Jordan Koeppl

October 4th, 2020

Final Project Proposal

*Project Name*

Bardcore Music

*Overview*

Tabletop Role-playing game (TTRPG) sessions such as “Dungeons and Dragons” involve players imagining themselves in scenarios and worlds beyond everyday life in an almost cinematic way. However, as an avid music listener, the silence in the background during these sessions, especially during scenes in taverns or battle, leaves something to be desired. I would like to make a site where people could collect and upload music from various genres and styles to add as a customizable background playlist for these sessions.

The goal of the site is to create a space in which users can find, upload and edit personalized playlists tailor-made for their TTRPG sessions. Nearly every genre of music would be accepted, as TTRPGs also come in a variety of genres, such as the futuristic noir of “CyberPunk 2020/Red” or the 1980’s suburban-mystery of “Kids on Bikes.” Pages to create would include a **front page**, **about us** and **contact us** information, a **register** and **log-in** user page, a **profile** page with the user-made playlists, along with a more detailed **playlist** page to play the songs from and edit/delete songs as needed, as well as a **search** page of songs so people can find and add other user-uploaded songs to their playlists.

*Tech Stack*

* Windows 10 OS
* Visual Studio 2019
  + ASP.NET Core
  + C# - for web framework
  + HTML5, CSS3, Bootstrap & JavaScript - for front end design
* MSSQL Server Management Studio v 18.5
  + SQL Database
* IIS (if this website decides to utilize a domain)

*Hosting*

Due to the fact that this site would require use of uploading songs that are likely protected by copyright laws, my first choice would be to leave the site as an offline application for personal use and to display in my portfolio. However, a second option I could choose would be that I could also upload the site using a domain, but restrict the library to sources of exclusively copyright-free music. Doing so would likely involve limiting the user upload ability, such as only allowing an administrator to add more songs to the catalog. For the time being, I will proceed along the lines of using an offline model.

*Functionality/Features*

Users of this site would be allowed to do the following:

* Register an account with a login and password.
* Upload songs
* Create and add or delete songs to personal playlists.
* Browse and search for songs uploaded by other users.
* View other profiles and their saved playlists for further song searching.
* Add tracks from searches/browsing to current playlists.
* Loop or auto-play music.

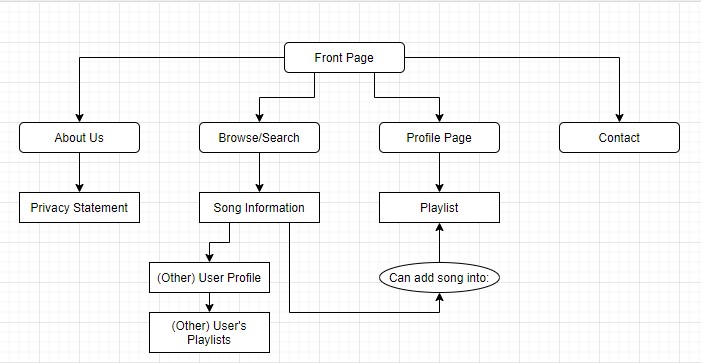
Administrators of this site would have the additional permissions to:

* Moderate and delete songs/audio tracks from other users playlists and the library.
* Delete user accounts.
* Re-edit song information.

*Security & User Roles*

The site would have two user roles: an admin role and a user role. A regular user will be allowed to upload music, save and edit their playlists of music, and delete songs they have uploaded from the system. An admin would have universal permission to edit or delete songs from the browse/search option, and make changes to a user account.

*Site Layout*

**

*Page Templates*

* Front Page
  + Listing of songs most recently added to the site’s library.
* Register & Log-in
  + Form pages to register and/or verify a user’s identity
* About Us, Privacy Policy & Contact Us
  + Populated primarily by text information and hyperlinks
* Profile Page
  + Basic user information along with cards linking to user-made playlists.
* Playlist Page
  + List of audio elements with titles and other song information.
  + Playlist owners have the options to delete or edit.
* Search Page
  + Input a search using text fields and drop-down menus, which will then list search results

*Browser and Device Support*

The plan is to design for both mobile and desktop, as either phones or laptops are common tools used at TTRPG sessions. The format of the elements will be specified using external CSS style sheets and Bootstrap to change based upon screen size.

*Database Design*

Tables would primarily be focused on categorizing music information separated by specific parameters and then linked utilizing foreign keys:

* Song Information
  + TrackID (PK, int, not null)
  + Name (varchar(max), not null)
  + ArtistName (FK\_ArtistID, int, not null)
  + GenreName (FK\_GenreID, int, not null)
  + ReleaseYear (smallint, null)
  + Uploader (FK\_UserID, int, not null)
  + Upload Date (datetime, not null)
* Artist Name
  + ArtistID (PK, int, not null)
  + Name (varchar(200), not null)
* Genre
  + GenreID (PK, int, not null)
  + Name (varchar(200), not null)
* Original Album
  + AlbumID (PK, int, not null)
  + Name (varchar(max), not null)

Additional tables would include:

* User Profile Information
  + UserID (PK, int, not null, identity)
  + First Name (varchar(50), not null)
  + Last Name (varchar(50), not null)
  + Display Name (varchar(50), not null)
* User Playlists
  + PlaylistID (PK, int, not null)
  + Playlist Creator (FK\_UserID, int, not null)
  + Playlist Name (varchar(140), not null)
* User Playlist Track
  + Playlist Track ID (PK, int, not null)
  + PlaylistID (FK\_PlaylistID, int, not null)
  + TrackID (FK\_TrackID, int, not null)