

**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI  
PHÂN HIỆU TẠI TP. HỒ CHÍ MINH  
BỘ MÔN CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO ĐỀ ÁN TỐT NGHIỆP**

**ĐỀ TÀI:**

**NGHIÊN CỨU HỆ THỐNG ĐỀ XUẤT NỘI DUNG ÁP DỤNG  
KĨ THUẬT CONTENT-BASED FILTERING TRONG ỨNG  
DỤNG NGHE NHẠC TRỰC TUYẾN INCOGNITO MUSIC**

Giảng viên hướng dẫn: TRẦN PHONG NHÃ

Sinh viên thực hiện: HOÀNG MINH TÀI

Lớp : CÔNG NGHỆ THÔNG TIN

Khoá : 60

Tp. Hồ Chí Minh, năm 2023

**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI  
PHÂN HIỆU TẠI TP. HỒ CHÍ MINH  
BỘ MÔN CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO ĐỒ ÁN TỐT NGHIỆP**

**ĐỀ TÀI:**

**NGHIÊN CỨU HỆ THỐNG ĐỀ XUẤT NỘI DUNG ÁP DỤNG  
KĨ THUẬT CONTENT-BASED FILTERING TRONG ỨNG  
DỤNG NGHE NHẠC TRỰC TUYẾN INCOGNITO MUSIC**

Giảng viên hướng dẫn: TRẦN PHONG NHÃ

Sinh viên thực hiện: HOÀNG MINH TÀI

Lớp : CÔNG NGHỆ THÔNG TIN

Khoá : 60

Tp. Hồ Chí Minh, năm 2023

**NHIỆM VỤ THIẾT KẾ TỐT NGHIỆP****BỘ MÔN: CÔNG NGHỆ THÔNG TIN**

-----\*\*\*-----

**Mã sinh viên:** 6051071102**Họ tên SV:** HOÀNG MINH TÀI**Khóa:** 60**Lớp:** CÔNG NGHỆ THÔNG TIN**1. Tên đề tài**

Nghiên cứu hệ thống đề xuất nội dung áp dụng kĩ thuật Content-based Filtering trong ứng dụng nghe nhạc trực tuyến Incognito Music

**2. Mục đích, yêu cầu****a. Mục đích**

- Xây dựng ứng dụng nghe nhạc trực tuyến hoàn toàn miễn phí với đầy đủ tính năng nhằm mang lại trải nghiệm nghe nhạc tốt hơn cho người dùng.
- Nghiên cứu hệ thống đề xuất nội dung giúp người dùng tìm kiếm thông tin một cách nhanh chóng và hiệu quả hơn.

**b. Yêu cầu**

- Có bộ dữ liệu đủ lớn về các thể loại và nghệ sĩ
- Phát nhạc ở chất lượng cao nhất và có thể tùy chọn chất lượng
- Giao diện dễ nhìn và thân thiện cho các tác vụ như tìm kiếm và trải nghiệm các bài hát
- Cá nhân hóa nội dung hiển thị dựa vào sở thích nghe nhạc của mỗi người dùng.
- Cung cấp chức năng tạo các danh sách phát và lưu danh sách yêu thích.
- Chia sẻ các bài hát yêu thích cho mọi người.
- Chuyển đổi linh hoạt giữa các ngôn ngữ.

### **3. Nội dung và phạm vi đề tài**

#### **a. Nội dung**

##### **i. Về ứng dụng**

- Xây dựng giao diện ứng dụng dựa trên Spotify.
- Sử dụng Spotify và YouTube API để tận dụng kho dữ liệu có sẵn làm tiền đề cho hệ thống đề xuất nội dung.
- Thực hiện các chức năng được đặt ra ở phần yêu cầu.
- Nghiên cứu sử dụng Firebase Realtime Database để lưu trữ dữ liệu theo thời gian thực.

##### **ii. Về hệ thống đề xuất nội dung**

- Sử dụng bộ dữ liệu từ Firebase làm dữ liệu đầu vào cho hệ thống.
- Nghiên cứu kỹ thuật Content-based Filtering bao gồm quá trình Tiền xử lý dữ liệu, xây dựng ma trận và sử dụng phép toán Cosine trong tính toán mức độ tương đồng.
- Thực hiện các đề xuất: những ca khúc người dùng có thể thích, bài hát tương tự với bài hát đang nghe, danh sách nhạc thịnh hành.

#### **b. Phạm vi**

- Nghiên cứu và thu thập dữ liệu: Sinh viên lớp CNTT-K60 và nhân viên văn phòng từ 20 đến 30 tuổi.
- Kỹ thuật: Dart(Flutter), Python (Flask, Jupyter Notebook), RESTful API.

### **4. Công nghệ, công cụ và ngôn ngữ lập trình**

- Công nghệ: Flutter, Flask, RESTful API, Jupyter Notebook.
- Công cụ: Visual Studio Code, Android Studio.
- Ngôn ngữ lập trình: Dart, Python.
- Cơ sở dữ liệu: Firebase Realtime Database

### **5. Các kết quả chính dự kiến sẽ đạt được và ứng dụng**

- Ứng dụng Incognito Music đáp ứng được tất cả các yêu cầu đề ra.
- Hệ thống đề xuất nội dung hoạt động với tỉ lệ ổn định.
- Bài báo cáo đề tài (Word, Powerpoint).

### **6. Giáo viên và cán bộ hướng dẫn**

Họ tên: Trần Phong Nhã

Đơn vị công tác: Trường Đại học Giao thông Vận tải phân hiệu tại Thành phố Hồ Chí Minh

Điện thoại: 0906 761 014

Email: tpnha@utc2.edu.vn

**Ngày      tháng 03 năm 2023**

**Đã giao nhiệm vụ TKTN**

**Trưởng BM Công nghệ Thông tin**

**Giáo viên hướng dẫn**

**ThS. Trần Phong Nhã**

**ThS. Trần Phong Nhã**

**Đã nhận nhiệm vụ TKTN**

Sinh viên: Hoàng Minh Tài

Ký tên:

Điện thoại: 0852 976 038

Email: 6051071102@st.utc2.edu.vn

## LỜI CẢM ƠN

Thông qua bài báo cáo lần này, em xin gửi lời cảm ơn chân thành đến tất cả những cá nhân đã giúp đỡ em trong hoàn thành Đồ án tốt nghiệp lần này. Dự án sẽ không thể diễn ra suôn sẻ nếu không có sự giúp đỡ của những người này.

Lời đầu tiên, em xin bày tỏ lòng biết ơn tới người hướng dẫn của mình, Thạc sĩ Trần Phong Nhã. Thầy là người đã cho em rất nhiều lời khuyên hữu ích cũng như những gợi ý mới lạ cho đề tài của mình. Đặc biệt là về lĩnh vực máy học, kĩ thuật mà em chưa từng nghĩ rằng sẽ có trong đồ án của mình. Nhờ thầy mà em có đủ tự tin để thử thách bản thân cho một xu hướng mới và mang đầy tính tương lai.

Dự án này là thành quả đúc kết sau 4 năm học tập và rèn luyện tại Trường Đại học Giao thông Vận tải phân hiệu tại Thành phố Hồ Chí Minh. Em xin gửi lời cảm ơn sâu sắc đến tất cả các giảng viên đã trực tiếp giảng dạy và giúp đỡ em trong chặng hành trình dài này. Lời cảm ơn này đi kèm với lời chúc sức khỏe dành cho thầy cô, mong sẽ có thêm nhiều thế hệ sinh viên Giao thông nữa nhận được sự kèm cặp và chỉ dạy của thầy cô.

Mặc dù đã cố gắng hết sức để hoàn thành đồ án nhưng cũng không thể tránh khỏi những sai sót trong ứng dụng. Em rất cảm kích và rất mong nhận được những đánh giá và đóng góp từ quý thầy cô Bộ môn Công nghệ Thông tin, những sự góp ý của thầy cô sẽ là hành trang quý báu mà em cần có trong chặng đường sắp tới.

Cuối cùng, em muốn nói lời cảm ơn đến cả gia đình, đặc biệt là bố mẹ và chị gái vì họ đã an ủi, hỗ trợ và động viên để em tiếp tục tiến về phía trước khi đối mặt với trở ngại trong cuộc sống đại học của mình.

Một lần nữa, em xin chân thành cảm ơn!

# NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

*Tp. Hồ Chí Minh, ngày ..... tháng ..... năm .....*

## **Giáo viên hướng dẫn**

ThS. Trần Phong Nhã

## MỤC LỤC

NHIỆM VỤ THIẾT KẾ TỐT NGHIỆP .....	i
LỜI CẢM ƠN .....	iv
NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN .....	v
MỤC LỤC .....	vi
DANH MỤC CHỮ VIẾT TẮT .....	ix
BẢNG BIÊU, SƠ ĐỒ, HÌNH VẼ .....	x
CHƯƠNG 1. TỔNG QUAN .....	1
1.1 Lý do chọn đề tài .....	1
1.2 Mục tiêu nghiên cứu .....	1
1.3 Phạm vi chọn đề tài .....	1
1.4 Cấu trúc báo cáo đồ án tốt nghiệp .....	1
CHƯƠNG 2. CƠ SỞ LÝ THUYẾT .....	2
2.1 Công nghệ sử dụng .....	2
2.1.1 Giới thiệu về Dart và Flutter .....	2
2.1.2 Giới thiệu về Python .....	2
2.1.3 Giới thiệu về Firebase Realtime Database .....	2
2.2 Tổng quan về Flutter .....	3
2.2.1 Tiện ích con - Widget .....	3
2.2.2 Hỗ trợ tương tác - Gesture .....	3
2.2.3 Quản lý trạng thái - State .....	3
2.2.4 Các lớp con trong ứng dụng Flutter .....	4
2.2.5 Ưu/nhược điểm của Flutter .....	5
2.3 Tổng quan về Python, Flask và Jupyter Notebook .....	6
2.3.1 Lịch sử của Python .....	6
2.3.2 Các tính năng của Python .....	7
2.3.3 Flask .....	7
2.3.4 Jupyter Notebook .....	8

2.4 Tổng quan về Firebase Realtime Database .....	8
2.4.1 Khái niệm .....	8
2.4.2 Các tính năng của Realtime Database .....	8
2.4.3 Các dịch vụ khác của Firebase .....	9
<b>CHƯƠNG 3. HỆ THỐNG ĐỀ XUẤT NỘI DUNG .....</b>	<b>10</b>
3.1 Giới thiệu .....	10
3.2 Các khái niệm chính .....	11
3.2.1 User - Item .....	11
3.2.2 Thông tin phản hồi từ người dùng và hai dạng bài toán chính trong RS ..	11
3.3 Các kĩ thuật chính trong RS .....	12
3.3.1 Lọc cộng tác .....	12
3.3.2 Lọc dựa trên nội dung .....	14
3.3.3 Hệ thống gợi ý lai (Hybrid recommender systems) .....	15
3.3.4 Các kỹ thuật không cá nhân hóa .....	17
3.4 Chi tiết về kĩ thuật lọc dựa trên nội dung (Content-based Filtering) .....	17
3.4.1 Tiền xử lý dữ liệu .....	17
3.4.2 Đề xuất các bài hát phổ biến .....	20
3.4.3 Đề xuất các bài hát phù hợp với từng người dùng .....	22
<b>CHƯƠNG 4. PHÂN TÍCH THIẾT KẾ HỆ THỐNG .....</b>	<b>25</b>
4.1 Phân tích hệ thống .....	25
4.2 Thiết kế hệ thống .....	27
4.2.1 Sơ đồ phân rã chức năng BFD .....	27
4.2.2 Sơ đồ Use Case .....	28
4.2.3 Sơ đồ hoạt động .....	29
4.3 Cơ sở dữ liệu .....	33
a) Thực thể Bài hát .....	33
b) Thực thể Người dùng .....	34
c) Thực thể Thông số .....	34
<b>CHƯƠNG 5. XÂY DỰNG ỨNG DỤNG .....</b>	<b>35</b>

5.1 Giới thiệu .....	35
5.1.1 Trang chủ .....	35
5.1.2 Màn hình YouTube .....	37
5.1.3 Thư viện .....	38
5.1.4 Cài đặt và Bảng xếp hạng .....	41
5.2 Các chức năng quan trọng .....	41
5.2.1 Nghe nhạc .....	41
5.2.2 Tìm kiếm .....	48
5.2.3 Tùy chỉnh cài đặt .....	48
CHƯƠNG 6. KẾT LUẬN VÀ KIẾN NGHỊ .....	51
6.1 Kết quả đạt được .....	51
6.2 Kiến nghị .....	51
PHỤ LỤC .....	52
Phụ lục 1: Hướng dẫn cài đặt .....	52
Phụ lục 2: Version Source Control : .....	52
TÀI LIỆU THAM KHẢO .....	53

## **DANH MỤC CHỮ VIẾT TẮT**

STT	Mô tả	Ý nghĩa	Ghi chú
1	CNTT	Công nghệ Thông tin	
2	API	Application Programming Interface	
3	AI	(Trí tuệ nhân tạo)	
4	ML	Machine Learning (Máy học)	
5	RS	Recommendation System	
6	FRD	Firebase Realtime Database	

## BẢNG BIỂU

Bảng 4.1 : Phân tích yêu cầu hệ thống .....	26
Bảng 4.2 : Thực thể Bài hát .....	33
Bảng 4.3 : Thực thể Người dùng .....	34
Bảng 4.4 : Thực thể Thông số .....	34

## HÌNH ẢNH

Hình 2.1: Sự khác biệt giữa Stateful và Stateless Widget .....	4
Hình 2.2: Tổng quan chung về một lớp trong Flutter .....	5
Hình 2.3: Những dịch vụ mà Firebase đang cung cấp .....	9
Hình 3.1: Ví dụ về hệ thống đề xuất nội dung .....	10
Hình 3.2: Ma trận biểu diễn dữ liệu trong RS (user-item-rating matrix) .....	11
Hình 3.3: Lấy ra danh sách bài hát từ FRB .....	18
Hình 3.4: Chuyển đổi dữ liệu danh sách sang DataFrame bằng pandas .....	18
Hình 3.5: Trích dữ liệu thông kê từ FRB .....	19
Hình 3.6: Tạo DataFrame chính thức dùng cho việc đề xuất nội dung .....	19
Hình 3.7: Chọn ra các thuộc tính phù hợp .....	20
Hình 3.8: Đề xuất những bài hát phổ biến .....	21
Hình 3.9: Danh sách các bài hát thịnh hành .....	21
Hình 3.10: Lấy ra những bài hát mà người dùng đã nghe .....	22
Hình 3.11: Đề xuất những bài hát có thể phù hợp với người dùng .....	23
Hình 3.12: Tính cá nhân hóa đã được mô hình thể hiện .....	24
Hình 4.1: Sơ đồ phân rã chức năng tổng quát .....	27
Hình 4.2: Sơ đồ phân rã chức năng Quản lý bài hát .....	28
Hình 4.3: Sơ đồ Use Case tổng quát .....	29
Hình 4.4: Sơ đồ hoạt động chức năng Nghe nhạc .....	30
Hình 4.5: Sơ đồ hoạt động chức năng Tải nhạc .....	31
Hình 4.6: Sơ đồ hoạt động chức năng Quản lý danh sách phát .....	32
Hình 4.7: Sơ đồ hoạt động chức năng Tìm kiếm .....	32
Hình 5.1: Màn hình trang chủ .....	35

Hình 5.2: Thanh sidetab .....	35
Hình 5.3: Màn hình YouTube .....	37
Hình 5.4: Thông tin Album của YouTube .....	37
Hình 5.5: Màn hình Thư viện .....	38
Hình 5.6: Danh sách đang phát .....	39
Hình 5.7: Danh sách nghe gần đây .....	39
Hình 5.8: Danh sách bài hát yêu thích .....	40
Hình 5.9: Quản lý các danh sách phát .....	40
Hình 5.10: Màn hình Cài đặt .....	41
Hình 5.11: Màn hình Bảng xếp hạng .....	41
Hình 5.12: Danh sách bài hát tương tự .....	42
Hình 5.13: Xem lời bài hát .....	43
Hình 5.14: Chia sẻ bài hát .....	44
Hình 5.15: Hẹn giờ tắt nhạc .....	45
Hình 5.16: Lựa chọn phương thức hẹn giờ .....	46
Hình 5.17: Tùy chỉnh tốc độ phát .....	47
Hình 5.18: Bộ điều khiển cường độ .....	47
Hình 5.19: Quá trình tìm kiếm bài hát .....	48
Hình 5.20: Tùy chọn thay đổi nền .....	49
Hình 5.21: Tùy chọn thay đổi ngôn ngữ .....	50

# CHƯƠNG 1. TỔNG QUAN

## 1.1 Lý do chọn đề tài

Ở thời điểm hiện tại, việc các ứng dụng bị chi phí quá nhiều bởi quảng cáo là một vấn đề nhức nhối đối với người dùng phổ thông. Bản thân em đã từng gặp trường hợp đang nghe một bản nhạc yêu thích trên ứng dụng Spotify và bị ngắt ngang với các quảng cáo yêu cầu nâng cấp gói tài khoản hoặc bán hàng và nhiều dẫn đến trải nghiệm không tốt. Ngoài ra, đối với các ứng dụng nghe nhạc nổi tiếng như Spotify, iTune,...thì quyền hạn dành cho tài khoản miễn phí rất hạn chế như: số lần bỏ qua quảng cáo bị giới hạn, không thể tải nhạc, thậm chí còn giới hạn cả việc chuyển bài...Điều đó đưa người dùng vào quyết định lựa chọn mua những gói có quyền hạn cao hơn hoặc nghe từ các ứng dụng không chính thống và có khả năng bị theo dõi lợi dụng bởi những mã độc được cài trong các ứng dụng đó.

Đặc biệt các đối tượng như học sinh, sinh viên với mức thu nhập hạn chế thì việc lựa chọn mua các gói tài khoản trả phí sẽ trở nên rất khó khăn. Với cương vị một sinh viên Công nghệ thông tin có sự yêu thích với âm nhạc thì mục tiêu của em đối với đồ án lần này là tạo ra ứng dụng nghe nhạc di động có thể giải quyết được các vấn đề trên.

## 1.2 Mục tiêu nghiên cứu

- Xây dựng ứng dụng nghe nhạc trực tuyến hoàn toàn miễn phí với đầy đủ tính năng nhằm mang lại trải nghiệm nghe nhạc tốt hơn cho người dùng.
- Nghiên cứu hệ thống đề xuất nội dung giúp người dùng tìm kiếm thông tin một cách nhanh chóng và hiệu quả hơn.

## 1.3 Phạm vi chọn đề tài

Ứng dụng phát triển đặt nền tảng cho một dự án với kỳ vọng sẽ hoạt động trên nhiều nền tảng khác nhau cả ứng dụng web và di động. Để đảm bảo trải nghiệm người dùng nên em quyết định phát triển ở phía người dùng trước khi mở rộng ra ứng dụng web cho quản lý dữ liệu bài hát, nghệ sĩ cũng như thông tin người dùng. Bên cạnh đó nghiên cứu các phương pháp tổ chức hệ thống, đảm bảo hệ thống luôn trong trạng thái dễ phát triển và bảo trì về sau.

## 1.4 Cấu trúc báo cáo đồ án tốt nghiệp

- Chương 1: Tổng quan
- Chương 2: Cơ sở lý thuyết
- Chương 3: Hệ thống đề xuất nội dung
- Chương 4: Phân tích thiết kế hệ thống
- Chương 5: Xây dựng ứng dụng

## CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

### 2.1 Công nghệ sử dụng

#### 2.1.1 Giới thiệu về Dart và Flutter

- Ngôn ngữ lập trình Dart

Dart là một ngôn ngữ lập trình đa năng mã nguồn mở ban đầu được phát triển bởi Google và sau đó được ECMA phê duyệt như một tiêu chuẩn.

Dart là một ngôn ngữ lập trình mới dành cho máy chủ cũng như trình duyệt. Được phát triển và giới thiệu bởi Google, Dart SDK đi kèm với trình biên dịch riêng - Dart VM. SDK cũng bao gồm tiện ích - dart2js, một trình chuyển tiếp tạo JavaScript tương đương với Dart Script.

- Flutter Framework

Flutter là framework mã nguồn mở dùng để tạo ra các ứng dụng di động với chất lượng và hiệu suất cao trên các hệ điều hành di động - Android và iOS. Flutter cung cấp một bộ SDK đơn giản, mạnh mẽ, hiệu quả và dễ hiểu để viết ứng dụng di động bằng Dart.

#### 2.1.2 Giới thiệu về Python

Python là một ngôn ngữ lập trình thông dịch, đa năng và dễ học. Nó được phát triển vào cuối những năm 1980 bởi Guido van Rossum tại Hà Lan. Python cho phép bạn làm việc với nhiều loại dữ liệu khác nhau và cung cấp cho bạn một bộ thư viện rộng lớn để giải quyết các vấn đề khác nhau. Nó cũng có thể được sử dụng để phát triển các ứng dụng web và phần mềm máy tính.

#### 2.1.3 Giới thiệu về Firebase Realtime Database

Firebase Realtime Database là một cơ sở dữ liệu thời gian thực được lưu trữ trên đám mây của Google. Nó cho phép bạn lưu trữ và đồng bộ hóa dữ liệu giữa các ứng dụng và các thiết bị khác nhau. Firebase Realtime Database sử dụng mô hình dữ liệu JSON (dạng cấu trúc key : value) để lưu trữ và truy cập dữ liệu. Ngoài ra, cơ sở dữ liệu này cũng cung cấp các tính năng như xác thực người dùng, quản lý lỗi và phân tích thống kê.

