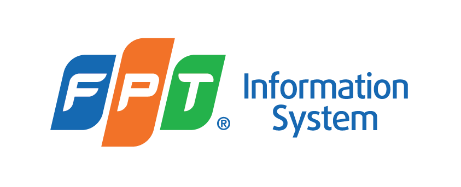
VU XUAN KHANH

FPT INFORMATION SYSTEM | PHẠM HÙNG, KEANGNAM, TỪ LIÊM, HÀ NỘI |

Tài liệu

Music similar



Contents

[I. Giới thiệu 2](#_Toc110584869)

[II. Tổng quan hệ thống 3](#_Toc110584870)

[1. Cấu hình hệ thống 3](#_Toc110584871)

[1.1. Cơ sở dữ liệu 3](#_Toc110584872)

[1.2 Cấu hình Minio 3](#_Toc110584873)

[2. Chức năng của hệ thống 4](#_Toc110584874)

[2.1 Fingerprinting 4](#_Toc110584875)

[2.2 Nhận diện âm thanh 4](#_Toc110584876)

[2.3 API nhận diện âm thanh 5](#_Toc110584877)

[3. Hướng dẫn sử dụng 5](#_Toc110584878)

[3.1 Fingerprinting 5](#_Toc110584879)

[3.2 Tìm âm thanh tương đồng 5](#_Toc110584880)

[3.3 Chạy API 5](#_Toc110584881)

[III. Kết luận 6](#_Toc110584882)

# I. Tổng quan dự án

## 1. Giới thiệu

Music Similar là một ứng dụng giúp phát hiện bài hát tương đồng. Nó có thể ghi nhớ âm thanh bằng cách nghe một lần và thực hiện *fingerprint* bản âm thanh đó. Sau đó, bằng cách phát một đoạn nhạc hay đọc đoạn âm thành từ bộ nhớ, Music Similar sẽ so khớp âm thanh với *fingerprint* được lưu trữ trong cơ sở dữ liệu và trả lại bài hát tương đồng với bài đang được phát.

## 2. Hướng tiếp cận

### a) Fingerprint

Diagram

Description automatically generated

Hình 1: Quá trình fingerprint âm thanh

Các bước thực hiện:

B1: Danh sách các bài hát được tải lên Object Storage thông qua 2 cách:

* Tải lên thủ công
* Sử dụng Scheduler table để lưu trữ thông tin bài hát sau đó Youtube Auto Crawl sẽ thu nhận thông tin này để tự động crawl các bài hát cần thiết.

B2: Các file âm thanh sẽ được cho qua quá trình Fingerprint để trích xuất các đặc trưng fingerprint.

B3. Các đặc trưng fingerprint được lưu trữ trong Fingerprint DB.

### b) Tìm bài hát tương đồng

Diagram

Description automatically generated

Hình 2: Quá trình tìm kiếm âm thanh tương đồng

Các bước thực hiện:

B1: Thu nhận thông tin bài hát cần kiểm tra có thể thực hiện bằng 2 cách:

* Nhập thủ công đường dẫn của các channel cần kiểm tra.
* Liệt kê danh sách bài hát cần kiểm tra, cho qua hệ thống Youtube Auto Find Channel để tự động tìm kiếm các channel.

B2: Các thông tin về channel sẽ được lưu trữ trong Scheduler table

B3: Nhận thông tin từ Scheduler table thực hiện tự động tải về các bài hát và lưu trữ bài hát thu được trong Object Storage.

B4: Thực hiện fingerprint các bài hát cần kiểm tra và so khớp âm thanh với *fingerprint* được lưu trữ trong cơ sở dữ liệu và trả lại bài hát tương đồng với bài hát cần kiểm tra.

# II. Phương pháp giải quyết

## 1. Cấu hình hệ thống

Music Similar sử dụng cơ sở dữ liệu để lưu trữ các thông tin metadata của các đoạn âm thanh giúp định danh các đoạn âm thanh với nhau. Ngoài ra, Music Similar còn sử dụng Minio như một kho dữ liệu giúp lưu trữ các đoạn âm thanh thô chờ được *fingerprinting* hoặc nhận diện.

### 1.1 Cơ sở dữ liệu

Cấu hình được khai báo trong file **db.cnf.SAMPLE**

A screenshot of a computer screen

Description automatically generated with medium confidence

Cấu hình bao gồm các thông tin về: host, user, pasword, database và loại cơ sở dữ liệu được sử dụng. Ở dự án này chúng tôi sử dụng cơ sở dữ liệu PostgreSql.

Text

Description automatically generated

### 1.2 Cấu hình Minio

Cấu hình được khai báo trong file **minio.cnf.SAMPLE**

Graphical user interface, text

Description automatically generated

Cấu hình bao gồm các thông tin: endpoint, access\_key, secret\_key và secure.

Text

Description automatically generated

## 2. Chức năng của hệ thống

Hệ thống có 3 chức năng chính:

* Thực hiện Fingerprinting đoạn âm thanh
* Thực hiện nhận diện một đoạn âm thanh thông qua đọc file
* API nhận âm thanh đầu vào và trả ra bài hát tương đồng.

### 2.1 Fingerprint

Các đoạn âm thanh gốc được lưu trữ thông tin trong bảng **detail\_crawl** sẽ được lấy ra theo điều kiện status = 0 và được hệ thống trích xuất các đặc trưng *fingerprint* và lưu trữ *fingerprint* lại nhằm mục đích so khớp sau này.

### 2.2 Nhận diện âm thanh

Tương tự với nhiệm vụ *fingerprint,* các thông tin về âm thanh cần nhận diện được lấy từ bảng **detail\_crawl** với điều kiện *item\_type = 'check\_song' and status = '0'*

### 2.3 API nhận diện âm thanh

Bổ sung thêm chức năng cho mục 2.2, chúng tôi tạo ra một API cho phép bạn upload file trực tiếp và hệ thống sẽ trả ra kết quả về các đoạn âm thanh tương đồng. Việc tạo ra API này giúp người dùng có thể phát triển thêm hệ thống thu nhận âm thanh trực tiếp từ môi trường và gửi tới hệ thống để nhận diện.

## 3. Hướng dẫn sử dụng

### 3.1 Fingerprint

C1: Run base script: python main.py --hash 1

C2: Run bash script: ./hash.sh

### 3.2 Tìm âm thanh tương đồng

C1: Run bash script: python main.py --recognize 1

C2: Run bash script: ./recognize.sh

### 3.3 Chạy API

Run base script: uvicorn service:app --reload

Vào giao diện web: <http://127.0.0.1:8000/docs> để đọc các docs API

Graphical user interface, text, application, email

Description automatically generated

# III. Kết luận

Hệ thống Music Similar đã giúp tự động phát hiện các âm thanh tương đồng, giúp người dùng tìm kiếm các bài hát chỉ dựa vào một vài câu hát hoặc phát hiện các hành vi reup sử dụng lại nội dung âm thanh mà chưa được phép của chủ sở hữu.