# Creating Music Recommendation System using Real-Time Data

**Minor Project Synopsis Report**

*Submitted in partial fulfillment of the requirement of the degree of*

**BACHELORS OF TECHNOLOGY**

**in**

**CSE with Specialization AI & ML**

*to*

## K.R Mangalam University

*by*


**Maanav Aditya Sinha (2201730085)**

**Muskaan Jain (2201730097)**

**Sachin Kumar (2301731117)**

**Tannu Rana (2201730093)**


Under the supervision of

**Mr. Rahul Singh**


Department of Computer Science and Engineering School of Engineering and Technology

K.R Mangalam University, Gurugram- 122001, India


September 2024

# INDEX

## ABSTRACT

Music recommendation systems use a variety of techniques to suggest music to users which includes user rating and acoustic features, content-based approaches, hybrid systems, contextual information, song similarity, pandora. Music recommenders have traditionally used content-based approaches, in contrast to other recommendation domains, which uses collaborative filtering techniques. These systems combine different recommendation models to make more efficient recommendations than systems that use a single technique.

These systems are important because they help users discover new music and save time finding songs, especially useful with a large amount of music available on digital streaming platforms. It analyzes the user's listening history and preferences to provide a tailored playlist. It increases user’s engagement and satisfaction, which can help Music Streaming Platform retrain customers. It can help Music Service Providers increase sales by suggesting similar tracks to users, who buy music and make database marketing decisions, as it provide insights into user’s behaviors.

Some systems, like Amazon Echo, Alexa, can listen to the user’s environment to infer their mood and music preferences. CARS (context-aware recommendation system), which takes contextual factors, such as mood, location and weather into consideration in order to recommend relevant songs to the user. These also suggest similar artists or songs to a user’s query; however, this is not always a true recommendation.

Pandora streaming service recommends music, based on a song’s musical traits. Several music recommendation apps use AI, including Spotify, Apple Music, Tidal, and Pandora. These apps analyze the user's listening habits and preferences to provide personalized music suggestions.

Music recommendation apps are software that suggests songs, by analyzing the choice of the user and based upon a Complex Algorithm. If you are a Music Lover Individual, then you will be in search of the Good Music Recommendation App.



***KEYWORDS:  Music Streaming Platform, Music Service Providers, CARS, Pandora, AI, Spotify***

# 1. INTRODUCTION

Music is an artform which is enjoyed by millions of people around the world. The music industry continues to grow and change. Today, artists have many access points in order to distribute, share and promote their music to their listeners.

One of the factors of the increased growth of artists and music is digital streaming services like Spotify, Apple Music and Amazon Music. Not only do these platforms allow convenient access to a vast library of songs, allowing listeners to discover and enjoy music easily, but also allows artists to connect directly to their fans and share their creative works. This raises an issue, since the digital music library is abundant, sorting them will be time consuming and can cause information fatigue. Therefore, a music recommendation system should be developed in order to search music libraries automatically and suggest songs that are suitable for users.

For the pre-final year project, our aim is to create a music recommendation system, with the purpose of recommending songs based on personal preference of the user. This synopsis explores the motivation, tools used, and methodology applied which was needed to achieve this project.

## 2. MOTIVATION

Earlier, music was recommended primarily through a more traditional approach. These approaches were personal recommendations, music charts and magazines, album covers and live performances. These methods made music less accessible, since they were largely subjected and heavily relied on human interaction and cultural context. However, the emergence of digitalization and efficient audio compression technologies like MP3 fulfilled the dream of making millions of music titles accessible to millions of users.

Music was easy to access due to digitalization and globalization. The rise of digital streaming services has revolutionized the music industry. Music platforms like Spotify provide an exponential amount of music. The drawback is that the listeners sometimes feel overwhelmed, since they have to make a choice from a vast collection of songs. There was also a need to manage music and improve user's satisfaction.

