SPOTIFY E STAZIONI RADIO

Data Management report

Baghrous Tariq 904027
Balbin Canchanya Gianni Eduard 901609
Galante Fabio 851242

Master's Degree in Data Science
Università degli Studi di Milano-Bicocca
2022/2023 Academic Year

Indice

1 Suddivisione progetto: ruoli

Ogni membro del progetto ha svolto almeno un compito chiave per la completezza del lavoro, arrivando ad ottenere il database finale. Questi i ruoli primari:

- Gianni Eduard Balbin Canchanya: data integration e data quality
- Tariq Baghrous: data acquisition, preparation e exploration
- Fabio galante: data integration, data quality e data storage

2 Introduzione

2.1 Idea del Progetto

Spotify è un popolare servizio di streaming musicale che ha trasformato l'industria musicale. Lanciato nel 2008, offre una vasta libreria di musica accessibile attraverso un modello freemium. Grazie a consigli personalizzati e all'integrazione sociale, Spotify offre un'esperienza user-friendly plasmando il modo in cui scopriamo e godiamo della musica nell'era digitale.

La popolare piattaforma di streaming musicale ha una forte influenza sulle classifiche musicali ARIA in Australia e Billboard negli Stati Uniti. Questo è dovuto alla sua vasta base di utenti e ai dati raccolti attraverso l'ascolto musicale, che influenzano la posizione delle canzoni nelle classifiche. Questa correlazione ha migliorato la precisione delle classifiche e riflette il cambiamento nel consumo musicale verso lo streaming.

Allo stesso modo le stazioni radio svolgono un ruolo cruciale nell'industria musicale promuovendo e diffondendo nuove uscite musicali. Curano playlist, collaborano con le etichette discografiche e offrono visibilità agli artisti, aiutandoli a ottenere riconoscimento e a raggiungere un pubblico più ampio. La trasmissione radiofonica ha un impatto significativo sulle vendite di musica e sulle posizioni nelle classifiche. Nonostante l'ascesa delle piattaforme digitali abbia cambiato il panorama musicale, le stazioni radio rimangono influenti nel plasmare i gusti popolari e stimolare conversazioni sulla musica, continuando a svolgere un ruolo fondamentale nella scoperta e nell'apprezzamento della musica.

Nel mondo moderno la musica è quindi diventata una parte essenziale della vita di molte persone e la disponibilità delle canzoni preferite in stazioni radio locali è un aspetto cruciale. Tuttavia, trovare informazioni precise sulle stazioni radio che trasmettono le canzoni preferite può risultare complesso.

In questo panorama, il progetto che presentiamo ha l'obiettivo di creare un database innovativo che consente agli utenti di identificare delle stazioni radio situate in un determinato paese tramite il genere delle proprie canzoni preferite e/o più popolari dalla piattaforma Spotify, offrendo di conseguenza agli utenti una soluzione efficace per scoprire le stazioni radio in cui è possibile ascoltare la musica che amano.

Questo progetto sfrutta dati provenienti da due fonti principali: i dati sulle Top 10.000 canzoni su Spotify secondo le classifiche ARIA e Billboard e i dati su più 66.000 stazioni radio in tutto il mondo.

3 Data Acquisition

3.1 Top 10000 Songs on Spotify 1960-Now

Questo dataset è stato scaricato da Kaggle [1] e consiste in una collezione completa di 10.000 delle canzoni più popolari che hanno dominato la scena musicale dal 1960 ai giorni nostri.

Questo dataset è stato curato basandosi sulle classifiche di entrambi ARIA (Australian Recording Industry Association) e Billboard, garantendo una rappresentazione diversificata di canzoni che hanno ottenuto un immenso successo commerciale e una notevole importanza culturale.

Dopo aver importato questo dataset abbiamo deciso di selezionare solamente alcune delle variabili disponibili, poichè considerate più utili e attinenti ai nostri scopi.

Gli attributi selezionati sono i seguenti:

- **Track URI:** Codice identificativo della traccia all'interno del database Spotify.
- **Track Name:** Il nome della traccia.
- **Artist URI(s):** Codice identificativo dell'artista all'interno del database Spotify.
- **Artist Name(s):** Il nome dell'artista.
- **Album URI:** Codice identificativo dell'album all'interno del database Spotify.
- **Album Name:** Il nome dell'album.
- **Popularity:** La popolarità del brano. Il valore sarà compreso tra 0 e 100, con 100 come valore più popolare.
- **Artist Genres:** Un elenco dei generi a cui l'artista è associato. Se non ancora classificato, l'array è vuoto.

