K. J. Somaiya College of Engineering, Mumbai-77
(Autonomous College Affiliated to University of Mumbai)

MUSIC RECOMMENDATION SYSTEM

Submitted at the end of semester IV in partial fulfillment of requirements

Of Bachelors in Technology in Computer Engineering

by

Esha Shelar
Roll No: 16010120136
Arva Kachwala
Roll No: 16010120152
Rahul Tripathi
Roll No: 16010120153

Guide

Prof. Sushma Kadge

Department of Computer Engineering
K. J. Somaiya College of Engineering, Mumbai-77

Batch 2020 -2024

Certificate

This is to certify that the MINIPROJECT report entitled **Music Recommendation System** submitted by Esha Shelar, Arva Kachwala, Rahul Tripathi at the end of semester IV of SY B. Tech is a bona fide record for partial fulfilment of requirements for the degree of Bachelors in Computer Engineering of Somaiya Vidyavihar University, Mumbai.

_____ _____

Guide Head of the Department

Date:

Place: Mumbai-77

Certificate of Approval of Examiners

We certify that this MiniProject report entitled **Music Recommendation System** is a bona fide record of Mini project work done by Esha Shelar, Arva Kachwala, Rahul Tripathi during semester IV.

This Mini project work is submitted at the end of semester IV in partial fulfillment of requirements for the degree of Bachelors in Technology in Computer Engineering of University of Mumbai.

_____

Internal Examiner 1

_____

Internal Examiner 2

Date:

Place: Mumbai-77

DECLARATION

We declare that this written report submission represents the work done based on our and / or others' ideas with adequately cited and referenced the original source. We also declare that we have adhered to all principles of intellectual property, academic honesty and integrity as we have not misinterpreted or fabricated or falsified any idea/data/fact/source/original work/ matter in our submission.

We understand that any violation of the above will be cause for disciplinary action by the college and may evoke the penal action from the sources which have not been properly cited or from whom proper permission is not sought.

Signature of the Student **Roll No.**	**Signature of the Student** **Roll No.**
Signature of the Student **Roll No.**	**Signature of the Student** **Roll No.**

Date:

Place: Mumbai-77

MUSIC RECOMMENDATION SYSTEM

Group:

Esha Shelar: 16010120136

Arva Kachwala: 16010120152

Rahul Tripathi: 16010120153

Division: Computer-C

Batch: C1

Index:

1. Introduction:

1.1 Background:

Music is one of the popular entertainment media in the digital era. Music is considered as the work of human creativity to express ideas and emotions in the form of sounds that consist of melody, harmony and rhythm. Music can be categorized into several genres, such as pop, rock, jazz, blues, folk etc. Listening to music in the digital age is easier because of the various applications on the smartphone that can play music offline and online. Nowadays, the availability of digital music is very abundant compared to the previous era, so to sort out all this digital music is very time consuming and causes information fatigue. Therefore, it is very useful to develop a music recommender system that can search music libraries automatically and suggest songs that are suitable for users.

1.2: Problem Statement:

It is necessary to develop a music recommender system that can search in the music libraries automatically and suggest suitable songs to users. In our recommender system, the music provider can predict and then offer the appropriate songs to their users based on the characteristics of the music that is present in the imported playlist. Our research would like to develop a music recommender system that can give recommendations based on similarity of features in the playlist. For this purpose, our system will use machine learning and deep learning techniques available in python.

1.3: Scope:

- Importing playlists from Spotify users.
- Feature extraction - From the imported playlist, genres, singers, song names,

year of release etc is noted and analyzed.

● A Public music dataset will be downloaded for the project. The recommendation system will generate appropriate suggestions based on the imported playlist.

● The generated results will be displayed to the user.

● Results of the recommender system will be compared with the suggestions offered by the Spotify music app. Appropriate conclusions will be made.

