

**TRƯỜNG ĐẠI HỌC THỦ DẦU MỘT  
VIỆN KỸ THUẬT CÔNG NGHỆ**



**BÁO CÁO TỔNG KẾT  
ĐỀ TÀI NGHIÊN CỨU KHOA HỌC CỦA SINH VIÊN CẤP ĐƠN VỊ  
NĂM HỌC 2021-2022  
XÂY DỰNG MÔ HÌNH HỆ TƯ VẤN TÌM VIỆC ONLINE CHO SINH  
VIÊN TRONG TRẠNG THÁI BÌNH THƯỜNG MỚI**

*Người hướng dẫn:* **Ths. DƯƠNG THỊ KIM CHI**

*Người thực hiện:* **HOÀNG KIM TUYẾN – 1824801040043  
TRẦN QUỐC DŨNG – 1824801040013**

Lớp: D18HT01

Khoá: D18

Viện: Kỹ Thuật – Công Nghệ

**BÌNH DƯƠNG, NĂM 2022**

## **LỜI CẢM ƠN**

Nhóm em xin gửi tới lời cảm ơn chân thành đến cô **Dương Thị Kim Chi** đã trang bị cho chúng em những kiến thức và kỹ năng cần có để thực hiện đề tài báo cáo nghiên cứu khoa học này.

Trong quá trình thực hiện, nhóm em đã cố gắng áp dụng những kiến thức **Cô** đã giảng dạy vào trong tiểu luận. Tuy nhiên trong quá trình thực hiện chắc chắn không thể không gặp phải những khó khăn, thiếu sót. Rất mong nhận được sự quan tâm, góp ý của hội đồng nghiệm thu để đề tài nhóm em được hoàn chỉnh và tốt hơn.

**Xin chân thành cảm ơn!**

## LỜI CAM ĐOAN

Chúng tôi xin cam đoan đây là sản phẩm đồ án của riêng chúng tôi và được sự hướng dẫn của **Cô Dương Thị Kim Chi**. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong đồ án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung nghiên cứu của mình. Trường Đại học Thủ Dầu Một không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

*TP. Thủ Dầu Một, ngày tháng năm*

*Tác giả*

*(ký tên và ghi rõ họ tên)*

*Hoàng Kim Tuyền*

*Trần Quốc Dũng*

**PHẦN ĐÁNH GIÁ CỦA HỘI ĐỒNG NGHIỆM THU**

---

---

---

---

---

---

---

TP. Thủ Dầu Một, ngày    tháng    năm  
(kí và ghi họ tên)

## TÓM TẮT

Trong quá trình thực hiện nghiên cứu đề tài, nhóm đã tiếp cận bài toán, và vấn đề bài toán của nhóm là xây dựng hệ tư vấn tìm việc bằng phương pháp lọc cộng tác. Nội dung của bài toán là giải quyết vấn đề hỗ trợ người tìm việc có một công việc phù hợp. Đề tài ứng dụng hệ tư vấn vào để giải quyết bài toán trên, nhằm tư vấn cho người tìm việc.

Đề tài báo cáo trình bày các kỹ thuật xử lý dữ liệu và đề xuất một thuật toán hiệu quả để giải quyết bài toán hệ tư vấn. Dựa vào yêu cầu bài toán, nhóm đã tập trung vào hai công việc chính: xử lý dữ liệu và xây dựng hệ tư vấn. Nhóm đã sử dụng phương pháp lọc cộng tác để tạo ra sự tương đồng giữa thông tin yêu cầu của nhà tuyển dụng và thông tin người tuyển dụng và từ đó tư vấn các công việc phù hợp cho người tìm việc.

UBND TỈNH BÌNH DƯƠNG  
TRƯỜNG ĐẠI HỌC THỦ DẦU MỘT

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

## THÔNG TIN VỀ SINH VIÊN CHỊU TRÁCH NHIỆM CHÍNH THỰC HIỆN ĐỀ TÀI

### I. SƠ LƯỢC VỀ SINH VIÊN:

Họ và tên: Hoàng Kim Tuyền

Sinh ngày: 3 tháng 5 năm 2000

Nơi sinh: Lạng Sơn

Lớp: D18HT01

Khóa: 2018-2022

Viện: Kỹ Thuật Công Nghệ

Địa chỉ liên hệ: 16 Đoàn Thị Liên, Phường Phú Lợi, TP Thủ Dầu Một, Bình Dương

Điện thoại: 0347133942

Email: hoangtuyenblogger@gmail.com



### II. QUÁ TRÌNH HỌC TẬP (kê khai thành tích của sinh viên từ năm thứ 1 đến năm đang học):

#### \* Năm thứ 1:

Ngành học: Hệ thống thông tin

Viện: Kỹ Thuật Công Nghệ

Kết quả xếp loại học tập: Khá

Sơ lược thành tích: Khá

#### \* Năm thứ 2:

Ngành học: Hệ thống thông tin

Viện: Kỹ Thuật Công Nghệ

Kết quả xếp loại học tập: Khá

Sơ lược thành tích: Khá

#### \* Năm thứ 3:

Ngành học: Hệ thống thông tin

Viện: Kỹ Thuật Công Nghệ

Kết quả xếp loại học tập: Khá

Sơ lược thành tích: Khá

**\* Năm thứ 4:**

Ngành học: Hệ thống thông tin

Viện: Kỹ Thuật Công Nghệ

Kết quả xếp loại học tập: Khá

Sơ lược thành tích: Khá

Ngày      tháng      năm

**Xác nhận của lãnh đạo khoa**  
(ký, họ và tên)

**Sinh viên chịu trách nhiệm chính**  
**thực hiện đề tài**  
(ký, họ và tên)

UBND TỈNH BÌNH DƯƠNG  
TRƯỜNG ĐẠI HỌC THỦ DẦU MỘT

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

## THÔNG TIN KẾT QUẢ NGHIÊN CỨU CỦA ĐỀ TÀI

### 1. Thông tin chung:

- Tên đề tài: *Xây dựng mô hình hệ tư vấn tìm việc online cho sinh viên trong trạng thái bình thường mới*
- Người hướng dẫn: **Ths. Dương Thị Kim Chi**
- Sinh viên/ nhóm sinh viên thực hiện:

STT	Họ và tên	MSSV	Lớp	Khoa	Năm thứ/ Số năm đào tạo
1	<b>Hoàng Kim Tuyền</b> (Nhóm trưởng)	<b>1824801040043</b>	<b>D18HT01</b>	<b>Kĩ Thuật – Công Nghệ</b>	<b>4/4</b>
2	<b>Trần Quốc Dũng</b>	<b>1824801040013</b>	<b>D18HT01</b>	<b>Kĩ Thuật – Công Nghệ</b>	<b>4/4</b>

### 2. Mục tiêu đề tài:

- Tìm hiểu về hệ thống gợi ý, phương pháp lọc cộng tác
- Xây dựng mô hình tư vấn cho sinh viên nhóm ngành Công nghệ thông tin trường

Đại Học Thủ Dầu Một

### 3. Tính mới và sáng tạo:

- Xây dựng được mô hình lọc cộng tác trên bộ dữ liệu tuyển dụng và ứng viên
- Xây dựng được website áp dụng mô hình lọc cộng tác
- Website đã xây dựng được có các chức năng tìm kiếm, xem bài viết và hiển thị các công việc gợi ý bởi hệ thống lọc cộng tác.

-

### 4. Kết quả nghiên cứu:

- Thành công tìm hiểu và xây dựng được mô hình lọc cộng tác
- Xây dựng thành công website và ứng dụng mô hình lọc cộng tác vào ứng dụng



- Website ứng dụng mô hình lọc cộng tác hoạt động được trên bộ dữ liệu đã thu thập và đưa ra được các công việc gợi ý bởi phương pháp lọc cộng tác

**5. Đóng góp về mặt kinh tế - xã hội, giáo dục và đào tạo, an ninh, quốc phòng và khả năng áp dụng của đề tài:**

- Đề tài có khả năng ứng dụng vào hệ thống tìm kiếm việc làm trên các website tuyển dụng, các ứng dụng mạng xã hội, thương mại điện tử.

**6. Công bố khoa học của sinh viên từ kết quả nghiên cứu của đề tài** (*ghi rõ họ tên tác giả, nhan đề và các yếu tố về xuất bản nếu có*) hoặc nhận xét, đánh giá của cơ sở đã áp dụng các kết quả nghiên cứu (*nếu có*):

Ngày      tháng      năm  
**Sinh viên chịu trách nhiệm chính**  
**thực hiện đề tài**  
(*ký, họ và tên*)

**Nhận xét của người hướng dẫn về những đóng góp khoa học của sinh viên thực hiện đề tài):**

**Xác nhận của lãnh đạo khoa**  
(*ký, họ và tên*)

Ngày      tháng      năm  
**Người hướng dẫn**  
(*ký, họ và tên*)

## MỤC LỤC

LỜI CẢM ƠN .....	i
TÓM TẮT .....	iv
MỤC LỤC .....	i
DANH MỤC KÍ HIỆU VÀ CHỮ VIẾT TẮT .....	iv
DANH MỤC CÁC HÌNH VẼ.....	v
DANH MỤC CÁC BẢNG.....	vii
CHƯƠNG 1: TỔNG QUAN BÀI TOÁN.....	1
1.1 Lý do chọn đề tài.....	1
1.2 Mục đích nghiên cứu.....	1
1.3 Đối tượng nghiên cứu và phạm vi nghiên cứu.....	2
1.4 Phương pháp nghiên cứu .....	2
1.5 Bố cục của đề án .....	2
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT VÀ CÁC NGHIÊN CỨU LIÊN QUAN .....	3
2.1 Tổng quan về học máy .....	3
2.1.1 Tổng Quan .....	3
2.1.2 Các ứng dụng của học máy .....	4
2.3 Tổng quan về hệ thống gợi ý .....	4
2.3.1 Tổng quan .....	4
2.3.2 Các phương pháp trong hệ thống gợi ý .....	5
2.4 Hệ thống gợi ý lọc cộng tác .....	6
2.4.1 Tổng quan phương pháp lọc cộng tác .....	6
2.4.2 Ưu và nhược điểm .....	7
2.5 Các ứng dụng của hệ thống gợi ý .....	8
2.5.1 Các hệ thống gợi ý.....	8

2.5.2	Hệ thống gợi ý dựa trên phương pháp lọc cộng tác .....	9
2.6	Framework Django .....	10
2.6.1	Ưu điểm của Django .....	10
2.6.2	Nhược điểm của Django .....	11
2.6.3	MVT Pattern của Django .....	12
2.6.4	Các thành phần cơ bản của ứng dụng Django .....	13
2.6.5	Lí do chọn Django trong lập trình web.....	14
CHƯƠNG 3:	MÔ HÌNH TỔNG QUÁT.....	15
3.1	Phương pháp lọc cộng tác .....	15
3.1.1	Lọc cộng tác dựa trên bộ nhớ.....	15
3.1.2	Lọc cộng tác dựa trên mô hình .....	17
3.2	Mô hình tổng quát bài toán .....	18
3.2.1	Mô hình tổng quát thu thập dữ liệu.....	19
3.2.2	Mô hình tổng quát xây dựng mô hình lọc cộng tác .....	21
3.2	Use Case của hệ thống gợi ý .....	23
CHƯƠNG 4:	THỰC NGHIỆM .....	25
4.1	Các công nghệ và thư viện sử dụng .....	25
4.2	Dữ liệu.....	25
4.2.2	Thu thập dữ liệu .....	25
4.2.2	Xem và xử lý dữ liệu.....	30
4.3	Xây dựng mô hình gợi ý .....	31
4.3.1	Trích xuất các trường dữ liệu.....	31
4.3.2	Xây dựng mô hình lọc cộng tác .....	34
4.4	Xây dựng Website demo ứng dụng phương pháp lọc cộng tác(Django Framework) .....	37
4.4.1	Khởi tạo project Django.....	37
4.4.2	Khai báo Class và khởi tạo database.....	38

4.3.3 Giao diện hệ thống gợi ý.....	39
4.3.4 Giao diện trang quản trị .....	43
CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN CHO ĐỀ TÀI .....	51
5.1 Kết luận .....	51
5.2 Hướng phát triển .....	51
TÀI LIỆU THAM KHẢO.....	52

## **DANH MỤC KÍ HIỆU VÀ CHỮ VIẾT TẮT**

### **CÁC KÝ HIỆU**

### **CÁC CHỮ VIẾT TẮT**

MVC ( Model-View-Controller)

MTV (Model-Template-Views)

## DANH MỤC CÁC HÌNH VẼ

Hình 1 Sơ đồ phương pháp lọc cộng tác .....	7
Hình 2 Django framework .....	10
Hình 3 Mô hình MVT của Django .....	12
Hình 4 Use case của user .....	23
Hình 5 Use case của Admin .....	24
Hình 6 Thông tin dữ liệu thông tin tuyển dụng .....	30
Hình 7 Thông tin dữ liệu thông tin người tìm việc .....	31
Hình 8 Tạo list ‘title’ chứa job_name, title của ứng viên và nhà tuyển dụng .....	32
Hình 9 Bảng điểm số giữa employer và jobs .....	33
Hình 10 Datafamre .....	33
Hình 11 Dataframe của scores_matrix .....	34
Hình 12 Dataframe của scores_matrix .....	35
Hình 13 Dataframe của scores_matrix .....	35
Hình 14 Hàm gợi ý việc làm truyền vào ID công việc & số lượng công việc .....	36
Hình 15 Lưu model .....	36
Hình 16 Cấu trúc project của ứng dụng gợi ý .....	37
Hình 17 Cấu trúc thư mục Web của ứng dụng gợi ý .....	37
Hình 18 Khai báo 2 class Employer và Jobs .....	38
Hình 19 Trang chủ tìm kiếm của ứng dụng .....	39
Hình 20 Trang chủ tìm kiếm của ứng dụng .....	40
Hình 21 Giao diện bài post .....	41
Hình 22 Phần footer .....	42
Hình 23 Giao diện đăng nhập trang admin .....	43
Hình 24 Giao diện đăng nhập trang admin .....	43
Hình 25 Giao diện đăng nhập trang admin .....	44

Hình 26 Quản trị Employer.....	45
Hình 27 Quản trị Employer.....	46
Hình 28 Thêm mới một employer.....	47
Hình 29 Sửa/xóa 1 employer.....	48
Hình 30 Thêm mới 1 jobs .....	49
Hình 31 Sửa/xóa 1 jobs .....	50

## **DANH MỤC CÁC BẢNG**

Bảng 1: các phương pháp hệ thống gợi ý.....	3
Bảng 2: Mô tả dữ liệu người tìm việc(employer.csv).....	27
Bảng 3: Mô tả dữ liệu thông tin tuyển dụng(jobs.csv) .....	28



# CHƯƠNG 1: TỔNG QUAN BÀI TOÁN

## 1.1 Lý do chọn đề tài

Ngày nay với sự phát triển của kinh tế, đòi hỏi nguồn lao động phải dồi dào để ứng được thị trường kinh tế mới, với sự tăng trưởng không ngừng của các nền kinh tế nên trong lao động luôn đòi hỏi nguồn lực phải có chuyên môn cao và kinh nghiệm làm việc. Chính vì vậy luôn tạo ra nhiều cơ hội cho người tìm việc, tuy nhiên để tìm một công việc phù hợp với chuyên môn chuyên ngành cũng như là kinh nghiệm làm việc của người tìm việc thì rất khó. Như chúng ta có thể thấy nhiều người làm việc tại các cơ quan hoặc các doanh nghiệp có xảy ra tình trạng các công nhân làm việc nhưng không đúng chuyên môn hay chuyên ngành của mình.