In 1994, Shardanand and Maes developed the first widely recognized music recommendation system, Ringo, which pioneered a significant step in the evolution of music recommendation by utilizing collaborative filtering techniques. Today's advancements in music recommendation systems focuses on hybrid recommendation approaches, machine learning algorithms, natural language processing etc.

Here, our goal is to develop a music recommendation system that will analyze the metrics of popular music streaming applications and return recommended songs, based on the genre the user had specified.

# 3. LITERATURE REVIEW

Music recommendation systems have their roots in the 1990s when the first internet-based music streaming services emerged. The pioneering systems of this era used **content-based filtering** and early forms of **collaborative filtering** to suggest music to users. Content-based filtering relied on attributes of the music, such as genre, tempo, and instrumentation, to generate recommendations. Collaborative filtering, on the other hand, focused on user behaviors, recommending music based on the preferences of users with similar listening histories.

One notable example from this period was **Pandora**, launched in 2000, which employed the **Music Genome Project** to recommend tracks based on their musical characteristics. Pandora's approach was highly granular, breaking down songs into hundreds of musical attributes such as harmony, melody, and rhythm to match songs to users' preferences based on these features.

During the 2000s, the development of **collaborative filtering algorithms** significantly advanced music recommendation systems. Systems such as **Last.fm** popularized the use of collaborative filtering by recommending music based on the listening habits of similar users.

Matrix factorization techniques, particularly **Singular Value Decomposition (SVD)**, became widely adopted during this time. These techniques were used to reduce the dimensionality of user-item interaction data, making it possible to handle large datasets and improve the accuracy of recommendations.

The **Netflix Prize** competition (2006) further pushed the development of recommendation systems, inspiring advancements that spilled over into music recommendations. While not directly about music, the techniques developed during the competition, such as hybrid recommendation models combining collaborative filtering with other methods, were later applied in music platform

In the 2010s, the music recommendation landscape saw the rise of **hybrid systems** that combined both collaborative filtering and content-based methods. Streaming services like **Spotify** and **Apple Music** led the charge, implementing advanced recommendation algorithms that fused user behavior data with song metadata. **Spotify's Discover Weekly**, introduced in 2015, became an iconic example of a highly personalized recommendation system, relying on both content-based features and collaborative filtering to offer curated weekly playlists.

Machine learning, particularly **deep learning**, began playing a significant role in music recommendations during this period. The use of deep learning models allowed systems to automatically extract features from audio signals, improving content-based filtering by eliminating the need for manual annotation of song attributes. Convolutional Neural Networks (CNNs) were often applied to audio spectrograms to capture musical patterns and make more accurate recommendations.

In recent years, music recommendation systems have incorporated **context-aware** models, which go beyond just user preferences and song attributes. These systems consider the user's location, time of day, current activity, and even mood to offer highly dynamic music suggestions. Platforms like **YouTube Music** and **Amazon Music** have introduced context-aware features that adapt recommendations based on listening environments.

**Deep learning** models, such as **Recurrent Neural Networks (RNNs)** and **Transformers**, have also been employed to model sequential user interactions with music. These models have enhanced the ability to recommend songs based on users' evolving tastes and habits over time. Additionally, advancements in **natural language processing (NLP)** have allowed recommendation systems to incorporate textual metadata such as song lyrics, user reviews, and social media posts into their algorithms.

Today's music recommendation systems combine a variety of approaches—ranging from deep learning to hybrid and context-aware models—offering users more personalized and sophisticated listening experiences

# 4. PROBLEM STATEMENT

Today, the users are overwhelmed by the large library of available music in the vast landscape of digital music streaming. This makes it challenging to discover music that align with their personal tastes. To address this issue, it is necessary to develop a music recommendation system which can provide personalized music suggestions.

The recommendation system must integrate various techniques, such as collaborative filtering, content-based filtering and hybrid methods, in order to predict user’s preference accurately. It also ensures scalability in order to handle datasets and incorporates real-time data processing, in order to adapt the evolving preference of the user.

