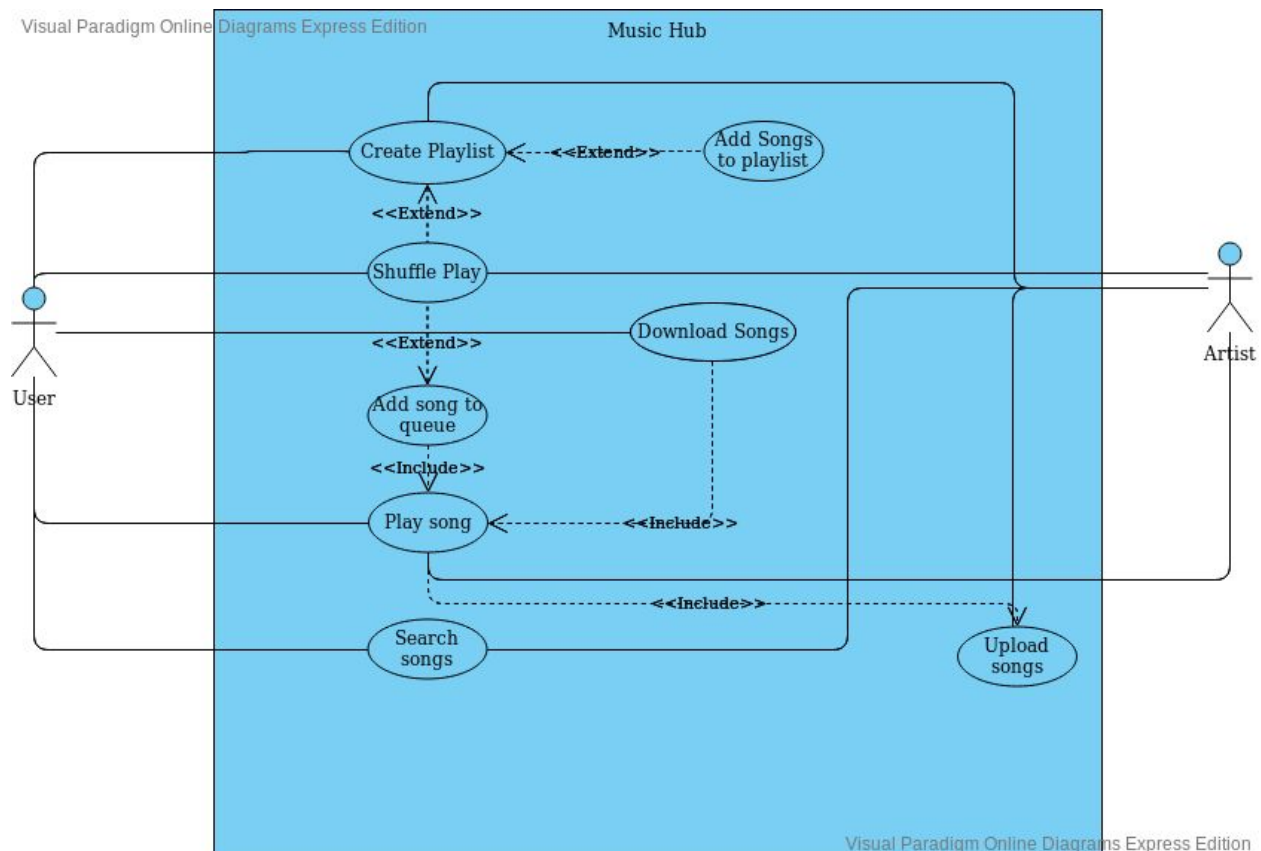


# Music Hub : Project Analysis and Design Document

SWASTI SHREYA MISHRA (IMT2017043)

The document describes the design of a Music App (Music Hub) where the actors are : User and Artist.