Trong thời gian vừa qua, nhiều địa phương trong tỉnh phải thực hiện các quy định giãn cách xã hội để phòng, chống dịch bệnh Covid-19 nên hầu hết doanh nghiệp, cơ sở sản xuất kinh doanh trên địa bàn đều tạm dừng hoạt động hoặc thu hẹp quy mô sản xuất kinh doanh, cắt giảm số lượng nhân công dẫn đến người lao động buộc phải tạm hoãn thực hiện hợp đồng lao động, ngừng làm việc tạm thời, thậm chí mất việc làm.

Sau khi tỉnh thực hiện tiêm vaccin cho toàn dân và bắt đầu “sống chung với dịch” và bắt đầu quay lại làm việc. Không chỉ gây ảnh hưởng tiêu cực đến tình trạng việc làm, đại dịch Covid-19 còn khiến hoạt động tuyển dụng việc làm gặp nhiều khó khăn. Chính vì vậy nhóm em chọn đề tài " Xây dựng mô hình hệ tư vấn tìm việc online cho sinh viên trong trạng thái bình thường mới" giúp các bạn sinh viên tìm được vị trí việc làm phù hợp với năng lực và mức lương mong muốn của mình.

## 1.2 Mục đích nghiên cứu

Đề tài “ **Xây dựng mô hình hệ tư vấn tìm việc online cho sinh viên trong trạng thái bình thường mới**” , là đề tài hướng tới xây dựng hệ thống gợi ý các công việc phù hợp với người tìm việc thông qua phương pháp lọc cộng tác. Việc áp dụng phương pháp

này vào việc tìm ra mối tương đồng của yêu cầu của nhà tuyển dụng và thông tin của người tìm việc.

### **1.3 Đối tượng nghiên cứu và phạm vi nghiên cứu**

Với mục đích xây dựng được hệ thống gợi ý trên, đề tài xác định các đối tượng và phạm vi nghiên cứu như sau:

- Đối tượng nghiên cứu là: thông tin của người tìm việc và thông tin tuyển dụng của nhà tuyển dụng.
- Phạm vi nghiên cứu: Sinh viên nhóm ngành Công Nghệ Thông Tin, thuộc trường Đại Học Thủ Dầu Một.

### **1.4 Phương pháp nghiên cứu**

- **Phương pháp nghiên cứu lý thuyết:** Nghiên cứu các tài liệu về hệ tư vấn, nghiên cứu sử dụng các kiến thức của học máy, nghiên cứu sử dụng phương pháp *lọc cộng tác*.
- **Phương pháp nghiên cứu thực tiễn:** Sau khi nghiên cứu các lý thuyết liên quan về hệ tư vấn, xác định được bài toán, đưa ra hướng giải quyết cho bài toán dựa vào các lý thuyết đã nghiên cứu ở trên, ứng dụng vào để đưa ra kết quả của hệ thống gợi ý.

### **1.5 Bố cục của đồ án**

Nội dung của đồ án được chia thành bố cục như sau:

Chương 1: Tổng quan về bài toán

Chương 2: Cơ sở lý thuyết và các nghiên cứu liên quan

Chương 3: Mô hình tổng quát

Chương 4: Thực nghiệm

Chương 5: Kết quả và cải tiến cho đề tài

## CHƯƠNG 2: CƠ SỞ LÝ THUYẾT VÀ CÁC NGHIÊN CỨU LIÊN QUAN

### 2.1 Tổng quan về học máy

#### 2.1.1 Tổng Quan

Học máy (Machine Learning) là một lĩnh vực của trí tuệ nhân tạo, nhiệm vụ là xây dựng các hệ thống tự động “học từ dữ liệu” để giải quyết những vấn đề cụ thể.

Phân loại: Có hai loại phương pháp học máy chính

- Phương pháp quy nạp: Máy học/phân biệt các khái niệm dựa trên dữ liệu đã thu thập được trước đó. Phương pháp này cho phép tận dụng được nguồn dữ liệu rất nhiều và sẵn có.
- Phương pháp suy diễn: Máy học/phân biệt các khái niệm dựa vào các luật. Phương pháp này cho phép tận dụng được các kiến thức chuyên ngành để hỗ trợ máy tính.

Các nhóm giải thuật học máy:

- **Học có giám sát:** thuật toán học nhận một số thí dụ luyện tập và nhận thí dụ tương ứng để xây dựng mô hình dự đoán. Sử dụng khi bạn muốn nhận dự đoán một kết quả đầu ra từ dữ liệu đầu vào và bạn có các cặp dữ liệu (đầu vào/đầu ra) tương ứng. Để xây dựng mô hình học có giám sát thường phải có sự nỗ lực từ con người để gán nhãn cho tập dữ liệu.
- **Học không giám sát:** Không bắt kì nhãn nào được cung cấp cho thuật toán học, thuật toán sẽ đi tìm cấu trúc bên dưới dữ liệu. Mảng này gồm nhiều bài toán khác nhau như: phân cụm (tự gán nhãn cho các cụ dữ liệu có cấu trúc tương tự nhau, giảm chiều dữ liệu để hỗ trợ các hình thức học khác,...)
- **Học nửa giám sát:** Một dạng lai giữa hai nhóm giải thuật trên.
- **Học tăng cường:** dữ liệu huấn luyện sẽ được thuật toán học xây dựng mô hình, mô hình này dùng để dự đoán và nhận phản hồi từ môi trường từ đó tự động cải thiện mô hình. Một số bài toán tiêu biểu cho hình thức học tăng cường là công nghệ xe tự lái, tự động chơi game,... Học tăng cường không có các cặp dữ liệu

vào/kết quả đúng, các hành động gần tối ưu cũng không được đánh giá đúng sai một cách tường minh.

### **2.1.2 Các ứng dụng của học máy**

Ứng dụng: Học máy có ứng dụng rộng khắp trong các ngành khoa học/sản xuất, đặc biệt những ngành cần phân tích khối lượng dữ liệu khổng lồ. Một số ứng dụng thường thấy

- Xử lý ngôn ngữ tự nhiên (Natural Language Processing): xử lý văn bản, giao tiếp người – máy, ...
- Nhận dạng (Pattern Recognition): nhận dạng tiếng nói, chữ viết tay, vân tay, thị giác máy (Computer Vision) ...
- Tìm kiếm (Search Engine)
- Chẩn đoán trong y tế: phân tích ảnh X-quang, các hệ chuyên gia chẩn đoán tự động.
- Tin sinh học: phân loại chuỗi gene, quá trình hình thành gene/protein
- Vật lý: phân tích ảnh thiên văn, tác động giữa các hạt ...
- Phát hiện gian lận tài chính (financial fraud): gian lận thẻ tín dụng Phân tích thị trường chứng khoán (stock market analysis)
- Chơi trò chơi: tự động chơi cờ, hành động của các nhân vật ảo

## **2.3 Tổng quan về hệ thống gợi ý**

### **2.3.1 Tổng quan**

Hệ thống gợi ý (Recommender systems hoặc Recommendation systems) là một dạng của hệ hỗ trợ ra quyết định, cung cấp giải pháp mang tính cá nhân hóa mà không phải trải qua quá trình tìm kiếm phức tạp. Hệ gợi ý học từ người dùng và gợi ý các sản phẩm tốt nhất trong số các sản phẩm phù hợp.

Hệ thống gợi ý sử dụng các tri thức về sản phẩm, các tri thức của chuyên gia hay tri thức khai phá học được từ hành vi con người dùng để đưa ra các gợi ý về sản phẩm mà họ thích trong hàng ngàn hàng vạn sản phẩm có trong hệ thống. Các website

thương mại điện tử, ví dụ như sách, phim, nhạc, báo...sử dụng hệ thống gợi ý để cung cấp các thông tin giúp cho người sử dụng quyết định sẽ lựa chọn sản phẩm nào. Các sản phẩm được gợi ý dựa trên số lượng sản phẩm đó đã được bán, dựa trên các thông tin cá nhân của người sử dụng, dựa trên sự phân tích hành vi mua hàng trước đó của người sử dụng để đưa ra các dự đoán về hành vi mua hàng trong tương lai của chính khách hàng đó. Các dạng gợi ý bao gồm: gợi ý các sản phẩm tới người tiêu dùng, các thông tin sản phẩm mang tính cá nhân hóa, tổng kết các ý kiến cộng đồng, và cung cấp các chia sẻ, các phê bình, đánh giá mang tính cộng đồng liên quan tới yêu cầu, mục đích của người sử dụng đó.

### 2.3.2 Các phương pháp trong hệ thống gợi ý

Phương pháp gợi ý	Dữ liệu cơ sở	Dữ liệu đầu ra	Tiến trình xử lý
Dựa theo lọc cộng tác	Các điểm số đánh giá của những người sử dụng trong U đối với các đối tượng trong I.	Các điểm số đánh giá của u cho các đối tượng trong I.	Nhận ra người sử dụng trong U tương tự với u ( về sở thích) và sau đó ngoại suy điểm số đánh giá của u cho i.
Dựa theo nội dung	Các đặc điểm của các đối tượng trong I.	Các điểm số đánh giá của u cho các đối tượng trong I.	Tạo ra một mô hình mô tả sở thích của người sử dụng u, sau đó sử dụng để đánh giá mức độ ưa thích của u với i.

Dựa trên cơ sở tri thức	Các đặc điểm của các đối tượng trong I. Các tri thức (hiểu biết) về sự phù hợp giữa các đối tượng với nhu cầu của người sử dụng.	Một sự mô tả nhu cầu và sở thích của người sử dụng u.	Suy luận sự phù hợp giữa I và nhu cầu của u.
-------------------------	--	---	--

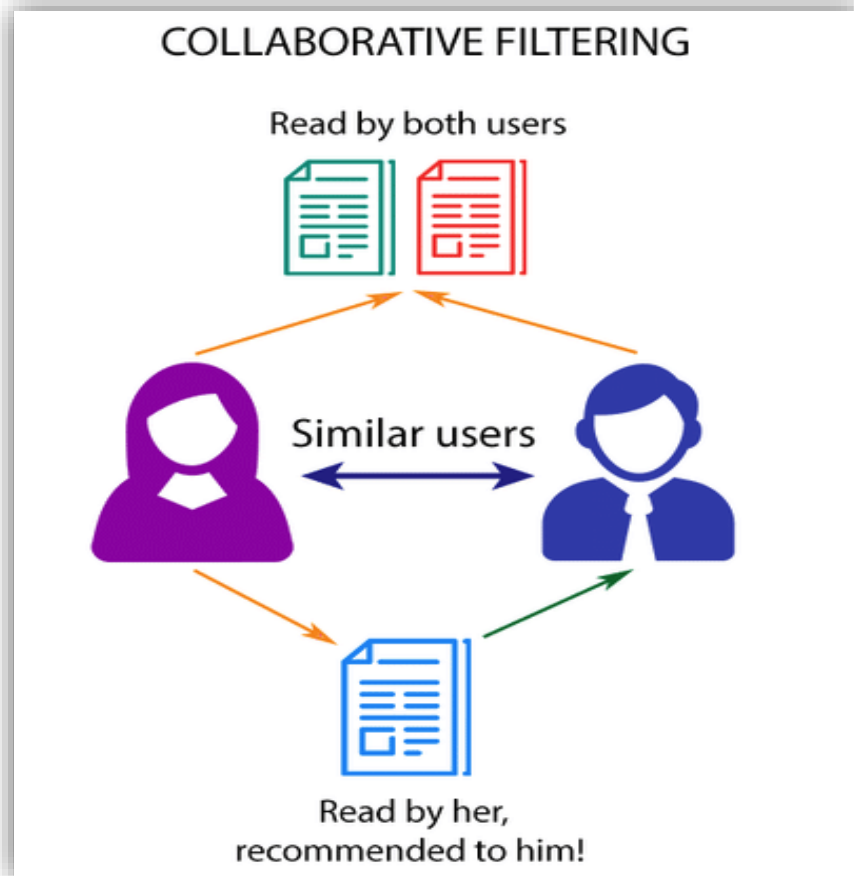
Bảng 1: các phương pháp hệ thống gợi ý

## 2.4 Hệ thống gợi ý lọc cộng tác

### 2.4.1 Tổng quan phương pháp lọc cộng tác

- Hệ thống gợi ý dựa theo lọc cộng tác (Collaborative recommendation systems): là phương pháp gợi ý được triển khai rộng rãi nhất và thành công nhất trong thực tế.
- Hệ thống theo lọc cộng tác phân tích và tổng hợp các điểm số đánh giá của các đối tượng, nhận ra sự tương đồng giữa những người sử dụng trên cơ sở các điểm số đánh giá của họ và tạo ra các gợi ý dựa trên sự so sánh này. Hồ sơ (profile) của người sử dụng điển hình trong hệ thống lọc cộng tác bao gồm một vector các đối tượng (item) và các điểm số đánh giá của chúng, với số chiều tăng lên liên tục khi người sử dụng tương tác với hệ thống theo thời gian.
- Một số hệ thống sử dụng phương pháp chiết khấu dựa trên thời gian (time-based discounting) để tính toán cho yếu tố “trượt” đối với sự quan tâm của người sử dụng. Trong một số trường hợp điểm số đánh giá (rating) có thể là nhị phân (thích/không thích) hoặc các giá trị số thực cho thấy mức độ ưu tiên.
- Thế mạnh lớn nhất của kỹ thuật gợi ý theo lọc cộng tác là chúng hoàn toàn độc lập với sự biểu diễn của các đối tượng đang được gợi ý, và do đó có thể làm việc tốt với các đối tượng phức tạp như âm thanh và phim. Schafer, Konstan & Riedl (1999)

gọi lọc cộng tác là “tương quan giữa người – với – người” (people-to-people correlation).



Hình 1 Sơ đồ phương pháp lọc cộng tác

#### 2.4.2 Ưu và nhược điểm

✓ **Ưu điểm:**

- Phương pháp này có khả năng dự đoán được sở thích và nhu cầu của người dùng để đưa ra gợi ý các sản phẩm phù hợp với từng khách hàng mà không cần hiểu sản phẩm.
- Gợi ý dựa trên trải nghiệm của người dùng tương tự khác nên có thể gợi ý được những sản phẩm mới phù hợp sở thích mới.

