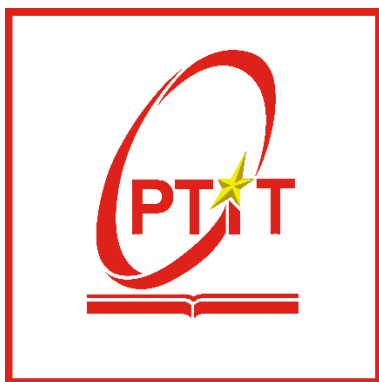


**BỘ THÔNG TIN VÀ TRUYỀN THÔNG  
HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG  
CƠ SỞ TẠI THÀNH PHỐ HỒ CHÍ MINH**

=====Ω=====



## **BÁO CÁO**

### **PHÁT TRIỂN CÁC HỆ THỐNG THÔNG MINH**

### **ĐỀ TÀI: XÂY DỰNG HỆ THỐNG QUẢN LÝ TÀI LIỆU ĐIỆN TỬ TÍCH HỢP ĐỀ XUẤT THÔNG MINH**

**Giảng viên hướng dẫn: Nguyễn Ngọc Duy**

**Thành viên nhóm 7:**

Nguyễn Tấn Nguyên                      N21DCCN156

Nguyễn Phi Long                        N21DCCN142

Nguyễn Phúc Minh Khang            N21DCCN133

*Thành phố Hồ Chí Minh, ngày 07 tháng 12 năm 2024*

# LỜI CẢM ƠN

Trong thời gian thực hiện đề tài, nhóm em đã cố gắng vận dụng những kiến thức đã học trên lớp, trong thực tế để hoàn thành tốt đề tài.

Nhóm em xin chân thành cảm ơn Thầy Nguyễn Ngọc Duy đã tận tình chỉ bảo và giúp đỡ, giải đáp các vướng mắc để nhóm em có thể hoàn thành đề tài cũng như môn học này. Trong quá trình thực hiện đề tài nghiên cứu sẽ không thể tránh khỏi những sai sót. Nhóm em sẵn sàng và mong muốn nhận được sự góp ý của Thầy và các bạn để nội dung đề tài này ngày càng hoàn thiện hơn.

## **MUC LUC**

I- GIỚI THIỆU ĐỀ TÀI:.....	1
1. Lý do chọn đề tài:.....	1
2. Mục tiêu nghiên cứu:.....	1
3. Đối tượng và phạm vi nghiên cứu:.....	1
4. Phương pháp nghiên cứu:.....	1
5. Nội dung báo cáo:.....	2
II- CƠ SỞ LÝ THUYẾT VÀ CÔNG NGHỆ:.....	3
1. Giới thiệu về Machine Learning (Học máy):.....	3
2. Kỹ thuật xây dựng hệ thống đề xuất dựa trên kỹ thuật cá nhân hóa:.....	3
3. Machine learning trong hệ thống đề xuất:.....	3
4. Thuật toán K-Means:.....	3
4.1. Giới thiệu: .....	3
4.2. Ý tưởng của bài toán K-Means: .....	4
4.3. Ưu điểm và nhược điểm:.....	5
5. Elbow trong việc phân cụm:.....	6
6. Công nghệ sử dụng:.....	7
III- PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG:.....	8
1. Giới thiệu hệ thống quản lý tài liệu điện tử tích hợp đề xuất thông minh cho người dùng: .....	8
2. Mô hình Usecase:.....	9
3. Sơ đồ ERD:.....	11
4. Sơ đồ lớp: .....	12
5. Mô hình dữ liệu quan hệ từ ERD: .....	12
6. Sơ đồ diagram: .....	13
7. Từ điển dữ liệu: .....	13
IV- THIẾT KẾ THÔNG MINH:.....	16
1. Hệ thống đề xuất thông minh dựa vào người dùng có tài khoản: .....	16
2. Thiết kế giao diện:.....	20
a. Giao diện chính: .....	20
b. Giao diện thông minh: .....	27
V- KẾT QUẢ THỰC NGHIỆM THỬ NGHIỆM: .....	29

1. Phân cụm dữ liệu bằng K-Means: .....	29
2. Đánh giá kết quả phân cụm: .....	29
3. Đánh giá độ hiệu quả của hệ thống: .....	32
4. Kết luận .....	32
VI- KẾT LUẬN: .....	34
TÀI LIỆU THAM KHẢO: .....	36

## **I- GIỚI THIỆU ĐỀ TÀI:**

### **1. Lý do chọn đề tài:**

Hiện nay, ngoài việc tìm kiếm tài liệu nghiên cứu tại các thư viện, tài liệu điện tử đang là một nguồn thông tin rộng lớn, dễ tiếp cận mà mọi người đang tìm cách khai thác một cách hiệu quả, đặc biệt là sinh viên. Không giống như tài liệu truyền thống, thường là các sách, báo có thể cầm, cảm nhận được, **Tài liệu điện tử** là những tài liệu, thông tin, hoặc dữ liệu được lưu trữ và truyền tải dưới dạng số hóa, có thể được truy cập và sử dụng thông qua các thiết bị điện tử như máy tính, điện thoại di động, máy tính bảng, và các thiết bị khác có kết nối internet. Các tài liệu này có thể có nhiều định dạng khác nhau, bao gồm văn bản, hình ảnh, âm thanh, video, hoặc các định dạng tài liệu phức tạp hơn như PDF, eBook, bài báo nghiên cứu, báo cáo, bài giảng trực tuyến, v.v. Lợi ích nổi bật của tài liệu điện tử đó chính là khả năng truy cập nhanh chóng, lưu trữ gọn nhẹ và khả năng dễ dàng chia sẻ. Chính vì vậy, các website, thư viện số giúp cho mọi người có thể tìm kiếm và tải các tài liệu điện tử đang trở phổ biến trong thời đại công nghệ thông tin phát triển nhanh chóng.

Để hỗ trợ cho việc tìm kiếm tài liệu hiệu quả, việc xây dựng một hệ thống dựa vào thói quen truy cập, lịch sử tải của người dùng, từ đó đề xuất các chủ đề liên quan giúp người dùng có thể tiết kiệm thời gian trong việc tìm kiếm các tài liệu mong muốn.

### **2. Mục tiêu nghiên cứu:**

- Tìm hiểu hệ thống khuyến nghị các tài liệu điện tử dựa vào thói quen truy cập, lịch sử tải của người dùng.
- Nghiên cứu về thuật toán K-Means và mô hình học máy không giám sát.
- Thu thập, tìm hiểu, phân tích các tài liệu liên quan đến đề tài.
- Ứng dụng của thuật toán K-Means vào việc phân cụm các tài liệu tương đồng để phục vụ cho hệ thống đề xuất.
- Cài đặt thử nghiệm phương pháp và đánh giá kết quả.

### **3. Đối tượng và phạm vi nghiên cứu:**

Đối tượng nghiên cứu: bài toán hệ thống đề xuất các tài liệu điện tử dựa vào lịch sử tải của người dùng, thuật toán K-Means và mô hình học máy không giám sát.

Phạm vi nghiên cứu: Tìm hiểu lý thuyết về hệ thống đề xuất các tài liệu điện tử dựa vào lịch sử tải của người dùng, mô hình học máy không giám sát, K-Means và giải quyết bài toán hệ thống đề xuất các tài liệu điện tử dựa vào lịch sử tải của người dùng kết hợp với mô hình học máy không giám sát và thuật toán K-Means.

### **4. Phương pháp nghiên cứu:**

#### ***Phương pháp nghiên cứu lý thuyết:***

- Nghiên cứu, thu thập thông tin về mô hình học máy không giám sát, thuật toán K-Means và hệ thống đề xuất các tài liệu điện tử dựa vào lịch sử tải của người dùng.
- Nghiên cứu ứng dụng K-Means vào giải quyết các bài toán phân cụm.

#### ***Phương pháp nghiên cứu thực nghiệm:***

- Xây dựng một website thư viện tài liệu điện tử và đưa hệ thống đề xuất tài liệu vào website. Dựa vào kết quả để phân tích, đánh giá hệ thống thông minh.

## **5. Nội dung báo cáo:**

Báo cáo được trình bày gồm 6 chương:

- Chương I: Giới thiệu đề tài.
- Chương II: Cơ sở lý thuyết và công nghệ.
- Chương III: Phân tích và thiết kế hệ thống.
- Chương IV: Xây dựng thiết kế thông minh.
- Chương V: Kết quả thực hiện thử nghiệm.
- Chương VI: Kết luận.

## II- CƠ SỞ LÝ THUYẾT VÀ CÔNG NGHỆ:

### 1. Giới thiệu về Machine Learning (Học máy):

**Machine Learning (Học máy)** là một lĩnh vực con của trí tuệ nhân tạo (AI) tập trung vào việc phát triển các thuật toán và mô hình giúp máy tính có thể học hỏi từ dữ liệu và tự cải thiện mà không cần được lập trình rõ ràng. Thay vì chỉ làm theo các quy tắc cố định, các hệ thống học máy có khả năng phân tích dữ liệu, tìm ra mẫu (pattern) và sử dụng các thông tin đó để đưa ra dự đoán hoặc quyết định trong tương lai.

Học máy được chia thành 2 loại chính gồm:

- Học có giám sát là phương pháp sử dụng những dữ liệu được gán nhãn sẵn để suy luận ra quan hệ giữa đầu vào và đầu ra. Sau khi tìm hiểu cách tốt nhất để mô hình hóa các mối quan hệ cho dữ liệu được gán nhãn, thuật toán huấn luyện sẽ được sử dụng cho các bộ dữ liệu mới. Học tập có giám sát có thể được nhóm lại thành các vấn đề về phân loại và hồi quy.
- Học không giám sát sử dụng những dữ liệu chưa được gán nhãn sẵn để suy luận và tìm cách để mô tả dữ liệu cùng cấu trúc của chúng. Ứng dụng của học không giám sát đó là hỗ trợ phân loại thành các nhóm có đặc điểm tương đồng.

### 2. Kỹ thuật xây dựng hệ thống đề xuất dựa trên kỹ thuật cá nhân hóa:

Hệ thống đề xuất cá nhân hóa (Personalized Recommender System) là một hệ thống đề xuất được thiết kế để cung cấp các gợi ý phù hợp với sở thích, nhu cầu và hành vi của từng người dùng cụ thể. Các hệ thống này sử dụng dữ liệu cá nhân, như lịch sử duyệt web, các đánh giá của người dùng, hay các hành động mà người dùng đã thực hiện trước đó (ví dụ: mua hàng, xem phim, nghe nhạc) để đưa ra những đề xuất chính xác và phù hợp.

Spotify sử dụng các kỹ thuật học máy để tạo ra danh sách phát (playlist) được cá nhân hóa. Một trong những tính năng nổi bật là Discover Weekly, nơi mỗi tuần người dùng nhận được một danh sách các bài hát mới mà Spotify cho là họ sẽ thích, dựa trên hành vi nghe nhạc trước đó. Họ đã sử dụng **Collaborative Filtering** (dựa vào dữ liệu hành vi của người dùng (như phim đã xem, các bài hát đã nghe) để tìm những người dùng tương tự và đề xuất các nội dung mà họ cũng đã thích) và **Content-based Filtering** (đề xuất các bộ phim hoặc bài hát tương tự dựa trên nội dung của chúng (thể loại phim, nghệ sĩ))

### 3. Machine learning trong hệ thống đề xuất:

Với một nguồn tài liệu điện tử đa dạng và phong phú, việc phân tích dữ liệu để xây dựng hệ thống đề xuất sẽ gặp rất nhiều khó khăn. Chính vì vậy, việc ứng dụng Machine learning vào xây dựng hệ thống đề xuất các tài liệu sẽ giúp cho việc nghiên cứu, phát triển trở nên hiệu quả.

### 4. Thuật toán K-Means:

#### 4.1. Giới thiệu:

**K-Means clustering** là một thuật toán phân cụm thuộc nhóm Học không giám sát dùng để hỗ trợ phân loại thành các nhóm sản phẩm khác nhau sao cho các sản phẩm trong cùng một cụm có đặc điểm, tính chất tương đồng

---

#### 4.2. Ý tưởng của bài toán K-Means:

Ý tưởng đơn giản nhất về cluster (cụm) là tập hợp các điểm ở gần nhau trong một không gian nào đó (không gian này có thể có rất nhiều chiều trong trường hợp thông tin về một điểm dữ liệu là rất lớn). Hình bên dưới (Hình 1) là một ví dụ về 3 cụm dữ liệu (từ giờ nhóm em sẽ viết gọn là *cluster*).



Hình 1: Bài toán với 3 cụm.

Giả sử mỗi cluster có một điểm đại diện (*center*) hình tròn. Và những điểm xung quanh mỗi center thuộc vào cùng nhóm với center đó. Một cách đơn giản nhất, xét một điểm bất kỳ, ta xét xem điểm đó gần với center nào nhất thì nó thuộc về cùng nhóm với center đó.

##### Các bước của Thuật toán K-Means:

- Bước 1: Chọn K điểm bất kỳ làm các center ban đầu.
- Bước 2: Phân mỗi điểm dữ liệu vào cụm có center gần nó nhất.
- Bước 3: Nếu việc gán dữ liệu vào từng cụm ở bước 2 không thay đổi so với vòng lặp trước nó thì dừng thuật toán.
- Bước 4: Cập nhật center cho từng cụm bằng cách lấy trung bình cộng của tất cả các điểm dữ liệu đã được gán vào cụm đó sau bước 2.
- Bước 5: Quay lại bước 2.

##### Điều kiện dừng thuật toán:

Xác định điều kiện dừng thuật toán theo một trong các số cách như sau:

- Tại 1 vòng lặp: có ít các điểm dữ liệu được gán sang cluster khác
- Điểm trung tâm (centroid) không thay đổi nhiều.
- Giá trị hàm mất mát không thay đổi nhiều:



$$Error = \sum_{i=1}^k \sum_{\mathbf{x} \in C_i} d(\mathbf{x}, \mathbf{m}_i)^2$$

Trong đó  $C_i$  là cluster thứ  $i$ ,  $m_i$  là điểm trung tâm của cluster  $C_i$  tương ứng.

**Xác định điểm trung tâm của cluster:**

Để xác định điểm trung tâm của cluster:

$$\mathbf{m}_i = \frac{1}{|C_i|} \sum_{\mathbf{x} \in C_i} \mathbf{x}$$

Trong đó  $C_i$  là cluster thứ  $i$ ,  $m_i$  là điểm trung tâm của cluster  $C_i$  tương ứng.

