ

hỗ trợ thực thi các tác vụ ngầm, là phần mềm mã nguồn mở. Sidekiq được xây dựng có khả năng tích hợp dễ dàng với framework Ruby on Rails, tuy nhiên cũng có thể sử dụng với các nền tảng khác. Ngoài phiên bản miễn phí kể trên, chỉ hỗ trợ việc thực hiện thực thi các tác vụ chạy ngầm, nhà phát triển còn cung cấp các phiên bản trả phí là bản mở rộng, trong đó tính năng lập lịch trình cho tác vụ là tính năng nổi bật. Hình 3.4 là logo thư viện Sidekiq. [9]

Sidekiq sử dụng Redis làm nơi lưu trữ các tác vụ để xử lý. Sidekiq cho phép hoạt động đa luồng để tăng tốc độ xử lý các tác vụ nhanh chóng. Tuân theo mô hình khách – chủ (client – server), từng phần của Sidekiq đảm nhận các chức năng riêng biệt từ việc khởi tạo và đẩy tác vụ lưu trữ trong hàng đợi, và lấy các tác vụ từ trong hàng đợi ra để thực thi. Nhờ có sự lưu trữ này mà các tác vụ được xử lý bởi Sidekiq có thể dễ dàng được khởi động lại, hoặc có thể xóa khi thiết lập các thuộc tính ban đầu cần thiết cho chúng.

Sidekiq phiên bản thông thường có hỗ trợ việc thiết lập thời gian để xử lý tác

4Nguồn: <https://sidekiq.org/assets/2018/horizontal.svg>

26

4. CÔNG CỤ XỬ LÝ TÁC VỤ NGẦM

vụ. Tác vụ được tạo ra có thể không được thực thi ngay mà được hoãn đến một thời điểm xác định để thi hành tác vụ này. Ngoài ra các tác vụ có thể được đánh mức độ ưu tiên để xác định thứ tự thực thi các tác vụ sao cho hiệu quả cao nhất.

Tổng kết lại, Sidekiq là một thư viện để xử lý các tác vụ ngầm, được phát triển bởi ngôn ngữ Ruby nên có tính tương thích cao với framework Ruby on Rails trong quá trình vận hành sử dụng. Các tính năng cùng với việc hỗ trợ xử lý đa luồng chính là điểm mạnh khiến cho Sidekiq có tốc độ xử lý nhanh, hoạt động ổn định và hiệu quả. Sidekiq cũng có hỗ trợ giao diện thân thiện với người sử dụng công cụ này.

Công nghệ lưu trữ các tác vụ ngầm: Redis



Hình 3.5: Logo của Redis⁵

Redis là một kho lưu trữ cấu trúc dữ liệu bên trong bộ nhớ của máy tính được phát triển và giới thiệu lần đầu vào năm 2009 bởi Salvatore Sanfilippo và được xây dựng bằng ngôn ngữ lập trình C. Redis được xây dựng để hoạt động trên các nền tảng hệ điều hành họ Unix. Redis hỗ trợ phát triển với nhiều ngôn ngữ lập trình khác nhau như C, C++, Java, PHP, Python, Ruby, Scala. Hình 3.5 là logo của Redis.

Redis là kho lưu trữ cơ sở dữ liệu dưới dạng key - value (khóa - giá trị). Do đặc trưng lưu trữ trong RAM – bộ nhớ trong nên tốc độ truy xuất nhanh. Tuy nhiên đặc tính của bộ nhớ trong là dữ liệu sẽ bị mất nếu như xảy ra sự cố như mất điện nên Redis có tính năng xuất dữ liệu ra ổ đĩa theo chu kỳ hoặc theo một khối lượng công việc nhất định để đảm bảo khả năng phục hồi nếu như có sự cố xảy ra. [10]

Redis được sử dụng làm nơi lưu trữ các tác vụ được xử lý bằng Sidekiq. Khi đó Sidekiq sẽ sử dụng Redis như một hàng đợi để lưu trữ các tác vụ theo mô hình First In First Out (FIFO – nghĩa là tác vụ nào đến trước sẽ được lấy ra và xử lý trước). Việc tận dụng được tốc độ truy xuất của Redis và khả năng xử lý đa luồng của Sidekiq đã trở thành yếu tố quyết định tính hiệu quả khi sử dụng bộ đôi này nhằm

⁵Nguồn: <https://redis.io>

27

5. KẾT CHƯỠNG

mục đích thực thi các tác vụ ngầm trong hệ thống.

Kết chương

Trong chương này, em đã trình bày lý thuyết và các công nghệ được sử dụng cùng với lý do lựa chọn các công nghệ này trong quá trình phát triển sản phẩm. Những công nghệ đó được áp dụng và phát triển sản phẩm, dẫn đến một sản phẩm hoàn chỉnh sẽ được trình bày trong chương tiếp theo – Chương 4.

28

CHƯƠNG 4. PHÂN TÍCH, THIẾT KẾ VÀ XÂY DỰNG HỆ THỐNG

Tổng quan

Trong chương trước em đã nêu ra và phân tích nguyên do lựa chọn các công nghệ, thư viện, tính năng được áp dụng trong phát triển sản phẩm cho đề án này. Tiếp theo trong chương này em sẽ đi vào phần thiết kế chi tiết hệ thống và sử dụng những công nghệ trên để giải quyết những vấn đề và tiến hành xây dựng hệ thống sản phẩm trong các phần kế tiếp.

2.

Thiết kế kiến trúc

1.

Lựa chọn kiến trúc phần mềm

Kiến trúc được em lựa chọn để xây dựng sản phẩm là kiến trúc MVC (Model – View – Controller). Trong mô hình kiến trúc này, hệ thống được chia thành ba thành lớp nhằm mục đích xử lý các hoạt động của hệ thống trong đó vai trò và chức năng của từng thành phần được mô tả dưới đây. Mô hình được minh họa trong hình 4.1.



vertopal_e8c26754c3724add878352862608edea/media/image14.png

Hình 4.1: Mô hình kiến trúc MVC

- Lớp Model (thành phần M): là nơi xử lý các tương tác với dữ liệu hoặc hệ quản trị cơ sở dữ liệu, thông thường sẽ bao gồm những lớp hay phương thức tạo kết nối với các cơ sở dữ liệu, các chức năng như thêm, sửa, xóa các đối tượng, truy vấn các dữ liệu được lưu trữ bên trong cơ sở dữ liệu, . . .
- Lớp View (thành phần V): là nơi có vai trò lưu trữ các thành phần giao diện như nút bấm, các biểu mẫu, . . . Thành phần này có tác dụng tạo ra các phần giao diện với mục đích hiển thị và giúp người sử dụng có thể tương tác được với hệ thống thông qua các giao diện này.
- Lớp Controller (thành phần C): đây là nơi tiếp nhận những yêu cầu xử lý từ người sử dụng, bao gồm những lớp và phương thức với khả năng xử lý

2. THIẾT KẾ KIẾN TRÚC

các nghiệp vụ logic, điều hướng để lấy ra được những dữ liệu cần thiết từ lớp Model và hiển thị những dữ liệu đó thông qua lớp View. Có thể nói lớp Controller này chính là trung tâm điều khiển các hoạt động của hệ thống. [11]

Mô hình kiến trúc MVC đem lại nhiều ưu điểm trong quá trình phát triển phần mềm. Việc tách rời hệ thống thành ba lớp là ưu điểm chính, khiến cho từng thành phần trong đó có thể phát triển độc lập với nhau, từ đó có thể có các giải pháp riêng để giải quyết cho từng thành phần con trong hệ thống. Điều đó dẫn đến việc mô hình này cho phép lập trình song song các thành phần và tăng tốc độ phát triển sản phẩm. Sự độc lập này đồng thời đem lại khả năng kiểm thử hệ thống một cách dễ dàng hơn, khi Model và View có thể được kiểm thử độc lập. Chính vì vậy, hệ thống phát triển theo MVC sẽ có khả năng bảo trì mã nguồn, mở rộng và phát triển tương đối dễ dàng.

Tuy nhiên mô hình này cũng có một số nhược điểm sau đây. Mô hình không phù hợp với những dự án nhỏ do tách biệt ba thành phần hệ thống, dẫn đến việc làm phức tạp đối với những dự án đó. Nhược điểm tiếp theo là dữ liệu được trung chuyển qua nhiều thành phần bên trong hệ thống, điều này ảnh hưởng trực tiếp tới hiệu năng hoạt động, sẽ gây ảnh hưởng tới những hệ thống cần tốc độ xử lý cao. Một nhược điểm nữa là mô hình này sẽ khó triển khai hơn do tách biệt các thành phần.

Tổng kết lại, tuy có một số nhược điểm nhưng các ưu điểm của mô hình kiến trúc MVC đem lại rất lớn, đặc biệt phù hợp với các sản phẩm được triển khai trên nền tảng web. Nên em quyết định lựa chọn mô hình kiến trúc này là mô hình để phát triển sản phẩm thực hiện trong đồ án này.

Kiến trúc MVC được áp dụng vào trong sản phẩm được xây dựng được giữ nguyên so với mô hình kiến trúc MVC gốc. Chỉ có một sự thay đổi là một thành phần nhỏ là lớp Service, nằm ở trung gian giữa luồng xử lý từ lớp Controller sang Model để giảm tải một phần gánh nặng của lớp Controller nhằm góp phần xử lý một vài logic riêng biệt.

Kiến trúc cụ thể cho ứng dụng được trình bày dưới đây.

Thành phần M (ứng với Model) chứa các lớp có vai trò tương tác với dữ liệu. Các lớp này cũng có khả năng tương tác liên kết với nhau để lấy ra các dữ liệu từ các bảng có quan hệ với nhau trong thiết kế. Một số lớp trong thành phần này như lớp Job – việc làm, lớp Candidate – ứng viên ứng tuyển, lớp Employee – nhân viên công ty, lớp Company – công ty tuyển dụng, . . . Chi tiết về các lớp trong thành phần M sẽ được trình bày trong phần sau của đồ án.

30

2. THIẾT KẾ KIẾN TRÚC

Tiếp đến là thành phần V (ứng với View) chứa các lớp hiển thị dữ liệu ra cho người sử dụng, bao gồm các đối tượng giao diện như nút bấm, biểu mẫu, hình ảnh, menu và cho phép người sử dụng tương tác để gửi các thông điệp về hệ thống. Các giao diện được chia thành các thư mục nhỏ, mỗi thư mục thường ứng với một lớp bên thành phần M và C. Ví dụ một số thư mục trong thành phần này như thư mục jobs – gồm các giao diện liên quan việc làm như hiển thị, biểu mẫu tạo mới, . . . thư mục companies – gồm các giao diện liên quan tới nội dung của công ty, . . . Chi tiết về các thư mục trong thành phần M sẽ được trình bày tiếp trong phần tiếp theo của đồ án.

Sau đó là thành phần C (ứng với Controller) gồm những lớp có vai trò nhận các yêu cầu từ người sử dụng, sau đó điều hướng, xử lý nghiệp vụ logic và nhận kết quả để trả lại cho người sử dụng hệ thống. Các lớp của thành phần C thường tương ứng với một lớp thuộc thành phần M và V. Một số lớp của thành phần này bao gồm Job- sController, CandidatesController, EmployeesController và CompaniesController. Chi tiết về các lớp trong thành phần C sẽ được trình bày tiếp trong phần sau của đồ án.

Một thành phần bổ trợ là thành phần S (là thành phần bổ sung Service), thành phần này có chức năng bổ trợ và giám đốc một phần công việc cho thành phần C. Thành phần này sẽ đảm nhiệm một phần logic của hệ thống và kết nối với thành phần M để lấy ra dữ liệu phù hợp và chuyển lại cho thành phần C. Tổng kết lại, thành phần có vai trò như là một trung gian giữa thành phần C và thành phần M để giảm bớt sự chồng chéo trong thành phần C. Chi tiết sẽ được mô tả trong phần sau của đồ án.