- Phương pháp này rất phù hợp với những hệ thống lớn có nhiều đánh giá từ phía người dùng.

Ngày nay, phương pháp này được sử dụng khá phổ biến trên các trang thương mại điện tử lớn như Amazon, Tiki, Youtube,... bởi tính đơn giản và một lượng dữ liệu sẵn có từ người dùng trên các website này.

✓ **Hạn chế:**

- Không thể gợi ý nếu khách hàng chưa có dữ liệu về lịch sử tương tác mặt hàng.
- Khi lượng sản phẩm lớn và số lượng khách hàng đánh giá không nhiều thì phương pháp này không hiệu quả.
- Phương pháp này cũng không thể gợi ý được các sản phẩm mới hoặc những sản phẩm chưa được ai đánh giá.
- Phương pháp này sẽ cho độ chính xác kém nếu như sở thích của người dùng thay đổi.

## **2.5 Các ứng dụng của hệ thống gợi ý**

### **2.5.1 Các hệ thống gợi ý**

Hệ thống gợi ý chủ yếu dùng trong các ứng dụng thương mại. Các hệ thống gợi ý được tận dụng trong nhiều lĩnh vực, chẳng hạn như trình tạo danh sách phát cho video và âm nhạc của một số ứng dụng như Netflix, YouTube và Spotify. Sau khi người xem một vài video hoặc nghe một số bài hát, hệ thống gợi ý sẽ nhận diện sở thích người dùng và mức độ tương tác với video/bài hát để tạo ra một danh sách gợi ý theo chủ đề và mức độ liên quan mà người dùng có thể ưa thích. Hệ thống gợi ý còn áp dụng cho các dịch vụ gợi ý sản phẩm ở Amazon, các gợi ý nội dung cho các nền tảng xã hội như Facebook hay Twitter.



### **2.5.2 Hệ thống gợi ý dựa trên phương pháp lọc cộng tác**

Không giống như mô hình truyền thống của phương tiện truyền thông chính thống, trong đó có ít biên tập viên đặt ra nguyên tắc, phương tiện truyền thông xã hội được lọc cộng tác có thể có số lượng biên tập viên rất lớn và nội dung được cải thiện khi số lượng người tham gia tăng lên. Các dịch vụ như Reddit , YouTube và Last.fm là những ví dụ điển hình của phương tiện truyền thông dựa trên lọc cộng tác.

Một kịch bản của ứng dụng lọc cộng tác là giới thiệu thông tin thú vị hoặc phổ biến theo đánh giá của cộng đồng. Ví dụ điển hình, các câu chuyện xuất hiện trên trang đầu của Reddit khi chúng được cộng đồng "bình chọn" (đánh giá tích cực). Khi cộng đồng trở nên lớn hơn và đa dạng hơn, các câu chuyện được quảng bá có thể phản ánh tốt hơn mức độ quan tâm trung bình của các thành viên trong cộng đồng.

Wikipedia là một ứng dụng khác của lọc cộng tác. Các tình nguyện viên đóng góp vào bộ bách khoa toàn thư bằng cách lọc ra các dữ kiện từ sự giả dối.

## 2.6 Framework Django



Hình 2 Django framework

Django là một trong số những web framework bậc cao miễn phí, là mã nguồn mở được tạo ra bởi ngôn ngữ Python dựa trên mô hình mô hình MTV (gồm Model-Template-Views). Hiện framework này được phát triển, quản lý bởi Django Software Foundation. Django ra đời với mục tiêu hỗ trợ thiết kế các website phức tạp dựa trên những CSDL có sẵn. Nó hoạt động dựa theo nguyên lý ‘cắm’ các thành phần và tái sử dụng để tạo nên các website với ít code, ít khớp nối, có khả năng phát triển và không bị trùng lặp.

### 2.6.1 Ưu điểm của Django

Lợi thế hàng đầu của Django là khả năng thiết kế, tạo lập website và các ứng dụng nhanh chóng đến bất ngờ. Ngoài ra, dưới đây Bizfly liệt kê những điểm cộng khiến framework này trở nên nổi bật và được lập trình viên sử dụng rộng rãi.

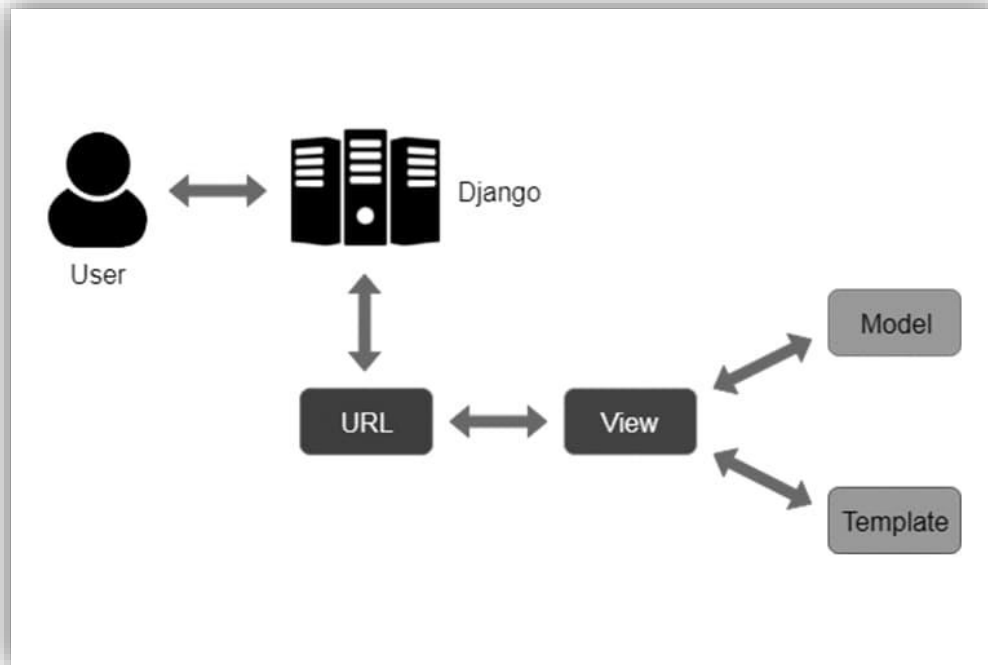
- **Bảo mật tốt:** Làm việc với Django, các lập trình viên gần như không có không gian để phạm bất cứ sai lầm về an ninh nào. Nó giúp các developer tránh được tất cả các lỗi thường gặp như nhấp chuột, kịch bản chéo trang, SQL tiêm, giả mạo yêu cầu,... Nhờ đó, sản phẩm được tạo bởi framework này có khả năng bảo mật cực tốt.
- **Mở rộng thỏa thích:** Django có sẵn tính năng mở rộng nhằm hỗ trợ các lập trình viên quản lý lưu lượng người truy cập, thích hợp với các trang có traffic lớn.
- **Dễ sử dụng:** Django được tạo ra bởi ngôn ngữ lập trình **Python** và mô hình MVC nên rất dễ ứng dụng trong các dự án. Đa ngôn ngữ và được hỗ trợ Multi-Site
- **Dễ học:** Có nhiều tài liệu hỗ trợ việc học Django, bao gồm cả tài liệu miễn phí trên mạng và sách in. Cộng đồng sử dụng Django hiện nay đang phát triển mạnh mẽ, newbie có thể tìm kiếm sự giúp đỡ trên các trang facebook, diễn đàn, blog,...

### 2.6.2 Nhược điểm của Django

Bên cạnh những ưu điểm kể trên, Django cũng tồn tại một số nhược điểm mà bạn cần cân nhắc trước khi sử dụng chúng. Cụ thể sau đây:

- Có thể xảy ra một vài vấn đề khi phát triển các ứng dụng/website quy mô nhỏ
- Định tuyến tương đối khó
- Không đưa ra các cảnh báo khi xuất hiện lỗi trong mẫu.

### 2.6.3 MVT Pattern của Django



Hình 3 Mô hình MVT của Django

Django sử dụng mô hình MVT (Model-View-Template) thay vì sử dụng mô hình MVC (Model-View-Controller).

Mô hình MVT được sử dụng trong khi tạo một ứng dụng với Tương tác người dùng.

Mô hình này thì bao gồm code HTML với Django Template Language (DTL).

Controller là mã được viết để kiểm soát sự tương tác giữa Model và View và Django dễ dàng chăm sóc nó.

Bất cứ khi nào người dùng người request, nó xử lý request của người dùng đó bằng Model, View và Template.

Nó hoạt động như một Controller để kiểm tra xem nó có khả dụng hay không bằng cách ánh xạ URL và nếu URL ánh xạ thành công thì View sẽ bắt đầu tương tác với Model và gửi lại Template cho người dùng dưới dạng response.

#### 2.6.4 Các thành phần cơ bản của ứng dụng Django

Dự án Django tạo ra một tập hợp các cài đặt bao gồm cấu hình cơ sở dữ liệu và các tùy chọn cụ thể cũng như các cài đặt cụ thể của ứng dụng mặc định trong dự án. Cấu trúc thư mục của django project như sau:

- `mysite/`
- `manage.py`
- `mysite/`
- `__init__.py`
- `settings.py`
- `urls.py`
- `wsgi.py`

Trong đó

- `manage.py`: Cho phép bạn tương tác với dự án Django theo các cách khác nhau
- `__init__.py`: Nói với trình thông dịch python là thư mục nên được coi là một python package. Tập tin này chủ yếu là trống.
- `settings.py`: Tập tin cấu hình
- `urls.py`: Bao gồm tất cả khai báo URL cho dự án Django và mục lục của trang web Django.
- `wsgi.py`: Đây là lối vào cho các máy chủ web tương thích WSGI để phục vụ các dự án của bạn và deploy với WSGI.

### **2.6.5 Lí do chọn Django trong lập trình web**

Django được xây dựng để giúp phát triển nhanh chóng với thiết kế sạch sẽ và thiết thực.

Khả năng dễ đọc của Python, đơn giản, đầy đủ của Django cho phép bạn tập trung vào các vấn đề phức tạp, logic nghiệp vụ hơn là mất nhiều thời gian cho các rắc rối đã được người khác giải quyết.

Và Django có một mô hình xác thực người dùng rất tốt với khả năng cấu hình người dùng. Điều này đã làm cho nó trở thành lựa chọn hàng đầu khi trang web, ứng dụng cần ưu tiên về bảo mật.

Nó sử dụng một loạt các thành phần Python là các thực thể riêng biệt không phụ thuộc vào nhau.

Nhận thấy lợi ích và ưu điểm của Django, nhóm quyết định sử dụng framework này để xây dựng website cho đề tài của mình.

## CHƯƠNG 3: MÔ HÌNH ĐỀ XUẤT

### 3.1 Phương pháp lọc cộng tác

Lọc cộng tác thực hiện tư vấn(gợi ý) các sản phẩm, dịch vụ, nội dung cho người dùng nào đó dựa trên mối quan tâm sở thích của những người dùng tương tự đối với các sản phẩm dịch vụ nội dung đó. Lọc cộng tác được xem là một trong ba cách tiếp cận chính trong xây dựng hệ tư vấn

Có nhiều kỹ thuật lọc cộng tác và được chia thành hai dạng chính:

- Memory-based: lọc cộng tác dựa trên việc ghi nhớ toàn bộ dữ liệu.
- Model-based: Lọc cộng tác dựa trên các mô hình phân lớp, dự đoán

#### 3.1.1 Lọc cộng tác dựa trên bộ nhớ

Phương pháp dựa trên bộ nhớ sử dụng dữ liệu xếp hạng của người dùng để tính toán mức độ giống nhau giữa người dùng hoặc các mục. Ví dụ điển hình của cách tiếp cận này là CF dựa trên vùng lân cận và các đề xuất top-N dựa trên mục / dựa trên người dùng. Ví dụ: trong phương pháp tiếp cận dựa trên người dùng, giá trị xếp hạng mà người dùng  $u$  cung cấp cho mặt hàng  $i$  được tính là tổng hợp của một số xếp hạng của người dùng tương tự đối với mặt hàng:

$$r_{u,i} = \text{aggr}_{u' \in U} r_{u',i}$$

trong đó  $U$  biểu thị tập hợp  $N$  người dùng hàng đầu giống nhất với người dùng  $u$  đã xếp hạng mục  $i$ . Một số ví dụ về hàm tổng hợp bao gồm:

$$r_{u,i} = \frac{1}{N} \sum_{u' \in U} r_{u',i}$$
$$r_{u,i} = k \sum_{u' \in U} \text{simil}(u, u') r_{u',i}$$

trong đó  $k$  là hệ số chuẩn hóa được định nghĩa là  $k = 1 / \sum_{u' \in U} |\text{simil}(u, u')|$ , và

$$r_{u,i} = \bar{r}_u + k \sum_{u' \in U} \text{simil}(u, u') (r_{u',i} - \bar{r}_{u'})$$

ở đây  $\bar{r}_u$  là xếp hạng trung bình của người dùng  $u$  cho tất cả các mục được xếp hạng bởi  $u$ .

Thuật toán dựa trên vùng lân cận tính toán mức độ giống nhau giữa hai người dùng hoặc các mục và đưa ra dự đoán cho người dùng bằng cách lấy trung bình có trọng số của tất cả các xếp hạng. Tính toán sự giống nhau giữa các mặt hàng hoặc người dùng là một phần quan trọng của phương pháp này. Nhiều thước đo, chẳng hạn như tương quan Pearson và độ tương tự dựa trên vector cosin được sử dụng cho việc này.

Sự giống nhau về tương quan Pearson của hai người dùng  $x, y$  được định nghĩa là

$$\text{simil}(x, y) = \frac{\sum_{i \in I_{xy}} (r_{x,i} - \bar{r}_x)(r_{y,i} - \bar{r}_y)}{\sqrt{\sum_{i \in I_{xy}} (r_{x,i} - \bar{r}_x)^2} \sqrt{\sum_{i \in I_{xy}} (r_{y,i} - \bar{r}_y)^2}}$$

Thuật toán đề xuất top-N dựa trên người dùng sử dụng mô hình vector dựa trên sự tương tự để xác định  $k$  người dùng tương tự nhất với người dùng đang hoạt động. Sau khi tìm thấy  $k$  người dùng tương tự nhất, ma trận mục người dùng tương ứng của họ sẽ được tổng hợp lại để xác định nhóm mục được đề xuất. Một phương pháp phổ biến để tìm những người dùng tương tự là bấm nhảy cảm với địa phương, thực hiện cơ chế láng giềng gần nhất trong thời gian tuyến tính.

Những ưu điểm của cách tiếp cận này bao gồm: khả năng giải thích được của các kết quả, đây là một khía cạnh quan trọng của hệ thống khuyến nghị; dễ dàng tạo và sử dụng; dễ dàng tạo thuận lợi cho dữ liệu mới; nội dung độc lập của các mục được khuyến nghị; mở rộng quy mô tốt với các mục đồng đánh giá.



