TRƯỜNG ĐẠI HỌC THỦ DẦU MỘT
VIỆN ĐÀO TẠO CÔNG NGHỆ THÔNG TIN, CHUYỂN ĐỔI SỐ

ĐỒ ÁN MÔN HỌC
KỸ THUẬT LẬP TRÌNH
TRONG PHÂN TÍCH DỮ LIỆU

CRAWL DỮ LIỆU, LOAD, TÌM KIẾM VIỆC LÀM TRÊN WEBSITE TOPDEV.VN

SVTH: 1. Hồ Tuấn Phước Mã SV: 2224802010872
2. Nguyễn Minh Nghi Mã SV: 2224802010934
3. Phan Phước Hồng Phúc Mã SV: 2224802010871

Lớp: CNTT.CQ.03
GVHD: ThS. Nguyễn Thế Bảo

Tháng 05/2023

TRƯỜNG ĐH THỦ DẦU MỘT
VIỆN ĐÀO TẠO CNTT & CĐS

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập - Tự do – Hạnh phúc

PHIẾU CHẤM TIỂU LUẬN

Thời gian: 23/05/2025.....Địa điểm: D-104.....
Học phần: Kỹ thuật lập trình trong phân tích dữ liệu (CNTT024).....
Tên đề tài: Crawl dữ liệu, load và tìm kiếm trên website Topdev.....
Sinh viên/ Nhóm SV thực hiện: …..... Lớp .....

Hồ Tuấn Phước	2224802010872	D22CNTT06
Nguyễn Minh Nghi	2224802010934	D22CNTT06
Phan Phước Hồng Phúc	2224802010871	D22CNTT06

Phần 1. Nội dung (6.0 điểm)

STT	Nội dung đánh giá	Điểm tối đa	Kém (25%)	Trung bình (50%)	Khá (75%)	Tốt (100%)	Điểm đánh giá
1	Thái độ tham gia	1.0	Không quan tâm lựa chọn ý tưởng	Chọn ý tưởng trong số được đề nghị	Tìm kiếm và đưa ra được ý tưởng khá tốt	Tích cực tìm kiếm và chủ động đưa ra ý tưởng	
2	Xác định được các kỹ thuật để thu thập và xử lý dữ liệu cho bài toán cần xây dựng	1.0	Sinh viên không xác định được kỹ thuật để thu thập và xử lý dữ liệu	Sinh viên xác định được kỹ thuật để thu thập và xử lý dữ liệu còn nhiều sai sót	Sinh viên xác định được kỹ thuật để thu thập và xử lý dữ liệu, còn một vài sai sót nhỏ	Sinh viên viết xác định được kỹ thuật để thu thập và xử lý dữ liệu. (sai sót không đáng kể)	
3	Mô tả dữ liệu đã thu thập	2.0	Sinh viên không thu thập được dữ liệu như yêu cầu	Sinh viên không thu thập được dữ liệu như yêu cầu	Sinh viên thu thập được dữ liệu khá tốt (từ 500 - 1000 mẫu)	Sinh viên thu thập được dữ liệu tốt (hơn 1000 mẫu)	

4	Xây dựng hệ thống tìm kiếm	2.0	Sinh viên không xây dựng được ứng dụng như yêu cầu	Sinh viên xây dựng được ứng dụng như yêu cầu, nhưng không có sắp xếp kết quả tìm kiếm được	Sinh viên xây dựng được ứng dụng như yêu cầu, có sắp xếp kết quả tìm kiếm được	Sinh viên xây dựng được ứng dụng như yêu cầu, có sắp xếp kết quả tìm kiếm được, có kết hợp phương pháp học máy trong tìm kiếm.	
Phần 2. Trình bày (2.0 điểm)							
1	Hình thức, bố cục của cuốn báo cáo	1.0	Không đúng mẫu và còn nhiều lỗi chính tả	Đúng mẫu, còn nhiều lỗi chính tả, lỗi định dạng	Đúng mẫu, còn một vài lỗi định dạng.	Đúng mẫu, đúng định dạng.	
2	Thuyết trình	1.0	Người thuyết trình chưa tự tin, chưa thu hút người nghe	Người thuyết trình còn mắc một số lỗi (giọng nhỏ, đọc là chủ yếu, …)	Người thuyết trình tự tin.	Người thuyết trình tự tin, thu hút người nghe.	
Phần 3. Trả lời câu hỏi (2.0 điểm)							
1	Trả lời câu hỏi của CB chấm	2.0	Không trả lời được câu hỏi đặt ra	Trả lời được 50% câu hỏi đặt ra, câu trả lời chưa hoàn chỉnh.	Trả lời được câu hỏi đặt ra, còn 1 vài sai sót nhỏ.	Trả lời chính xác hầu hết câu hỏi đặt ra	

Tổng điểm	
ĐIỂM CỦA CÁ NHÂN **(do GV ghi)**	***Danh sách thành viên của Nhóm:*** Họ tên: Hồ Tuấn Phước ……..Điểm:……. *TÊN HỌ VÀ CHỮ KÝ XÁC NHẬN CỦA GV:*

TRƯỜNG ĐH THỦ DẦU MỘT
VIỆN ĐÀO TẠO CNTT & CĐS

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập - Tự do – Hạnh phúc

PHIẾU CHẤM TIỂU LUẬN

Thời gian: 23/05/2025..... Địa điểm: D-104.....
Học phần: Kỹ thuật lập trình trong phân tích dữ liệu (CNTT024).....
Tên đề tài: Crawl dữ liệu, load và tìm kiếm trên website Topdev.....
Sinh viên/ Nhóm SV thực hiện: …..... Lớp .....

Hồ Tuấn Phước	2224802010872	D22CNTT06
Nguyễn Minh Nghi	2224802010934	D22CNTT06
Phan Phước Hồng Phúc	2224802010871	D22CNTT06

Phần 1. Nội dung (6.0 điểm)

STT	Nội dung đánh giá	Điểm tối đa	Kém (25%)	Trung bình (50%)	Khá (75%)	Tốt (100%)	Điểm đánh giá
1	Thái độ tham gia	1.0	Không quan tâm lựa chọn ý tưởng	Chọn ý tưởng trong số được đề nghị	Tìm kiếm và đưa ra được ý tưởng khá tốt	Tích cực tìm kiếm và chủ động đưa ra ý tưởng	
2	Xác định được các kỹ thuật để thu thập và xử lý dữ liệu cho bài toán cần xây dựng	1.0	Sinh viên không xác định được kỹ thuật để thu thập và xử lý dữ liệu	Sinh viên xác định được kỹ thuật để thu thập và xử lý dữ liệu còn nhiều sai sót	Sinh viên xác định được kỹ thuật để thu thập và xử lý dữ liệu, còn một vài sai sót nhỏ	Sinh viên viết xác định được kỹ thuật để thu thập và xử lý dữ liệu. (sai sót không đáng kể)	
3	Mô tả dữ liệu đã thu thập	2.0	Sinh viên không thu thập được dữ liệu như yêu cầu	Sinh viên không thu thập được dữ liệu như yêu cầu	Sinh viên thu thập được dữ liệu khá tốt (từ 500 - 1000 mẫu)	Sinh viên thu thập được dữ liệu tốt (hơn 1000 mẫu)	

4	Xây dựng hệ thống tìm kiếm	2.0	Sinh viên không xây dựng được ứng dụng như yêu cầu	Sinh viên xây dựng được ứng dụng như yêu cầu, nhưng không có sắp xếp kết quả tìm kiếm được	Sinh viên xây dựng được ứng dụng như yêu cầu, có sắp xếp kết quả tìm kiếm được	Sinh viên xây dựng được ứng dụng như yêu cầu, có sắp xếp kết quả tìm kiếm được, có kết hợp phương pháp học máy trong tìm kiếm.	
Phần 2. Trình bày (2.0 điểm)							
1	Hình thức, bố cục của cuốn báo cáo	1.0	Không đúng mẫu và còn nhiều lỗi chính tả	Đúng mẫu, còn nhiều lỗi chính tả, lỗi định dạng	Đúng mẫu, còn một vài lỗi định dạng.	Đúng mẫu, đúng định dạng.	
2	Thuyết trình	1.0	Người thuyết trình chưa tự tin, chưa thu hút người nghe	Người thuyết trình còn mắc một số lỗi (giọng nhỏ, đọc là chủ yếu, …)	Người thuyết trình tự tin.	Người thuyết trình tự tin, thu hút người nghe.	
Phần 3. Trả lời câu hỏi (2.0 điểm)							
1	Trả lời câu hỏi của CB chấm	2.0	Không trả lời được câu hỏi đặt ra	Trả lời được 50% câu hỏi đặt ra, câu trả lời chưa hoàn chỉnh.	Trả lời được câu hỏi đặt ra, còn 1 vài sai sót nhỏ.	Trả lời chính xác hầu hết câu hỏi đặt ra	

Tổng điểm	
ĐIỂM CỦA CÁ NHÂN **(do GV ghi)**	***Danh sách thành viên của Nhóm:*** Họ tên: Nguyễn Minh Nghi ……..….Điểm:……. *TÊN HỌ VÀ CHỮ KÝ XÁC NHẬN CỦA GV:*

TRƯỜNG ĐH THỦ DẦU MỘT
VIỆN ĐÀO TẠO CNTT & CĐS

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập - Tự do – Hạnh phúc

PHIẾU CHẤM TIỂU LUẬN

Thời gian: 23/05/2025.....Địa điểm: D-104.....
Học phần: Kỹ thuật lập trình trong phân tích dữ liệu (CNTT024).....
Tên đề tài: Crawl dữ liệu, load và tìm kiếm trên website Topdev.....
Sinh viên/ Nhóm SV thực hiện: …..... Lớp .....