1.4: Brief Description of undertaken project:

The main objective of this project is to develop a recommendation system to recommend songs such that it is close to the user's choice. The system will determine the musical preferences of the users based on the analysis of a playlist that the user will be asked to import. The user will have to enter his/her Spotify user id and password and the name of the playlist. By importing the playlist, the system is able to estimate what artist or group would match user preferences to the user at a given time. The project will make use of cosine similarity to recommend songs similar to the user's taste. The system will be integrated on a website.

2. Literature review:

In this section, we survey existing research on music recommendation.

There are various approaches to music recommendations, including:

- Expert-based approaches [6] that are based on human annotation of music data
- Social based approaches [7] that characterize items based on social
- Content based approach [4] that analyzes the audio content for characterizing tracks.
- Collaborative Filtering (CF) approaches [3] that analyze listening patterns by many users, in order to establish similarities across users and items

Among the above approaches, CF is widely used in large scale recommendation tasks and has received relatively good performance. However, music preference is subjective. So, the assumption behind the CF method that users with similar listening behaviour have similar taste on music is vulnerable. CF also suffers that it can hardly recommend a new song that no user ever listened to. Different from CF, CB methods try to discover music similarity based on their audio or signal information and recommend users the songs similar to their previously listening ones.

Content based approach attempts to recommend music that is perceptually similar to what the user has previously listened to, by measuring the similarity between audio signals. This approach requires the definition of a suitable similarity metric. Such metrics are often defined ad hoc, based on prior knowledge about music audio, and as a result they are not necessarily optimal for the task of music recommendation. Because of this, user preference data is used to tune similarity metrics.

3. Implementation Details:

The overall architecture of the system can be divided into the following parts:

3.1 Modules:

1) Collection of Data
2) Pre-processing of data
3) Feature Engineering
4) Importing playlist from Spotify
5) Model Building
6) Testing and evaluation

3.2 Responsibility Division:

Task	Esha	Arva	Rahul
Data Analysis			
Collection	X	X	X
Pre-processing	X	X	X
Research on existing models:			
Literature review	X	X	X
Model Building:			
Importing playlist	X	X	X
ML model	X	X	X
Testing:			
Testing approach	X	X	X
Test Cases	X	X	X
Presentation:			
Powerpoint	X	X	X
Report	X	X	X

3.3 Project Timeline:

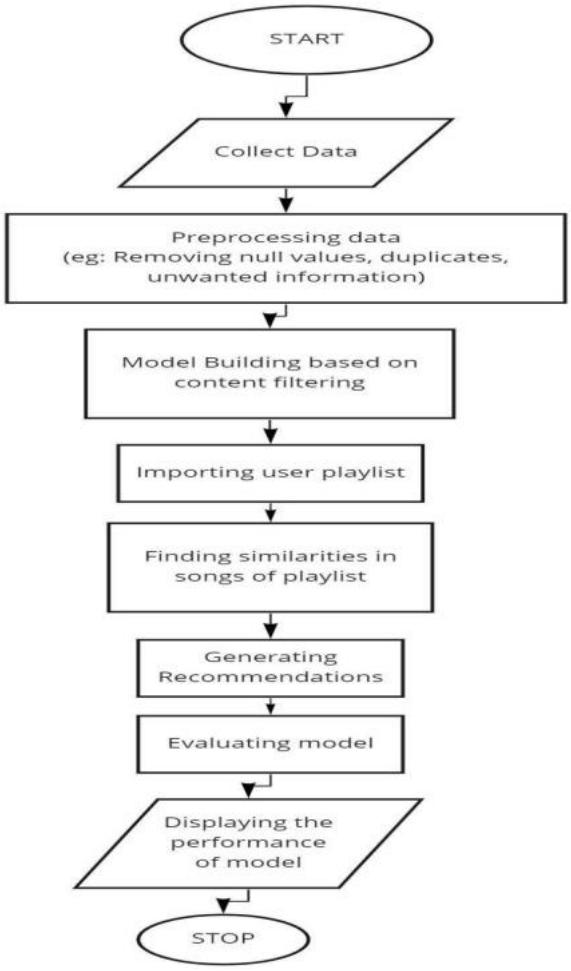
After necessary research about the optimal tools and softwares, the project was divided into smaller parts, whose responsibilities were taken according to the team members area of expertise/ interest. Each part had its timeline and most of the time were worked upon parallelly.

