Date:

May 11, 2016

Team Members:

Alec Duong
Joe Urbano

Application URL:

http://cs.drexel.edu:8080/~jau35/browse.html (Open in Google Chrome)

Project Description:

We chose to create a music database to store different types of music-related data. The web front end provides users with various searching functionality as well as the ability to save songs as favorites for his or her own account. We modeled the following entities in our application: Songs, Artists, Writers, Albums, Producers, Music Videos, Venues, Concert Dates, Genres, and Users. Most of the model is centered around songs, which relates to things like artists, writers, albums, genres, music videos, and users. Venues and dates are related to artists to represent concerts and other performances, and producers are related to albums.

Entity sets, Relationship sets, Business Rules:

Songs:

- Songs have a unique ID as the primary key.
- Other attributes include title, duration, and release date.
- Songs are performed by at least one Artist.
- Songs are written by at least one Writer.
- Songs have at least one Genre.
- Songs can have a Music Video.
- Songs can be on an Album.
- Songs can be favorites of a User.

Artists:

- Artists have a unique name as the primary key.
- Other attributes include the country of origin.
- Artists perform as least one song.
- Artists can perform at a Venue on a Concert Date. All records of performances at the same venue are kept.

Writers:

- Writers have a unique name as the primary key.
- Other attributes include the country of origin.
- Writers have written at least one song.

Albums:

- Albums have a unique ID as the primary key.
- Other attributes include a title, release date, and the number of songs.

- Albums have at least one Song.
- Albums can be produced by a Producer.

Producers:

- Producers have a unique name as the primary key.
- Other attributes include the country of origin.
- Producers produced at least one Album.

Music Videos:

- Music Videos have a unique ID as the primary key.
- Other attributes include a release date, record label, and video link.
- Music videos are for exactly one song.

Venues:

- Venues have a unique ID as the primary key.
- Other attributes include a name and location.
- Venues have hosted at least one performance by an Artist on a Concert Date. All records of performances of an Artist at the same Venue are kept.

Concert Dates:

- Concert Dates have a unique date as the primary key.
- Concert Dates have been for at least one performance of an Artist at a Venue. All records of performances of an Artist at the same Venue are kept.

Genres:

- Genres have a unique name as the primary key.
- Genres can have Songs categorized as them.

Users:

- Users have a unique username as the primary key.
- Other attributes include the date joined.
- Users can favorite Songs.

Data Acquisition:

We used accurate information of real songs in our database. All data is loaded into the database through insert statements in our schema file.

Translation of the ER diagram:

We used the techniques from class to translate the ER model to a Relational schema. Most of the entities are represented in their own tables. The only ones that don’t have their own table are the Concert Dates and the Genres. Most of the relationship sets are represented with tables as well. The only relationship set without it’s own table is the relationship linking Songs and Music Videos. In order to represent the constraint that a Music Video is for exactly one song, that relationship is a part of the table for music videos. The other relationship set represented slightly differently is the ternary relationship for artists performing at venues on a date. The artists_performed_at_venues table represents this relationship, and uses the artist name, venue name, and date as the primary key to accurately model the ternary relationship. We created the following tables for our database:

- `music_videos`
- `songs`
- `artists`
- `writers`
- `venues`
- `users`
- `albums`
- `producers`
- `artists_perform_songs`
- `writers_write_songs`
- `artists_performed_at_venues`
- `users_favorite_songs`
- `songs_on_albums`
- `producers_produce_albums`

User Interface:

When you open up the website for the first time, it will prompt the user to enter a username. If you enter a username that was already added to the database, then it will log in that user and display a welcome back message. If you enter a new username, then it will add that username to the database, log in that user, and display a message thanking you for registering. Now that you are logged in (verify that the username is displayed in the upper right corner), there are three main sections.

In the **Search Artists and Songs** section, you can search our database for songs. In the Artists section, use the drop down selection to list all of the songs for the selected artist. In the Search Songs section, enter a string of characters to search for. It will list all songs with a title that contains the string of characters you entered. In the list of songs, click on the song to be directed to its music video. If there is no music video for that song in the database, then it will let you know. Click on the "Add to Favorites" button to add the song to your favorites, and click the "Song info" button to view more info about the song, including the artist, writers, duration, release date, genre, and album.

In the **My Favorites** section, click the "View" button to view your list of favorites. From this list, you can still click a song to be directed to its music video and click on the "Song info" button to see more information about the song. Click on the "Remove" button to remove the song from your list of favorites.

In the **Search Venues for Upcoming Concerts** section, select a venue from the list of venues in our database and click the "Search Upcoming Concerts" button to view which artists will be playing there and when.