**Phép đo khoảng cách:**

Trong K-means để đánh giá mức độ giống nhau hay khoảng cách giữa 2 điểm dữ liệu ta có thể sử dụng các phép đo khoảng cách khác nhau. Ngoài khoảng cách Euclidean, tùy thuộc vào từng bài toán có thể sử dụng phương pháp đo khác

$$d(\mathbf{x}, \mathbf{m}_i) = \|\mathbf{x} - \mathbf{m}_i\| = \sqrt{(x_1 - m_{i1})^2 + (x_2 - m_{i2})^2 + \dots + (x_n - m_{in})^2}$$

Mọi phương pháp tính khoảng cách giữa 2 vector đều có thể được sử dụng. Mỗi cách tính khoảng cách thể hiện cách nhìn nhận về dữ liệu

**Ảnh hưởng của outlier:**

Outlier (hay còn gọi là điểm ngoại lệ) là một hoặc nhiều cá thể khác hẳn đối với các thành viên còn lại của nhóm. Sự khác biệt này có thể dựa trên nhiều tiêu chí khác nhau như giá trị hay thuộc tính.

Outlier có thể xuất hiện vì nhiều lý do, chẳng hạn như lỗi trong quá trình thu thập dữ liệu, sự kiện bất thường, hoặc đặc điểm tự nhiên của dữ liệu. Tùy vào bối cảnh, điểm ngoại lệ có thể là thông tin hữu ích hoặc là sự cố cần phải loại bỏ.

#### 4.3. Ưu điểm và nhược điểm:

**Ưu điểm:**

- Đơn giản và dễ hiểu: Cấu trúc thuật toán dễ hiểu và dễ làm việc với các bộ dữ liệu nhỏ và vừa.
- Tốc độ tính toán nhanh: Với tập dữ liệu lớn, K-Means có thể hoạt động nhanh nên số lượng cụm không quá lớn.
- Hiệu quả với dữ liệu phân phối đều: Nếu dữ liệu có sự phân phối đồng đều, K-Means sẽ phân nhóm khá chính xác và nhanh chóng.

- Dễ dàng mở rộng: Thuật toán có thể mở rộng và áp dụng trong nhiều lĩnh vực khác nhau, từ phân tích khách hàng, phân tích thị trường, đến nhận diện mẫu.

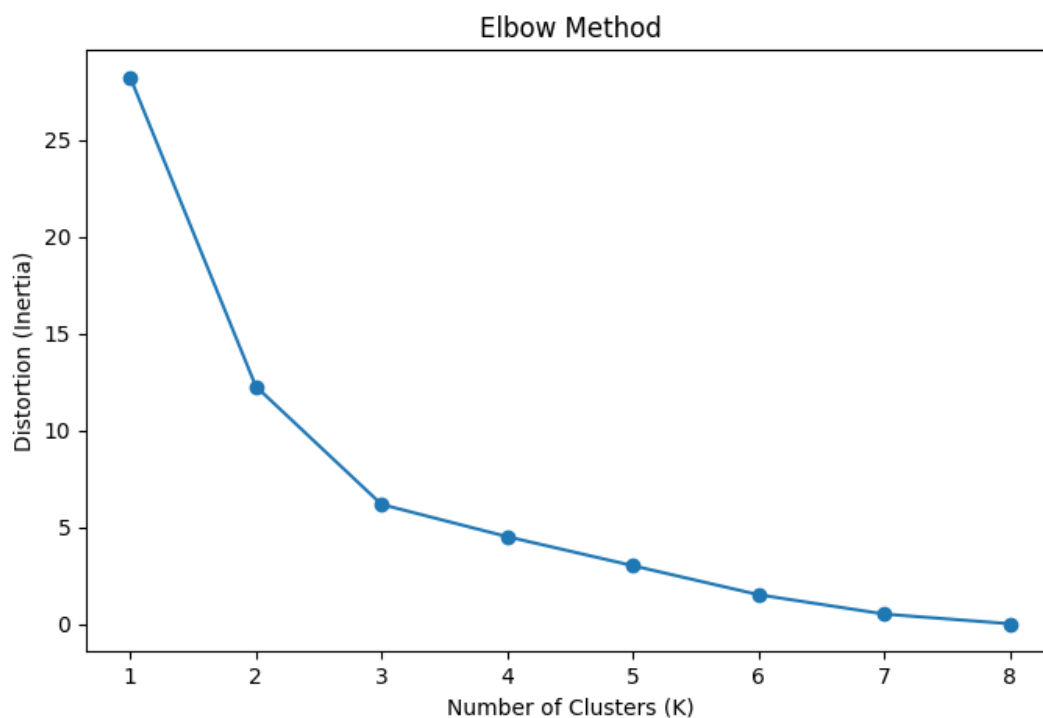
**Nhược điểm:**

- Xác định được số lượng cụm.
- Kết quả thuật toán có thể khác nhau tùy vào cách khởi tạo ban đầu.
- Khó khăn với dữ liệu có dạng phân phối phức tạp.
- Việc lựa chọn cách tính khoảng cách cho bài toán cụ thể khó.
- Nhạy cảm với các điểm dữ liệu outlier.
- Chỉ có thể áp dụng khi tính được trọng tâm.

## 5. Elbow trong việc phân cụm:

Trong phương pháp Elbow, thay đổi số lượng cụm  $K$  từ 1 – 10. Đối với mỗi giá trị của  $K$ , nhóm em tính toán Trung bình cộng khoảng cách trong cụm), hay còn được gọi là WCSS

WCSS chính là tổng bình phương khoảng cách giữa mỗi điểm và trọng tâm trong một cụm. Sau đó vẽ đồ thị với giá trị  $K$ , đồ thị trông giống như một khuỷu tay. Khi số cụm tăng lên, giá trị WCSS sẽ bắt đầu giảm. Giá trị WCSS lớn nhất khi  $K = 1$ . Khi phân tích đồ thị, ta nhận thấy rằng đồ thị sẽ thay đổi nhanh chóng tại một điểm và do đó tạo ra hình dạng khuỷu tay. Từ thời điểm này, đồ thị bắt đầu di chuyển gần như song song với trục  $X$ . Giá trị  $K$  tương ứng với điểm này là giá trị  $K$  tối ưu hoặc số cụm tối ưu. Dưới đây là hình ảnh minh họa về đồ thị của WCSS.



Hình 2: đồ thị Elbow minh họa

**6. Công nghệ sử dụng:**

- Frontend: Vuejs
- Backend: Python (Fast API), Java Spring Boot
- Database: SQL Server.

### III- PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG:

#### 1. Giới thiệu hệ thống quản lý tài liệu điện tử tích hợp đề xuất thông minh cho người dùng:

##### a. Giới thiệu về hệ thống quản lý tài liệu điện tử tích hợp đề xuất thông minh cho người dùng:

Các admin (người quản trị viên) sẽ luôn đăng tải các tài liệu điện tử mới lên hệ thống. Mỗi khi admin đăng tải bất kỳ tài liệu mới nào, hệ thống sẽ lưu thông tin tài liệu đó. Mỗi tài liệu sẽ gồm có mã riêng, loại tài liệu, mã tác giả, hình ảnh, tiêu đề tài liệu, nội dung tóm tắt tài liệu, trạng thái, ngày đăng, ngày cập nhật. Thông tin ngày cập nhật của tài liệu sẽ được cập nhật mỗi khi admin điều chỉnh thông tin của tài liệu đó.

Người dùng (bao gồm cả người dùng chưa có tài khoản và đã có tài khoản) được phép tìm kiếm tài liệu điện tử và xem trước tài liệu điện tử đó. Tính năng xem trước nhằm mục đích cho phép người dùng xem trước một đoạn đầu của tài liệu điện tử mà không được phép xem toàn bộ tài liệu. Để thuận tiện cho việc quản lý, hệ thống sẽ lưu thông tin của người dùng. Mỗi người dùng sẽ gồm có mã riêng, ngày tháng năm sinh, tên, số điện thoại, giới tính, địa chỉ, tài khoản email và mã tài khoản. Thông tin mã tài khoản chỉ có khi người dùng đã đăng ký tài khoản.

Mỗi người dùng sẽ có một tài khoản và chỉ một. Dựa vào thông tin đăng ký của người dùng, người quản lý sẽ cấp các quyền tương ứng để thực hiện các chức năng dựa vào quyền của mình (gồm quyền người quản lý và người dùng có tài khoản). Một tài khoản có thể có 1 hoặc nhiều quyền.

Nếu như người dùng không có tài khoản thì họ không thể tải được tài liệu điện tử của hệ thống, chỉ có thể tìm kiếm và xem trước tài liệu. Họ bắt buộc phải đăng nhập hoặc đăng ký tài khoản mới.

Người dùng có tài khoản có thể tải bất kỳ tài liệu điện tử nào họ muốn. Mỗi tài khoản có thể tải nhiều tài liệu điện tử khác nhau và có thể tải một tài liệu nhiều lần không giới hạn thời gian và số lượng. Sau khi tải, hệ thống sẽ lưu lại lịch sử tải của tài khoản. Dựa vào lịch sử tải, hệ thống sẽ đề xuất thông minh các tài liệu điện tử liên quan đến các lĩnh vực tài khoản đó hay quan tâm và thường xuyên tải. Từ đó, tối ưu thời gian tìm kiếm tài liệu điện tử và giúp tài khoản đó có nhiều sự lựa chọn để chọn nguồn tham khảo tài liệu. Ngoài ra, hệ thống thông minh này còn dựa vào thông tin của người dùng có tài khoản (độ tuổi, giới tính,...) để đề xuất.

Sau một thời gian người dùng sử dụng tài liệu điện tử mà họ đã tải, họ có thể quay lại để đánh giá tài liệu đó có thực sự có ích hay không. Một tài khoản có thể đánh giá nhiều tài liệu điện tử khác nhau và đối với mỗi tài liệu điện tử, tài khoản đó chỉ có thể cho một điểm số và họ vẫn có thể thay đổi điểm số đó nếu như họ muốn.

Người quản trị viên có chức năng quản lý các tài khoản của người dùng. Họ có thể tạo tài khoản mới, xóa tài khoản, chỉnh sửa thông tin tài khoản, phân quyền tài khoản, xem thông tin các tài khoản của người dùng.

Người dùng có tài khoản có thể đăng tải tài liệu điện tử mà họ muốn. Khi đăng tải, trạng thái của tài liệu điện tử đó sẽ ở trạng thái chờ duyệt. Sau khi được

quản trị viên phê duyệt, tài liệu điện tử đó sẽ ở trạng thái hiển thị, khi đó, người dùng có tài khoản khác có quyền tải tài liệu điện tử này. Người dùng có tài khoản có thể xóa, chỉnh sửa các tài liệu điện tử mà họ đăng tải nhưng vẫn phải được phê duyệt bởi quản trị viên.

**b. Đối tượng sử dụng hệ thống và các chức năng của từng đối tượng:**

**Đối tượng gồm:**

- Admin (người quản trị viên)
- Người dùng khách (người dùng chưa có tài khoản)
- Người dùng có tài khoản

**Các chức năng của từng đối tượng:**

❖ Admin:

- Đăng tải các tài liệu điện tử mới.
- Cập nhật các tài liệu điện tử.
- Đọc thông tin cá nhân của người dùng.
- Tạo và xóa tài khoản.
- Phân quyền tài khoản.
- Chỉnh sửa thông tin cá nhân của người dùng.
- Chỉnh sửa tài khoản của người dùng.
- Phê duyệt các tài liệu điện tử của người dùng có tài khoản đăng tải.

❖ Người dùng khách:

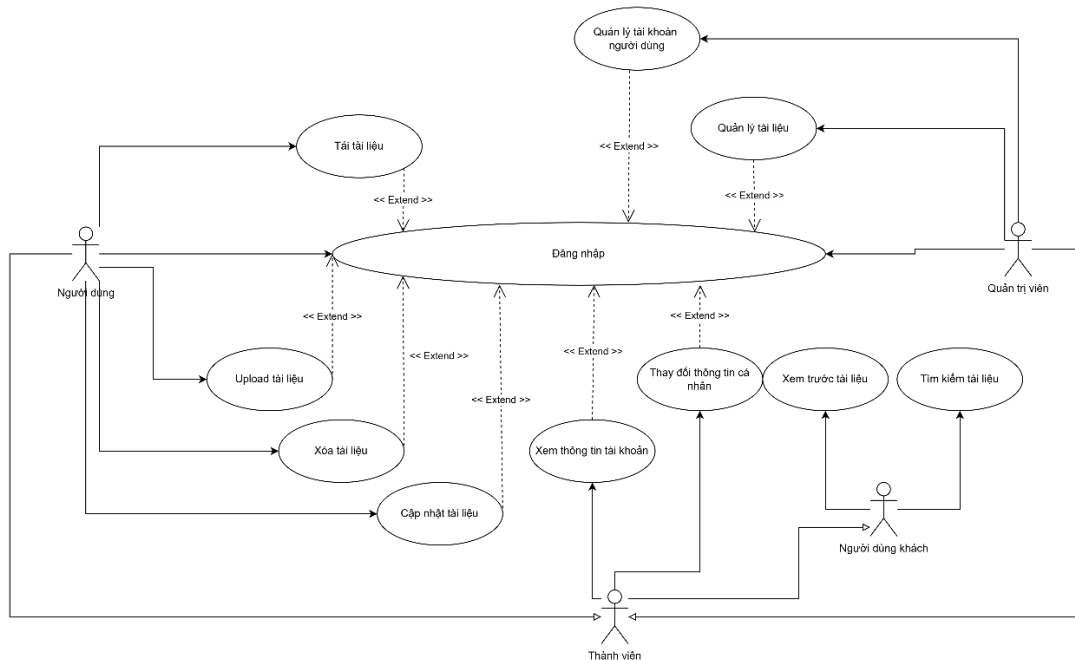
- Tìm kiếm tài liệu điện tử.
- Xem trước tài liệu điện tử.

❖ Người dùng có tài khoản:

- Tìm kiếm tài liệu điện tử.
- Xem trước tài liệu điện tử.
- Tải tài liệu điện tử.
- Chỉnh sửa thông tin cá nhân.
- Đổi mật khẩu.
- Được hệ thống đề xuất thông minh các tài liệu điện tử.
- Đăng tải các tài liệu điện tử
- Xóa và chỉnh sửa các tài liệu điện tử của mình đăng tải.

