**BỘ GIÁO DỤC VÀ ĐÀO TẠO TPHCM**

**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**

**Khoa CNTT CLC**

****

**BÁO CÁO**

**Bộ môn: NHẬP MÔN KHOA HỌC DỮ LIỆU**

**Đồ án 1: Thu thập dữ liệu**

**| GIẢNG VIÊN HƯỚNG DẪN |**

**TS Lê Ngọc Thành**

**Thành viên nhóm:**

Mạc Văn Hưng

Lữ Thế Vỹ

Nguyễn Ngọc Phước

Nguyễn Ngọc Uyên Trang

THÀNH PHỐ HỒ CHÍ MINH - THÁNG 11 NĂM 2021

# *Bảng phân công nhiệm vụ*

| **STT** | **Họ và Tên** | **MSSV** | **Nhiệm vụ** | **% hoàn thành** |
| --- | --- | --- | --- | --- |
| **1** | **Mạc Văn Hưng** | **19127416** | **Parse HTML** | **100** |
| 2 | Lữ Thế Vỹ | 19127009 | Parse HTML | 100 |
| 3 | Nguyễn Ngọc Phước | 19127519 | API | 100 |
| 4 | Nguyễn Ngọc Uyên Trang | 19127074 | API | 100 |

***Chúng em cam đoan đồ án này chúng em tự xây dựng và nghiên cứu không sao chụp bất kỳ nhóm nào.***

# *Mục lục*

[***Bảng phân công nhiệm vụ***](#_heading=h.gjdgxs) ***2***

[***Mục lục***](#_heading=h.30j0zll) ***3***

[***Phần 1: Parse HTML***](#_heading=h.ug4trx3jpn4o) ***4***

[Bảng phân công chi tiết](#_heading=h.esqnu6cn3aen) 4

[Quy trình thực hiện](#_heading=h.hggj5ypzn78) 4

[Chuẩn bị kiến thức](#_heading=h.ybq18bg6s97o) 4

[Xác định nơi khai thác dữ liệu](#_heading=h.2ffl21bb98g0) 5

[Giải thích quá trình thực hiện](#_heading=h.uwsg3xra52di) 5

[Cách 1: khai thác từ trang https://soundcloud.com/discover](#_heading=h.gfltixnqjiz1) 5

[Nhược điểm](#_heading=h.jsdkp3vf0542) 7

[Cách 2: khai thác từ trang https://soundcloud.com/search (cải tiến)](#_heading=h.7nhux1f3qb0n) 8

[Đánh giá dữ liệu](#_heading=h.5xf15aa4m0gr) 10

[Nhận xét](#_heading=h.qzg0qwfo4lh1) 10

[***Phần 2: API***](#_heading=h.3znysh7) ***11***

[**Bảng phân công công việc**](#_heading=h.q0hbivqge4cm) **11**

[**Quá trình thực hiện**](#_heading=h.k1fmoc1odml2) **11**

[Tools preparation (24/10)](#_heading=h.ue2dv0u56a10) 11

[API research (24/10 - 30/10)](#_heading=h.fhq6exqbtgjr) 12

[Thiết kế flow (31/10 - 1/11)](#_heading=h.z3qu9ta96rmb) 13

[Code (2/11 - 10/11)](#_heading=h.5s56otfjph5q) 14

[Thu thập data (11/11)](#_heading=h.90a1562q4d7s) 14

[Báo cáo (11/11-13/11)](#_heading=h.q08evgifkv6z) 16

[**Đánh giá dữ liệu**](#_heading=h.pd10567qy7pu) **16**

[User](#_heading=h.8dc6y1665788) 17

[Track](#_heading=h.nqwghxjbg2qp) 17

[Playlist](#_heading=h.a7luw8sppsec) 18

# *Phần 1: Parse HTML*

## Bảng phân công chi tiết

| **STT** | **Họ và Tên** | **Nhiệm vụ** | **% hoàn thành** |
| --- | --- | --- | --- |
| 1 | Mạc Văn Hưng | - Thiết lập Github, các file thư mục theo yêu cầu đề tài | 100 |
| - Parse users và tracks trong từng genre ngoài trang chủ discover | 100 |
| - Parse tất cả playlist có trong từng genre của trang discover | 100 |
| - Parse playlists, tracks, users hiển thị trên trang search (sử dụng search bar) | 100 |
| - Viết các hàm chức năng cần thiết (lưu trữ dữ liệu vào file tsv, cuộn trang,...) | 100 |
| 2 | Lữ Thế Vỹ | - Parse tracks, users, playlists bằng cách truy cập vào profile từng user trong từng genre | 100 |
| - Tạo vòng lặp parse tất cả dữ liệu trong một genre | 100 |
| - Tạo vòng lặp parse tất cả dữ liệu trong một profile user | 100 |
| - Viết các hàm chức năng cần thiết (lưu trữ dữ liệu vào file tsv, cuộn trang,...) | 100 |
| - Viết báo cáo (phần parse HTML) | 100 |

## Quy trình thực hiện

### 2.1 Chuẩn bị kiến thức

* Tuần đầu tiên sẽ là tuần bổ sung kiến thức về Selenium Webdriver.
* Đọc document ở các nguồn:
* [WebDriver | Selenium](https://www.selenium.dev/documentation/webdriver/)
* [Selenium with Python — Selenium Python Bindings 2 documentation (selenium-python.readthedocs.io)](https://selenium-python.readthedocs.io/index.html)
* Lên kế hoạch, xác định phương pháp tiếp cận khai thác dữ liệu, tìm hiểu tổng quan về giao diện trang web SoundCloud.