3.2 Spotify Web API

La Spotify Web API [2] è un insieme di strumenti e servizi forniti da Spotify che possono essere utilizzati per integrare le funzionalità di Spotify in altre piattaforme. Tramite un account developer siamo riusciti ad aquisire diversi attributi utili per arricchire il database iniziale:

- **artist genres:** Un elenco dei generi a cui l'artista è associato. Se non ancora classificato, l'array è vuoto.
- **available markets:** Un elenco dei Paesi in cui è possibile riprodurre il brano, identificati dal loro codice ISO 3166-1 alpha-2.
- **danceability:** Descrive quanto un brano sia adatto al ballo in base a una combinazione di elementi musicali, tra cui il tempo, la stabilità del ritmo, la forza del battito e la regolarità generale. Un valore di 0,0 è il meno ballabile e di 1,0 è il più ballabile.
- **energy:** Misura che va da 0,0 a 1,0 e rappresenta una misura percettiva dell'intensità e dell'attività.
- **loudness:** Quantità totale di db generata dalla track

- **speechiness:** Presenza di parole parlate in una traccia. Più la registrazione è esclusivamente di tipo parlato, più il valore dell'attributo si avvicina a 1,0.

- **acousticness:** Una misura di fiducia da 0,0 a 1,0 che indica se il brano è acustico. 1,0 rappresenta un'elevata fiducia che il brano sia acustico.

- **instrumentalness:** Più il valore di strumentalità è vicino a 1,0, maggiore è la probabilità che il brano non contenga contenuti vocali.

- **liveness:** Rileva la presenza di un pubblico nella registrazione. Valori di liveness più elevati rappresentano una maggiore probabilità che il brano sia stato eseguito dal vivo.

- **valence:** Una misura da 0,0 a 1,0 che descrive la positività musicale trasmessa da un brano.

- **tempo:** Il tempo complessivo stimato di un brano in battiti al minuto (BPM).

- **track_href:** Un link all'endpoint dell'API Web che fornisce tutti i dettagli della traccia.

- **duration_ms:** La durata della traccia in millisecondi.

3.3 Radio stations API

L'API Radio World [3] è un API Rest che consente di accedere a una vasta collezione di stazioni radio di vari paesi e generi. Con circa 64.000 stazioni radio disponibili, gli utenti possono esplorare una vasta gamma di scelte musicali che più corrispondono alle loro preferenze. gli attributi estratti da questa API sono i seguenti:

- **radio_name:** Nome della stazione radio.

- **radio_url:** Link url della stazione radio

- **genre:** Genere della radio.

- **country_name:** Paese di provenienza.

4 Data Preparation

L'intero processo è stato effettuato sulla piattaforma Jupyter Notebook utilizzando quindi python come linguaggio di programmazione.

Il processo inizia con l'importazione del primo dataset "Top 10000 Songs on Spotify 1960-Now" con il nome di "top10k" che contiene 9951 righe e 35 colonne, delle quali selezioniamo 8 di nostro interesse usando il comando *usecols*.

Attraverso L'API Radio World, creiamo il dataset "radio_metadata" costituito da 64230 righe e 4 colonne, considerate le più pertinenti, ed esportiamo un csv in modo tale da avere il file nella directory per successivi utilizzi.

4.1 Data cleansing

Il passo successivo consiste nel pulire i dataset. A tal proposito abbiamo controllato la presenza di dati nulli e abbiamo ottenuto il seguente output:

```
Track URI           0
Track Name          1
Artist URI(s)       2
Artist Name(s)      1
Album URI           2
Album Name          1
Popularity          0
Artist Genres     550
dtype: int64
```

Figura 1: missing values on "Top 10000 Songs on Spotify 1960-Now"

Siamo quindi andati ad indagare sulla natura dei valori mancanti e se possibile usare la web API di spotify per sostituirli. Per quanto riguarda la traccia con il nome mancante siamo risaliti, tramite la Track URI, all'url della traccia su spotify e abbiamo notato che non esiste, motivo per cui l'abbiamo conseguentemente eliminiata.

Estraendo le altre due tracce con Artist URI(s) e Album URI mancanti invece abbiamo notato che il formato delle Track URI inizia con "spotify:local:" il che significa che sono file locali, ovvero brani musicali che non sono disponibili direttamente nella libreria di Spotify, ma che sono stati aggiunti dalla libreria personale all'interno dell'applicazione.