The ultimate goal is to enhance user satisfaction and engagement by delivering precise and diverse music recommendations that cater to the user’s taste.

# 5. OBJECTIVES

The objective of this project is to develop a music recommendation system using real-time data from popular music streaming services, where the user will specify the genre and show the recommended songs based on the genre.

To achieve this objective, this project is divided into following sub-objectives:

1. Study different APIs provided by Music Streaming Applications and understand the metric of determining a song's popularity.

2. Extract the data from the APIs and process the extracted data by performing normalization.

3. To create a ranking algorithm that will rank songs based on the combined popularity score.

4. Develop a machine learning model to refine the recommendation system by predicting the rank of the song from a specific genre from the custom dataset.

5. Create a User-Interface using Streamlit where the user can input a genre and get real-time recommendations.

6. Record the prediction results provided by the model, compare the predicted and the actual outcome, and integrate it in the User-Interface once fully developed.

By completing the following sub-objectives, the development of the music recommendation system will be completed. The completion of objective will return the following benefits:

1. Discovery: This project will broaden user's musical horizons by allowing them to discover new music they might haven't found on their own.

2. Retention: The music recommendation system improves the user retention by consistently delivering relevant and enjoyable recommendations. This allows users to stay with the platform longer.

3. Engagement: The project encourages users for repeat visits by providing

music that they are likely to enjoy. This increases the user engagement.

4. Scalability: The recommendation system handles large volumes of data. Due to this, the music recommendation system provides real-time recommendations to the users as their preference evolves.

## 6. Tools/Technologies Used

For this project, following APIs were used:

1. Spotify API: It provides a wide range of functionality, such as data retrieval from artists, albums or shows, for developers.

2. Youtube API: It provides the ability to retrieve feeds related to videos, users and playlists.

In order to achieve this project, we have used python as the primary programming language due to its simplicity and allowing creating an application prototype easily and following python library were used:

1. Pandas: Pandas is a powerful data analysis and manipulation library which is ideal for loading and working with structured data. It is capable of summarizing information, handling missing and inconsistent data in the dataset and performing operations such as filtering, sorting and transformation.

In order to install Pandas, run the following command:

pip install pandas

2. Spotipy: It is a Python client for the Spotify Web API. This library is capable of fetching track details, artist data, playlists and popularity scores. OAuth 2.0 is required for authentication. A Spotify Developer account needs to be set up and credentials should be created.

   In order to install this library, run the following pip command:

   pip install spotipy

3. Google-api-python-client: This is the official client library for Youtube API in order to retrieve videos and metadata (including views and other metrics). API key is required for basic access and OAuth 2.0 for user-specific data.

   In order to install this library, run the following pip command:

   pip install google-api-python-client

4. Streamlit: In order to create a fully interactive web interface, Streamlit library is used. This library allows implementation of the model made, creating input fields for users to specify their preferences, and real-time interaction.

In order to install Streamlit, run the following command:

```
pip install streamlit
```

# 7. METHODOLOGY

The aim of this project is to create a music recommendation system, using real-time data taken from various music streaming platforms, that will recommend popular songs when a user specifies a genre. It will be based on the song's popularity, play count and other factors.

This task of creating Music Recommendation System can be structured into following parts:

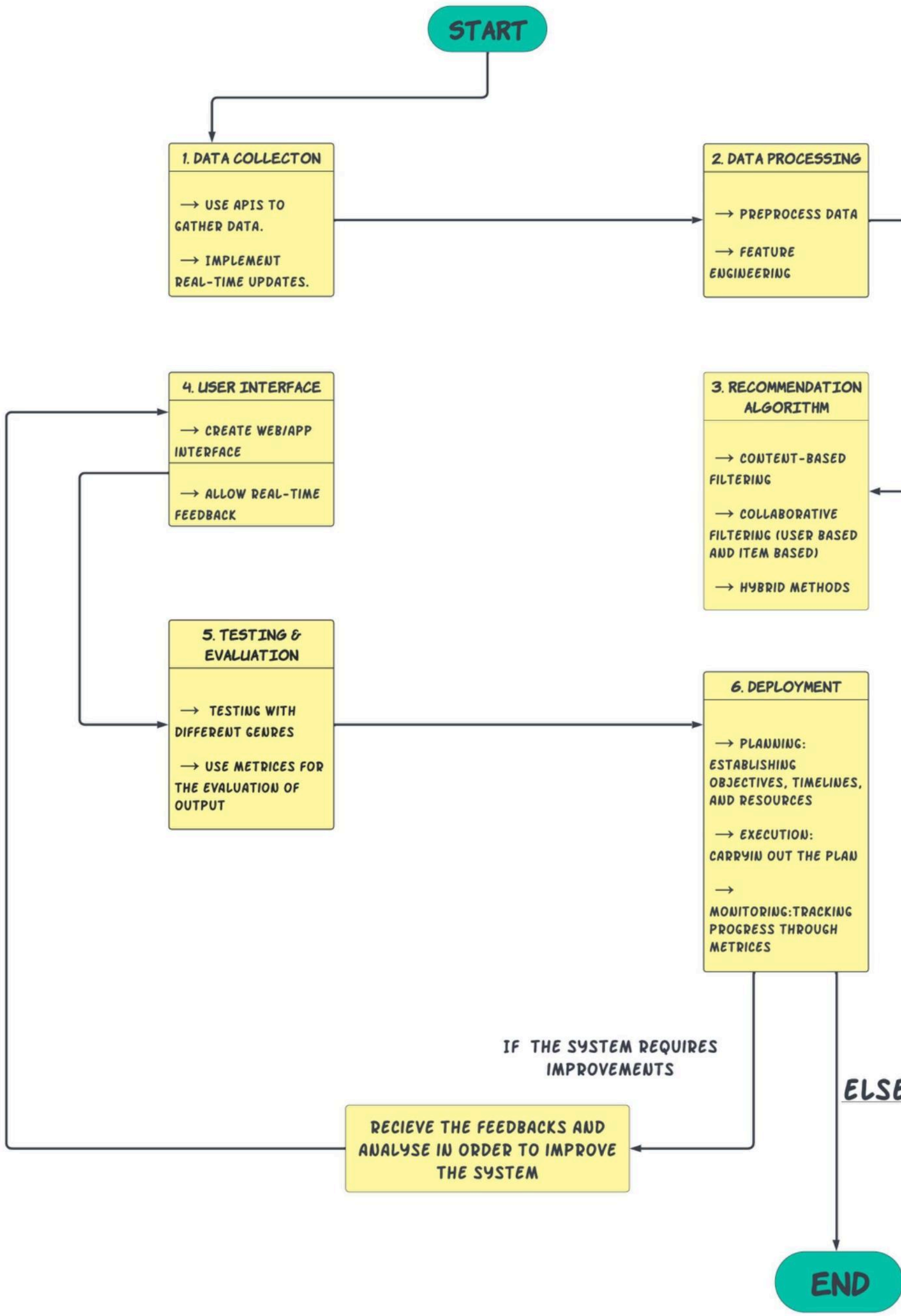
1. Data Collection: Music APIs will be chosen to access song data, popularity metrics and real-time user interactions. Once the APIs are selected, the real-time data is collected.

2. Data Normalization: Different music streaming platforms have different standards to determine a song’s popularity. For example: Spotify determines a song’s popularity by providing popularity scores on a scale of 0 to 100, whereas Youtube Music determines a song’s popularity through the views. In order to rank the songs consistently, the data needs to be normalized using techniques such as min-max scaling in order to bring all the metrics onto the same scale.

3. Ranking Algorithm: Based on the combined popularity score, the songs are ranked by implementing a weighted ranking system.