Cũng có một số nhược điểm với cách tiếp cận này. Hiệu suất của nó giảm khi dữ liệu trở nên thừa thớt, điều này xảy ra thường xuyên với các mục liên quan đến web. Điều này cản trở khả năng mở rộng của cách tiếp cận này và tạo ra các vấn đề với bộ dữ liệu lớn. Mặc dù nó có thể xử lý hiệu quả người dùng mới vì nó dựa vào cấu trúc dữ liệu, việc thêm các mục mới trở nên phức tạp hơn vì cách biểu diễn đó thường dựa vào một không gian vector cụ thể. Thêm mục mới yêu cầu phải bao gồm mục mới và chèn lại tất cả các phần tử trong cấu trúc.

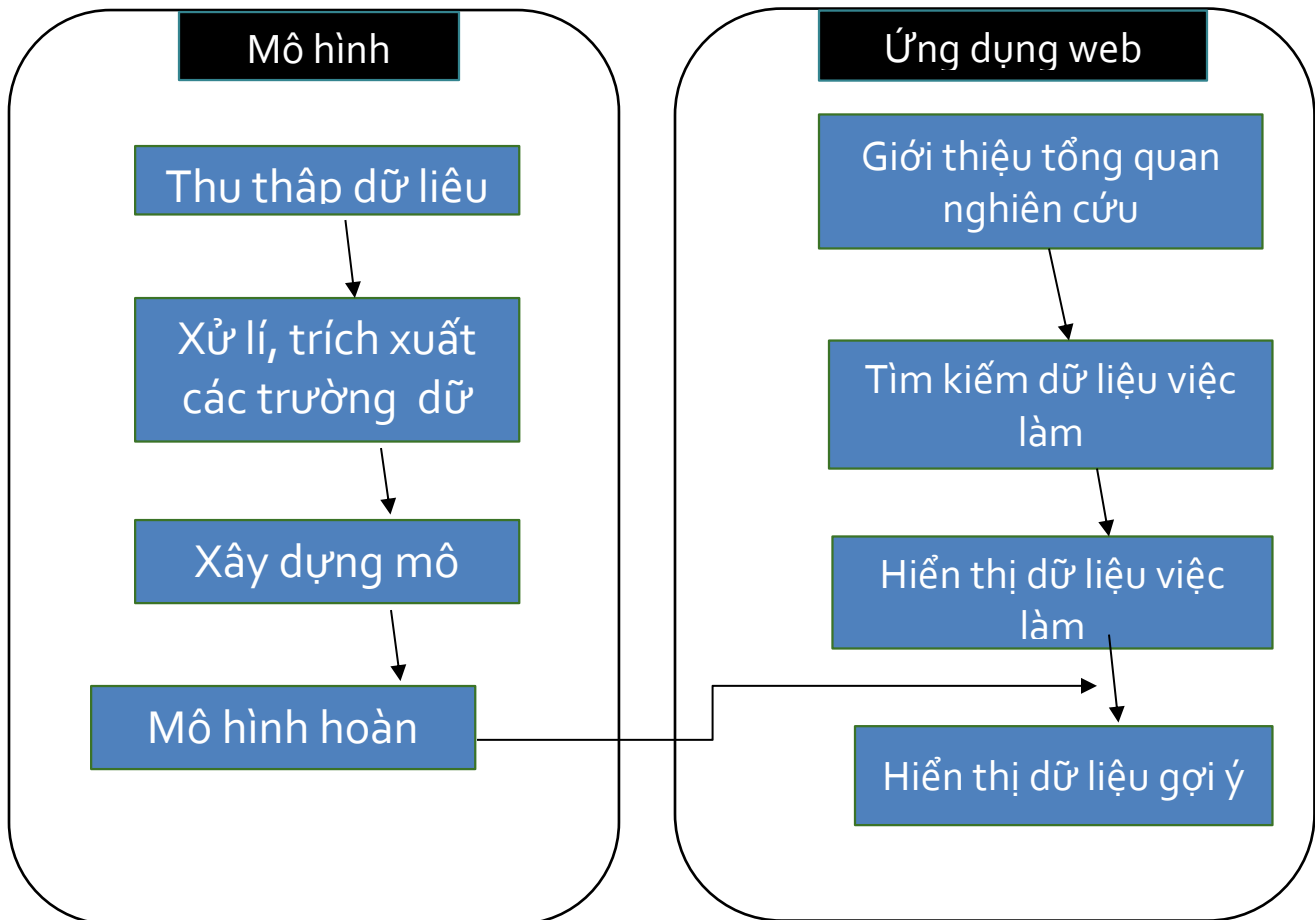
### **3.1.2 Lọc cộng tác dựa trên mô hình**

Trong cách tiếp cận này, các mô hình được phát triển bằng cách sử dụng các thuật toán khai thác dữ liệu, máy học khác nhau để dự đoán xếp hạng của người dùng đối với các mục chưa được xếp hạng. Có rất nhiều thuật toán CF dựa trên mô hình. Mạng Bayes, mô hình phân nhóm, mô hình ngữ nghĩa tiềm ẩn như phân hủy ít giá trị, phân tích ngữ nghĩa tiềm ẩn xác suất, yếu tố nhiều nhân giống, phân bố Dirichlet tiềm ẩn và quá trình ra quyết định Markov mô hình dựa.

Thông qua cách tiếp cận này, các phương pháp giảm số chiều chủ yếu được sử dụng như một kỹ thuật bổ sung để cải thiện tính mạnh mẽ và độ chính xác của phương pháp tiếp cận dựa trên bộ nhớ. Theo nghĩa này, các phương pháp như phân tích giá trị đơn lẻ, phân tích thành phần chính, được gọi là mô hình yếu tố tiềm ẩn, nén ma trận mục người dùng thành một biểu diễn chiều thấp về các yếu tố tiềm ẩn. Một lợi thế của việc sử dụng cách tiếp cận này là thay vì có ma trận chiều cao chứa nhiều giá trị bị thiếu, chúng ta sẽ xử lý ma trận nhỏ hơn nhiều trong không gian chiều thấp hơn. Bản trình bày rút gọn có thể được sử dụng cho các thuật toán lân cận dựa trên người dùng hoặc dựa trên mục đã được trình bày trong phần trước. Có một số lợi thế với mô hình này. Nó xử lý độ thừa thớt của ma trận gốc tốt hơn so với ma trận dựa trên bộ nhớ. Ngoài ra, so sánh độ tương tự trên ma trận kết quả có thể mở rộng hơn nhiều, đặc biệt là trong việc xử lý các bộ dữ liệu thừa thớt lớn.

### 3.2 Mô hình tổng quát bài toán

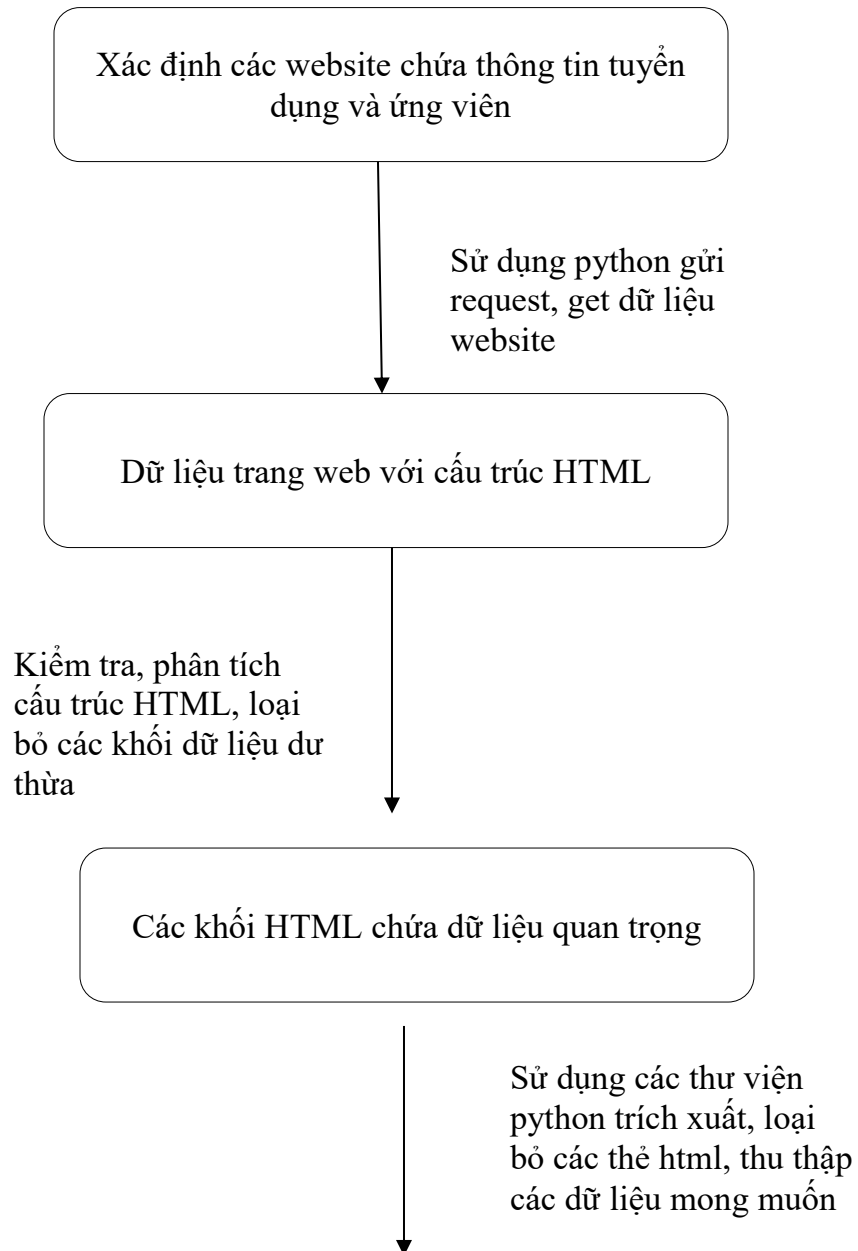
Trong đề tài nghiên cứu này, nhóm sử dụng phương pháp lọc cộng tác dựa trên mô hình để giải quyết bài toán gợi ý các công việc.

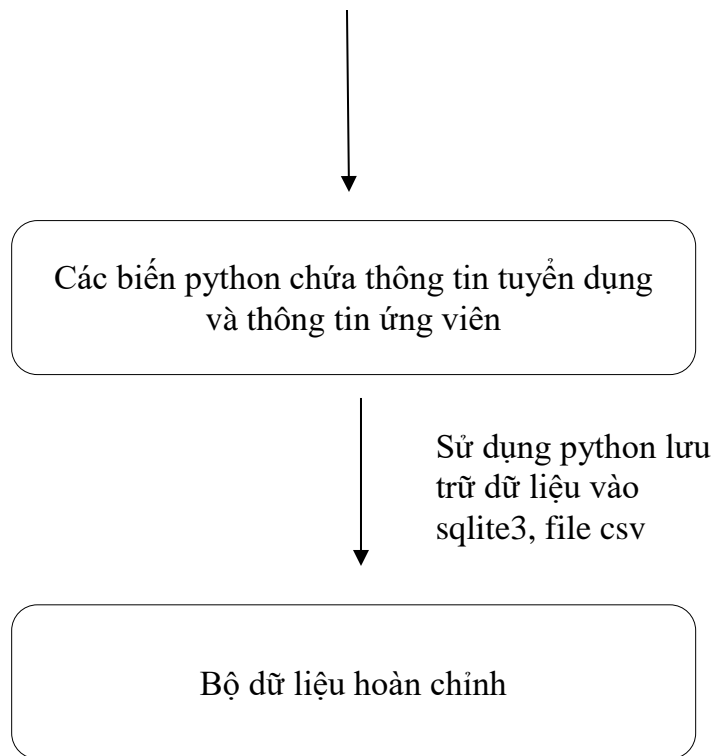


Khi người dùng truy cập hệ thống -> người dùng xem 1 bài viết bất kì trên hệ thống -> hệ thống đưa ra các bài viết chứa các công việc liên quan tới bài viết hiện tại -> người dùng có thể click xem và đi tới trang web bên thứ ba của nhà tuyển dụng.

### 3.2.1 Mô hình tổng quát thu thập dữ liệu

Sau đây là mô hình tổng quát của việc thu thập dữ liệu:





Từ bộ dữ liệu chứa thông tin tuyển dụng và thông tin ứng viên, tiến hành đánh giá, chọn lọc các trường dữ liệu phục vụ cho việc xây dựng mô hình lọc cộng tác.

Thực chất, vấn đề của hệ gợi ý là xác định ánh xạ  $(\mathbf{u}, \mathbf{i}) \rightarrow \mathbf{R}$ , trong đó  $\mathbf{u}$  là biểu diễn cho 1 người dùng,  $\mathbf{i}$  biểu diễn cho 1 sản phẩm và  $\mathbf{R}$  là đánh giá của  $\mathbf{u}$  lên  $\mathbf{i}$ . Sau đó, các đánh giá của người dùng  $\mathbf{u}$  lên tất cả các sản phẩm  $\mathbf{i}$  tương ứng sẽ được sắp xếp, và lấy  $N$  sản phẩm có đánh giá cao nhất để đưa ra gợi ý cho người dùng  $\mathbf{u}$ .

Tuy nhiên, khái niệm ‘**đánh giá**’ ở đây là khá trừu tượng, có thể được đo lường bằng hành động của người dùng như mua sản phẩm, click chuột vào sản phẩm, đánh giá số sao hoặc click vào “không hiển thị lại”, ...

Cụ thể đối với bài toán gợi ý công việc của đề tài này, do bộ dữ liệu thu thập chưa có đánh giá của người dùng, có thể hiểu là qua 1 đối tượng người dùng, chưa có 1 trường dữ liệu nào thể hiện cho việc “người dùng này phù hợp với công việc kia như thế nào”, chính vì vậy, đối với một ứng viên và một công việc tuyển dụng được thể hiện bằng

vector  $(\mathbf{u}, \mathbf{i})$ , chúng ta cần khởi tạo 1 trường dữ liệu  $\mathbf{R}$  để đo lường mức độ phù hợp giữa ứng viên  $\mathbf{u}$  và công việc  $\mathbf{i}$ .

### 3.2.2 Tổng quát xây dựng mô hình lọc cộng tác

Lọc cộng tác (Collaborative Filtering) là một phương pháp gợi ý sản phẩm với ý tưởng chính dựa trên các hành vi của các users khác (collaborative) cùng trên một item để suy ra mức độ quan tâm (filtering) của một user lên sản phẩm. Việc suy ra này được thực hiện dựa trên *similarity\_maxtrix* đo độ giống nhau giữa các users.

Công việc quan trọng nhất trong việc xây dựng mô hình là phải xác định sự giống nhau (similarity) giữa 2 ứng viên hoặc 2 công việc.