Dopo avere rimosso queste ultime tracce, ci siamo occupati di individuare ed eliminare i duplicati tramite il comando *.drop_duplicates()* andando a riscontrare circa 48 righe duplicate.

l'ultimo ostacolo è dovuto dal numero esiguo di canzoni con l'Artist Genres mancante. Per questo problema abbiamo provato a servirci della web API di spotify per trovare, inserendo come input le Track URI, i generi mancanti. Come possibile vedere dall'immagine sottostante, abbiamo trovato solamente tre corrispondenze e le restanti righe verranno in seguito scartate.

	Artist URIs2	Artist Genres2
0	spotify:artist:2No2vspr2KORUEcZKH4xUi	classic australian country
1	spotify:artist:0lmRxVVpx9hSmeQv9jGYYR	classic australian country
2	spotify:artist:1b5taabb9eKSbyzVFVtEjh	deep adult standards

Figura 2: Artist Genres

4.2 Data Enrichment

In questa fase del processo abbiamo arricchito il dataset iniziale con informazioni aggiuntive che ci permetteranno di effettuare sia l'integrazione con il *radio_metadata* che di estrarre tutte le caratteristiche musicali per ogni traccia.

In particolare abbiamo aggiunto una colonna denominata "Country" nel quale per ogni traccia viene riportato una lista dei Paesi in cui è possibile riprodurre il brano, identificati dal loro codice ISO 3166-1 alpha-2. Ciò è stato possibile tramite la Web API di spotify *"sp.tracks"* inserendo come input le varie track URI e ottenendo come output gli *available_markets*.

Ora che abbiamo tutti gli attributi desiderati per top10k, andiamo a identificare le righe con valori nulli/vuoti di Artist Genre e Country e le rimuoviamo dal dataframe con il comando *dropna()* (COMPLETENESS)

```
Track URI            0
Track Name           0
Artist URI(s)        0
Artist Name(s)       0
Album URI            0
Album Name           0
Popularity           0
Artist Genres      539
Country           2902
dtype: int64
```

Figura 3: missing artist genres and countries

Impiegando la Web API di spotify *"sp.audio_features"* e le track URI del dataset abbiamo creato un ulteriore dataframe denominato *top10k_features* che riporta tutte le features interessanti per ogni canzone (precedentemente descritte nella sezione 2.2).

Per quanto riguarda il dataset delle radio abbiamo riscontrato un problema di CONSISTENCY con il dataset top10k, in quanto i paesi sono riportati con il loro nome completo mentre nel dataset delle tracce musicali sono riportati con il loro codice ISO 3166-1 alpha-2, rendendo impossibile la successiva integrazione. Abbiamo quindi deciso di aggiungere una colonna nel dataframe *radio_metadata* denominato "country_code" il quale contiene il codice ISO 3166-1 alpha-2 in base al country_name. Per eseguire questa mappatura utilizziamo una libreria in python chiamata *"pycountry"*. La corrispondenza non è stata del 100% per via del formato insolito con cui sono stati scritti alcuni paesi (CONSISTENCY) dunque per i paesi che non hanno fornito un codice lo abbiamo dovuto aggiungere manualmente.

Abbiamo così preparato tre dataset denominati **top10k_spotify.csv**, **radio_metadata.csv** e **top10k_features.csv** pronti per la successiva fase del progetto.

4.3 Data Integration

Utilizzando i 2 dataset ottenuti *top10k_spotify* e *radio_metadata* abbiamo cercato un modo di integrarli, con l'obiettivo finale di avere un dataframe contente tutte le canzoni con associate le rispettive radio su cui possono essere riprodotte.

A tal scopo abbiamo creato dei dataset intermedi per facilitare il lavoro data l'elevata mole di dati con cui dobbiamo lavorare.

Per prima cosa abbiamo creato a partire dal dataset *top10k_spotify*:

- il file *generi2* contenente le informazioni ritenute da noi importanti di tutte le canzoni, di conseguenza si è fatto a meno degli indirizzi URI di spotify, e si è divisa la colonna *ArtistGenres* in più colonne, pari al numero di generi musicali possibili del dataset *top10k_spotify* ossia 32. È da notare che vi è la possibilità che alcune colonne per molte canzoni possano essere incomplete in quanto non è detto che ogni canzoni abbia esattamente 32 generi musicali.