**2. Mô hình Usecase:**

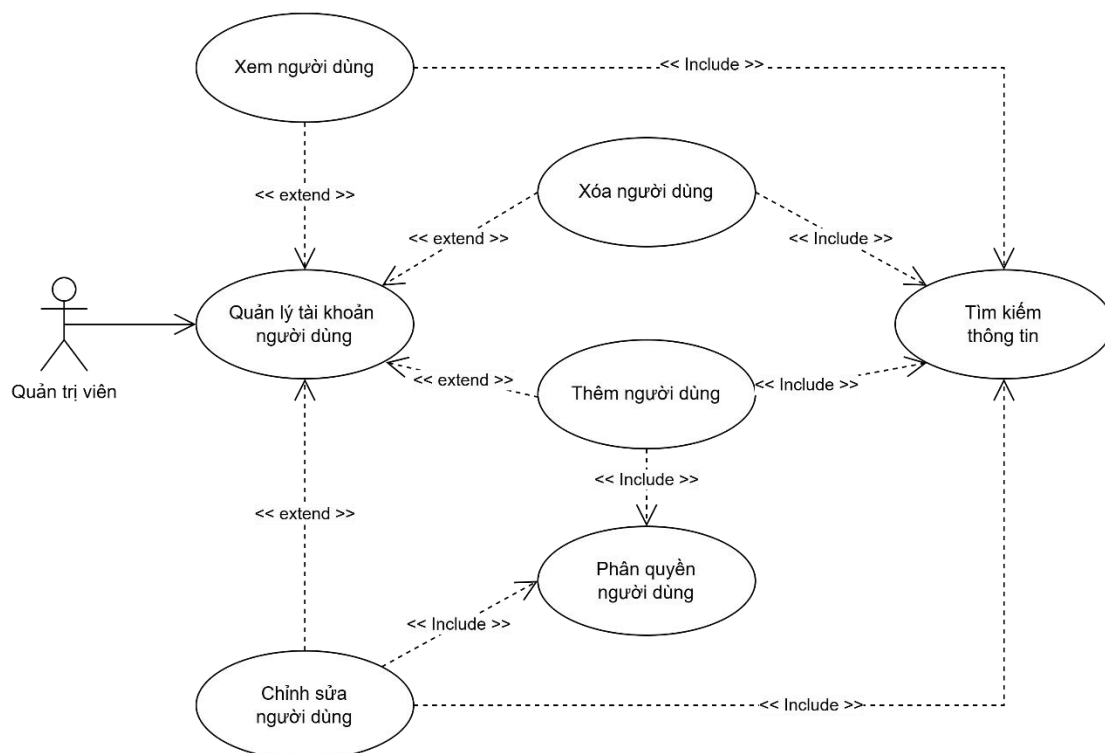
**a. Biểu đồ ca sử dụng mức tổng quát:**



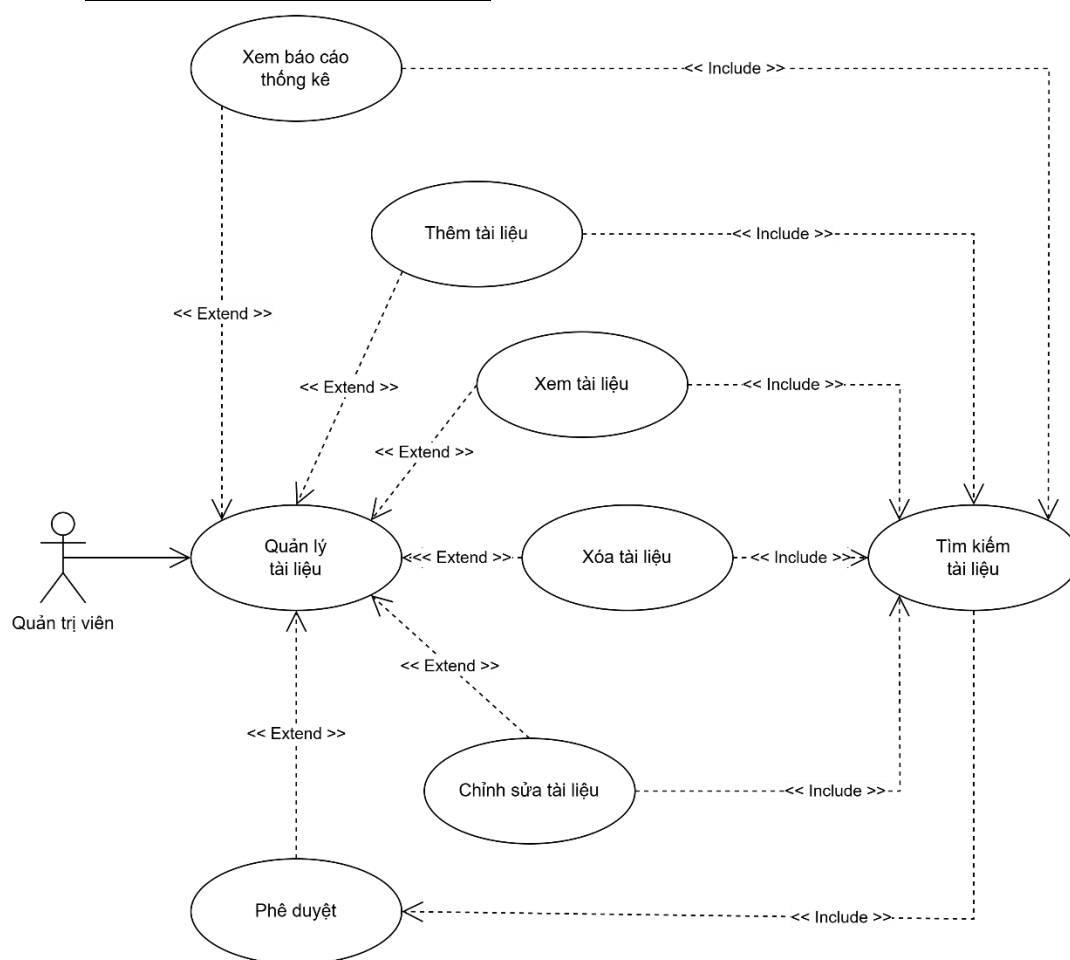
Hình 3: Biểu đồ ca sử dụng mức tổng quát

**b. Biểu đồ phân rã ca sử dụng:**

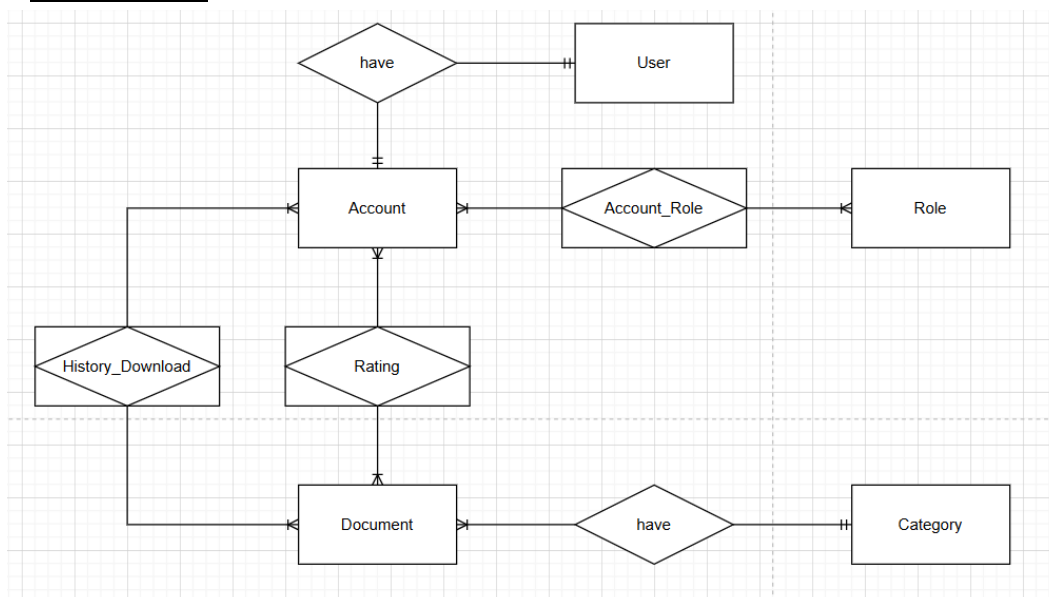
**Quản trị viên quản lý tài khoản người dùng:**



Hình 4: Quản trị viên quản lý tài khoản người dùng

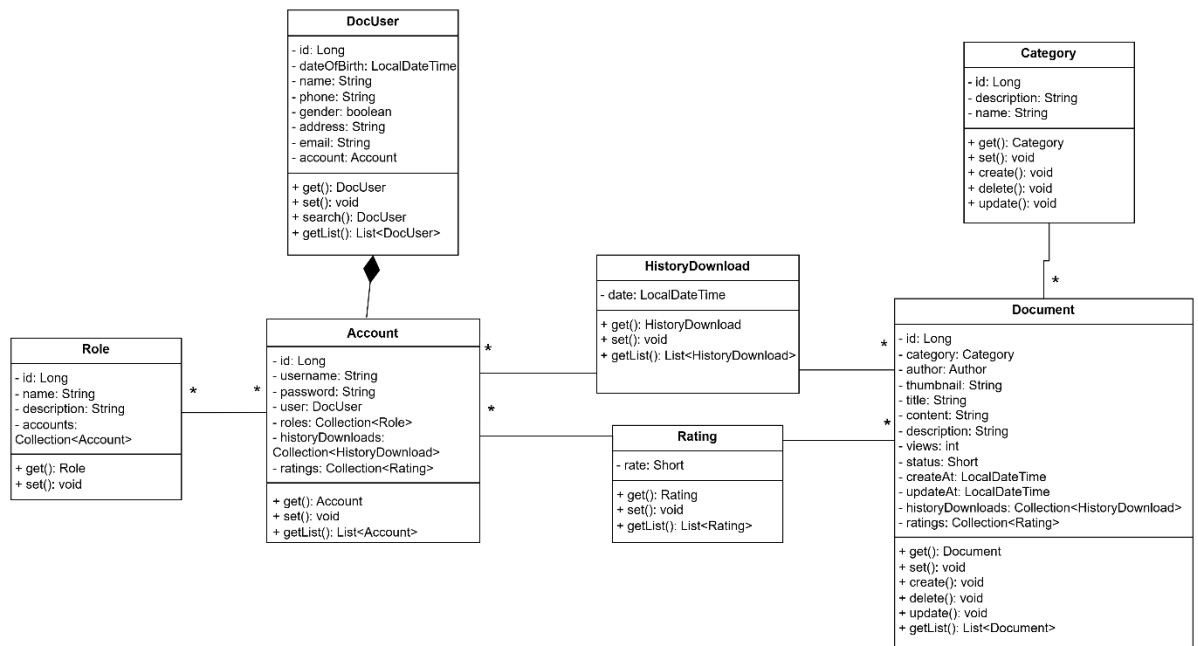
**Quản trị viên quản lý tài liệu:**

Hình 5: Quản trị viên quản lý tài liệu

**3. Sơ đồ ERD:**

Hình 6: Sơ đồ ERD

#### 4. Sơ đồ lớp:



Hình 7: Sơ đồ lớp.

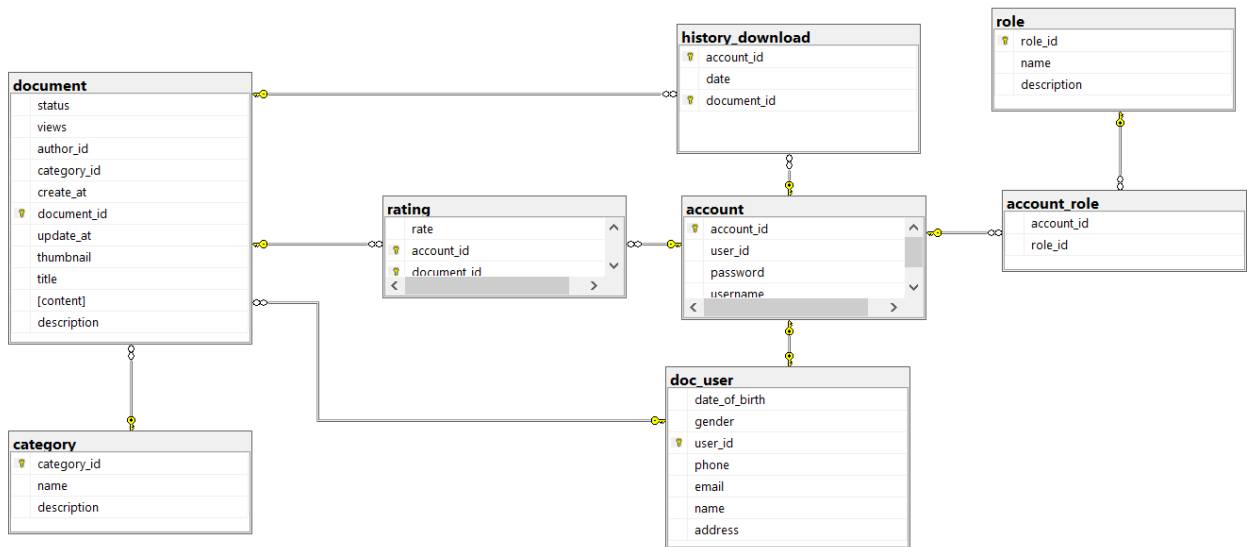
#### 5. Mô hình dữ liệu quan hệ từ ERD:

*Ghi chú:* Khóa chính: **role\_id** (được bôi đậm)

- doc\_user (**user\_id**, dateOfBirth, name, phone, gender, address, email)
- account (**account\_id**, username, password, user\_id)
- account\_Role (**role\_id**, **account\_id**)
- role (**role\_id**, name, description)
- document (**document\_id**, category\_id, author\_id, thumbnail, title, content, status, create\_at, update\_at, views, pages)
- category (**category\_id**, description, name)
- rating (**account\_id**, **document\_id**, rate)
- history\_Download (**document\_id**, **account\_id**, date)



## 6. Sơ đồ diagram:



Hình 8: Sơ đồ Diagram.