Hồ Tuấn Phước	2224802010872	D22CNTT06
Nguyễn Minh Nghi	2224802010934	D22CNTT06
Phan Phước Hồng Phúc	2224802010871	D22CNTT06

Phần 1. Nội dung (6.0 điểm)

STT	Nội dung đánh giá	Điểm tối đa	Kém (25%)	Trung bình (50%)	Khá (75%)	Tốt (100%)	Điểm đánh giá
1	Thái độ tham gia	1.0	Không quan tâm lựa chọn ý tưởng	Chọn ý tưởng trong số được đề nghị	Tìm kiếm và đưa ra được ý tưởng khá tốt	Tích cực tìm kiếm và chủ động đưa ra ý tưởng	
2	Xác định được các kỹ thuật để thu thập và xử lý dữ liệu cho bài toán cần xây dựng	1.0	Sinh viên không xác định được kỹ thuật để thu thập và xử lý dữ liệu	Sinh viên xác định được kỹ thuật để thu thập và xử lý dữ liệu còn nhiều sai sót	Sinh viên xác định được kỹ thuật để thu thập và xử lý dữ liệu, còn một vài sai sót nhỏ	Sinh viên viết xác định được kỹ thuật để thu thập và xử lý dữ liệu. (sai sót không đáng kể)	
3	Mô tả dữ liệu đã thu thập	2.0	Sinh viên không thu thập được dữ liệu như yêu cầu	Sinh viên không thu thập được dữ liệu như yêu cầu	Sinh viên thu thập được dữ liệu khá tốt (từ 500 - 1000 mẫu)	Sinh viên thu thập được dữ liệu tốt (hơn 1000 mẫu)	

4	Xây dựng hệ thống tìm kiếm	2.0	Sinh viên không xây dựng được ứng dụng như yêu cầu	Sinh viên xây dựng được ứng dụng như yêu cầu, nhưng không có sắp xếp kết quả tìm kiếm được	Sinh viên xây dựng được ứng dụng như yêu cầu, có sắp xếp kết quả tìm kiếm được	Sinh viên xây dựng được ứng dụng như yêu cầu, có sắp xếp kết quả tìm kiếm được, có kết hợp phương pháp học máy trong tìm kiếm.	
Phần 2. Trình bày (2.0 điểm)							
1	Hình thức, bố cục của cuốn báo cáo	1.0	Không đúng mẫu và còn nhiều lỗi chính tả	Đúng mẫu, còn nhiều lỗi chính tả, lỗi định dạng	Đúng mẫu, còn một vài lỗi định dạng.	Đúng mẫu, đúng định dạng.	
2	Thuyết trình	1.0	Người thuyết trình chưa tự tin, chưa thu hút người nghe	Người thuyết trình còn mắc một số lỗi (giọng nhỏ, đọc là chủ yếu, …)	Người thuyết trình tự tin.	Người thuyết trình tự tin, thu hút người nghe.	
Phần 3. Trả lời câu hỏi (2.0 điểm)							
1	Trả lời câu hỏi của CB chấm	2.0	Không trả lời được câu hỏi đặt ra	Trả lời được 50% câu hỏi đặt ra, câu trả lời chưa hoàn chỉnh.	Trả lời được câu hỏi đặt ra, còn 1 vài sai sót nhỏ.	Trả lời chính xác hầu hết câu hỏi đặt ra	

Tổng điểm	
ĐIỂM CỦA CÁ NHÂN **(do GV ghi)**	***Danh sách thành viên của Nhóm:*** Họ tên: Phan Phước Hồng Phúc …….Điểm:……. *TÊN HỌ VÀ CHỮ KÝ XÁC NHẬN CỦA GV:*

MỤC LỤC

DANH MỤC HÌNH

DANH MỤC BẢNG

LỜI MỞ ĐẦU

Trong thời đại công nghệ phát triển mạnh mẽ như hiện nay, dữ liệu đóng vai trò quan trọng trong hầu hết các lĩnh vực, đặc biệt là trong ngành công nghệ thông tin. Các nền tảng tuyển dụng trực tuyến như TopDev.vn không chỉ là nơi kết nối giữa nhà tuyển dụng và ứng viên, mà còn chứa đựng nhiều thông tin hữu ích về xu hướng nghề nghiệp, kỹ năng phổ biến và nhu cầu thị trường lao động.

Xuất phát từ nhu cầu khai thác và phân tích các dữ liệu này, nhóm chúng em thực hiện đề tài "Thu thập (Crawl) dữ liệu từ website TopDev.vn" nhằm xây dựng một hệ thống đơn giản có khả năng tự động lấy thông tin việc làm từ trang web TopDev. Dữ liệu thu thập bao gồm tên công việc, công ty, kỹ năng, mức lương, địa điểm làm việc,… và sẽ được lưu trữ để phục vụ cho việc phân tích, thống kê hoặc các ứng dụng liên quan.

Trong quá trình thực hiện, nhóm sử dụng ngôn ngữ Python cùng các thư viện như Requests, BeautifulSoup, và Flask-SQLAlchemy để xây dựng hệ thống crawl và quản lý dữ liệu. Đề tài giúp nhóm củng cố kiến thức về lập trình web, xử lý dữ liệu, cũng như hiểu rõ hơn về thực trạng tuyển dụng trong ngành CNTT.

Nhóm xin chân thành cảm ơn thầy/cô đã hướng dẫn và hỗ trợ trong suốt quá trình thực hiện đề tài.

CHƯƠNG 1. TỔNG QUAN

1.1. Giới thiệu tổng quan đề tài

Trong những năm gần đây, lĩnh vực công nghệ thông tin (CNTT) tại Việt Nam đang phát triển mạnh mẽ, kéo theo nhu cầu tuyển dụng các vị trí kỹ thuật như lập trình viên, kỹ sư dữ liệu, kiểm thử phần mềm,… ngày càng tăng cao. Các nền tảng tuyển dụng trực tuyến như TopDev.vn đã trở thành cầu nối quan trọng giữa nhà tuyển dụng và ứng viên trong ngành CNTT. TopDev không chỉ cung cấp thông tin việc làm mà còn phản ánh trực tiếp xu hướng kỹ năng, mức lương và nhu cầu nhân lực của thị trường công nghệ Việt Nam.

Đề tài "Thu thập dữ liệu từ website TopDev.vn" được thực hiện với mục tiêu thu thập và phân tích dữ liệu tuyển dụng nhằm phục vụ các ứng dụng như:

- Phân tích xu hướng thị trường lao động ngành CNTT.
- Xây dựng tập dữ liệu phục vụ cho các bài toán học máy hoặc hệ thống gợi ý việc làm.
- Hỗ trợ sinh viên, người học định hướng nghề nghiệp, bổ sung kỹ năng cần thiết.

1.2. Tổng quan Python

Python là một ngôn ngữ lập trình thông dịch, được phát triển vào đầu những năm 1990 bởi Guido van Rossum. Với cú pháp đơn giản, dễ đọc và dễ hiểu, Python đã trở thành một trong những ngôn ngữ lập trình phổ biến nhất trên thế giới.

Python được sử dụng rộng rãi trong nhiều lĩnh vực khác nhau, từ phát triển web đến xử lý dữ liệu và trí tuệ nhân tạo. Điều này là do sự linh hoạt và mạnh mẽ của ngôn ngữ này, cùng với sự hỗ trợ từ cộng đồng lập trình viên lớn mạnh

Hình 1.1: Logo python

Một số điểm nổi bật của Python bao gồm:

- Cú pháp rõ ràng: Python có cú pháp đơn giản và dễ đọc, giúp cho việc phát triển và bảo trì mã nguồn trở nên dễ dàng hơn.
- Thư viện phong phú: Python có một loạt các thư viện và framework phong phú, giúp cho việc phát triển ứng dụng trở nên nhanh chóng và hiệu quả.
- Hỗ trợ đa nền tảng: Python có thể chạy trên nhiều hệ điều hành khác nhau như Windows, macOS và Linux.

- Hỗ trợ cộng đồng mạnh mẽ: Python có một cộng đồng lập trình viên rộng lớn và nhiệt tình, với nhiều tài liệu, hướng dẫn và hỗ trợ trực tuyến.

1.3. Tổng quan về thu thập dữ liệu

1.3.1. Request và Beautifulsoup

Trong quá trình phát triển dự án, nhóm chúng em sử dụng thư viện Requests để gửi các yêu cầu HTTP đến trang web topdev.vn và nhận về dữ liệu HTML. Requests là một thư viện Python cung cấp giao diện dễ sử dụng để tạo và gửi các yêu cầu HTTP.

Sau đó, nhóm chúng em sử dụng thư viện BeautifulSoup để phân tích và 4 Format các thẻ html có trong dữ liệu Crawl về và chỉ để lấy dữ liệu gốc của nó. BeautifulSoup cung cấp các công cụ mạnh mẽ để phân tích cây HTML và trích xuất thông tin một cách linh hoạt và hiệu quả.

1.3.2. Flask-SqlAlchemy

Flask-SQLAlchemy là một extension của framework Flask giúp kết nối ứng dụng Flask với cơ sở dữ liệu SQL một cách dễ dàng. nhóm chúng em sử dụng Flask SQLAlchemy để tạo và quản lý cơ sở dữ liệu cho dự án của mình.

Điều này giúp chúng em tương tác với cơ sở dữ liệu một cách thuận tiện thông qua các đối tượng Python, mà không cần phải viết các truy vấn SQL phức tạp. Flask-SQLAlchemy cũng cung cấp các tính năng để quản lý việc thay đổi cấu trúc của cơ sở dữ liệu một cách an toàn và dễ dàng

1.3.3. Selenium

Trong quá trình phát triển dự án, nhóm chúng em sử dụng thư viện Selenium để tự động hóa việc tương tác với trình duyệt web. Selenium cho phép mô phỏng các thao tác của người dùng như click, nhập liệu, cuộn trang, và lấy dữ liệu từ các trang web có nội dung động (JavaScript render). Điều này rất hữu ích khi crawl dữ liệu từ những trang web mà Requests và BeautifulSoup không thể lấy được toàn bộ nội dung do dữ liệu được tải động sau khi trang web đã được tạo hoặc là cần đăng nhập để lấy dữ liệu.

1.3.4. Matplotlib

matplotlib là thư viện vẽ đồ thị và trực quan hóa dữ liệu phổ biến trong Python. Nhóm chúng em sử dụng matplotlib để trực quan hóa các kết quả phân tích dữ liệu, giúp dễ dàng nhận diện các xu hướng, mẫu dữ liệu và trình bày kết quả một cách trực quan, sinh động.

1.3.5. sentence_transformers

sentence_transformers là một thư viện mạnh mẽ trong Python dùng để chuyển đổi các câu văn bản thành vector số (embedding) có ý nghĩa ngữ nghĩa. Trong dự án, nhóm chúng em sử dụng sentence_transformers để mã hóa các mô tả công việc hoặc thông tin ứng viên thành vector, giúp so sánh mức độ tương đồng giữa các văn bản một cách hiệu quả. Điều này hỗ trợ rất tốt cho việc xây dựng hệ thống tìm kiếm nội dung

1.3.6. faiss_cpu

faiss_cpu là một thư viện tối ưu hóa cho việc tìm kiếm gần đúng các vector trong không gian nhiều chiều, được phát triển bởi Facebook AI Research. Trong dự án, nhóm chúng em sử dụng faiss_cpu để xây dựng công cụ tìm kiếm nhanh chóng trên tập dữ liệu lớn đã được mã hóa bằng sentence_transformers. Nhờ faiss_cpu, hệ thống có thể tìm kiếm và trả về các kết quả tương tự nhất với truy vấn của người dùng một cách hiệu quả và tiết kiệm tài nguyên.

1.4. Câu hỏi nghiên cứu

Làm sao để crawl dữ liệu về 1 cách nhanh chóng ?

Làm sao khi crawl dữ liệu thì hệ thống có thể tự động vào trang cần crawl và tự động nhấn nút “Hiển thị thêm dữ liệu” để thu thập?