## 2.2 Tổng quan về Flutter

Với nhu cầu tạo ra một công nghệ có thể tương thích trên nhiều hệ điều hành thì Flutter - một framework đơn giản và hiệu suất cao dựa trên ngôn ngữ lập trình Dart, cung cấp đầy đủ mọi yêu cầu bằng cách hiển thị giao diện người dùng trực tiếp trong khuôn mẫu của hệ điều hành thay vì thông qua các công nghệ gốc như Java với Android hay Objective-C với iOS.

Flutter cung cấp nhiều tiện ích con sẵn sàng sử dụng để tạo một ứng dụng hiện đại mới. Các widget này được tối ưu hóa cho môi trường di động và thiết kế ứng dụng bằng widget cũng đơn giản như thiết kế HTML.

### 2.2.1 Tiện ích con - Widget

Ứng dụng Flutter tự nó là một widget. Các widget Flutter cũng hỗ trợ các hình ảnh động và cử chỉ. Logic ứng dụng dựa trên lập trình phản ứng, Widget có thể xử lý các thay đổi theo thời gian thực kể cả đối với nội dung tĩnh. Bằng cách thay đổi trạng thái của tiện ích, Flutter sẽ tự động so sánh trạng thái của widget (cũ và mới) và chỉ hiển thị widget với những thay đổi cần thiết thay vì hiển thị lại toàn bộ.

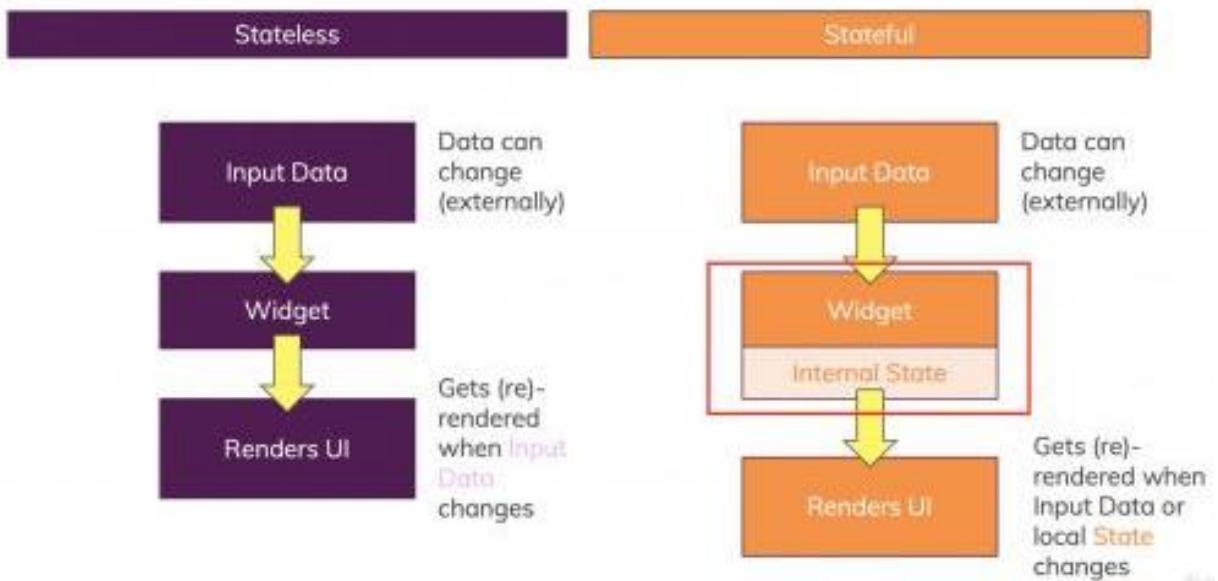
### 2.2.2 Hỗ trợ tương tác - Gesture

Các widget trong Flutter hỗ trợ tương tác thông qua một widget đặc biệt có tên là GestureDetector - một tiện ích con vô hình có khả năng kiểm soát các tương tác của người dùng như chạm, kéo, v.v. của tiện ích con chứa trong nó. Nhiều widget gốc của Flutter hỗ trợ tương tác thông qua việc sử dụng GestureDetector. Lập trình viên cũng có thể kết hợp các tính năng tương tác vào tiện ích hiện có bằng cách đóng gói nó với GestureDetector.

### 2.2.3 Quản lý trạng thái - State

Các widget Flutter hỗ trợ quản lý trạng thái bằng cách cung cấp một widget đặc biệt mang tên StatefulWidget. Tiện ích con cần được bắt nguồn từ StatefulWidget để hỗ trợ duy trì trạng thái và tất cả các widget khác phải bắt nguồn từ StatelessWidget. Điều này tương tự như ReactJs và StatefulWidget sẽ được tự động hiển thị bất cứ khi nào trạng thái bên trong bị thay đổi. Kết xuất được tối ưu hóa bằng cách tìm ra sự khác biệt giữa giao diện người dùng widget cũ và mới và chỉ hiển thị những thay đổi cần thiết.

## Stateless vs Stateful

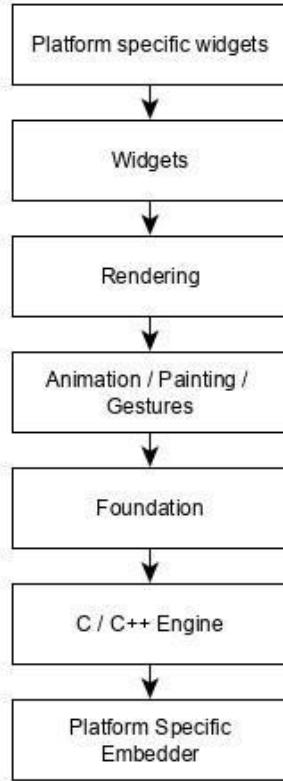


Hình 2.1: Sự khác biệt giữa StatefulWidget và StatelessWidget

### 2.2.4 Các lớp con trong ứng dụng Flutter

Khái niệm quan trọng nhất của khung Flutter là khung được nhóm thành nhiều loại về mức độ phức tạp và được sắp xếp rõ ràng theo các lớp giảm dần độ phức tạp. Một lớp được xây dựng bằng cách sử dụng lớp cấp tiếp theo ngay sau nó. Lớp trên cùng là các widget dành riêng cho Android và iOS. Lớp tiếp theo có tất cả các widget gốc. Lớp tiếp theo là lớp Rendering, đây là một thành phần trình kết xuất cấp thấp và hiển thị mọi thứ trong ứng dụng Flutter. Các lớp dưới dành riêng cho việc phát triển ứng dụng.

- Trong Flutter, mọi thứ đều là một widget và một widget phức tạp bao gồm các widget đã có sẵn.
- Các tính năng tương tác có thể được kết hợp bất cứ khi nào cần thiết bằng widget GestureDetector.
- Trạng thái của một widget có thể được duy trì bất cứ khi nào cần thiết bằng cách sử dụng widget StatefulWidget.
- Flutter cung cấp thiết kế phân lớp để bất kỳ lớp nào cũng có thể được lập trình tùy thuộc vào mức độ phức tạp của nhiệm vụ.



Hình 2.2: Tổng quan chung về một lớp trong Flutter

### 2.2.5 Ưu/nhược điểm của Flutter

- **Ưu điểm**

Flutter đi kèm với các widget đẹp và có thể tùy chỉnh để ứng dụng có hiệu suất cao và tính năng nổi bật. Nó đáp ứng tất cả các yêu cầu của khách hàng. Bên cạnh những điều này, Flutter còn có nhiều ưu điểm khác được đề cập dưới đây:

- ❖ Dart có một kho phần mềm lớn cho phép bạn mở rộng khả năng của ứng dụng của mình.
- ❖ Các nhà phát triển chỉ cần viết một mã duy nhất cho cả hai ứng dụng (cả nền tảng Android và iOS). Flutter cũng có thể được mở rộng sang một nền tảng khác trong tương lai.
- ❖ Flutter ít kiểm thử hơn. Do cơ sở mã duy nhất của nó là đủ nếu chúng ta viết các test case một lần cho cả hai nền tảng.
- ❖ Sự đơn giản của Flutter khiến nó trở thành một ứng cử viên sáng giá để phát triển ứng dụng. Khả năng tùy chỉnh và mở rộng làm cho framework này trở nên mạnh mẽ hơn.

- ❖ Với Flutter, các nhà phát triển có toàn quyền kiểm soát các widget và bộ cục của chúng.
- ❖ Flutter cung cấp các công cụ tuyệt vời dành cho nhà phát triển, với khả năng hot reload đáng kinh ngạc.
- Nhược điểm

Mặc dù có nhiều ưu điểm, nhưng Flutter cũng có những nhược điểm sau:

- ❖ Vì được phát triển trên ngôn ngữ Dart nên khi bắt đầu sử dụng các nhà phát triển phải học ngôn ngữ mới (mặc dù rất dễ học).
- ❖ Framework hiện đại cố gắng tách logic và giao diện càng nhiều càng tốt, nhưng trong Flutter, giao diện và logic được trộn lẫn với nhau. Chúng ta có thể khắc phục điều này bằng cách sử dụng mã hóa thông minh và sử dụng mô-đun cấp cao để phân tách UI và phần mã logic.
- ❖ Flutter là framework mới để tạo ứng dụng di động. Các nhà phát triển đang gặp khó khăn trong việc lựa chọn công cụ phát triển phù hợp.

## 2.3 Tổng quan về Python, Flask và Jupyter Notebook

### 2.3.1 Lịch sử của Python

Python được phát triển bởi Guido van Rossum vào cuối thập niên 80 và đầu thập niên 90 tại CWI (Centrum voor Wiskunde en Informatica – Trung tâm Toán-Tin học) ở Hà Lan và được phát hành lần đầu tiên vào tháng 2 năm 1991.

Python có nguồn gốc từ nhiều ngôn ngữ khác, bao gồm ABC, Modula-3, C, C++, Algol-68, SmallTalk và Unix shell và các ngôn ngữ script khác.

Python hiện được duy trì bởi một nhóm các nhà phát triển tại CWI và Guido van Rossum vẫn giữ một vai trò quan trọng trong việc chỉ đạo.

Sự phát triển Python đến nay có thể chia làm các giai đoạn:

Python 1: bao gồm các bản phát hành 1.x. Giai đoạn này kéo dài từ đầu đến cuối thập niên 1990. Từ năm 1990 đến 1995, Guido làm việc tại CWI tại Amsterdam.

Python 2: bao gồm các bản phát hành từ 2.0 đến 2.7. Giai đoạn này kéo dài từ năm 2000 đến năm 2010.

Python 3: bao gồm các bản phát hành từ 3.0 trở đi. Giai đoạn này kéo dài từ năm 2008 đến nay.

### 2.3.2 Các tính năng của Python

Những đặc điểm nổi bật của Python bao gồm :

- Là ngôn ngữ lập trình bậc cao.
- Hỗ trợ nhiều mô hình lập trình khác nhau như lập trình hướng đối tượng, lập trình hàm và lập trình hướng thủ tục.
- Cú pháp đơn giản và dễ hiểu.
- Có nhiều thư viện và framework để giúp lập trình viên giải quyết các vấn đề phức tạp một cách dễ dàng.
- Có thể chạy trên nhiều hệ điều hành khác nhau.

Các thư viện nổi bật:

- NumPy: Thư viện hỗ trợ tính toán khoa học và toán học.
- Pandas: Thư viện hỗ trợ xử lý dữ liệu.
- Matplotlib: Thư viện hỗ trợ vẽ đồ thị.
- Scikit-learn: Thư viện hỗ trợ machine learning.
- TensorFlow: Thư viện hỗ trợ deep learning.

### 2.3.3 Flask

Flask là một framework web của Python, được phát triển bởi Armin Ronacher và được coi là một microframework bởi vì nó không yêu cầu các công cụ hoặc thư viện cụ thể. Flask phát triển dựa trên bộ công cụ Werkzeug WSGI và công cụ mẫu Jinja2. Cả hai đều là các dự án của Pocco.

Flask giúp tạo các RESTful API một cách đơn giản và nhanh chóng, rất phù hợp cho người mới bắt đầu lập trình Python.

Các tính năng của Flask bao gồm:

- Tính năng phát triển server được tích hợp sẵn.
- Tính năng debug mạnh mẽ.
- Hỗ trợ unit testing.
- Nhiều extension đa năng được phát triển bởi cộng đồng.

### 2.3.4 Jupyter Notebook

Jupyter Notebook là một ứng dụng web mã nguồn mở cho phép bạn tạo và chia sẻ các tài liệu có chứa mã, chương trình, hình ảnh và văn bản giải thích. Ứng dụng này được sử dụng rộng rãi trong cộng đồng khoa học dữ liệu, AI cũng như ML.

Một số tính năng nổi bật của Jupyter Notebook bao gồm:

- Cho phép tạo các tài liệu có chứa mã, chương trình, hình ảnh và văn bản giải thích.
- Hỗ trợ nhiều ngôn ngữ lập trình như Python, R, Julia...
- Có thể chạy trực tiếp các đoạn mã trong tài liệu.
- Cho phép tạo các widget để tương tác với dữ liệu.
- Có thể xuất ra các tài liệu dưới nhiều định dạng như HTML, PDF...

## 2.4 Tổng quan về Firebase Realtime Database

### 2.4.1 Khái niệm

Firebase Realtime Database là một cơ sở dữ liệu NoSQL với khả năng lưu và đồng bộ dữ liệu trên mây. Dữ liệu được đồng bộ trên tất cả máy khách trong thời gian thực và vẫn khả dụng khi ứng dụng ngoại tuyến.

Firebase Realtime Database là cơ sở dữ liệu lưu trữ trên mây. Dữ liệu được lưu trữ và đồng bộ hóa theo thời gian thực với mỗi kết nối. Khi xây dựng ứng dụng đa nền tảng với iOS, Android, và Javascript SDK, tất cả các người dùng chia sẻ một phiên bản Realtime Database và tự động tiếp nhận các thay đổi với dữ liệu mới nhất.

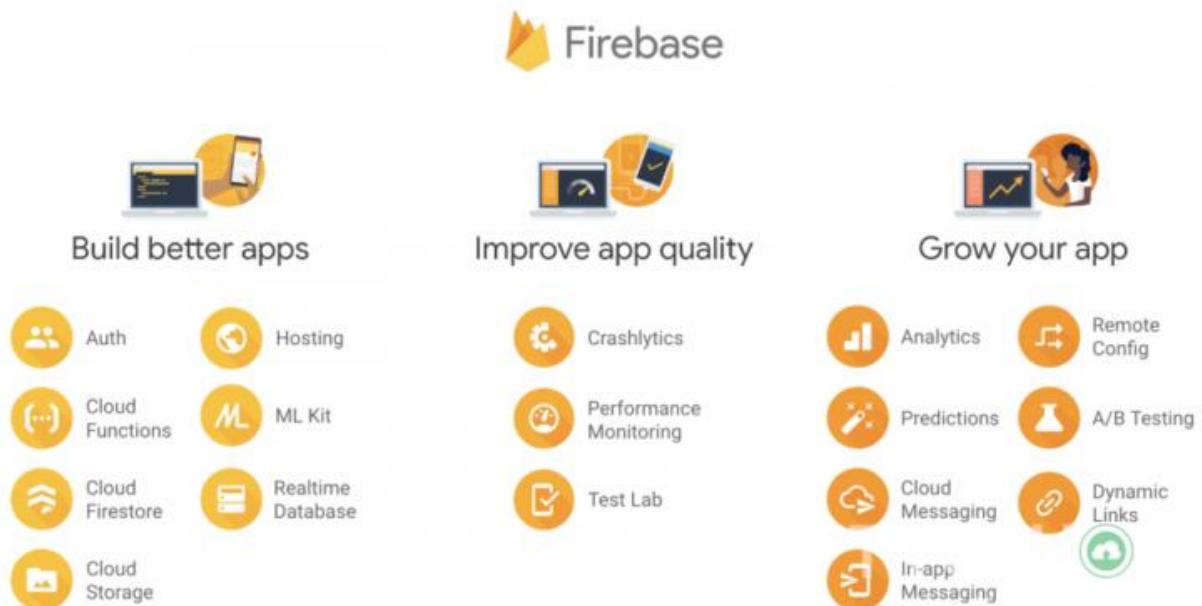
### 2.4.2 Các tính năng của Realtime Database

- Lưu trữ dữ liệu dưới dạng JSON và đồng bộ hóa theo thời gian thực với tất cả các máy khách kết nối.
- Cho phép truy cập an toàn vào cơ sở dữ liệu từ mã phía máy khách.
- Dữ liệu được lưu trữ cục bộ ngay cả khi không có kết nối mạng.
- Khả năng mở rộng linh hoạt trên nhiều cơ sở dữ liệu.
- Cho phép nhập/xuất dữ liệu ra file JSON

#### 2.4.3 Các dịch vụ khác của Firebase

- Cloud Firestore là một cơ sở dữ liệu linh hoạt, có thể mở rộng để phát triển thiết bị di động, web và máy chủ từ Firebase và Google Cloud Platform.
  - Firebase Remote Config lưu trữ các cặp key-value do nhà phát triển chỉ định để thay đổi hành vi và giao diện ứng dụng của bạn mà không yêu cầu người dùng tải xuống bản cập nhật.
  - Firebase Hosting lưu trữ HTML, CSS và JavaScript cho trang web của bạn cũng như các nội dung khác do nhà phát triển cung cấp như đồ họa, phông chữ và biểu tượng.
  - Cloud Storage lưu trữ các tệp như hình ảnh, video và âm thanh cũng như nội dung khác do người dùng tạo.
  - ML Kit chứa các mẫu máy học căn bản có thể được dùng trong việc phát triển ứng dụng. Ngoài ra, ML Kit còn cung cấp tính năng Hosting mẫu máy học do các lập trình viên sáng tạo ra.

Ngoài ra còn rất nhiều các dịch vụ cực kì hữu ích khác được liệt kê ở hình bên dưới.



Hình 2.3: Những dịch vụ mà Firebase đang cung cấp

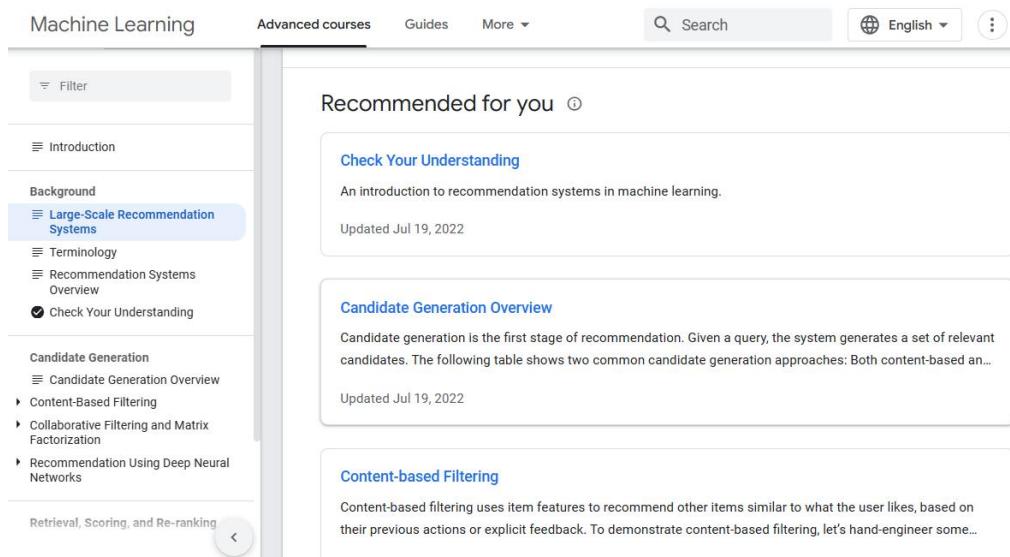
# CHƯƠNG 3. HỆ THỐNG ĐỀ XUẤT NỘI DUNG

## 3.1 Giới thiệu

Hệ thống đề xuất nội dung - RS là một dạng con của hệ thống lọc thông tin (information filtering), nó được sử dụng để dự đoán theo sở thích (preferences) hay xếp hạng (rating) mà người dùng có thể dành cho một mục thông tin (item) nào đó mà họ chưa xem xét tới trong quá khứ (item có thể là bài hát, bộ phim, đoạn video clip, sách, bài báo...).

Ví dụ, trong các ứng dụng nghe nhạc như Spotify, iTunes, nhằm tối ưu hóa trải nghiệm nghe nhạc của người dùng (user), người ta quan tâm đến việc những user nào đã ‘yêu thích’ những bài hát (item) nào bằng cách dựa vào dữ liệu quá khứ của họ (dữ liệu này có thể là xếp hạng mà người dùng đã bình chọn cho bài hát, số lượt nghe bài hát, thời lường nghe...) từ đó hệ thống sẽ dự đoán được người dùng có thể thích bài hát nào và đưa ra những gợi ý phù hợp cho họ.

Hiện tại RS được ứng dụng khá thành công trong nhiều lĩnh vực khác như: gợi ý phim ảnh trên hệ thống của Netflix, gợi ý các video clip trên hệ thống của YouTube; trong giáo dục và đào tạo RS được dùng để gợi ý nguồn tài nguyên học tập như sách, bài báo, địa chỉ web,... cho người học. Dưới đây là hình ảnh minh họa về hệ thống đề xuất của Firebase đã gợi ý những bài viết tương tự với bài viết mà user đang đọc.