- Project planning and resource research: (14th January – 28th January)
- Installing tools: (28th January – 31st January)
- Dataset collection: (1st February – 5th February)
- Data pre-processing: (6th February – 15th February)
- Feature Engineering: (16th February – 20th February)
- Importing playlist from Spotify: (21st February – 28th February)
- Model Building: (1st March – 20th March)
- Testing: (21st March – 2nd April)
- Website and Deployment: (3rd April – 9th April)
- Report: (10th April – 22nd April)

3.4 Flowcharts/Algorithms:

System Architecture

Module wise flow diagram



3.5 Software Requirements

3.5.1 Programming languages

Python is a critical technology in recommendation systems. We have used python in our project as it is a popular interpreted language that in combination with machine learning has become one

of the most common methods for building recommendation systems.

TensorFlow package that makes building, evaluating, and serving sophisticated recommender models easy. TensorFlow Recommenders (TFRS) is a library for building recommender system models. We've used it in our project as it helps

with the full workflow of building a recommender system: data preparation, model formulation, training, evaluation, and deployment.

3.6 Software development tools

- Jupyter Notebook - Jupyter Notebook is an interactive computational environment, which allows us to combine code execution, rich text, mathematics, plots and rich media. Its flexible interface allows users to configure and arrange workflows in data science, scientific computing, computational journalism, and machine learning. Hence, we've used a Jupyter notebook for the implementation of our code.
- Anaconda Navigator is a desktop graphical user interface (GUI) included in Anaconda® distribution that allows users to launch applications and easily manage conda packages, environments, and channels without using command-line commands. We've used anaconda navigator to access Jupyter Notebook.

3.7 Difficulties faced

1. Users' tastes and musical needs are highly dependent on a multitude of factors, which are not considered in sufficient depth in our recommender system because satisfying the users' musical entertainment needs requires taking into account intrinsic, extrinsic, and contextual aspects of the listeners. For instance, personality and emotional state of the listeners (intrinsic) as well as their activity (extrinsic) are known to influence musical tastes and needs.
2. Memory-based collaborative filtering is a form of technique that tries to find users who are similar to active users and use their preferences to predict the rankings of active users. Use community preferences. Our project hasn't used collaborative filtering because of the unavailability of data of the vast majority of users and their music preferences.

4. Working Outputs:

- **Importing playlists from spotify using spotipy library**

```
In [25]:
scope = 'user-library-read'
token = util.prompt_for_user_token(
    scope,
    client_id= client_id,
    client_secret=client_secret,
    redirect_uri='http://localhost:8888/callback'
  )
sp = spotipy.Spotify(auth=token)
playlist_dic = {}
playlist_cover_art = {}

for i in sp.current_user_playlists()['items']:
    playlist_dic[i['name']] = i['uri'].split(':')[2]
    playlist_cover_art[i['uri'].split(':')[2]] = i['images'][0]['url']

print(playlist_dic)
```

```
{'My Playlist #1': '7cPlkVusc4vTUXaQzM0GqA'}
```