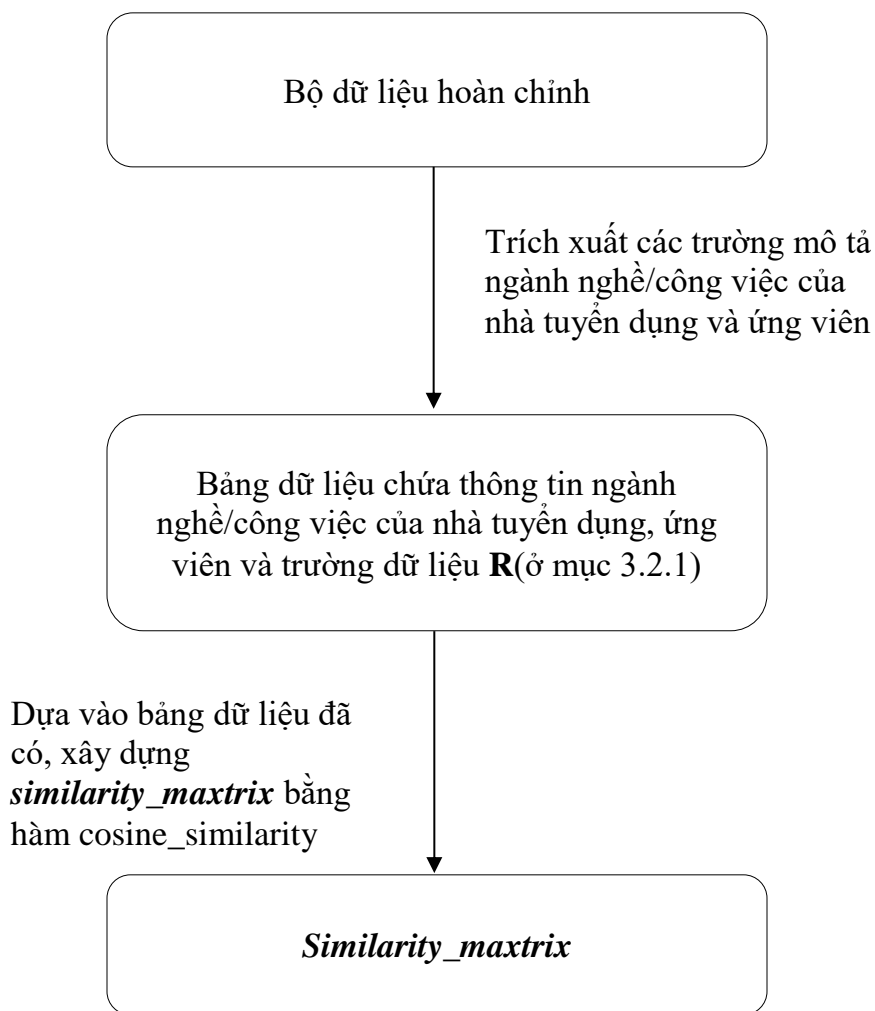
Công thức phổ biến và được sử dụng nhiều nhất trong việc xác định *similarity* giữa 2 vector  $\mathbf{u}_1, \mathbf{u}_2$  là công thức Cosin. Nó được tính như sau:

$$\text{cosine\_similarity}(\mathbf{u}_1, \mathbf{u}_2) = \cos(\mathbf{u}_1, \mathbf{u}_2) = \frac{\mathbf{u}_1^T \mathbf{u}_2}{\|\mathbf{u}_1\|_2 \cdot \|\mathbf{u}_2\|_2}$$

Sau khi tính được *similarity* giữa các công việc hoặc ứng viên, chúng ta sẽ xây dựng được ma trận *similarity\_maxtrix* có giá trị  $[0:1]$

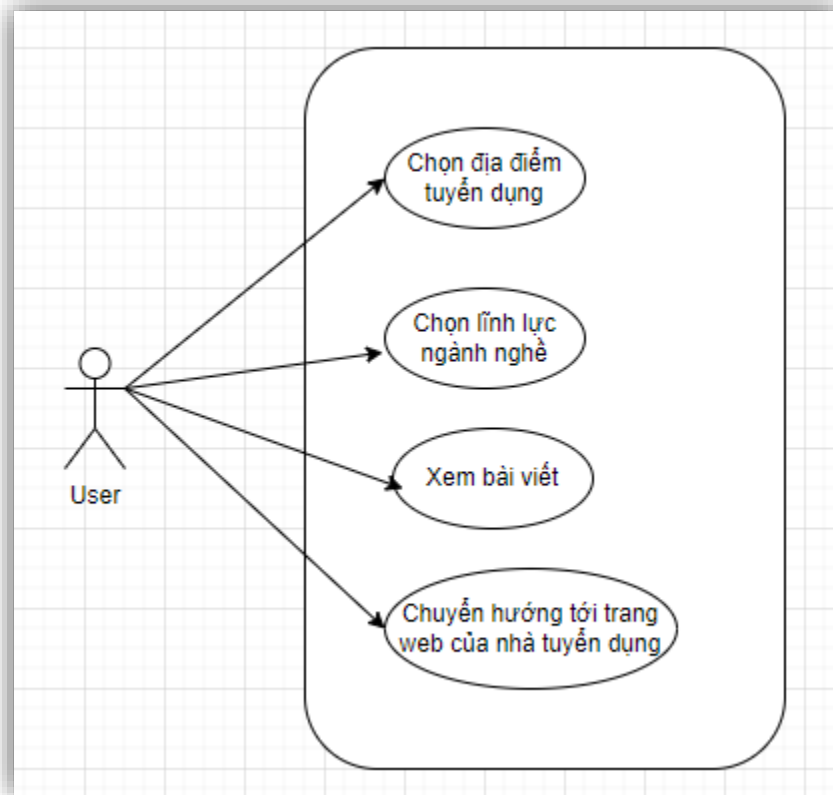
Dựa vào *similarity\_maxtrix* để xem xét các ứng viên hoặc công việc nào có độ tương quan cao để đưa ra gợi ý. Giá trị của *similarity* càng cao thì mức độ phù hợp càng cao và ngược lại.

Sau đây là mô hình tổng quan của việc xây dựng *similarity\_maxtrix*:



Sau khi xây dựng xong *similarity\_maxtrix*, dựa vào nó để đưa ra các công việc có mức độ phù hợp với nhau để đưa ra gợi ý cho hệ thống.

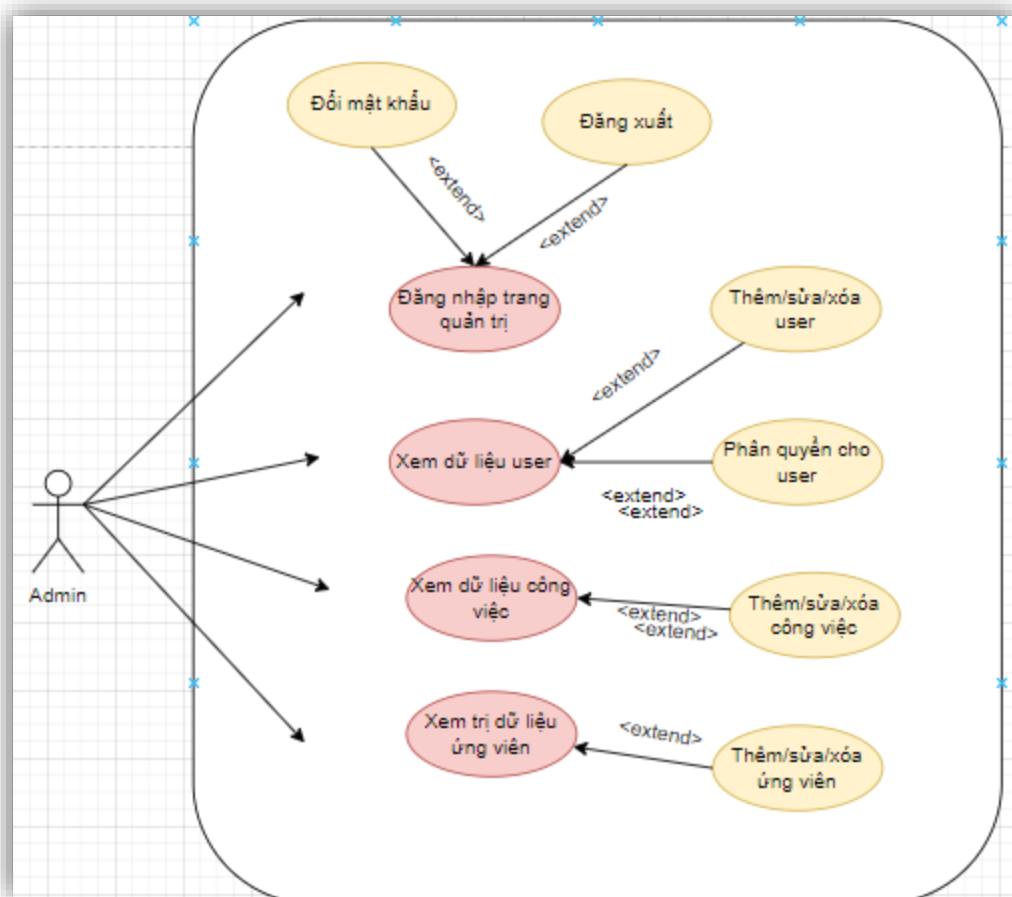
### 3.2 Use Case của hệ thống gợi ý



Hình 4 Use case của user

Khi người dùng(ứng viên) truy cập vào hệ thống gợi ý, user sẽ có các usecase cơ bản như sau:

- Chọn địa điểm tuyển dụng
- Chọn lĩnh vực ngành nghề mong muốn
- Click xem bài viết tuyển dụng
- Click vào button để chuyển hướng tới trang web của nhà tuyển dụng để xem chi tiết các yêu cầu về vị trí đang ứng tuyển



Hình 5 Use case của Admin

Admin muốn vào trang quản trị phải đăng nhập được vào hệ thống.

Các chức năng cơ bản của hệ thống đối với admin như sau:

- Quản trị các user đăng nhập vào hệ thống: thêm/sửa/xóa và phân quyền user
- Quản trị dữ liệu công việc của nhà tuyển dụng: Xem, thêm/sửa/xóa dữ liệu công việc của nhà tuyển dụng
- Quản trị dữ liệu ứng viên: Xem, thêm/sửa/xóa dữ liệu ứng viên.



## CHƯƠNG 4: THỰC NGHIỆM

### 4.1 Các công nghệ và thư viện sử dụng

Để giải quyết bài toán xây dựng hệ tư vấn việc làm, nghiên cứu này sử dụng các công nghệ sau:

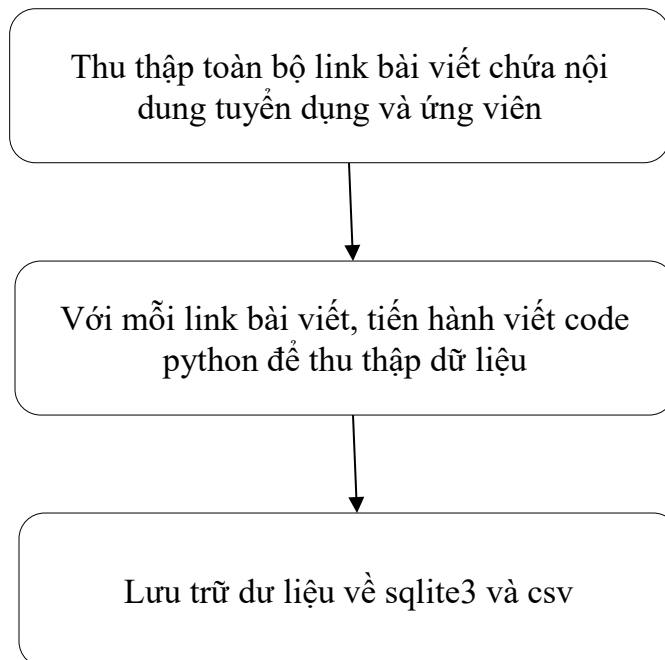
- Các nền tảng và thư viện: Python, Django Framework, Bootstrap
- Công cụ sử dụng: Google Colab, Pycharm, SQLiteStudio

### 4.2 Dữ liệu

#### 4.2.2 Thu thập dữ liệu

Để tiến hành thực nghiệm xây dựng hệ tư vấn việc làm cho đề tài, bộ dữ liệu được sử dụng sẽ là thông tin nhà tuyển dụng và thông tin ứng viên, được thu thập trên các trang web tuyển dụng **timviec.com.vn** và **careerbuilder.vn**

Sau đây là các bước thu thập dữ liệu từ 2 trang web trên:



- Hàm thu thập tất cả các link bài viết chứa nội dung tuyển dụng:

```
def get_links_job(url):
    response = requests.get(url)
    soup = BeautifulSoup(response.content, "html.parser")
    a = soup.findAll('a', class_="title-job")
    links = [link.attrs['href'] for link in a]
    #add database
    conn = sqlite3.connect("data/DBTimviec.db")
    query = """
    INSERT INTO LINKS_JOB(LINK) VALUES (?)
    """
    for i in links:
        link = "https://timviecnhanh.com" +
str(i).replace("?svs=max_box", "")
        conn.execute(query, (link,))
        conn.commit()
        print("added to database ", link)
```

Tất cả các link tuyển dụng đều nằm trong thẻ ‘a’ có class là “title-job”, đường link nằm trong thuộc tính ‘attrs’. Thực hiện vòng lặp và lưu tất cả chúng vào database

- Hàm thu thập nội dung tuyển dụng với mỗi một link bài viết:

```
def get_data(conn,url):
    response = requests.get(url)
    soup = BeautifulSoup(response.content, "html.parser")

    title = soup.find("span", class_="title").text
    address = soup.find("span", class_="jsx-1425348829 d-none d-md-block").text.replace("Địa chỉ: ", "").strip()

    ul = soup.find("ul", class_="jsx-1425348829 no-style p-0")
    li = ul.findAll('li')

    salary = str(li[0].text).replace("- Mức lương:", "").strip()
    ex = str(li[1].text).replace("- Kinh nghiệm:", "").strip()
    level = str(li[2].text).replace("- Trình độ:", "").strip()
    location = str(li[3].text).replace("- Tỉnh/Thành phố:",
    "").strip()
    job_name = str(li[4].text).replace("- Ngành nghề:",
    "").strip()

    try:
        query = """
        INSERT INTO JOBS_DATA
```

```

(TITLE, LINK, ADDRESS, SALARY, EXPERIENCE, LEVEL, LOCATION, JOB_NAME)
VALUES (?, ?, ?, ?, ?, ?, ?, ?)
"""
    conn.execute(query, (title, url, address, salary, ex,
level, location, job_name)) # thêm dữ liệu crawl vào db
    conn.commit()
    print("Added {} to database".format(title))
except Exception as erro:
    print("Có lỗi: ", + erro)