Hình 3.1: Ví dụ về hệ thống đề xuất nội dung

Hệ thống đề xuất nội dung không chỉ đơn thuần là một dạng Hệ thống thông tin mà nó còn là cả một lĩnh vực nghiên cứu hiện đang rất được các nhà khoa học quan tâm. Kể từ năm 2007 đến nay, hàng năm đều có hội thảo chuyên về hệ thống gợi ý của ACM (ACM RecSys) cũng như các tiểu bang dành riêng cho RS trong các hội nghị lớn khác như ACM KDD, ACM CIKM,..

### 3.2 Các khái niệm chính

#### 3.2.1 User - Item

Trong RS, thông thường người ta quan tâm đến ba thông tin chính là người dùng (user), mục tin (item, item có thể là sản phẩm, bộ phim, bài hát, bài báo,... tùy hệ thống), và phản hồi (feedback) của người dùng trên mục tin đó (thường là các xếp hạng/đánh giá – rating biểu diễn mức độ thích/quan tâm của họ). Các thông tin này được biểu diễn thông qua một ma trận như trong Hình 3.2. Ở đó, mỗi dòng là một user, mỗi cột là một item, và mỗi ô là một giá trị phản hồi biểu diễn “mức độ thích” của user trên item tương ứng. Các ô có giá trị là những item mà các user đã xếp hạng trong quá khứ. Mặt khác, có thể thấy là mỗi user chỉ xếp hạng cho một vài item trong quá khứ, do vậy có rất nhiều ô trống trong ma trận này – còn gọi là ma trận thưa(sparse matrix).

		Items					
		1	2	...	i	...	m
Users	1	5	3		1	2	
	2		2				4
	:			5			
	u	3	4	?	2	1	
	:					4	
	n			3	2		

Hình 3.2: Ma trận biểu diễn dữ liệu trong RS (user-item-rating matrix)

Nhiệm vụ chính của RS là dựa vào các ô đã có giá trị trong ma trận trên tức là dữ liệu thu được từ quá khứ, thông qua mô hình đã được xây dựng và dự đoán các ô còn trống của user hiện hành, sau đó sắp xếp kết quả dự đoán từ cao xuống thấp và chọn ra Top-N items trong danh sách, từ đó gợi ý chúng cho người dùng.

#### 3.2.2 Thông tin phản hồi từ người dùng và hai dạng bài toán chính trong RS

Trong RS, giá trị phản hồi (feedback)  $r_{ui}$  của mỗi người dùng trên mục tin sẽ được ghi nhận lại để làm cơ sở cho việc dự đoán các giá trị kế tiếp. Tùy theo hệ thống mà giá trị này sẽ có ý nghĩa khác nhau, ví dụ nó có thể dùng để đo độ “phù hợp” hay “mức độ thích” thường là các đánh giá trên các sản phẩm trong các hệ thống thương mại điện tử hay “năng lực/kết quả thực hiện” của người dùng trong các hệ thống e-learning. Trong hệ thống lần này thông được lựa chọn để ghi nhận đó chính là số lượt nghe bài hát.

Giá trị  $r_{ui}$  có thể được xác định một cách tường minh như thông qua việc đánh giá/xếp hạng từ ★ đến ★★★★★; hay like (1) và dislike (0),... mà người dùng  $u$  đã bình chọn cho item  $i$ ; hoặc  $r_{ui}$  có thể được xác định một cách không tường minh thông qua số lần click chuột, thời gian mà  $u$  đã duyệt/xem  $i$ ,... những cách này hiện tại đang

gây nghi ngại về quyền riêng tư khi các hệ thống gần như đã theo dõi người dùng mà chưa được sự đồng ý của họ.

Có 2 dạng bài toán chính trong RS là *dự đoán xếp hạng* (*rating prediction*) của các hệ thống có phản hồi tương minh như đã trình bày ở trên và *dự đoán mục thông tin* (*item prediction/recommendation*) là việc xác định xác suất mà người dùng thích mục tin tương ứng.

### 3.3 Các kĩ thuật chính trong RS

Hiện tại, trong RS có rất nhiều giải thuật được đề xuất, tuy nhiên có thể gom lại vào trong các nhóm chính: nhóm giải thuật lọc theo nội dung (*content-based filtering*), nhóm giải thuật lọc cộng tác (*collaborative filtering*), nhóm giải thuật lai ghép (*hybrid filtering*) và nhóm giải thuật không cá nhân hóa (*non-personalization*).

#### 3.3.1 Lọc cộng tác

Một cách tiếp cận để thiết kế các hệ thống recommender được sử dụng rộng rãi là lọc cộng tác. Các phương pháp lọc cộng tác dựa trên việc thu thập và phân tích một lượng lớn thông tin về hành vi, hoạt động hoặc sở thích của người dùng và dự đoán những gì người dùng sẽ thích dựa trên sự tương đồng của họ với người dùng khác. Một lợi thế quan trọng của phương pháp lọc cộng tác là nó không dựa vào nội dung phân tích máy và do đó nó có khả năng đề xuất chính xác các mục phức tạp như phim mà không yêu cầu “hiểu biết” về mục đó. Nhiều thuật toán đã được sử dụng để đo lường sự giống nhau của người dùng hoặc sự tương đồng về mặt hàng trong các hệ thống giới thiệu. Ví dụ, cách tiếp cận hàng xóm gần nhất (*k-nearest neighbor*) và Pearson Correlation được Allen triển khai lần đầu tiên.

Lọc cộng tác dựa trên giả định rằng những người đã đồng ý trong quá khứ sẽ đồng ý trong tương lai và rằng họ sẽ thích các loại mặt hàng tương tự như họ thích trong quá khứ.

Khi xây dựng mô hình từ hành vi của người dùng, sự phân biệt thường được thực hiện giữa các hình thức thu thập dữ liệu rõ ràng và tiềm ẩn.

Ví dụ về thu thập dữ liệu rõ ràng bao gồm:

- Yêu cầu người dùng xếp hạng một mục trên thang trượt.
- Yêu cầu người dùng tìm kiếm.
- Yêu cầu người dùng xếp hạng một bộ sưu tập các mục từ yêu thích đến ít yêu thích nhất.
- Trình bày hai mục cho một người dùng và yêu cầu anh ta / cô ấy chọn một trong số chúng tốt hơn.

- Yêu cầu người dùng tạo danh sách các mục mà anh / cô ấy thích.

Ví dụ về thu thập dữ liệu ngầm bao gồm:

- Quan sát các mục mà người dùng xem trong cửa hàng trực tuyến.
- Phân tích thời gian xem mục / người dùng.
- Lưu giữ một bản ghi các mục mà người dùng mua trực tuyến.
- Lấy danh sách các mục mà người dùng đã nghe hoặc xem trên máy tính của họ.
- Phân tích mạng xã hội của người dùng và khám phá những lượt thích và không thích tương tự.

Hệ thống recommender so sánh dữ liệu đã thu thập với dữ liệu tương tự và khác nhau được thu thập từ những người khác và tính toán danh sách các mục được đề xuất cho người dùng. Một số ví dụ thương mại và phi thương mại được liệt kê trong bài viết về các hệ thống lọc cộng tác .

Một trong những ví dụ nổi tiếng nhất về lọc cộng tác là lọc cộng tác theo từng mục (những người mua x cũng mua y), một thuật toán được phổ biến rộng rãi bởi hệ thống gợi ý của Amazon.com. Các ví dụ khác bao gồm:

Như đã đề cập chi tiết ở trên, Last.fm đề xuất âm nhạc dựa trên so sánh thói quen nghe của những người dùng tương tự, trong khi Readgeek so sánh xếp hạng sách cho các đề xuất.

Facebook , MySpace , LinkedIn và các mạng xã hội khác sử dụng tính năng lọc cộng tác để giới thiệu bạn bè, nhóm và các kết nối xã hội khác (bằng cách kiểm tra mạng kết nối giữa người dùng và bạn bè của họ). Twitter sử dụng nhiều tín hiệu và tính toán trong bộ nhớ để giới thiệu cho người dùng của họ rằng họ nên “theo dõi”.

Các phương pháp lọc cộng tác thường gặp phải ba vấn đề: Cold Start, khả năng mở rộng và sự thưa thớt (sparsity).

- Cold Start: Các hệ thống này thường yêu cầu một lượng lớn dữ liệu hiện có của người dùng để đưa ra các đề xuất chính xác.
- Khả năng mở rộng: Trong nhiều môi trường mà các hệ thống này đưa ra các khuyến nghị, có hàng triệu người dùng và sản phẩm. Do đó, một lượng lớn công suất tính toán thường là cần thiết để tính toán các gợi ý.
- Sparsity: Số lượng các mặt hàng được bán trên các trang web thương mại điện tử lớn là cực kỳ lớn. Những người dùng tích cực nhất sẽ chỉ đánh giá

một tập con nhỏ của cơ sở dữ liệu tổng thể. Do đó, ngay cả những mặt hàng phổ biến nhất cũng có rất ít xếp hạng.

Một loại thuật toán lọc cộng tác cụ thể sử dụng hệ số ma trận hóa (matrix factorization), kỹ thuật xấp xỉ ma trận cấp thấp (low-rank matrix approximation).

Các phương pháp lọc cộng tác được phân loại là bộ lọc cộng tác dựa trên bộ nhớ và dựa trên mô hình. Một ví dụ nổi tiếng về các phương pháp dựa trên bộ nhớ là thuật toán dựa trên người dùng và các phương pháp dựa trên mô hình là Kernel-Mapping Recommender.

### 3.3.2 Lọc dựa trên nội dung

Một cách tiếp cận phổ biến khác khi thiết kế hệ thống recommender là lọc nội dung . Phương pháp lọc dựa trên nội dung dựa trên mô tả về mặt hàng và hồ sơ về các tùy chọn của người dùng.

Trong hệ thống gợi ý dựa trên nội dung, từ khóa được sử dụng để mô tả các mục và hồ sơ người dùng được xây dựng để chỉ ra loại mục mà người dùng này thích. Nói cách khác, các thuật toán này cố gắng đề xuất các mục tương tự với các mục mà người dùng đã thích trong quá khứ (hoặc đang kiểm tra trong hiện tại). Cụ thể, các mục đề cử khác nhau được so sánh với các mục được đánh giá trước đây bởi người dùng và các mục phù hợp nhất được đề xuất. Cách tiếp cận này có nguồn gốc từ việc thu thập thông tin và nghiên cứu lọc thông tin .

Để tóm tắt các tính năng của các mục trong hệ thống, một thuật toán trình bày mục được áp dụng. Một thuật toán được sử dụng rộng rãi là biểu diễn tf – idf (còn được gọi là biểu diễn không gian vecto).

Để tạo hồ sơ người dùng, hệ thống chủ yếu tập trung vào hai loại thông tin:

1. Một mô hình ưu tiên của người dùng.
2. Lịch sử tương tác của người dùng với hệ thống gợi ý.

Về cơ bản, các phương thức này sử dụng một hồ sơ mặt hàng mô tả mục trong hệ thống. Hệ thống tạo hồ sơ dựa trên nội dung của người dùng dựa trên vecto trọng số của các đối tượng địa lý. Trọng số biểu thị tầm quan trọng của từng tính năng đối với người dùng và có thể được tính từ các vecto nội dung được xếp hạng riêng lẻ bằng nhiều kỹ thuật. Các phương pháp đơn giản sử dụng các giá trị trung bình của vector hạng mục trong khi các phương pháp phức tạp khác sử dụng các kỹ thuật máy học như Bayesian Classifiers , phân tích cụm , cây quyết định và mạng thần kinh nhân tạo (artificial neural networks) để ước tính xác suất người dùng sẽ thích mục đó.

Phản hồi trực tiếp từ người dùng, thường dưới dạng nút thích hoặc không thích , có thể được sử dụng để gán trọng số cao hơn hoặc thấp hơn về tầm quan trọng của các thuộc tính nhất định (sử dụng phân loại Rocchio hoặc các kỹ thuật tương tự khác).

Một vấn đề quan trọng với lọc dựa trên nội dung là liệu hệ thống có thể tìm hiểu các tùy chọn của người dùng từ hành động của người dùng liên quan đến một nguồn nội dung hay không và sử dụng chúng trên các loại nội dung khác. Khi hệ thống bị hạn chế để xuất nội dung cùng loại với người dùng đang sử dụng, giá trị từ hệ thống để xuất thấp hơn đáng kể so với các loại nội dung khác từ các dịch vụ khác có thể được để xuất. Ví dụ: giới thiệu các bài viết tin tức dựa trên việc duyệt tin tức hữu ích nhưng sẽ hữu ích hơn nhiều khi bạn có thể để xuất âm nhạc, video, sản phẩm, cuộc thảo luận, v.v. từ các dịch vụ khác nhau dựa trên duyệt tin tức.

Pandora Radio là một ví dụ về hệ thống giới thiệu dựa trên nội dung phát nhạc có các đặc điểm tương tự như một bài hát do người dùng cung cấp làm hạt giống ban đầu. Ngoài ra còn có một số lượng lớn các hệ thống gợi ý dựa trên nội dung nhằm cung cấp các đề xuất phim, một vài ví dụ như Rotten Tomatoes , Internet Movie Database , Jinni , Rovi Corporation và Jaman . Các hệ thống gợi ý giới thiệu tài liệu liên quan nhằm mục đích cung cấp các đề xuất tài liệu cho các nhà nghiên cứu. Các chuyên gia y tế công cộng đã nghiên cứu các hệ thống gợi ý để cá nhân hóa giáo dục sức khỏe và các chiến lược phòng ngừa.

### 3.3.3 Hệ thống gợi ý lai (Hybrid recommender systems)

Nghiên cứu gần đây đã chứng minh rằng một phương pháp lai, kết hợp lọc cộng tác và lọc dựa trên nội dung có thể hiệu quả hơn trong một số trường hợp. Các phương pháp lai có thể được thực hiện theo nhiều cách:

- **Bằng cách đưa ra các dự đoán dựa trên nội dung và dựa trên lọc cộng tác riêng biệt và sau đó kết hợp chúng.**
- **Bằng cách thêm các khả năng dựa trên nội dung vào phương pháp cộng tác (và ngược lại).**
- **Bằng cách thống nhất các phương pháp tiếp cận thành một mô hình.**

Một số nghiên cứu thực nghiệm so sánh hiệu suất của phương pháp lai với các phương pháp cộng tác thuần túy và chứng minh rằng các phương pháp lai có thể cung cấp các khuyến nghị chính xác hơn các phương pháp thuần túy. Những phương pháp này cũng có thể được sử dụng để khắc phục một số vấn đề thường gặp trong hệ thống gợi ý như Cold Start và vấn đề thưa thớt.

Netflix là một ví dụ tốt về việc sử dụng các hệ thống hybrid recommender. Trang web đưa ra các đề xuất bằng cách so sánh thói quen xem và tìm kiếm của những người dùng tương tự (ví dụ: lọc cộng tác) cũng như bằng cách cung cấp những bộ phim có

chung đặc điểm với những bộ phim mà người dùng đánh giá cao (lọc dựa trên nội dung).

Một loạt các kỹ thuật đã được đề xuất làm cơ sở cho các hệ thống gợi ý: các kỹ thuật hợp tác (collaborative), dựa trên nội dung (content-based), dựa trên kiến thức (knowledge-based) và nhân khẩu học (demographic techniques). Mỗi kỹ thuật này đều có những thiếu sót, như vấn đề Cold Start cho các hệ thống cộng tác và dựa trên nội dung (phải làm gì với người dùng mới với ít xếp hạng) và tắc nghẽn kỹ thuật tri thức (knowledge engineering bottleneck) trong các phương pháp dựa trên tri thức. Một hệ thống gợi ý lai là một hệ thống trong đó kết hợp nhiều kỹ thuật với nhau để đạt được một số sức mạnh tổng hợp giữa chúng.

Cộng tác – Collaborative: Hệ thống tạo đề xuất chỉ sử dụng thông tin về hồ sơ xếp hạng cho những người dùng hoặc mục khác nhau. Các hệ thống cộng tác định vị “người dùng/mục” ngang hàng với lịch sử xếp hạng tương tự như người dùng hoặc mục hiện tại và tạo đề xuất sử dụng vùng lân cận này. Các thuật toán dựa trên người dùng và dựa trên hàng gần nhất có thể được kết hợp để giải quyết vấn đề Cold Start và cải thiện kết quả đề xuất.

Dựa trên nội dung – Content-based: Hệ thống tạo đề xuất từ hai nguồn: các tính năng liên quan đến sản phẩm và xếp hạng mà người dùng đã cung cấp cho họ. Đề xuất dựa trên nội dung coi đề xuất là sự cố phân loại người dùng cụ thể và tìm hiểu trình phân loại cho lượt thích và không thích của người dùng dựa trên các tính năng của sản phẩm.

Nhân khẩu học – demographic techniques: Trình giới thiệu nhân khẩu học cung cấp các đề xuất dựa trên hồ sơ nhân khẩu học của người dùng. Sản phẩm được đề xuất có thể được sản xuất cho các mục nhân khẩu học khác nhau, bằng cách kết hợp xếp hạng của người dùng trong các mục đó.

Dựa trên tri thức – knowledge-based: Trình giới thiệu dựa trên kiến thức gợi ý các sản phẩm dựa trên các suy luận về nhu cầu và sở thích của người dùng. Kiến thức này đôi khi sẽ chứa kiến thức chức năng rõ ràng về cách các tính năng sản phẩm nhất định đáp ứng nhu cầu của người dùng.

Thuật ngữ Hybrid recommender systems được sử dụng ở đây để mô tả bất kỳ hệ thống recommender nào kết hợp nhiều kỹ thuật đề xuất với nhau để tạo dữ liệu đầu ra của nó.

Có bảy kỹ thuật lai cơ bản (hybridization techniques):

- Có trọng số (Weighted): Điểm số của các thành phần đề xuất khác nhau được kết hợp theo số lượng.
- Chuyển đổi (Switching): Hệ thống chọn giữa các thành phần đề xuất và áp dụng hệ thống được chọn.

- Hỗn hợp (Mixed): Các khuyến nghị từ những người giới thiệu khác nhau được trình bày cùng nhau để đưa ra đề xuất.
- Kết hợp tính năng (Feature Combination): Các tính năng được lấy từ các nguồn tri thức khác nhau được kết hợp với nhau và được đưa ra cho một thuật toán gợi ý duy nhất.
- Tính năng tăng cường (Feature Augmentation): Một kỹ thuật gợi ý được sử dụng để tính toán một tính năng hoặc tập hợp các tính năng, sau đó là một phần của đầu vào cho kỹ thuật tiếp theo.
- Cascade: Các khuyến nghị được ưu tiên nghiêm ngặt, với những ưu tiên thấp hơn phá vỡ các mối quan hệ trong việc tính điểm của những người cao hơn.
- Cấp độ meta (Meta-level): Một kỹ thuật đề xuất được áp dụng và tạo ra một số loại mô hình, sau đó là đầu vào được sử dụng bởi kỹ thuật tiếp theo.

### 3.3.4 Các kỹ thuật không cá nhân hóa

Trong nhóm kỹ thuật này, do chúng khá đơn giản, dễ cài đặt nên thường được các website/hệ thống tích hợp vào, gồm cả các website thương mại, website tin tức, hay giải trí. Chẳng hạn như trong các hệ thống bán hàng trực tuyến, người ta thường gợi ý các sản phẩm được xem/mua/bình luận/.. nhiều nhất; gợi ý các sản phẩm mới nhất; gợi ý các sản phẩm cùng loại/ cùng nhà sản xuất/..; gợi ý các sản phẩm được mua/chọn cùng nhau. Một ví dụ khá điển hình là thông qua luật kết hợp (như Apriori), Amazon đã áp dụng khá thành công để tìm ra các sản phẩm hay được mua cùng nhau.

Tuy vậy, bất lợi của các phương pháp này là không cá nhân hóa cho từng người dùng, nghĩa là tất cả các user đều được gợi ý giống nhau khi chọn cùng sản phẩm

## 3.4 Chi tiết về kỹ thuật lọc dựa trên nội dung (Content-based Filtering)

### 3.4.1 Tiền xử lý dữ liệu

Trước tiên, chúng ta cần kết nối mẫu python với FRD chứa dữ liệu về bài hát và các số liệu thống kê cần thiết để sử dụng cho kỹ thuật lọc theo nội dung.

Vì dữ liệu được lưu dưới dạng cặp key: value với key được sinh ra từ động nên key sẽ không có ý nghĩa gì cho việc đề xuất của hệ thống. Trong Hình 3.3 ta có thể thấy rằng giá trị trả về chỉ là danh sách các bài hát.

```

ref = db.reference('/')
data = ref.child('songs').order_by_child('id').get()
values = []
for key in data:
    values.append(data[key])
values

```

[{'320kbps': 'false',  
'album': 'PHÚC DU',  
'album\_id': 'UCPQM-wqVHzULADG9zJir5HQ',  
'artist': 'PHÚC DU',  
'duration': '263',  
'genre': 'YouTube',  
'has\_lyrics': 'false',  
'id': '-7sISWuTdj0',  
'language': 'Vietnamese',  
'lyrics': 'https://img.youtube.com/vi/-7sISWuTdj0/maxresdefault.jpg',  
'model': 'YouTube',  
'path': '/songs/-7sISWuTdj0',  
'songId': '-7sISWuTdj0',  
'title': 'PHÚC DU - 320kbps',  
'video': 'https://img.youtube.com/vi/-7sISWuTdj0/maxresdefault.jpg'}]

Hình 3.3: Lấy ra danh sách bài hát từ FRB

Sử dụng thư viện pandas để trích xuất và xử lý dữ liệu dạng danh sách.

