

# 19CSE202 – DATABASE MANAGEMENT SYSTEM

## Phase 1 – Submission

GROUP-11

TEAM :

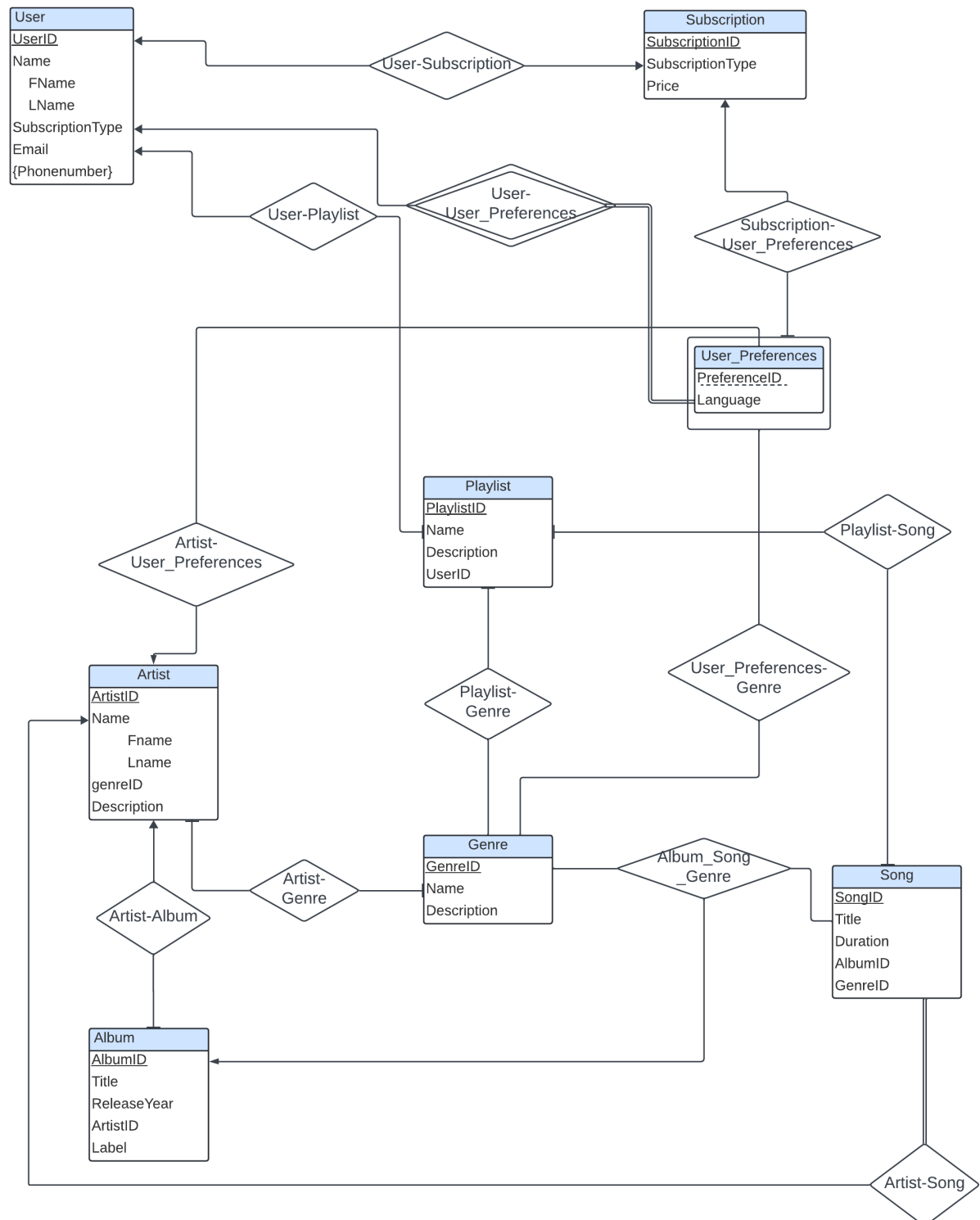
Roll No	Name
AM.EN.U4CSE22151	Sunkari Ravindra
AM.EN.U4CSE22124	G.Umesh
AM.EN.U4CSE22164	Sachin
AM.EN.U4CSE22133	Louis Lara