- **Creating playlist dataframe**

```
playlist_df.head()
```

:7]:

	artist	track_name	track_id	url	date_added
8	The Wallflowers	One Headlight	0z1b34WikhOH9ZxU8QDWcv	https://i.scdn.co/image/ab67616d00001e02e542fb...	2022-03-15 11:28:11+00:00
2	Elton John	Can You Feel the Love Tonight - End Title/ Fro...	0EANX0OVKSCcmarY50Xa4p	https://i.scdn.co/image/ab67616d00001e02660aad...	2022-03-15 11:25:03+00:00
1	Demi Lovato	Give Your Heart a Break	3sP3c86WFjOzHHnbhhZcLA	https://i.scdn.co/image/ab67616d00001e0283fc43...	2022-03-15 11:24:51+00:00
0	Elvis Presley	Are You Lonesome Tonight	6vXZs9rEQF7Nd6O8Ue7NrT	https://i.scdn.co/image/ab67616d00001e02a22455...	2022-03-15 11:23:19+00:00

- **Visualizing cover art**

```
visualize_cover_art(playlist_df)
```

```
C:\Users\ishas\AppData\Local\Temp/ipykernel_2016/3708628251.py:10: MatplotlibDeprecationWarning: Pas
ee-element position specification is deprecated since 3.3 and will be removed two minor releases lat
  plt.subplot(len(temp) / columns + 1, columns, i + 1)
```

- **Building the model**

```
def generate_recommendation(spotify_data, playlist_vector, nonplaylist_df):

    non_playlist = spotify_data[spotify_data['track_id'].isin(nonplaylist_df['track_id'].values)]
    non_playlist['sim'] = cosine_similarity(nonplaylist_df.drop(['genre','track_id','artist_name','track_name'], axis = 1).va
    non_playlist_top20 = non_playlist.sort_values('sim',ascending = False).head(20)
    non_playlist_top20['url'] = non_playlist_top20['track_id'].apply(lambda x: sp.track(x)['album']['images'][1]['url'])

    return  non_playlist_top20
```

- **Creating dataframe for recommended songs**

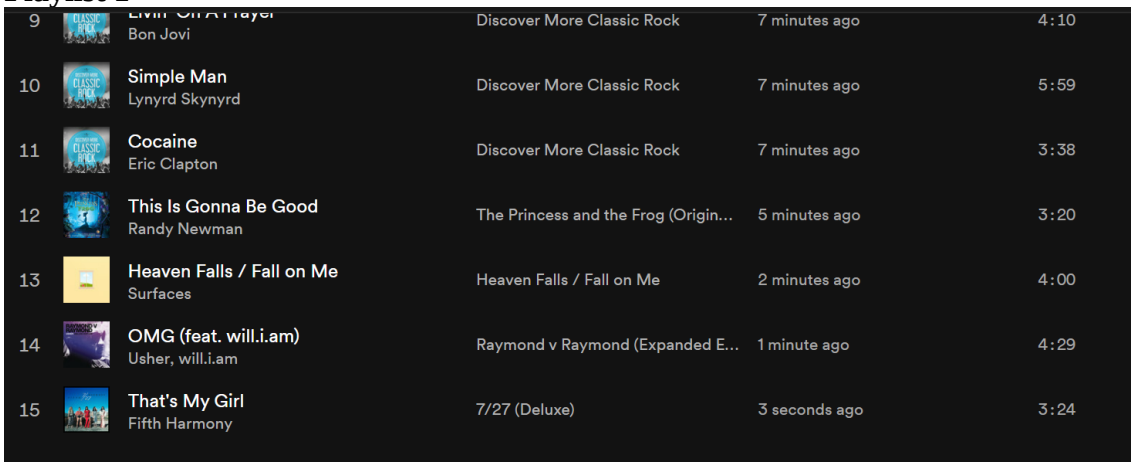
	genre	artist_name	track_name	track_id	popularity	acousticness	danceability	duration_ms	energy	instrumentalness	...
166392	Rock	Aerosmith	I Don't Want to Miss a Thing	225xvV8r1yKMHErSWivnow	78	0.042172	0.043898	298760	0.046107	0.040866	...
166976	Rock	Red Hot Chili Peppers	Give It Away	0uppYCG86ajpV2hSR3dJJ0	74	0.097752	0.104290	282907	0.106950	0.098575	...
166291	Rock	Mötley Crüe	Kickstart My Heart	7Gonnnall2s19OCQO1J7Tf	74	0.041144	0.046211	284027	0.051600	0.041164	...
225864	Rock	Los Fabulosos Cadillacs	Vasos Vacíos - Remasterizado 2008	198qSChSMQFuSimdeeY9gK	73	0.041977	0.044559	279987	0.044302	0.039508	...
225700	Rock	Cristian Castro	Por Amarte Así	2bvO1KXbUd4vySCWGNqIla	71	0.048173	0.048173	272200	0.048373	0.043646	...