Dal dataset *radio_metadata*, abbiamo notato che anch'esso in genere poteva avere più categorie, fino ad un massimo 3. Di conseguenza si è creato:

- il file *radio1genere* il quale ha le stesse informazioni del dataset iniziale ma si è raggruppato per quelle radio che hanno un genere solo;
- il file *radio2genere* dove in questo caso vi sono solo quelli con due generi; e,
- il file *radio3genere* che contiene solo le radio che includono solo tre generi musicali.

Dato che tutti i generi musicali delle canzoni sono in minuscolo, abbiamo modificato la grandezza dei caratteri in minuscolo anche per il genere delle radio in *radio1genere*.

Abbiamo proceduto allora trovando il match tra canzoni e radio sia per genere che per paese e inserendolo nella *listaok*.

Una volta ottenuto *listaok* che presenta per ogni canzone un vettore contenente i numeri associati alle radio che possono riprodurla, vogliamo trasformarle questa lista in un dataframe, in modo che possa essere esportato e utilizzato all'occorrenza.

Notiamo che diverse canzoni presentano lo stesso vettore di radio, perciò identifichiamo tutti i vettori diversi tra loro, ottenendo un totale di 15 vettori, ciò vuol dire che che solo 15 diverse combinazioni di radio possono essere elaborate per poi essere assegnate alle canzoni corrispondenti.

Iniziamo quindi il nostro processo di conversione della lista in dataframe, ottenendo *match_finale3*.

Questo dataset ha dimensioni 3114x5979, 3114 è il numero di tracce, 5975 è il numero massimo di radio che possono riprodurre una certa canzone, a cui aggiungiamo in seguito altre 4 colonne per indicare il nome e l'uri dell'artista e della traccia. Questo dataset contiene all'interno di ogni cella che non appartenga alle prime quattro colonne l'url di una radio su cui può essere riprodotta la canzone corrispondente, tuttavia non tutte le tracce possono essere riprodotte su 5975 radio, perciò quelle con X radio presenteranno l'url nelle colonne da 5 a X+5, e conterranno degli 0 in quelle seguenti.

Poichè questo dataset è di grandi dimensioni, abbiamo avuto problemi nell'importarlo su SQL che non è in grado di leggere così tante colonne, perciò abbiamo deciso di spezzarlo in 4 dataset ognuno da 1500 colonne(eccetto l'ultimo che ne ha 1479, in questo modo abbiamo creato le seguenti tabelle:

match1_1,*match1_2*,*match1_3*,*match1_finale4*.

5 Data Storage

Una volta ottenuto il nostro dataset finale con radio e canzoni integrati, abbiamo splittato nei 4 dataset poco fa menzionati e li abbiamo caricati su SQL insieme a *generi1radio* e *generi2*, ottenendo così il nostor database.

Abbiamo deciso di eseguire delle brevi query per mostrare come possono essere esplorati i nostri dati e come possono essere ottneute diverse informazioni.

Query 1

Con la query 1 vogliamo ottenere dati sulle stazioni radio a partire dal dataset *match1_1*, che contiene per ogni canzone tutte le radio su cui può essere riprodotta.

Per farlo uniamo questo dataset con *radio1genere*, trovando la corrispondenza tra l'url della radio contenuto in *radio1genere* e la quinta colonna di *match1_1*, che contiene l'url relativo alla prima radio identificata su cui la canzone della riga corrispondente può essere ascoltata.

Perciò come risultato si otterrà un dataset che presenta nella prima colonna il nome della canzone, nella seconda il nome della radio, nella terza il genere della radio e nella quarta il paese in cui si può ascoltare questa radio.

Questa tabella contiene meno di 3114 canzoni(numero di canzoni in *match1_1*) poichè elimina i duplicati, tenendo una sola osservazione per ogni singola canzone .

Query 2

In questa query invece partiamo da *radio1genere* e troviamo quali di queste radio vengono identificate per prime come radio che possono riprodurre le canzoni in *match1_1*), dopodichè vogliamo sapere qual è la prima canzone a cui sono stati associati ed infine aggiungere dettagli riguardo questa canzone.

Per farlo troviamo la corrispondenza tra url in *radio1genere* ee *match1_1*), dopo aver eseguito questa parte troviamo la corrispondenza tra il nome della canzone nel risultato ottenuto della query e il nome della canzone contenuto nel dataset *generi2*, dal quale otteniamo le altre informazioni sulle canzoni(album e artista).