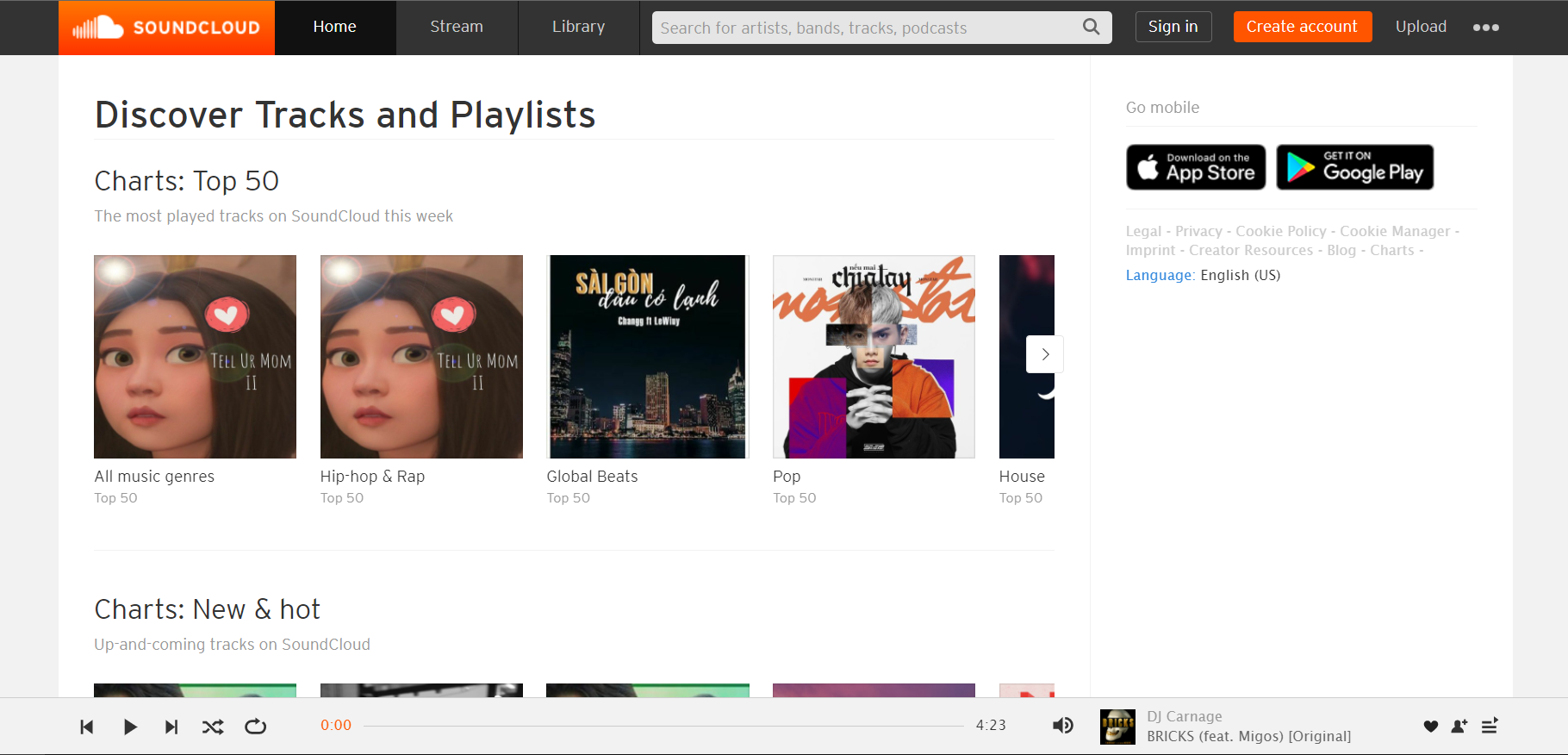
### 2.2 Xác định nơi khai thác dữ liệu

* Dữ liệu cần khai thác theo yêu cầu đồ án được thể hiện rõ ràng, thuận tiện cho việc khai thác chủ yếu nằm ở các địa chỉ sau:
* <https://soundcloud.com/discover>
* <https://soundcloud.com/search>