Tổng hợp lại, kiến trúc để phát triển hệ thống gồm ba thành phần

Model, View, Controller và có một thành phần bổ trợ là Service. Trong đó phần View đưa ra các giao diện cho người sử dụng, cung cấp khả năng thao tác trên giao diện để người sử dụng đưa ra các yêu cầu. Các yêu cầu sẽ được gửi tới thành phần Controller, tại đây các yêu cầu được xử lý và điều hướng. Các yêu cầu được điều hướng sẽ chuyển tiếp tới thành phần Model, tại đây các yêu cầu được chuyển hóa thành các truy vấn để lấy dữ liệu từ cơ sở dữ liệu, và các dữ liệu sẽ được đưa quay trở lại thành phần Model để xử lý. Các dữ liệu này sẽ tiếp tục được chuyển trở lại Controller và từ Controller sẽ được đưa ra View để hiển thị cho người sử dụng. Trong quá trình trên, một phần các yêu cầu được xử lý và điều hướng tại Controller sẽ được chuyển cho phần Service như là một trạm trung gian để giảm lượng công việc cho Controller.

31

2. THIẾT KẾ KIẾN TRÚC

1.

Thiết kế tổng quan



Hình 4.2: Biểu đồ gói tổng quan

Hình 4.2 trên đây minh họa cho biểu đồ gói tổng quan của sản phẩm được phát triển. Do sử dụng mô hình kiến trúc MVC làm mô hình gốc, biểu đồ gói tổng quan cũng thể hiện điều đó. Các gói bao gồm những thành phần phụ trách các nhiệm vụ tương đồng và có mối liên hệ chặt chẽ với nhau. Trong đó các gói ở lớp trên phụ thuộc vào lớp dưới và không phụ thuộc theo chiều ngược lại.

Đầu tiên là gói về User, gói này không nằm trong hệ thống mà đại diện cho nhóm đối tượng người sử dụng hệ thống, có khả năng tương tác với hệ thống dựa trên những gì được đưa ra bởi gói View trong quá trình hoạt động.

Kế đến là gói View, gói này có nhiệm vụ lưu trữ, quản lý các đối tượng giao diện như nút bấm, biểu mẫu, . . . Tiếp đến gói này cung cấp khả năng tương tác cho các đối tượng người sử dụng được mô tả trong gói User, nhận những yêu cầu từ người sử dụng chuyển xuống cho gói Controller xử lý, và ngược lại nhận những phản hồi, những dữ liệu từ gói Controller để xử lý hiển thị ra cho người sử dụng. Gói này phụ thuộc vào gói Controller để xử lý được hoạt động của mình.

Tiếp đến là gói Controller, gói này đảm nhận vai trò nhận yêu cầu được chuyển

32

2. THIẾT KẾ KIẾN TRÚC

về từ người dùng, tiến hành điều hướng, xử lý các nghiệp vụ logic cần thiết, và chuyển hướng các yêu cầu xuống dưới cho gói Model phụ trách tiếp các yêu cầu đó. Sau đó, gói sẽ nhận các trả về từ gói Model là các dữ liệu cần thiết, tiếp tục xử lý nghiệp vụ logic và chuyển kết quả lên phía trên cho gói View thực hiện tiếp nhiệm vụ của mình. Trong gói Controller được chia nhỏ thành hai gói Controller và gói Service, trong đó gói Controller bên trong có chức năng giống với gói Controller bên ngoài, và gói Service sẽ trợ giúp gói Controller bên trong thông qua việc xử lý một số tác vụ khó khăn, hay tác vụ chạy ngầm. Gói này phụ thuộc vào gói Model để xử lý được hoạt động của mình.

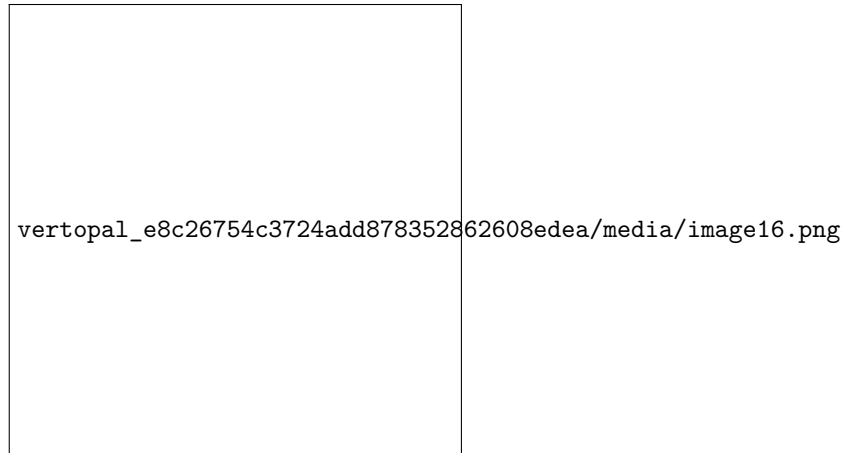
Phía dưới là gói Model, gói này có vai trò là nơi thiết lập liên kết giữa hệ thống với hệ quản trị cơ sở dữ liệu, là nơi xử lý những thao tác liên quan đến dữ liệu như truy vấn nội dung, chỉnh sửa thông tin dữ liệu trong hệ thống. Gói này sẽ nhận những yêu cầu từ gói Controller và sau đó tiến hành các truy vấn, các yêu cầu thực hiện đúng mục đích chuyển xuống cho gói Database xử lý, sau đó nhận lại kết quả, xử lý dữ liệu và truyền lên phía trên cho gói Controller. Gói này phụ thuộc vào gói Database để xử lý được hoạt động của mình.

Cuối cùng là gói Database, gói này chứa hệ quản trị cơ sở dữ liệu, có vai trò lưu trữ và thực hiện các truy vấn dữ liệu. Gói sẽ nhận những

yêu cầu truy vấn từ gói Model, sau đó thực thi các truy vấn và gửi trả lại kết quả lên cho gói Model tiếp tục xử lý.

Trên đây là tổng quan về biểu đồ gói cho hệ thống được phát triển. Trong phần tới em sẽ đi sâu vào chi tiết thiết kế gói cho hệ thống.

Thiết kế chi tiết gói



Hình 4.3: Biểu đồ gói chi tiết cho gói View

33

2. THIẾT KẾ KIẾN TRÚC

Trong hình 4.3 này mô tả biểu đồ gói chi tiết cho gói View trong hệ thống. Gói View này có phụ thuộc vào một số thư viện bên ngoài như thư viện bootstrap và thư viện jquery để giúp tạo giao diện nhanh chóng hơn, hỗ trợ sử dụng và tạo ra những tính năng dùng Javascript nhanh chóng và dễ dàng hơn.

Gói View này được chia ra các lớp theo các nhóm đối tượng hiển thị. Trong đó có một lớp LayoutView bao gồm khuôn mẫu định dạng hiển thị tiêu chuẩn, các lớp giao diện khác sẽ kế thừa và chỉnh sửa các thành phần nội dung phù hợp với lớp đó. Tại đây có lớp LoginView về giao diện đăng nhập sẽ hợp thành bên trong HomePageView là giao diện về trang chủ. Giao diện của ứng viên và nhân viên sẽ được kế thừa chung từ giao diện tài khoản người dùng – lớp AccountsView.



Hình 4.4: Biểu đồ gói Model chi tiết

Kế tiếp là thiết kế chi tiết cho biểu đồ gói Model được minh họa trên hình 4.4. Gói Model có phụ thuộc vào một số thư viện, trong đó một thư viện quan trọng là mysql2, thư viện này có vai trò giúp thiết lập và tạo kết nối giữa gói Model và gói Database để có khả năng truy cập đồng thời thực hiện các truy vấn với hệ quản trị cơ sở dữ liệu MySQL được đặt nằm trong gói Database này.

Gói Model bao gồm các lớp liên quan tới quản lý tài khoản người dùng, là lớp Account và kế thừa là lớp Candidate đại diện cho ứng viên và lớp Employee đại diện cho nhân viên. Tiếp theo là các lớp liên quan tới việc làm như lớp Company, lớp Job, lớp Category. Kế tiếp là lớp Apply đại diện cho các đơn ứng tuyển do ứng viên ứng tuyển vào một việc làm nào đó. Lớp Invite đại diện cho các đơn mời ứng

34

2. THIẾT KẾ KIẾN TRÚC

tuyển. Các mối liên hệ kế thừa và kết tập giữa các lớp được thể hiện trong hình 4.4.



Hình 4.5: Biểu đồ gói Model chi tiết

Cuối cùng là biểu đồ gói chi tiết cho gói Controller, được minh họa trên hình

4.5. Trong gói này, gói được chia nhỏ thành hai gói bên trong, gồm gói Controller trong và gói Service. Gói này có các lớp có tính tương ứng với các lớp trong gói View và gói Model có liên hệ với gói. Ví dụ lớp JobsController là lớp điều khiển luồng xử lý logic của việc làm, có mối liên hệ với lớp JobsView bên gói View và liên hệ với lớp Job trong lớp Model để tạo thành một liên kết xuyên suốt.

Gói Controller bên trong chứa những lớp xử lý nghiệp vụ logic cơ bản ứng với các lớp bên dưới gói Model và các lớp trên bên gói View. Trong đó có lớp AccountsController quản lý tài khoản, và được kế thừa với hai lớp CandidatesController quản lý bên ứng viên cùng với EmployeesController quản lý bên nhân viên. Các lớp controller còn lại tương đối độc lập với nhau. Còn lại gói Service chứa những nghiệp vụ có thể được thực hiện riêng hay thực hiện bên ngoài luồng chính, giúp giảm tải công việc cho bên Controller. Trong gói Service này có một vài lớp như lớp SendNotification gửi các thông báo đến cho các tài khoản của người sử dụng, lớp RejectPastDeadline là lớp phụ trợ để kiểm tra các đơn ứng tuyển quá hạn duyệt sẽ bị từ chối, lớp RecommendJobs có tác dụng tạo danh sách việc làm gợi ý cho người sử dụng.

3. THIẾT KẾ CHI TIẾT

1.

Thiết kế chi tiết

1.

Thiết kế giao diện

Các thông số của màn hình mà ứng dụng hướng tới được đặc tả trong bảng 4.1. Các màn hình được sử dụng bao gồm cả màn hình máy tính để bàn và màn hình máy tính xách tay.

| Tính chất | Giá trị |
|-------------------------|--------------------------------------|
| Kích cỡ | 14 inch, 15.6 inch, 19 inch, 20 inch |
| Độ phân giải | 1280x720, 1366x768, 1920x1080 |
| Tỉ lệ màn hình | 16:9, 4:3 |
| Số lượng màu sắc hỗ trợ | 16 triệu |
| Tần số quét | 60 Hz |

Bảng 4.1: Đặc tả thông tin về màn hình

Các chuẩn hóa thiết kế giao diện với các đối tượng được mô tả dưới đây.

Về nút bấm, nút bấm có kích thước phù hợp với nội dung bên trong nút bấm. Màu sắc của nút bấm được thể hiện cho các chức năng tương ứng, trong đó màu xanh bao gồm các chức năng lên quan đến xác nhận, chuyển trạng thái, nút bấm màu xám đặc trưng cho không thực hiện hành động, còn màu đỏ thể hiện việc thực hiện các hành động như xóa hay từ chối một nội dung nào đó.

Các thông báo được đưa ra màn hình ở góc trên bên phải. Các thông báo này chỉ xuất hiện trong vòng vài giây và tự động biến mất. Màu sắc của thông báo thể hiện được kết quả của các hành động trước đó. Màu xanh lá thể hiện hành động trước đó đã được thực thi thành công, màu đỏ thể hiện hành động trước đó đã thất bại hoặc có xảy ra lỗi trong quá trình thực thi, còn màu xanh dương thường mang nội dung thông báo một thông tin nào đó, không nghiêng về thành công hay thất bại trong quá trình thực hiện hành động.