Làm sao để khi crawl dữ liệu về thì có thể tự động lọc trùng dữ liệu và tự động lưu xuống file .json, sau đó thực hiện đẩy data từ .json vào database?

Làm thế nào để format lại dữ liệu thô từ database thành dữ liệu text?

CHƯƠNG 2. THU THẬP DỮ LIỆU VÀ XÂY DỰNG HỆ THỐNG

2.1. Mô tả dữ liệu đã thu thập

Dữ liệu được thu nhập bằng cách crawl nội dung từ website "https://topdev.vn/" sử dụng thư viện "BeautifulSoup" , "requests" và kết hợp với "Selenium" .

Các bước tiến hành:

Bước 1 : Chọn lọc các trang để crawl dữ liệu, nhóm em sẽ chọn trang chứa dữ liệu chứa các công ty và trang chứa dữ liệu các công việc, lần lượt là "https://topdev.vn/nha-tuyen-dung" và "https://topdev.vn/viec-lam-it" để tiến hành crawl dữ liệu trong website .

Bước 2 : Tải toàn bộ trang bằng Selenium

- Sử dụng thư viện Selenium để điều khiển trình duyệt tự động load hết nội dung bằng cách nhấn nút "Hiển thị thêm nhà tuyển dụng" nhiều lần có giới hạn max_clicks =100. Bên cạnh đó có những thông tin cần đăng nhập để hiển thị như lương nên sử dụng Selenium để đăng nhập để lấy đầy đủ các dữ liệu
- Sau khi tải xong, lấy mã HTML của toàn bộ trang.

Bước 3 :Trích xuất dữ liệu công ty từ HTML

- Sử dụng BeautifulSoup để phân tích HTML và tách dữ liệu công ty và công việc với mỗi công ty lấy các thông tin cơ bản như name, logo, link, banner,description, location, industry, followers, jobs_count và trích xuất tất cả các link của công việc.
- Sau đó lưu toàn bộ dữ liệu vào file JSON (companyData.json, job_links.json)

Bước 4 : Crawl chi tiết

- Đối với công ty:
  - Đọc file companyData.json, duyệt qua từng công ty và truy cập vào trang chi tiết của công ty .
  - Trích xuất thêm các thông tin chi tiết như mô tả chi tiết, các kỹ năng, công nghệ sử dụng và hình ảnh liên quan.
  - Tổng hợp lại thành 1 object chi tiết cho mỗi công ty và lưu lại vào file JSON (companyDetails.json)
- Đối với công việc:
  - Đọc file job_links.json, duyệt qua từng công việc và truy cập vào trang chi tiết của công việc.
  - Trích xuất các thông tin chi tiết của từng công việc như: tên công việc, công ty, mức lương, địa điểm, mô tả, yêu cầu, kỹ năng, ngày đăng tuyển, và đường dẫn chi tiết đến công việc đó.
  - Chỉnh lại định dạng như lương về số, thời gian về datetime
  - Tổng hợp lại thành 1 object chi tiết cho mỗi công ty và lưu lại vào file JSON (job_info.json)

Bước 5: Đưa dữ liệu vào cơ sở dữ liệu SQLite

- Dùng SQLiteStudio để tạo sẵn các bảng trong file crawl.db
- Sử dụng thư viện sqlite3 để kết nối với cơ sở dữ liệu và thực hiện các truy vấn để cập nhật dữ liệu từ các files .json() xuống database .
- Mở file companyDetails.json đã crawl ở trên, duyệt qua từng công ty, và đưa các trường dữ liệu như name, logo, shortDescription, industry, address, followers, …. vào bảng company.
- Tương tự cũng thêm các công việc từ file job_info.json vào cơ sở dữ liệu

- Cuối cùng là kết nối giữa các công việc và công ty lại với nhau

Bộ dữ liệu:

Bộ dữ liệu được thu nhập vào ngày 18/05/2025 .

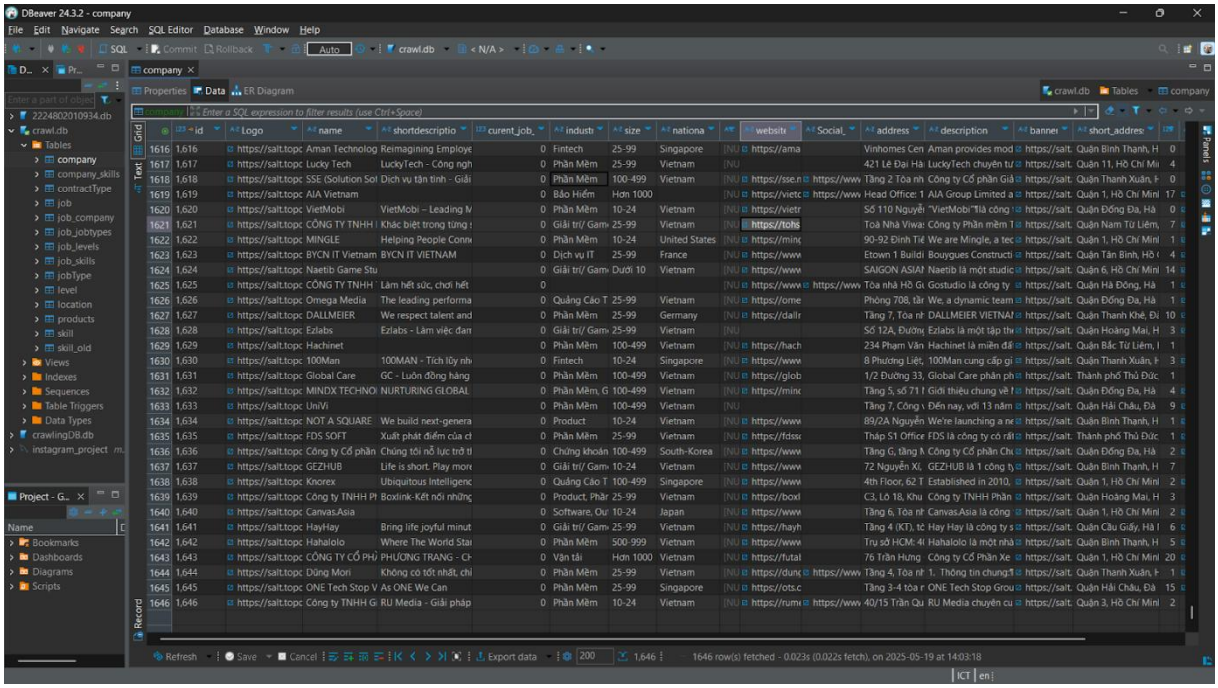
Cơ sở dữ liệu của dự án có tên là: **crawl.db** .

2.1.1. Bộ dữ liệu bảng company

Bộ dữ liệu thu nhập được lưu vào cơ sở dữ liệu crawl trong table company có 1646 dòng dữ liệu và 16 cột bao gồm:

Bảng 2.1: Bộ dữ liệu bảng company

STT	Tên trường	Kiểu dữ liệu	Mô tả
1	id	INTEGER	Mã id
2	Logo	TEXT	Đường dẫn ảnh logo
3	name	TEXT	Tên công ty
4	shortdescription	TEXT	Mô tả thông tin ngắn
5	current_job_opening	INTEGER	Các công việc cần tuyển dụng
6	industry	TEXT	Loại hình công ty
7	size	TEXT	Kích thước công ty
8	nationality	TEXT	Quốc gia công ty
9	website	TEXT	Đường dẫn đến trang website của công ty
10	Social_media	TEXT	Đường dẫn mạng xã hội của công ty
11	address	TEXT	Địa chỉ gốc
12	description	TEXT	Mô tả chi tiết
13	banner	TEXT	Đường dẫn hình ảnh banner
14	short_address	TEXT	Địa chỉ rút gọn
15	followers	INTEGER	Số người theo dõi ứng tuyển
16	about_images	TEXT	Các đường dẫn hình ảnh mô tả



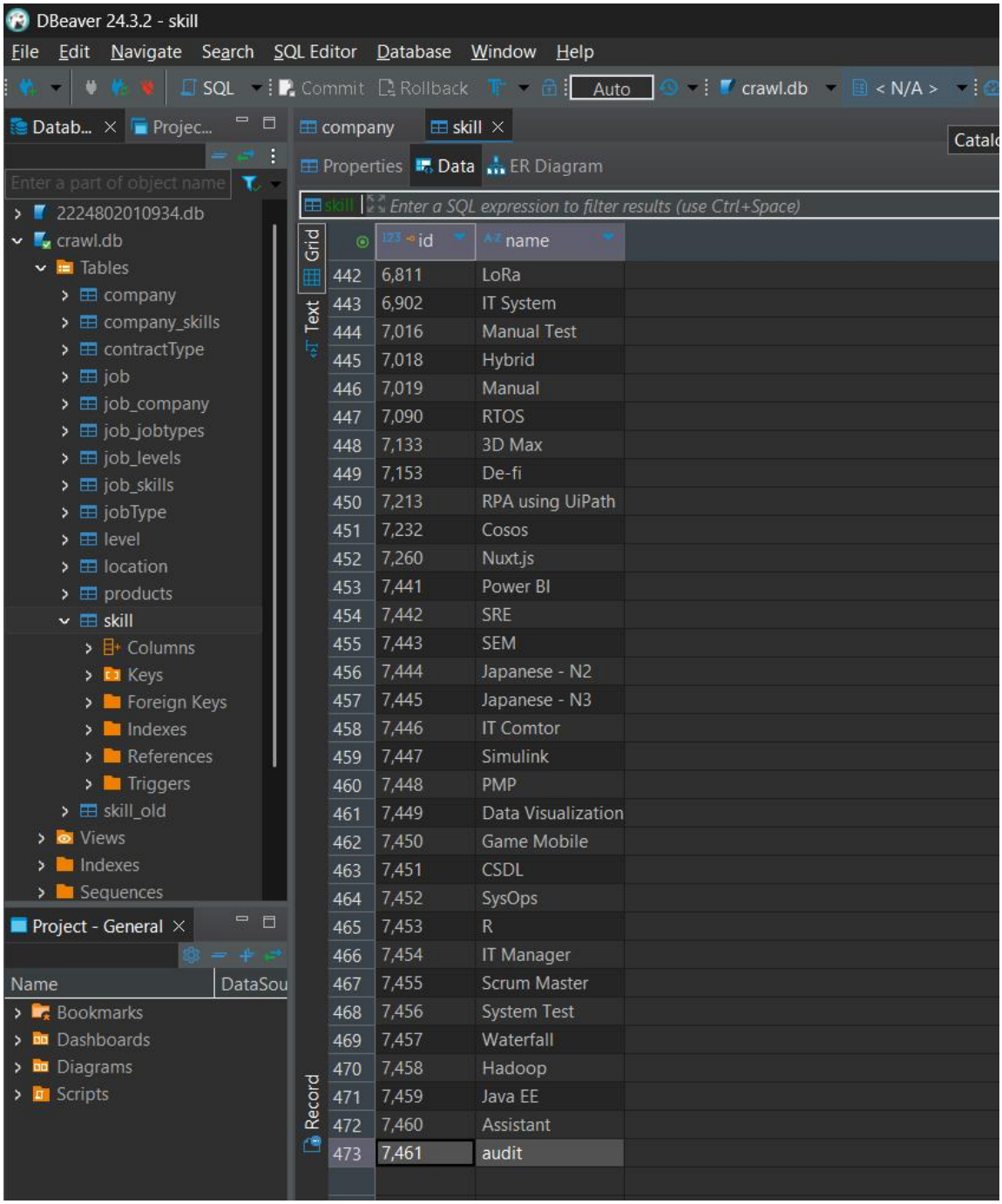
Hình 2.2: Bộ dữ liệu bảng company

2.1.2. Bộ dữ liệu bảng skill

Bộ dữ liệu thu nhập được lưu vào cơ sở dữ liệu crawl trong table skill có 473 dòng dữ liệu và 2 cột bao gồm:

Bảng 2.2: Bộ dữ liệu bảng skill

STT	Tên trường	Kiểu dữ liệu	Mô tả
1	id	INTEGER	Mã id
2	name	TEXT	Tên kỹ năng



Hình 2.3: Bộ dữ liệu bảng skill

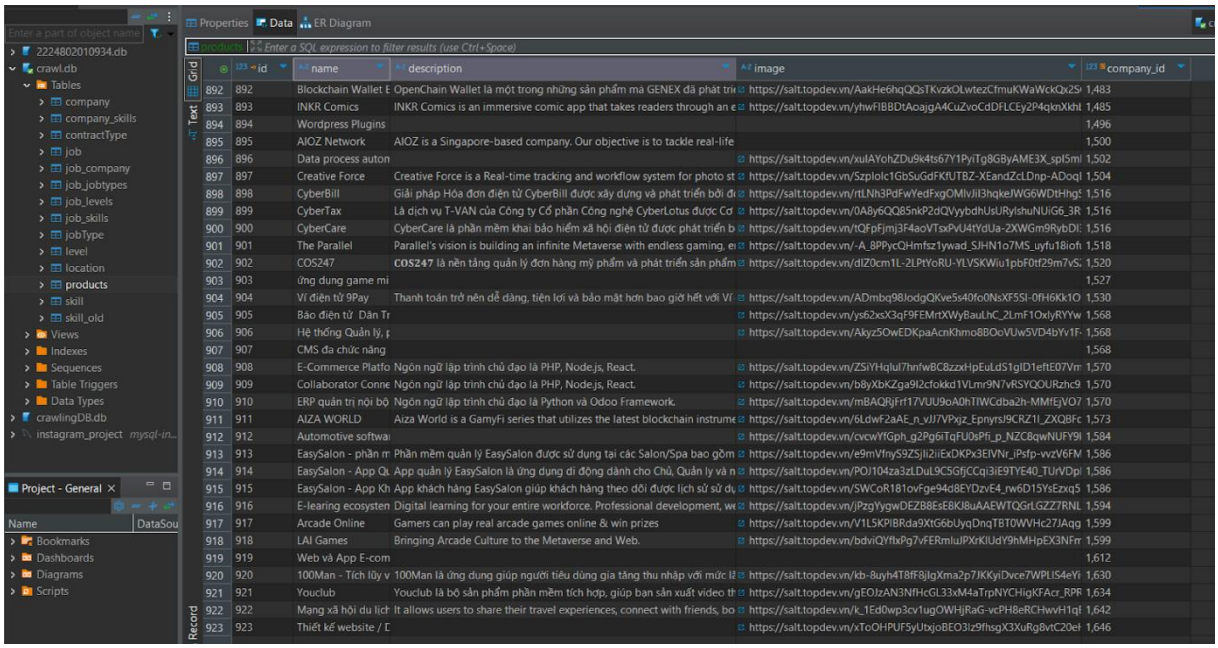
2.1.3. Bộ dữ liệu bảng products

Bộ dữ liệu thu nhập được lưu vào cơ sở dữ liệu crawl trong table products có 923 dòng dữ liệu và 5 cột bao gồm:

Bảng 2.3: Bộ dữ liệu bảng products

STT	Tên trường	Kiểu dữ liệu	Mô tả

1	id	INTEGER	Mã id
2	name	TEXT	Tên sản phẩm của công ty
3	description	TEXT	Mô tả về sản phẩm
4	image	TEXT	Đường dẫn ảnh của sản phẩm
5	company_id	INTEGER	Mã id của công ty



Hình 2.4: Bộ dữ liệu bảng products

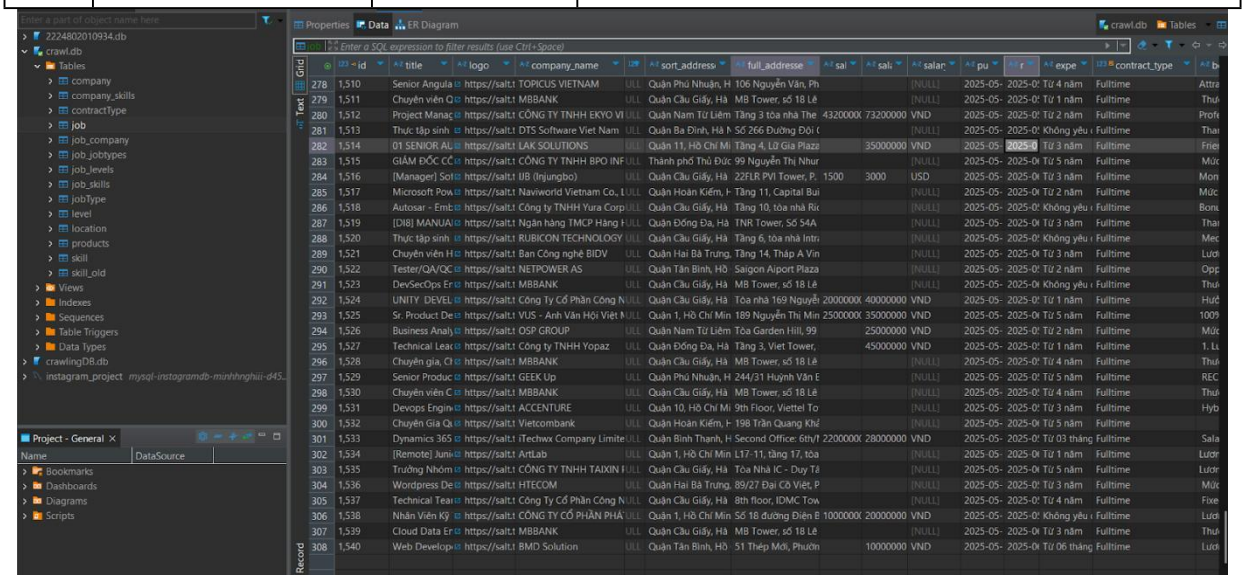
2.1.4. Bộ dữ liệu bảng job

Bộ dữ liệu thu nhập được lưu vào cơ sở dữ liệu crawl trong table job có 308 dòng dữ liệu và 24 cột bao gồm:

Bảng 2.4: Bộ dữ liệu bảng job

STT	Tên trường	Kiểu dữ liệu	Mô tả
1	id	INTEGER	Mã id
2	title	TEXT	Tiêu đề của công việc
3	logo	TEXT	Đường dẫn đến hình ảnh logo
4	company_name	TEXT	Tên của công ty
5	id_company	INTEGER	Mã của công ty
6	sort_addresses	ANY	Địa chỉ ngắn
7	full_addresses	TEXT	Địa chỉ chi tiết
8	salary_min	TEXT	Lương tối thiểu
9	salary_max	TEXT	Lương tối đa
10	salary_currency	TEXT	Đơn vị tiền
11	published_date	TEXT	Ngày đăng tuyển dụng công việc
12	refreshed_date	ANY	
13	experience	TEXT	Yêu cầu số năm kinh nghiệm
14	contract_type	INTEGER	Loại ứng tuyển (Full Time,Freelance,Part Time)

15	benefits	TEXT	Lợi ích
16	content	TEXT	Nội dung
17	responsibilities	TEXT	Trách nhiệm công việc
18	requirements	TEXT	Yêu cầu công việc
19	benefits_origninal	TEXT	Lợi ích thực tế
20	job_url	TEXT	Đường dẫn đến công việc
21	interview	TEXT	Các vòng phỏng vấn
22	job_type	TEXT	Loại công việc
23	level	TEXT	Mực độ
24	skills	TEXT	Các kỹ năng



Hình 2.5: Bộ dữ liệu bảng job

2.2. Tiền xử lý dữ liệu

- Sau khi crawl dữ liệu từ trang web, chúng em tiến hành tiền xử lý dữ liệu để tiện cho việc phân tích và xây dựng các chức năng:

- Chuẩn hóa và làm sạch dữ liệu đầu vào: Kiểm tra và chuyển đổi kiểu dữ liệu cho các trường lương (salary_min, salary_max) từ chuỗi sang số thực, xử lý giá trị rỗng hoặc không hợp lệ.
- Chuyển đổi đơn vị tiền tệ: Nếu lương tính bằng USD, chuyển đổi sang VND theo tỷ giá cố định (usd_to_vnd = 25000).
- Tiền xử lý cho tìm kiếm vectorChuẩn bị dữ liệu mô tả công ty, công việc để xây dựng và tìm kiếm trên vector database (vectordb).
- Tiền xử lý thời gian: Chuyển đổi thời gian từ dạng tương đối dang kiểu thời gian thực ví dụ 2 giờ trước chuyển về 18/05/2025 18:00:00

2.3. Xây dựng hệ thống tìm kiếm

2.3.1. Tìm kiếm theo từ khóa

Hệ thống tìm kiếm được xây dựng trên nền tảng Flask framework và sử dụng thư viện SQLAlchemy ORM để truy xuất dữ liệu từ cơ sở dữ liệu. Tìm kiếm được thực hiện thông qua các truy vấn lọc động như (query.filter()) dựa trên các tham số đầu vào từ URL (GET parameters) như từ khóa, địa điểm, kỹ năng, cấp bậc, …

Cụ thể, trong chức năng tìm kiếm công ty (/company), hệ thống thực hiện lọc theo tên của công ty (Company.name) và (Company.short_address) bằng việc sử dụng toán tử ILIKE, cho phép so khớp không phân biệt hoa thường và hỗ trợ tìm kiếm gần đúng (fuzzy search) .

```python
@main.route("/company")
def company():
    per_page = 40
    page = int(request.args.get("page", 1))

    keyword = request.args.get("keyword", "").strip().lower()
    city = request.args.get("city", "").strip()
    district = request.args.get("district", "").strip()

    query = Company.query

    if keyword:
        query = query.filter(Company.name.ilike(f"%{keyword}%"))
    if district:
        query = query.filter(Company.short_address.ilike(f"%{district}%"))
    elif city:
        query = query.filter(Company.short_address.ilike(f"%{city}%"))

    total_companies = query.count()
    total_pages = math.ceil(total_companies / per_page)
    paginated = query.offset((page - 1) * per_page).limit(per_page).all()
```

Hình 2.6: Tìm kiếm theo tiêu đề

2.3.2. Tìm kiếm theo nội dung

Bước 1: Xây dựng cơ sở dữ liệu vector cho công việc và công ty

Lấy toàn bộ dữ liệu công việc hoặc công ty từ cơ sở dữ liệu.