## 7. Từ điển dữ liệu:

a. Bảng doc\_user:

doc\_user (**user\_id**, dateOfBirth, name, phone, gender, address, email)

Thuộc tính	Kiểu dữ liệu	Độ dài	Ràng buộc	Ghi chú
user_id	bigint		Primary Key	Mã người dùng
dateOfBirth	date		Not Null	Ngày tháng năm sinh
name	nvarchar	255	Not Null	Tên người dùng
phone	varchar	15		Số điện thoại
gender	bit		Not Null	Giới tính
address	text			Địa chỉ
email	varchar	100		Địa chỉ email

b. Bảng account:

account (**account\_id**, username, password, user\_id)

Thuộc tính	Kiểu dữ liệu	Độ dài	Ràng buộc	Ghi chú
account_id	bigint		Primary Key	Mã tài khoản
username	varchar	255	Unique, Not Null	Tên tài khoản
password	varchar	255	Not Null	Mật khẩu
user_id	bigint		Foreign Key	Mã người dùng

c. Bảng account\_Role:

account\_Role (**role\_id**, **account\_id**)

Thuộc tính	Kiểu dữ liệu	Độ dài	Ràng buộc	Ghi chú
role_id	bigint		Primary Key, Foreign Key	Mã vai trò
account_id	bigint		Primary Key, Foreign Key	Mã tài khoản

d. Bảng role:

Role (**role\_id**, name, description)

Thuộc tính	Kiểu dữ liệu	Độ dài	Ràng buộc	Ghi chú
role_id	bigint		Primary Key	Mã vai trò
name	nvarchar	255	Unique, Not Null	Tên vai trò
description	text			Mô tả về vai trò

e. Bảng document:

document (**document\_id**, category\_id, author\_id, thumbnail, title, content, status, create\_at, update\_at, views, pages)

Thuộc tính	Kiểu dữ liệu	Độ dài	Ràng buộc	Ghi chú
document_id	bigint		Primary Key	Mã tài liệu điện tử
category_id	bigint		Foreign Key, Not Null	Mã loại tài liệu điện tử
author_id	bigint		Foreign Key, Not Null	Mã người đăng tải tài liệu điện tử
thumbnail	nvarchar	255	Not Null	Hình ảnh tài liệu
title	nvarchar	255	Not Null	Tiêu đề tài liệu
content	text			Tệp tài liệu
status	smallint		Not Null	Trạng thái (hiển thị, chờ duyệt, ẩn)
create_at	datetime		Not Null	Ngày giờ đăng tải tài liệu
update_at	datetime		Not Null	Ngày giờ cập nhật tài liệu
views	int		default = 0	Số lượt xem
pages	int		default = 0	Số trang của tài liệu

f. Bảng category:

category (**category\_id**, description, name)

Thuộc tính	Kiểu dữ liệu	Độ dài	Ràng buộc	Ghi chú
category_id	bigint		Primary Key	Mã loại tài liệu
description	text			Mô tả về loại tài liệu
name	nvarchar	255	Not Null	Tên loại tài liệu

g. Bảng rating:

rating (**account\_id**, **document\_id**, rate)

Thuộc tính	Kiểu dữ liệu	Độ dài	Ràng buộc	Ghi chú
account_id	bigint		Primary Key, Foreign Key	Mã tài khoản
document_id	bigint		Primary Key, Foreign Key	Mã tài liệu điện tử
rate	smallint		Not Null	Điểm đánh giá

h. Bảng history\_Download:

history\_Download (**document\_id**, **account\_id**, date)

Thuộc tính	Kiểu dữ liệu	Độ dài	Ràng buộc	Ghi chú
document_id	bigint		Primary Key, Foreign Key	Mã tài liệu điện tử
account_id	bigint		Primary Key, Foreign Key	Mã tài khoản
date	datetime		Not Null	Ngày giờ tải tài liệu

## IV- THIẾT KẾ THÔNG MINH:

### 1. Hệ thống đề xuất thông minh dựa vào người dùng có tài khoản:

#### a. Mô tả hệ thống:

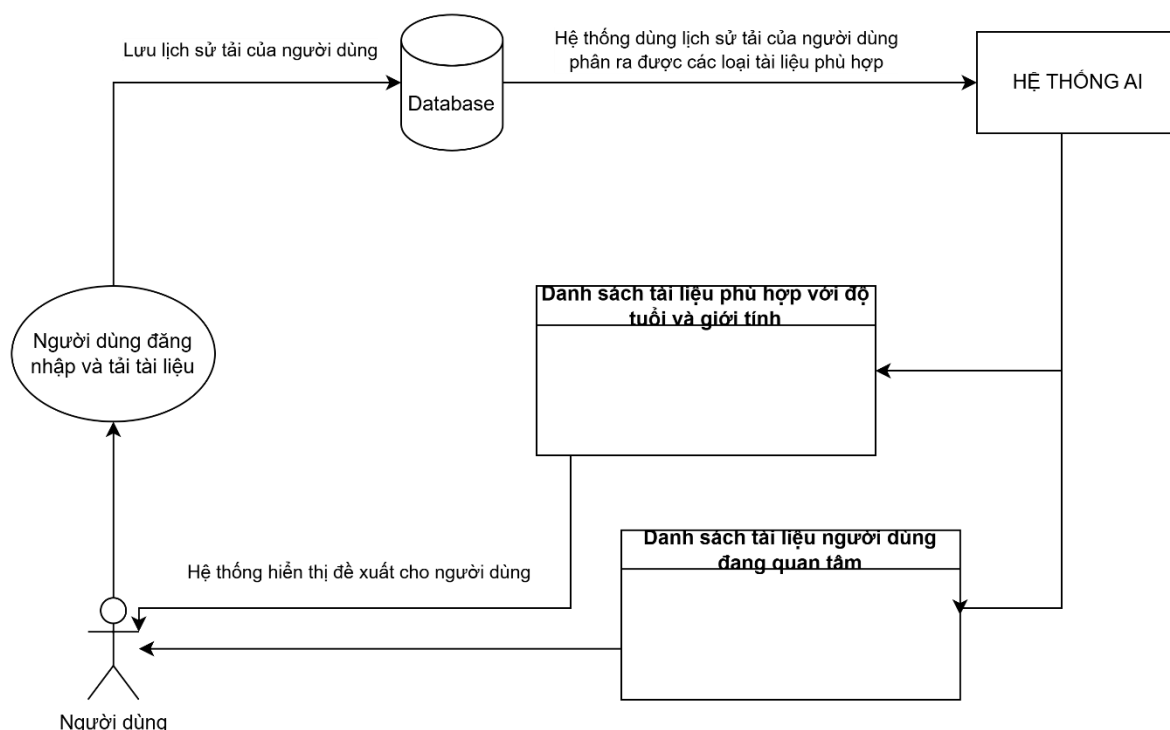
Hệ thống đề xuất thông minh dựa vào lịch sử tải tài liệu điện tử của người dùng có tài khoản là một hệ thống sử dụng lịch sử tải tài liệu, thông tin (giới tính, độ tuổi) của người dùng để phân tích sở thích, từ đó gợi ý các tài liệu liên quan hoặc được quan tâm. Dựa trên thuật toán máy học, hệ thống liên tục cải thiện độ chính xác, mang đến trải nghiệm cá nhân hóa, tiết kiệm thời gian tìm kiếm cho người dùng.

Ví dụ, dựa vào thông tin của người dùng, hệ thống biết được độ tuổi của người dùng này nằm trong độ tuổi 10 – 15 tuổi. Khi đó, hệ thống sẽ đề xuất các tài liệu liên quan đến các tài liệu học thuật phù hợp trong độ tuổi. Hoặc là biết được thói quen của một người dùng thường xuyên tải về các tài liệu điện tử liên quan đến khoa học, hệ thống sẽ đề xuất các tài liệu mới thuộc lĩnh vực đó mà được nhiều người dùng khác quan tâm.

#### b. Quy trình hoạt động:

- Bước 1: Hệ thống ghi nhận các thông tin của người dùng như độ tuổi, giới tính, và lịch sử tải tài liệu
- Bước 2: Phân tích dữ liệu: Áp dụng thuật toán K-Means.
- Bước 3: Gợi ý tài liệu cho người dùng thông qua giao diện
- Bước 4: Thu nhập phản hồi của người dùng (người dùng tải các tài liệu đó hoặc bỏ qua) để điều chỉnh thuật toán.

#### c. Mô hình hoạt động:



Hình 9: Mô hình hoạt động.

#### d. Dữ liệu đầu vào:


Thông tin người dùng (giới tính, độ tuổi), nhóm tài liệu (category)

e. **Xử lý dữ liệu:**

**Tiền xử lý dữ liệu:**

- Biến đổi dữ liệu chữ về dạng số nhằm hỗ trợ cho thuật toán K-means đọc dữ liệu số
- Các thuộc tính dùng để phân cụm bao gồm: Loại tài liệu (category), giới tính (gender), độ tuổi (age). Trong đó:
  - Nhóm tài liệu (Category): Đây là thuộc tính phân loại được chuyển đổi từ dạng chuỗi sang dạng số nguyên áp dụng phương pháp LabelEncoder,
  - Giới tính (Gender): Vì đây là thuộc tính dạng phân loại nhị phân, ta có thể mã hóa nó thành các giá trị số 0 và 1.
  - Độ tuổi (Age): sẽ được tính từ thuộc tính date\_of\_birth của doc\_user

**Kết nối với Database:**



```
1 SERVER = 'MSI'
2 DATABASE = 'HTTM'
3 USERNAME = 'sa'
4 PASSWORD = 'sa'
5
6 cnxn = pyodbc.connect('DRIVER={SQL Server};SERVER=' + SERVER + ';DATABASE=' +
7                       DATABASE + ';UID=' + USERNAME + ';PWD=' + PASSWORD)
8 cursor = cnxn.cursor()
9
```

Hình 10: Kết nối Database.

**Cấu hình K-Means:**

Sử dụng phương pháp Elbow để tìm ra K cụm:

```
1 def find_K(df):
2     distortions = []
3     max_clusters = min(10, len(df))
4     K = range(1, max_clusters + 1)
5
6     for k in K:
7         kmeanModel = KMeans(n_clusters=k,
8                               init='k-means++',
9                               max_iter=300,
10                              n_init=10,
11                              random_state=0)
12         kmeanModel.fit(df)
13         distortions.append(kmeanModel.inertia_)
14         print(f'Inertia for {k} clusters: {kmeanModel.inertia_}')
15
16     for i in range(1, len(distortions)):
17         if distortions[i - 1] != 0 and distortions[i] / distortions[i - 1] > 0.93:
18             return i + 1
19
20     # Nếu không có sự giảm nhanh, trả về số cụm lớn nhất
21     return max_clusters
22
```

Hình 11: Thuật toán tìm số lượng cluster K.

Huấn luyện model K-Means bằng data đã được xử lý và lưu lại model sử dụng module pickle:

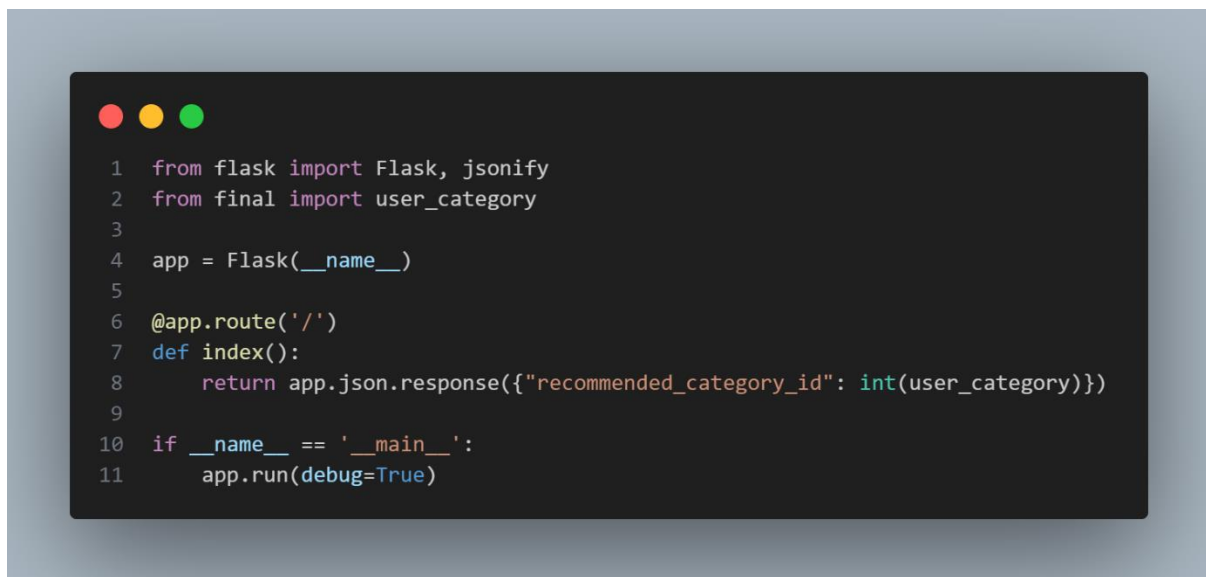
```
1 query = '''
2     SELECT
3     USERINF.gender AS gender,
4     YEAR(GETDATE()) - YEAR(USERINF.date_of_birth) AS age,
5     CategoryDocument.category_id AS category_id
6 FROM [dbo].[account]
7 INNER JOIN (
8     SELECT user_id, gender, date_of_birth
9     FROM [dbo].[doc_user]
10 ) AS USERINF ON USERINF.user_id = account.user_id
11 INNER JOIN (
12     SELECT
13         history_download.account_id,
14         document.category_id
15     FROM document
16     INNER JOIN history_download ON history_download.document_id = document.document_id
17 ) AS CategoryDocument ON CategoryDocument.account_id = account.account_id;
18 '''
19 df = pd.read_sql(query, cnxn)
20
21 scaler = StandardScaler()
22 scaled_data = scaler.fit_transform(df[['gender', 'age', 'category_id']])
23
24 kmeans = KMeans(n_clusters=find_K(df), random_state=42)
25
26 df['cluster'] = kmeans.fit_predict(scaled_data)
27
28 user_data = np.array([[1, 27]])
29 nn = NearestNeighbors(n_neighbors=1)
30 nn.fit(df[['gender', 'age']])
31 _, indices = nn.kneighbors(user_data)
32
33 user_cluster = df.iloc[indices[0][0]]['cluster']
34 user_category = df.iloc[indices[0][0]]['category_id']
35 print(f"User belongs to cluster {user_cluster} and suggested category is {user_category}")
36
37 cluster_summary = df.groupby('cluster').mean()
38 print(cluster_summary)
```

Hình 12: Tìm các cụm và giá trị các cụm. (Đoạn code trên lấy ví dụ người dùng giới tính nữ (1) và độ tuổi là 27)