## Use Case Diagram



## Use case description

**Use Case ID :** 1

**Use Case Name :** Create playlist

**Use Case Description :** User or an artist can create a playlist. A playlist comprises of multiple songs.

**Actors :** User, Artist

**Flow:**

1. Normal Flow -
  - a. The user/artist clicks on the “Create Playlist” button.
  - b. The user/artist names the created playlist.
  - c. The created playlist is now visible on the home page of Music Hub.
2. Alternative Flow -
  - a. If the user/artist is not logged in, the user/artist first has to login.
  - b. The user/artist names the created playlist.
  - c. The created playlist is now visible on the home page of Music Hub.
3. Exceptional Flow -
  - a. The name of the playlist can only contain alphanumeric characters.
  - b. The name of the playlist shouldn't be empty or blank.

**Use Case ID :** 2

**Use Case Name :** Add songs to playlist

**Use Case Description :** User or an artist can add songs to a playlist.

**Actors :** User, Artist

**Flow:**

1. Normal Flow -
  - a. If the user/artist is not logged in, the user/artist first has to login.
  - b. The user/artist clicks of the “+” button next to the playlist name.
  - c. The user/artist searches the song he/she wants to add to the playlist.
  - d. The user/artist then selects the song from the list.
2. Alternative Flow -
  - a. If the user/artist is not logged in, the user/artist first has to login.

- b. The user/artist searches for a song or opens a song already in the queue.
  - c. The user/artist can long press on the song name and a pop up dialog box appears.
  - d. From the dialog box, the user/artist can choose “Add to playlist”.
  - e. On clicking on “Add to playlist”, the list of existing playlists show up.
  - f. The user/artist chooses one playlist and the song gets added to the corresponding playlist.
3. Exceptional Flow -
- a. If a song is already present in a playlist, it cannot be added to the playlist. That is, duplicate songs cannot exist in a playlist.
  - b. If there are no existing playlists, a song cannot be added to any playlist.

**Use Case ID : 3**

**Use Case Name :** Shuffle play

**Use Case Description :** User or an artist can do “shuffle play” on a playlist and the songs in the playlist will get queued up in a random order.

**Actors :** User, Artist

**Flow:**

1. Normal Flow -
- a. If the user/artist is not logged in, the user/artist first has to login.
  - b. The user/artist selects a playlist from the home page.
  - c. There’s a button of “Shuffle Play” on the left corner of the Playlist View page or the Queue View page.
  - d. On clicking on the “Shuffle Play”, the songs in the playlist or queue will get shuffled in a random order and start playing.
2. Alternative Flow -
- a. If the user/artist is not logged in, the user/artist first has to login.
  - b. The user/artist goes to the queue.
  - c. The user/artist can click on “Shuffle Play” button on the Queue View page.

- d. On clicking on the “Shuffle Play”, the songs in the queue will get shuffled in a random order and start playing.
- 3. Exceptional Flow -
  - a. A playlist should exist to perform “Shuffle Play” on it.
  - b. There should be songs in the queue to perform “Shuffle Play” on it.

**Use Case ID : 4**

**Use Case Name : Play songs**

**Use Case Description : User or an artist can play a selected song.**

**Actors : User, Artist**

**Flow:**

- 1. Normal Flow -
  - a. If the user/artist is not logged in, the user/artist first has to login.
  - b. The user/artist can search a song.
  - c. The user/artist selects a song from the search results.
  - d. The selected song starts playing and gets added to the queue.
- 2. Alternative Flow -
  - a. If the user/artist is not logged in, the user/artist first has to login.
  - b. The user/artist can go to an existing playlist from the home page.
  - c. From the playlist, the user/artist can select a particular song.
  - d. The selected song starts playing and gets added to the queue.
- 3. Exceptional Flow -
  - a. For a song to be selected from a playlist, a playlist should exist.
  - b. While searching for the song, the results should fetch some songs.

**Use Case ID : 5**

**Use Case Name : Download songs**

**Use Case Description : User or an artist can download a selected song.**

**Actors : User, Artist**

**Flow:**

1. Normal Flow -

- a. If the user/artist is not logged in, the user/artist first has to login.
- b. The user/artist can search a song.
- c. The user/artist selects a song from the search results.
- d. The selected song starts getting downloaded and gets added to "Downloads".

2. Alternative Flow -

- a. If the user/artist is not logged in, the user/artist first has to login.
- b. The user/artist can go to an existing playlist from the home page.
- c. From the playlist, the user/artist can select a particular song.
- d. The selected song starts getting downloaded and gets added to "Downloads".