5 rows × 43 columns

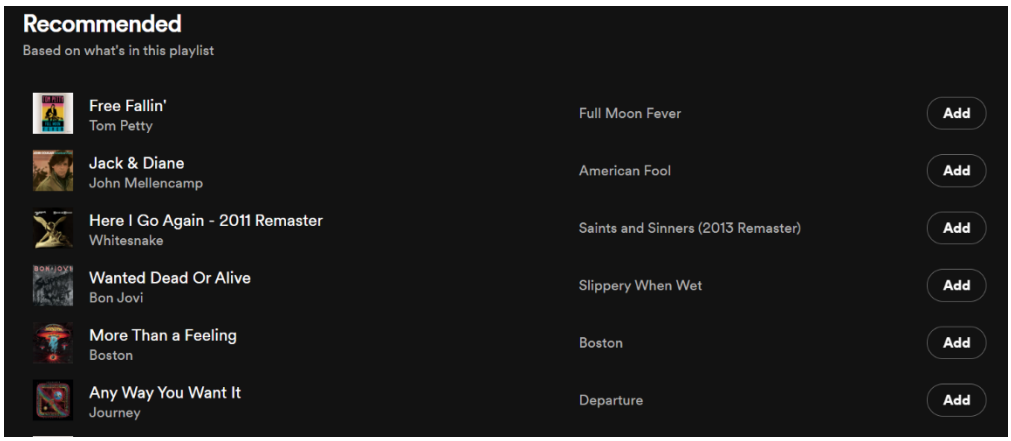
- **Visualizing cover art for top 20 songs**

5. Test Cases

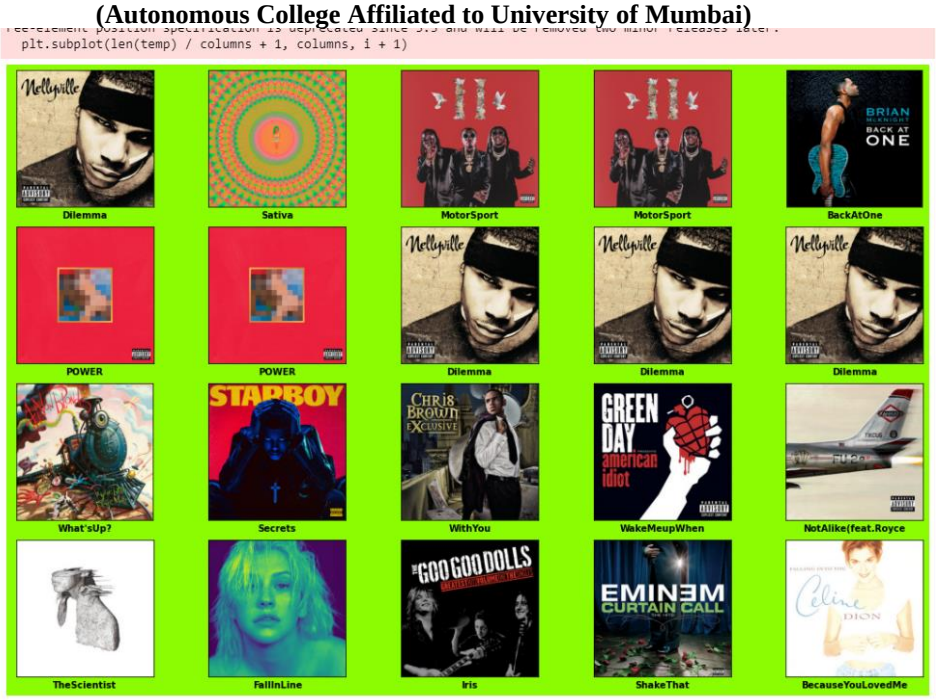
Test Case One: Playlist Contains Only Rock and Pop music
Playlist 1