Tải thư viện flask và chạy đoạn code dưới để nhận được kết quả đề xuất:

pip install flask

python server/server.py

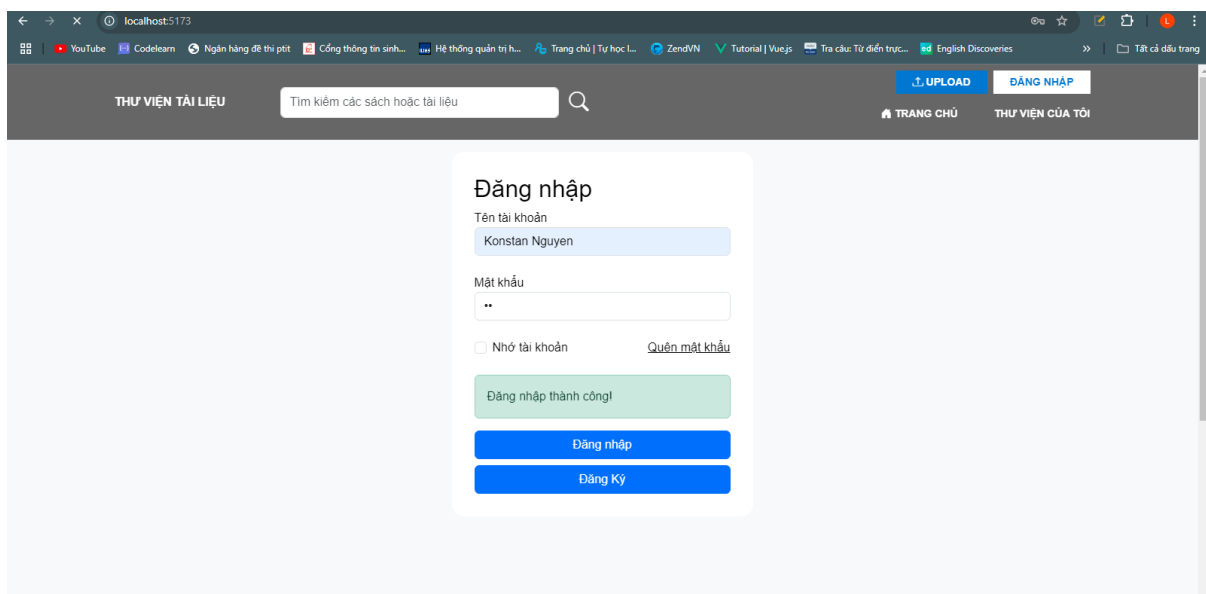


Hình 13: Kết quả của hệ thống.

## 2. Thiết kế giao diện:

### a. Giao diện chính:

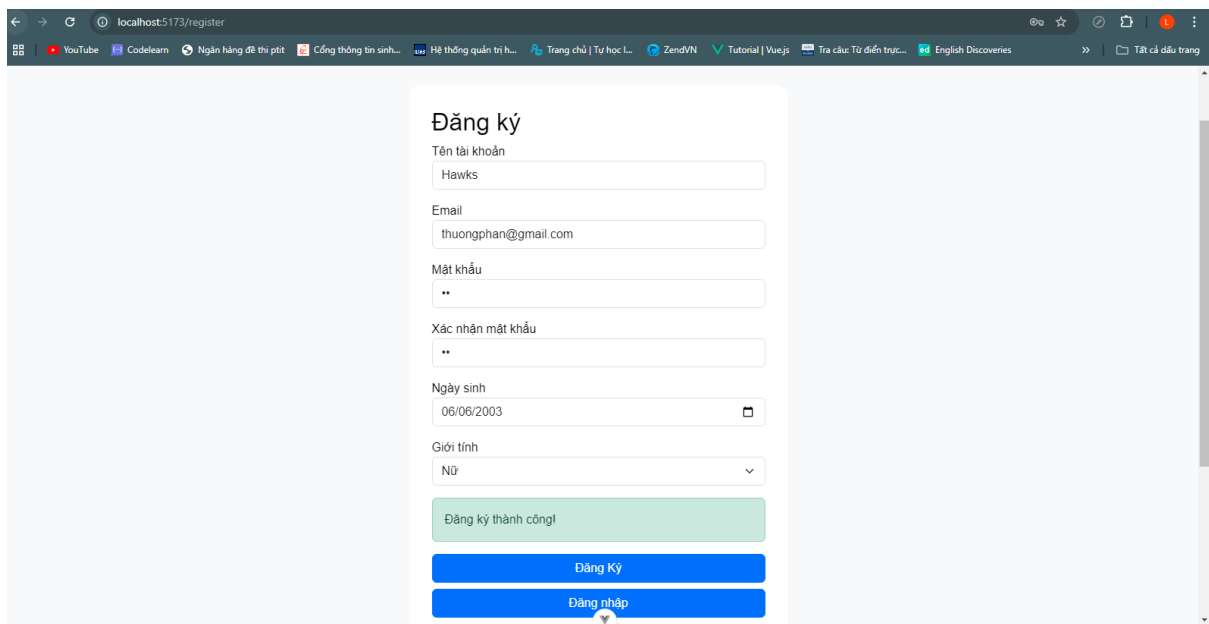
Giao diện đăng nhập



Hình 14: Giao diện đăng nhập.

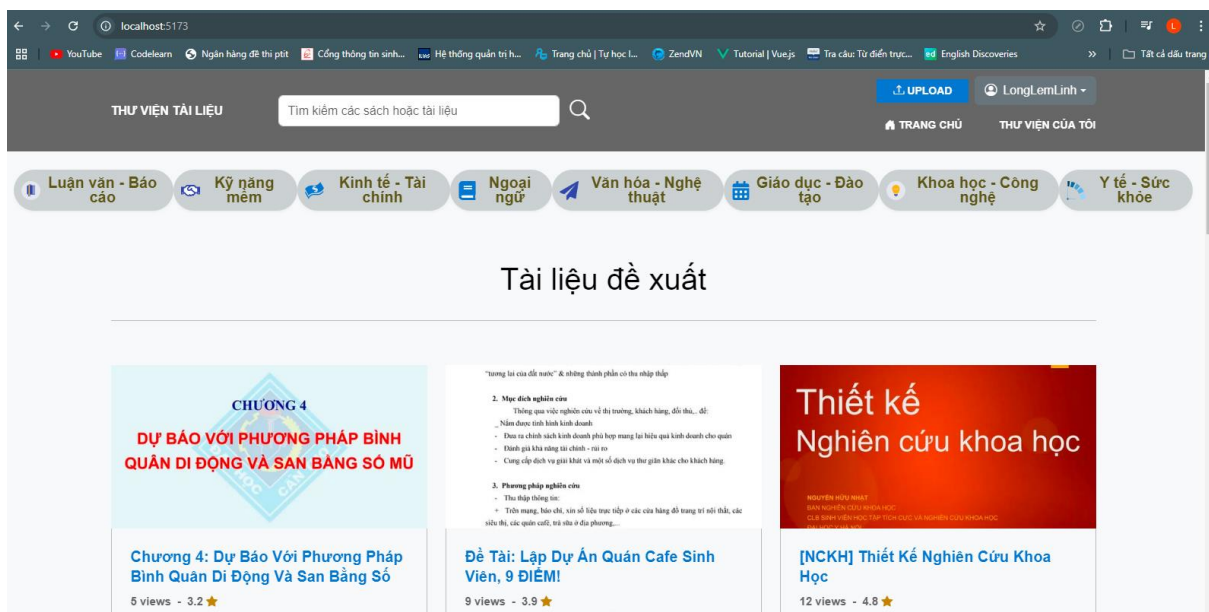
Giao diện đăng ký



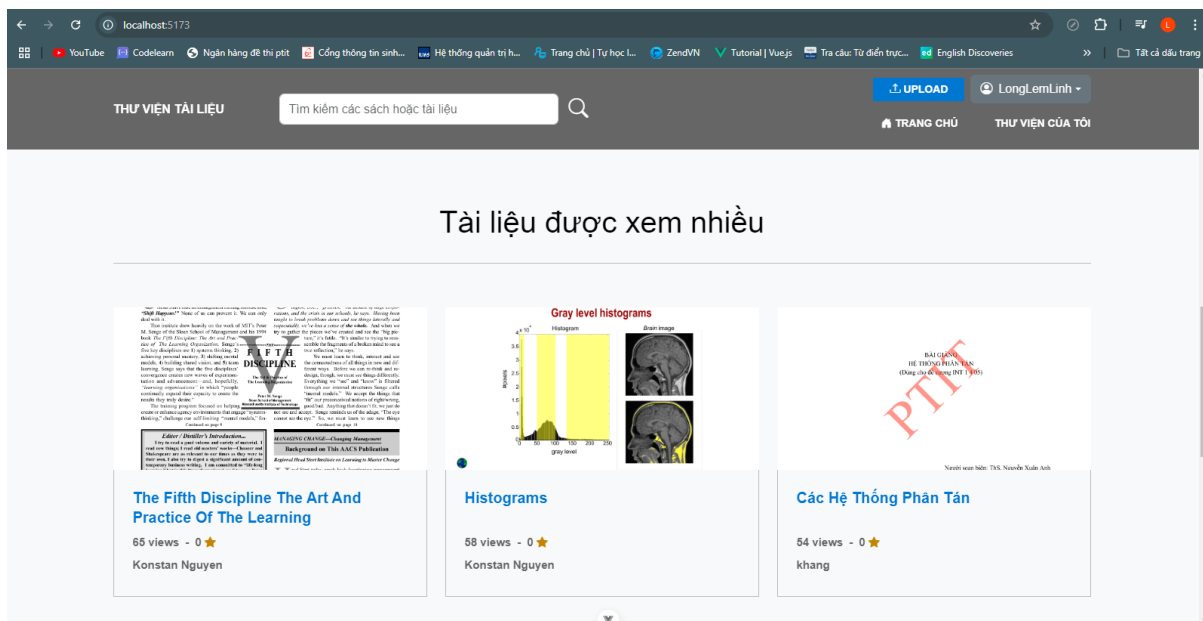


Hình 15: Giao diện đăng ký.

Giao diện trang chủ:



Hình 16: Giao diện trang chủ đầu tiên.



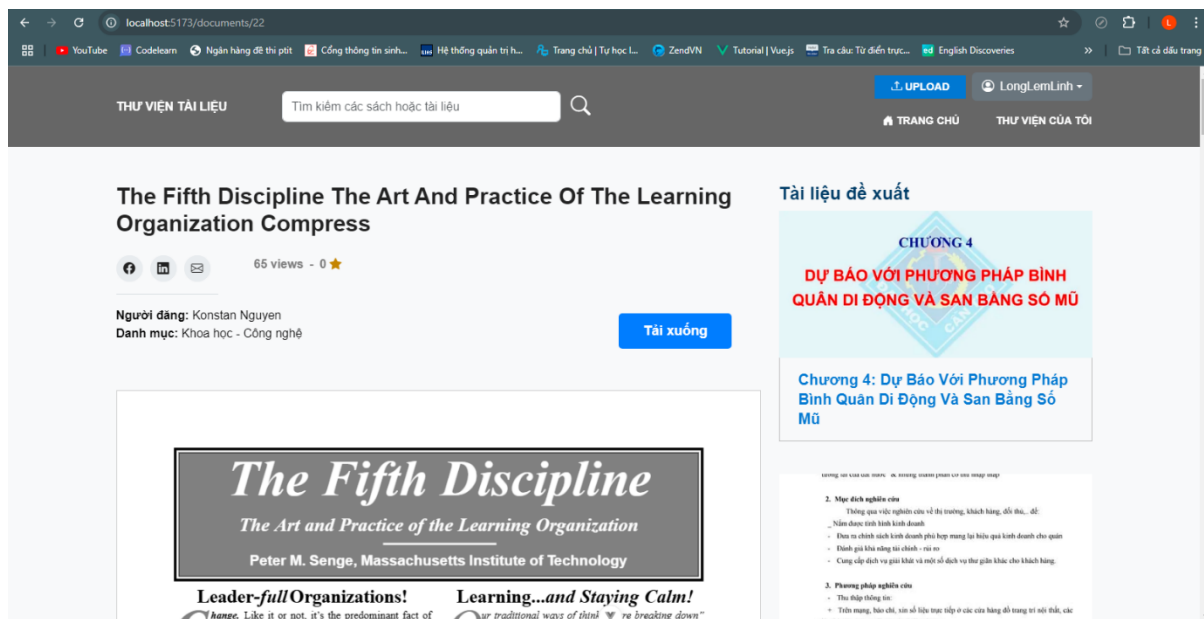
Hình 17: Giao diện trang chủ tiếp theo.

Giao diện khi chọn mục thể loại:



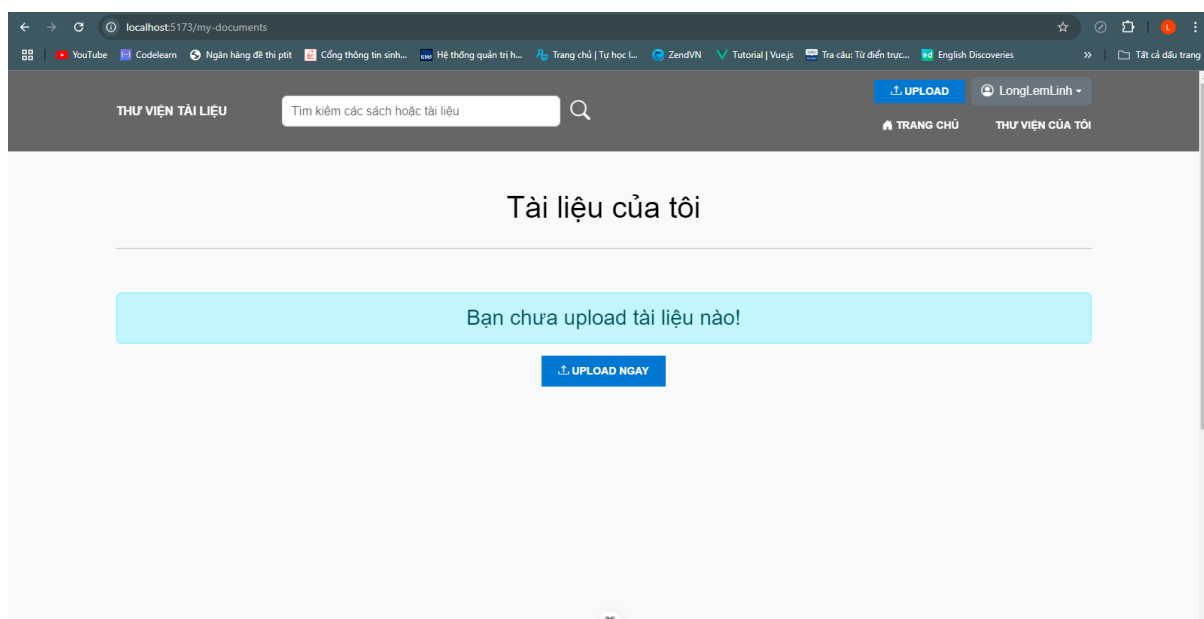
Hình 18: Giao diện các tài liệu thuộc cùng thể loại.

