**TRƯỜNG ĐẠI HỌC SÀI GÒN**

**KHOA CÔNG NGHỆ THÔNG TIN**

A blue and white logo

Description automatically generated

**PHÁT TRIỂN PHẦN MỀM MÃ NGUỒN MỞ**

**BÁO CÁO ĐỀ TÀI:**

PHÁT TRIỂN PHẦN MỀM SPOTIFY CLONE

GVHD: Từ Lãng Phiêu

SV:Ngô Phú Khang – 3119410180

TP. HỒ CHÍ MINH, THÁNG 5 NĂM 2025

[I. PHẦN GIỚI THIỆU 2](#_Toc197375925)

[ii. NỘI DUNG 2](#_Toc197375926)

[2.1 Mục tiêu 2](#_Toc197375927)

[2.2 Công cụ và thư viện sử dụng 2](#_Toc197375928)

[2.2.1 Python 2](#_Toc197375929)

[2.2.2 Sử dụng Framework web: Django 2](#_Toc197375930)

[2.2.3 Sử dụng venv 3](#_Toc197375931)

[2.2.4 Sử dụng thư viện youtube-search 4](#_Toc197375932)

[2.2.5 Sử dụng thư viện spotipy 4](#_Toc197375933)

[III. Thiết kế ứng dụng 6](#_Toc197375934)

[3.1. Chức năng phát nhạc 6](#_Toc197375935)

[3.2. Chức năng phát video âm nhạc 7](#_Toc197375936)

[3.3. Chức năng User tạo album, bài hát yêu thích 8](#_Toc197375937)

[3.4. Chức năng tìm kiếm 9](#_Toc197375938)

[IV. Kết quả 10](#_Toc197375939)

[V. Kết luận 11](#_Toc197375940)

I. PHẦN GIỚI THIỆU

Spotify là một nền tảng nghe nhạc trực tuyến hàng đầu thế giới, cho phép người dùng truy cập kho nhạc khổng lồ với hàng triệu bài hát, podcast và chương trình âm thanh từ khắp nơi trên thế giới. Được thành lập vào năm 2006 tại Thụy Điển, Spotify không chỉ cung cấp trải nghiệm âm nhạc cá nhân hóa dựa trên sở thích người dùng, mà còn hỗ trợ nhiều thiết bị khác nhau như điện thoại, máy tính, TV và loa thông minh. Với hai hình thức sử dụng miễn phí (kèm quảng cáo) và trả phí (Spotify Premium), nền tảng này mang đến sự linh hoạt và tiện lợi cho mọi đối tượng yêu nhạc.

ii. NỘI DUNG

# 2.1 Mục tiêu

Mục tiêu của việc xây dựng một phiên bản website mô phỏng Spotify bằng Python là nhằm tái hiện lại các chức năng cơ bản của một nền tảng nghe nhạc trực tuyến hiện đại, đồng thời áp dụng kiến thức lập trình web, xử lý dữ liệu và thiết kế giao diện người dùng vào thực tế. Thông qua dự án này, người học sẽ rèn luyện kỹ năng sử dụng các framework Python như Flask hoặc Django để xây dựng backend, kết nối cơ sở dữ liệu để quản lý người dùng, bài hát và danh sách phát, cũng như triển khai các tính năng phát nhạc, tìm kiếm và gợi ý nội dung. Ngoài ra, dự án còn hướng tới mục tiêu giúp người học hiểu rõ hơn về cách hoạt động của các dịch vụ stream nhạc phổ biến hiện nay, từ đó có thể phát triển thêm các tính năng nâng cao như đăng nhập, bảo mật, và gợi ý cá nhân hóa dựa trên hành vi người dùng.

# 2.2 Công cụ và thư viện sử dụng

## 2.2.1 Python

Ngôn ngữ lập trình chính được sử dụng để phát triển ứng dụng.