3. Exceptional Flow -

- a. For a song to be downloaded from a playlist, a playlist should exist.
- b. While searching for the song, the results should fetch some songs.

**Use Case ID : 6**

**Use Case Name :** Upload songs

**Use Case Description :** Artist can upload a song to the Music Hub.

**Actors :** Artist

**Flow:**

1. Normal Flow -

- a. If the artist is not logged in, the artist first has to login.
- b. The artist clicks on "Upload song" button on their Profile Page.
- c. The dialog box appears showing the device's folders.
- d. The artist selects a song from the device.
- e. The song gets uploaded and shows on the Profile Page of the artist.

2. Alternative Flow -

- a. A user who is logged in can click on "Verify Account".

- b. After the verification, the “Upload song” button appears on their Profile Page.
  - c. The user is now verified as an artist.
  - d. The artist clicks on “Upload song” button on their Profile Page.
  - e. The dialog box appears showing the device’s folders.
  - f. The artist selects a song from the device.
  - g. The song gets uploaded and shows on the Profile Page of the artist.
3. Exceptional Flow -
- a. A user cannot upload a song unless he/she is a verified artist.
  - b. Only one song can be uploaded at a time.
  - c. The format of the song can only be one of the following formats  
- MP3, MP4, WAV.

**Use Case ID : 7**

**Use Case Name :** Add songs to queue

**Use Case Description :** User or an artist can add songs to his/her queue.

**Actors :** User, Artist

**Flow:**

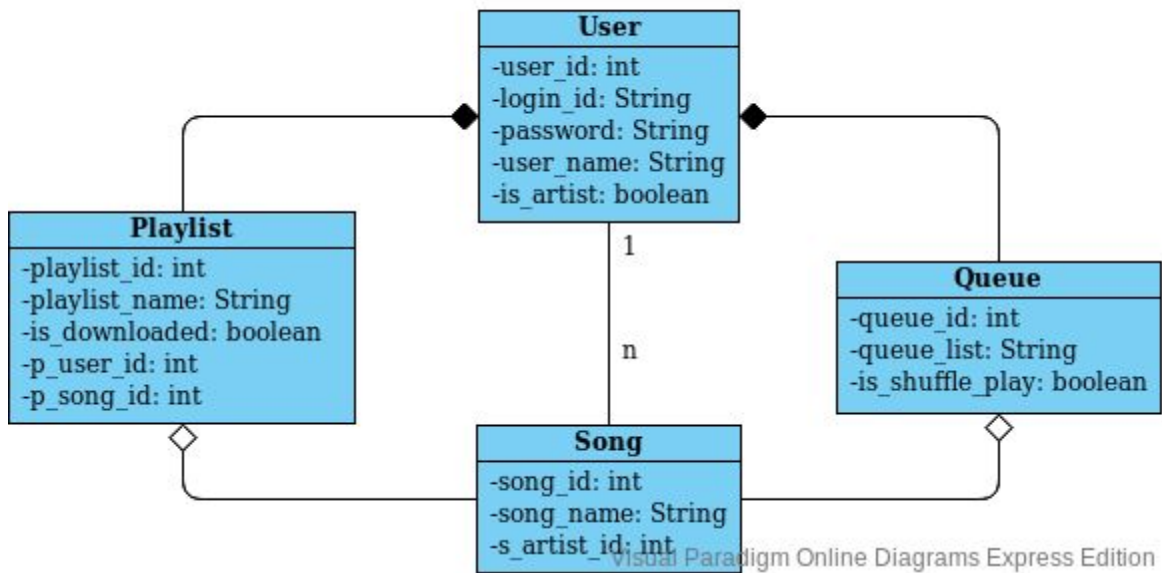
4. Normal Flow -
- a. If the user/artist is not logged in, the user/artist first has to login.
  - b. The user/artist clicks of the “+” button next to the queue.
  - c. The user/artist searches the song he/she wants to add to the queue.
  - d. The user/artist then selects the song from the list.