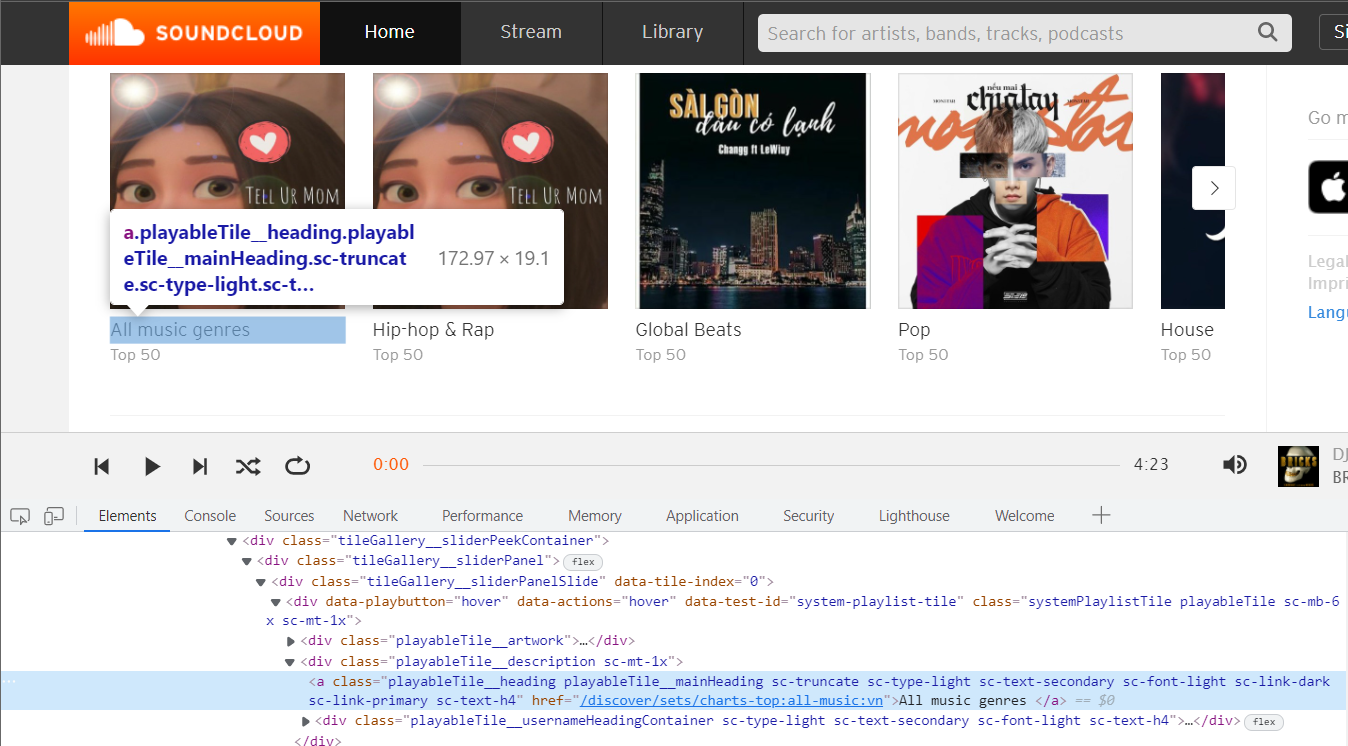
### 2.3 Giải thích quá trình thực hiện

#### 2.3.1 Cách 1: khai thác từ trang <https://soundcloud.com/discover>

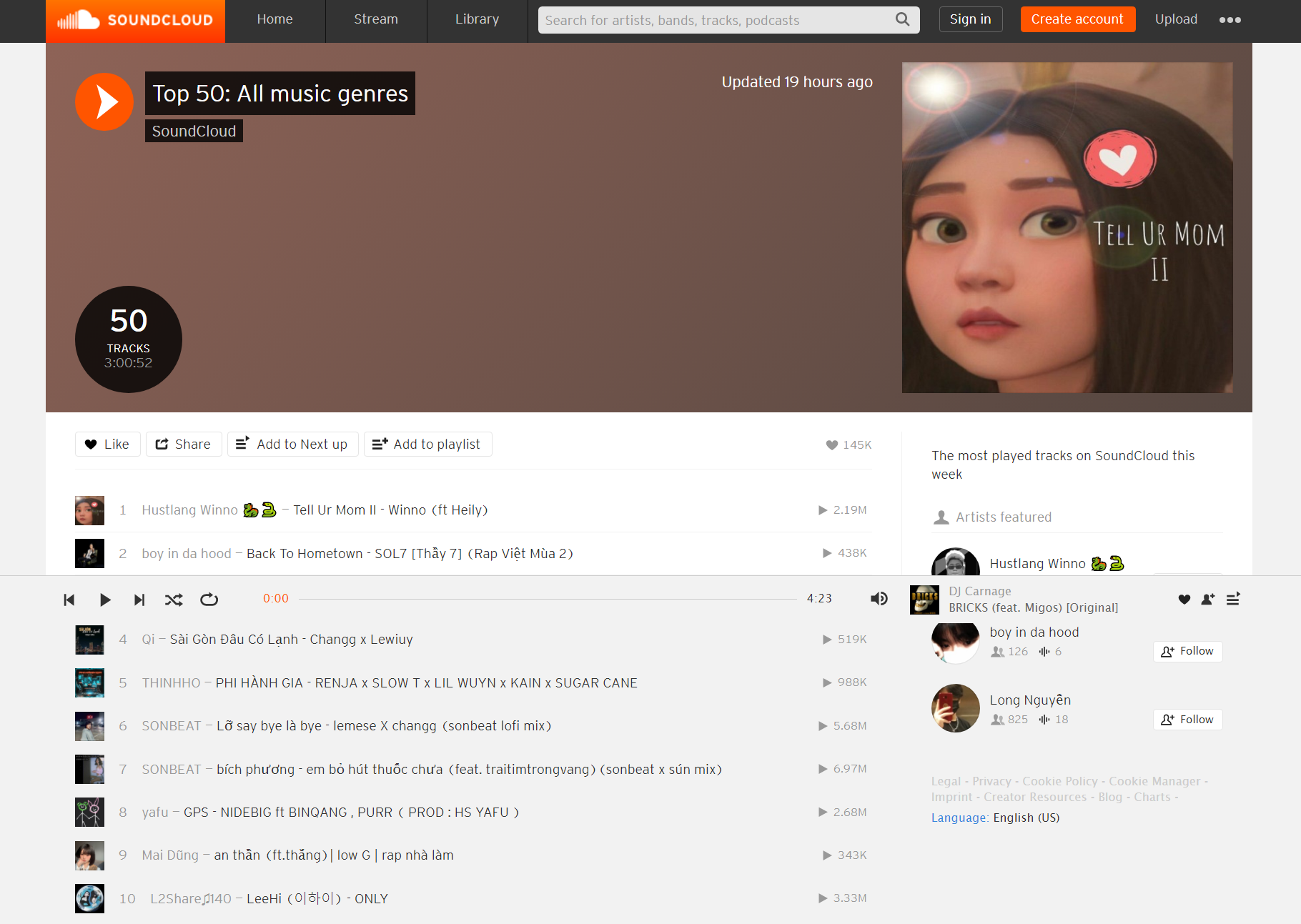
* *Giao diện trang web:*



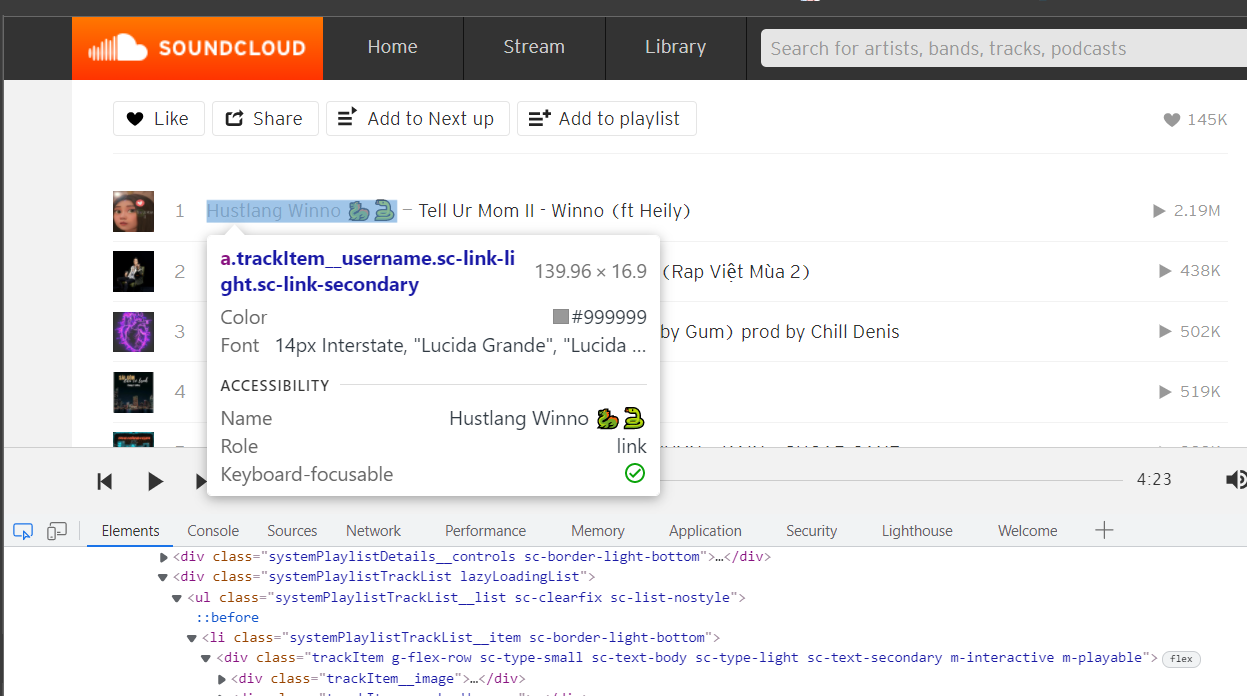
* *Soundcloud/Discover là nơi lưu trữ tác phẩm của các nghệ sĩ đã được phân loại thành các genre (thể loại nhạc) hoặc được gom lại ở dạng bảng xếp hạng (top 50, theo thể loại nhạc)*
* Chúng ta sử dụng tính năng Inspect của Web Browser để phân tích thành phần, cấu trúc của trang web. ***Nhờ nắm được ID của các thành phần , chúng ta có thể sử dụng kĩ thuật Selenium để định vị các thành phần trên trang web và tương tác với chúng (giả sử: click chọn).***