Giao diện có màu sắc tươi sáng, tông màu chính là màu trắng, một số màu phụ đạo chủ yếu là màu xanh nhạt để giúp cho hiển thị trở nên rõ ràng. Các biểu mẫu được thiết kế căn lề phân tách giữa cột tên các trường và cột điền nội dung. Một số biểu mẫu được thiết kế

dưới dạng một popup hiển thị để không cần chuyển trang trong quá trình sử dụng hệ thống.

Dưới đây là một số hình ảnh minh họa thiết kế giao diện. Đầu tiên là hình 4.6 minh họa phần tìm kiếm việc làm trên giao diện trang chủ, kèm theo đó là minh họa về phần tiêu đề của trang web.

36

3. THIẾT KẾ CHI TIẾT



Hình 4.6: Minh họa giao diện trang chủ phần tìm kiếm

Tiếp đến, hình 4.7 minh họa giao diện trang việc làm, nội dung của trang được phân bổ thành các vùng bao gồm tên và hình ảnh, khu vực mô tả thông tin chi tiết của việc làm và khu vực mô tả các thông tin khác.



Hình 4.7: Minh họa giao diện trang việc làm

Cuối cùng, hình 4.8 là giao diện minh họa cho biểu mẫu được thiết kế trong sản phẩm, các biểu mẫu sẽ tuân theo thiết kế này, chỉ thay đổi nội dung các trường và nội dung ô nhập dữ liệu sẽ phụ thuộc vào kiểu dữ liệu của trường.

37

3. THIẾT KẾ CHI TIẾT



Hình 4.8: Minh họa giao diện trang biểu mẫu

Thiết kế lớp



Hình 4.9: Biểu đồ lớp thiết kế chi tiết cho lớp Employee và LoginController

Hình 4.9 là thiết kế chi tiết cho lớp Employee và LoginController bao gồm các thuộc tính và các phương thức được sử dụng trong các lớp này. Trong đó lớp LoginController chỉ chứa duy nhất một phương

thức login để xác thực tài khoản của người dùng xem đó có phải là một tài khoản hợp lệ bên trong hệ thống.

Lớp Employee là lớp mô hình hóa thực thể Nhân viên trong hệ thống, lớp này có những thuộc tính mô tả các thông tin cá nhân cơ bản của một người dùng, bao gồm "email", được sử dụng làm tên đăng nhập của tài khoản, "password" là trường mật khẩu, hai trường này là để định danh một tài khoản. Các trường về thông tin

38

3. THIẾT KẾ CHI TIẾT

cá nhân bao gồm "full_name" xác định tên người dùng, "phone_number" lưu trữ số điện thoại, "address" để lưu trữ địa chỉ của người dùng, "birthday" là trường về ngày sinh, "gender" lưu trữ về giới tính. Các trường trên đây cũng tương ứng với lớp Candidate để mô hình hóa Ứng viên có tài khoản sử dụng hệ thống. Ngoài ra lớp Employee có một trường "company" để lưu trữ công ty mà nhân viên đó trực thuộc. Lớp Employee có các phương thức để truy xuất các trường dữ liệu của nhân viên được mô tả trong hình vẽ. Kể đến là hai phương thức để cập nhật dữ liệu cho ứng viên, đó là "updatePassword" là phương thức để cập nhật lại mật khẩu, và "updateInformation" để cập nhật lại các thông tin cá nhân khác. Ngoài ra lớp Employee có một số phương thức lớp dùng để lấy ra các nhân viên, tạo tài khoản nhân viên mới và xác thực tài khoản của nhân viên đăng nhập vào hệ thống.

Để minh họa cho hoạt động của lớp Employee và LoginController, biểu đồ trình tự mô tả cho hoạt động đăng nhập của nhân viên trong hình 4.10 sẽ mô tả luồng truyền thông điệp trong hệ thống để tiến hành use case đăng nhập dưới đây. Luồng này sẽ mô tả hoạt động của người dùng để tiến hành đăng nhập vào hệ thống với tư cách nhân viên của một công ty sử dụng hệ thống này.



Hình 4.10: Biểu đồ trình tự mô tả use case đăng nhập đối với nhân viên

Trong biểu đồ 4.10 trên đây, chúng ta xuất phát từ người sử dụng hệ thống là nhân viên. Tác nhân là nhân viên truy cập vào trang hiển thị đăng nhập thuộc lớp LoginView sau đó điền đầy đủ các thông tin và gửi yêu cầu đăng nhập. Sau khi nhận được yêu cầu đăng nhập, lớp LoginController sẽ thực thi phương thức đăng nhập với các thông tin được tác nhân là nhân viên nhập gửi vào hệ thống, và gọi đến phương thức xác thực người dùng bên lớp Employee. Tại lớp Employee này, lớp sẽ thực thi xác nhận người dùng dựa trên thông tin Controller chuyển xuống, sau khi xác thực thành công sẽ lấy ra thông tin của nhân viên đó và gửi trả lên phía trên cho controller. Từ controller sẽ trả ngược lại kết quả để hiển thị ra view và cuối

39

3. THIẾT KẾ CHI TIẾT

cùng thông báo cho tác nhân (là nhân viên) quá trình thực hiện đăng nhập thành công.

Tiếp đến là thiết kế chi tiết cho lớp Apply, lớp đại diện cho các đơn ứng tuyển trong hệ thống. Biểu đồ thiết kế chi tiết cho lớp Apply và lớp controller tương ứng AppliesController được minh họa trong hình 4.11 sau đây.



Hình 4.11: Biểu đồ lớp thiết kế chi tiết cho lớp Apply và AppliesController

Tại đây, lớp thực thể Apply mô hình hóa cho đơn ứng tuyển trong thực tế. Lớp này cần có những thông tin bao gồm job là thông tin về việc làm mà đơn ứng tuyển này hướng tới, kế đến là "candidate" là ứng viên ứng tuyển cho đơn này. Kế đến là thuộc tính "employee" xác định nhân viên phụ trách đơn ứng tuyển trên nếu có. Các thuộc tính về nội dung bao gồm thuộc tính "status" xác định trạng thái của đơn ứng tuyển trong quá trình xử lý và thuộc tính "attachments" là một mảng để lưu trữ những tệp, hình ảnh được gắn kèm với đơn ứng tuyển này. Phần phương thức của lớp bao gồm các phương thức để lấy ra thông tin các thuộc tính đối với một đơn ứng tuyển cụ thể, cùng với đó là phương thức về cập nhật trạng thái của đơn ứng tuyển và cập nhật các thông tin của đơn ứng tuyển này. Lớp cũng cung cấp một số phương thức lớp bao gồm tạo ra một đơn ứng tuyển mới, lấy ra nội dung của đơn ứng tuyển, hoặc xóa đơn ứng tuyển nếu như người sử dụng có yêu cầu.

Lớp controller tương ứng của lớp Apply là AppliesController có một thuộc tính là apply để lưu trữ đơn ứng tuyển được xét đến trong quá trình hoạt động của controller này. Lớp này cung cấp các phương thức để có thể tạo một đơn ứng tuyển trống để người dùng điền thông tin và sử dụng thông tin đó để tạo đơn ứng tuyển mới. Còn có các phương thức để lấy ra đơn ứng tuyển, cập nhật nội dung của đơn ứng tuyển, và xóa đơn ứng tuyển nếu có yêu cầu.

Để minh họa cho hoạt động của lớp Apply, hình 4.12 dưới đây sẽ mô tả lại trình

3. THIẾT KẾ CHI TIẾT

tự thực hiện trong quá trình ứng viên nộp hồ sơ, hay nộp đơn ứng tuyển vào một việc làm xác định. Luồng này sẽ mô tả hoạt động của ứng viên khi nộp hồ sơ từ khi bắt đầu cho đến khi đơn ứng tuyển được hệ thống nhận.



Hình 4.12: Biểu đồ trình tự mô tả use case nộp đơn ứng tuyển

Trong biểu đồ 4.12 trên đây, ban đầu ứng viên sẽ xuất phát từ trang giao diện của việc làm mà ứng viên muốn ứng tuyển và yêu cầu ứng tuyển. Khi đó yêu cầu sẽ chuyển xuống AppliesController để sinh ra yêu cầu tạo ra một đơn ứng tuyển trống để ứng viên có thể điền các thông tin chi tiết. Sau đó thông tin được truyền xuống lớp Apply xử lý, và trả về đơn trống, cuối cùng chuyển lại trên giao diện để hiển thị cho người dùng biểu mẫu trống để điền thông tin đơn ứng tuyển. Sau khi tác nhân là ứng viên đã điền xong nội dung đơn ứng tuyển, ứng viên sẽ gửi yêu cầu tạo đơn ứng tuyển mới, yêu cầu này được truyền thành thông điệp tới AppliesController và tiếp tục tới Apply. Tại đây yêu cầu được thực hiện và một đơn ứng tuyển mới đã được tạo ra, được lưu trữ lại trên cơ sở dữ liệu. Kết quả sẽ được trả ngược về cho ứng viên với thông báo đơn ứng tuyển đã được hoàn thành.

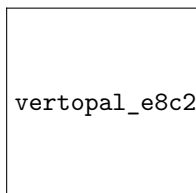
Thiết kế cơ sở dữ liệu

Từ hoạt động của các thực thể tương tác với nhau trong hệ thống, cùng với sự liên kết và thực thi các quy trình nghiệp vụ, em thiết kế sơ đồ thực thể liên kết thể hiện mối liên kết giữa các đối tượng trong

hình 4.13 này.

41

3. THIẾT KẾ CHI TIẾT



Hình 4.13: Sơ đồ thực thể liên kết

Visual Paradigm Online Free Edition

Các thực thể trong hệ thống bao gồm nhân viên (Employee), ứng viên (Candidate), công ty (Company), việc làm (Job), ngành nghề (Category), đơn ứng tuyển (Apply), và các thông báo (Notification).

Các thuộc tính của từng thể đã được mô tả chi tiết trong sơ đồ thực thể liên kết trong hình 4.13. Giữa các thực thể này xuất hiện các liên kết để mô tả mối quan hệ giữa chúng. Đầu tiên là thực thể công ty, thực thể này có liên kết mô tả công ty có nhiều việc làm đang mong muốn tuyển dụng, công ty cũng có một số tài khoản nhân viên được dùng trên hệ thống và trong các nhân viên đó, có một người nắm vai trò quản lý nội dung của công ty trên hệ thống này.

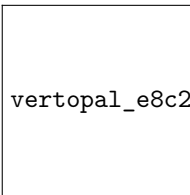
Tiếp đến là các liên kết của việc làm, thực thể việc làm có liên kết thuộc về một công ty nào đó, tiếp đến việc làm có thể thuộc về nhiều ngành nghề khác nhau, mỗi việc làm có thể được giới thiệu cho nhiều ứng viên và mỗi việc làm có thể có nhiều đơn ứng tuyển vào việc làm đó.

Về phía thực thể nhân viên, mỗi nhân viên thuộc về một công ty nào đó, trong số các nhân viên thì có nhân viên có thể làm chức vụ quản lý công ty trên hệ thống. Một nhân viên có thể phụ trách nhiều đơn ứng tuyển vào công ty của mình, và nhân viên có thể có nhiều thông báo để hiện thị.