```

- Hàm thu thập dữ liệu ứng viên:

```

- def get_ung_vien(conn, url):
    try:
        response = requests.get(url)
        soup = BeautifulSoup(response.content, "html.parser")
        # lay ten ung vien
        ung_vien_element = soup.findAll('p', class_="fw-400 mb-0
fs-14")
        # lay ten cong viec ung tuyen
        job_name_element = soup.findAll('span', class_='job-
name')
        # lat kinh nghiem
        kinh_nghiem_element = soup.findAll('span', class_="new-
2021-item-candidate-exp")
        # lay dia diem lam viec
        locations = soup.findAll('span', class_="mr-15")
        query = """
INSERT INTO EMPLOYER (NAME, JOB_NAME, LOCATION, EX)
VALUES (?, ?, ?, ?)
"""
        for i in range(len(ung_vien_element)):

            conn.execute(query, (str(ung_vien_element[i].text).replace("Đang
tìm việc", "").strip(),

str(job_name_element[i].text).strip(),

str(locations[i].text).replace("Địa điểm:", "").strip(),

str(kinh_nghiem_element[i].text).replace("Kinh
nghiem:", "").strip()))
            conn.commit()
            print("Đã thêm ", " . .
.", str(ung_vien_element[i].text).replace("Đang tìm
việc", "").strip())

    except Exception as ex:
        print("Erro: ", ex)
        pass

```

Sau khi hoàn thành thu thập dữ liệu, các trường dữ liệu được lưu lại trong sqlite3 và được export ra file csv như sau:

- **Employer.csv:** chứa thông tin về ứng viên. Mỗi hàng của tệp này mô tả một ứng viên. Các trường dữ liệu được định nghĩa và mô tả như sau:

STT	Tên thuộc tính	Kiểu dữ liệu	Giải thích
1	ID	Integer	Id của người tìm việc
2	NAME	Text	Họ tên của người tìm việc
3	JOB_NAME	Text	Loại hình công việc mong muốn của người tìm việc
4	LOCATION	Text	Địa chỉ/Tỉnh/Thành Phố hiện tại của người tìm việc
5	EX	Text	Kinh nghiệm làm việc của người tìm việc

Bảng 2: Mô tả dữ liệu người tìm việc(employer.CSV)

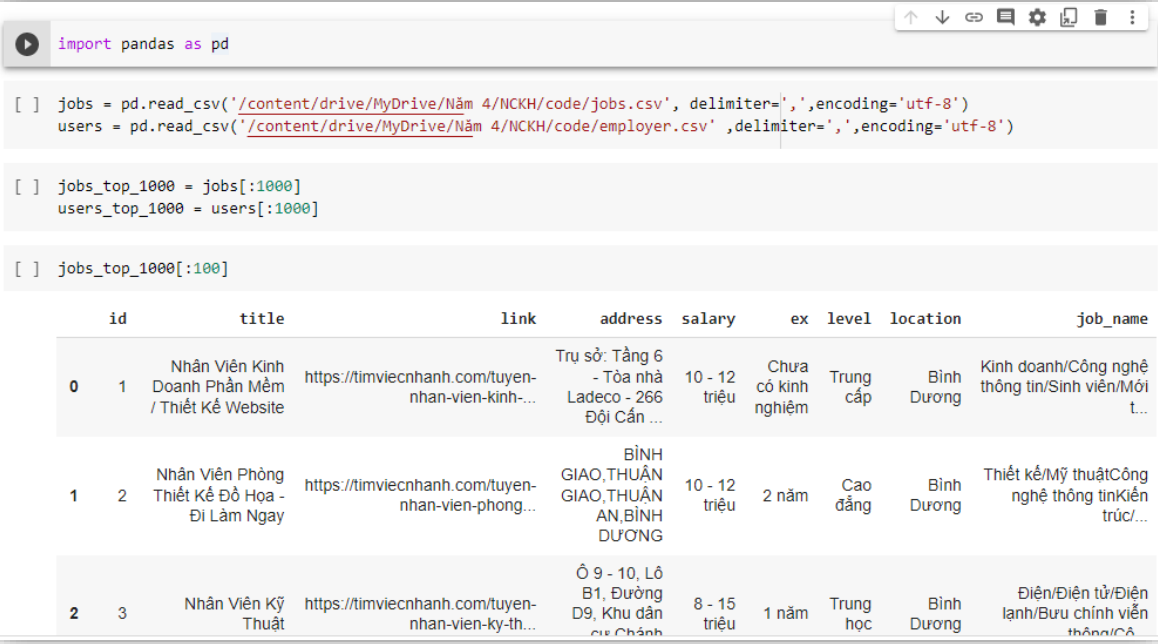
- **Jobs.csv:** chứa thông tin về các tin tuyển dụng. Mỗi hàng của tệp này mô tả một công việc. Các trường dữ liệu được định nghĩa và mô tả như sau:

STT	Tên thuộc tính	Kiểu dữ liệu	Giải thích
1	ID	Integer	ID của nhà tuyển dụng
2	TITLE	Text	Tiêu đề bài post tuyển dụng/tiêu đề công việc tuyển dụng
3	LINK	Text	Đường dẫn liên kết tới bài post tuyển dụng trên website
4	ADDRESS	Text	Địa chỉ làm việc của nhà tuyển dụng

5	<b>SALARY</b>	Text	Mức lương offer của nhà tuyển dụng
6	<b>EXPERIENCE</b>	Text	Kinh nghiệm yêu cầu của nhà tuyển dụng
7	<b>LEVEL</b>	Text	Yêu cầu bằng cấp của nhà tuyển dụng
8	<b>LOCATION</b>	Text	Tỉnh/Thành Phố/Khu vực tuyển dụng của nhà tuyển dụng
9	<b>JOB_NAME</b>	Text	Các lĩnh vực ngành nghề của công việc đang tuyển dụng

Bảng 3: Mô tả dữ liệu thông tin tuyển dụng(jobs.csv)

## 4.2.2 Xem và xử lý dữ liệu



```
import pandas as pd

[ ] jobs = pd.read_csv('/content/drive/MyDrive/Năm 4/NCKH/code/jobs.csv', delimiter=',', encoding='utf-8')
    users = pd.read_csv('/content/drive/MyDrive/Năm 4/NCKH/code/employer.csv', delimiter=',', encoding='utf-8')

[ ] jobs_top_1000 = jobs[:1000]
    users_top_1000 = users[:1000]

[ ] jobs_top_1000[:100]
```

	id	title	link	address	salary	ex	level	location	job_name
0	1	Nhân Viên Kinh Doanh Phần Mềm / Thiết Kế Website	<a href="https://timviecnhanh.com/tuyen-nhan-vien-kinh-...">https://timviecnhanh.com/tuyen-nhan-vien-kinh-...</a>	Trụ sở: Tầng 6 - Tòa nhà Ladeco - 266 Đội Cấn ...	10 - 12 triệu	Chưa có kinh nghiệm	Trung cấp	Bình Dương	Kinh doanh/Công nghệ thông tin/Sinh viên/Mới t...
1	2	Nhân Viên Phòng Thiết Kế Đồ Họa - Đi Làm Ngay	<a href="https://timviecnhanh.com/tuyen-nhan-vien-phong-...">https://timviecnhanh.com/tuyen-nhan-vien-phong-...</a>	BÌNH GIAO, THUẬN GIAO, THUẬN AN, BÌNH DƯƠNG	10 - 12 triệu	2 năm	Cao đẳng	Bình Dương	Thiết kế/Mỹ thuật/Công nghệ thông tin/Kiến trúc/...
2	3	Nhân Viên Kỹ Thuật	<a href="https://timviecnhanh.com/tuyen-nhan-vien-ky-th-...">https://timviecnhanh.com/tuyen-nhan-vien-ky-th-...</a>	Ô 9 - 10, Lô B1, Đường D9, Khu dân cư Chánh	8 - 15 triệu	1 năm	Trung học	Bình Dương	Điện/Điện tử/Điện lạnh/Bưu chính viễn thông/Cá...

Hình 6 Thông tin dữ liệu thông tin tuyển dụng

Để từng bước xây dựng được hệ thống gợi ý, đầu tiên chúng ta phải khảo sát dữ liệu để xem tính chất và đặc trưng của dữ liệu như thế nào. Load dữ liệu có được lên, sau đó là giai đoạn xử lý dữ liệu:

users\_top\_1000[:100]

	id	name	job_name	location	ex
0	1	Nguyen Minh Hieu	Designer	Hồ Chí Minh	Chưa có
1	2	Vũ Chính Trực	Thực Tập Sinh Thiết Kế Website	Hồ Chí Minh	Chưa có
2	3	Nguyễn Trung Vĩnh	Internship Ux/ui Designer	Hồ Chí Minh	Chưa có
3	4	Dương Thanh Trường	Sinh Viên Thực Tập Mobile, Front End Web	Hồ Chí Minh	1 năm
4	5	Huỳnh Tấn Phúc	Thực Tập Sinh Front End	Hồ Chí Minh	Chưa có
5	6	Nguyễn Hoàng Lâm	Front end dev	Bình Dương	1 năm
6	7	Bùi Phương Nam	Nhân Viên Thiết Kế Website	Hồ Chí Minh	Dưới 1 năm
7	8	Phạm Đăng Khoa	Kỹ Thuật Phần Mềm	Hồ Chí Minh	2 năm
8	9	Huỳnh Tấn Thành	It System/network/helpdesk	Bình Dương	4 năm
9	10	Trương Minh Hiếu	Nhân Viên Marketing	Bình Dương, Hồ Chí Minh	1 năm
10	11	Nguyễn Trần Trí Tuệ	It Technical	Bình Dương	1 năm

Hình 7 Thông tin dữ liệu thông tin người tìm việc

## 4.3 Xây dựng mô hình gợi ý

### 4.3.1 Trích xuất các trường dữ liệu

Để xây dựng hệ tư vấn sử dụng phương pháp lọc tác, ở bộ dữ liệu này, chúng ta cần sử dụng các trường dữ liệu quan trọng như sau:

```
▶ title_job_user= users_top_1000["job_name"]  
title_job_sup = jobs_top_1000["title"]
```

```
▶ title_job_user[:5]
```

```
0          Designer  
1      Thực Tập Sinh Thiết Kế Website  
2      Internship Ux/ui Designer  
3      Sinh Viên Thực Tập Mobile, Front End Web  
4      Thực Tập Sinh Front End  
Name: job_name, dtype: object
```

```
[ ] title_job_sup[:5]
```

```
0      Nhân Viên Kinh Doanh Phần Mềm / Thiết Kế Website  
1      Nhân Viên Phòng Thiết Kế Đồ Họa - Đi Làm Ngay  
2          Nhân Viên Kỹ Thuật  
3          Nhân Viên Marketing  
4      Nhân Viên Quản Trị Web  
Name: title, dtype: object
```

Hình 8 Tạo list 'title' chứa job\_name, title của ứng viên và nhà tuyển dụng



Dựa vào mức độ phù hợp giữa công việc của ứng viên và lĩnh vực tuyển dụng của nhà tuyển dụng, chúng ta xây dựng được bảng thể hiện mức độ phù hợp giữa công việc của ứng viên và lĩnh vực tuyển dụng của nhà tuyển dụng

```
df_score[["EMPLOYER_JOB", "JOBS_NAME", "SCORE"]][:5]
```

	EMPLOYER_JOB	JOBS_NAME	SCORE
0	Designer	Nhân Viên Kinh Doanh Phần Mềm / Thiết Kế Website	5
1	Thực Tập Sinh Thiết Kế Website	Nhân Viên Phòng Thiết Kế Đồ Họa - Đi Làm Ngay	5
2	Internship Ux/ui Designer	Nhân Viên Kỹ Thuật	1
3	Sinh Viên Thực Tập Mobile, Front End Web	Nhân Viên Marketing	3
4	Thực Tập Sinh Front End	Nhân Viên Quản Trị Web	5

```
[ ] jobs_id = df_score["ID_JOBS"]
users_id = df_score["ID_EMPLOYER"]
scores = df_score["SCORE"]
```

Hình 9 Bảng điểm số giữa employer và jobs

Chuyển bảng trên thành Dataframe trong python:

```
jobs_id = df_score["ID_JOBS"]
users_id = df_score["ID_EMPLOYER"]
scores = df_score["SCORE"]
```

```
[ ] dict = {'jobs_id': jobs_id, 'users_id': users_id, 'scores': scores}
df = pd.DataFrame(dict)
```

```
[ ] df[:10]
```

	jobs_id	users_id	scores
0	1	1	5
1	2	2	5
2	3	3	1

Hình 10 Dataframe

### 4.3.2 Xây dựng mô hình lọc cộng tác

Xây dựng ma trận *scores\_matrix* biểu diễn mức độ phù hợp giữa ứng viên và công việc tuyển dụng dựa trên điểm số trong dataframe

```
# tạo bảng dòng = users_id, cột = jobs_id, giá trị = scores

[ ] scores_matrix = df.pivot_table(values='scores', index='jobs_id', columns='users_id', fill_value=0)
scores_matrix.head()
```

users_id	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
jobs_id																			
1	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Hình 11 Dataframe của scores\_matrix

Bước tiếp theo là phân rã ma trận. Kỹ thuật phân rã ma trận là một hướng tiếp cận khác của collaborative Filtering hay còn gọi là matrix factorization. Kỹ thuật phân rã ma trận là phương pháp chia ma trận lớn  $X$  thành hai ma trận có kích thước nhỏ hơn, sao cho ta có thể xây dựng lại  $X$  từ hai ma trận nhỏ hơn.