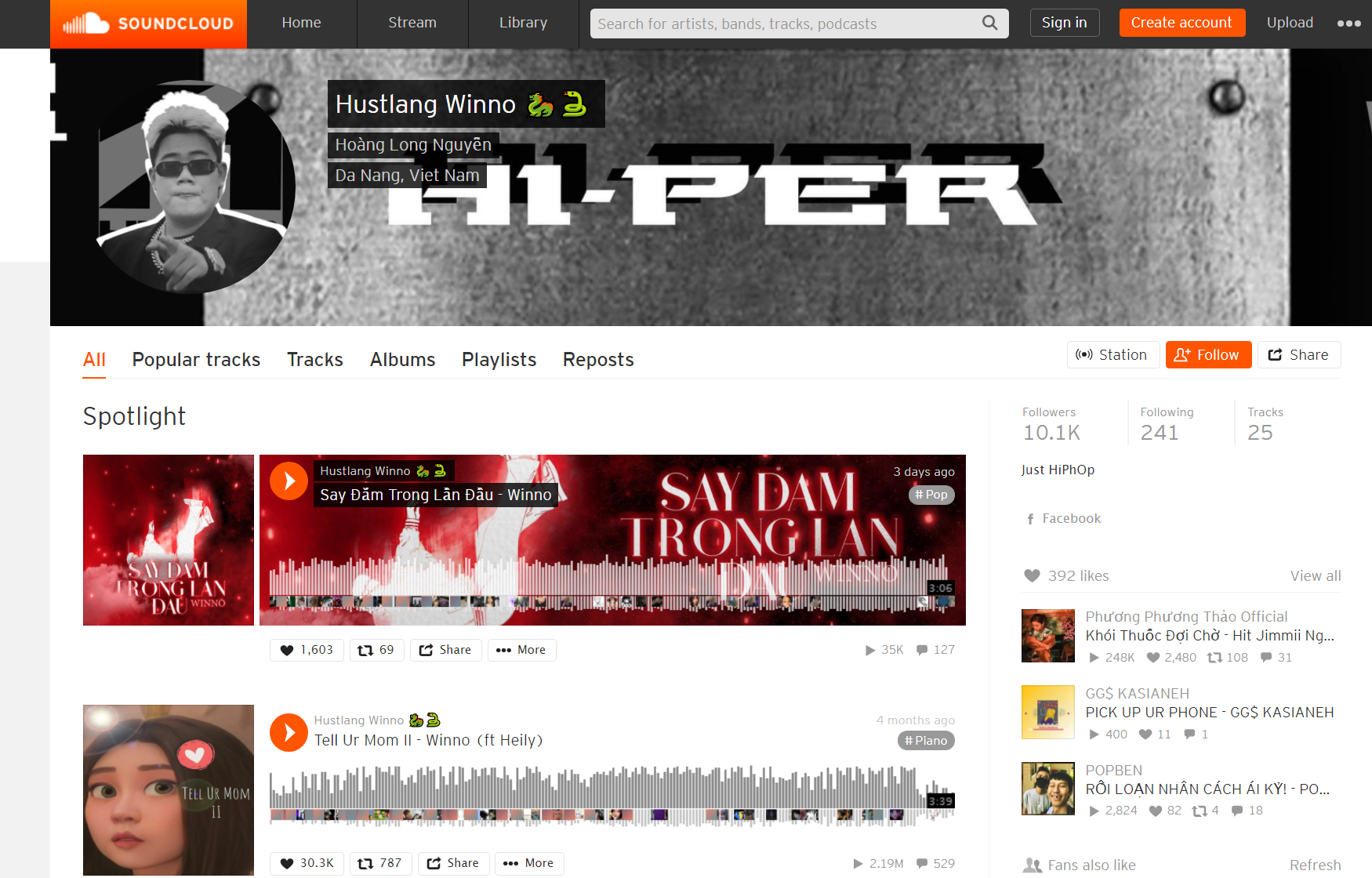
* Mỗi genre (thể loại nhạc) sẽ là một tập hợp tác phẩm của các nghệ sĩ, liệt kê dưới dạng danh sách (như hình bên dưới). Dựa vào các kĩ thuật Selenium, chúng ta có thể lấy được các dữ liệu liên quan thông qua các ID của thành phần trang web

**

* *Sử dụng tính năng Inspect để lấy ID của các thành phần, trong ảnh là ID của thành phần “Tên nghệ sĩ”.*

**

* *Profile của các user chứa đầy đủ các tệp dữ liệu có thể đáp ứng các yêu cầu khai thác của đồ án (tracks, user, playlists). Ví dụ như hình:*