Giao diện chi tiết một tài liệu:

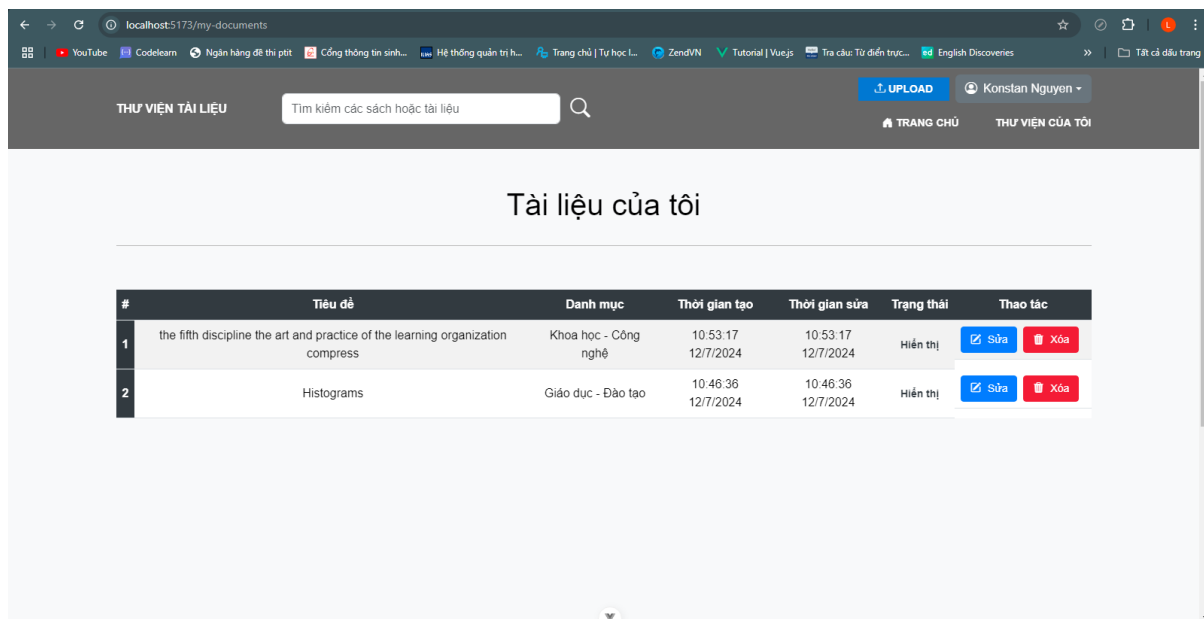


Hình 19: Giao diện chi tiết một tài liệu điện tử.

Giao diện các tài liệu thuộc sở hữu của người:



Hình 20: Giao diện danh sách tài liệu của người dùng khi chưa có tài liệu đăng tải.



Hình 21: Giao diện danh sách tài liệu của người dùng sau khi đã có tài liệu đăng tải.

Giao diện khi đăng tải một tài liệu mới bởi người dùng:

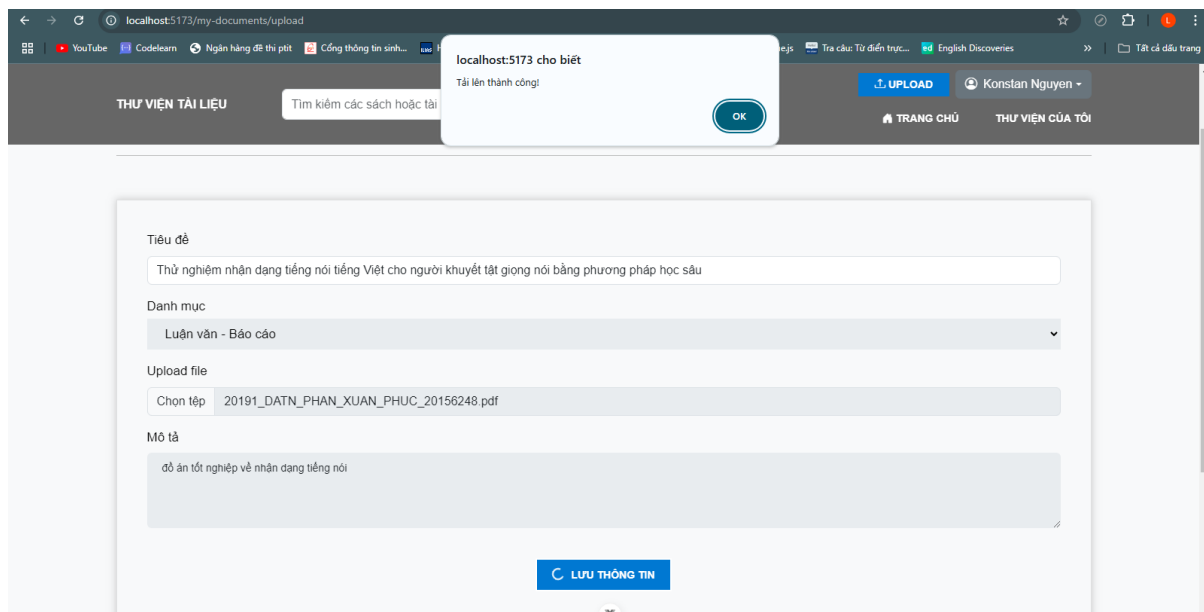


Figure 22: Người dùng đăng tải tài liệu mới.

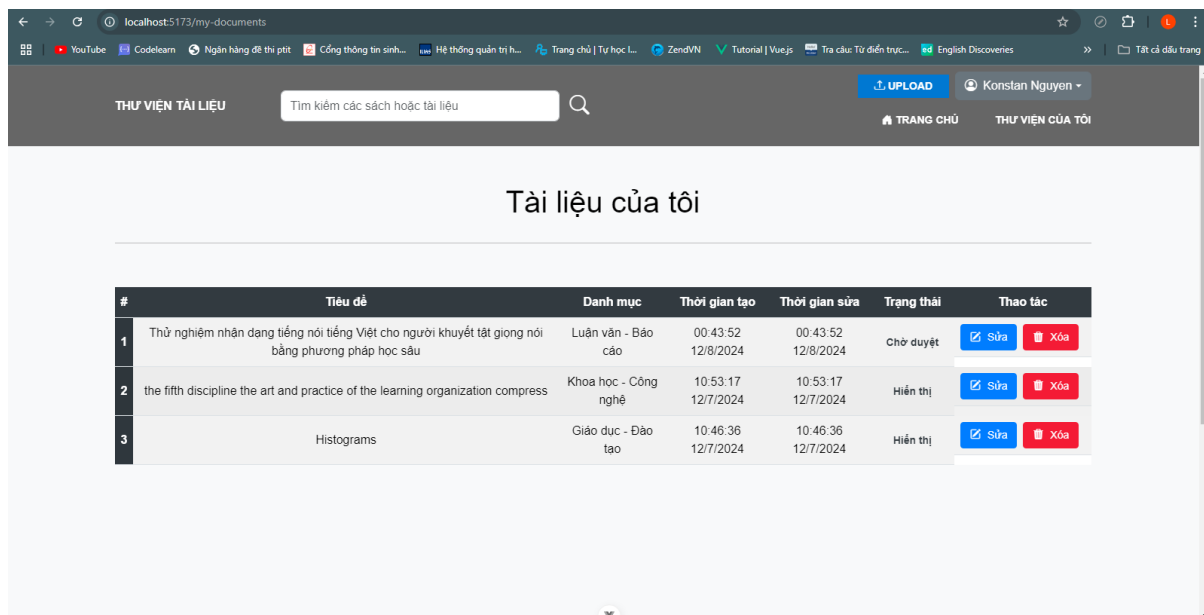
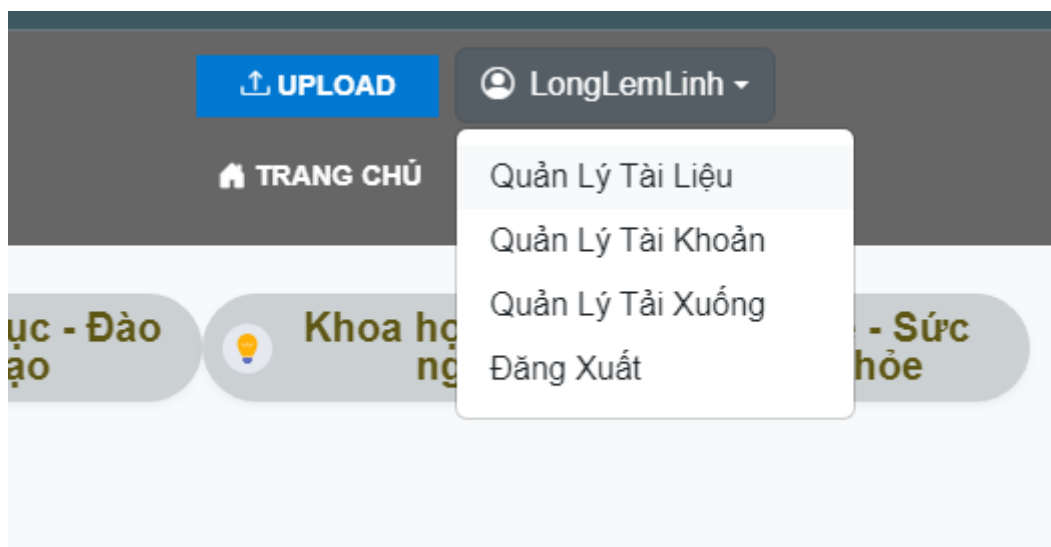


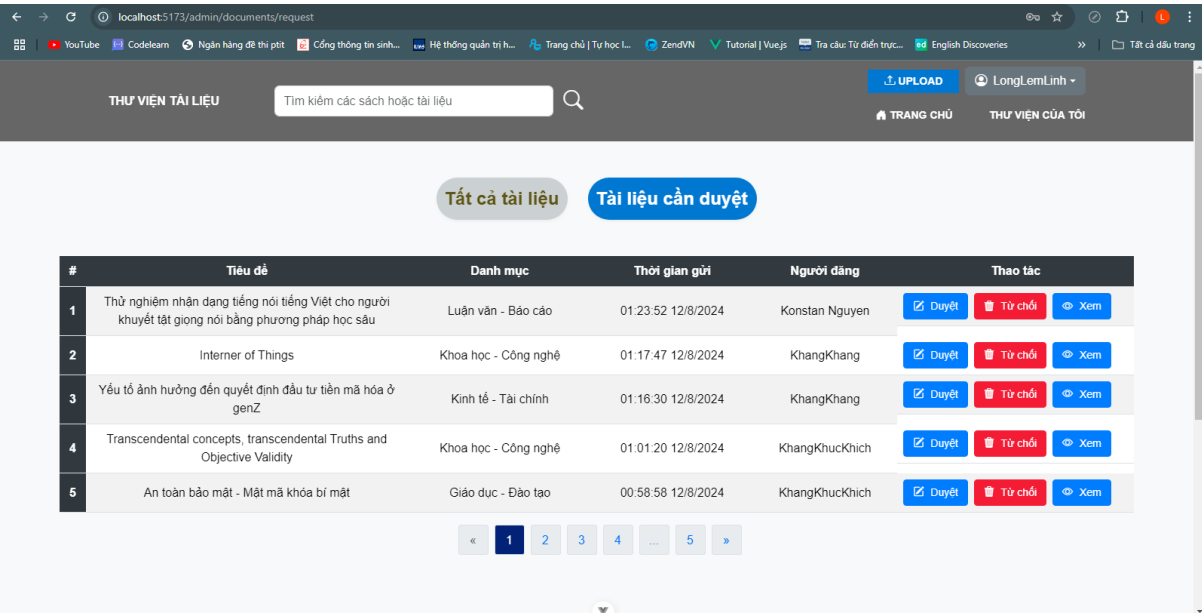
Figure 23: Tài liệu được để vào trạng thái chờ duyệt (được hiển thị trong danh sách các tài liệu đã đăng tải của người dùng)

Giao diện các chức năng của người quản trị:



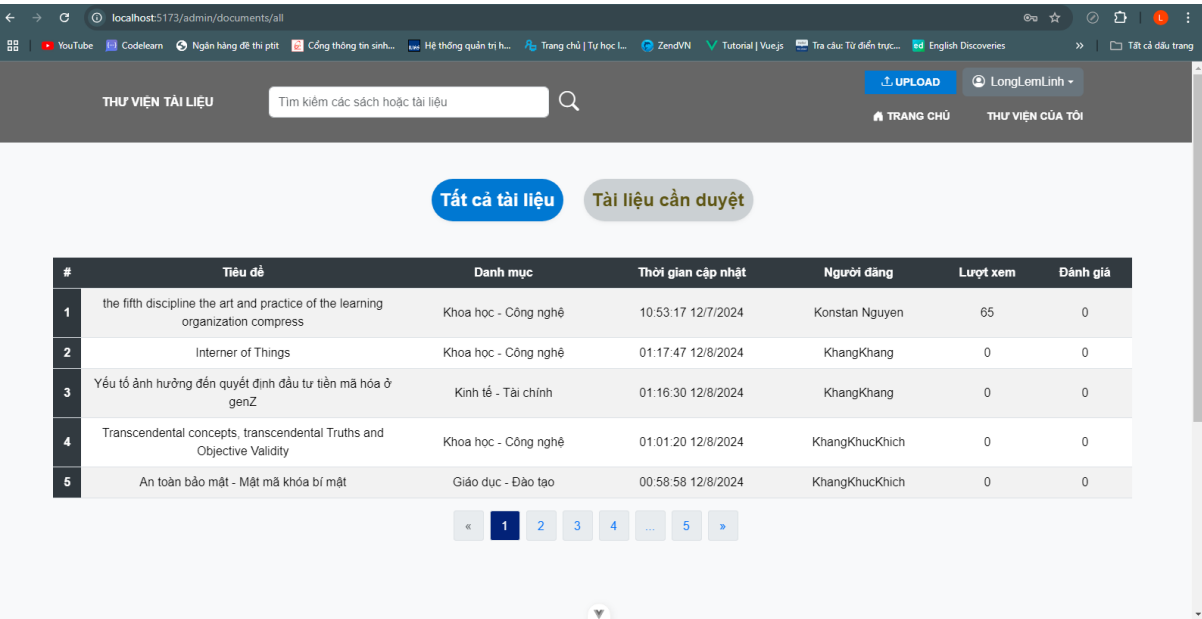
Hình 24: Giao diện các nút chức năng của tài khoản có quyền quản trị viên.