Sử dụng thư viện Truncated SVD để cắt ngắn và giảm kích thước của ma trận scores\_matrix

```

▶ #Decomposing the Matrix | phân rã ma trận ( làm ngắn ma trận) Truncated SVDs
from sklearn.decomposition import TruncatedSVD

SVD = TruncatedSVD(n_components=10)
decomposed_matrix = SVD.fit_transform(X)
decomposed_matrix.shape

↳ (124, 10)

[ ] decomposed_matrix[:2]

array([[0.00000000e+00, 0.00000000e+00, 0.00000000e+00, 0.00000000e+00,
        0.00000000e+00, 0.00000000e+00, 0.00000000e+00, 0.00000000e+00,
        0.00000000e+00, 5.00000000e+00],
       [2.79976979e-15, 2.13458428e-14, 2.06142588e-14, 2.83849188e-14,
        5.73564973e-14, 1.43725657e-13, 5.00000000e+00, 1.96367637e-15,
        1.20843230e-14, 0.00000000e+00]])

```

Hình 12 Dataframe của scores\_matrix

Tiếp theo là xây dựng ma trận lọc cộng tác:

```

▶ #Correlation Matrix | tạo ma trận tương quan
import numpy as np
correlation_matrix = np.corrcoef(decomposed_matrix)
correlation_matrix.shape

↳ /usr/local/lib/python3.7/dist-packages/numpy/lib/function_base.py:2691:
  c /= stddev[:, None]
/usr/local/lib/python3.7/dist-packages/numpy/lib/function_base.py:2692:
  c /= stddev[None, :]
(124, 124)

```

Hình 13 Dataframe của scores\_matrix

Từ ma trận lọc cộng tác, tiếp tục viết hàm gợi ý việc làm truyền vào *ID công việc* & *số lượng công việc muốn gợi ý*. Hàm này được gọi sẽ trả về list ID công việc gợi ý

```
[ ] def get_recommend_job_by_id(id_job, n_recommend= 4): # mặc định số công việc recommend = 4
    list_jobs_id = list(X.index) # lấy list jobs_id, X score_matrix

    id = list_jobs_id.index(id_job) # tìm ID trong list
    correlation_by_ID = correlation_matrix[id] # lấy ma trận tương quan của job có id_jobs ra

    # Đề xuất n jobs_id có mức độ tương quan cao theo thứ tự
    recommend = list(X.index[correlation_by_ID > 0.6])
    # xóa công việc user đã tìm (đã tìm thì k cần gợi ý )
    recommend.remove(id_job)
    return recommend[:n_recommend]
```

Hình 14 Hàm gợi ý việc làm truyền vào ID công việc & số lượng công việc

Cuối cùng là lưu lại ma trận đã tạo

```
# lưu model
import os
import pickle
MODEL_PATH = "/content/drive/MyDrive/Năm 4/NCKH/code/model/"
pickle.dump(correlation_matrix, open(os.path.join(MODEL_PATH,
                                                    "collaborative_filtering_tuvanvieclam.pkl"), 'wb'))
pickle.dump(X, open(os.path.join(MODEL_PATH, "scores_matrix.pkl"), 'wb'))
```

Hình 15 Lưu model

## 4.4 Xây dựng Website demo ứng dụng phương pháp lọc cộng tác(Django Framework)

### 4.4.1 Khởi tạo project Django

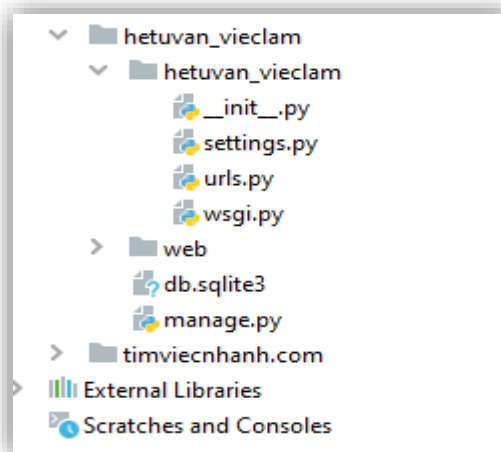
- Tạo project hetuvan\_vieclam bằng Django trên Pycharm trên Terminal:

```
> django-admin startproject hetuvan_vieclam
```

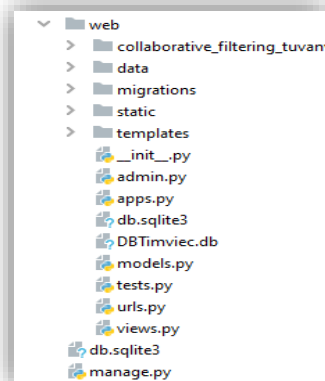
- Tạo 1 app tên *Web*

```
> python manage.py startapp polls
```

- Cấu trúc project được tạo như sau:



Hình 16 Cấu trúc project của ứng dụng gợi ý



Hình 17 Cấu trúc thư mục Web của ứng dụng gợi ý

#### 4.4.2 Khai báo Class và khởi tạo database

- Khai báo trong file *models.py* gồm 2 class là **Employer** và **Jobs**

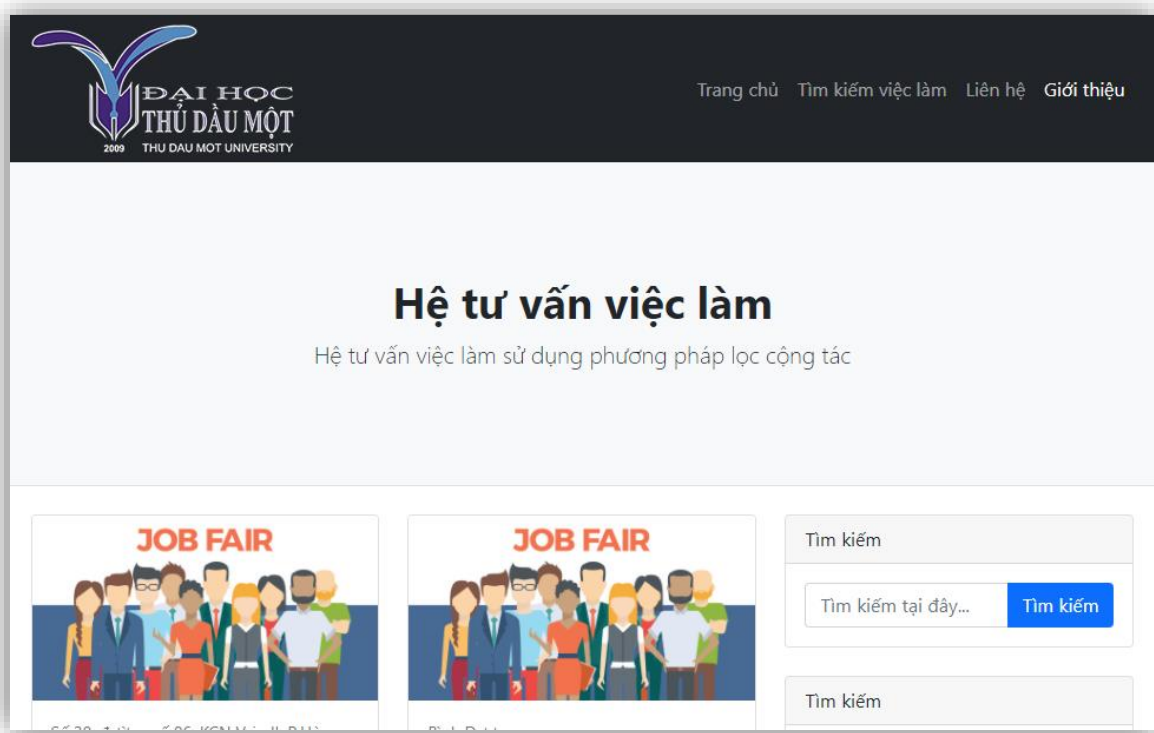
```
class employer(models.Model):
    name = models.CharField(max_length=100)
    job_name = models.CharField(max_length=200)
    location = models.CharField(max_length=200)
    ex = models.CharField(max_length=100)
    def __str__(self):
        return self.name

class jobs(models.Model):
    title = models.CharField(max_length=200)
    link = models.CharField(max_length=200)
    address = models.CharField(max_length=200)
    salary = models.CharField(max_length=20)
    ex = models.CharField(max_length=100)
    level = models.CharField(max_length=100)
    location = models.CharField(max_length=100)
    job_name = models.CharField(max_length=200)
    def __str__(self):
        return self.title
```

Hình 18 Khai báo 2 class Employer và Jobs

- Tạo database bằng 2 câu lệnh sau:
  - > python manage.py makemigrations
  - > python manage.py migrate

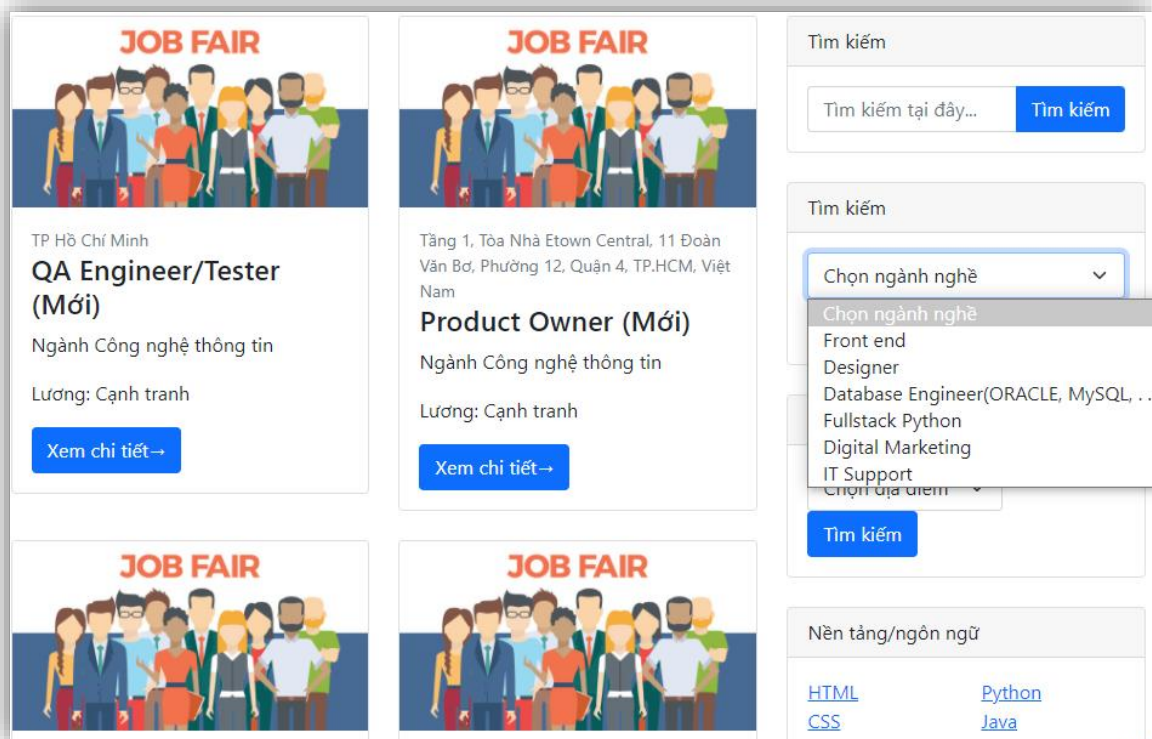
### 4.3.3 Giao diện hệ thống gợi ý



Hình 19 Trang chủ tìm kiếm của ứng dụng

Bố cục chính gồm có:

- + Header: chứa logo, thanh menu gồm các page: trang chủ, tìm kiếm việc làm, liên hệ, giới thiệu
- + Body: chứa nội dung cụ thể của từng bài post, các công việc được gợi ý
- + Sidebar bên phải: chứa ô tìm kiếm, dropdown list tìm kiếm theo ngành nghề, địa điểm, nội dung giới thiệu ngắn về phương pháp lọc cộng tác
- + Footer: Copyright © Thu Dau Mot University 2022



Hình 20 Trang chủ tìm kiếm của ứng dụng

- Trang chủ tìm kiếm gồm có: kết quả gợi ý, sidebar tìm kiếm chứa tìm kiếm theo ngành nghề, tìm kiếm theo địa điểm
- Mỗi công việc gợi ý nằm trong 1 ô chứa các thông tin sau: Địa điểm tuyển dụng, tiêu đề tuyển dụng, ngành nghề tuyển dụng, mức lương offer và nút xem chi tiết



23/03/2022

Nhân Viên Phòng Thiết Kế Đồ Họa - Đi Làm Ngay

Vị trí tuyển dụng: Nhân Viên Phòng Thiết Kế Đồ Họa - Đi Làm Ngay

Link tuyển dụng: <https://timviecnhanh.com/tuyen-nhan-vien-phong-thiet-ke-do-hoa-di-lam-ngay-binh-duong-100171123.html>

Nơi làm việc: **BÌNH GIAO, THUẬN GIAO, THUẬN AN, BÌNH DƯƠNG**

Mức lương: **10 - 12 triệu**

Ngành nghề: Thiết kế/Mỹ thuật Công nghệ thông tin Kiến trúc/Nội thất

Xem chi tiết tại web tuyển dụng→

Các công việc liên quan

Bình Dương

Nhân Viên Phòng Thiết Kế Đồ Họa - Đi Làm Ngay

Ngành Thiết kế/Mỹ thuật Công nghệ thông tin Kiến trúc/Nội thất

Lương: 10 - 12 triệu

Bình Dương TP.HCM

Nhân Viên Digital Marketing

Ngành Quảng cáo/Marketing/PR Công nghệ thông tin Môi trường/Xử lý chất thải

Lương: 10 - 12 triệu

Tim kiếm

Tim kiếm tại đây...

Tim kiếm

Tim kiếm

Chọn ngành nghề

Tim kiếm

Tim kiếm theo tỉnh thành

Chọn địa điểm

Tim kiếm

Nền tảng/ngôn ngữ

HTML

Python

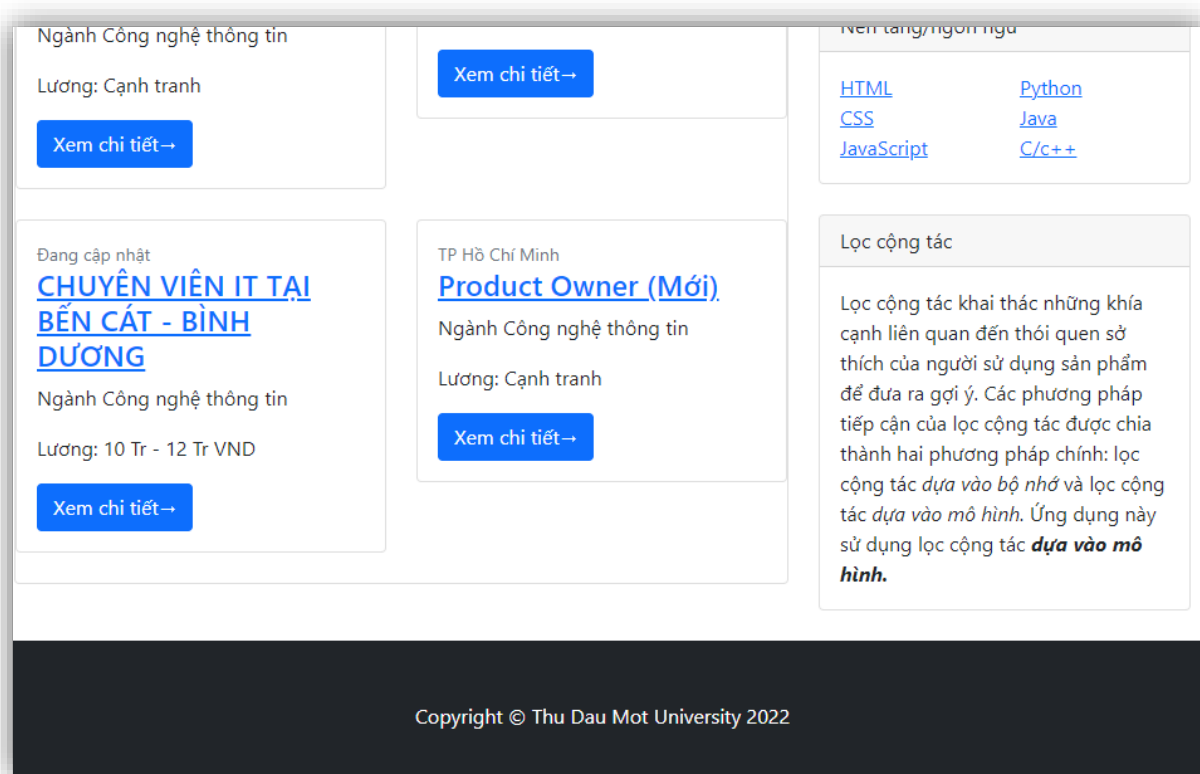
CSS

Java

Hình 21 Giao diện bài post

- Giao diện bài post gồm có: tiêu đề bài post, vị trí tuyển dụng, link tuyển dụng, nơi làm việc, mức lương offer, lĩnh vực ngành nghề tuyển dụng, nút xem chi tiết web tuyển dụng và các công việc liên quan đến công việc hiện tại (các công việc này được hệ thống gợi ý bằng phương pháp lọc cộng tác)