	320kbps	album	album_id	artist	duration	genre	has_lyrics	songId	image	language
0	false	PHÚC DU	UCPQM-wqVHzULADG9zJir5HQ	PHÚC DU	263	YouTube	false	-7sISWuTdj0	https://img.youtube.com/vi/-7sISWuTdj0/maxresdefault.jpg	YouT
1	false	Sơn Tùng M-TP	UC3mulvzjhulNpj4Pn_0kCQw	Sơn Tùng M-TP	258	YouTube	false	-E2u6exhzPE	https://img.youtube.com/vi/-E2u6exhzPE/maxresdefault.jpg	YouT
2	false	HYBE LABELS	UC3lZKseVpdzPSBaWxBundA	HYBE LABELS	213	YouTube	false	GQg25oP0S4	https://img.youtube.com/vi/-GQg25oP0S4/maxresdefault.jpg	YouT
3	false	1theK (원더케이)	UCweOkPb1wVVH0QQTj4a5Pw	1theK (원더케이)	226	YouTube	false	-lOYj7NLxu0	https://img.youtube.com/vi/-lOYj7NLxu0/maxresdefault.jpg	YouT

Hình 3.4: Chuyển đổi dữ liệu danh sách sang DataFrame bằng pandas

Trong Hình 3.4 ta đổi tên cột id thành songId, cột này đóng vai trò như khóa ngoại để kết nối bảng bài hát và thông số. Hiển thị dạng DataFrame giúp so sánh dữ liệu trước và sau khi xử lý trở nên trực quan hơn thay vì dạng danh sách trước đó.

```

stat = ref.child('stats').order_by_child('songId').get()
stats = []
for key in stat:
    stats.append(stat[key])
stat_df = pd.DataFrame(stats)
stat_df.head()

```

	listenCount	songId	userId
0	2	-E2u6exhzPE	0c2983d0-f16d-11ed-9c88-514c38a48ebe
1	1	-sUp0_bLC64	0c2983d0-f16d-11ed-9c88-514c38a48ebe
2	1	-tEH-bLsv34	0c2983d0-f16d-11ed-9c88-514c38a48ebe
3	5	-tKPSWPr	d9155580-f7c7-11ed-ab2f-f146d31a9014
4	1	09R8_2nJtjg	0c2983d0-f16d-11ed-9c88-514c38a48ebe

Hình 3.5: Trích dữ liệu thống kê từ FRB

DataFrame trong Hình 3.5 chứa các thông tin đánh giá của các user đối với các item trong bảng bài hát thông qua số lượt nghe.

```

song_df = pd.merge(stat_df, song.drop_duplicates(['songId']), on='songId', how='left')
song_df.head()

```

	listenCount	songId	userId	320kbps	album	album_id	artist	duration	genre	has_lyrics
0	2	-E2u6exhzPE	0c2983d0-f16d-11ed-9c88-514c38a48ebe	false	Sơn Tùng M-TP	UC3mulvzjhuhNpJ4Pn_0kCQw	Sơn Tùng M-TP	258	YouTube	false
1	1	-sUp0_bLC64	0c2983d0-f16d-11ed-9c88-514c38a48ebe	false	Jaymes Young	UCCK1-D6wWIRuQgaCdIAKsMA	Jaymes Young	228	YouTube	false
2	1	-tEH-bLsv34	0c2983d0-f16d-11ed-9c88-514c38a48ebe	false	Bảo Anh	UCRWrdiC0N2AFnDfbfgy2-7w	Bảo Anh	228	YouTube	false
3	5	-tKPSWPr	d9155580-f7c7-11ed-ab2f-f146d31a9014	true	Bhaiji Bhaiya Se Pay Phone Karal	41855844	Sunil Hero, Jyoti Raaj	183	Bhojpuri	false

Hình 3.6: Tạo DataFrame chính thức dùng cho việc đề xuất nội dung

Chúng ta gộp 2 DataFrame đã lấy được từ FRB thông qua cột songId. Lưu ý, ở đây chỉ lấy những bài hát có songId xuất hiện trong bảng thông số (stat), việc làm này là để tối ưu hóa thời gian xử lý khi các bài hát chưa được bắt cứ người dùng nào nghe gần như không có tác dụng cho hệ thống. Ngoài ra, để tránh sai lệch bước này cũng loại bỏ những bài hát bị trùng lặp.

```

song_df['song'] = song_df['title'] + ' - ' + song_df['artist']
song_df.head()

```

							url	year	song
https://img.youtube.com/vi/-E2u6exhzPE/maxresd...	YouTube	https://www.youtube.com/watch?v=-E2u6exhzPE	2023-05-04 00:00:00.000	Sơn Tùng M-TP - Topic	MAKING MY WAY	https://rr3---sn-8qj-nboey.googlevideo.com/vid...	2023		MAKING MY WAY - Sơn Tùng M-TP
https://img.youtube.com/vi/-sUp0_bLC64/maxresd...	YouTube	https://www.youtube.com/watch?v=-sUp0_bLC64	2022-01-27 00:00:00.000	Jaymes Young - Topic	Happiest Year	https://rr1---sn-8qj-nbo6d.googlevideo.com/vid...	2022		Happiest Year - Jaymes Young
https://img.youtube.com/vi/-tEH-bLsv34/maxresd...	YouTube	https://www.youtube.com/watch?v=-tEH-bLsv34	2018-10-10 00:00:00.000	Bảo Anh	Như Lời Đồn   Bảo Anh (Official MV) (#NLD)	https://rr7---sn-8qj-nboey.googlevideo.com/vid...	2018		Như Lời Đồn   Bảo Anh (Official MV) (#NLD) - B..

Hình 3.7: Chọn ra các thuộc tính phù hợp

Tiêu đề và tên ca sĩ là các thuộc tính được lựa chọn để đơn giản hóa vấn đề trong khuôn khổ bài báo cáo này. Thực tế, chúng ta có thể sử dụng nhiều thuộc tính hơn để tăng mức độ chính xác của mô hình. Mặt khác, sử dụng nhiều thuộc tính có thể là cách chính xác nhất nhưng chưa chắc là cách tốt nhất cho mô hình vì ngoài tính chính xác thì mô hình cần có hiệu suất đủ nhanh để không gây khó chịu cho người dùng.

### 3.4.2 Đề xuất các bài hát phổ biến

Trendings là một cụm từ không còn xa lạ đối với các ứng dụng đa phương tiện hiện nay. Các bài hát, video, bộ phim...càng nhận được nhiều sự chú ý sẽ càng có vị trí cao trên bảng xếp hạng này. “Sự chú ý” theo từng hệ thống sẽ dựa trên những thông số không giống nhau có thể là các đánh giá, lượt like/dislike..., hệ thống lần này sử dụng số lượt nghe để biểu hiện khái niệm này.

```

song_grouped = song_df.groupby(['songId']).agg({'listenCount': ['sum', 'count']})
song_grouped.columns = ['sum_score', 'count_score']
song_grouped= song_grouped.sort_values(['sum_score', 'count_score', 'songId'],ascending=False)
trending_songs = song_grouped.head(20)
trending_songs #list songId to return by API
#Get Trending Songs: done

```

✓ 0.0s

songId	sum_score	count_score
sTgsNebIHu8	430	28
icJam_5I	191	10
6o8JoQ8b	162	10
c-6-mB8N	159	12
kd8JSDbB	146	7

Hình 3.8: Đề xuất những bài hát phổ biến

Như được thể hiện trên Hình 3.8, dữ liệu trong DataFrame mẫu được nhóm lại thông qua songId, trong mỗi nhóm sẽ thực hiện hai phép toán là tính tổng số lượt nghe và đếm số user mà bài hát đó tiếp cận tới. Hai thông số này quyết định những bài hát nào nổi bật theo thứ tự giảm dần.

```

top_trending = song.loc[song['songId'].isin(trending_songs.index)]
top_trending.drop_duplicates(['songId'])

```

✓ 0.1s

	320kbps	album	album_id	artist	duration	genre	has_lyrics	songId
78	NaN	Equals	NaN	Ed Sheeran	214	NaN	NaN	496697
139	true	÷	10436096	Ed Sheeran	263	English	false	6o8JoQ8b
238	false	RayeVEVO	UC1aQVIVnCGa4hDtvNxf11g	RayeVEVO	273	YouTube	false	DII6VJ2C7wo
362	false	Evolve	UC0aXrjVxG5pZr99v77wZdPQ	Imagine Dragons	204	YouTube	false	Kx7B-XvmFtE
425	true	Dandelions (slowed + reverb)	29160455	Ruth B. sped up + slowed	268	English	false	POhlVMwJ
454	false	FIFTY FIFTY Official	UCJEER74X9kBenMT_x9iK9Mw	FIFTY FIFTY Official	203	YouTube	false	Qc7_zRjH808

Hình 3.9: Danh sách các bài hát thịnh hành

### 3.4.3 Đề xuất các bài hát phù hợp với từng người dùng

Cá nhân hóa là mục đích lớn nhất của RS, thay vì chỉ đơn thuần hiển thị dữ liệu từ cơ sở dữ liệu lên thì mô hình dựa trên những bài hát mà người dùng đã nghe trước đó để đưa ra những bài hát mà có thể người dùng đó sẽ thích.

```
self_sv.get_user_similar_songs('b74d04a0-f5e4-11ed-81a7-85dc82d623d5')
✓ 0.8s

Vì Anh Đâu Có Biết - Madihu (Feat. Vũ.) | Official MV - Madihu
Ed Sheeran - Perfect (Official Music Video) - Ed Sheeran
Vai Sentando - Ludmilla, King Doudou
Grupo Frontera x Bad Bunny - un x100to (Video Oficial) - Grupo Frontera
Nắng Ấm Xa Dân - Sơn Tùng M-TP - Sơn Tùng M-TP Official
That's Hilarious - Charlie Puth
Hold My Hand - Lady Gaga
Attention - The Weeknd
Toosii - Favorite Song (Official Video) - Toosii2xVEVO
"Can You Feel The Love Tonight" - English/Chinese/Korean/Japanese/Thai Cover (Lion King) - MusicNeverSleeps
Alec Benjamin - Let Me Down Slowly [Official Music Video] - Alec Benjamin
Iris (Acoustic) - Adam Christopher
Sơn Tùng MTP - Remember Me (SlimV 2017 Mix) - SlimV
Because I Love You (EarthBound) Cover w/ lyrics | Michelle Heafy - michelleheafy
Chuyện Đôi Ta - Emcee L (Da LAB) ft Muôii (Official MV) - Da LAB Official
Deep Sea - Binz x Thanh Nguyễn x Triple D - Binz Da Poet
It's You - Ali Gatie
Jis Din Teri Meri Baat Nahin Hoti - Udit Narayan, Anuradha Paudwal
Lonely (feat. Ronie, Dominika Mirgova) - dvatri tridva
Christina Perri ft. Steve Kazee - A Thousand Years, Pt. 2 - Christina Perri
```

Hình 3.10: Lấy ra những bài hát mà người dùng đã nghe

Cụ thể, mô hình dựa trên mã người dùng được cung cấp sau đó lấy ra danh sách tất cả các ca khúc họ đã nghe mục đích ngoài việc dùng làm cơ sở để tìm ra những bài hát phù hợp thì còn là để tránh việc đề xuất lại những bài hát mà người dùng đã nghe từ trước đó. Chúng ta có thể thấy trong Hình 3.10 ở trên xuất hiện khá nhiều bài hát của Sơn Tùng MTP và Ed Sheeran.

```
No. of unique songs for the user: 89
no. of unique songs in the training set: 661
Non zero values in cooccurrence_matrix :20067
```

	user_id	song	score	rank
0	b74d04a0-f5e4-11ed-81a7-85dc82d623d5	Có Chắc Yêu Là Đây - Sơn Tùng M-TP	0.150153	1
1	b74d04a0-f5e4-11ed-81a7-85dc82d623d5	Perfect - Ed Sheeran	0.129191	2
2	b74d04a0-f5e4-11ed-81a7-85dc82d623d5	Shape of You - Ed Sheeran	0.127693	3
3	b74d04a0-f5e4-11ed-81a7-85dc82d623d5	Flower - Jisoo, BLACKPINK	0.122057	4
4	b74d04a0-f5e4-11ed-81a7-85dc82d623d5	Boy's a liar, Pt. 2 - PinkPantheress, Ice Spice	0.120782	5
5	b74d04a0-f5e4-11ed-81a7-85dc82d623d5	Six Feet Under - The Weeknd	0.119872	6
6	b74d04a0-f5e4-11ed-81a7-85dc82d623d5	SAD B!TCH - Anne-Marie	0.116390	7
7	b74d04a0-f5e4-11ed-81a7-85dc82d623d5	The Chainsmokers - Closer (Official Video) ft....	0.114451	8
8	b74d04a0-f5e4-11ed-81a7-85dc82d623d5	I Will Remember You - Ed Sheeran	0.114241	9
9	b74d04a0-f5e4-11ed-81a7-85dc82d623d5	ĐAU ĐỂ TRƯỞNG THÀNH   ONLYC   OFFICIAL MV - On...	0.113019	10
10	b74d04a0-f5e4-11ed-81a7-85dc82d623d5	Matmos - Amplifier	0.112832	11
11	b74d04a0-f5e4-11ed-81a7-85dc82d623d5	RAYE, 070 Shake - Escapism. (Official Music Vi...	0.112520	12
12	b74d04a0-f5e4-11ed-81a7-85dc82d623d5	You Only Love Me - Rita Ora	0.112114	13
13	b74d04a0-f5e4-11ed-81a7-85dc82d623d5	Happier - Marshmello	0.111954	14
14	b74d04a0-f5e4-11ed-81a7-85dc82d623d5	Shawn Mendes, Camila Cabello - Señorita - Shaw...	0.111584	15
15	b74d04a0-f5e4-11ed-81a7-85dc82d623d5	The Chainsmokers - Don't Let Me Down (Official...	0.111428	16
16	b74d04a0-f5e4-11ed-81a7-85dc82d623d5	STAY - Justin Bieber, The Kid Laro	0.110719	17

Hình 3.11: Đề xuất những bài hát có thể phù hợp với người dùng

Đây là kết quả của mô hình khi đề xuất bài hát, Hình 3.11 cho thấy mô hình đề xuất được những bài hát đã từng được nghe của người dùng, cụ thể ở đây là các bài hát khác của Sơn Tùng MTP và Ed Sheeran.

		self_sv.get_user_similar_songs('0c2983d0-f16d-11ed-9c88-514c38a48ebe')	
✓	1.0s		
No. of unique songs for the user: 164			
no. of unique songs in the training set: 661			
Non zero values in cooccurrence_matrix :38781			
	user_id	song	score rank
0	0c2983d0-f16d-11ed-9c88-514c38a48ebe	Dandelions (slowed + reverb) - Ruth B., sped up by ...	0.084370 1
1	0c2983d0-f16d-11ed-9c88-514c38a48ebe	Flowers - Miley Cyrus	0.083022 2
2	0c2983d0-f16d-11ed-9c88-514c38a48ebe	土岐麻子 / 「HOME」Music Video - 土岐麻子	0.082536 3
3	0c2983d0-f16d-11ed-9c88-514c38a48ebe	Matmos - Amplifier	0.076858 4
4	0c2983d0-f16d-11ed-9c88-514c38a48ebe	KHÔNG THỂ SAY - HIEUTHUHAI	0.075734 5
5	0c2983d0-f16d-11ed-9c88-514c38a48ebe	Rod Wave - Fight The Feeling (Official Video) ...	0.073693 6
6	0c2983d0-f16d-11ed-9c88-514c38a48ebe	Dù Cho Mai Về Sau (Acoustic Version) - buitruong...	0.073316 7
7	0c2983d0-f16d-11ed-9c88-514c38a48ebe	MONO - Waiting For You (Album 22 - Track No.10...)	0.072777 8
8	0c2983d0-f16d-11ed-9c88-514c38a48ebe	Iris (Acoustic) - Adam Christopher	0.072254 9
9	0c2983d0-f16d-11ed-9c88-514c38a48ebe	DJ Snake - Let Me Love You ft. Justin Bieber ...	0.072141 10
10	0c2983d0-f16d-11ed-9c88-514c38a48ebe	Doja Cat - Woman (Official Video) - dojacatVEVO	0.069527 11
11	0c2983d0-f16d-11ed-9c88-514c38a48ebe	SIMPLE LOVE - Obito x Seachains x Davis x Lena...	0.069527 12
12	0c2983d0-f16d-11ed-9c88-514c38a48ebe	Beyoncé - Halo - BeyoncéVEVO	0.069527 13
13	0c2983d0-f16d-11ed-9c88-514c38a48ebe	In The Stars - Benson Boone	0.069527 14
14	0c2983d0-f16d-11ed-9c88-514c38a48ebe	Angel Pt. 1 - Jimin	0.069527 15

Hình 3.12: Tính cá nhân hóa đã được mô hình thể hiện

Kết quả khi yêu cầu mô hình đề xuất các bài hát cho những người dùng khác nhau là khác nhau, từ đó có thể khẳng định rằng mô hình này đảm bảo tính cá nhân hóa.

## CHƯƠNG 4. PHÂN TÍCH THIẾT KẾ HỆ THỐNG

### 4.1 Phân tích hệ thống

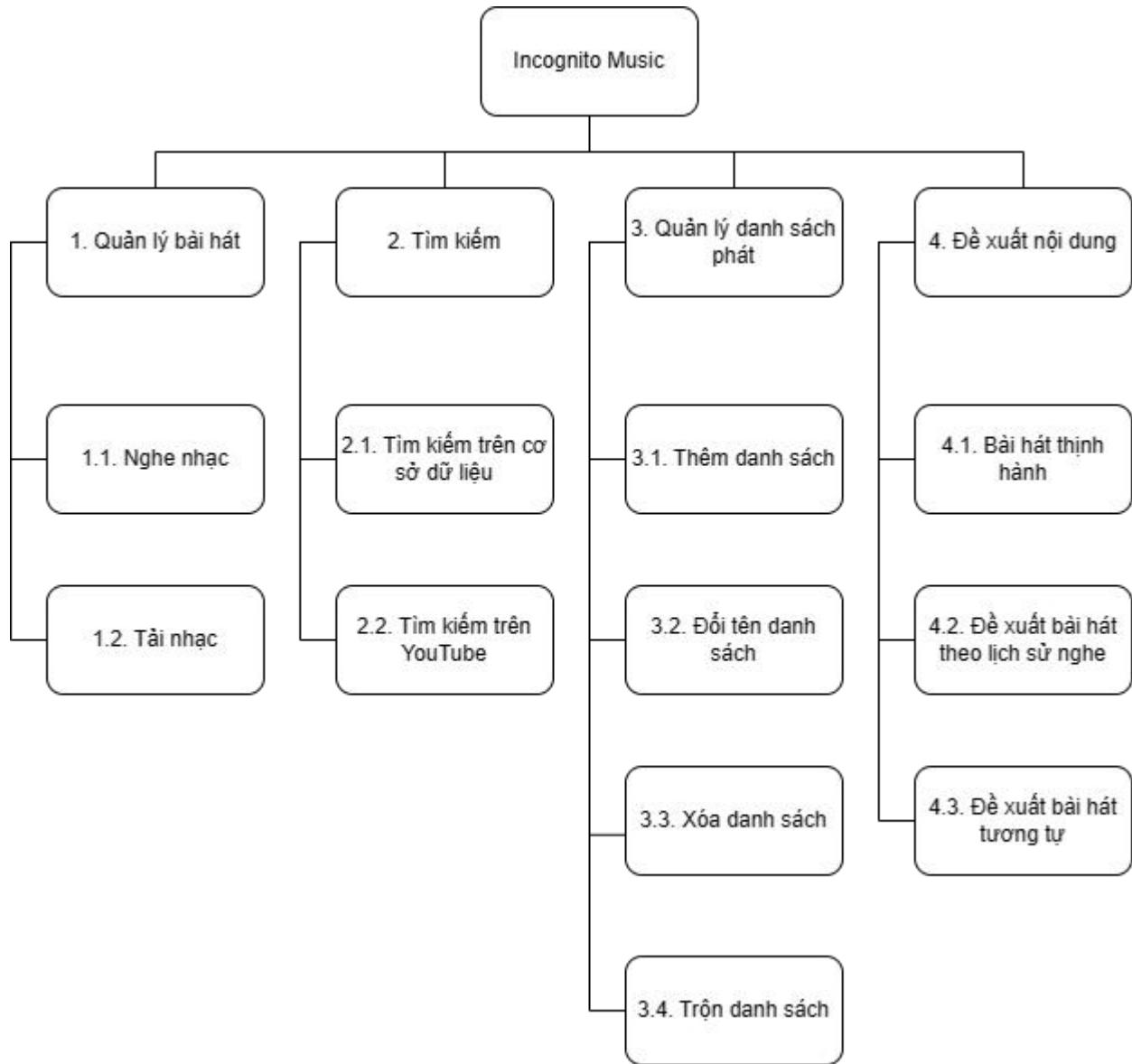
STT	Tên yêu cầu	Mô tả
1	Đăng nhập tài khoản	Người dùng chỉ cần nhập tên để gợi nhớ mà không nhất thiết phải cung cấp quá nhiều thông tin cá nhân.
2	Nghe nhạc	Khi lựa chọn được bài nhạc ưa thích người dùng nhấn vào item đó, ứng dụng sẽ chuyển tới trang phát nhạc và nhạc sẽ tự động phát (trang này dùng cho cả nhạc online và offline). Ở đây người dùng có thể chuyển bài, tạm dừng, thay đổi tốc độ phát.
3	Tải nhạc	Chức năng này dùng để lưu lại những bài hát mà người dùng tâm đắc, nó được tích hợp dưới dạng nút bấm trong trang phát nhạc cũng như trong danh sách.
4	Thông báo	Khi phát nhạc, sẽ hiển thị thanh thông báo với các thông tin cơ bản của item đang được phát cùng với những tác vụ căn bản như chuyển bài, tạm dừng, yêu thích...(các tác vụ sẽ do người dùng tự điều chỉnh)
5	Bộ điều khiển âm thanh	Tích hợp trong màn hình phát nhạc giúp người dùng điều chỉnh mức độ to nhỏ của các dải tần số âm thanh. Tùy theo từng thể loại nhạc mà điều chỉnh cho phù hợp.
6	Quản lý danh sách phát	Các danh sách phát (playlist) được tạo mới, đổi tên, xóa và trộn lại với nhau. Chức năng này giúp cho người dùng phân loại các ca khúc theo chủ đề mà mình mong muốn.
7	Tùy chỉnh cài đặt	Người dùng có thể thay đổi các cài đặt của ứng dụng như giao diện, ngôn ngữ, chất lượng phát nhạc, cursive...