For ease of implementation, the following functionalities have been implemented:

- Play a song
- Create a playlist
- Download current song
- Add song to the queue
- Upload a song
- Shuffle play the queue

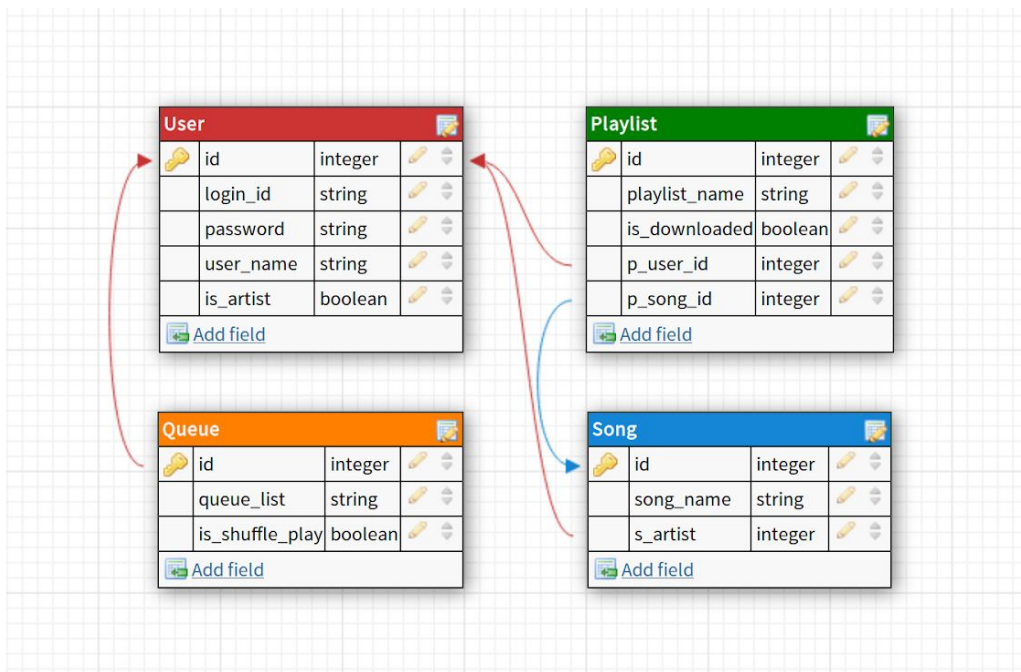
# Class Diagram

Visual Paradigm Online Diagrams Express Edition



Visual Paradigm Online Diagrams Express Edition

# Relational Diagram



## Demo Screenshots:

- Users in the database:

```
mysql> select * from user;
+-----+-----+-----+-----+-----+
| user_id | login_id | password | user_name | is_artist |
+-----+-----+-----+-----+-----+
| 1 | swasti | swasti_pass | swasti | 0 |
| 2 | ananya | ananya_pass | ananya | 1 |
| 3 | lalitha | lalitha_pass | lalitha | 0 |
+-----+-----+-----+-----+-----+
3 rows in set (0.01 sec)
```

- Play song

```
swasti@swasti:~/Work/sem5/DBMS/Lab/Project$ javac *.java
swasti@swasti:~/Work/sem5/DBMS/Lab/Project$ java -classpath "/usr/share/java/mysql-connector-java-8.0.18.jar:." Driver
Connecting to database...
Creating statement...
Enter login id:
swasti
Enter password:
swasti_pass
swasti_pass
Login successful...
Let's play some songs! :) Get started:
1: Play song
2: Create playlist
3: Download current song
4: Add song to queue
5: Upload a song
6: Shuffle play
7: Exit
1
Currently playing: counting_stars
```

- Create a playlist with playlist\_name: pop and first song as: counting\_stars

```
1: Play song
2: Create playlist
3: Download current song
4: Add song to queue
5: Upload a song
6: Shuffle play
7: Exit
2
Enter playlist name:
pop
Enter song name:
counting_stars
1: Play song
2: Create playlist
3: Download current song
4: Add song to queue
5: Upload a song
```

### The playlist table:

```
mysql> select * from playlist;
+-----+-----+-----+-----+-----+
| playlist_id | playlist_name | is_downloaded | p_user_id | p_song_id |
+-----+-----+-----+-----+-----+
| 1 | pop | 0 | 1 | 1 |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```



- Download the current song that is playing for the user:

```
1: Play song
2: Create playlist
3: Download current song
4: Add song to queue
5: Upload a song
6: Shuffle play
7: Exit
3
```

The updated playlist table :

```
mysql> select * from playlist;
+-----+-----+-----+-----+
| playlist_id | playlist_name | is_downloaded | p_user_id | p_song_id |
+-----+-----+-----+-----+
| 1 | pop | 1 | 1 | 1 |
+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

- Add song with song\_name: aakhri\_salaam to the queue for the current user:

Queue table before adding the song

```
mysql> select * from queue;
+-----+-----+-----+
| queue_id | queue_list | is_shuffle_play |
+-----+-----+-----+
| 1 | counting_stars,humnava_mere | 0 |
| 2 | | 0 |
| 3 | sweet_but_psycho | 0 |
+-----+-----+-----+
3 rows in set (0.00 sec)
```

```
1: Play song
2: Create playlist
3: Download current song
4: Add song to queue
5: Upload a song
6: Shuffle play
7: Exit
4
Enter song name:
aakhri_salaam
```

Queue table after adding the song

```
mysql> select * from queue;
+-----+-----+-----+
| queue_id | queue_list | is_shuffle_play |
+-----+-----+-----+
| 1 | counting_stars,humnava_mere,aakhri_salaam | 0 |
| 2 | | 0 |
| 3 | sweet_but_psycho | 0 |
+-----+-----+-----+
3 rows in set (0.00 sec)
```

- Upload a song (CASE1: user is NOT a verified artist) :

```

1: Play song
2: Create playlist
3: Download current song
4: Add song to queue
5: Upload a song
6: Shuffle play
7: Exit
5
Enter song name:
la_la_land
Hey! You're not a verified artist yet!
1: Play song
2: Create playlist
3: Download current song
4: Add song to queue
5: Upload a song
6: Shuffle play
7: Exit
7
Goodbye!

```

- Upload a song (CASE2: user is a verified artist) :  
Song table before the action:

```

mysql> select * from song;
+-----+-----+-----+
| song_id | song_name      | s_artist_id |
+-----+-----+-----+
| 1       | counting_stars | 2           |
| 2       | humnava_mere   | NULL        |
| 3       | sweet_but_psycho | NULL        |
| 4       | aakhri_salaam  | NULL        |
| 5       | i_dont_care    | NULL        |
+-----+-----+-----+
5 rows in set (0.00 sec)

```

```

Connecting to database...
Creating statement...
Enter login id:
ananya
Enter password:
ananya_pass
ananya_pass
Login successful...
Let's play some songs! :) Get started:
1: Play song
2: Create playlist
3: Download current song
4: Add song to queue
5: Upload a song
6: Shuffle play
7: Exit
5
Enter song name:
dariya
true
1: Play song
2: Create playlist
3: Download current song
4: Add song to queue
5: Upload a song
6: Shuffle play
7: Exit
7
Goodbye!

```

## Song table after the action:

```
mysql> select * from song;
+-----+-----+-----+
| song_id | song_name | s_artist_id |
+-----+-----+-----+
| 1 | counting_stars | 2 |
| 2 | humnava_mere | NULL |
| 3 | sweet_but_psycho | NULL |
| 4 | aakhri_salaam | NULL |
| 5 | i_dont_care | NULL |
| 6 | dariya | 2 |
+-----+-----+-----+
6 rows in set (0.00 sec)
```

- **Shuffle play:**

```
Connecting to database...
Creating statement...
Enter login id:
swasti
Enter password:
swasti_pass
swasti_pass
Login successful...
Let's play some songs! :) Get started:
1: Play song
2: Create playlist
3: Download current song
4: Add song to queue
5: Upload a song
6: Shuffle play
7: Exit
6
```

## Queue table after doing Shuffle Play:

```
mysql> select * from queue;
+-----+-----+-----+
| queue_id | queue_list | is_shuffle_play |
+-----+-----+-----+
| 1 | counting_stars,humnava_mere,aakhri_salaam | 1 |
| 2 | i_dont_care | 0 |
| 3 | sweet_but_psycho | 0 |
+-----+-----+-----+
3 rows in set (0.00 sec)
```