Songs Recommended by Spotify

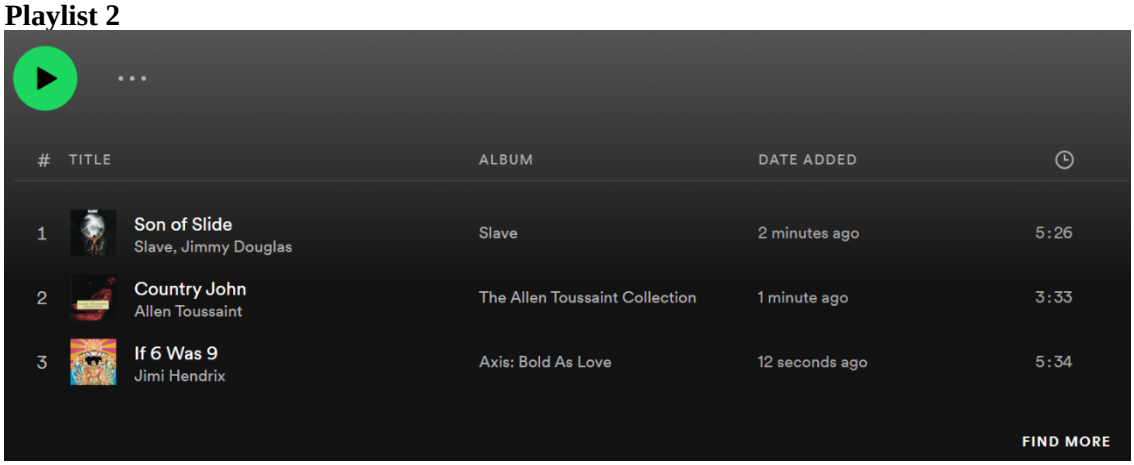


Songs by our Recommender

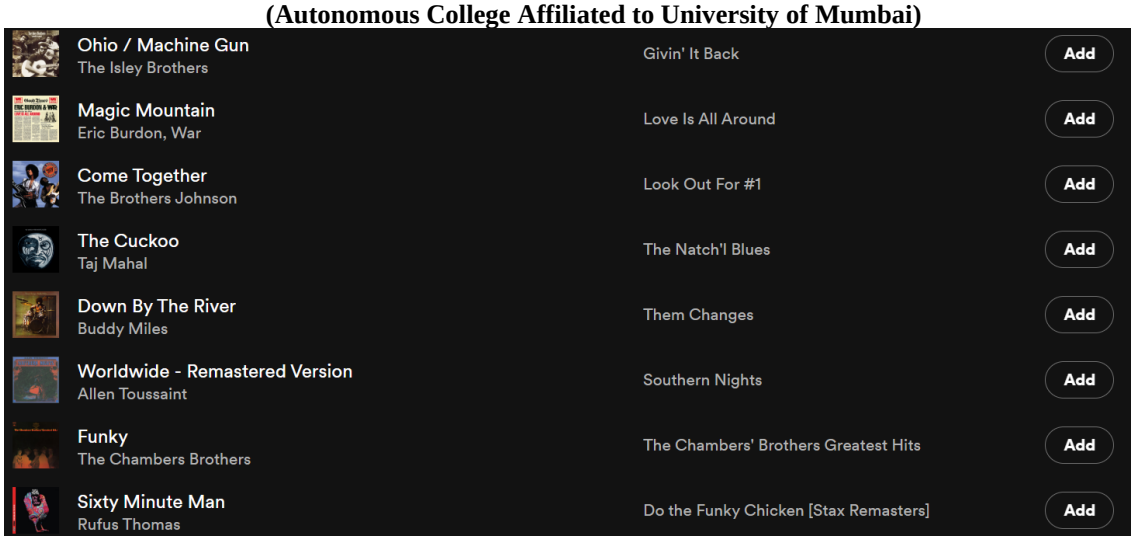


Conclusion For test case: Iris" is a song by American alternative rock band Goo Goo Dolls.
Dilemma is a famous pop song

Test Case Two: Playlist Contains genre "soul"
Playlist 2



Songs Recommended by Spotify



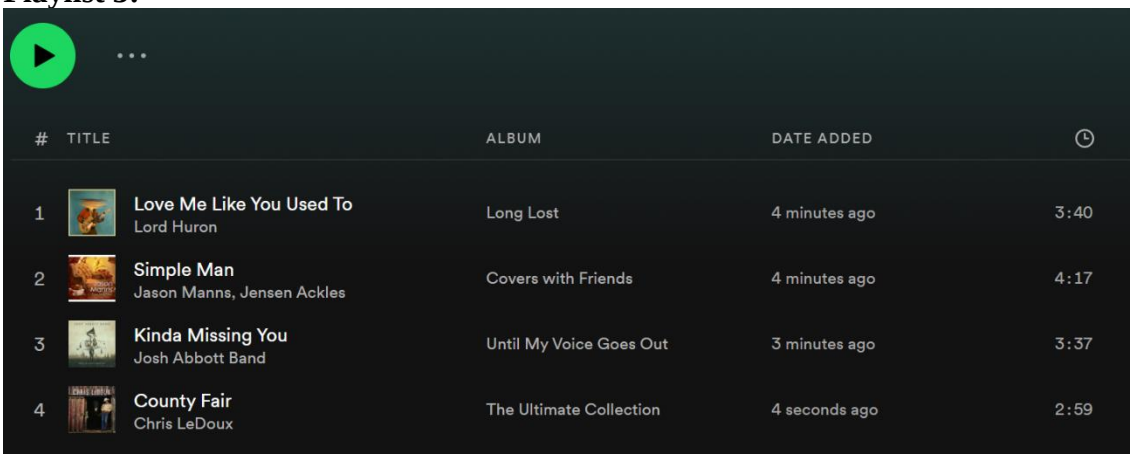