42

3. THIẾT KẾ CHI TIẾT

Về phía ứng viên, mỗi ứng viên có thể quan tâm đến nhiều lĩnh vực ngành nghề khác nhau. Mỗi ứng viên có thể có nhiều đơn ứng tuyển vào nhiều việc làm khác nhau, và ứng viên có thể được giới thiệu ứng tuyển nhiều việc làm. Ứng viên cũng có nhiều thông báo để hiển thị. Về thực thể đơn ứng tuyển, một đơn ứng tuyển thuộc về một việc làm, thuộc về một ứng viên ứng tuyển, và có thể do một nhân viên phụ trách xử lý.



vertopal_e8c26754c3724add878352862608edea/media/image27.png

Hình 4.14: Thiết kế cơ sở dữ liệu cho hệ thống

Từ sơ đồ thực thể liên kết trong hình 4.13, em chuyển sang thiết kế cơ sở dữ liệu cho hệ thống được phát triển. Hệ thống được xây dựng trên hệ quản trị cơ sở dữ liệu quan hệ sử dụng SQL. Thiết kế các bảng được vẽ ra trên hình 4.14. Trong đó các bảng có các thuộc tính được liệt kê trong sơ đồ thực thể liên kết. Mỗi bảng đều được bổ sung thêm hai trường "created_at" và "updated_at" có kiểu dữ liệu timestamp để lưu trữ lại thời điểm tạo bản ghi và thời điểm cập nhật bản ghi. Đối với hai thực thể nhân viên (Employee) và ứng viên (Candidate) có nhiều điểm chung, nên khi

43

3. THIẾT KẾ CHI TIẾT

thiết kế chuyển sang cơ sở dữ liệu sẽ khởi tạo chung một bảng là bảng accounts để lưu trữ các tài khoản, và sử dụng trường type để phân biệt hai thực thể trên. Các thông tin bổ sung sẽ được thêm vào bảng riêng và có khóa ngoài liên kết quay trở lại bảng này, ví dụ nhưng thông tin bổ sung của ứng viên sẽ được thêm trong bảng candidate_infos, còn thông tin nhân viên làm việc cho công ty nào sẽ được thể hiện trong bảng employee_companies.

Các liên kết giữa các thực thể trên sơ đồ thực thể liên kết nếu là dạng liên kết một-một, hay một-nhiều sẽ được chuyển thành khóa ngoài để liên kết giữa hai bản ghi thuộc hai bảng. Còn các liên kết dạng nhiều-nhiều sẽ được chuyển thành một bảng mới. Các bảng như vậy bao gồm job_invites để lưu trữ những lời mời ứng tuyển đến ứng viên, bảng job_categories để làm tham chiếu ngành nghề cho việc làm và bảng candidate_categories để lưu trữ những ngành nghề mà ứng viên quan tâm.

44

4. XÂY DỰNG ỨNG DỤNG

2.

Xây dựng ứng dụng

1.

Thư viện và công cụ sử dụng

| Mục đích | Công cụ | Phiên bản |
|----------------------------|------------------------------|-----------|
| IDE lập trình | Microsoft Visual Studio Code | 1.69.2 |
| Ngôn ngữ | Ruby | 2.7.3 |
| Framework | Ruby on Rails | 6.1.5 |
| Cơ sở dữ liệu | MySQL | 5.7.37 |
| Quản lý phiên bản mã nguồn | git | 2.17.1 |
| Thư viện liên kết CSDL | mysql2 | 0.5.3 |
| Xử lý tác vụ ngầm | sidekiq | 6.4.2 |
| Thư viện lập lịch tác vụ | sidekiq-scheduler | 4.0.2 |
| Lưu trữ tác vụ | Redis | 4.0.9 |
| Kết nối đến Redis | redis | 4.6.0 |
| Thư viện mã hóa | bcrypt | 3.1.13 |
| Thư viện tìm kiếm | ransack | 3.1.0 |
| Hỗ trợ kiểm thử | rspec-rails | 4.0.1 |
| Tạo dữ liệu mẫu | faker | 2.1.2 |
| Khởi tạo đối tượng mẫu | factory-bot-rails | 6.2.0 |
| Web server cho Ruby | puma | 5.6.4 |
| Phân trang | kaminari | 1.2.2 |
| Kết nối jquery với Rails | jquery-rails | 4.4.0 |
| Hỗ trợ javascript | jquery | 3.6.0 |
| Hỗ trợ html | bootstrap | 4.5.0 |
| Bootstrap cho phân trang | bootstrap4-kaminari-views | 1.0.1 |
| Quản lý biến môi trường | figaro | 1.2.0 |
| Hỗ trợ hiển thị thông báo | toastr-rails | 1.0.3 |

| Mục đích | Công cụ | Phiên bản |
|---------------------|---------|-----------|
| Công cụ gỡ lỗi | pry | 0.14.1 |
| Thiết lập tham số | config | |
| Công cụ quản lý gói | yarn | 4.0.0 |
| | | 1.22.18 |

Bảng 4.2: Danh sách thư viện và công cụ sử dụng

Bảng 4.2 trên đây liệt kê những danh sách thư viện và các công cụ quan trọng nhất được sử dụng để tiến hành phát triển sản phẩm. Các công cụ và thư viện được chia ra thành các nhóm bao gồm nhóm hỗ trợ và lập trình các chức năng (gồm liên quan tới ngôn ngữ lập trình, framework, công cụ để lập trình), nhóm thiết lập và liên kết với cơ sở dữ liệu, nhóm hỗ trợ xử lý giao diện (liên quan tới các thư viện hiển thị, tạo nút bấm) và nhóm hỗ trợ các hoạt động xử lý hoạt động cốt lõi, triển khai hệ thống (thư viện server), ngoài ra còn sử dụng thêm công cụ quản lý mã nguồn (git) để lưu trữ và quản lý các phiên bản giúp quá trình bảo trì và nâng cấp trở nên dễ dàng hơn.

45

4. XÂY DỰNG ỨNG DỤNG

1.

Kết quả đạt được

Kết quả thu được khi thực hiện xây dựng sản phẩm về trang web môi giới việc làm được thực hiện trong đồ án này là một hệ thống phần mềm được triển khai trên nền tảng web. Hệ thống được chia thành các phần, bao gồm các hệ thống kết nối bên ngoài gồm khu vực hệ quản trị cơ sở dữ liệu dùng hệ quản trị MySQL để lưu trữ dữ liệu của hệ thống, khu vực kho lưu trữ dữ liệu trong quá trình xử lý tác vụ ngầm Redis có nhiệm vụ làm hàng đợi lưu trữ các tác vụ ngầm khi thư viện Sidekiq tiến hành tạo và thực thi các tác vụ đó, đây là hai thành phần kết nối ra bên ngoài, cần cài đặt riêng.

Các thành phần bên trong được chia thành các gói theo cấu trúc có sẵn của framework Ruby on Rails để phát triển. Đầu tiên là gói config bao gồm những thiết lập cho hệ thống bao gồm các cài đặt ban đầu và khởi tạo các kết nối với ứng dụng bên ngoài bao gồm cả cơ sở dữ liệu, nơi đây cũng quản lý danh sách các biến môi trường được sử dụng trong vận hành hệ thống. Kế đến là gói db lưu trữ lại thiết kế của cơ sở dữ liệu được sử dụng trong hệ thống, trong đây còn có mô tả về việc tạo dữ liệu mẫu cho sản phẩm được phát

triển. Tiếp đó là một vài gói dùng để lưu trữ về các thư viện ngoài không được hỗ trợ sẵn trong framework. Cuối cùng là gói app chứa mã nguồn của sản phẩm được đóng gói.

Trong gói app được chia ra thành các gói nhỏ để lưu trữ những thành phần quan trọng trong quá trình phát triển. Đầu tiên là gói controllers để lưu trữ những controller có tác dụng điều hướng và phân luồng, đón nhận những yêu cầu của người dùng được phát triển, tương ứng với gói controller trong quá trình thiết kế biểu đồ gói tổng quan. Kế đến là gói models dùng để lưu trữ những model mô tả dữ liệu các lớp và các đối tượng tham gia hệ thống, tương ứng với gói model trong thiết kế biểu đồ gói tổng quan. Tiếp đến là gói views dùng để lưu trữ giao diện để hiển thị ra các thiết bị cuối phía người sử dụng, tương ứng với gói views trong thiết kế biểu đồ gói. Kế đến sau đó là gói services, cung cấp những dịch vụ hỗ trợ cho hoạt động của controller, được mô tả đến trong biểu đồ gói tổng quan. Ngoài ra có gói assets lưu trữ những tệp định dạng cho giao diện dưới dạng scss giúp cho việc thiết kế giao diện trở nên hiệu quả hơn. Còn có gói jobs dùng để lưu trữ những tác vụ thêm, trong đồ án này là nơi để tạo ra những tác vụ ngầm được thực thi trong hệ thống.

Bảng 4.3 dưới đây sẽ thống kê lại các thông tin về ứng dụng đã được phát triển trong đồ án theo một số tiêu chí quan trọng.

46

4. XÂY DỰNG ỨNG DỤNG

| Thông tin | Giá trị |
|-----------------------------|---------|
| Số dòng code | 7700 |
| Số lớp | 36 1 |
| Số gói | 12 |
| Dung lượng toàn bộ mã nguồn | 4.4 MB |

Bảng 4.3: Thông tin thống kê về ứng dụng

Minh họa các chức năng chính

Một số màn hình giao diện trong sản phẩm sẽ được liệt kê dưới đây.

Đầu tiên là hình 4.15 bao gồm một phần của màn hình trang chủ, trong đó có thanh tiêu đề cùng kèm theo hiển thị số lượng thông báo người dùng đăng nhập chưa đọc. Giao diện này được bố trí thêm từng phân vùng để hiển thị một số công việc thuộc các lĩnh vực khác nhau.



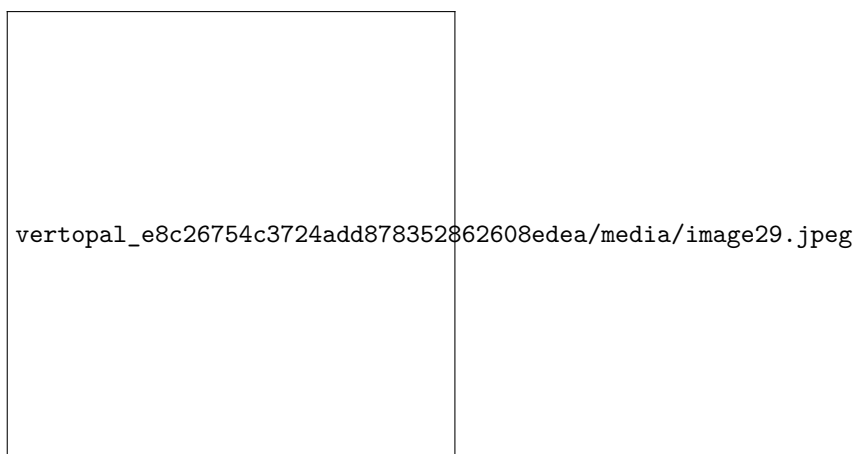
Hình 4.15: Hình ảnh trang chủ với danh sách việc làm theo chủ đề

Tiếp đến là hình 4.16 là giao diện của trang thông tin chi tiết của việc làm. Trang thông tin này chia ra thành các vùng gồm vùng hiển thị hình ảnh, vùng đi vào chi tiết nội dung việc làm và vùng gồm các thông tin bổ trợ như thời gian ứng tuyển.