6 Data Exploration

Importando le librerie *matplotlib.pyplot* e *seaborn* di Python abbiamo esplorato i tre dataset **top10k_spotify.csv**, **radio_metadata.csv** e **top10k_features.csv**.

nel primo abbiamo identificato e visualizzato i primi 10 artisti che hanno contribuito con il maggior numero di tracce musicali al dataset, come riportato nel grafico a barre orizzontali di seguito:

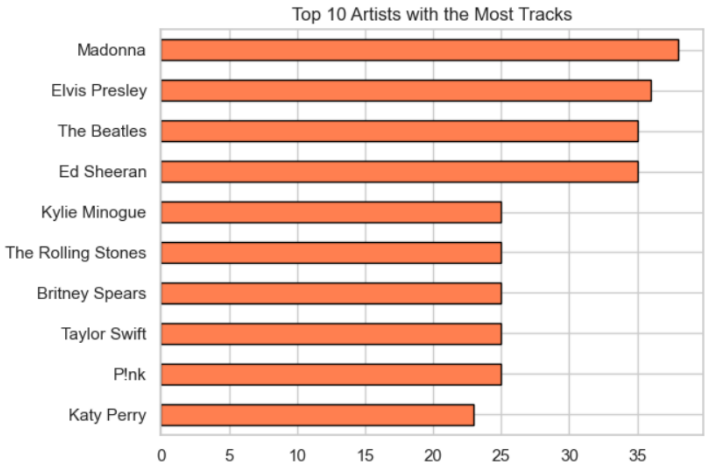

Figura 4: Top 10 Artists with the Most Tracks

Dall'analisi dei dati, emerge chiaramente che Madonna è l'artista con il maggior numero di tracce nel nostro dataset, seguito da Elvis Presley e The Beatles.

Nell'analisi dei dati relativi alle stazioni radio, abbiamo identificato le top 10 nazioni con il maggior numero di stazioni radio:

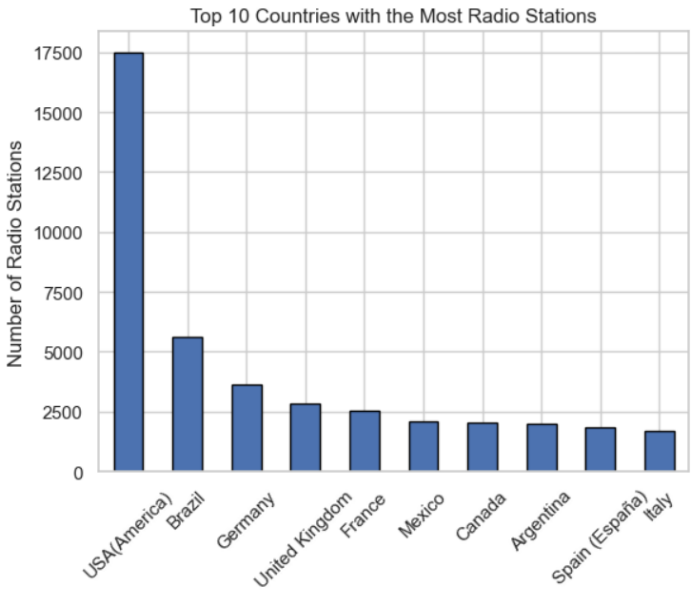

Figura 5: Top 10 Countries with the Most Radio Stations

Dall'osservazione del grafico, emerge che gli Stati Uniti d'America sono in testa alla lista, seguita dal Brasile e dalla Germania. Questi risultati evidenziano la predominante localizzazione delle stazioni radio in territorio statunitense, circa 17500 su 64230, e forniscono un punto di partenza per ulteriori analisi e ricerche nel settore radiofonico.