Songs by our Recommender (most of the songs belong to soul genre)

```
visualize_cover_art(top15)
```

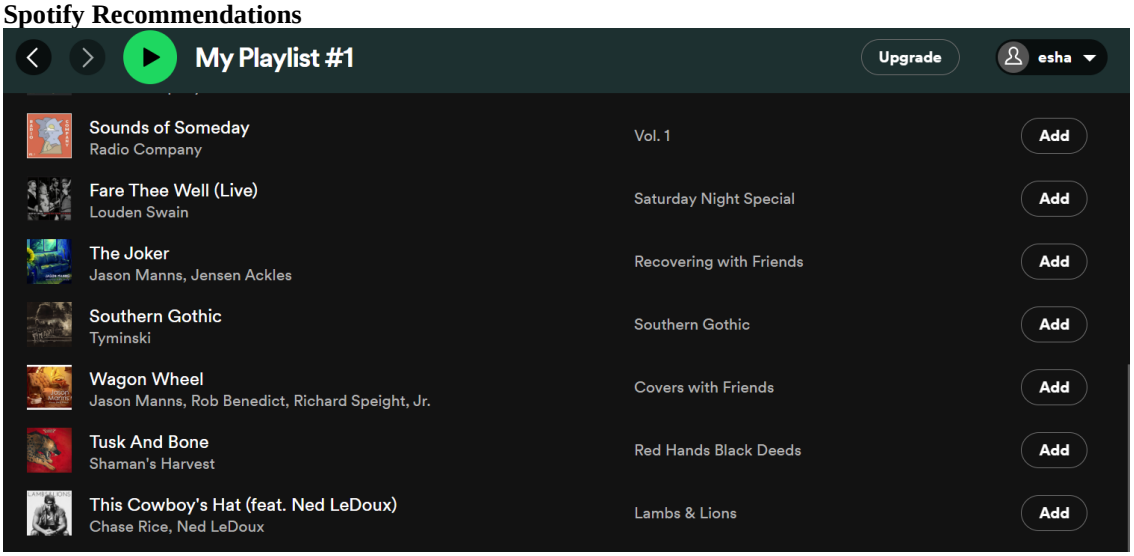
```
C:\Users\ishas\AppData\Local\Temp/ipykernel_28460/3708628251.py:10: MatplotlibDeprecationWarning: Passing non-integers as th
ree-element position specification is deprecated since 3.3 and will be removed two minor releases later.
  plt.subplot(len(temp) / columns + 1, columns, i + 1)
```