Title : Music System Database

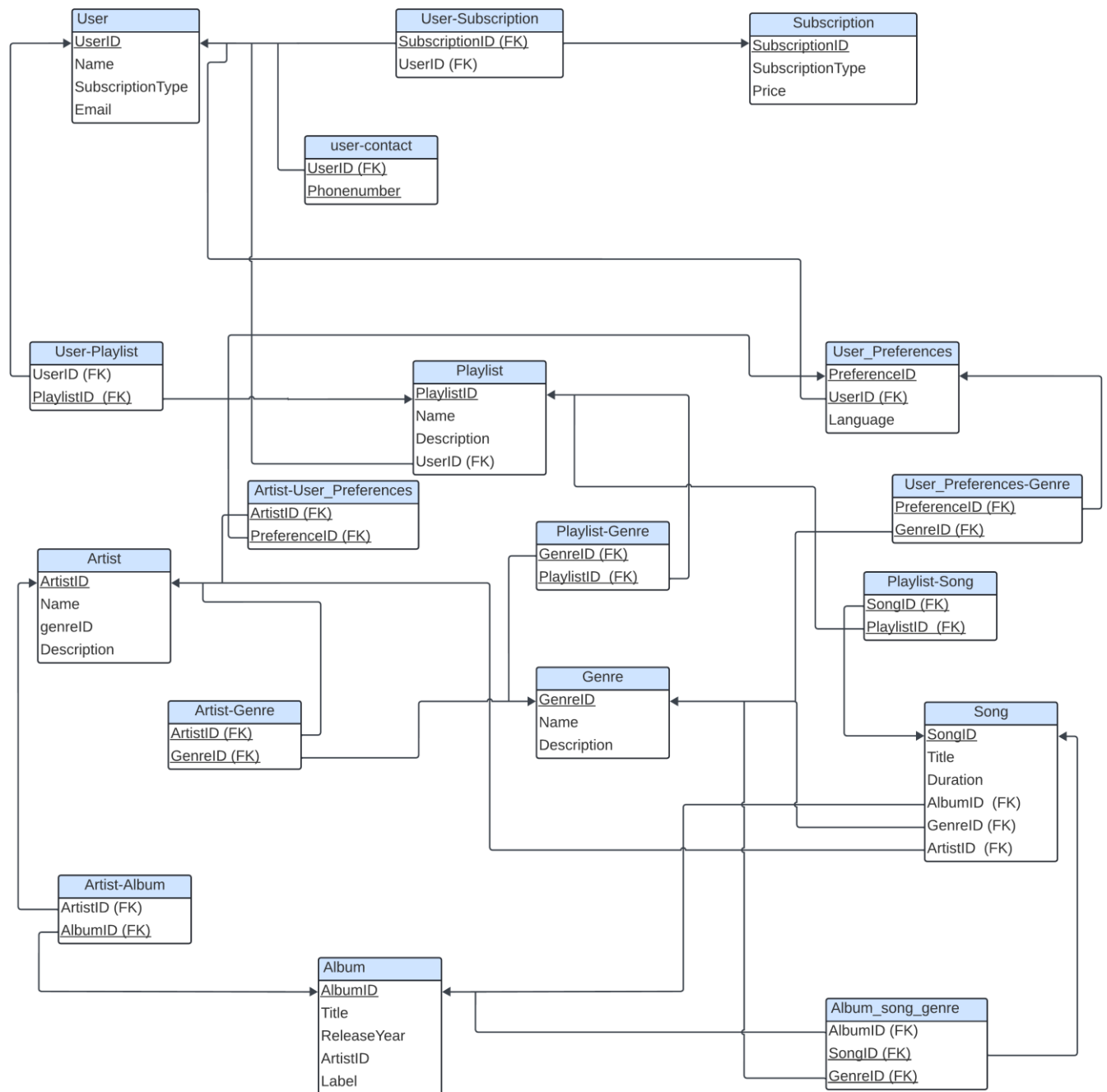
### Description:

The Music Database System's Entity-Relationship (ER) Diagram provides a visual representation of the structural framework underlying our comprehensive music database. Through meticulous design, we have mapped out the interrelationships between entities such as artists, albums, tracks, genres, and users, capturing the intricate web of data crucial for efficient music management. This ER Diagram serves as a blueprint for our database implementation, facilitating seamless organization, retrieval, and analysis of music-related information. With a clear depiction of entity attributes and their associations, our system ensures robust functionality and scalability to meet the diverse needs of music enthusiasts, industry professionals, and researchers alike.

### ER – Diagram :



### Schema Diagram :



# Relational Schema:

**User(userId , name , email , subscriptionType)**

User entity helps us create an account for the user. The user is allowed to enter their name, username, and their information including the type of subscription they want.

**User-Subscription(SubscriptionID (FK),UserID (FK))**

**Subscription(SubscriptionID, SubscriptionType,price)**

Subscription entity shows us the different type of subscriptions that a user is allowed to subscribe to.

**user-Contact(UserID (FK) ,PhoneNumber )**

**User-Playlist(UserID (FK),playlistID (FK))**

**Playlist(PlaylistID,Name,Description,UserID (FK))**

This entity gives us the information of the playlist that the user uses. It gives us the name as well as the type of playlist of the user.

**User\_Preferences(PreferenceID , UserID (FK), Language)**

This entity allows us to identify the user's preference regarding the music they would like to listen to.

**Artist(ArtistID,Name,genreID,Description)**

This entity gives us the artist and the various information of the artist. Like the type of music the artist makes and the genre of their music.

**Artist-User\_Preferences(ArtistID (FK),PreferencesID (FK))**

**User\_Preferences-Genre(PreferencesID (FK),GenreID(FK))**

**Playlist-Genre(GenreID (FK),PlaylistID (FK))**

**Playlist-Song(SongID (FK),PlaylistID (FK))**

**Artist-Genre(ArtistID (FK),GenreID (FK))**

**Genre(GenreID,Name,Description)**

This entity gives us the information of the different type of genres that an artist makes and allows the user to identify the type of music best suited to them

**Song(SongID,Title,Duration,AlbumID (FK),GenreID (FK),ArtistID (FK))**

This entity allows us to identify the song, its duration, its genre and the artist behind the song.

**Artist-Album(ArtistID (FK),AlbumID (FK))**

**Album(AlbumID,Title,ReleaseYear,ArtistID,Label)**

The album entity gives us information about all the songs that are a part of it. It also gives us information about the artist behind the album. It also gives us various other information about the album like its release year and label.

**Album\_song\_genre(AlbumID (FK),SongID (FK),GenreID (FK))**

This is a ternary relationship between album, song and genre. This allows us to understand that the song is a part of the album, and each album has its own genre.