8	Nghe nhạc từ YouTube	Để tăng thêm trải nghiệm cho người dùng, ứng dụng có sẵn tính năng phát các video trên YouTube dưới dạng audio. Người dùng có thể tìm kiếm, nghe và tải nhạc từ nguồn này.
9	Xem BXH từ Spotify	Sử dụng Spotify API lấy dữ liệu về bảng xếp hạng theo khu vực, người dùng có thể nhấn chọn bài hát mình muốn nghe và ứng dụng sẽ tìm kiếm dữ liệu của bài hát đó trên YouTube và chuyển tới trang phát nhạc.
10	Chia sẻ bài hát	Chức năng được tích hợp thông qua một nút ở màn hình phát nhạc. Khi nhấn vào, ứng dụng sẽ chia sẻ đường dẫn của bài hát đến tài khoản mạng xã hội mà bạn chỉ định.
11	Quản lý hàng đợi	Danh sách đang phát sẽ được hiển thị dưới dạng hàng đợi mà ở đó người dùng có thể thay đổi thứ tự phát, xóa bài hát.
12	Cá nhân hóa nội dung	Thông qua hệ thống đề xuất nội dung được tích hợp, ứng dụng sẽ dựa vào lịch sử nghe nhạc của người dùng và hiển thị những nội dung tương tự nhằm tăng trải nghiệm.

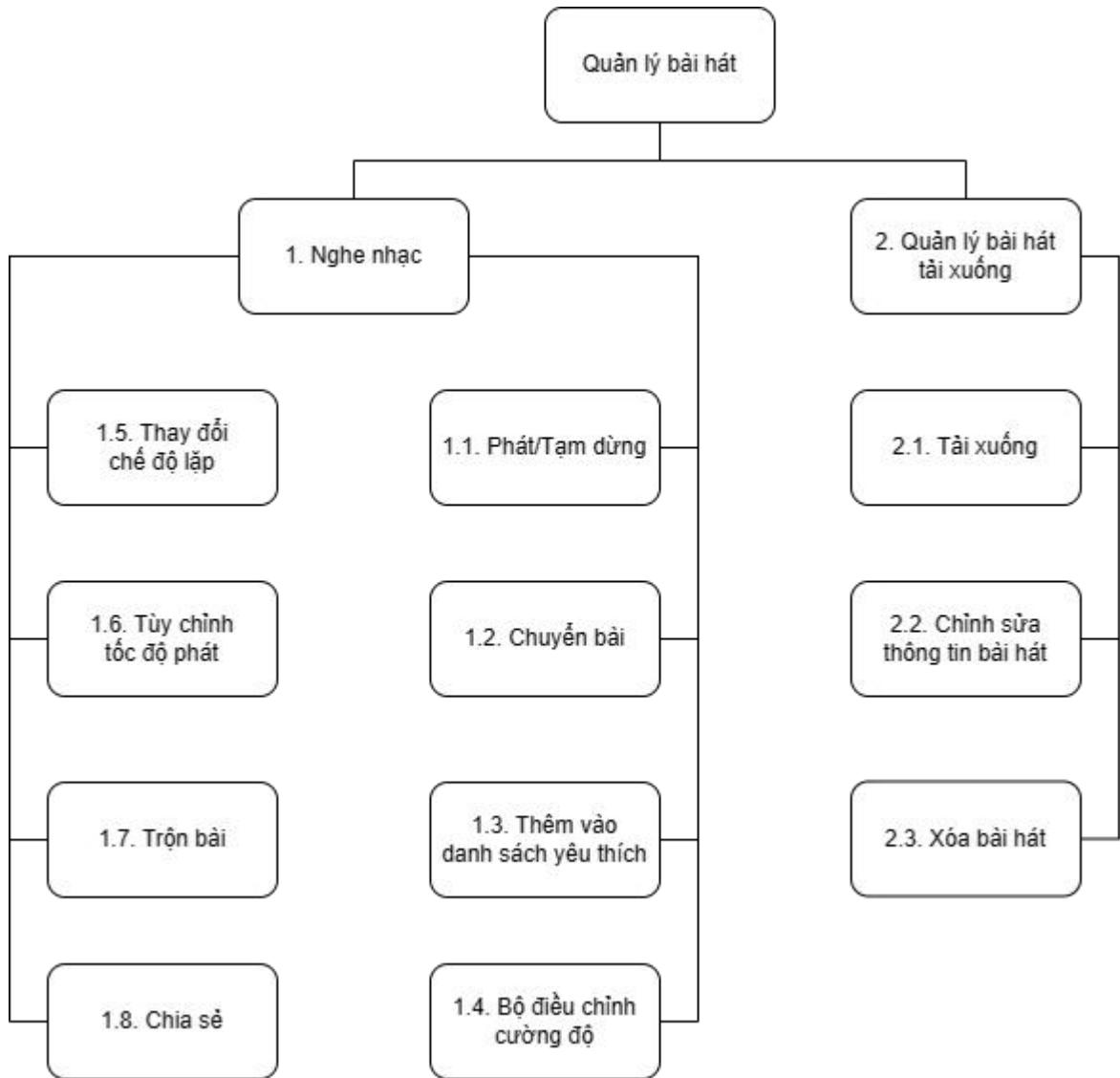
Bảng 4.1: Phân tích yêu cầu hệ thống

## 4.2 Thiết kế hệ thống

### 4.2.1 Sơ đồ phân rã chức năng BFD

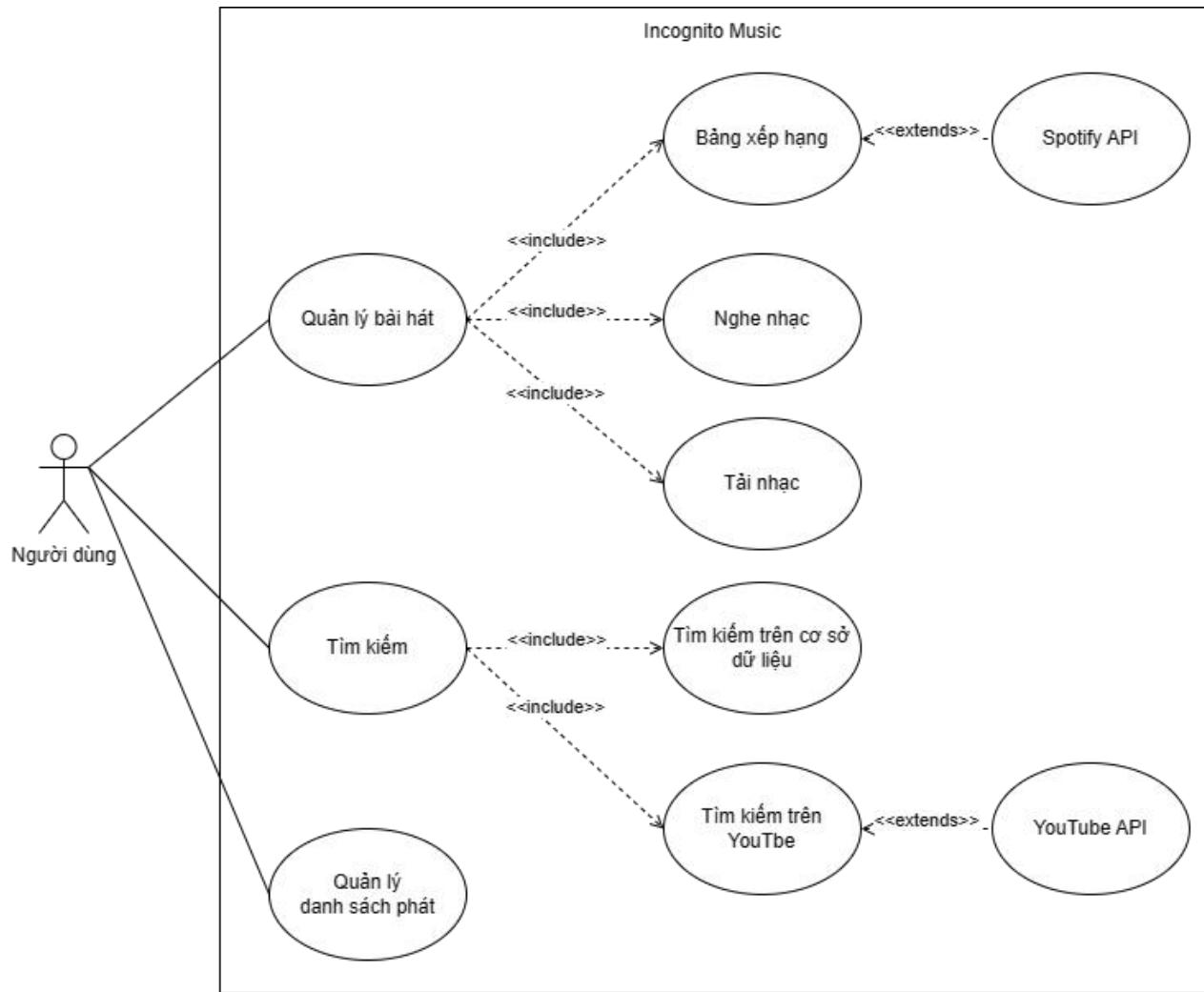


Hình 4.1: Sơ đồ phân rã chức năng tổng quát



Hình 4.2: Sơ đồ phân rã chức năng Quản lý bài hát

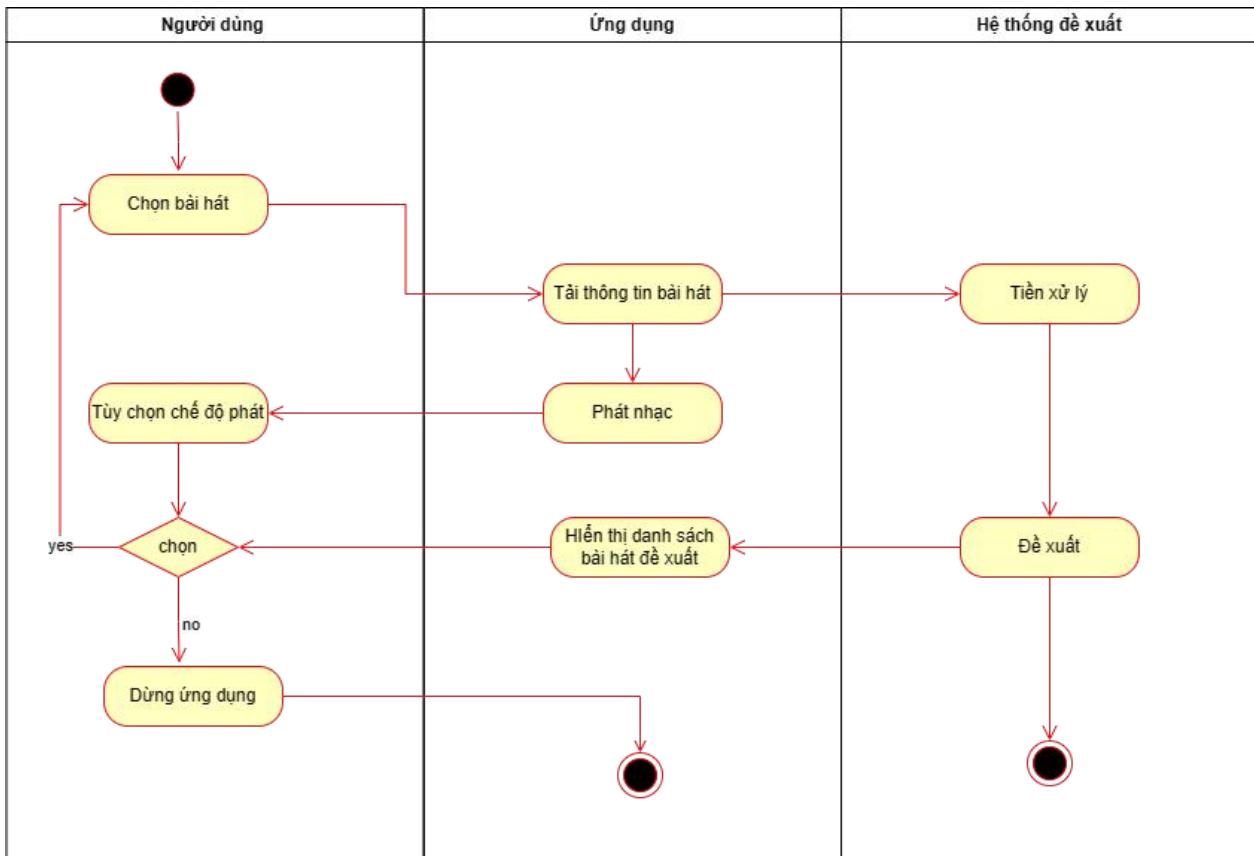
#### 4.2.2 Sơ đồ Use Case



Hình 4.3: Sơ đồ Use Case tổng quát

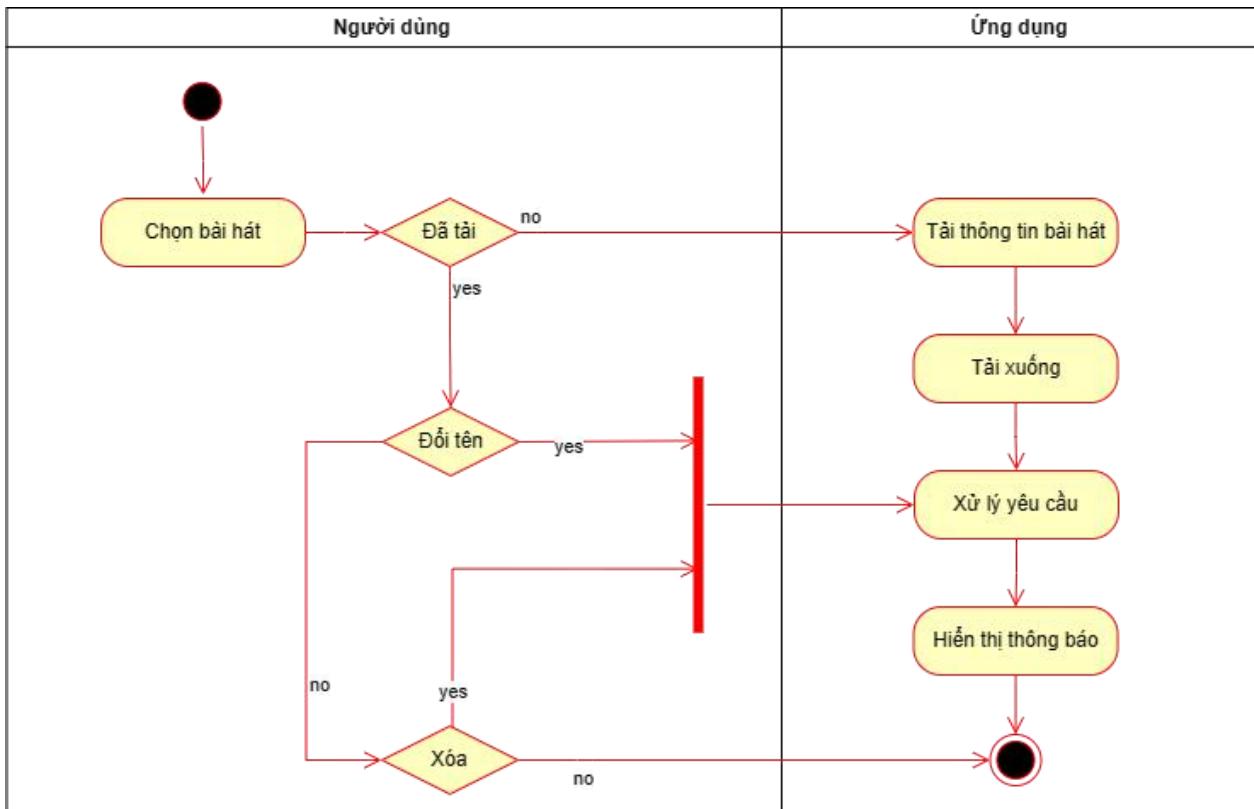
#### 4.2.3 Sơ đồ hoạt động

a) Nghe nhạc



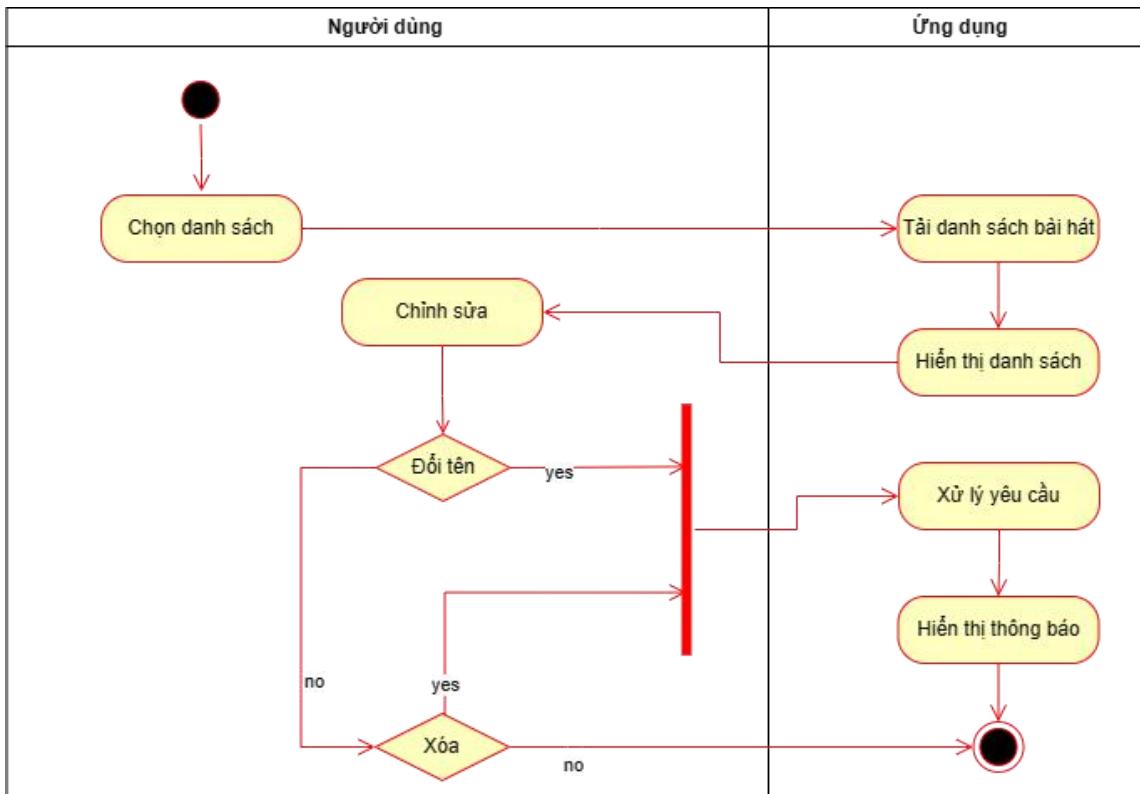
Hình 4.4: Sơ đồ hoạt động chức năng Nghe nhạc

b) Tải nhạc



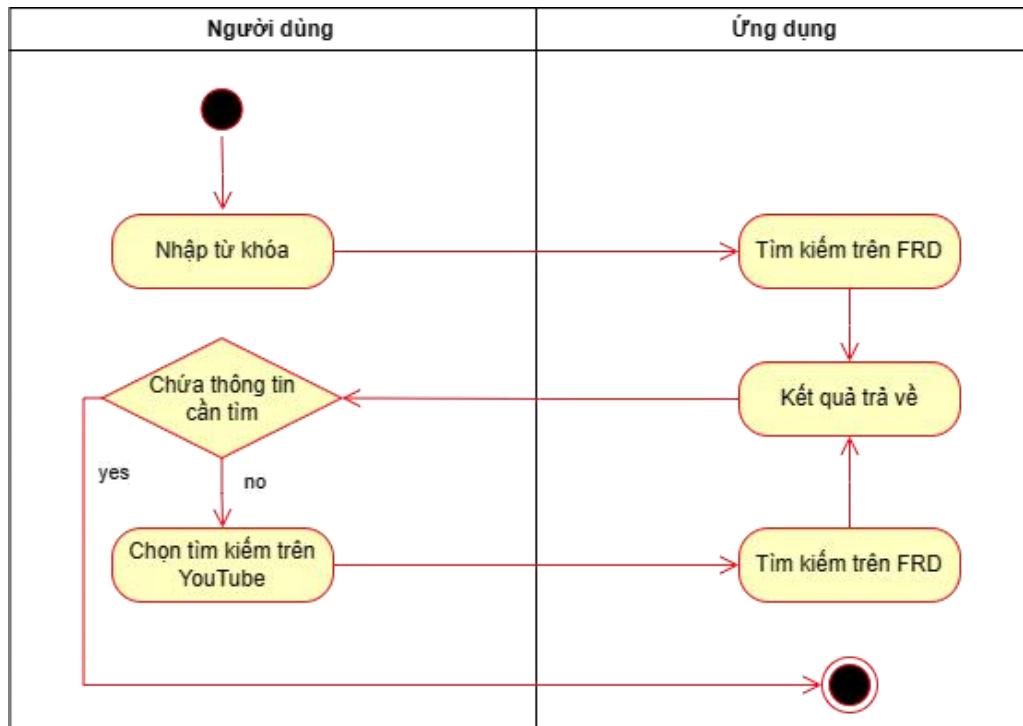
Hình 4.5: Sơ đồ hoạt động chức năng Tải nhạc

c) Quản lý danh sách phát



Hình 4.6: Sơ đồ hoạt động chức năng Quản lý danh sách phát

#### d) Tìm kiếm



Hình 4.7: Sơ đồ hoạt động chức năng Tìm kiếm

### 4.3 Cơ sở dữ liệu

Vì sử dụng NoSQL nên các thực thể dưới đây chỉ mô tả tên trường và vai trò của trường đó đối với ứng dụng.

