Data and Applications Project Phase - 3

Team **AGDP**

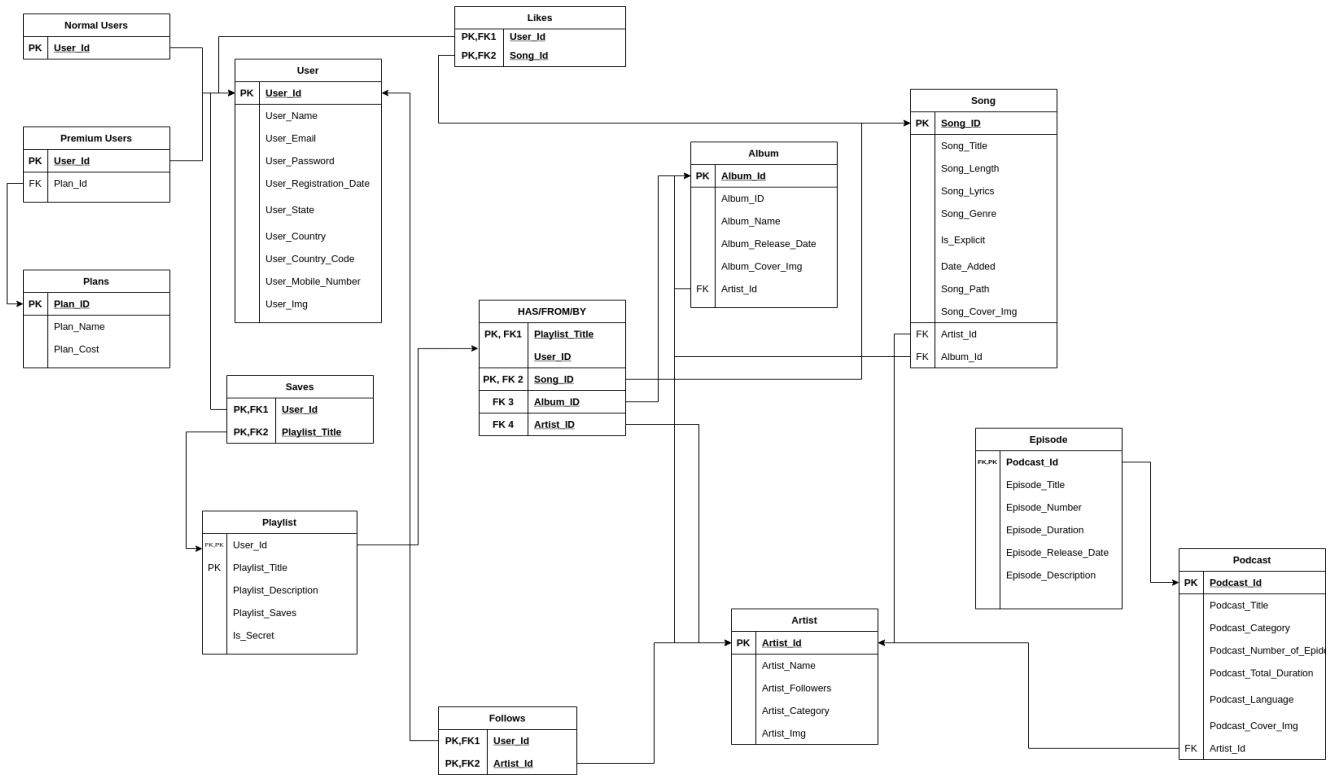
Team Members:
Prerak Srivastava | 2020111013
Parth Maradia | 2020111006
Harshita Gupta | 2020101078

Changes Made

- **Playlist_Title** is a partial key attribute in **Playlist Entity**.

