

Ad ID (PK)	Length of Ad	Cost of Ad	Advertiser (FK)	Date Created
<pre>graph LR subgraph Advertiser AID[Advertiser ID (PK)] ADID[Ad ID (FK)] end subgraph Ad ADID2[Ad ID (PK)] AID2[Advertiser ID (FK)] end AID --> ADID2</pre>				

Full

[illegible]

Full

Artist Name (PK)	Total Duration Listened to	Revenue Generated	Email	Password
<div><div></div><div></div></div>	<div><div></div><div></div></div>	<div><div></div><div></div></div>	<div><div></div><div></div></div>	<div><div></div><div></div></div>
	<div><div></div><div></div></div>	<div><div></div><div></div></div>	<div><div></div><div></div></div>	<div><div></div><div></div></div>

Full

Candidate

Event ID (PK)	Artist (FK) (PK)	Date	Time	Location

Full

[illegible]

Tier Name (PK)	Price	Features Included	Duration
<div><div></div><div></div></div>			

Full

Candidate

Features Included needs its own
table? How would we go about this?

[illegible]

Media ID (PK)	Media Name	Total Duration Listened To	Media Ranking	Date Created	Album ID (FK)	Artist (FK)

2.3.1 Normalized To 3NF

Playlist ID (PK)	Description	User/Creator (FK)

Playlists(PlaylistID, Description, User)

FK User references Users(email)

Full

Playlist ID (PK)	Media ID (PK)	Date Added

MediaPlaylists(PlaylistID, MediaID, DateAdded)

FK PlaylistID references Playlists(PlaylistID)

FK MediaID references Media(MediaID)

Full

2.3.2 Opportunities for Combining Relations

Media ID (PK)	Media Name	Total Duration Listened To	Media Ranking	Date Created	Album ID (FK)
<div><div></div><div></div></div>					

Full

Removed Artist since it can be accessed through album*** thus eliminating redundancy. We found no other oppurtunities to reduce redundancy other than FeaturesIncluded attribute in the SubscriptionTiers.