Chỉ tính trên các gói chính theo kiến trúc MVC

47

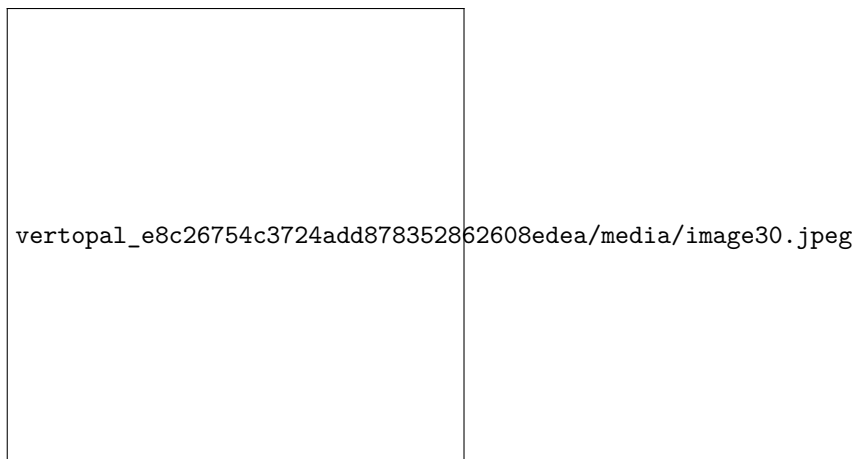
5. KIỂM THỬ



Hình 4.16: Hình ảnh trang việc làm

Một màn hình khác là màn hình cập nhật thông tin của việc làm được mô tả trong hình 4.17. Giao diện này tuân theo nội dung thiết kế giao diện của biểu mẫu, trong đó các thông tin về việc làm có thể cập nhật, kiểu dữ liệu để cập nhật sẽ ảnh hưởng đến phương pháp

hiển thị và cách thức cập nhật của từng trường trong nội dung chi tiết của việc làm.



Hình 4.17: Hình ảnh trang chỉnh sửa thông tin việc làm

Kiểm thử

Để kiểm tra việc hệ thống hoạt động đúng đắn có như mong muốn theo thiết kế ban đầu hay không, chúng ta cần phải kiểm thử các chức năng của hệ thống được phát triển. Mục tiêu cần kiểm tra xem các chức năng có hoạt động đúng hay không, có gặp phải lỗi trong quá trình vận hành hay không. Các kết quả hiển thị từ hệ thống có đúng như mong muốn hay không, các thông báo lỗi có chuẩn xác không. Một số chức năng quan trọng được kiểm thử sẽ được nêu ra trong các phần tiếp sau đây.

48

5. KIỂM THỬ

1.

Kiểm thử chức năng đăng nhập

Trường hợp 1: Người dùng nhập đúng tên đăng nhập và mật khẩu, sau đó nhấn nút đăng nhập. Kết quả mong muốn: người dùng đăng nhập vào hệ thống và hệ thống thông báo đăng nhập thành công.

Trường hợp 2: Người dùng nhập sai tên đăng nhập hoặc nhập sai mật khẩu, sau đó nhấn nút đăng nhập. Kết quả mong muốn: người dùng không đăng nhập được vào hệ thống và hệ thống thông báo đăng nhập thất bại, đồng thời hiển thị lại trang đăng nhập.

Trường hợp 3: Người dùng nhập thiếu một trong hai trường tên đăng nhập hoặc mật khẩu, sau đó nhấn nút đăng nhập. Kết quả mong muốn: người dùng không đăng nhập được vào hệ thống và hệ thống thông báo đăng nhập thất bại, đồng thời hiển thị lại trang đăng nhập.

Trong việc kiểm thử chức năng đăng nhập này, kĩ thuật kiểm thử bảng quyết định được sử dụng để thiết kế ra các trường hợp có thể sinh ra từ việc nhập nội dung các trường cần thiết để tiến hành đăng nhập vào hệ thống.

Kiểm thử chức năng cập nhật thông tin việc làm

Để kiểm thử tính năng này, cần phải đăng nhập với tài khoản nhân viên và truy cập vào trang cập nhật của một việc làm tuyển dụng từ công ty nơi nhân viên đó trực thuộc

Trường hợp 1: Nhân viên nhập đầy đủ thông tin hợp lệ, tiến hành cập nhật thông tin chi tiết của việc làm. Kết quả mong muốn: cập nhật thành công, hệ thống thông báo cập nhật thành công và chuyển về trang thông tin việc làm kèm các thông tin đã được cập nhật.

Trường hợp 2: Nhân viên xóa một trường cần thiết, tiến hành nhấn nút cập nhật việc làm. Kết quả mong muốn: cập nhật thất bại, hệ thống thông báo cảnh báo thất bại ra màn hình.

Trường hợp 3: Nhân viên điền giá trị không nằm trong miền giá trị của trường thông tin, ví dụ với số lượng người ứng tuyển, nhân viên điền giá trị -1, sau đó nhấn nút cập nhật việc làm. Kết quả mong muốn: cập nhật thất bại, hệ thống thông báo giá trị cập nhật không hợp lệ.

Trường hợp 4: Nhân viên điền giá trị không đúng kiểu dữ liệu với kiểu dữ liệu của trường thông tin, ví dụ với số người ứng tuyển, nhân viên điền giá trị là chuỗi “một”, sau đó nhấn nút cập nhật việc làm. Kết quả mong muốn: cập nhật thất bại, hệ thống thông báo giá trị cập nhật không hợp lệ.

49

5. KIỂM THỬ

Trong việc xây dựng các chức năng để kiểm thử tính năng trên, các kĩ thuật kiểm thử bảng quyết định, cùng với kiểm thử giá trị biên dùng để xác định giới hạn cận trên dưới của miền giá trị đã được áp dụng.

Kiểm thử chức năng nộp đơn ứng tuyển cho ứng viên

Để kiểm thử chức năng, đầu tiên cần phải truy cập vào trang thông tin của một việc làm nào đó.

Trường hợp 1: Người sử dụng chưa đăng nhập, tiến hành vào trang thông tin chi tiết của một việc làm, nhấn vào nút ứng tuyển. Kết quả mong muốn: hệ thống chuyển hướng đến trang đăng nhập và yêu cầu người dùng cần đăng nhập vào hệ thống.

Trường hợp 2: Người sử dụng đăng nhập với tư cách nhân viên của một công ty, tiến hành vào trang thông tin chi tiết của một việc làm. Kết quả mong muốn: nút ứng tuyển không xuất hiện.

Trường hợp 3: Người sử dụng đăng nhập với tư cách ứng viên, tiến hành vào trang thông tin chi tiết của một việc làm, nhấn vào nút ứng tuyển, điền đủ thông tin yêu cầu vào đơn ứng tuyển. Hiện tại ứng viên này chưa ứng tuyển vào việc làm trên. Kết quả mong muốn: ứng viên ứng tuyển thành công, hệ thống thông báo ứng tuyển thành công và trong danh sách đơn ứng tuyển ứng viên đã tham gia có đơn ứng tuyển này.

Trường hợp 4: Người sử dụng đăng nhập với tư cách ứng viên, tiến hành vào trang thông tin chi tiết của một việc làm, nhấn vào nút ứng tuyển, điền đủ thông tin yêu cầu vào đơn ứng tuyển. Ứng viên này trước đó đã ứng tuyển vào việc làm trên và đơn đang trong quá trình ứng tuyển. Kết quả mong muốn: ứng viên ứng tuyển thất bại, hệ thống thông báo ứng viên đã ứng tuyển vào việc làm này trước đó.

Trường hợp 5: Người sử dụng đăng nhập với tư cách ứng viên, tiến hành vào trang thông tin chi tiết của một việc làm, nhấn vào nút ứng tuyển, điền không đầy đủ thông tin yêu cầu. Kết quả mong muốn: ứng viên ứng tuyển thất bại, hệ thống thông báo ứng tuyển không thành công.

Trong việc kiểm thử chức năng này, kĩ thuật kiểm thử bằng quyết định để xác định các trường hợp và kĩ thuật kiểm thử bằng trạng thái để xác định thay đổi của đầu ra phụ thuộc vào trạng thái của các đối tượng đã được áp dụng để tiến hành xây dựng các trường hợp kiểm thử trong hệ thống.

Tổng kết phần kiểm thử chức năng

Các chức năng được kiểm thử theo nhiều kĩ thuật kiểm thử khác nhau, bao gồm cả kĩ thuật kiểm thử hộp đen và kiểm thử hộp trắng, cùng với đó là một số kĩ thuật

6. TRIỂN KHAI

dùng để xây dựng các kịch bản kiểm thử cho các tính năng trong hệ thống như kiểm thử sử dụng phân tích giá trị biên, sử dụng bảng quyết định để tạo ra những tham số đầu vào xây dựng lên những trường hợp kiểm thử cụ thể mà vẫn đảm bảo được tính hiệu quả.

Tổng kết quá trình kiểm thử, tổng cộng có hơn ba mươi trường hợp kiểm thử được xây dựng trong quá trình kiểm thử chức năng của hệ thống. Không có trường hợp nào trong đó có kết quả không như mong muốn.

Triển khai

Để triển khai được hệ thống trên thực tế, cần phải có hai server để tiến hành chạy hệ thống. Server đầu tiên là nơi cài đặt hệ quản trị cơ sở dữ liệu, tại đó có cài hệ quản trị cơ sở dữ liệu MySQL là nơi để lưu trữ các dữ liệu trong quá trình hoạt động và vận hành của hệ thống. Server còn lại là nơi để chạy hệ thống, đóng vai trò tiếp nhận các yêu cầu của người dùng, sau đó xử lý và trả lại kết quả cho người dùng thông qua các thiết bị đầu cuối. Hai server này cần liên kết với nhau để có thể truy xuất được dữ liệu, chúng được đặt cùng trong một mạng nội bộ để thuận tiện trong quá trình giao tiếp. Server chạy hệ thống để tiếp nhận xử lý của người dùng cần kết nối mạng internet để có thể nhận được yêu cầu của người sử dụng.

Sau khi thiết lập xong các phần cứng cần thiết để tiến hành hoạt động, cần phải cài các phần mềm để vận hành hệ thống. Cài đặt cơ sở dữ liệu, tiến hành chạy khởi tạo các bảng để lưu trữ dữ liệu trong server đóng vai trò xử lý dữ liệu. Trên server chạy hệ thống, triển khai cài đặt các thư viện được sử dụng, sau đó chạy phần mã nguồn của chương trình để hệ thống hoạt động. Hệ thống sau khi vận hành có thể bắt đầu tiếp nhận các yêu cầu từ người dùng thông qua nền tảng web.

Hệ thống được triển khai với mục tiêu có thể đáp ứng nhu cầu cho hàng trăm người có thể truy cập cùng lúc, được vận hành và bảo trì thường xuyên. Sẽ có những phiên bản cập nhật theo định kỳ vận hành của hệ thống.

Hệ thống hiện chưa được thử nghiệm triển khai trên thực tế.

Kết chương

Trong chương này em đã đề cập đến việc lựa chọn mô hình kiến trúc phát triển sản phẩm. Tiếp đến là phần thiết kế từ tổng quan đến chi tiết cho các thành phần bên trong của sản phẩm được phát triển cùng với đó là xây dựng và kết quả thu được. Hai phần tiếp theo về

cách thức và phương án kiểm thử một số tính năng của sản phẩm cùng với mô hình để triển khai trong thực tế như thế nào.

Trong chương kế tiếp – chương 5, em sẽ trình bày về nội dung đóng góp chính của sản phẩm được thực hiện trong đề án này.

51

CHƯƠNG 5. CÁC GIẢI PHÁP VÀ ĐÓNG GÓP NỔI BẬT

Tổng quan

Phần tiếp theo đây, em sẽ trình bày những giải pháp và đóng góp trong quá trình thực hiện đề án tốt nghiệp này. Trong đó trọng tâm chính là về quy trình ứng tuyển được áp dụng để nâng cao chất lượng và giảm thiểu sự phụ thuộc vào các hệ thống bên ngoài.

2.

Hỗ trợ quá trình ứng tuyển

1.

Giới thiệu vấn đề

Như trong nội dung đã được đề cập trong các phần trước, trong các sản phẩm về đăng tin tuyển dụng, môi giới việc làm hiện tại, hầu hết chúng thường chỉ hỗ trợ khâu nộp hồ sơ và thông báo kết quả bước đầu, một vài hệ thống cho phép cung cấp thêm một số bước trung gian nhưng chưa thật sự tạo ra một quy trình hoàn thiện.

Chính vì thế việc xây dựng ra một sản phẩm hỗ trợ quy trình tuyển dụng hoàn chỉnh để áp dụng trở thành một vấn đề cần được quan tâm, nhờ đó đem lại trải nghiệm tốt hơn cho người sử dụng, góp phần hạn chế sự phụ thuộc vào dịch vụ của một bên thứ ba, hoặc nhà tuyển dụng phải tự xây dựng một nền tảng riêng để thực hiện công việc tuyển dụng nguồn nhân lực.