Relational Model Before Normalisation

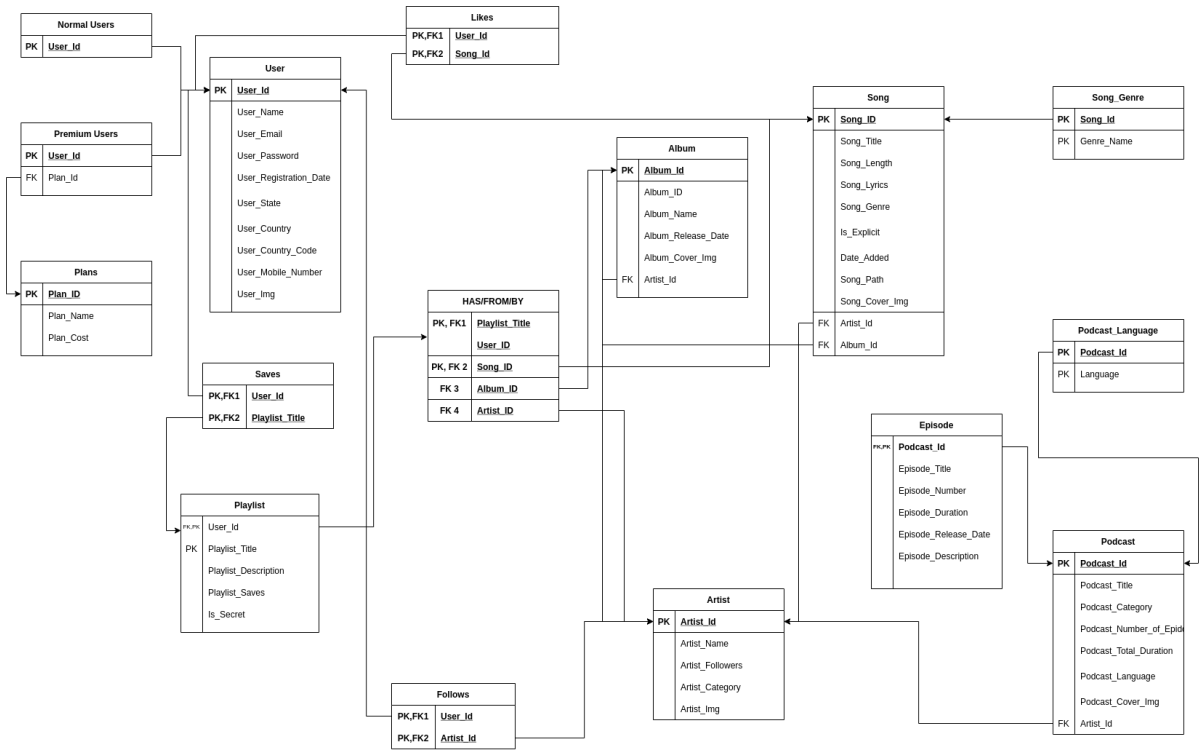
- For each strong entity E in ER model, we created a relation R that includes all the simple attributes in R.
- Primary key of R is one of the key attributes of E.
- For each weak entity W in ER model, with owner entity E, we created a relation R which includes all the simple attributes of W and the primary key attribute of E as the foreign key.
- For mapping one to many binary relationships, we used **foreign key approach** and for many to many binary relations, we created a new relation S which includes the primary keys of the participating relations as foreign keys, and their combination will form the primary key of S.
- For three degree relationships, we used **foreign key approach** and for four degree relationships, we created a new relation S which includes the primary key of participating relations as foreign keys.
- For subclasses of User relation, we made new relations “**Premium Users**” and “**Normal Users**” with primary key as “**User_ID**”, and their respective attributes.


Normal Users
PK User_Id
Premium Users
PK User_Id
FK Plan_Id
Plans
PK Plan_ID
Plan_Name
Plan_Cost
User
PK User_Id
User_Name
User_Email
User_Password
User_Registration_Date
User_State
User_Country
User_Country_Code
User_Mobile_Number
User_Img
Likes
PK,FK1 User_Id
PK,FK2 Song_Id
Song
PK Song_ID
Song_Title
Song_Length
Song_Lyrics
Song_Genre
Is_Explicit
Date_Added
Song_Path
Song_Cover_Img
FK Artist_Id
FK Album_Id
Album
PK Album_Id
Album_ID
Album_Name
Album_Release_Date
Album_Cover_Img
FK Artist_Id
HAS/FROM/BY
PK, FK1 Playlist_Title
User_ID
PK, FK 2 Song_ID
FK 3 Album_ID
FK 4 Artist_ID
Saves
PK,FK1 User_Id
PK,FK2 Playlist_Title
Playlist
FK,PK User_Id
PK Playlist_Title
Playlist_Description
Playlist_Saves
Is_Secret
Follows
PK,FK1 User_Id
PK,FK2 Artist_Id
Artist
PK Artist_Id
Artist_Name
Artist_Followers
Artist_Category
Artist_Img
Episode
FK,PK Podcast_Id
Episode_Title
Episode_Number
Episode_Duration
Episode_Release_Date
Episode_Description
Podcast
PK Podcast_Id
Podcast_Title
Podcast_Category
Podcast_Number_of_Epid
Podcast_Total_Duration
Podcast_Language
Podcast_Cover_Img
FK Artist_Id

Normalisation

1NF

- A relation is in first normal form if it does not contain any **composite** or **multi-valued attribute**. In other words, a relation is in first normal form if every attribute in that relation is single valued attribute.
- We represented multi-valued attributes **Song_Genre** and **Podcast_Language** as separate relations with primary keys as **Song_Id** and **Podcast_Id** respectively.
- For composite attributes **User_Location** and **User_Contact**. Each component of the composite attribute **User_Location, User_Country and User_State** becomes a separate attribute. Similarly for User_Contact.



2NF

- In the second normal form, all non-key attributes are **fully functional dependent** on the primary key.
- **Second Normal Form** applies to relations with composite keys. A relation with a single-attribute primary key is automatically in at least 2NF.
- In our model there is one entity that is Playlist that has a primary key containing more than one attribute (User_Id and Playlist_Title)(partial key attribute). In which all the non-prime attributes are **fully functionally dependent** on every primary key attribute. Hence it is already in the second normal form.
- Thus our relational model as all non-key attributes are **fully functional dependent** on the primary key.

3NF

A relation will be in 3NF if it is in 2NF and does not contain any transitive partial dependency.

- In the **User** entity, the attribute **Country_Code** is dependent on the Country attribute. So we made a new relation **CountryToCountry_Code** with **Country_Code** as the primary key and foreign key.
- In the **User** entity, the attribute **Country** is dependent on the State attribute. So we made a new relation **StateToCountry** with **State** as the primary key and foreign key.
- In the **Song_Lyrics** entity, the attribute **Is_Explicit** is dependent on the Song_Lyrics attribute. So we made a new relation **Is_ExplicitToSong_Lyrics** with **Song_Lyrics** as the primary key and foreign key.