#### a) Thực thể Bài hát

Tên trường	Chú thích
320kbps	Phát nhạc với chất lượng cao nhất
album	Tên Album, dùng trong RS
album_id	Mã Album, dùng để truy xuất dữ liệu
artist	Tên ca sĩ, dùng để hiển thị và xử lý dữ liệu trong RS
duration	Thời lượng bài hát
genre	Thể loại, dùng trong RS
Id	Mã bài hát, dùng để truy xuất dữ liệu
Image	Đường dẫn của ảnh, hiển thị trên giao diện
Language	Ngôn ngữ, dùng trong RS
Perma_id	Đường dẫn đến bài hát, dùng để chia sẻ bài hát
Release_date	Ngày ra mắt bài hát
Subtitle	Tiêu đề phụ, dùng để xử lý dữ liệu trong RS
Title	Tên bài hát, hiển thị và xử lý dữ liệu trong RS
Url	Đường dẫn bài hát, dùng để phát nhạc
Year	Năm phát hành

Bảng 4.2: Thực thể Bài hát

b) Thực thể Người dùng

Tên trường	Chú thích
id	Mã tài khoản, dùng để truy vấn dữ liệu
name	Tên tài khoản, dùng để hiển thị
timeZone	Múi giờ
accountCreateOn	Thời điểm tạo tài khoản

Bảng 4.3: Thực thể Người dùng

c) Thực thể Thông số

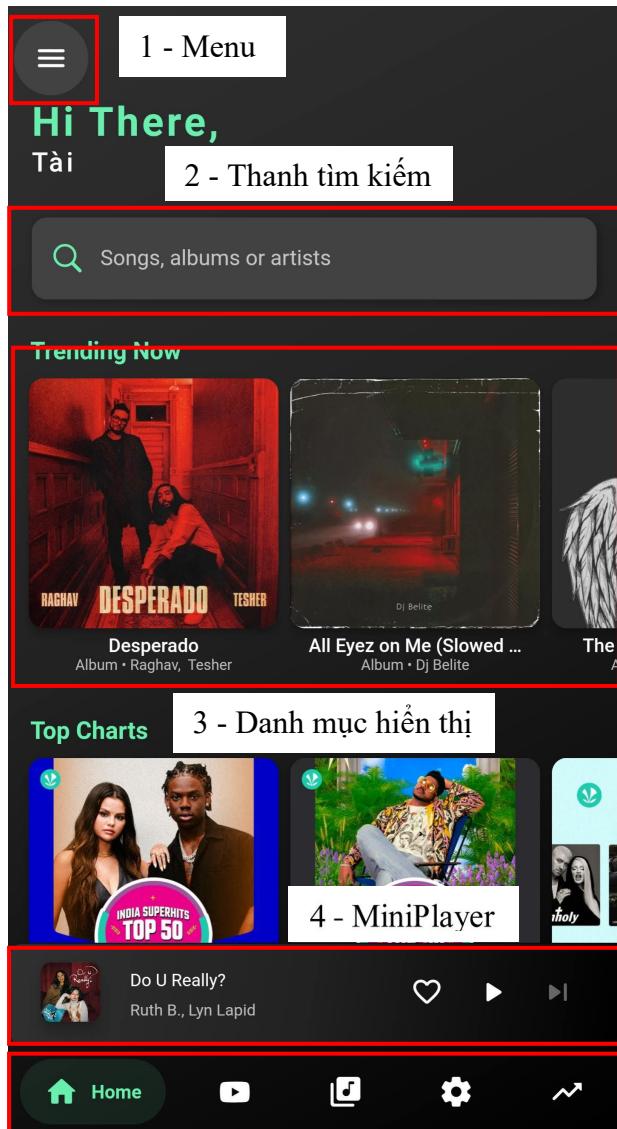
Tên trường	Chú thích
songId	Mã bài hát, dùng để ánh xạ dữ liệu
userId	Mã người dùng, dùng để ánh xạ dữ liệu
listenCount	Số lượt nghe của người dùng với bài hát tương ứng, dùng để tính toán và đánh giá mức độ tương tự trong RS

Bảng 4.4: Thực thể Thông số

# CHƯƠNG 5. XÂY DỰNG ỨNG DỤNG

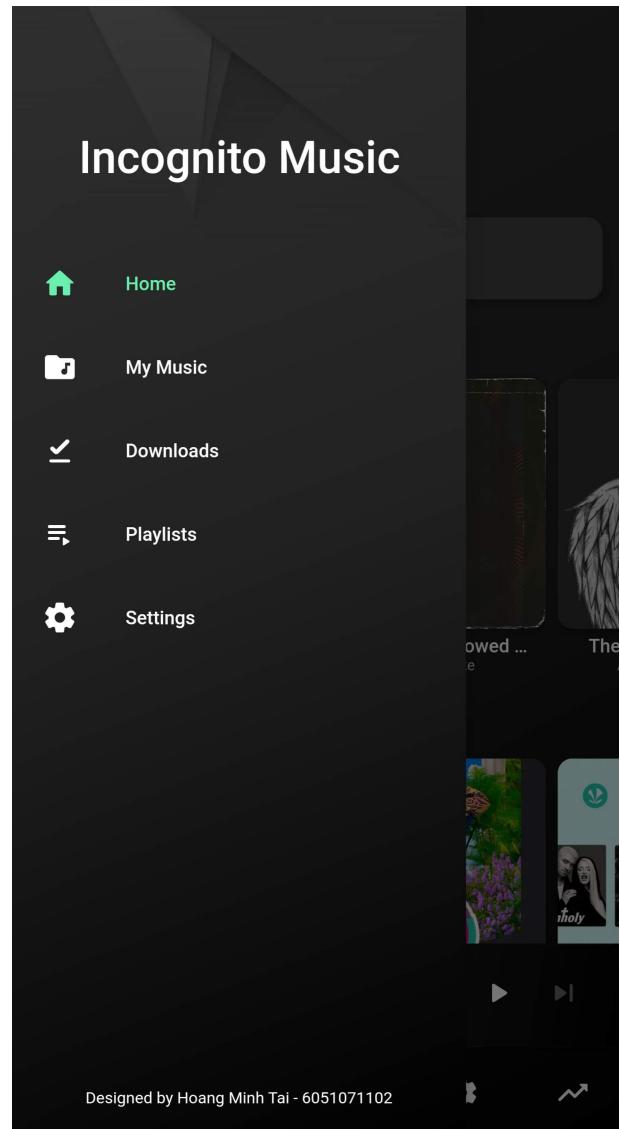
## 5.1 Giới thiệu

### 5.1.1 Trang chủ



5 - Các Tab truy cập nhanh

Hình 5.1: Màn hình trang chủ



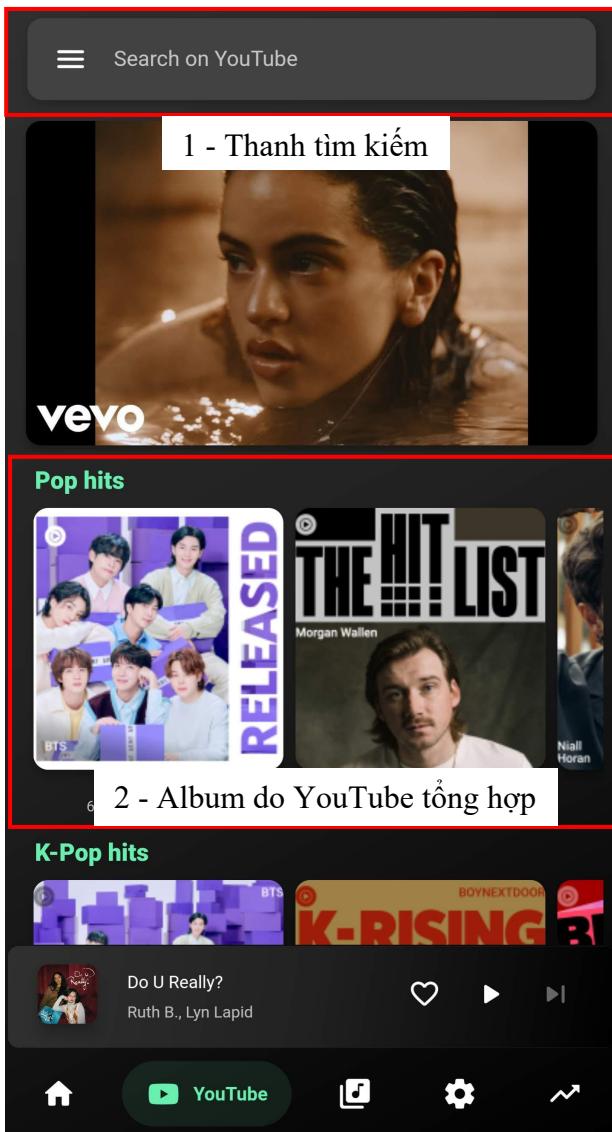
Hình 5.2: Thanh sidetab

Màn hình chính ứng dụng bao gồm những thông tin quan trọng sau:

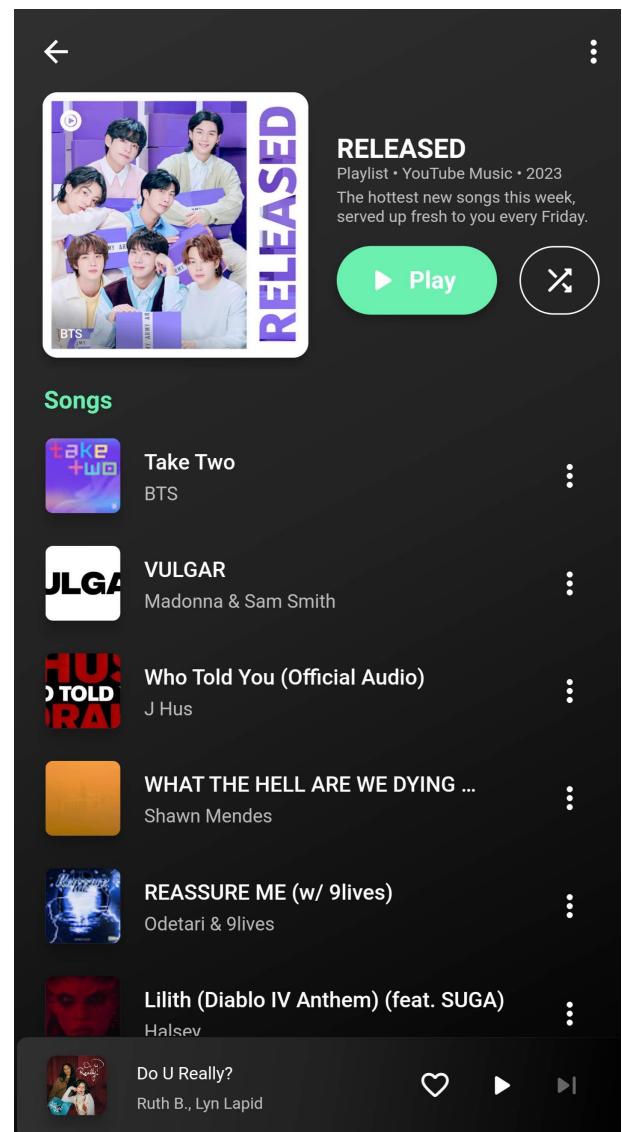
1. Nút Menu: Khi nhấn vào sẽ hiển thị thanh SideTab như hình 5.2, ở đó người dùng có thể chuyển hướng đến các màn hình khác nhau mà mình muốn.
2. Thanh tìm kiếm: Sử dụng để tìm kiếm trên cơ sở dữ liệu, sẽ chuyển hướng đến trang tìm kiếm khi nhấn vào.
3. Danh mục hiển thị: Sẽ xuất hiện nhiều danh mục khác nhau trên trang chủ, mục đích là để hiển thị thông tin về những bài hát theo từng chủ đề khác nhau. Hệ thống để xuất nội dung sẽ hiển thị danh sách các bài hát thịnh hành ở đây.
4. Miniplayer: Một bản thẻ nhỏ hơn của trang phát nhạc với các thông tin cơ bản về bài hát cùng với những cử chỉ tác động đến bài hát hiện đang phát như Tạm dừng, chuyển bài...
5. Các Tab truy cập nhanh: Có vai trò tương tự như thanh SideTab nhưng ở đây chuyển hướng tới những chức năng mà người dùng sẽ sử dụng nhiều.

Các mục 1, 4 và 5 là các widget riêng biệt và được xuất hiện trong nhiều trang với chức năng tương tự.

### 5.1.2 Màn hình YouTube



Hình 5.3: Màn hình YouTube



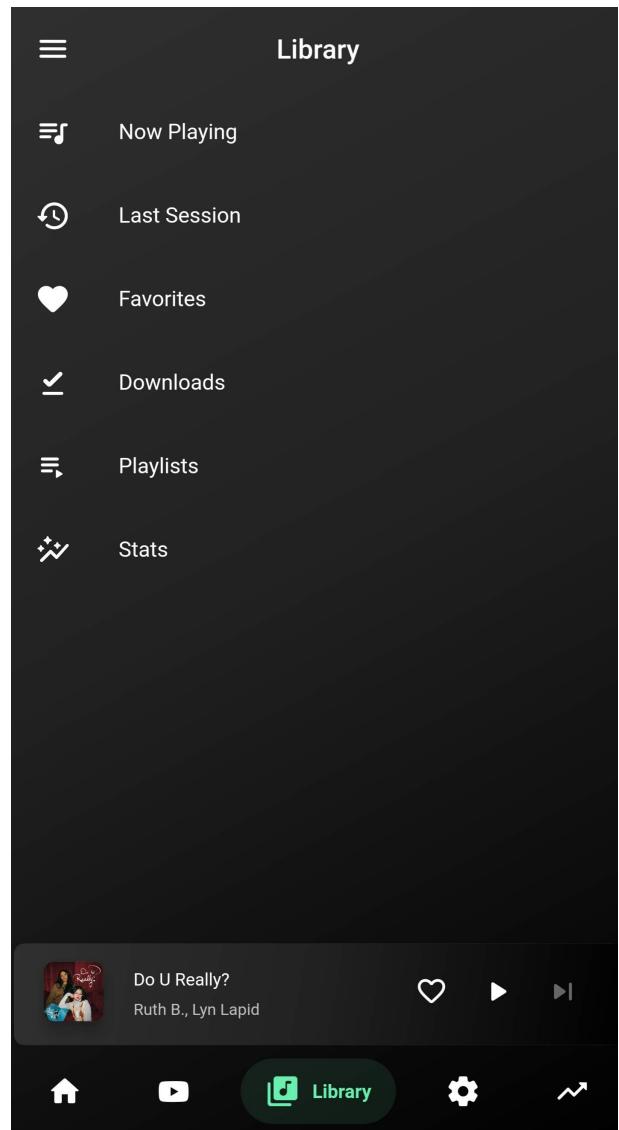
Hình 5.4: Thông tin Album của YouTube

- Màn hình này bao gồm:
- Thanh tìm kiếm: Sử dụng để tìm kiếm thông qua YouTube API, sẽ chuyển hướng đến trang tìm kiếm khi nhấn vào.
  - Album do YouTube tổng hợp: Sẽ xuất hiện nhiều album khác nhau được phân loại theo những nền âm nhạc nổi tiếng như K-Pop, US-UK... Chi tiết thông tin và các bài hát trong đó sẽ xuất hiện như hình 5.4 sau khi người dùng nhấn vào xem album.

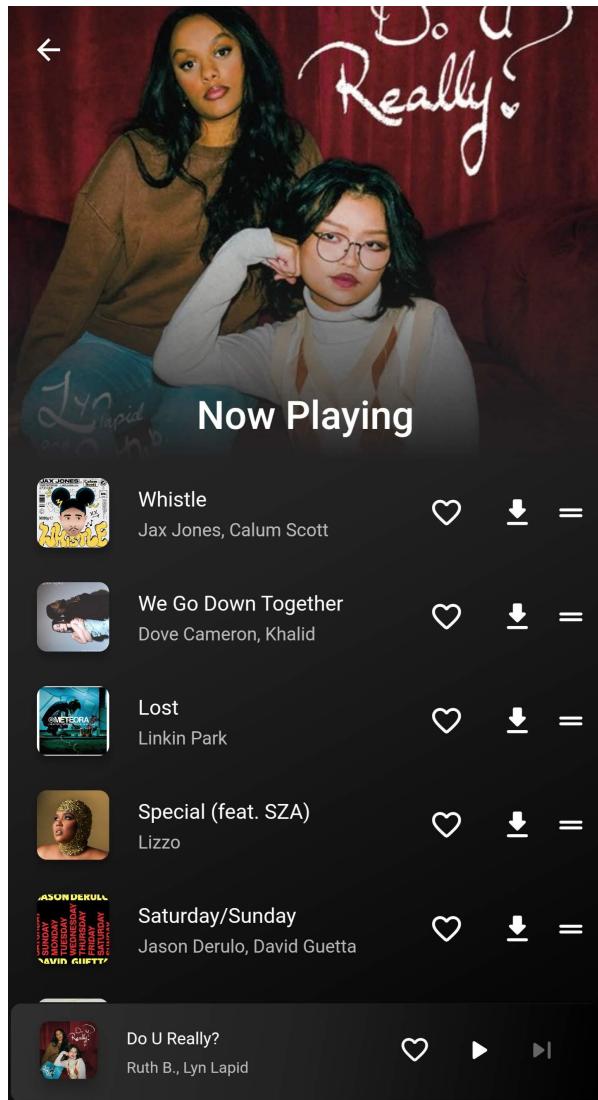
### 5.1.3 Thư viện

Màn hình này được dùng để quản lý các danh sách bài hát. Cụ thể:

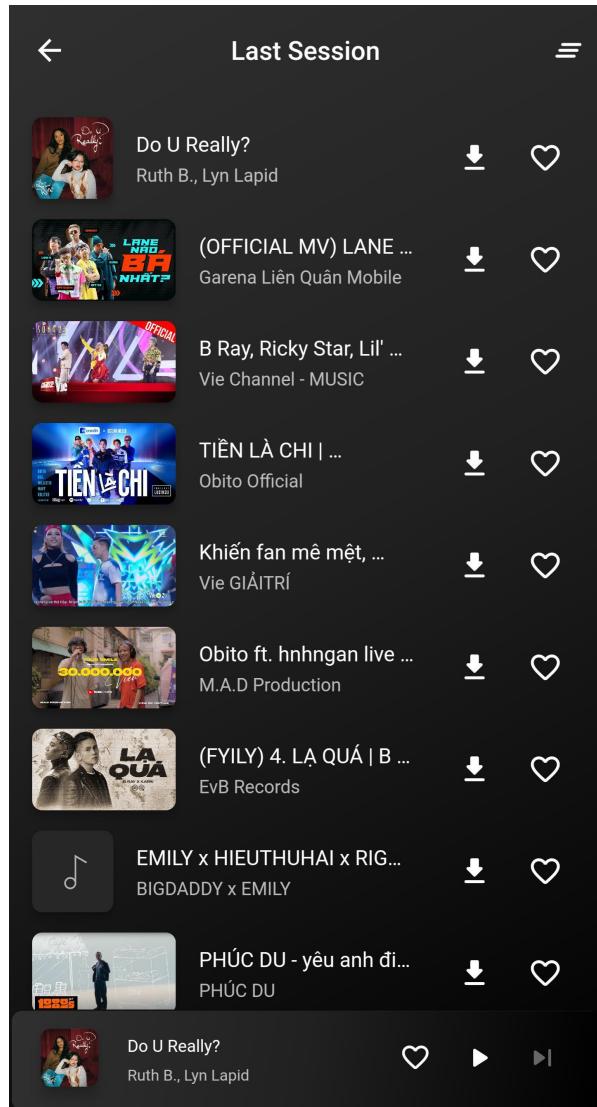
1. Danh sách đang phát: Khi ứng dụng đang phát 1 bài nhạc bất kì thì danh sách này sẽ hiển thị danh sách các bài hát liên quan dưới dạng hàng đợi chi tiết ở hình 5.6.
2. Danh sách nghe gần đây: Lưu lại lịch sử các bài hát được nghe gần nhất, chi tiết ở hình 5.7.
3. Danh sách bài hát yêu thích: Các bài hát được người dùng “thả tim”, chi tiết ở hình 5.8.
4. Danh sách các bài hát đã tải xuống
5. Quản lý danh sách phát: Tùy chỉnh các danh sách phát của người dùng, chi tiết ở hình 5.9.
6. Thông kê: Nêu ra số lượt nghe và bài hát được nghe nhiều nhất.