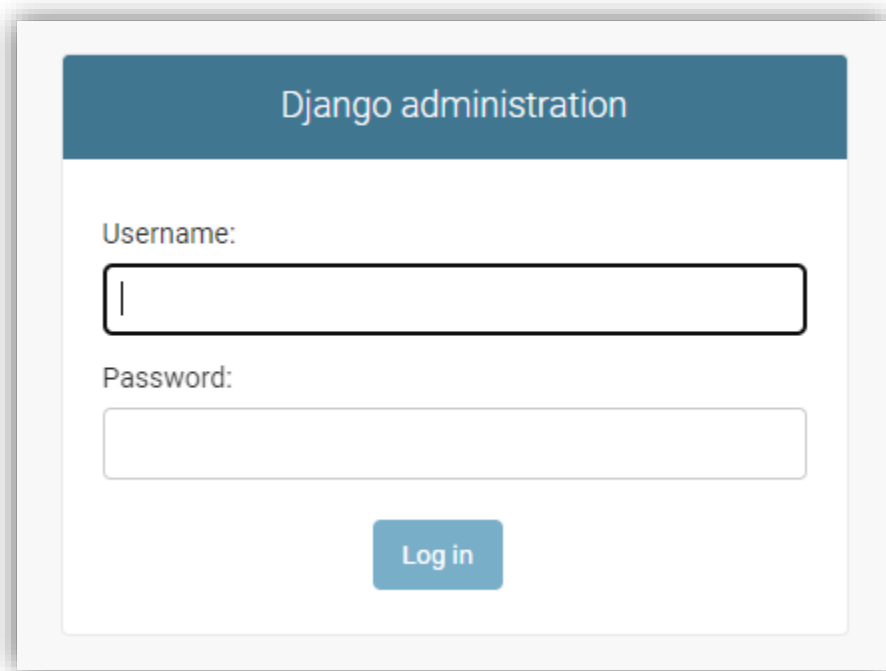
41



Hình 22 Phần footer

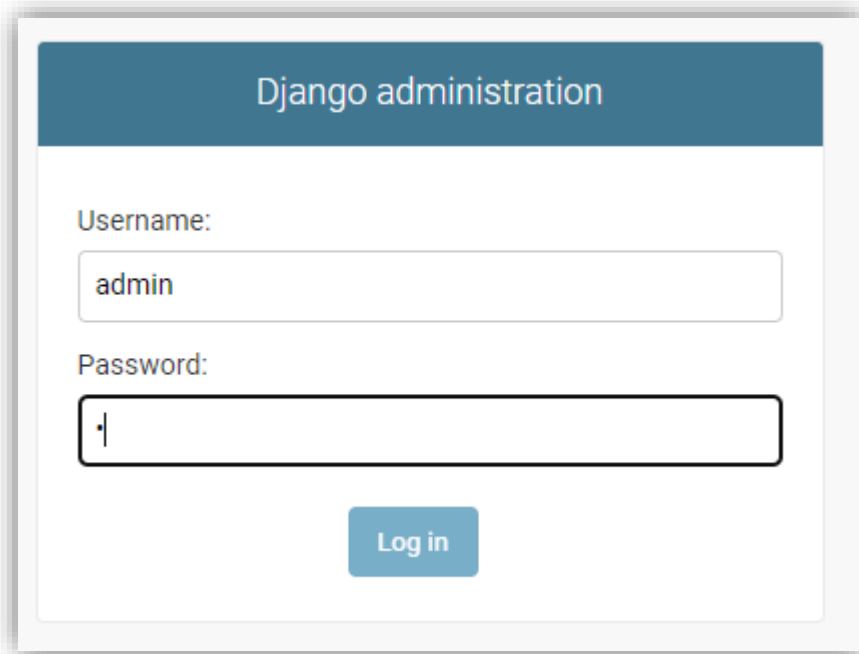
- Phần footer chứa copyright

#### 4.3.4 Giao diện trang quản trị



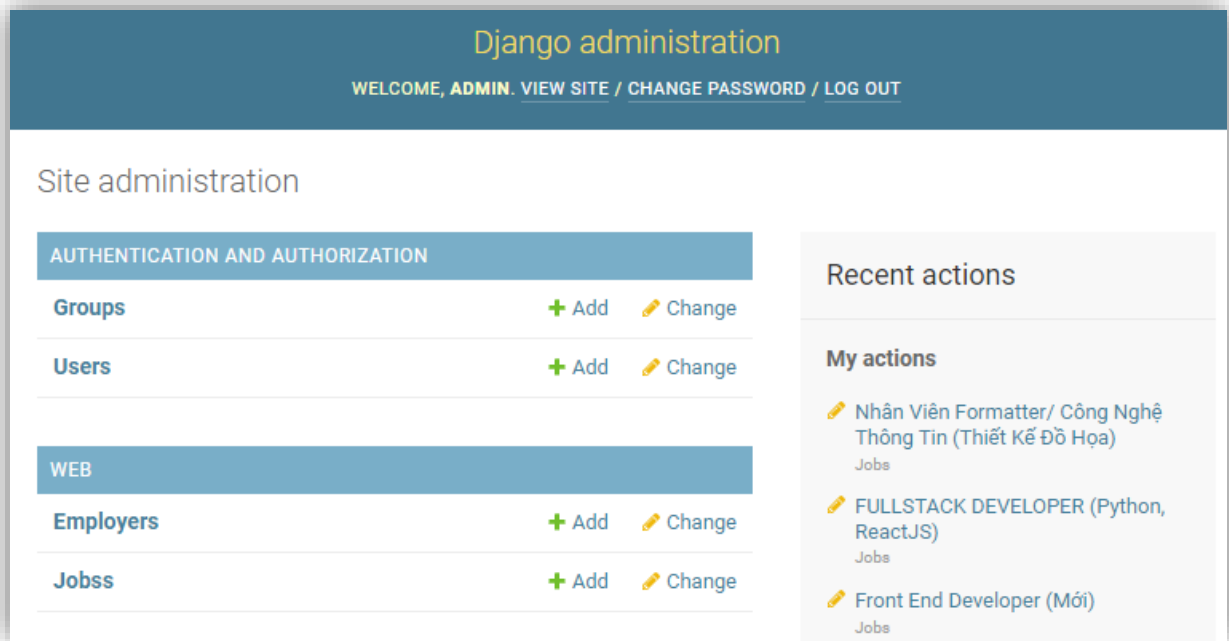
The screenshot shows the Django administration login interface. At the top, there is a dark blue header with the text "Django administration" in white. Below the header, the page has a white background. There are two labels: "Username:" and "Password:". Under "Username:" is a text input field with a single vertical line cursor. Under "Password:" is a text input field. Below these fields is a blue button with the text "Log in" in white.

Hình 23 Giao diện đăng nhập trang admin



This screenshot is similar to the previous one, showing the Django administration login interface. The "Username:" input field now contains the text "admin". The "Password:" input field has a single character, possibly a dot or a small symbol, followed by a vertical line cursor. The "Log in" button remains at the bottom.

Hình 24 Giao diện đăng nhập trang admin



Hình 25 Giao diện đăng nhập trang admin

- Phần quản trị gồm 2 mục chính:
  - + Quản trị các user đăng nhập, phân quyền cho user
  - + Quản trị dữ liệu của ứng dụng: gồm 2 class được khai báo trong file *models.py* đó là **Employer** và **Jobs**
- Quản trị Employer: chứa danh sách employer, chức năng thêm, xóa, sửa employer

Django administration

WELCOME, **ADMIN**. [VIEW SITE](#) / [CHANGE PASSWORD](#) / [LOG OUT](#)

Home › Web › Employers

Select employer to change ADD EMPLOYER +

Action:   0 of 19 selected

<input type="checkbox"/>	EMPLOYER
<input type="checkbox"/>	Nguyễn Văn Nam
<input type="checkbox"/>	Nguyễn Thị Hạnh Dung
<input type="checkbox"/>	Nguyễn Đức Long
<input type="checkbox"/>	Trần Quốc Dũng
<input type="checkbox"/>	Hoàng Kim Tuyến
<input type="checkbox"/>	Võ Trọng Hiến
<input type="checkbox"/>	Lê Quang Vinh
<input type="checkbox"/>	Trịnh Trọng Thịnh
<input type="checkbox"/>	Nguyễn Trần Trí Tuệ

Hình 26 Quản trị Employer

- Quản trị Jobs: chứa danh sách jobs, chức năng thêm, xóa, sửa jobs

Django administration

WELCOME, ADMIN. [VIEW SITE](#) / [CHANGE PASSWORD](#) / [LOG OUT](#)

Home > Web > Jobss

Select jobs to change

ADD JOBS +

Action:  Go 0 of 100 selected

☐ JOBS

☐ [Urgent] - Project Manager

☐ [E-commerce] IT Product Assistant Manager

☐ Software Developers

☐ UI/UX DESIGNER

☐ Senior Credit Risk Analyst

☐ Senior System Administrator

☐ UI/UX DESIGNER

☐ JAVA DEVELOPER (SPRING)

☐ Nhân Viên Quản Trị Hệ Thống Mạng Data Center

Hình 27 Quản trị Employer

Django administration

WELCOME, **ADMIN**. [VIEW SITE](#) / [CHANGE PASSWORD](#) / [LOG OUT](#)

Home › Web › Employers › Add employer

### Add employer

Name:	<input type="text" value="Hoàng Kim Tuyển"/>
Job name:	<input type="text" value="Python Fullstack"/>
Location:	<input type="text" value="Bình Dương"/>
Ex:	<input type="text" value="1 năm"/>

[Save and add another](#) [Save and continue editing](#) [SAVE](#)

Hình 28 Thêm mới một employer

Django administration

WELCOME, **ADMIN**. [VIEW SITE](#) / [CHANGE PASSWORD](#) / [LOG OUT](#)

Home › Web › Employers › Nguyễn Đức Long

Change employer HISTORY

**Name:**

**Job name:**

**Location:**

**Ex:**

[Delete](#) [Save and add another](#) [Save and continue editing](#) [SAVE](#)

Hình 29 Sửa/xóa 1 employer



Add jobs

Title:

Quản trị hệ thống mạng

Link:

https://careerbuilder.vn/vi/tim-viec-lam/nh

Address:

Nguyễn Huệ, Phường Bến Nghé, Quận 1, T

Salary:

Thương lượng

Ex:

1 năm

Level:

Đại học

Location:

TP Hồ Chí Minh

Job name:

Công nghệ thông tin

Save and add another

Save and continue editing

SAVE

Hình 30 Thêm mới 1 jobs

Change jobs

HISTORY

Title:	PYTHON DEVELOPER
Link:	<a href="https://careerbuilder.vn/vi/tim-viec-lam/py">https://careerbuilder.vn/vi/tim-viec-lam/py</a>
Address:	TP Hồ Chí Minh
Salary:	Cạnh tranh
Ex:	Đang cập nhật
Level:	Đang cập nhật
Location:	TP Hồ Chí Minh
Job name:	Công nghệ thông tin

Delete

Save and add another

Save and continue editing

SAVE

Hình 31 Sửa/xóa 1 jobs

## **CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN CHO ĐỀ TÀI**

### **5.1 Kết luận**

Trong đề tài Xây dựng mô hình hệ tư vấn tìm việc online cho sinh viên trong trạng thái bình thường mới” chúng tôi đã nghiên cứu về phương pháp lọc cộng tác, trong phương pháp này chúng tôi đã nghiên cứu các kỹ thuật để xây dựng nên một hệ tư vấn việc làm.

Qua một quá trình xây dựng phần back-end và front-end cho ứng dụng, cơ bản đã xây dựng được 1 sản phẩm demo về hệ tư vấn sử dụng phương pháp lọc cộng tác, biểu diễn một cách trực quan trên website.

### **5.2 Hướng phát triển**

Trong thời gian tới, nhóm dự định sẽ phát triển thêm các chức năng đăng kí, đăng nhập cho ứng viên và nhà tuyển dụng, thêm các chức năng cập nhật thông tin cho ứng viên, đăng tin tuyển dụng cho nhà tuyển dụng, triển khai và đưa ứng dụng lên server để mọi người có thể truy cập.

Bên cạnh đó không chỉ dựa vào đề tài nhóm sẽ phát triển đề tài dựa trên bộ dữ liệu của Việt Nam nhằm tạo ra hệ tư vấn giúp người Việt giải quyết một phần nào đó để tìm công việc phù hợp với chuyên môn chuyên ngành cũng như kinh nghiệm của mình.

## TÀI LIỆU THAM KHẢO

- [1] Yunkyoung Lee (Fall 2015) Đại học Công lập San Jose, *Recommend recommendation sy tion system using coll stem using collaborative fil tive filtering*:  
[https://scholarworks.sjsu.edu/cgi/viewcontent.cgi?article=1444&context=etd\\_projects](https://scholarworks.sjsu.edu/cgi/viewcontent.cgi?article=1444&context=etd_projects)
- [2] Nguyen Thi Hop (2020) Viblo.asia forum, *Introduction to Recommender Systems*:  
<https://viblo.asia/p/introduction-to-recommender-systems-aWj53LQ8K6m>
- [3] Machine Learning cơ bản(2017) *Neighborhood-Based Collaborative Filtering*:  
<https://machinelearningcoban.com/2017/05/24/collaborativefiltering/>
- [4] Học viện đào tạo CNTT NIIT - ICT Hà Nội(08/09/2020) *Lập trình Web với Django*:  
<https://niithanoi.edu.vn/django-la-gi.html>
- [5] Django documentation(2.2), *Writing your first Django app, part 1*:  
<https://docs.djangoproject.com/en/2.2/intro/tutorial01/>
- [6] Thư viện cào dữ liệu trang web, BeautifulSoup(4 4.10.0):  
<https://pypi.org/project/beautifulsoup4/>
- [7] Thư viện Bootstrap(v5.1):  
<https://getbootstrap.com/docs/5.1/getting-started/introduction/>