## 2.2.2 Sử dụng Framework web: Django

**Django** là một framework phát triển web mã nguồn mở được viết bằng ngôn ngữ Python, ra đời vào năm 2005. Mục tiêu chính của Django là giúp các lập trình viên xây dựng các website nhanh chóng, an toàn và có khả năng mở rộng cao. Với triết lý “**Don’t Repeat Yourself (DRY)”** – không lặp lại mã và **“batteries-included”** – tích hợp sẵn nhiều chức năng, Django cung cấp đầy đủ công cụ cần thiết để xây dựng một ứng dụng web hoàn chỉnh, từ quản lý cơ sở dữ liệu, xác thực người dùng, đến hệ thống admin tự động.

| **Ưu điểm** | **Khuyết điểm** |
| --- | --- |
| Tích hợp sẵn ORM, admin, authentication, bảo mật. | Hơi nặng, phức tạp cho người mới bắt đầu. |
| Phù hợp cho dự án lớn, có cấu trúc rõ ràng. | Ít linh hoạt hơn Flask trong một số trường hợp. |
| Có cộng đồng lớn, tài liệu phong phú. | Có thể "quá mức cần thiết" với các dự án nhỏ. |

## 2.2.3 Sử dụng venv

venv (viết tắt của *virtual environment*) là một môi trường ảo trong Python, cho phép tạo một không gian riêng biệt để cài đặt các thư viện và phụ thuộc cho một dự án cụ thể mà không ảnh hưởng đến hệ thống Python toàn cục trên máy tính.

| **Ưu điểm** | **Mô tả** |
| --- | --- |
| **Tránh xung đột thư viện** | Mỗi dự án có thể dùng phiên bản thư viện riêng, không ảnh hưởng lẫn nhau. |
| **Gọn nhẹ, tích hợp sẵn** | venv có sẵn trong Python 3.3 trở lên, không cần cài thêm. |
| **Quản lý môi trường tốt hơn** | Dễ dàng chia sẻ cấu hình qua requirements.txt, triển khai đơn giản. |
| **An toàn** | Không cài thư viện trực tiếp vào Python hệ thống, tránh làm hỏng môi trường toàn cục. |
| **Hỗ trợ rộng rãi** | Tích hợp tốt với các IDE như VS Code, PyCharm, và công cụ CI/CD. |

| **Khuyết điểm** | **Mô tả** |
| --- | --- |
| **Không có chức năng nâng cao** | Không hỗ trợ quản lý nhiều phiên bản Python như pyenv hoặc conda. |
| **Không chia sẻ được môi trường** | Các môi trường venv thường không di chuyển được giữa máy nếu không đóng gói đúng cách. |
| **Không tự động tạo khi vào thư mục** | Không có tính năng tự động kích hoạt khi thay đổi thư mục như pipenv hay conda có thể cung cấp. |
| **Không quản lý dependency tree phức tạp** | Không phát hiện xung đột giữa các phiên bản phụ thuộc như pipenv hoặc poetry. |

## 2.2.4 Sử dụng thư viện youtube-search

Thư viện youtube-search là một thư viện Python nhỏ gọn cho phép **tìm kiếm video trên YouTube mà không cần dùng YouTube Data API hoặc API key**. Nó sử dụng kỹ thuật **web scraping** để truy cập và phân tích kết quả tìm kiếm trực tiếp từ trang YouTube.

Nó có chức năng như tìm kiếm video theo từ khóa, trả về kết quả dạng Json hoặc python dictonary,..

| **Ưu điểm** | **Giải thích** |
| --- | --- |
| Không cần API Key | Có thể sử dụng ngay mà không cần đăng ký Google Cloud API. |
| Dễ sử dụng | Giao diện lập trình đơn giản, chỉ cần 1-2 dòng là có kết quả. |
| Nhẹ, không phụ thuộc nhiều | Thư viện nhỏ, cài đặt nhanh, không phụ thuộc vào Google API client. |
| Trả về kết quả rõ ràng | Kết quả dạng dictionary dễ xử lý trong Python. |
| **Khuyết điểm** | Giải thích |
| Phụ thuộc vào HTML của YouTube | Vì sử dụng web scraping, nếu YouTube thay đổi cấu trúc trang -> lỗi. |
| Không truy cập dữ liệu nâng cao | Không lấy được lượt like, dislike, thống kê, hay thông tin kênh chi tiết. |
| Không phân trang | Không hỗ trợ xem tiếp các trang sau (chỉ một đợt tìm kiếm). |
| Có thể chậm hoặc bị block | Nếu chạy nhiều truy vấn, có thể bị chặn bởi bot detection của YouTube. |