Nel dataset **top10k_features.csv**, abbiamo esaminato le correlazioni tra varie caratteristiche musicali e il livello di popolarità delle tracce. Questo è stato rappresentato tramite un grafico a heatmap della matrice di correlazione:

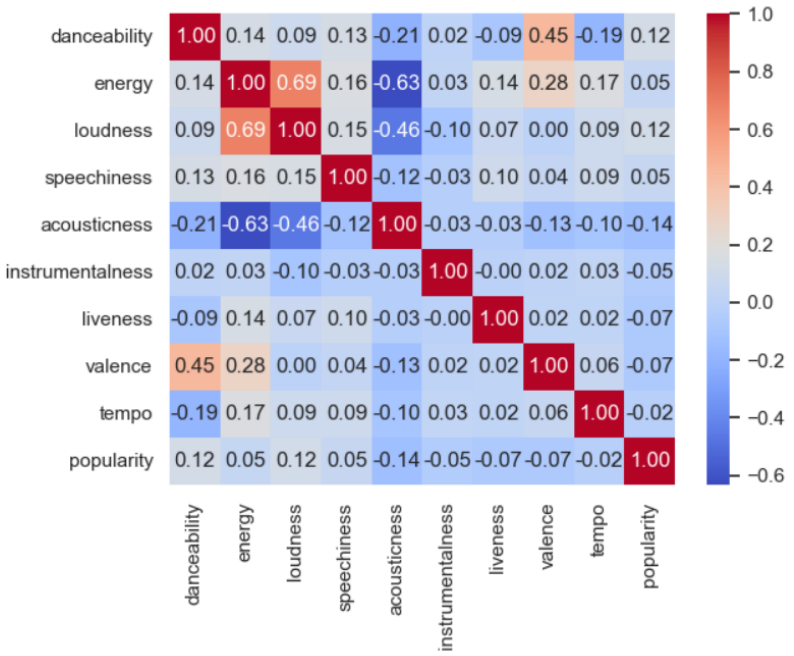

Figura 6: Top 10 Countries with the Most Radio Stations

Dall'osservazione del grafico, possiamo notare che gran parte delle correlazioni tra queste caratteristiche musicali e la popolarità delle tracce si aggirano attorno allo 0. Questo indica che, in generale, non esiste una correlazione forte tra queste caratteristiche e la popolarità. Tuttavia, è interessante notare che alcune coppie di caratteristiche possono mostrare una correlazione positiva o negativa, il che può fornire ulteriori dettagli sull'influenza delle caratteristiche musicali sulla popolarità delle tracce.

Questo tipo di analisi è fondamentale per comprendere meglio i fattori che contribuiscono al successo di una traccia musicale e può essere utilizzato per scoprire tendenze o informazioni rilevanti all'interno dei migliori brani degli ultimi anni.

7 Conclusioni

Il progetto ha raggiunto con successo l'obiettivo di creare un database innovativo che consente agli utenti di identificare stazioni radio in un determinato paese utilizzando il genere delle loro canzoni preferite o le più popolari su Spotify.

Guardando al futuro, il nostro progetto offre un ampio spazio per l'innovazione e l'espansione. Alcune delle direzioni possibili includono l'implementazione di una ricerca avanzata basata su preferenze musicali, ad esempio consentendo agli utenti di cercare stazioni radio in base a una combinazione di generi musicali, decenni specifici, paesi di origine e altro ancora. Questa funzionalità potrebbe essere implementata attraverso filtri avanzati e un'interfaccia utente intuitiva.

Per migliorare ulteriormente l'esperienza dell'utente, si potrebbe sviluppare un sistema di raccomandazione basato sulle preferenze musicali degli utenti, in particolare utilizzando algoritmi di machine learning per analizzare le abitudini di ascolto e suggerire stazioni radio o brani correlati che potrebbero piacere all'utente.

un'area di sviluppo interessante potrebbe riguardare il monitoraggio delle tendenze musicali, ovvero rilevare e analizzare costantemente le tendenze emergenti nella musica. Tale sistema potrebbe sfruttare algoritmi sofisticati per tenere traccia delle nuove uscite musicali, delle classifiche di streaming e dei comportamenti degli utenti, i quali integrati anche l'analisi delle conversazioni e delle condivisioni sui social media relative alla musica, potrebbe fornire un quadro completo delle tendenze in atto.

In sintesi, l'evoluzione del progetto potrebbe essere guidata dalla volontà di migliorare l'esperienza degli utenti, offrendo funzionalità più avanzate e un servizio più completo nel campo della scoperta musicale.

Riferimenti bibliografici

[1] Kaggle, *Top 10000 Songs on Spotify 1960-Now*, https://www.kaggle.com/datasets/joebeachcapital/top-10000-spotify-songs-1960-now?resource=download

[2] Spotify, *Web API*, https://developer.spotify.com/

[3] Rapid API Hub, *Radio World - 75,000+ Worldwide FM Radio stations*, https://rapidapi.com/dpthapaliya19/api/radio-world-75-000-worldwide-fm-radio-stations