Kết hợp các trường thông tin quan trọng (ví dụ: tiêu đề, mô tả, kỹ năng, địa chỉ, v.v.) thành một chuỗi văn bản duy nhất cho mỗi bản ghi.

Sử dụng mô hình SentenceTransformer để chuyển các chuỗi văn bản này thành vector đặc trưng (embedding).

Lưu các vector này vào chỉ mục FAISS cùng với danh sách ID tương ứng để phục vụ tìm kiếm nhanh.

```
    You, 14 hours ago • tìm kiếm theo mô hình nhúng
def build_job_vector_db():
    jobs = Job.query.all()
    texts = []
    ids = []
    for job in jobs:
        # Kết hợp các trường chính
        fields = [
            str(job.title or ""),
            str(job.company_name or ""),
            str(job.sort_addresses or ""),
            str(job.full_addresses or ""),
            str(job.experience or ""),
            str(job.level or ""),
            str(job.job_type or ""),
            str(job.skills or ""),
            str(job.content or ""),
            str(job.requirements or ""),
            str(job.responsibilities or ""),
        ]
        text = " | ".join(fields)
        texts.append(text)
        ids.append(job.id)
    model = get_model()
    embeddings = model.encode(texts, show_progress_bar=True, convert_to_numpy=True)
    dim = embeddings.shape[1]
    index = faiss.IndexFlatL2(dim)
    index.add(embeddings)
    with open(VECDB_JOB_META, "wb") as f:
        pickle.dump(ids, f)
    faiss.write_index(index, VECDB_JOB_PATH)
```

Hình 2.7: Xây dựng vector db

Bước 2: Tiến hành tìm kiếm theo nội dung

Khi người dùng nhập từ khóa tìm kiếm nội dung, hệ thống sử dụng mô hình SentenceTransformer để chuyển từ khóa đó thành vector embedding.

Sử dụng FAISS để tìm các vector gần nhất (theo khoảng cách L2) trong cơ sở dữ liệu vector đã xây dựng.

Lấy ra danh sách ID các công việc hoặc công ty phù hợp nhất với nội dung tìm kiếm.

Trả về kết quả tìm kiếm

```python
def search_jobs(query, top_k=10):
    if job_index is None or job_meta is None:
        load_job_vector_db()
    model = get_model()
    emb = model.encode([query], convert_to_numpy=True)
    D, I = job_index.search(emb, top_k)
    results = []
    for idx in I[0]:
        if idx < len(job_meta):
            results.append(job_meta[idx])
    return results
```

Hình 2.8: Tìm kiếm theo công việc

Bước 3:

Lọc và truy vấn lại các bản ghi công việc hoặc công ty dựa trên danh sách ID đã tìm được.

Hiển thị kết quả cho người dùng theo mức độ liên quan đến nội dung tìm kiếm.

2.3.3. Sắp xếp kết quả tìm kiếm:

Nhóm em đã triển khai chức năng sắp xếp kết quả tìm kiếm công việc theo hai tiêu chí chính:

- Thời gian đăng bài (Mới nhất):

Khi người dùng chọn sắp xếp theo thời gian, các công việc sẽ được hiển thị theo thứ tự mới nhất trước, dựa trên trường refreshed_date của mỗi công việc. Điều này giúp người dùng dễ dàng tiếp cận các tin tuyển dụng vừa được cập nhật.

- Mức lương giảm dần:

Khi người dùng chọn sắp xếp theo mức lương, hệ thống sẽ lấy toàn bộ danh sách công việc sau khi lọc, sau đó sắp xếp theo mức lương tối đa từ cao xuống thấp. Đặc biệt, đối với các công việc có mức lương tính bằng USD, nhóm em đã quy đổi sang VND với tỷ giá 1 USD = 25000 VND

Việc sắp xếp này giúp người dùng dễ dàng tìm kiếm các công việc phù hợp với nhu cầu về thời gian và mức thu nhập mong muốn.

```python
# --- Sắp xếp ---
if sort == "date_desc":
    query = query.order_by(Job.refreshed_date.desc())      You, 1 minute ago • Uncommitted changes
elif sort == "salary_desc":
    jobs_all = query.all()
    usd_to_vnd = 25000
    def get_salary_max_vnd(job):
        max_salary = job.salary_max
        try:
            max_salary = float(max_salary) if max_salary is not None and max_salary != '' else None
        except Exception:
            max_salary = None
        currency = (job.salary_currency or '').strip().upper()
        if max_salary is not None and currency == "USD":
            max_salary = max_salary * usd_to_vnd
        return max_salary if max_salary is not None else 0
    jobs_all_sorted = sorted(jobs_all, key=get_salary_max_vnd, reverse=True)
    total_jobs = len(jobs_all_sorted)
    total_pages = math.ceil(total_jobs / per_page)
    jobs = jobs_all_sorted[(page - 1) * per_page : page * per_page]
else:
    total_jobs = query.count()
    total_pages = math.ceil(total_jobs / per_page)
    jobs = query.offset((page - 1) * per_page).limit(per_page).all()
```

Hình 2.9: Sắp xếp theo thời gian và lương công việc

2.3.4. Xây dựng hệ thống gợi ý:

Sử dụng thông tin của công ty hiện tại để đưa ra các gợi ý công ty tương tự khi người dùng vào mục phần chi tiết của một công ty. Mục tiêu của cây quyết định sẽ đưa ra các công ty khác nhau theo thứ tự ưu tiên là :

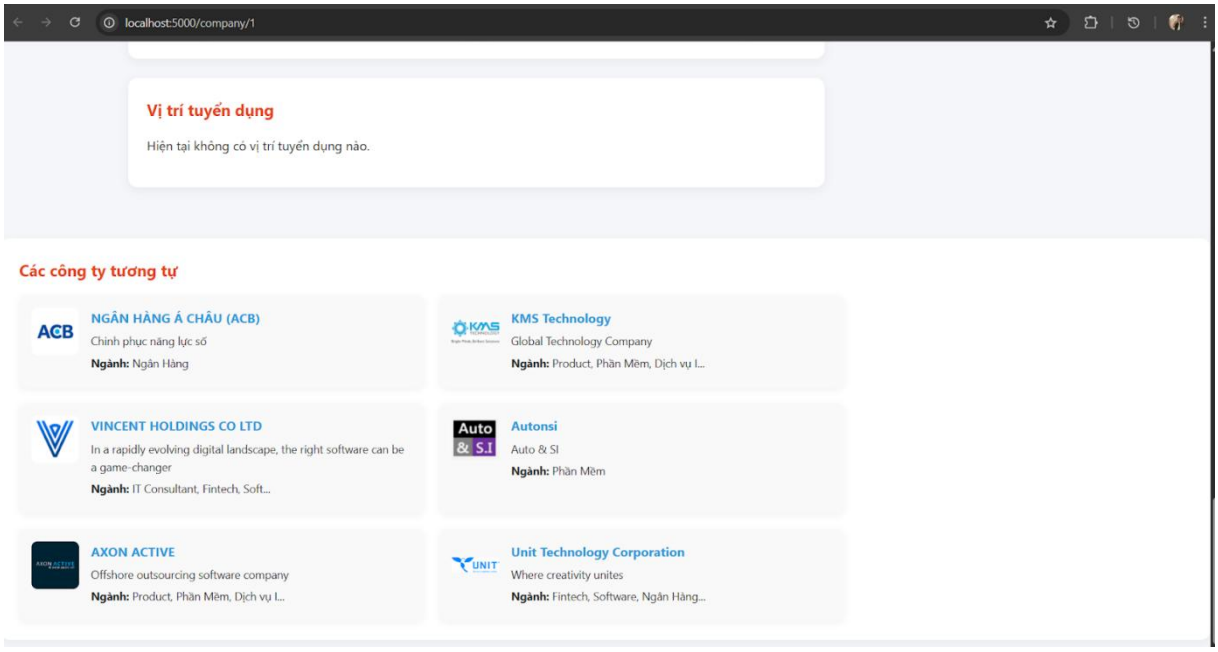
Cùng lĩnh vực
Cùng quy mô
Cùng thành phố
Cùng skills

```python
suggestions = []
for comp in all_other_companies:
    if comp.id == company_id:
        continue
    score = 0
    # 1. Industry match
    if comp.industry == company_info.industry:
        score += 3
    # 2. Size match
    if comp.size == company_info.size:
        score += 2
    # 3. City match
    comp_city = extract_city(comp.short_address)
    if this_city and comp_city and this_city == comp_city:
        score += 2
    # 4. Shared skills
    shared_skills = company_skill_ids & all_skills_map.get(comp.id, set())
    score += len(shared_skills)

    if score > 0:
        suggestions.append((comp, score))

# Sort by score descending
sorted_suggestions = sorted(suggestions, key=lambda x: x[1], reverse=True)      You, 41 minutes ago
```