Giao diện quản lý tài liệu cần duyệt: sau khi quản trị viên duyệt tài liệu, tài liệu đó sẽ được hiển thị trên giao diện “Tài liệu của hệ thống”.



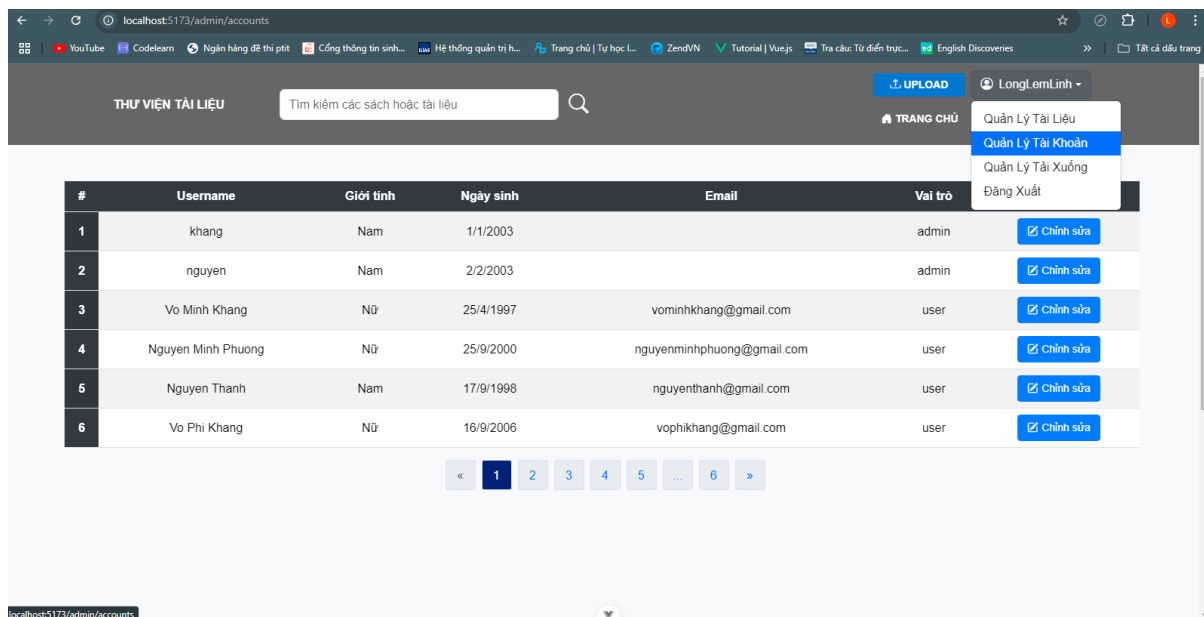
Hình 25: Giao diện danh sách các tài liệu cần duyệt.

Giao diện tất cả tài liệu của hệ thống:



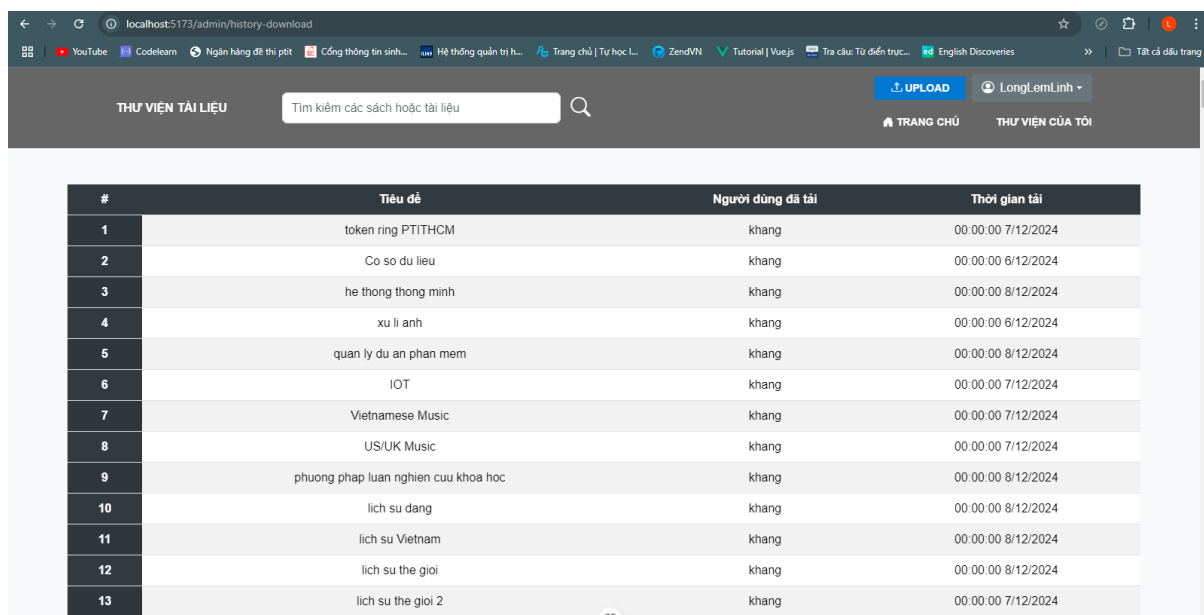
Hình 26: Giao diện tất cả tài liệu của hệ thống.

Giao diện quản lý tài khoản của quản trị viên:



Hình 27: Giao diện quản lý các tài khoản.

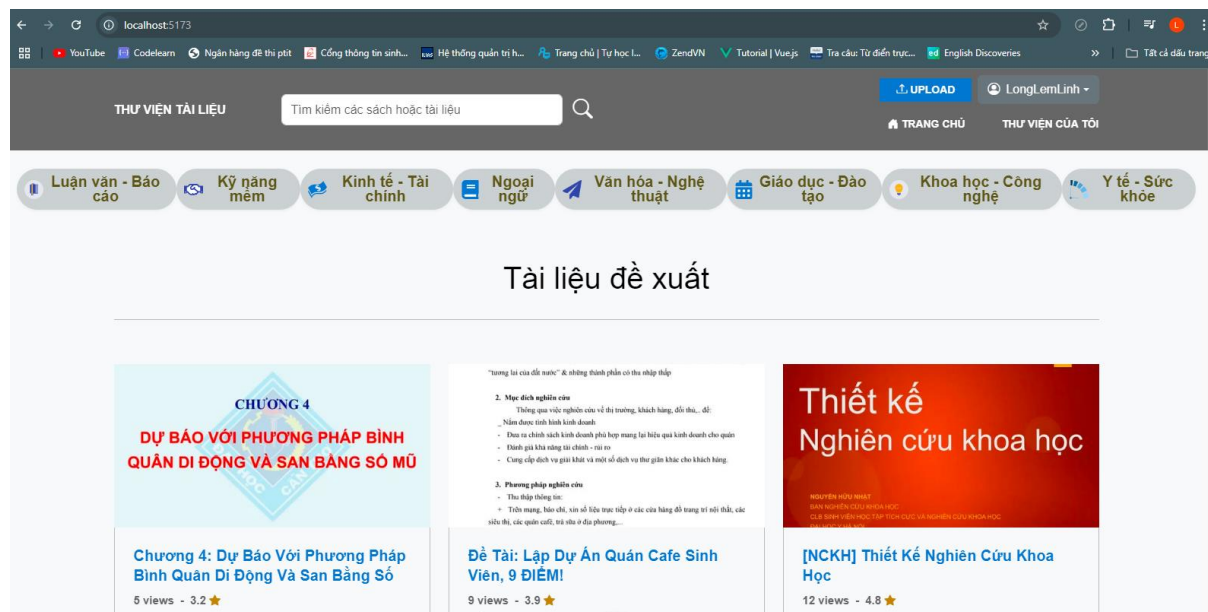
Giao diện quản lý lịch sử tải của các người dùng:



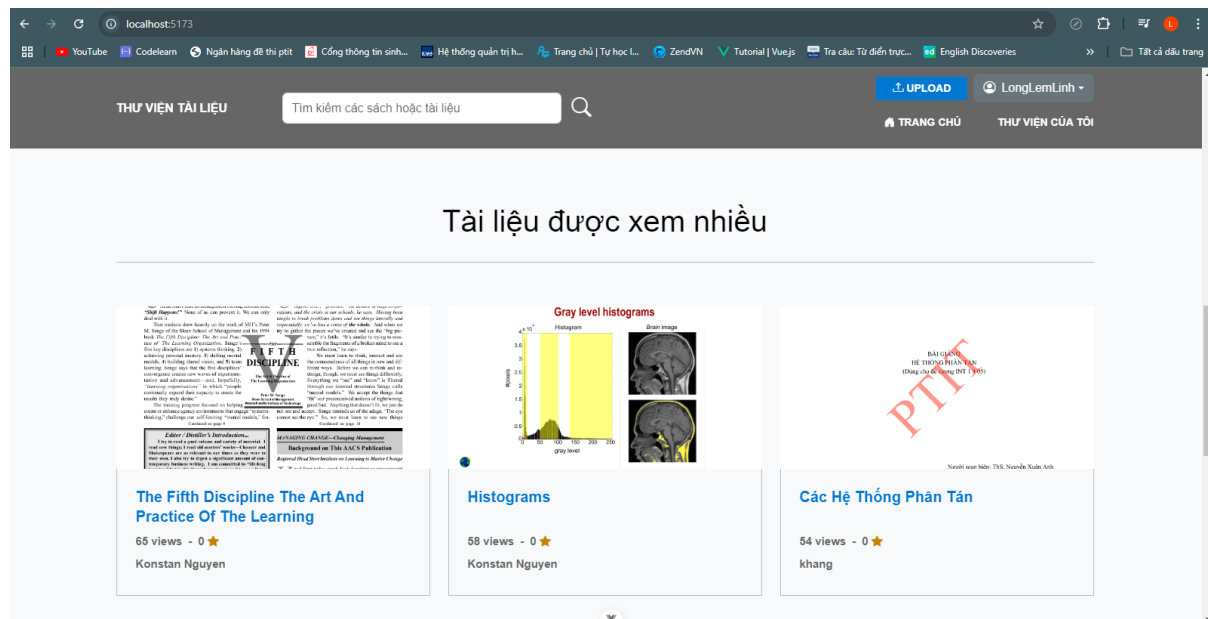
Hình 28: Giao diện lịch sử tải của các người dùng.

## b. Giao diện thông minh:

Giao diện đề xuất thông minh:



Hình 29: Giao diện tài liệu đề xuất.



Hình 30: Giao diện các tài liệu được xem nhiều.



## V- KẾT QUẢ THỰC NGHIỆM THỬ NGHIỆM:

### 1. Phân cụm dữ liệu bằng K-Means:

Khi người dùng mới truy cập chưa có lịch sử tải thì thuật toán K-Means được áp dụng với việc lấy dữ liệu từ CSDL bao gồm **các tài liệu điện tử đã được người dùng khác tải**, sau đó, tiến hành áp dụng Elbow, thuật toán đã chia dữ liệu thành các cụm phù hợp và dựa vào thông tin của người dùng sẽ phân người dùng vào cụm thích hợp. Từ đó, đề xuất cho người dùng thể loại phù hợp với độ tuổi và giới tính người dùng. Ngoài ra, khi người dùng đã có lịch sử tải, hệ thống sẽ dựa vào 10 tài liệu mà họ đã tải gần đây nhất để đề xuất cho người dùng nhất tài liệu cùng chủ đề họ quan tâm.

Hệ thống đề xuất có thu thập dữ liệu của người dùng bao gồm độ tuổi, giới tính và lịch sử tải của người dùng. Ngoài ra, khi người dùng đã có lịch sử tải tài liệu, hệ thống sẽ dùng thêm dữ liệu từ lịch sử tải của người dùng kết hợp với dữ liệu đề xuất dựa vào độ tuổi và giới tính, từ đó đề xuất cho người dùng những tài liệu vừa phù hợp với độ tuổi, giới tính, vừa phù hợp với xu hướng quan tâm của người dùng.

### 2. Đánh giá kết quả phân cụm:

Bằng thuật toán tìm số lượng K-Means phù hợp đã đề cập ở phần tiền xử lý dữ liệu, thuật toán đã tìm ra được số lượng cluster K phù hợp để có thể áp dụng vào thuật toán K-Means.



```
1 def find_K(df):
2     distortions = []
3     max_clusters = min(10, len(df))
4     K = range(1, max_clusters + 1)
5
6     for k in K:
7         kmeanModel = KMeans(n_clusters=k,
8                             init='k-means++',
9                             max_iter=300,
10                            n_init=10,
11                            random_state=0)
12         kmeanModel.fit(df)
13         distortions.append(kmeanModel.inertia_)
14         print(f'Inertia for {k} clusters: {kmeanModel.inertia_}')
15
16     for i in range(1, len(distortions)):
17         if distortions[i - 1] != 0 and distortions[i] / distortions[i - 1] > 0.93:
18             return i + 1
19
20     # Nếu không có sự giảm nhanh, trả về số cụm lớn nhất
21     return max_clusters
22
```

Hình 31: Thuật toán tìm số lượng cụm

```
Inertia for 1 clusters: 8090.150000000003
Inertia for 2 clusters: 1496.0100072969865
Inertia for 3 clusters: 763.3934656741112
Inertia for 4 clusters: 526.1486096063566
Inertia for 5 clusters: 391.40340406719724
Inertia for 6 clusters: 333.6345160924267
Inertia for 7 clusters: 284.25098896660546
Inertia for 8 clusters: 263.6089012014893
Inertia for 9 clusters: 246.0374916681719
Inertia for 10 clusters: 227.32869958762592
```

Hình 32: Các giá trị ứng với mỗi cluster.

**Giải thích:**

**Inertia trong K-Means:**

- Inertia là tổng bình phương khoảng cách từ các điểm dữ liệu đến tâm cụm gần nhất.
- Khi số cụm tăng, Inertia giảm do mỗi cụm mới làm giảm khoảng cách trung bình của các điểm đến tâm cụm gần nhất.