Giải pháp xây dựng

Biểu đồ hoạt động cho quy trình ứng tuyển đã được thể hiện trong hình 2.8.

Có thể chia quy trình ứng tuyển này thành bốn giai đoạn chính. Giai đoạn đầu tiên là giai đoạn mở đầu, tại đây ứng viên sẽ nộp hồ sơ để ứng tuyển vào việc làm và bên nhân viên của nhà tuyển dụng sẽ tiếp nhận hồ sơ để xét duyệt. Giai đoạn thứ hai – giai đoạn sơ tuyển, trong giai đoạn này bên nhà tuyển dụng sẽ đánh giá hồ sơ của ứng viên, trong trường hợp hồ sơ có cần bổ sung loại giấy tờ gì thì nhân viên sẽ yêu cầu ứng viên bổ sung và tiếp tục sơ tuyển

hồ sơ. Qua giai đoạn sơ tuyển sẽ đến giai đoạn thứ ba – giai đoạn phỏng vấn, trong giai đoạn này bên nhà ứng tuyển sẽ hẹn gặp ứng viên để phỏng vấn thông qua hình thức gặp mặt trực tiếp hoặc qua hình thức trực tuyến, giai đoạn phỏng vấn này có thể có nhiều vòng phỏng vấn khác nhau. Cuối cùng là giai đoạn chấp nhận đơn ứng tuyển, tại giai đoạn này hai bên sau khi đồng ý thì có thể thương thảo về hợp đồng chính thức. Trong cả quy trình ứng tuyển trên đây, hai bên đều có thể dừng tham gia tuyển dụng bất cứ lúc nào, và khi đó quy trình chấm dứt. Chi tiết giải pháp cho từng giai đoạn sẽ được trình bày tiếp sau đây.

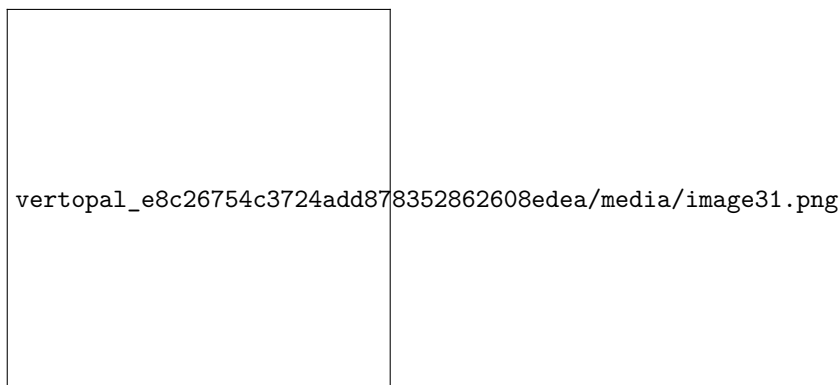
Để tiến hành xác định được quy trình ứng tuyển đang ở trạng thái nào thì trong

2. HỖ TRỢ QUÁ TRÌNH ỨNG TUYỂN

quá trình xây dựng mô hình cho lớp đơn ứng tuyển, chúng ta cần một thuộc tính để xác định được trạng thái đơn ứng tuyển đó đang ở vào giai đoạn đó. Trong thiết kế, thuộc tính này chính là thuộc tính “status” được thể hiện cả trong thiết kế lớp và trong sơ đồ thiết kế cơ sở dữ liệu của hệ thống này.

a, Giai đoạn mở đầu

Đây là giai đoạn mở đầu cho quy trình ứng tuyển. Giai đoạn này gồm một trạng thái của đơn ứng tuyển là trạng thái “Đang chờ”. Hình 5.1 dưới đây là sơ đồ trạng thái minh họa cho giai đoạn đầu tiên của quy trình ứng tuyển đối với đơn ứng tuyển.



Hình 5.1: Sơ đồ trạng thái Giai đoạn mở đầu của đơn ứng tuyển

Để bắt đầu, người tìm việc cần đăng nhập hệ thống với tư cách ứng viên để có thể nộp đơn ứng tuyển vào việc làm mà họ mong muốn. Sau khi điền đầy đủ thông tin và nộp hồ sơ thì một đơn ứng tuyển mới đã được tạo ra với trạng thái “Đang chờ” với ý nghĩa đơn ứng

tuyển đang chờ được nhân viên nhận để sơ tuyển. Trong trạng thái này, ứng viên có thể cập nhật đơn ứng tuyển tùy theo mong muốn, hoặc có thể xóa đơn nếu không còn nhu cầu nữa. Hành động này có thể áp dụng cho tới khi nhân viên bên nhà ứng tuyển tiếp nhận hồ sơ, khi đó ứng viên không thể cập nhật tự do hoặc xóa đơn.

Sau đó nhân viên truy cập vào trang thông tin chi tiết của việc làm, lựa chọn đơn ứng tuyển trong danh sách các đơn ứng tuyển mới của việc làm đó, chọn xem chi tiết và chọn nhận đơn ứng tuyển để xét duyệt. Khi nhân viên nhận đơn, khi đó đơn ứng tuyển sẽ được cập nhật người phụ trách là nhân viên đó, và lúc này ứng viên không thể tự do cập nhật lại đơn ứng tuyển hoặc xóa đơn được nữa. Sau khi nhận đơn, đơn sẽ được chuyển sang trạng thái “Sơ tuyển” và tiến vào giai đoạn sơ tuyển.

Trong giai đoạn này, có một điểm cần lưu ý là về nội dung của đơn ứng tuyển. Do đối với mỗi ngành nghề, mỗi việc làm khác nhau sẽ có yêu cầu về các loại thông tin, giấy tờ khác nhau, ví dụ như nếu việc làm thuộc ngành nghề công nghệ thông

53

2. HỖ TRỢ QUÁ TRÌNH ỨNG TUYỂN

tin thì cần giấy tờ về tốt nghiệp đại học, các chứng chỉ về hệ thống liên quan thì đối với ngành nghề lái xe cần giấy tốt nghiệp Trung học phổ thông, giấy khám sức khỏe và bằng lái xe. Chính vì vậy nội dung đơn ứng tuyển phải có khả năng thay đổi được theo yêu cầu của nhà tuyển dụng.

Để giải quyết được vấn đề đơn ứng tuyển có thể thay đổi này, em thiết kế thuộc tính cho đơn ứng tuyển gồm các trường mặc định để lưu trữ văn bản và tệp dưới tên dạng “file_field_1”, “text_field_1” để lưu trữ các trường thông tin. Trong lớp việc làm sẽ có một thuộc tính dùng để lưu trữ định dạng biểu mẫu đơn ứng tuyển, tại đó sẽ lưu trữ một song ánh giữa tên trường mà nhà tuyển dụng muốn với tên trường được lưu trữ trong cơ sở dữ liệu để có thể tạo ra biểu mẫu đơn ứng tuyển có thể thay đổi, còn thực tế lưu dưới cơ sở dữ liệu là như nhau.

b, Giai đoạn sơ tuyển

Tiếp sau giai đoạn mở đầu sẽ chuyển sang giai đoạn sơ tuyển. Giai đoạn này bắt đầu sau khi nhân viên đã nhận đơn ứng tuyển từ ứng viên và bắt đầu đánh giá đơn ứng tuyển này. Giai đoạn này gồm hai trạng thái “Sơ tuyển” và “Bổ sung hồ sơ”. Hình 5.2 dưới đây sẽ mô tả sơ đồ trạng thái của đơn ứng tuyển trong giai đoạn sơ tuyển này.



vertopal_e8c26754c3724add878352862608edea/media/image32.png

Hình 5.2: Sơ đồ trạng thái Giai đoạn sơ tuyển của đơn ứng tuyển

Trong trạng thái “Sơ tuyển”, vào lúc này nhân viên sẽ tiến hành đọc đơn ứng tuyển của ứng viên đã nộp. Nhân viên dựa trên các thông tin đã được ứng viên cung cấp trong nội dung đơn ứng tuyển để đánh giá xem liệu đơn ứng tuyển có đạt yêu cầu hay không. Trong trường hợp đạt yêu cầu, nhân viên có thể chuyển đơn ứng tuyển lên trạng thái tiếp theo là “Mời phỏng vấn” và đơn ứng tuyển sẽ được chuyển sang giai đoạn kế tiếp là giai đoạn phỏng vấn. Trong trường hợp đơn ứng tuyển không qua được vòng sơ tuyển, nhân viên sẽ từ chối đơn ứng tuyển này, và đơn ứng tuyển được chuyển sang trạng thái “Từ chối đơn” và kết thúc quy trình ứng tuyển tại đây. Trường hợp nhân viên muốn ứng viên cung cấp lại một số nội dung,

54

2. HỖ TRỢ QUÁ TRÌNH ỨNG TUYỂN

ví dụ như cung cấp thêm các minh chứng cho nội dung đã khai trong đơn, nhân viên có thể yêu cầu ứng viên bổ sung đơn ứng tuyển này, và đơn ứng tuyển được chuyển trạng thái sang “Bổ sung hồ sơ”. Nhân viên sẽ lựa chọn ra các trường nội dung yêu cầu bổ sung, được lấy từ các trường thông tin ban đầu của đơn ứng tuyển để yêu cầu ứng viên cập nhật các thông tin này, đồng thời thiết lập hạn chót để ứng viên có thể thực hiện hành động bổ sung hồ sơ.

Trong trạng thái “Bổ sung hồ sơ”, ứng viên sẽ cần phải cập nhật lại các thông tin đã được nhân viên phụ trách đơn ứng tuyển yêu cầu. Lưu ý, ứng viên chỉ có thể cập nhật được các trường thông tin mà nhân viên lựa chọn, không thể cập nhật các trường ngoài nội dung đó. Sau khi cập nhật thành công các thông tin kể trên, đơn ứng tuyển sẽ lại được chuyển về trạng thái “Sơ tuyển” để nhân viên phụ trách tiếp tục đánh giá hồ sơ. Trong trường hợp quá hạn chót được đặt ra để ứng viên cập nhật thông tin, đơn ứng tuyển sẽ được chuyển trạng thái về “Từ chối đơn” và kết thúc quy trình ứng tuyển.

c, Giai đoạn phỏng vấn

Tiếp theo sau giai đoạn sơ tuyển sẽ đến giai đoạn phỏng vấn. Sơ đồ trạng thái của giai đoạn phỏng vấn sẽ được mô tả trong hình 5.3 dưới đây.



Hình 5.3: Sơ đồ trạng thái Giai đoạn phỏng vấn của đơn ứng tuyển

Trong trạng thái “Mời phỏng vấn”, nhân viên phụ trách sẽ mời ứng viên tham gia phỏng vấn với các thông tin bao gồm hình thức phỏng vấn, thời gian phỏng vấn, nếu gặp mặt trực tiếp sẽ kèm thêm thông tin về địa chỉ nơi gặp, trong trường hợp phỏng vấn trực tuyến sẽ đính kèm đường dẫn đến phần mềm, công cụ hỗ trợ phỏng vấn trực tuyến.

Tại đây ứng viên sẽ có thể chấp nhận lời mời này, khi đó trạng thái của đơn ứng tuyển được chuyển sang “Chấp nhận phỏng vấn”, hoặc từ chối lời mời, trạng thái

55

2. HỖ TRỢ QUÁ TRÌNH ỨNG TUYỂN

khi đó của đơn ứng tuyển được chuyển sang “Từ chối phỏng vấn”.

Trong trạng thái “Từ chối phỏng vấn”, nhân viên sẽ đặt lại lịch hẹn để mời ứng viên phỏng vấn vào một thời điểm khác, hoặc với hình thức phỏng vấn khác.