Hình 2.10: Xây dựng hệ thống gợi ý công ty

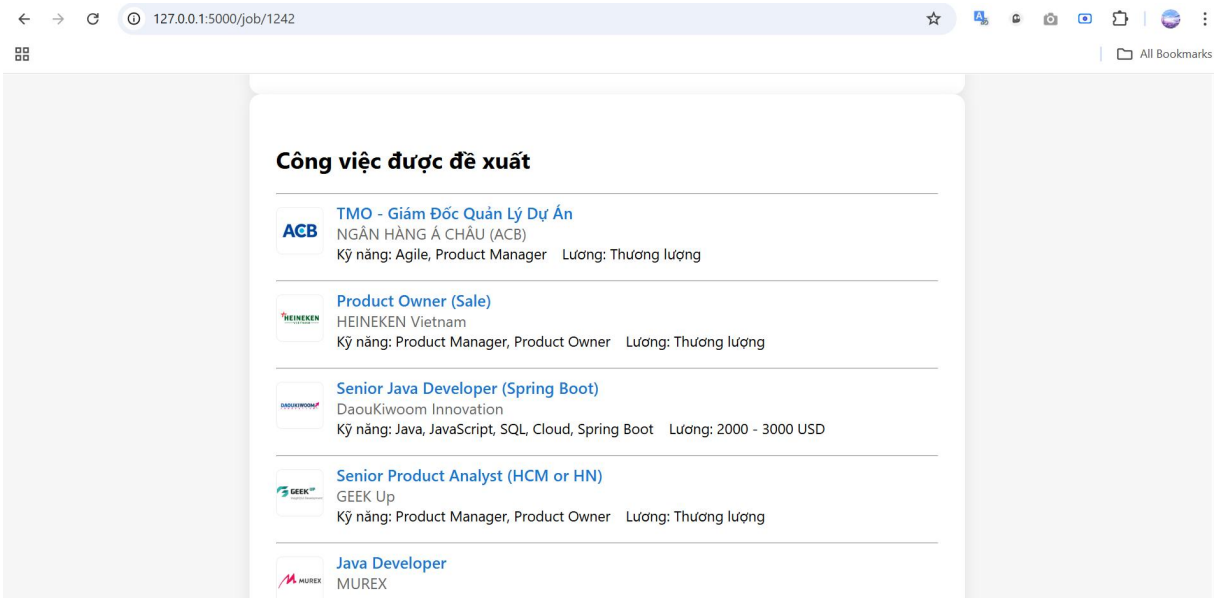


Hình 2.11: Kết quả hệ thống gợi ý công ty

Tương tự, đối với công việc, hệ thống sẽ đề xuất các công việc khác dựa trên các tiêu chí như kỹ năng, địa điểm và kinh nghiệm, đảm bảo các gợi ý phù hợp nhất với thông tin mà người dùng đang quan tâm.

```
for other in all_jobs:
    score = 0
    other_skills = set()
    if isinstance(other.skills, str):
        other_skills = set([s.strip().lower() for s in other.skills.split(',') if s.strip()])
    elif isinstance(other.skills, list):
        other_skills = set([s.strip().lower() for s in other.skills if s.strip()])
    shared_skills = current_skills & other_skills
    score += len(shared_skills)
    other_city = extract_city(other.sort_addresses)
    if current_city and other_city and current_city == other_city:
        score += 2
    other_exp = (other.experience or '').strip().lower()
    if current_exp and other_exp and current_exp == other_exp:
        score += 1
    if score > 0:
        suggestions.append((other, score))
```

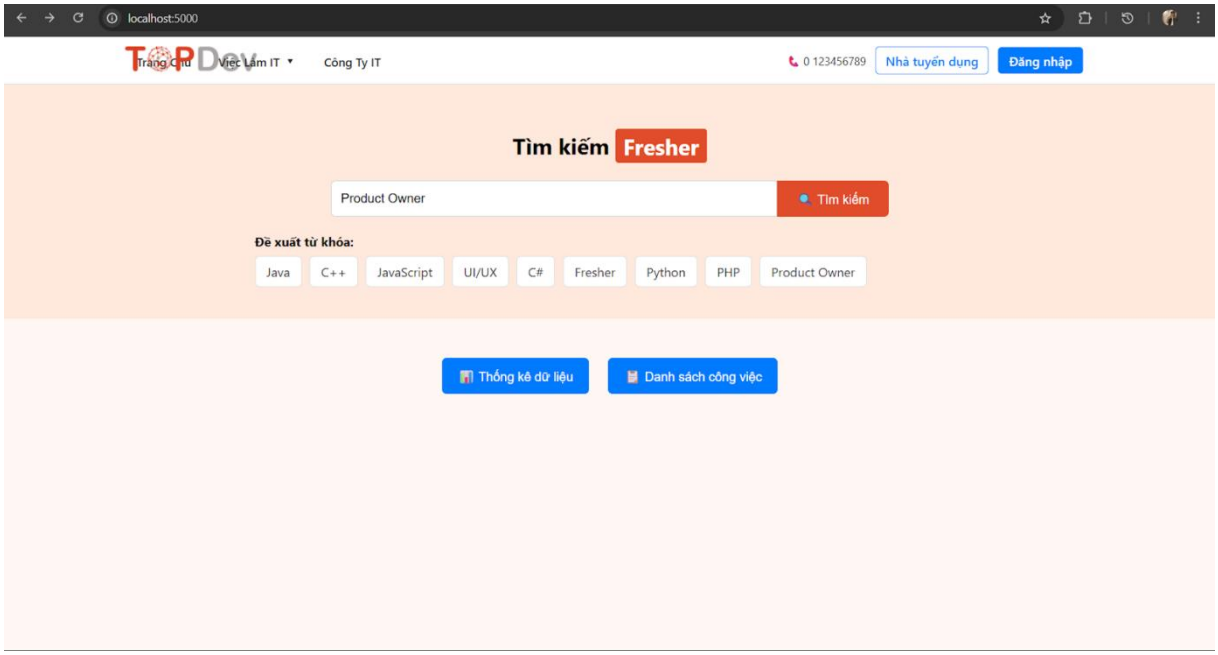
Hình 2.12: Xây dựng hệ thông gợi ý công việc



Hình 2.13: Kết quả hệ thông gợi ý công việc

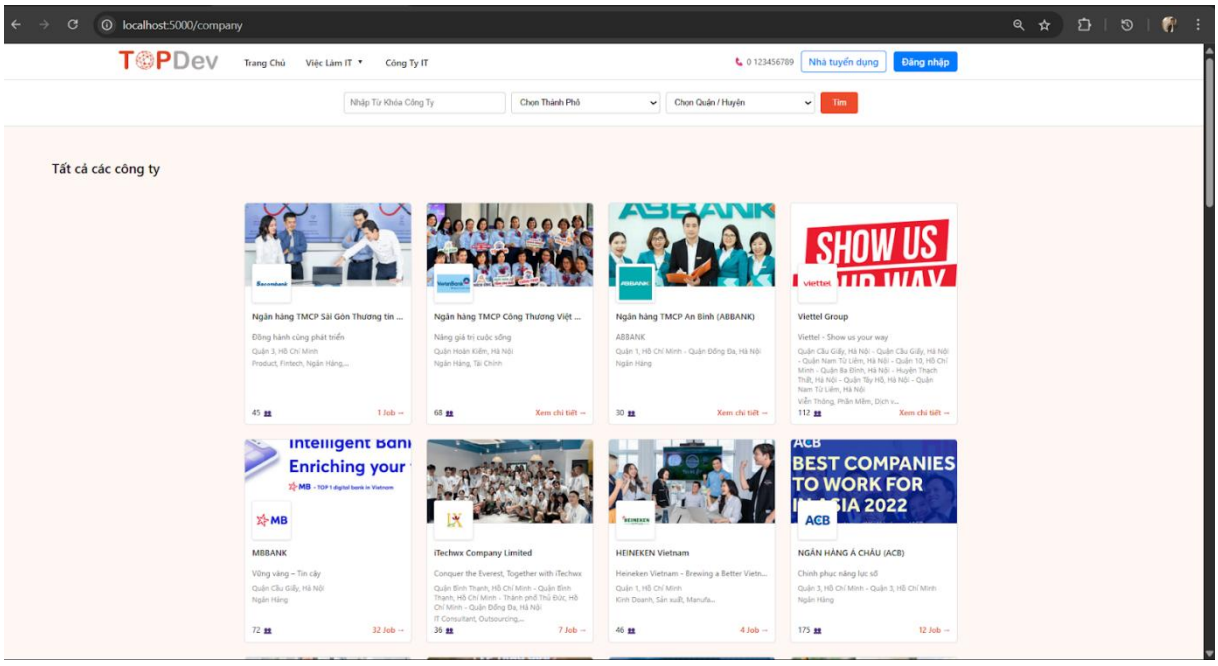
2.4. Xây dựng giao diện hệ thống tìm kiếm

