User

PK	user_id int NOT NULL
	user_name char(50) NOT NULL

User Playlist

PK	user_playlist_id int NOT NULL
FK	user_id
FK	playlist_id

Playlist

PK	playlist_id int NOT NULL
	PlayList_Type
	Active Flag

Playlist Detail

PK	playlist_detail_id
FK	playlist_id
FK	song_id
	delete_Flag

Songs

PK	song_id int NOT NULL

UserPlayListService

Long createNewPlaylist(Long userId,List<Songs>songs)

PlayList addToPlayList(playlistId,List<Songs>songs)

PlayList removeFromPlayList(playlistId,List<Songs>songs)

Backend Recommendation Service

Long createRecommendationPlaylist(Long userId,List<Songs>songs)

Tracks UserActivity and hot list new additions and user rankings

Can Use Song Ratings,visits and algorithms to prepare recommendation list

UserDashBoardService

Song searchSongByName(String songName)

Song searchPlayList(String playListName)

PlayList getRecommendedPlayList(Long userId)

Song searchSongByArtist(String artistName)

Song searchSongByGenre(String genre)

Song searchSongByMovie(String movie)

Song searchSongByAlbumName(String movie)

UserService

registerNewUser(User user)

modifyUser(User user)

deleteUser(Long userId)

getUser(Long userId)

Backend Song Service

addSong(Song song)

deleteSong(Long songId)

modifySongInfo(Song song)

Song searchSong(Song song)

Scaling

Apis that need scaling:All

Apis that need throughput:User PlayList Service

Country dependency:Geo cluster required with hotspot India

Algorithms used for Recommendation

collaborative filtering algorithm

Caching of playlist Data

User

User
userId
List<PlayList> playList

PlayList

PlayList
id
name
userCreatedOrRecommended
List<Songs>

UserPlayList

UserPlayList
playListId
userId

Songs

Songs
songId
artist
genre
songName
albumName
movieName

1:n

1:1

1:n