##### Nhược điểm:

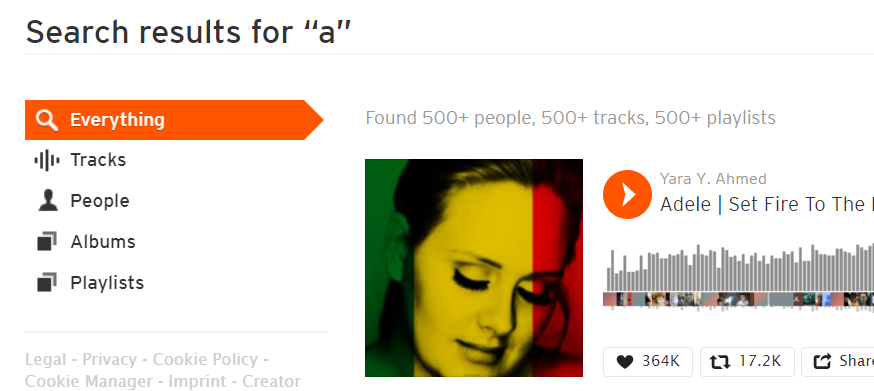
* Phải thao tác với các thành phần nhiều lần (click chọn, phải di chuyển nhiều lần giữa các screen,..) do có nhiều danh sách genre.
* Mỗi genre chỉ lưu trữ tối đa 50 tác phẩm, khi kết thúc lượt phải trỏ về trang trước và chọn genre khác.
* Việc thay đổi screen liên tục sẽ làm giảm hiệu quả, tốc độ khai thác một cách rõ rệt đối với tệp dữ liệu lớn như trên trang Soundcloud.
* Soundcloud là một trang web sử dụng kĩ thuật **Lazy Loading** (tức là việc hiển thị dữ liệu chỉ diễn ra ở phần chúng ta đang cuộn đến). Di chuyển giữa các screen, cuộn trang web nhiều lần sẽ làm xảy ra hiện tượng lag (tăng thời gian loading screen).
* Hiện tượng tăng thời gian loading screen sẽ làm cho một số thành phần của trang web không hiển thị đầy đủ. Dẫn đến việc Selenium báo lỗi không thể định vị được thành phần.

#### 2.3.2 Cách 2: khai thác từ trang <https://soundcloud.com/search> (cải tiến)

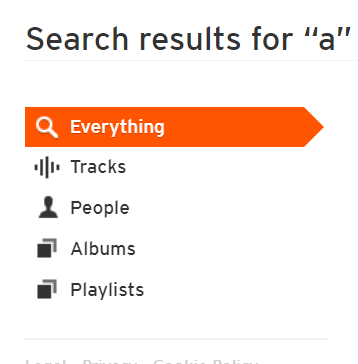
* *Giao diện trang web:*

**

* Sử dụng thanh Search của SoundCloud giúp ta hạn chế được rất nhiều các thao tác với trang web (không phải click chọn các thành phần nhiều lần)
* **Ưu điểm:** Chỉ cần nhập một ký tự chữ cái, chúng ta đã có ngay hơn 500+ tác phẩm liệt kê theo dạng danh sách.



* Chúng ta chỉ cần sử dụng thanh cuộn là có thể lấy thêm các tác phẩm khác. Việc các dữ liệu được thể hiện trực quan như thế này sẽ giúp tiết kiệm công sức, nâng cao hiệu suất khai thác dữ liệu rất nhiều lần
* **Ngoài ra, các tab cần thiết cho việc khai thác dữ liệu cũng được trình bày bên ngoài trang chủ, giúp hạn chế tối đa số lần click - chuyển trang**



## 

## 

## 

## 

## Đánh giá dữ liệu

* **Từ trang Discover**: đơn vị bản lưu (records)  
  **\* Tổng thời gian chạy: 6 giờ**
* Tracks (gồm trường: tên track, tên user sở hữu): *11709*
* Users: *6508*
* Playlists (gồm trường: tên playlist, tên user và tên track): *1227*
* **Từ trang Search (dùng các chữ cái alphabet)**:  
  **\* Tổng thời gian chạy: 1.5 giờ**
* Tracks (gồm trường: tên track, tên user sở hữu): *16830*
* Users: *12538*
* Playlists (gồm trường: tên playlist, tên user và tên track): *4800*

##### Nhận xét:

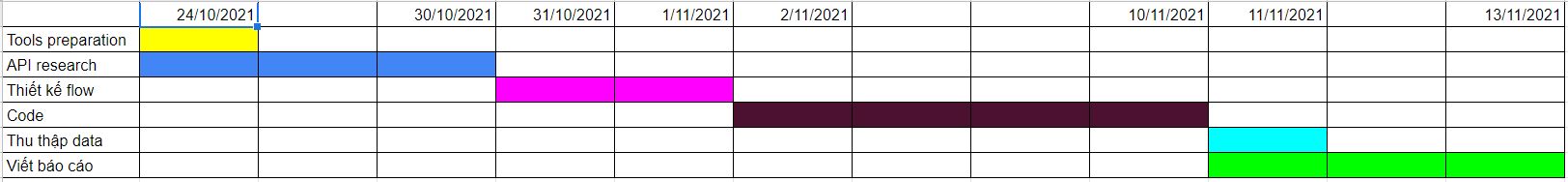
* Phương pháp tiếp cận từ *trang Search tỏ ra hiệu quả gấp bội*. Bằng chứng là chỉ trong thời gian rất ngắn hơn nhiều lần, nhưng đã khai thác được số lượng record nhiều hơn rõ rệt.
* Việc thay đổi hướng tiếp cận khai thác, từ đó làm giảm thời gian thao tác trên web đã cho ra kết quả tốt hơn so với thử nghiệm ban đầu.
* Đối với phương pháp 1 (tiếp cận từ trang Discover), thu thập theo kiểu Representative Sampling, dữ liệu dễ bị bias bởi mỗi genre có số lượng tác phẩm khác nhau.
* Ở phương pháp 2, thu thập theo kiểu Random Samping, dữ liệu không bị bias