2.4.1. Giao diện trang chủ



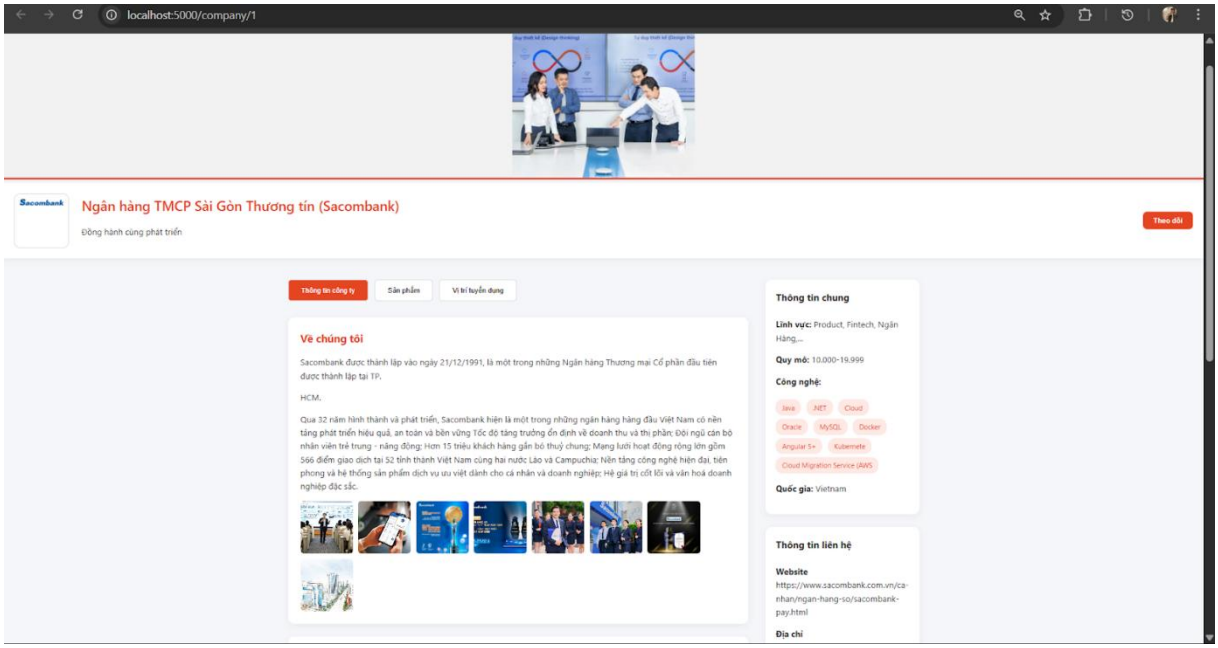
Hình 2.14: Giao diện trang chủ

2.4.2. Giao diện trang công ty



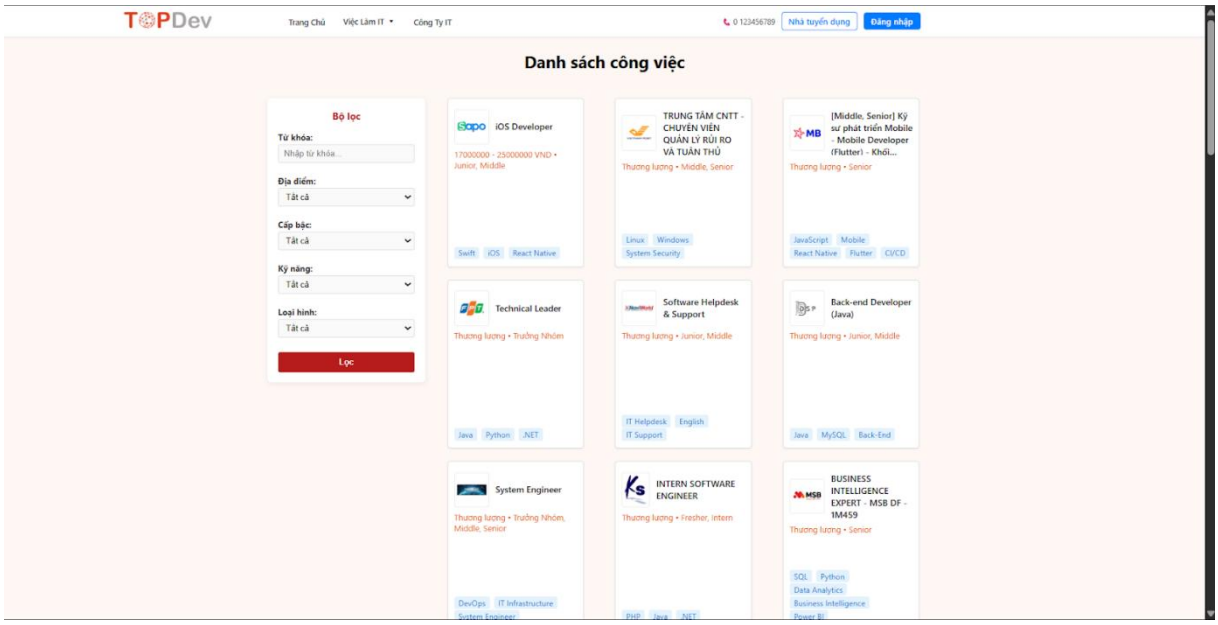
Hình 2.15: Giao diện trang công ty

2.4.3. Giao diện trang chi tiết công ty



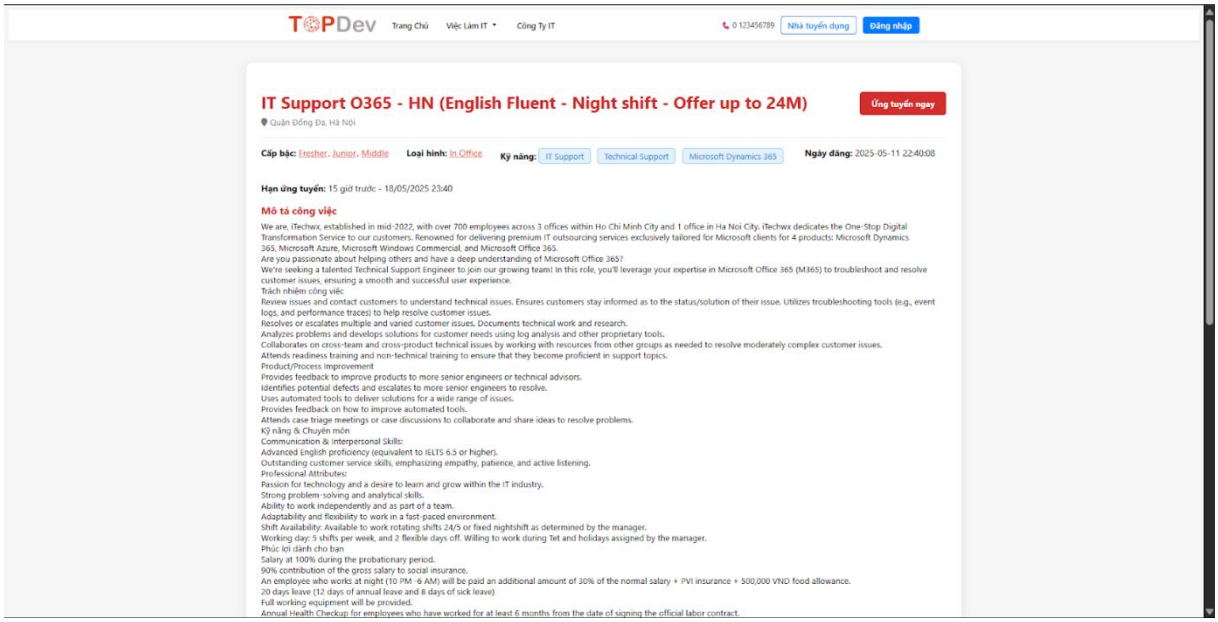
Hình 2.16: Giao diện trang chi tiết công ty

2.4.4. Giao diện trang công việc



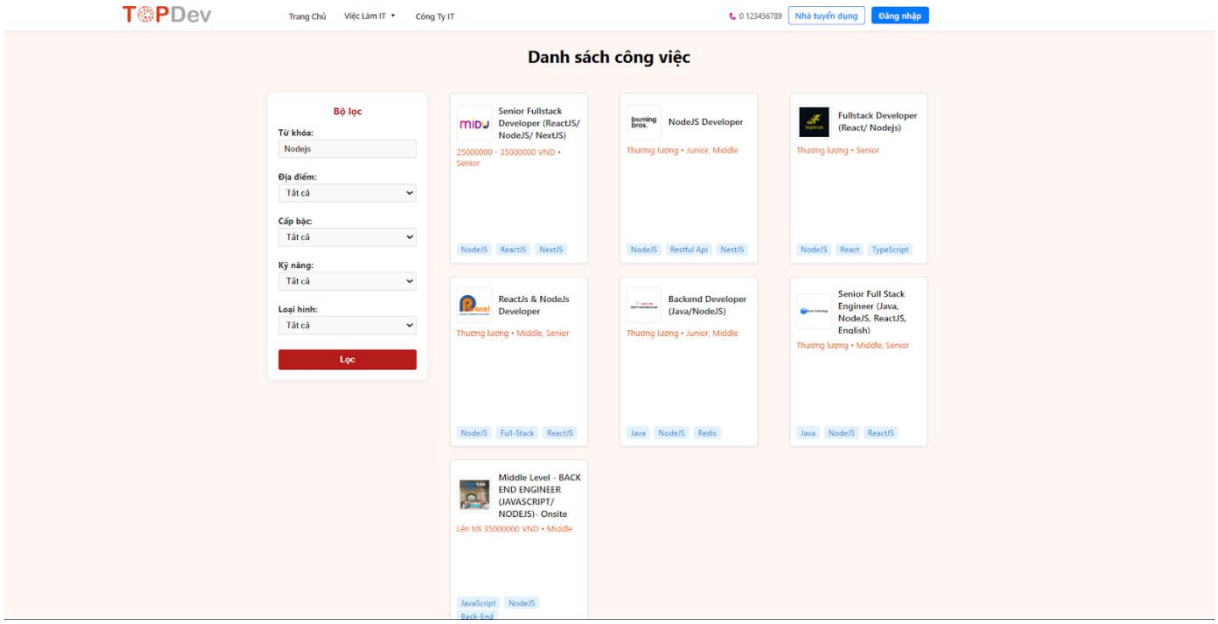
Hình 2.17: Giao diện trang công việc

2.4.5. Giao diện trang chi tiết công việc



Hình 2.18: Giao diện trang chi tiết công việc

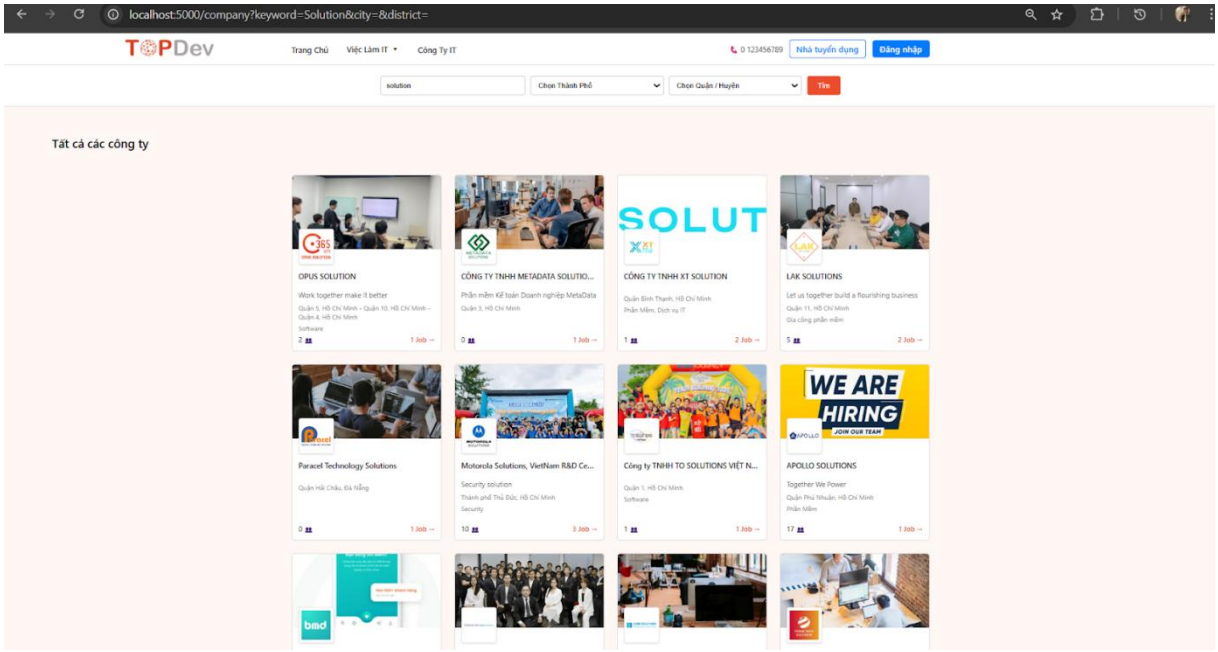
2.4.6. Giao diện trang tìm kiếm công việc



Hình 2.19: Giao diện trang tìm kiếm công việc

Thử với tìm kiếm ở trang công việc là “Nodejs”, hệ thống sẽ tìm kiếm dựa trên keyword được đưa lên url, và tìm kiếm theo title của công việc .