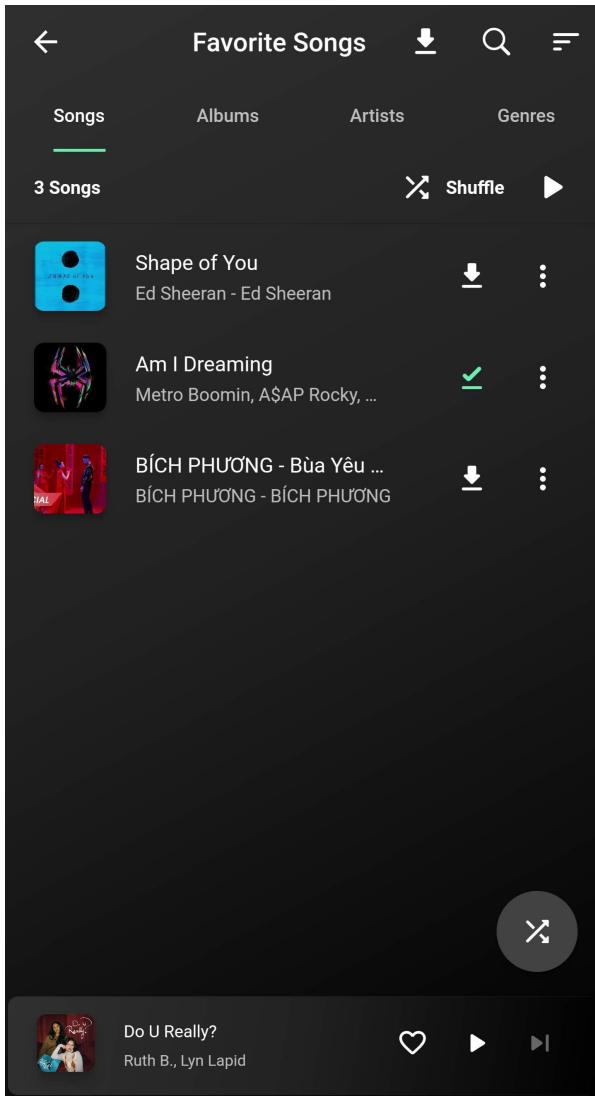
Hình 5.5: Màn hình Thư viện



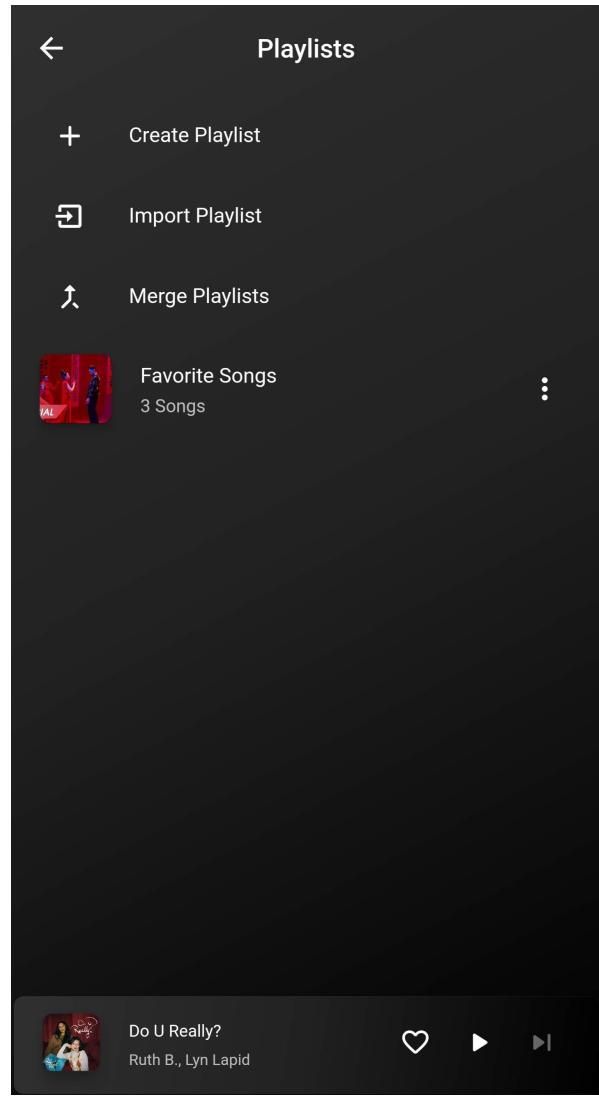
Hình 5.6: Danh sách đang phát



Hình 5.7: Danh sách nghe gần đây

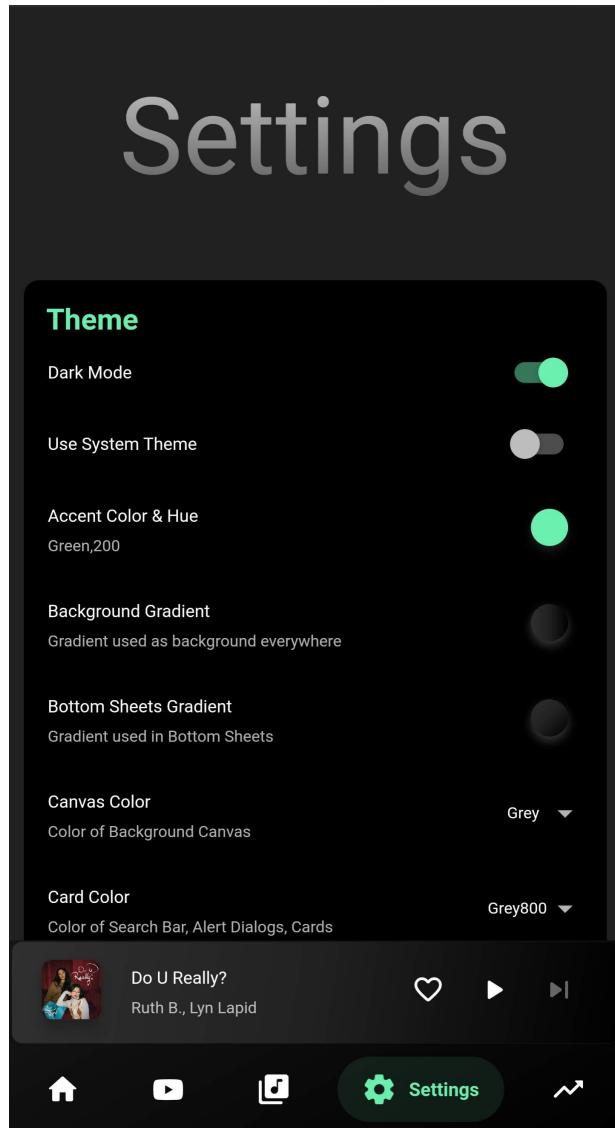


Hình 5.8: Danh sách bài hát yêu thích

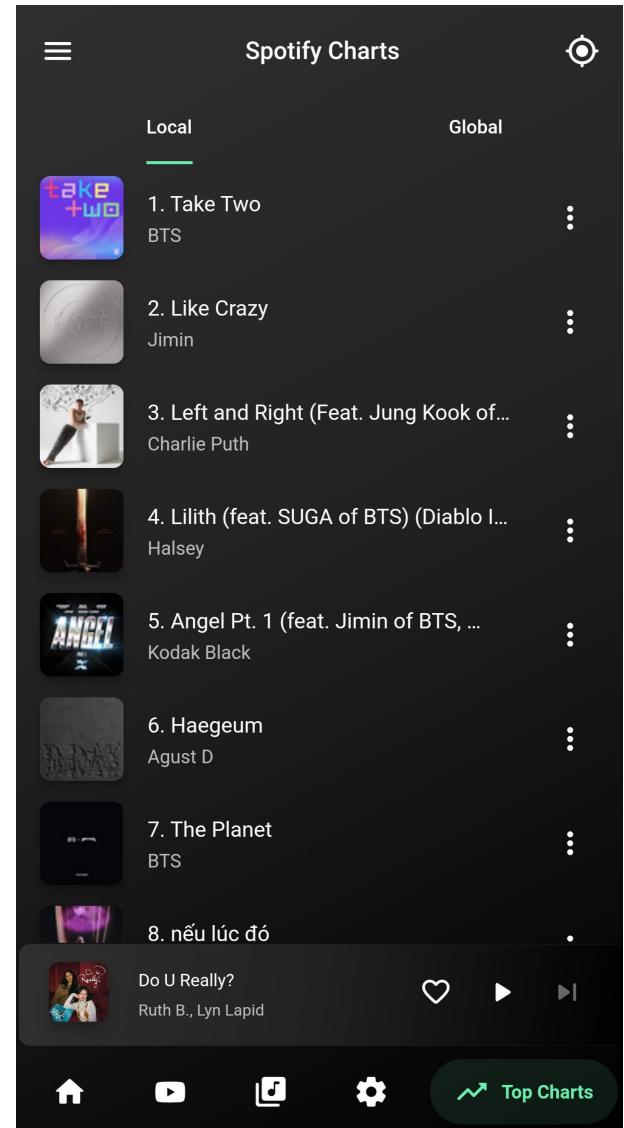


Hình 5.9: Quản lý các danh sách phát

### 5.1.4 Cài đặt và Bảng xếp hạng



Hình 5.10: Màn hình Cài đặt



Hình 5.11: Màn hình Bảng xếp hạng

1. Cài đặt: Thay đổi các cài đặt của hệ thống về hiển thị, các tác vụ liên quan đến việc nghe và tải nhạc và một vài tùy chọn hỗ trợ.
2. Bảng xếp hạng: Hiển thị danh sách các bài hát thịnh hành theo dữ liệu của Spotify .

## 5.2 Các chức năng quan trọng

### 5.2.1 Nghe nhạc

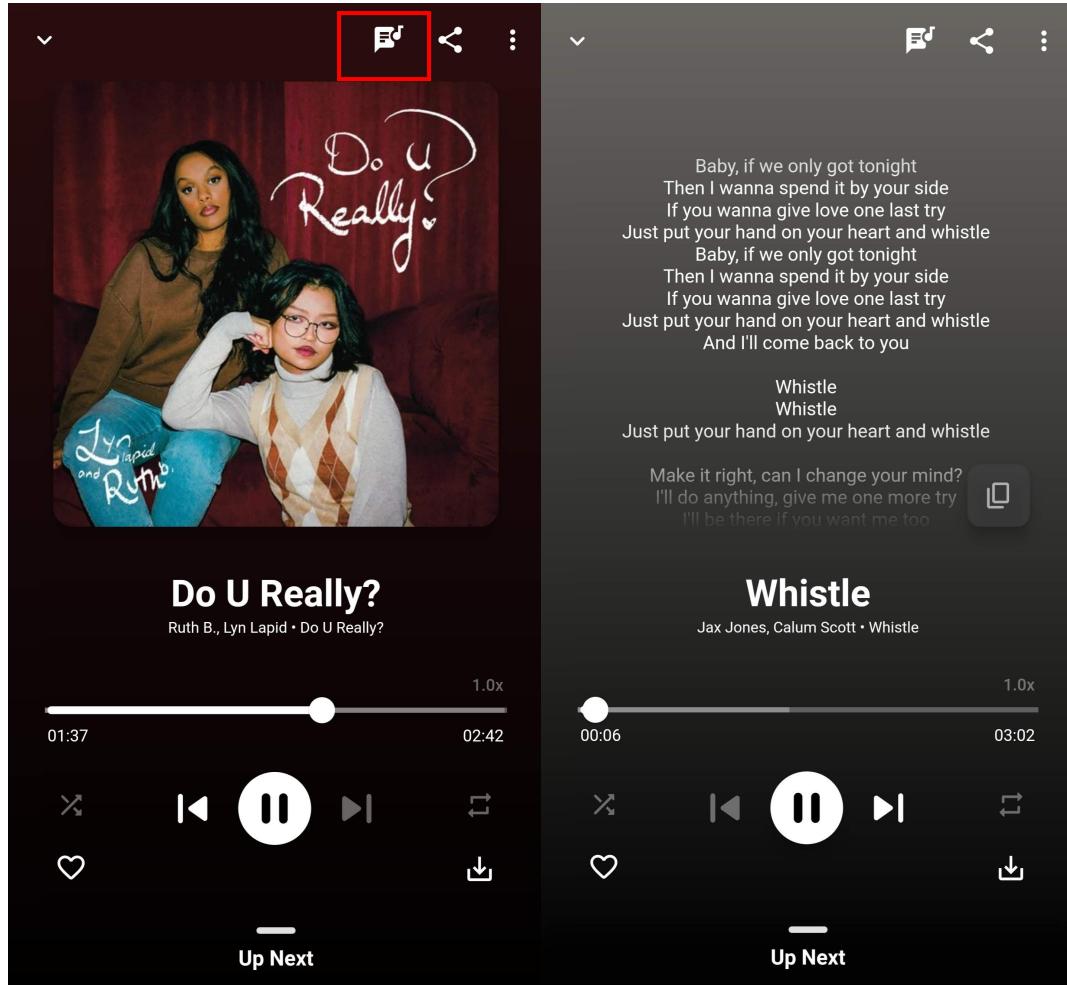
- a) Các bài hát sẽ được phát tiếp theo



Hình 5.12: Danh sách bài hát tương tự

Trong giao diện phát nhạc, khi nhấn vào nút được khoanh vùng để xem những ca khúc gợi ý sẽ hiển thị ra danh sách như hình 5.12 ở danh sách này người dùng có thể tùy chọn thứ tự phát và loại bỏ những bài hát mình không thích ra khỏi danh sách bằng việc vuốt vào các item.

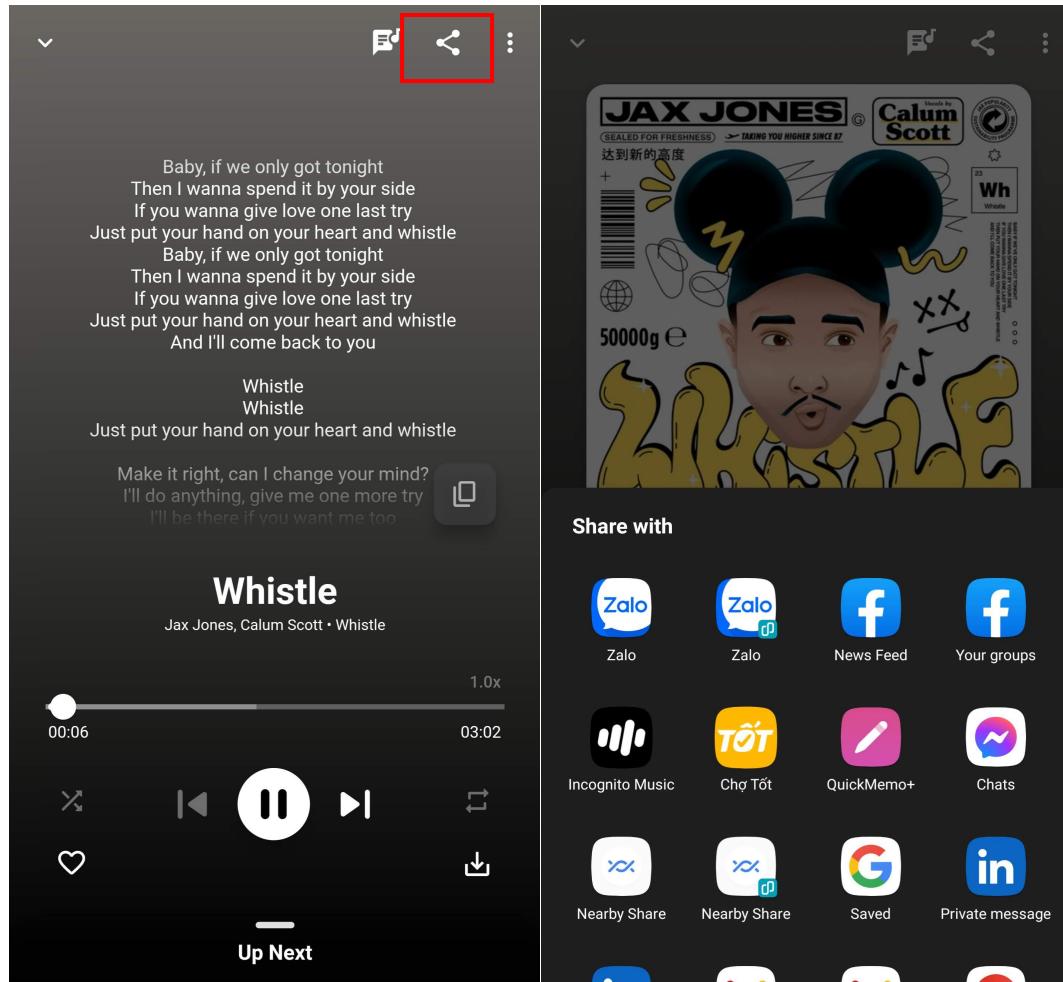
b) Xem lời bài hát



Hình 5.13: Xem lời bài hát

Để xem lyric người dùng nhấn vào nút được khoanh hoặc nhấn đúp vào chính giữa hình ảnh. Lúc này hệ thống sẽ truy vấn lời bài hát và hiển thị lên giao diện, lưu ý sẽ có trường hợp không tìm thấy lời.

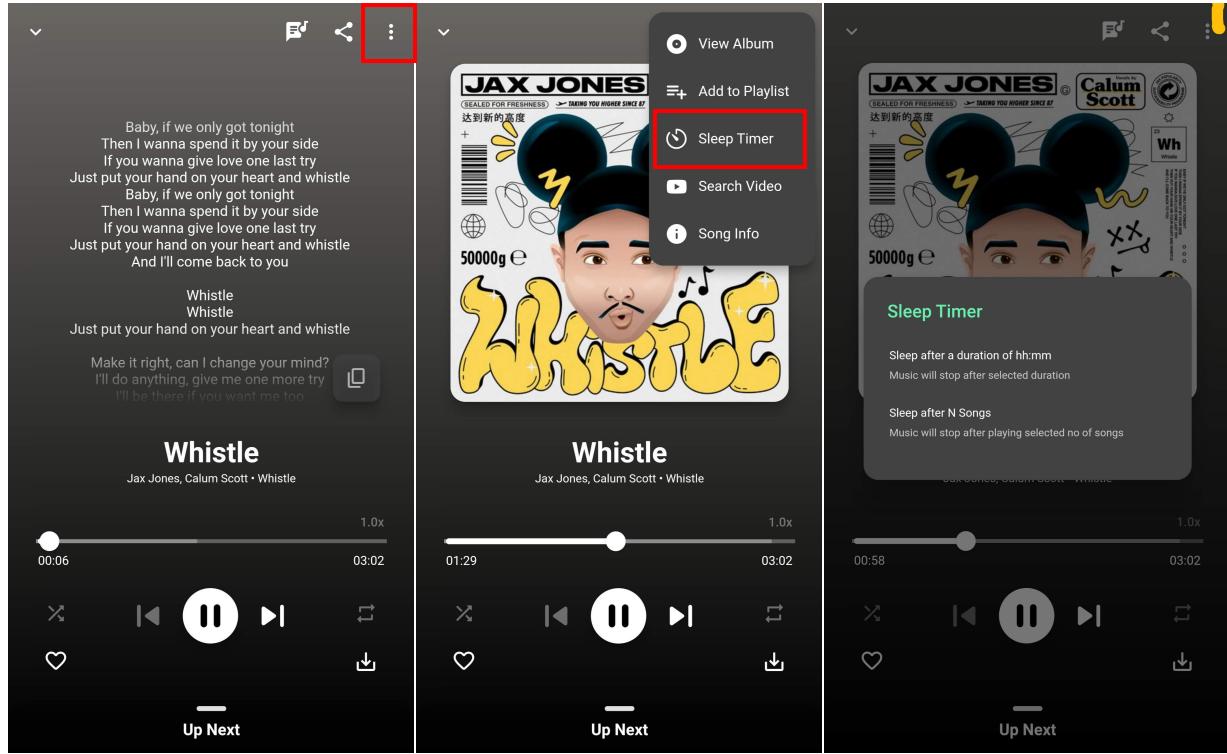
c) Chia sẻ bài hát



Hình 5.14: Chia sẻ bài hát

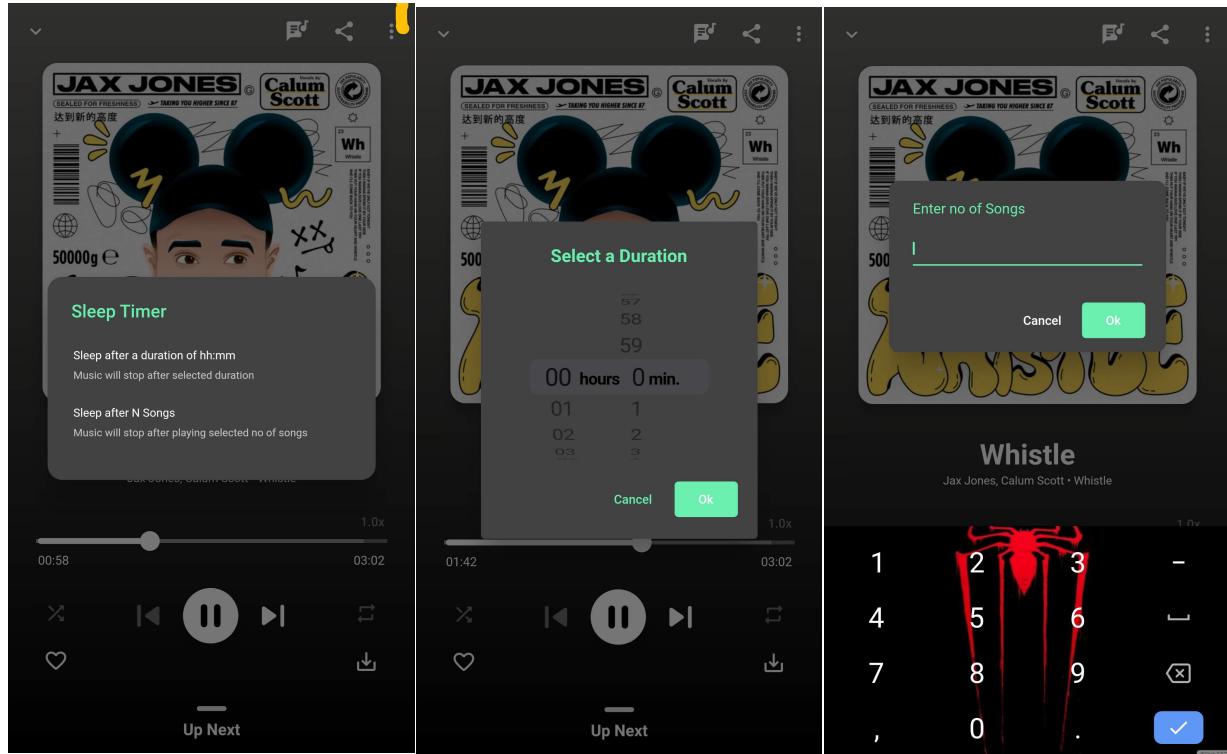
Chức năng chia sẻ bài hát lên mạng xã hội sẽ được kích hoạt khi người dùng nhấp vào nút được khoanh và lựa chọn người muốn chia sẻ. Lúc này hệ thống sẽ gửi cho người nhận đường dẫn phát ca khúc đó.