# *Phần 2: API*

# Bảng phân công công việc

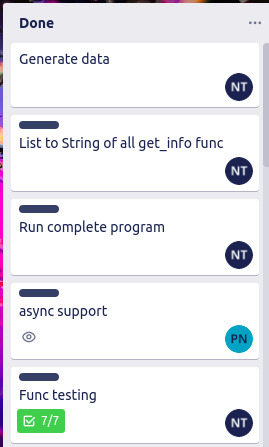
| Công việc | Người làm |
| --- | --- |
| API research | Phước |
| Thiết kế flow | Phước, Trang |
| Code | Phước (50%), Trang (50%) |
| Báo cáo | Phước, Trang |
| Viết jupyter notebook | Phước, Trang |

# Quá trình thực hiện



## Tools preparation (24/10)

* Nhóm thành lập và bắt đầu họp bàn chia thành 2 nhóm nhỏ, mỗi nhóm 2 người làm về mảng khác nhau là API và HTML
* Đây cũng là giai đoạn thiết lập trello, github, google drive để nhóm sử dụng trong suốt quá trình thực hiện



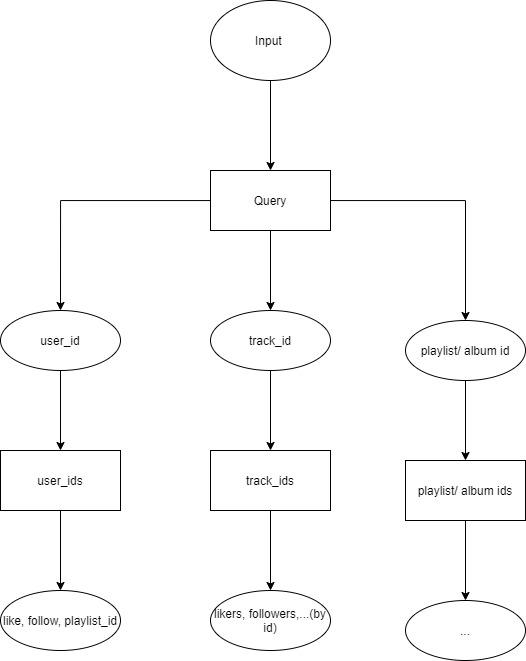
* Nhóm API sử dụng trello để quản lý các tác vụ từ việc viết hàm, viết báo cáo
* Nhóm cũng sử dụng google drive và github để phối hợp với nhau trong quá trình làm việc

## API research (24/10 - 30/10)

* Khi tìm hiểu về API của SoundCloud, nhóm nhận thấy SoundCloud có 2 loại API khác nhau:
  + Public API: hay API-V1 là api có public document đầy đủ trên mạng, tuy nhiên api này lại thiếu đi những chức năng chính mà API SoundCloud sử dụng trên website của họ, liệt kê một số chức năng thiếu như: những track liên quan, chart,...
  + API-V2: đây là API nội bộ, không có public document và được SoundCloud sử dụng trên website. Nhóm tìm hiểu trên các diễn đàn (stack-overflow, quora) thì thấy nhiều khuyến nghị không nên sử dụng để phát triển ứng dụng cá nhân, tuy nhiên api này lại bao gồm rất nhiều tính năng mà API-V1 còn thiếu
* Qua cân nhắc, vì đây là project nhỏ phục vụ học tập, cũng vì việc sử dụng API-V2 chứng tỏ có nhiều lợi hơn là hại, nhóm quyết định sử dụng API-V2 này vào quá trình làm project.
* Vấn đề thiếu document, nhóm giải quyết bằng việc **tự viết document** cho API-V2, thông qua việc theo dõi tab network dưới sự hỗ trợ của trình duyệt, nhóm thành công trích ra những request-url có ích, có thể sử dụng cho project và đã cố gắng tận dụng triệt để trong quá trình thực hiện.
* Kết thúc giai đoạn này, nhóm đưa ra được 1 document cho API-V2, tuy không phải document chính thức, nhưng đã là nền tảng cho việc thu thập dữ liệu bằng API. File này đã được đính kèm vào các tập tin nộp đồ án.

## Thiết kế flow (31/10 - 1/11)

* Sau khi cân nhắc về kích cỡ đồ án, nhóm nhận thấy việc đưa ra 1 thiết kế tổng quát cho chương trình là cần thiết. Nhóm đã tiến hành thảo luận, thiết kế, chỉnh sửa các bước chính của việc thu thập dữ liệu qua API.
* Giai đoạn này nhóm đã đưa ra được hướng đi cho việc thu thập dữ liệu, gồm:
* Thu thập dữ liệu một cách tổng quát, đó là thu thập dữ liệu không phụ thuộc vào input của người dùng, thu thập qua 3 trang:
  + <https://soundcloud.com/>
  + <https://soundcloud.com/discover>
  + <https://soundcloud.com/charts>
* Thu thập dữ liệu dựa vào keyword, input vào 1 hoặc nhiều keyword và output ra các thông tin liên quan thuộc 3 nhóm: user, playlist, track