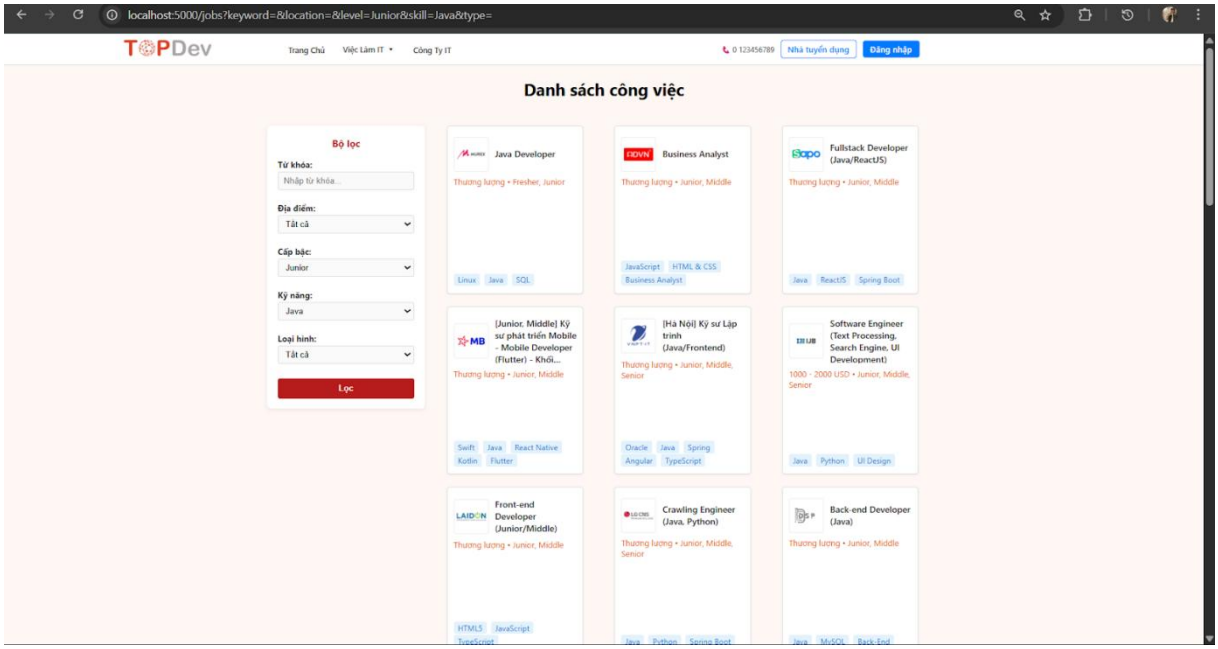
2.4.7. Giao diện trang tìm kiếm công ty



Hình 2.20: Giao diện trang tìm kiếm công ty

Thử tìm kiếm ở trang công ty, tìm kiếm với từ khóa là "solution", hệ thống cũng sẽ lấy keyword trên url và tra cứu theo name của công ty.

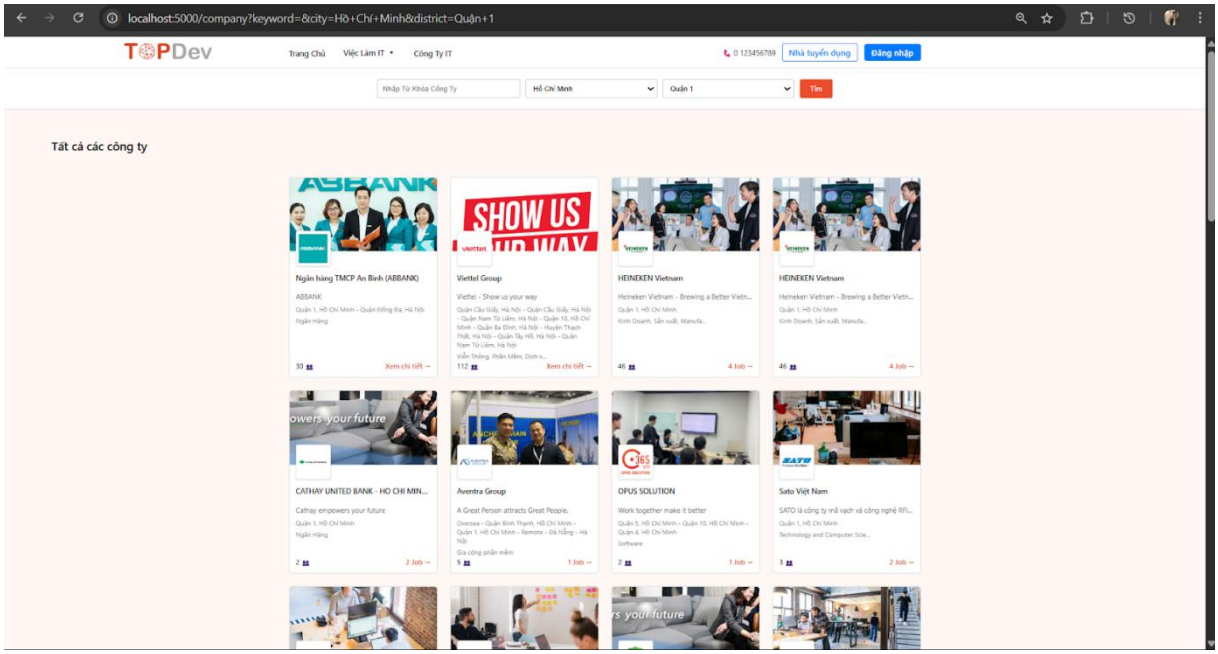
2.4.8. Giao diện trang lọc công việc



Hình 2.21: Giao diện trang lọc công việc

Thử lọc ở trang công việc với cấp bậc là Junior, kỹ năng là Java, hệ thống đưa hai keywords này lên url và bên server lấy chúng và xử lý tìm kiếm trong database và trả về kết quả như hình.

2.4.9. Giao diện trang lọc công ty



Hình 2.22: Giao diện trang lọc công ty

Thử lọc ở trang công ty, với địa điểm là "Thành phố Hồ Chí Minh" và quận là "Quận 1", thì hệ thống tiến hành lấy các keywords trên url và tra cứu với thuộc tính address trong cơ sở dữ liệu và trả về kết quả như hình.

2.4.10. Giao diện trang thống kê

Hình 2.23: Giao diện trang thống kê

CHƯƠNG 3. KẾT LUẬN

1. Kết quả đã làm được

Sau một thời gian tìm hiểu và triển khai đề tài, nhóm đã hoàn thành được các nội dung chính như sau:

- Xây dựng thành công chương trình thu nhập dữ liệu việc làm từ website topdev.vn bằng ngôn ngữ lập trình Python.
- Áp dụng các thư viện như Requests và BeautifulSoup để gửi yêu cầu đến các trang tin tuyển dụng và phân tích, trích xuất dữ liệu HTML .
- Thu thập được các thông tin cơ bản từ bài đăng tuyển dụng như, tên công việc, tên công ty , địa điểm làm việc, mức lương và thời gian đăng tin.
- Xây dựng cơ sở dữ liệu với Flask-SQLAlchemy để lưu trữ dữ liệu thu thập được, đảm bảo dữ liệu có cấu trúc và dễ dàng truy vấn,xử lý.

2. Hạn chế (chưa làm được)

Dù đã đạt được những kết quả nhất định, đề tài vẫn còn một số điểm hạn chế như sau:

- Chưa xử lý được các trang tải động bằng JavaScript: Một số nội dung trên TopDev được render sau khi trang tải xong (dùng JavaScript), khiến Requests không lấy được đầy đủ dữ liệu. Chưa tích hợp thư viện Selenium để giải quyết vấn đề này.
- Chưa crawl toàn bộ thông tin chi tiết công việc: Hiện tại chỉ lấy được thông tin từ trang danh sách, chưa đi sâu vào từng trang chi tiết của công việc để lấy mô tả đầy đủ.
- Chưa xử lý chống trùng lặp dữ liệu: Khi crawl nhiều lần, dữ liệu có thể bị lặp nếu không có kiểm tra ràng buộc hoặc xác định khóa chính.
- Chưa có giao diện người dùng (UI): Chương trình hiện tại chỉ chạy qua dòng lệnh, chưa có giao diện web hoặc dashboard để hiển thị và tra cứu dữ liệu trực quan.

3. Thuận lợi và khó khăn

a. Thuận lợi

Đề tài thực tế và có ứng dụng rõ ràng: Việc thu thập dữ liệu tuyển dụng từ TopDev phù hợp với xu hướng phân tích dữ liệu ngành CNTT, đồng thời hữu ích trong định hướng nghề nghiệp và nghiên cứu thị trường lao động.

Nguồn tài liệu học tập và thư viện hỗ trợ phong phú: Python có nhiều thư viện mạnh để xử lý web crawling, đồng thời có cộng đồng hỗ trợ lớn giúp việc giải quyết lỗi và mở rộng chương trình dễ dàng hơn.

Website TopDev có cấu trúc HTML tương đối rõ ràng: Các thẻ chứa thông tin công việc được đặt tên theo class dễ hiểu, giúp dễ dàng định vị và bóc tách dữ liệu với BeautifulSoup.

b. Khó khăn

Nội dung trang tải động bằng JavaScript: Requests không thể lấy dữ liệu được nếu nội dung không nằm sẵn trong mã HTML, gây khó khăn trong việc crawl đầy đủ thông tin.

Thiết kế chống bot và chặn IP: Website có thể áp dụng các cơ chế chống crawl như giới hạn tần suất truy cập hoặc yêu cầu xác thực, điều này khiến việc thu thập dữ liệu liên tục gặp trở ngại.

Không có API công khai: TopDev không cung cấp API mở nên nhóm buộc phải phân tích mã HTML thủ công để tìm kiếm thông tin cần thiết, tốn thời gian và dễ gặp lỗi khi giao diện thay đổi.

Phải cập nhật khi cấu trúc web thay đổi: Khi trang web thay đổi cấu trúc, mã crawl có thể không còn hoạt động đúng, cần cập nhật liên tục.

TÀI LIỆU THAM KHẢO

[1] Python Software Foundation, Python Documentation. [Online]. https://docs.python.org/3/. Ngày truy cập: 18/05/2025.
[2] Pallets Community, Flask's Documentation. [Online]. https://flask.palletsprojects.com/en/stable/. Ngày truy cập: 18/05/2025.
[3] Viblo, Crawl data đầy đủ hơn với thư viện selenium, 1 January 2024. [Online]. https://viblo.asia/p/crawl-data-day-du-hon-voi-thu-vien-selenium-EbNVQ5GmVvR. Ngày truy cập: 18/05/2025.
[4] CodeLearn, Web Crawling Với BeautifulSoup4 Trong Python, 11 July 2020. [Online]. https://codelearn.io/sharing/web-crawling-voi-beautifulsoup4-python?srsltid=AfmBOorszdWD-HdqdKTXS9iFlId6UKO8xbEAmwwOfE7zg0iXD5mHJMpZ. Ngày truy cập: 18/05/2025.
[5] Pinecone, Introduction to Facebook AI Similarity Search (Faiss). [Online]. https://www.pinecone.io/learn/series/faiss/faiss-tutorial/. Ngày truy cập: 18/05/2025.
[6] LangChain, Faiss. [Online]. https://python.langchain.com/docs/integrations/vectorstores/faiss/. Ngày truy cập: 18/05/2025.