Trong trạng thái “Chấp nhận phỏng vấn”, nhân viên phụ trách và ứng viên sẽ tổ chức cuộc phỏng vấn với nhau. Sau phỏng vấn, nếu như ứng viên không vượt qua được vòng này thì nhân viên phụ trách sẽ chuyển thành trạng thái “Từ chối đơn” và kết thúc quy trình ứng tuyển tại đây. Còn nếu ứng viên vượt qua được vòng này, nhân viên phụ trách sẽ chuyển trạng thái “Phỏng vấn thành công”.

Tại trạng thái “Phỏng vấn thành công”, nếu như kết thúc quá trình phỏng vấn tại đây, nhân viên sẽ chuyển đơn ứng tuyển sang giai đoạn cuối, giai đoạn chấp nhận. Trong trường hợp đây mới chỉ kết thúc một vòng phỏng vấn và chuẩn bị chuyển sang vòng phỏng vấn tiếp theo (áp dụng cho quy trình ứng tuyển gồm nhiều vòng phỏng vấn), nhân viên sẽ chuyển trạng thái đơn ứng tuyển về lại “Mời phỏng vấn” để phục vụ cho vòng phỏng vấn tiếp theo trong quy trình.

d, Giai đoạn chấp nhận

Giai đoạn cuối cùng trong quy trình là giai đoạn chấp nhận. Sơ đồ trạng thái của giai đoạn này được thể hiện bên trong hình 5.4.



Hình 5.4: Sơ đồ trạng thái Giai đoạn chấp nhận của đơn ứng tuyển

Khi vào giai đoạn này, có nghĩa là nhà tuyển dụng đã chấp nhận đơn ứng tuyển của ứng viên. Đơn ứng tuyển của ứng viên cùng với các giai đoạn phỏng vấn đã được đánh giá đạt yêu cầu và sẵn sàng tuyển dụng ứng viên vào tiếp nhận vị trí đang tuyển.

Tại đây, ứng viên còn lựa chọn để đánh giá lại quyết định xem mình có thật sự muốn nhận công việc này hay không. Nếu có, đơn ứng tuyển được chuyển thành trạng thái “Ứng viên chấp nhận” và kết thúc quy trình ứng tuyển. Và nếu không, đơn ứng tuyển được chuyển thành trạng thái “Từ chối đơn” cùng với kết thúc quy

56

2. HỖ TRỢ QUÁ TRÌNH ỨNG TUYỂN

trình tại đây.

e, Kết thúc của quy trình ứng tuyển

Quy trình ứng tuyển sẽ kết thúc khi xảy ra một trong các điều kiện sau: (i) Đơn ứng tuyển được chấp nhận hoặc bị từ chối trong quá

trình ứng tuyển, (ii) Đơn ứng tuyển quá hạn chót, (iii) Khi ứng viên đã chấp nhận một đơn ứng tuyển khác.

Đối với trường hợp (ii), đơn ứng tuyển sẽ bị dừng khi quá hạn chót. Để giải quyết cho trường hợp này, em tạo ra một tác vụ được lên lịch trước. Cụ thể, thiết lập cho một tác vụ được lên lịch chạy vào thời điểm cố định, thường được chọn là thời điểm khi lượng người truy cập thấp, tác vụ này sẽ có nhiệm vụ quét tất cả các đơn ứng tuyển có thiết lập hạn chót trong đơn. Những đơn nào đã quá hạn chót sẽ được tác vụ chuyển trạng thái về “Từ chối đơn” với lý do đã quá hạn chót cho đơn ứng tuyển này.

Đối với trường hợp (iii), khi ứng viên quyết định chuyển trạng thái của một đơn ứng tuyển khác sang trạng thái “Ứng viên chấp nhận”, mọi đơn ứng tuyển còn lại của ứng viên chưa kết thúc quy trình ứng tuyển sẽ bị buộc dừng ngay lập tức. Để thực hiện việc này, cần thiết đặt một tiến trình sẽ tự động chạy nếu như có một đơn ứng tuyển được chuyển về trạng thái “Ứng viên chấp nhận”. Tiến trình sẽ rà soát tất cả những đơn ứng tuyển của ứng viên đó mà đang trong quá trình tuyển dụng, và sẽ cập nhật về trạng thái “Từ chối đơn” với lý do đã nhận một công việc tại nơi khác.

f, Thông báo khi đơn ứng tuyển được cập nhật

Để có thể nhanh chóng cập nhật được thông tin khi đơn ứng tuyển được cập nhật, thông báo sẽ được gửi tới tài khoản của ứng viên và của nhân viên phụ trách để họ nhanh chóng nắm bắt được thông tin này, đặc biệt khi số lượng đơn ứng tuyển của hai đối tượng trên tham gia đạt một số lượng tương đối lớn.

Việc tạo thông báo sẽ được xử lý thông qua sử dụng tác vụ ngầm để tránh gây ảnh hưởng đến thời gian phản hồi khi đơn ứng tuyển được cập nhật. Khi đơn ứng tuyển được chuyển trạng thái, một tác vụ ngầm sẽ được chạy và tạo ra hai thông báo, một cho ứng viên, và một cho nhân viên phụ trách thông báo rằng đơn ứng tuyển họ tham gia đã được cập nhật trạng thái kèm với đường dẫn đến trang nội dung chi tiết của đơn ứng tuyển đó.

Kết quả thu được

Từ giải pháp thiết kế và triển khai trên đây, em đã xây dựng được một tính năng hỗ trợ quy trình ứng tuyển trên sản phẩm của mình, cùng với đó là một số các tác vụ hỗ trợ để giúp quy trình trở nên rõ ràng hơn như thông báo cập nhật, tự động

57

2. HỖ TRỢ QUÁ TRÌNH ỨNG TUYỂN

quét để kết thúc quy trình của các đơn quá hạn, . . .

Một vài hình ảnh từ hệ thống cho ứng dụng hỗ trợ quy trình này sẽ được trình bày dưới đây.

Đầu tiên là hình ảnh kết quả thu được cho khả năng chỉnh sửa thông tin cần thiết trong đơn ứng tuyển đối với từng việc làm cụ thể. Trong hình 5.5 mô tả việc lựa chọn các trường và đặt tên trường theo mong muốn của nhà tuyển dụng. Trường thông tin nào nhà tuyển dụng muốn sử dụng sẽ đánh dấu tích vào và đặt tên cho thông tin đó.



Hình 5.5: Giao diện thay đổi nội dung biểu mẫu đơn ứng tuyển

Hình kế tiếp là hình 5.6 mô tả khi ứng viên cập nhật thông tin trong giai đoạn "Bổ sung hồ sơ" khi được nhân viên phụ trách yêu cầu bổ sung thông tin cần thiết.



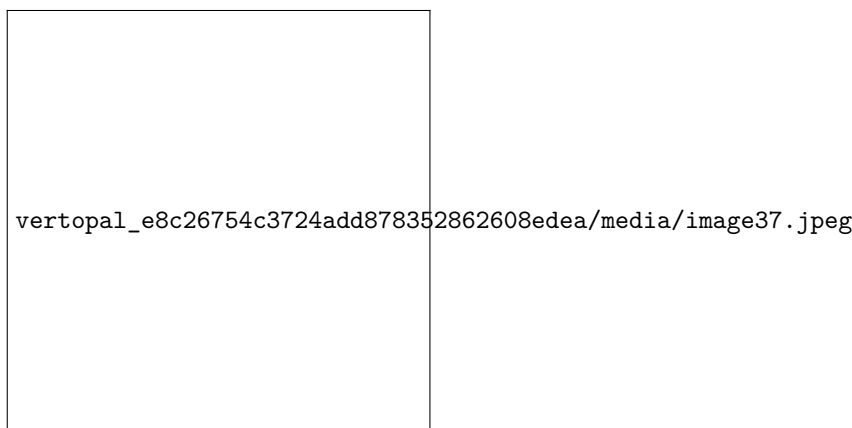
Hình 5.6: Giao diện cập nhật đơn ứng tuyển của ứng viên tại trạng thái "Bổ sung hồ sơ"

Hình cuối cùng mô tả trạng thái của đơn ứng tuyển sau khi cập nhật thông tin bổ sung từ giai đoạn "Bổ sung hồ sơ" ở trên. Lúc này đơn ứng tuyển đã quay trở lại trạng thái "Sơ tuyển". Trong hình 5.7 còn mô tả việc thông báo khi đơn ứng tuyển được cập nhật, khi đó ứng viên nhận được thông báo mới như được mô tả trong

58

3. KẾT CHƯƠng

hình.



Hình 5.7: Giao diện đơn ứng tuyển sau khi cập nhật và có thông báo cho ứng viên

Kết chương

Trong chương này em đã đề cập đến các đóng góp chính của đề tài, trong đó trọng tâm là nội dung hỗ trợ quy trình ứng tuyển được xây dựng áp dụng vào trong hệ thống. Chương kế tiếp, chương 6 sẽ đề cập đến kết luận và hướng phát triển cho đề tài về trang web môi giới việc làm này.

59

CHƯƠNG 6. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Kết luận

Từ những nội dung đã được trình bày trong đồ án này, chúng ta đã thấy được vấn đề tuyển dụng đóng vai trò quan trọng để phân bổ được nguồn lực lao động, giúp cho hai bên tìm việc và tuyển việc có thể đạt được ngành nghề phù hợp và nguồn nhân lực phù hợp đối với mục tiêu của mình. Từ đó chỉ ra được việc áp dụng thành tựu khoa học kĩ thuật, cụ thể ở đây là sử dụng hệ thống máy tính và

mạng Internet có thể giúp cho thực hiện công tác tuyển dụng, môi giới việc làm hiệu quả hơn mà còn tiết kiệm chi phí xét về lâu dài. Sử dụng các công nghệ và thiết bị trên giúp cho người tìm việc dễ dàng tiếp cận với bên nhà tuyển dụng hơn, giúp quản lý các thông tin một cách đáng tin cậy, dễ dàng trong vấn đề so sánh, tiếp cận đến từng đối tượng phù hợp với hệ thống, đồng thời nâng cao tính tương tác giữa hai bên trong quá trình tìm kiếm việc làm.

Các sản phẩm hiện tại trong nước đã giải quyết được một phần những vấn đề xoay quanh việc môi giới việc làm, nhưng đối với những sản phẩm trang đăng tin tuyển dụng, môi giới việc làm hiện tại không phải là không có những hạn chế. Những sản phẩm kể trên đã thực hiện tốt việc quản lý danh sách việc làm, quản lý danh sách các ứng viên có nhu cầu ứng tuyển. Do có hợp tác với nhiều các tổ chức doanh nghiệp nên số lượng công việc được đăng trên các trang đó có số lượng lớn, đa dạng từ ngành nghề đến vị trí cấp bậc, có mức độ phân cấp với mức lương đối với những việc làm đó. Nhờ có số lượng việc làm lớn, cùng với số ứng viên đăng ký tài khoản đạt một lượng đáng kể nên những hệ thống này có lượng dữ liệu đủ lớn để có thể sử dụng các tính năng gợi ý việc làm cho đúng đối tượng yêu cầu được hiệu quả. Tuy nhiên thường những sản phẩm này chưa hỗ trợ được một quy trình ứng tuyển rõ ràng mà chỉ có phần nộp hồ sơ và sau đó bên nhà tuyển dụng sẽ chủ động liên hệ với ứng viên thông qua các kênh khác.

