

---

## An user-friendly algorithm which shuffles music according to mood

Soumalya Pakrashi<sup>1\*</sup>, Sannoy Mitra<sup>2\*</sup>

<sup>1</sup>Computer Science, The Bhawanipur Education Society College, India

<sup>2</sup>Electronics and Communication Engineering, Institute of Engineering & Management, India

Email: [soumalyapakrashi@gmail.com](mailto:soumalyapakrashi@gmail.com), [officialsannoy@gmail.com](mailto:officialsannoy@gmail.com)

### Abstract

Music is one of the most widely recognised art forms all around the world. Also, music is something people connect with in every mood. Our objective in this paper is to design a framework where whenever the user plays the shuffle option, the online music system suggests songs according to the mood. The suggestions will be directly simulated on the basis of the previous song the person has heard and the duration for which he/she has heard. Depending on a person's liking to a genre at certain period of time, the system shall roll another song of the same genre so that the user is satisfied. With consideration to the Fisher-Yates algorithm, we don't randomly select a song. Rather we aim to create a database where the songs are categorised and played according to the user's choice.

**Keywords:** *Shuffle, duration, genre, Fisher-Yates*

---

### 1. Introduction

Music over the years has evolved dramatically in terms of lyrics, voice treatment, soundscape and approach. Each part of the world has brought its own taste to the world of music. The evolution of various genres of music is basically catering to the choice of music lovers. Predominantly music can be classified according to the following genres:

Blues; Comedy; Country; Easy Listening; Electronic; Folk; Hip Hop; Jazz; Pop; Rock

A lover of Blues may or may not be a lover of Electronic music. Or a person's musical choice can change according to his/her current state of mind. For example, a man who has been working all day may prefer Country songs on his way home while a party reveller may like Hip Hop music. Thus the choice of music is temporary and varies with situations. In the following paper, we aim to create an UI which will shuffle songs according to the user's current state of mind or choice. It tries to steer clear of playing unwanted songs of unwanted genres thereby irritating the user all the more. Rather our algorithm may be of immense help in music therapy and act as a mood lifter. It is a modified approach to the stereo typical Fisher- Yates algorithm used when the user plays shuffle on any music playing device.

### 2. Fisher-Yates Principle

The Fisher- Yates principle is the basic algorithm followed when shuffle is used on any music system generally available. The Fisher-Yates algorithm follow the principle in the following manner:

---

Considering there are  $n$  songs in a list (data structure) and we also have a function called `random(intlowerBound, intupperBound)` which returns a random number within the range provided to it (lowerBound to upperBound-1).

FisherYates(List original, int  $n$ )

1. Create new List final which is initially empty
2. Declare counter = 1 and nos = 0
3. while(counter <=  $n$ )
4. nos = Call random(1,  $n+1$ )
5. Remove the song in position nos from list original and add it to list final.
6. counter = counter + 1
7. end while loop
8. return list final

### **3. Proposed Algorithm**

#### *3.1 Framework*

Firstly, we create a database where we store a variety of songs based on some popular genres some of which are as follows:

Blues; Comedy; Country; Easy Listening; Electronic; Folk; Hip Hop; Jazz; Pop; Rock

Suppose we have  $n$  songs. Normally, if we apply the popular Fisher-Yates Algorithm to select the next song to be played, the algorithm would select a song randomly from all the  $n$  songs we have in all the respective genres. But the aim of our paper is to automatically play songs from the same or similar genre while the user will still have the option to select another song which can be in a completely different genre.

For implementing this, we aim to design an UI in which the songs would be categorized into different genres and presented to the user. The user can manually choose to listen to any song of any genre at any given point of time. Initially, when the user starts playing a song, we record the genre to which the song belongs. When the current song being played ends, we apply Fisher-Yates algorithm on the list of songs present in that genre in which the last song was present. We choose a random song from the list of songs in that particular genre and start playing it.

Additionally, we would deploy another feature which would keep track of the duration for which any song was played.

For instance, while the user is listening to a song, a progress variable would automatically be maintained in the background which would be updated regularly to reflect the duration for which the user listened to

---

a particular song. If the user listened to more than 60% of a particular song in any genre, the next song would automatically be selected from that genre itself.

#### 4. Algorithm

We consider that we have a variable 'progress' which contains the information about the duration for which the last song has been played. We express it as a percentage. For example, if half of the song has been played, that essentially means 50% has been played and the progress variable would have a value 50.

We also consider that we have another variable 'genre' which contains information about the genre of the last song played. This variable points to the list containing the songs from the corresponding genre.

genreList is a list which contains all the genres of songs currently present in the database.

shuffle is a boolean variable which tells whether the playlist needs to be shuffled or not. Each time all the songs from a particular playlist have been played, shuffle becomes true which indicates that a new playlist needs to be started and also each time the user changes a song to a different genre, shuffle changes to true indicating that the playlist again needs to be shuffled to start playing the songs from the new genre.

Selection(int progress, List genre)

1. Start
2. Declare List playList which is initially empty and shuffle = false
3. if(progress > 60 AND shuffle = false)
4.    Declare songNos = 1
5.    while(all songs has not been played from playList)
6.        Play the song at position songNos from playList
7.        Increment songNos by 1
8.        Update progress to reflect the duration for which the song is being played
9.        At any time, if the user wants to go back to the previous song, decrement songNos by 1, reset progress and continue iteration
10.       Else, if the user wants to skip the current song, increment songNos by 1, reset progress and continue iteration.
11.    end while
12.    shuffle = true
13. end if

14. else if(progress > 60 AND shuffle = true)
15.   playList = Call FisherYates(genre, number of songs in list pointed to by genre)
16.   shuffle = false
17. end else if
18. Goto step 4 and start playing the new playList
  
19. else if(progress < 60)
20.   playList = Call FisherYates(Call FisherYates(genreList, number of genres), number of songs in list returned by the call to FisherYates in the first argument)
21.   shuffle = false
22. end else if
23. Goto step 4 and start playing the new playList
24. End

## **5. Applications and future work**

The paper gives a basic algorithm to sort the shuffle option in the user's playlist according to his/her mood and choice. Once the UI has been designed, it can be applied for future use.

1.     It eases the problem of the user when unwanted or unfavorable songs are played. Rather our algorithm will cater to the user's choice.
2.     This algorithm can be used in the future if someday an app is created which can cure depression through music. It will be able to incorporate music therapy according to situation of the user.

## **REFERENCES**

- [1] Hazra, Ghosh, Kumar, Dutta, Chakraborty : 'File Encryption using Fisher-Yates Shuffle',6th International Conference and Workshop on Computing and Communication 2015 sponsored by IEEE
- [2] Singh, Batheja,Chowdhury : 'Predictive Music Shuffling Algorithm' , International Journal of Computer Science and Information Technologies, Vol. 6 (5), 4159 (2015)