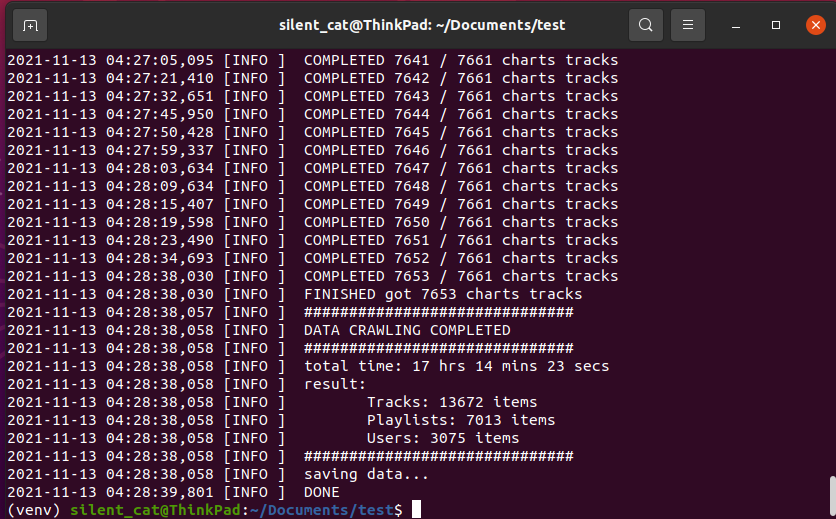
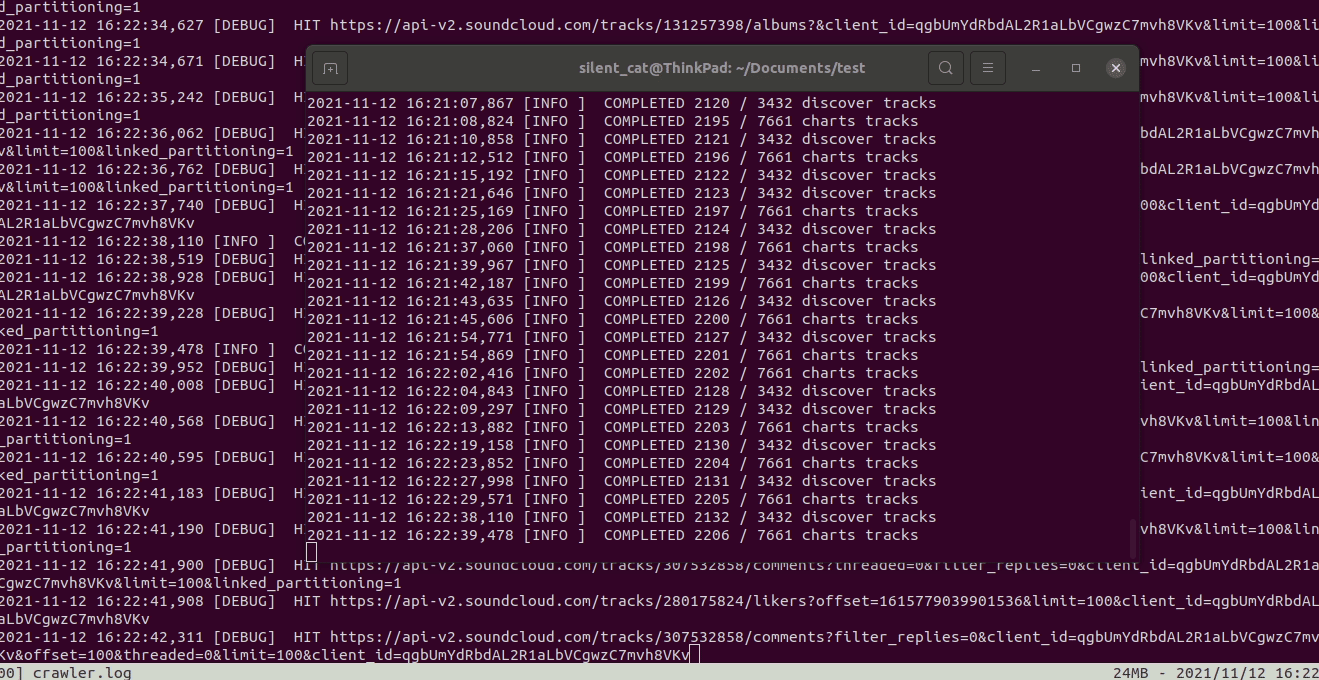
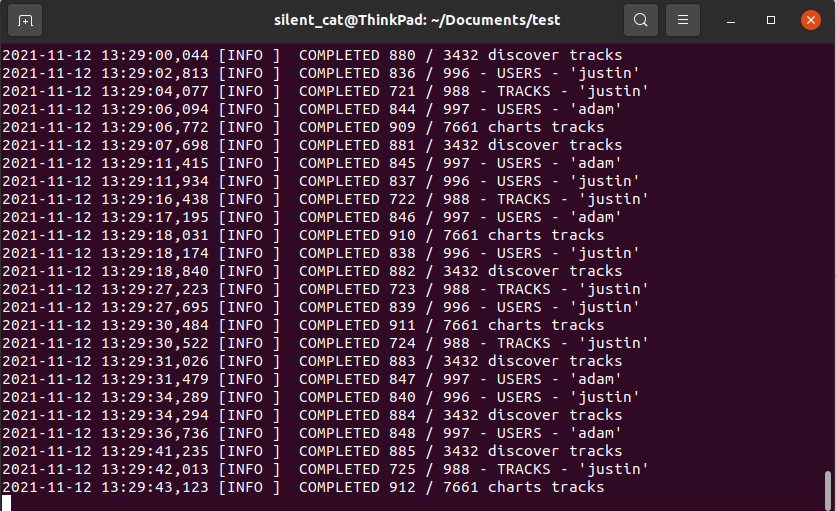
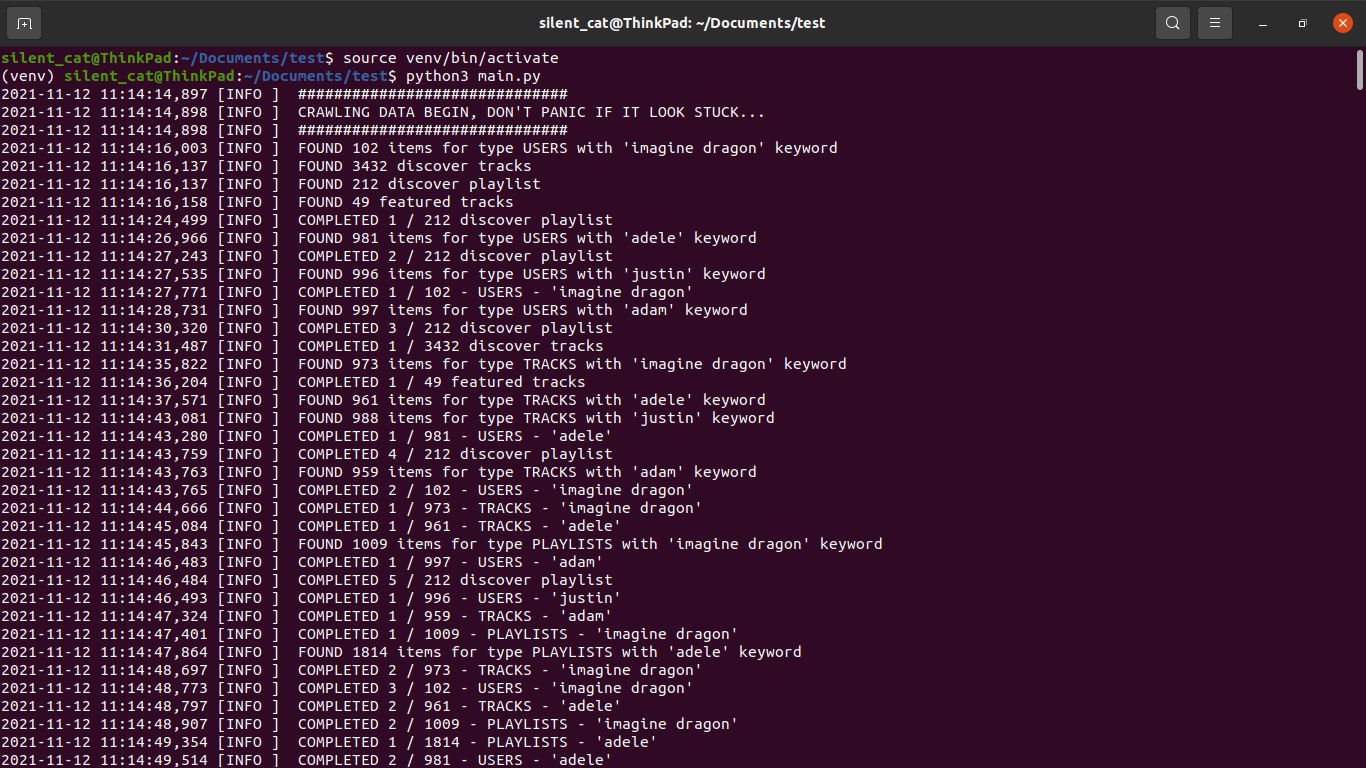
* Các quá trình trên được đưa ra sau khi có sự cân nhắc về các API hiện có, khả năng thực hiện của nhóm và một số nguyên nhân khác.
* Flow đã đưa ra có thể hiệu chỉnh trong quá trình làm việc, nhưng về cơ bản, đây sẽ là hướng đi chính mà nhóm tiếp cận vấn đề.