4. User Interface: The user interface is made, where the user can input a genre and get the real-time song recommendation.

5. Testing and evaluation: The system is tested with different genres in order to ensure that the output system returns are relevant and accurate.


START
1. DATA COLLECTON
→ USE APIS TO GATHER DATA.
→ IMPLEMENT REAL-TIME UPDATES.
2. DATA PROCESSING
→ PREPROCESS DATA
→ FEATURE ENGINEERING
3. RECOMMENDATION ALGORITHM
→ CONTENT-BASED FILTERING
→ COLLABORATIVE FILTERING (USER BASED AND ITEM BASED)
→ HYBRID METHODS
4. USER INTERFACE
→ CREATE WEB/APP INTERFACE
→ ALLOW REAL-TIME FEEDBACK
5. TESTING & EVALUATION
→ TESTING WITH DIFFERENT GENRES
→ USE METRICES FOR THE EVALUATION OF OUTPUT
6. DEPLOYMENT
→ PLANNING: ESTABLISHING OBJECTIVES, TIMELINES, AND RESOURCES
→ EXECUTION: CARRYIN OUT THE PLAN
→ MONITORING:TRACKING PROGRESS THROUGH METRICES
IF THE SYSTEM REQUIRES IMPROVEMENTS
ELSE
RECIEVE THE FEEDBACKS AND ANALYSE IN ORDER TO IMPROVE THE SYSTEM
END

# REFERENCES


1. Gómez, E., Herrera, P., & Amatriain, X. (2001). Music recommendation: collaborative filtering and beyond. *International Symposium on Music Information Retrieval*.

2. Linden, G., Smith, B., & York, J. (2003). Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing*, 7(1), 76–80.

3. Van den Oord, A., Dieleman, S., & Schrauwen, B. (2013). Deep content-based music recommendation. *Advances in Neural Information Processing Systems*.

4. Pachet, F. (2003). Content management for electronic music distribution. *Communications of the ACM*, 46(4), 71–75.

5. Schedl, M., Gómez, E., & Urbano, J. (2014). Music information retrieval: Recent developments and applications. *Foundations and Trends in Information Retrieval*, 8(2-3), 127-261.

6. researchgate.net/publication/277714802_A_Survey_of_Music_Recommendation_Systems_and_Future_Perspectives

7. music recommendation system using python thecleverprogrammer.com/2023/07/31/music-recommendation-system-using-python/

8. Music Recommendation System Using Real Time Parameters ieeexplore.ieee.org/document/10134257

9. Y. Zheng, "Context-aware collaborative filtering using context similarity: An empirical comparison", Information, vol. 13, no. 1, pp. 42, Jan. 2022.

10. Building a Spotify Recommendation System https://medium.com/@obielinda/building-a-spotify-recommendation-system-d4b67018eac2

11. Music Recommendation System Using Deep Learning by Raj Kumar Saw, Sumit Kumar, Nidhi Mishra on https://www.ijraset.com/research-paper

12. researchgate.net/publication/379009533_Music_Recommendation_Systems_Techniques_Use_Cases_and_Challenges

13. https://www.researchgate.net/publication/234057252_ContextAware_Recommender_Systems_for_Learning_A_Survey_and_Future_Challenges

14. researchgate.net/publication/360114620_Collaborative_Filtering_Based_Hybrid_Music_Recommendation_System

15. www.researchgate.net/publication/366244851_Machine_Learning_Based_Music_Genre_Classification_and_Recommendation_System

16. www.researchgate.net/publication/314132367_Machine_Learning_Algorithms_for_Recommender_System_-_a_comparative_analysis

17. www.researchgate.net/publication/369466059_Machine_Learning_Classification_Techniques_Applied_in_modern-day_Music_Recommendation_Systems

18. www.researchgate.net/publication/381853790_SPOTIFY_RECOMMENDATION_SYSTEM