Test case Three: Playlist contains Country music

Playlist 3:



Spotify Recommendations



Recommendations by our model

```
plt.subplot(len(temp) / columns + 1, columns, i + 1)
```

6. Future Scope:

- Building a website for the recommender model with appealing UI/UX design.
- Along with recommendations, the app will feature music streaming.
- Social media integrations.
- Advanced search is one of the most demanded features. We'll allow users to search not only by the song name or artist but also by mood, genre, remixes, etc.

7. GitHub Implementation:

https://github.com/iamrahultripathi/MRS.git

8. Acknowledgements

This project would not have been possible without the guidance of Mrs. Sushma Kadge ma'am who helped in all ways possible. She would not only solve our doubts but also shared her ideas which helped us bring our project to a new level.

9. References

Citation Map

1. Music Recommender System Based on Genre using Convolutional Recurrent Neural Networks, 4th International Conference on Computer Science and Computational Intelligence 2019 (ICCSCI), 12–13 September 2019 https://www.sciencedirect.com/science/article/pii/S1877050919310646/pdf?md5=4f 9a5242eb223b5c96c9ebf130855467&pid=1-s2.0-S1877050919310646-main.pdf

2. Schedl M, Zamani H, Chen CW, Deldjoo Y, Elahi M. Current challenges and visions in music recommender systems research. Int J Multi Inform Ret. (2018) 7:95–116. doi: 10.1007/s13735-018-0154-2 https://link.springer.com/article/10.1007/s13735-018-0154-2

3.J. B. Schafer, D. Frankowski, J. Herlocker and S. Sen, "Collaborative filtering recommender systems" in the adaptive web, Springer, pp. 291324, 2007. https://link.springer.com/chapter/10.1007/978-3-540-72079-9_9

4. M. J. Pazzani and D. Billsus, "Content-based recommendation systems" in the adaptive web, Springer, pp. 325341, 2007. https://link.springer.com/chapter/10.1007/978-3-540-72079-9_10

5. Schedl M, Gómez E, Urbano J. Music information retrieval: recent developments and applications. Foundations and Trends in Information Retrieval. 2014 Sept 12; 8 (2-3): 127-261. DOI: 10.1561/1500000042 http://hdl.handle.net/10230/27565

6.M. Castelluccio, The music genome project Strategic Finance, vol. 88, no. 6, pp. 57, 2006. https://www.proquest.com/docview/229789135?parentSessionId=jbI9jrlUUJoZAxPMv71VitnU0jNfh4d17nvLIeWkUyQ%3D

7.A. Nanopoulos, D. Rafailidis, P. Symeonidis and Y. Manolopoulos, "Musicbox: Personalized music recommendation based on cubic analysis of social tags Audio Speech and Language Processing", IEEE Transactions on, vol. 18, no. 2, pp. 407412, 2010.