c) Hẹn giờ tắt nhạc



Hình 5.15: Hẹn giờ tắt nhạc

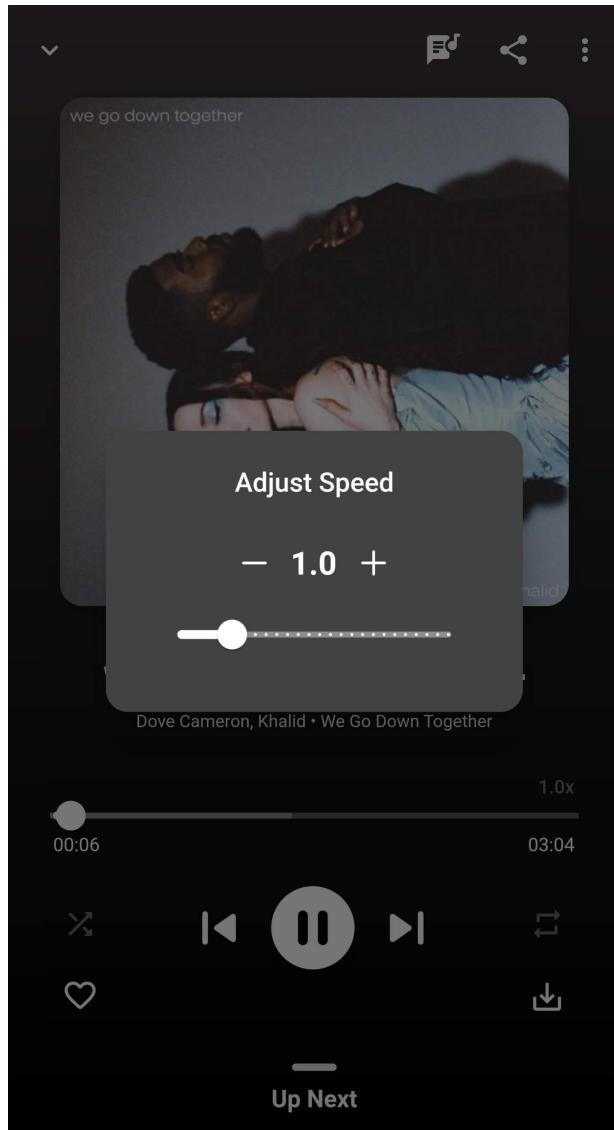
Thực hiện theo các nút được khoanh trên hình 5.15 từ trái sang phải ta được kích hoạt được tính năng hẹn giờ tắt nhạc. Tính năng này giúp người dùng kiểm soát được mức độ hao tốn tài nguyên của điện thoại đi kèm với việc nghe nhạc vừa đủ. Có hai phương thức hẹn giờ đó là hẹn theo một khoảng thời gian cụ thể và hẹn theo số lượng bài hát cụ thể.



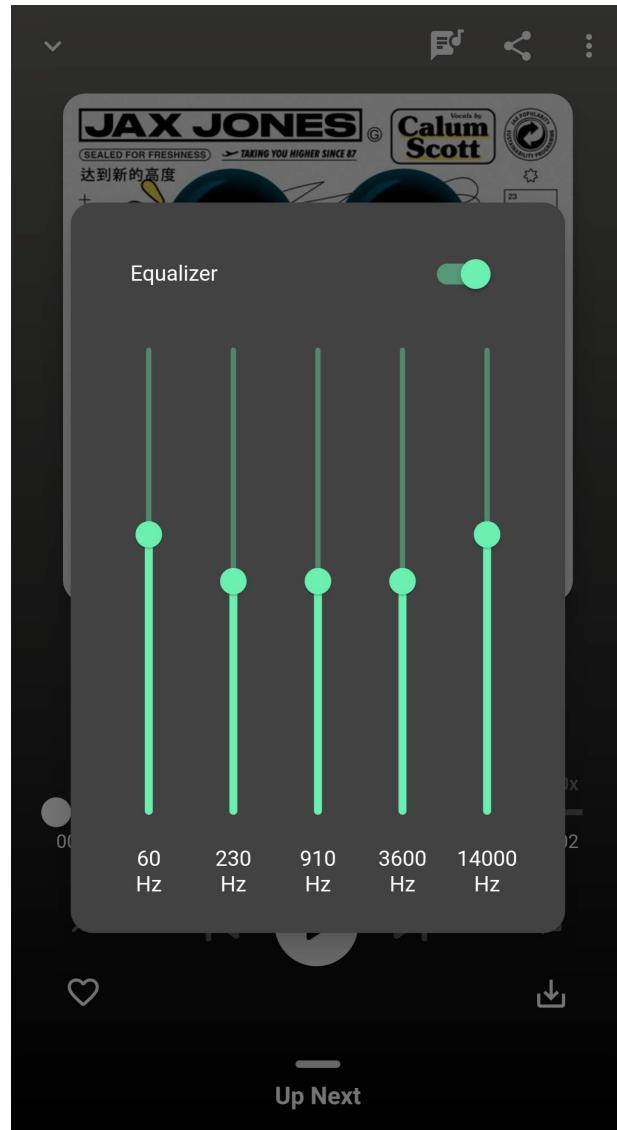
Hình 5.16: Lựa chọn phương thức hẹn giờ

Chọn một trong hai phương thức:

1. Chọn hẹn giờ với một khoảng thời gian cụ thể và điều chỉnh mức thời gian.
  2. Chọn hẹn giờ theo số lượng bài hát và nhập số lượng bài hát.
- d) Tùy chỉnh tốc độ phát và cường độ âm thanh



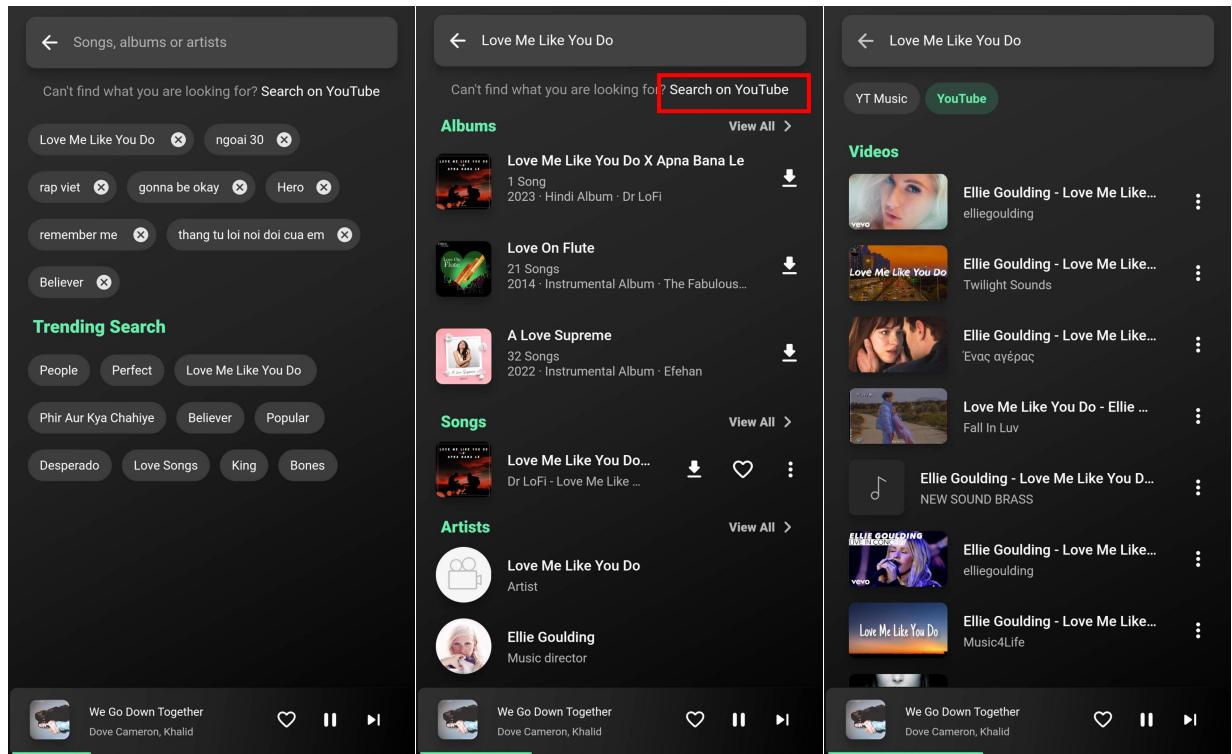
Hình 5.17: Tùy chỉnh tốc độ phát



Hình 5.18: Bộ điều khiển cường độ

Hai chức năng phục vụ chủ yếu theo nhu cầu nghe nhạc của mỗi cá nhân.

### 5.2.2 Tìm kiếm



Hình 5.19: Quá trình tìm kiếm bài hát

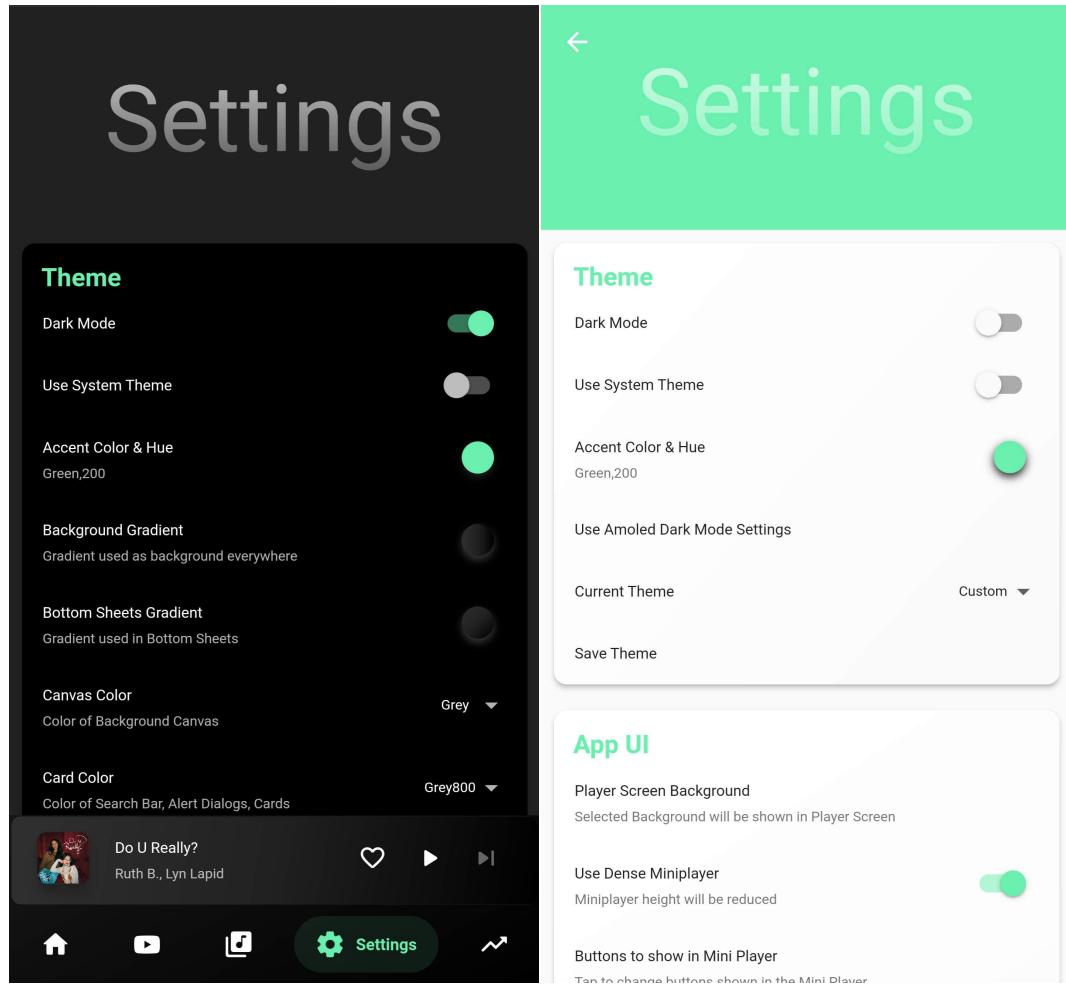
Ngoại trừ thanh tìm kiếm ở màn hình YouTube sẽ trả về thắng kết quả từ dữ liệu của YouTube, tất cả tính năng tìm kiếm đều thực hiện tuần tự như hình 5.19. Cụ thể:

1. Nhập từ khóa vào thanh tìm kiếm
2. Hệ thống trả về kết quả từ cơ sở dữ liệu trên FRD
3. Nếu người dùng chưa hài lòng với kết quả (do dữ liệu của hệ thống chưa đủ nhiều) có thể nhấn vào nút được khoanh ở hình 5.19.
4. Hệ thống sẽ dùng API và trả về kết quả từ dữ liệu của YouTube - một nguồn dữ liệu vô cùng phong phú không chỉ về nhạc mà là về đa phương tiện nói chung.

### 5.2.3 Tùy chỉnh cài đặt

Có hai tính năng tiêu biểu sẽ có nhiều người dùng tới đó là đổi màu nền và đổi ngôn ngữ cho ứng dụng của mình.

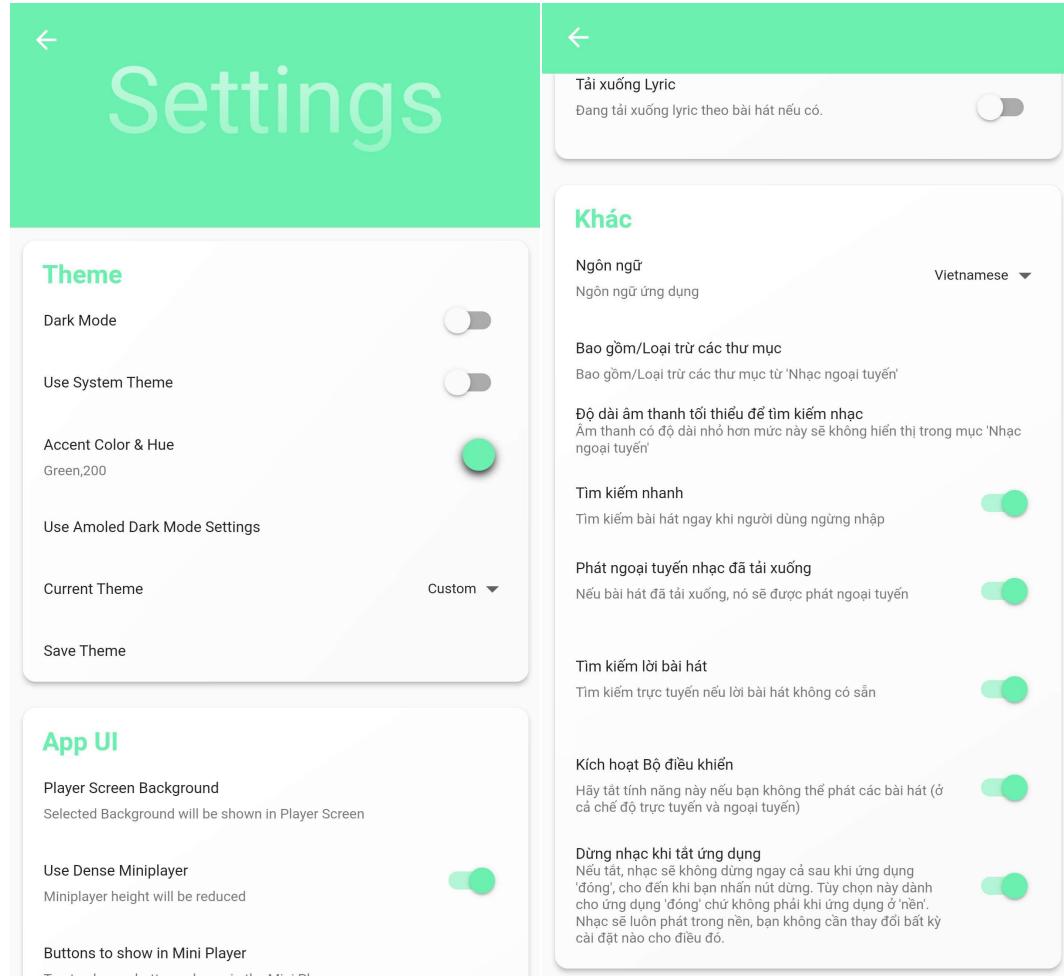
#### a) Đổi nền



Hình 5.20: Tùy chọn thay đổi nền

Ví dụ ở hình 5.20, một công tắc dùng để chuyển đổi giữa chế độ sáng và tối khác nhau, ngay lập tức nền của ứng dụng đã thay đổi phụ thuộc vào lựa chọn của người dùng - mặc định sẽ là màu tối.

b) Đổi ngôn ngữ



Hình 5.21: Tùy chọn thay đổi ngôn ngữ

Chúng ta có thể thấy khi đổi ngôn ngữ sang Vietnamese thì văn bản trong ứng dụng đã chuyển đổi theo.

## **CHƯƠNG 6. KẾT LUẬN VÀ KIẾN NGHỊ**

### **6.1 Kết quả đạt được**

Trong quá trình làm đồ án, em đã học được rất nhiều kiến thức mới cũng như củng cố được những kiến thức cũ. Kết quả em đạt được :

- Làm quen với việc sử dụng Flutter để thiết kế và phát triển được một ứng dụng đáp ứng ổn các chức năng yêu cầu đề ra.
- Biết cách sử dụng Python xây dựng hệ thống để xuất nội dung, tích hợp các thuật toán để xử lý dữ liệu và tạo API.
- Học và vận dụng thành công NoSQL vào một bài toán thực tế.
- Nâng cao thêm khả năng xử lý logic và giải quyết vấn đề trong lập trình.

### **6.2 Kiến nghị**

Trong tương lai, em có một vài ý tưởng để cải tiến ứng dụng của mình. Cụ thể:

- Vận dụng thêm kĩ thuật collaboration filtering vào hệ thống để xuất nội dung.
- Phát triển chức năng cho người dùng đăng tải bài hát của chính mình.
- Phát triển hệ thống theo hướng một hệ thống giải trí đa nền tảng.

## PHỤ LỤC

### Phụ lục 1: Hướng dẫn cài đặt

- Nhấn vào [link](#) này để cài đặt
- Quét mã QR để cài đặt

Lưu ý: Đăng nhập bằng tài khoản gmail do trường cung cấp



### Phụ lục 2: Version Source Control :

[IncognitoMusic](#)



## TÀI LIỆU THAM KHẢO

- [1]. [Online], “Flutter Tutorial - Tutorials point”, Available: [“https://www.tutorialspoint.com/flutter/flutter\\_tutorial.pdf”](https://www.tutorialspoint.com/flutter/flutter_tutorial.pdf) [Accessed 18/02/2023].
- [2]. Marine Chemeque-Rabe, *Content-based music recommendation system: A comparison of supervised Machine Learning models and music features*, STOCKHOLM, SWEDEN, 2020.
- [3]. [Online], “Machine Learning cơ bản”, Available: [“https://machinelearningcoban.com/2017/05/17/contentbasedrecommendersys/”](https://machinelearningcoban.com/2017/05/17/contentbasedrecommendersys/), [Accessed 08/05/2023].
- [4]. [Online], “Recommendation Overview”, Available: [“https://developers.google.com/machine-learning/recommendation/overview”](https://developers.google.com/machine-learning/recommendation/overview), [Accessed 20/04/2023].