**Ý nghĩa kết quả:**

- Inertia for 1 clusters: 8090.15: Toàn bộ dữ liệu được gộp vào một cụm, nên tổng khoảng cách là rất lớn.
- Inertia for 2 clusters: 1490.01: Khi chia thành 2 cụm, khoảng cách giảm đáng kể.
- Sau đó, sự giảm dần của Inertia trở nên ít rõ rệt hơn (đặc trưng của K-Means).

Nếu bạn đã có các giá trị cluster từ kết quả K-means, mỗi người dùng sẽ được phân vào một trong các cụm (cluster) dựa trên độ tuổi và giới tính. Ví dụ, nếu bạn có một người dùng với giới tính và độ tuổi nhất định, bạn có thể so sánh với các cụm đã phân để xác định họ thuộc vào cụm nào.

Giả sử bạn có một người dùng mới với độ tuổi 27 và giới tính 1(giới tính nữ), bạn sẽ tìm ra cụm có độ tuổi và giới tính gần nhất từ các kết quả phân cụm đã có

```

1 query = '''
2     SELECT
3     USERINF.gender AS gender,
4     YEAR(GETDATE()) - YEAR(USERINF.date_of_birth) AS age,
5     CategoryDocument.category_id AS category_id
6 FROM [dbo].[account]
7 INNER JOIN (
8     SELECT user_id, gender, date_of_birth
9     FROM [dbo].[doc_user]
10 ) AS USERINF ON USERINF.user_id = account.user_id
11 INNER JOIN (
12     SELECT
13         history_download.account_id,
14         document.category_id
15     FROM document
16     INNER JOIN history_download ON history_download.document_id = document.document_id
17 ) AS CategoryDocument ON CategoryDocument.account_id = account.account_id;
18 '''
19 df = pd.read_sql(query, cnxn)
20
21 scaler = StandardScaler()
22 scaled_data = scaler.fit_transform(df[['gender', 'age', 'category_id']])
23
24 kmeans = KMeans(n_clusters=find_K(df), random_state=42)
25
26 df['cluster'] = kmeans.fit_predict(scaled_data)
27
28 user_data = np.array([[1, 27]])
29 nn = NearestNeighbors(n_neighbors=1)
30 nn.fit(df[['gender', 'age']])
31 _, indices = nn.kneighbors(user_data)
32
33 user_cluster = df.iloc[indices[0][0]]['cluster']
34 user_category = df.iloc[indices[0][0]]['category_id']
35 print(f"User belongs to cluster {user_cluster} and suggested category is {user_category}")
36
37 cluster_summary = df.groupby('cluster').mean()
38 print(cluster_summary)

```

Hình 33: Phân người dùng vào cụm phù hợp.

cluster	gender	age	category_id
0	1.0	26.421053	2.315789
1	0.0	17.205128	1.000000
2	1.0	17.320000	1.000000
3	0.0	18.400000	3.000000
4	0.0	17.935484	2.000000
5	1.0	17.407407	2.296296
6	1.0	26.325000	1.000000
7	0.0	26.757576	2.242424
8	0.0	25.800000	1.000000

Hình 34: Kết quả của thuật toán.

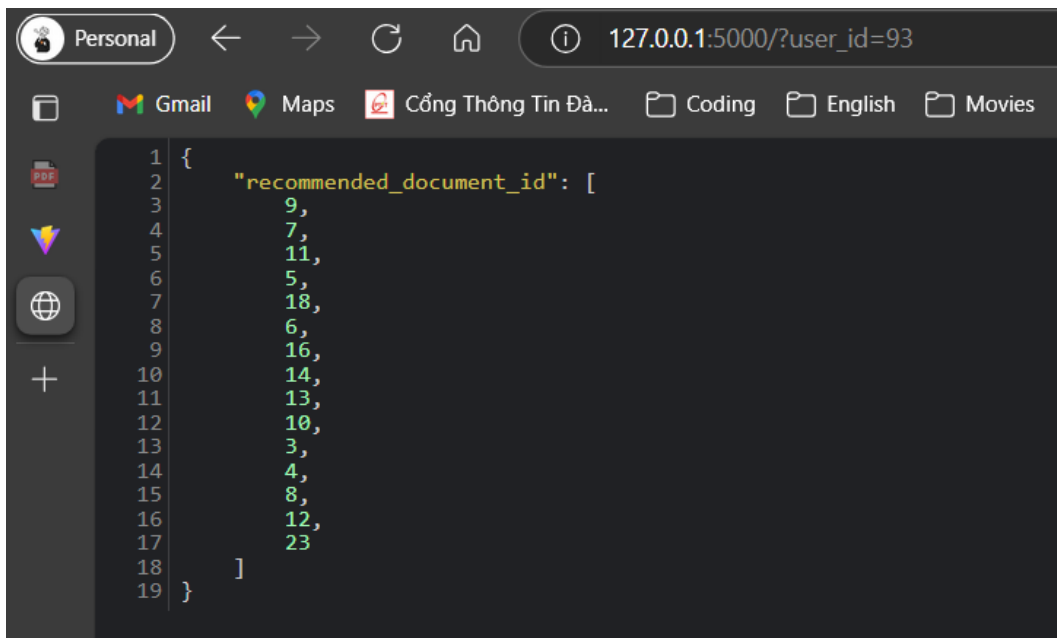
Như vậy, người dùng này sẽ được đề xuất các tài liệu thuộc thể loại có mã id là 1.

Ngoài ra, khi người dùng đã tải một số tài liệu, hệ thống sẽ dựa thêm dữ liệu lịch sử tải của người dùng kết hợp với dữ liệu đề xuất bằng thuật toán K-Means để đề xuất cho người dùng những tài liệu dựa theo xu hướng quan tâm của họ.

```
category_ids = user_history['category_id'].values
user_documents = recently_documents(category_ids)
user_category = recommended_category_id(user_id=user_id)
recommended_document_based_on_Kmeans = recommended_documents(user_category)
final_recommendations = pd.concat([
    pd.Series(user_documents.values.flatten().tolist()),
    pd.Series(recommended_document_based_on_Kmeans.values.flatten().tolist())
], ignore_index=True).drop_duplicates()

return app.json.response({
    "recommended_document_id": final_recommendations.tolist()
})
```

Hình 35: Hệ thống xử lý dữ liệu để đề xuất tài liệu dựa vào lịch sử tải và thuật toán K-Means.



Hình 36: Danh sách mã tài liệu đề xuất

### 3. Đánh giá độ hiệu quả của hệ thống:

Để đánh giá mức độ hiệu quả của hệ thống, nhóm sử dụng cách tính như sau:

- Với mỗi lần đề xuất, nếu người dùng nhấp vào một trong các tài liệu được đề xuất thì được tính là thành công. Ngược lại sẽ được coi là không thành công.
- Độ hiệu quả của hệ thống được tính theo công thức: Số lần khuyến nghị thành công / Tổng số lần khuyến nghị.

### 4. Kết luận

Thông qua những dữ liệu thu thập từ sản phẩm và phản hồi của người dùng, thuật toán K-Means đã chứng tỏ được tính hiệu quả trong việc phân nhóm dữ liệu. Thuật

toán đã phân tích và xử lý khối lượng lớn thông tin và đạt được mục tiêu phân loại các đối tượng vào các nhóm tương tự nhau. Đặc biệt, nhờ vào phương pháp Elbow, hệ thống đã chọn ra số cụm tối ưu mà không cần phải can thiệp thủ công, từ đó giảm thiểu sai sót và giúp thuật toán hoạt động hiệu quả hơn. Ngoài ra, phương pháp này giúp hệ thống có thể dễ dàng điều chỉnh và thay đổi dữ liệu sản phẩm mà không phải gặp sự cố hay ảnh hưởng đến hiệu suất hoạt động của thuật toán trong việc thêm, xóa hay sửa dữ liệu sản phẩm.

Tóm lại, việc sử dụng K-Means kết hợp với phương pháp Elbow không chỉ giúp hệ thống đạt được hiệu quả trong thời gian ngắn mà còn đảm bảo tính linh hoạt và khả năng mở rộng trong tương lai. Khi hệ thống phát triển và yêu cầu xử lý các dữ liệu mới, thuật toán này vẫn có thể thích nghi một cách dễ dàng, đảm bảo hệ thống luôn duy trì được tính ổn định và khả năng phân nhóm chính xác.

## VI- KẾT LUẬN:

Đề tài nghiên cứu "*Xây dựng hệ thống quản lý tài liệu điện tử tích hợp đề xuất thông minh*" đã đạt được một số mục tiêu quan trọng trong việc ứng dụng thuật toán phân cụm K-Means vào hệ thống khuyến nghị tài liệu. Mục tiêu của đề tài không chỉ là tìm hiểu và áp dụng thuật toán K-Means trong hệ thống quản lý tài liệu điện tử mà còn cung cấp một cái nhìn tổng quan và chi tiết về hệ thống đề xuất, từ cơ sở lý thuyết đến việc triển khai thực tế.

Về mặt lý thuyết, đề tài đã trình bày một cái nhìn tổng quan về hệ khuyến nghị và các phương pháp tiếp cận cơ bản trong việc xây dựng một hệ thống khuyến nghị cho hệ thống quản lý tài liệu điện tử. Ngoài ra, thuật toán K-Means được chọn để phân cụm các tài liệu điện tử, từ đó giúp hệ thống đề xuất những tài liệu phù hợp cho từng người dùng dựa trên sở thích và hành vi của họ.

Ngoài việc cung cấp cái nhìn lý thuyết, đề tài cũng đi sâu vào các vấn đề kỹ thuật liên quan đến việc áp dụng thuật toán K-Means. Các khái niệm như học máy (machine learning), thuật toán K-Means, cấu hình thuật toán và các tham số cần điều chỉnh đã được trình bày chi tiết. Các hàm đo lường hiệu quả của thuật toán K-Means, như độ đồng nhất trong các cụm, khoảng cách giữa các cụm, và các phương pháp tối ưu hóa số lượng cụm (như phương pháp Elbow), cũng đã được giới thiệu và phân tích rõ ràng. Những yếu tố ảnh hưởng đến kết quả của thuật toán, như độ chính xác của dữ liệu, sự phân bố của các điểm dữ liệu và các yếu tố khác, cũng được xem xét kỹ càng.

Về mặt thử nghiệm, đề tài đã giới thiệu việc triển khai thuật toán K-Means trong môi trường thực tế, sử dụng các ngôn ngữ lập trình như Python và Java để xây dựng phần mềm khuyến nghị sản phẩm. Phần mềm này một công cụ khuyến nghị giúp người dùng dễ dàng tìm kiếm.

Về mặt dữ liệu, đề tài sử dụng nguồn dữ liệu tự tạo ra (random data) và dữ liệu thu thập từ khách hàng. Do dữ liệu thường ở dạng chuỗi, việc tiền xử lý dữ liệu là bước quan trọng để cải thiện chất lượng bộ dữ liệu, đảm bảo dữ liệu có sự đồng nhất và sẵn sàng cho quá trình huấn luyện thuật toán. Quá trình tiền xử lý bao gồm việc làm sạch dữ liệu, chuẩn hóa các đặc trưng và xử lý các dữ liệu thiếu hoặc không hợp lệ, giúp mô hình học máy đạt hiệu quả tối ưu.

Về mặt ứng dụng, giao diện người dùng của hệ thống được thiết kế đơn giản, dễ sử dụng và thân thiện với người dùng. Hệ thống quản lý tài liệu điện tử được thiết kế đáp ứng được những nhu cầu cơ bản. Đối với mỗi vai trò, người dùng sẽ có các quyền khác nhau.

### ❖ Admin:

- Đăng tải các tài liệu điện tử mới.
- Cập nhật các tài liệu điện tử.
- Đọc thông tin cá nhân của người dùng.
- Tạo và xóa tài khoản.
- Phân quyền tài khoản.
- Chỉnh sửa thông tin cá nhân của người dùng.

- Chỉnh sửa tài khoản của người dùng.
- Phê duyệt các tài liệu điện tử của người dùng có tài khoản đăng tải.
- ❖ Người dùng khác:
  - Tìm kiếm tài liệu điện tử.
  - Xem trước tài liệu điện tử.
- ❖ Người dùng có tài khoản:
  - Tìm kiếm tài liệu điện tử.
  - Xem trước tài liệu điện tử.
  - Tải tài liệu điện tử.
  - Chỉnh sửa thông tin cá nhân.
  - Đổi mật khẩu.
  - Được hệ thống đề xuất thông minh các tài liệu điện tử.
  - Đăng tải các tài liệu điện tử
  - Xóa và chỉnh sửa các tài liệu điện tử của mình đăng tải.

Trong quá trình hoàn thành đề tài, nhóm đã tìm hiểu và tham khảo nhiều nguồn tài liệu liên quan. Tuy nhiên, trong quá trình làm việc, hệ thống vẫn cần nhiều sự thay đổi và phát triển thêm các chức năng cần thiết của một hệ thống quản lý. Chính vì vậy, nhóm em tiếp tục sửa chữa, khắc phục những hạn chế mà ứng dụng đang có; hoàn thiện một cách tối ưu nhất để khi áp dụng hệ thống vào thực tế sẽ giúp cho người dùng sử dụng một cách thuận tiện nhất. Đặc biệt, nhóm mong nhận được những nhận xét và góp ý của thầy để hoàn thiện hơn đề tài này.

Tổng kết lại, đề tài đã chứng minh rằng việc ứng dụng thuật toán K-Means trong hệ thống đề xuất thông minh trong hệ thống quản lý tài liệu điện tử không chỉ giúp nâng cao hiệu quả phân tích và đề xuất tài liệu, mà còn tạo ra một hệ thống khuyến nghị thông minh, dễ sử dụng và có khả năng mở rộng trong tương lai. Với việc xây dựng một mô hình kết hợp giữa học máy và quản lý, nhóm em đã vận dụng được những kiến thức của môn học vào thực tiễn, từ đó là giúp xây dựng nền tảng để phát triển các hệ thống khác sau này.

**TÀI LIỆU THAM KHẢO:**

- [1] **Geoff Hulten.** 2018. *Building Intelligent Systems: A Guide to Machine Learning Engineering.* Apress.
- [2] **Emmanuel Ameisen.** 2020. *Building Machine Learning Powered Applications: Going from Idea to Product.* O'Reilly Media, USA.
- [3] **Shai Shalev-Shwartz and Shai Ben-David.** 2014. *Understanding Machine Learning: From Theory to Algorithms.* Cambridge University Press, New York, NY, USA.
- [4] **Dennis, Wixom and Tegarden,** *System Analysis and Design: An object-oriented approach with UML*, 8th Edition, Wiley Pub., 2015