Trong hệ thống môi giới việc làm đã được xây dựng trong đồ án này, sản phẩm đã có cải tiến để phát triển được khả năng hỗ trợ một quy trình tuyển dụng hoàn chỉnh từ khi ứng viên nộp hồ sơ cho đến khi đơn ứng tuyển được chấp nhận hay bị từ chối, đây chính là đóng góp chính của đồ án và cũng chính là điểm nổi bật so với các sản phẩm hiện có. Tuy nhiên sản phẩm trong đồ án vẫn còn nhiều hạn chế, vấn đề do chưa có những dữ liệu thực tế và chưa được triển khai nên chưa đánh giá được hết những vấn đề của sản phẩm. Vì vậy các chức năng liên quan đến tìm kiếm, gợi ý việc làm chưa thể đạt bằng các sản phẩm hiện đã có với lượng người dùng lớn với số đơn vị tuyển dụng nhiều. Các vấn đề này có thể được giải quyết

2. HƯỚNG PHÁT TRIỂN

bằng việc đưa hệ thống hoạt động thực tế để có những dữ liệu thực, từ đó có thể xây dựng các giải pháp tốt hơn để giải quyết.

Trong quá trình thực hiện đồ án tốt nghiệp này em đã tiến hành phân tích được sự quan trọng của việc áp dụng công nghệ thông tin vào quá trình môi giới việc làm, dựa trên phân tích về một số sản phẩm hiện tại và đề xuất ra một giải pháp để cải thiện và nâng cao chất lượng quá trình thực hiện môi giới việc làm đó. Từ đó đề xuất xây dựng một hệ thống môi giới việc làm với đóng góp chính trong

đó tập trung vào xây dựng một sản phẩm hỗ trợ một quy trình tuyển dụng hoàn chỉnh, và để hỗ trợ cho quy trình đó, một vài chức năng bổ sung đã được thêm vào để quy trình được vận hành tốt hơn.

Tuy nhiên, trong khuôn khổ thời gian cho phép vẫn còn nhiều điều em chưa thực hiện được trong đồ án này. Các dữ liệu được sử dụng là dữ liệu mẫu, chưa thu được dữ liệu thực tế để có một kết quả đánh giá tốt hơn. Tính năng gợi ý được đưa vào hệ thống mới chỉ thực hiện dựa trên nội dung mà người sử dụng, như ứng viên hay nhân viên nhập vào hệ thống chứ chưa đưa ra gợi ý một cách tự động dựa trên việc tổng hợp từ nhiều nguồn, so sánh giữa các ứng viên với nhau để tìm ra điểm chung phục vụ cho quá trình gợi ý.

Việc thực hiện đồ án đã đem lại cho em nhiều những bài học kinh nghiệm bổ ích sau đây. Đầu tiên là các sản phẩm được phát triển với mục tiêu hướng đến người sử dụng, các sản phẩm sau phải cải tiến so với trước đó để làm tăng trải nghiệm cho người sử dụng, giúp họ sử dụng dễ dàng thuận tiện hơn. Đồ án tốt nghiệp hướng tới một sản phẩm hoàn chỉnh, chính việc thực hiện xây dựng sản phẩm từ đầu đưa cho em một góc nhìn rộng hơn về quá trình một sản phẩm được xây dựng, từ khâu lên ý tưởng cho đến việc lựa chọn các công cụ phù hợp để giải quyết vấn đề, và từ những công cụ đó để xây dựng được sản phẩm đúng theo yêu cầu ban đầu đặt ra. Cuối cùng là khâu thiết kế đóng vai trò rất quan trọng trong quá trình hình thành nên sản phẩm cuối cùng, sau khi thực hiện đến giai đoạn xây dựng mới phát hiện ra vấn đề trong khâu thiết kế sẽ mất nhiều thời gian để có thể chỉnh sửa lại được như mong muốn.

Hướng phát triển

Một sản phẩm, trong quá trình phát triển và vận hành, luôn cần được nâng cao hơn cả về các tính năng và trải nghiệm để đem lại hiệu quả tốt hơn, giúp tạo ra và giữ được những người đã sử dụng tiếp tục sử dụng sản phẩm đó. Chính vì vậy bản thân hệ thống được phát triển trong đồ án này cũng cần những hướng đi mới để cải thiện được các chức năng sao cho tiện dụng với người sử dụng nhất.

Trước hết, để sản phẩm trở nên hoàn thiện hơn, cần phải bổ sung thêm tính năng

61

2. HƯỚNG PHÁT TRIỂN

gửi e-mail cho người dùng. Hệ thống sẽ gửi thông báo cho người dùng bằng e-mail khi có một số thông tin như xác thực tài khoản của người dùng, người dùng có thông báo mới nhưng chưa đọc trong vòng 24 giờ vừa qua, khi đơn ứng tuyển đang tham gia được cập nhật. Ngoài ra có thể gửi các e-mail quảng cáo và giới thiệu việc

làm, hay e-mail thông báo các việc làm mới được tuyển dụng đối với những ứng viên đang theo dõi một công ty nào đó. Để tính năng cuối có thể hoạt động, hệ thống cũng cần bổ sung chức năng theo dõi công ty hoặc theo dõi việc làm để có thông tin cập nhật chi tiết sớm nhất có thể.

Để nâng cao được chất lượng giao tiếp giữa ứng viên và bên nhà tuyển dụng, một hướng phát triển mới là tạo ra một kênh giao tiếp trực tiếp giữa hai bên, có thể tích hợp với tính năng nhắn tin của các ứng dụng như Zalo, Messenger để nâng cao hiệu quả sử dụng tính năng này. Một kênh giao tiếp khác thông qua e-mail cũng có thể được phát triển, tích hợp các ứng dụng như Gmail, Outlook, giúp cho người dùng có thể trực tiếp soạn e-mail trên hệ thống môi giới việc làm này, từ đó làm giảm được công sức và sử dụng hệ thống hiệu quả hơn.

Đối với chức năng gợi ý việc làm, hiện tại hệ thống mới chỉ gợi ý dựa trên thông tin đầu vào do người dùng cung cấp. Khi lượng dữ liệu thu thập đã đủ nhiều, tính năng gợi ý này có thể áp dụng các công cụ gợi ý nâng cao hơn như sử dụng công nghệ học máy, học sâu kèm các phương pháp gợi ý nội dung ví dụ như gợi ý dựa trên lọc cộng tác, để từ đó đưa ra được kết quả gợi ý tốt hơn phương pháp hiện tại. Các phương pháp gợi ý này tận dụng được sự tương đồng giữa các ứng viên ứng tuyển nên sẽ tìm ra được các việc làm gợi ý tốt hơn với phương pháp thuần túy trên nội dung đầu vào mang lại.

Cuối cùng, hiện tại, hệ thống mới chỉ cung cấp các tính năng đăng tin tuyển dụng, ứng viên tìm việc làm, và nhà tuyển dụng tìm ứng viên. Nên một hướng phát triển quan trọng là hệ thống môi giới việc làm sẽ có khả năng tìm kiếm ứng viên phù hợp với việc làm yêu cầu, và làm trung gian kết nối giữa họ một cách tự động. Để thực hiện theo hướng phát triển này, cần phải kết hợp chức năng gợi ý việc làm và chức năng gợi ý ứng viên để có thể tạo ra chức năng môi giới một cách hiệu quả. Công nghệ áp dụng có thể là công nghệ để phát triển tính năng gợi ý việc làm đã được nêu phía trên.

Tổng kết lại, hệ thống còn có nhiều hướng đi để phát triển góp phần đem lại trải nghiệm tốt hơn, hiệu quả hơn và thân thiện hơn với người dùng trong tương lai.

62

TÀI LIỆU THAM KHẢO

1. Công ty Cổ phần TopCV Việt Nam, *Quy chế hoạt động website cung cấp dịch vụ thương mại điện tử topcv.vn*. [Online]. Available: https://static.topcv.vn/manual/Quy_che_san_TMDT_TopCV.pdf (visited on 07/31/2022).

2. Công ty Cổ phần CareerBuilder, *Quy chế hoạt động của sàn giao dịch thương mại điện tử careerbuilder.vn*. [Online]. Available: <https://images.careerbuilder.vn/regulations.pdf> (visited on 07/31/2022).
3. Công ty Trách nhiệm hữu hạn CareerLink, *Về careerlink*. [Online]. Available: <https://www.careerlink.vn/ve-careerlink> (visited on 07/31/2022).
4. E. F. Codd, "The relational model of data for large shared data banks," *CACM*, vol. 13, no. 6, 377–387, 1970.
5. Đại học Công nghệ Đại học Quốc gia Hà Nội, *Giáo trình Lập trình hướng đối tượng*. Đại học Công nghệ - Đại học Quốc gia Hà Nội, 2013.
6. Ruby community, *About ruby*. [Online]. Available: <https://www.ruby-lang.org/en/about/> (visited on 07/31/2022).
7. M. Hartl, *The Ruby on Rails Tutorial Sixth Edition*. Addison-Wesley Professional, 2020.
8. Microsoft Corporation, *Background jobs*. [Online]. Available: <https://docs.microsoft.com/en-us/azure/architecture/best-practices/background-jobs> (visited on 07/31/2022).
9. Contributed Systems LLC., *Sidekiq*. [Online]. Available: <https://sidekiq.org/> (visited on 07/31/2022).
10. Redis Ltd., *Introduction to redis*. [Online]. Available: <https://redis.io/docs/about/> (visited on 07/31/2022).
11. Microsoft Corporation, *Asp.net mvc overview*. [Online]. Available: <https://docs.microsoft.com/en-us/aspnet/mvc/overview/older-versions-1/overview/asp-net-mvc-overview> (visited on 07/31/2022).

PHỤ LỤC

A.

ĐẶC TẢ USE CASE

1.

Đặc tả use case “Quản lý quản lý danh sách nhân viên”

| Mã Use case | Tên use case | Quản lý nhân viên |
|-------------|--------------|-------------------|
| UC004 | | |
| Tác nhân | Quản lý | |

| Mã Use case | | Tên use case | Quản lý nhân viên |
|-------------------------------|----------------------|---|--|
| | | UC004 | |
| Tiền điều kiện | | Quản lý đã đăng nhập | |
| | | Mã: UC0041 - Thêm nhân viên: Mã: UC0042 – Xóa nhân viên: | |
| | | Hành động | |
| | STT | Thực hiện bởi | |
| Luồng sự kiện chính | 1. 2. 3. 4. | Quản lý Hệ thống Quản lý Hệ thống | Trong phần danh sách nhân viên trong trang công ty, chọn tính năng xóa nhân viên đối với tài khoản muốn xóa Hiển thị thông báo yêu cầu quản lý xác nhận việc xóa Xác nhận xóa tài khoản nhân viên Xóa tài khoản và thông báo xóa thành công Kết thúc use case nếu quản lý xác nhận không xóa |
| Luồng thay thế | 4a. | Hệ thống | |
| Hậu điều kiện | | Không | |

Bảng A.1: Đặc tả use case nhân viên quản lý danh sách nhân viên

2. ĐẶC TẢ USE CASE “ĐĂNG KÝ TÀI KHOẢN ỨNG VIÊN”

1.

Đặc tả use case “Đăng ký tài khoản ứng viên”

| Mã Use case | UCDK | Tên use case | Ứng viên đăng ký |
|--|------------------------------------|--|--|
| Tác nhận Tiền điều kiện | Ứng viên | Ứng viên muốn đăng ký tài khoản | |
| | STT | Thực hiện bởi | Hành động |
| Luồng sự kiện chính | 1. 2. 3. 4. | Ứng viên Hệ thống Ứng viên Hệ thống | Truy cập vào trang chủ, chọn tính năng tạo tài khoản Hiển thị biểu mẫu điền thông tin đăng ký tài khoản mới Điền đầy đủ thông tin, nhấn nút tạo tài khoản ứng viên Thông báo tạo việc tài khoản ứng viên mới thành công Kết thúc use case nếu người dùng không lựa chọn tạo tài khoản Thông báo lỗi nếu thông tin tạo tài khoản không hợp lệ Thông báo lỗi nếu như có trường bắt buộc chưa được điền giá trị |
| | Luồng sự kiện thay thế | 4a. 4b. 4c. Hệ thống Hệ thống Hệ thống | |
| Hậu điều kiện | Không | | |

Bảng A.2: Đặc tả use case ứng viên đăng ký tài khoản