## Code (2/11 - 10/11)

* Bắt đầu bằng việc tạo các task trong trello và lần lượt thực hiện, nhóm dựa theo flow có sẵn để tiến hành.
* Nhóm đã code xong những hàm chính trước khoảng thời gian dự liệu khá sớm (còn dư khoảng 3-4 ngày), như hàm lấy dữ liệu từ 3 trang chính, hàm lấy dữ liệu khi input keywords và đưa ra được bản code hoàn chỉnh đầu tiên.
* Tuy nhiên, sớm nhận thấy đoạn code này có thể được tối ưu về tốc độ thông qua lợi dụng khoảng thời gian chờ response của server để làm việc khác, nhóm sử dụng kĩ thuật lập trình đa tiến trình thông qua 2 thư viện là : asyncio, aiohttp
* Chương trình của nhóm sẽ chạy và lấy dữ liệu từ các trang có trước và keyword đã input, 3 loại thông tin chính sẽ được lưu về và save vào 3 file khác nhau tương ứng với user, track, playlist
* Việc tìm hiểu, ứng dụng kĩ thuật trên khá thành công, sau giai đoạn này, nhóm đã có file source hoàn chỉnh, cùng với những tài liệu liên quan, sẵn sàng để viết document về các hàm
* Kết thúc giai đoạn coding và bắt đầu giai đoạn chạy code lấy dữ liệu và viết báo cáo

## Thu thập data (11/11)

* Quá trình code và debug code diễn ra thuận lợi, kết thúc quá trình, nhóm đã đưa ra được 3 file như yêu cầu là: playlist.csv, user.csv và track.csv
* Quá trình thu thập dữ liệu gồm các bước cơ bản: tìm kiếm id tương ứng cho mỗi loại, và dựa trên mỗi id tìm được, tiến hành request thông tin
* Đối với mỗi keyword, nhóm tiến hành lấy 1000 kết quả đầu tiên
* Đối với các dữ liệu phụ như số lượt like, số lượt follow, nhóm lấy 100 kết quả đầu tiên



* Chi tiết kết quả sẽ được báo cáo ở phần đánh giá dữ liệu
* Bên cạnh dữ liệu, file log đi kèm sẽ lưu trữ thông tin chi tiết về quá trình thu thập

## Báo cáo (11/11-13/11)

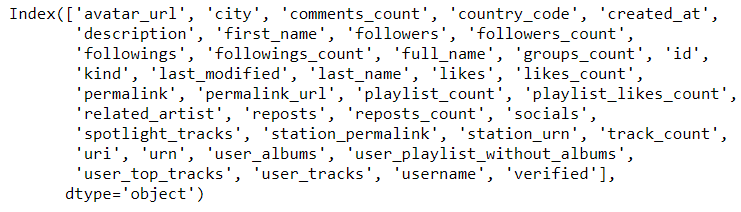
* Nhóm bắt đầu tổng hợp, viết jupyter notebook cho từng hàm, báo cáo cũng bắt đầu được hoàn thành

# Đánh giá dữ liệu

* Thời gian thu thập: 17 giờ 14 phút 23 giây
* Phương pháp thu thập chung: thực hiện gửi yêu cầu cho SoundCloud API server và trả về kết quả dưới dạng json.
* Có 2 hướng tiếp cận chính: dựa vào các URL đã được định nghĩa từ trước trong API Document để thực hiện quá trình query (ở đây, nhóm đã truyền vào 1 *keyword\_list* như sau ) hoặc ta có thể lấy thông tin mà không cần truyền keyword nào từ các trang General .
* Khuyết điểm: dễ bị bias do dữ liệu thu thập được còn phụ thuộc nhiều vào keyword.
* Đối với dữ liệu phụ thuộc vào số lượng cũng có thể dẫn đến bias:
  + Việc lấy dữ liệu theo 1000 kết quả đầu tiên ở mỗi loại, mỗi keyword
  + VIệc lấy 100 kết quả đầu tiên cho các data phụ như liker, follower

## User

* Số lượng : 3 075
* Các trường đã thu thập: 35 trường



## Track

* Số lượng: 13 672
* Các trường đã thu thập: 51 trường

## 

## Playlist

* Số lượng: 7 013
* Các trường đã thu thập: 51 trường