## 2.2.5 Sử dụng thư viện spotipy

Thư viện spotipy là một thư viện Python chính thức không phải do Spotify phát triển, dùng để tương tác với Spotify Web API. Thư viện này đơn giản hóa quá trình gửi và nhận dữ liệu từ Spotify, cho phép bạn:

* Tìm kiếm bài hát, nghệ sĩ, album, playlist
* Lấy thông tin và phân tích nhạc (audio features, audio analysis)
* Tạo, sửa, xóa playlist của người dùng
* Thêm, xóa bài hát khỏi playlist
* Điều khiển nhạc đang phát (pause/play/skip...) nếu bạn có **Spotify Premium**
* Truy cập dữ liệu cá nhân của người dùng (lịch sử nghe nhạc, top tracks,...)

| **Ưu điểm** | **Mô tả** |
| --- | --- |
| Giao diện dễ hiểu | API đơn giản, dễ đọc, phù hợp cho cả người mới học lập trình |
| Tương thích đầy đủ Spotify API | Hầu như mọi tính năng Spotify Web API đều được hỗ trợ |
| Hỗ trợ OAuth2 | Có thể truy cập dữ liệu cá nhân với quyền người dùng |
| Tự động xử lý token | Tự động làm mới access token, không cần viết lại mã xác thực |
| Tích hợp linh hoạt | Có thể dùng với Flask, Streamlit, Django, hoặc bất kỳ ứng dụng Python nào |
| **Khuyết điểm** | **Mô tả** |
| Bắt buộc phải có API Key | Không dùng được nếu bạn không đăng ký ứng dụng tại Spotify Developer |
| Một số tính năng yêu cầu Premium | Như điều khiển nhạc đang phát |
| Không có giao diện GUI | Đây là thư viện backend, không hỗ trợ hiển thị nhạc hay phát nhạc |
| Không xử lý nâng cao (AI, ML) | Không hỗ trợ gợi ý nâng cao, phải tự xây dựng thêm |

III. Thiết kế ứng dụng

# 3.1. Chức năng phát nhạc

A diagram of a person with text

Description automatically generated

# 3.2. Chức năng phát video âm nhạc

A diagram of a person with text

Description automatically generated

# 3.3. Chức năng User tạo album, bài hát yêu thích

**A diagram of a person with text

Description automatically generated**

# 3.4. Chức năng tìm kiếm

A diagram of a person with text

Description automatically generated

IV. kết quả

1. Chức năng phát nhạc
2. Chức năng phát video âm nhạc

A group of people standing in a room

Description automatically generated

1. Chức năng tạo user album, bài hát yêu thích

A screenshot of a computer

Description automatically generated

1. Chức năng tìm kiếm

A screenshot of a music album

Description automatically generated

V. Kết luận

Việc phát triển phần mềm Spotify Clone không chỉ giúp người học nắm vững kiến thức về lập trình web với Python và Django, mà còn rèn luyện kỹ năng xử lý dữ liệu, giao tiếp giữa frontend và backend, thao tác với cơ sở dữ liệu, cũng như triển khai giao diện người dùng thân thiện. Qua dự án này, nhóm/lập trình viên đã học cách kết hợp nhiều công nghệ như HTML, CSS, JavaScript, Django, SQLite/MySQL, đồng thời áp dụng quy trình phát triển phần mềm cơ bản như phân tích yêu cầu, thiết kế, xây dựng và kiểm thử.

Mặc dù ứng dụng vẫn còn đơn giản so với Spotify thực tế, nhưng đây là bước khởi đầu quan trọng, tạo nền tảng vững chắc để mở rộng tính năng như đăng ký người dùng, tìm kiếm bài hát, playlist cá nhân, streaming nâng cao, và tích hợp API. Dự án là minh chứng rõ ràng cho khả năng vận dụng kiến thức vào thực tế, góp phần nâng cao tư duy lập trình, khả năng thiết kế hệ thống, và triển khai phần mềm có tính ứng dụng cao.