

CoogMusic

Team 8: Koren Stalnaker, Ron Alex, Joshua Nguyen, Santiago Reyes, Sharmake Mohamud

CoogMusic is an online music library, where users (primarily UH students and alumni) can upload tracks and listen to tracks by other artists. Users can choose to browse and listen to music without an account, however users must be registered (as artists) to post songs. The application has a community aspect as well — users can follow artists, “like” uploads, comment on uploads, and create playlists of songs that can be shared with other users.

Entities

Core Entities

Users

Represents a authenticated CoogMusic user with role-based permissions.

Attributes

- `id` - Unique, auto-generated identifier
- `username` - User’s chosen display name
- `email` - User’s email address
- `password_hash` - Encrypted password
- `authenticated_with` - Authentication provider (default: “CoogMusic”)
- `role` - User permission level
- `bio` - User profile bio (optional)
- `profile_picture_url` - Profile image URL
- `created_at` - Account creation timestamp
- `updated_at` - Last profile update

Songs

Represents individual music tracks uploaded by artists.

Attributes

- `id` - Unique, auto-generated identifier
- `title` - Song title
- `album_id` - Associated album (optional)
- `genre` - User-entered genre
- `audio_url` - Audio file storage URL

- `duration` - Song length in seconds
- `image_url` - Cover art URL
- `created_at` - Upload timestamp
- `updated_at` - Last updated timestamp

Albums

Represents albums created by artists.

Attributes

- `id` - Unique, auto-generated identifier
- `title` - Album title
- `image_url` - Album cover art URL
- `created_by` - Artist who created the album
- `created_at` - Creation timestamp
- `updated_at` - Last updated timestamp

Playlists

Represents user-created song collections.

Attributes

- `id` - Unique, auto-generated identifier
- `title` - Playlist title
- `description` - Playlist description
- `created_by` - Playlist creator
- `created_at` - Creation timestamp
- `updated_at` - Last updated timestamp

Relational Entities

User Likes

Tracks user likes on songs.

Attributes

- `user_id` - User that liked the song
- `song_id` - Song the user liked
- `liked_at` - When the song was liked

User History

Tracks user listening history.

Attributes

- `user_id` - User that played the song
- `song_id` - Song the user played
- `played_at` - When the song was played

User Followers

Tracks artist-follower relationships.

Attributes

- `follower_id` - Follower user
- `following_id` - Artist the user follows
- `followed_at` - When the artist was followed

Song Artists

Handles many-to-many relationship between songs and artists.

Attributes

- `song_id` - ID of the song
- `artist_id` - ID of the artist
- `role_at` - Role of song's artist (default: "Main")

Playlist Songs

Associates songs with playlists.

Attributes

- `playlist_id` - Playlist the song is on
- `song_id` - Song that is on a playlist
- `added_at` - When the song was added to the playlist

Song Comments

Stores user comments on songs.

Attributes

- `user_id` - Commenter ID
- `song_id` - Song the user played
- `comment_text` - The text body of the comment
- `commented_at` - When the user commented on the song

User Authentication & Roles

User Roles

- **Unauthenticated User** - Can browse the site, view songs, artist profiles, and playlists.
- **User** - Basic authenticated user. Can browse the site, like songs, comment on songs, follow artists, create playlists, and save tracks and playlists to their library.
- **Artist** - Authenticated user that has registered as an artist. Has all the privileges of the basic authenticated user, but can also upload songs, update and delete their own songs, and upload and create albums.
- **Admin** - Has all the privileges of an authenticated user, but can update and delete songs, albums, and playlists, as well as update and ban users.

Authentication System

- The CoogMusic authentication system implements server authentication with site-based sign-up, and Google OAuth authentication, allowing users to sign up with their Google account.

Primary Authentication

- Email/password with `bcrypt` hashing

OAuth Integration

- Authentication providers, including Google OAuth

Security

- JWT refresh and access tokens and CSRF protection

Session Management

- Token blacklisting for secure logout

Data Entry Forms

Song Upload

- Song metadata (title, genre, album)
- Audio file upload with format validation
- Cover art image upload

Album Creation

- Album information (title, description, release date)
- Cover art upload
- Song selection

Playlist Management

- Playlist metadata (title, description)
- Song addition/removal interface
- Playlist sharing options

User Profiles

- Profile information (bio, profile picture)
- Account settings and preferences
- Authentication method management

Admin Management

- Content moderation tools
- User management interface
- System configuration options

Data Queries

General Site Queries

- Song page (song, artist, album, comments, related songs)
- Album page (songs, artist, related albums)
- Playlist page (profile information, followers/following, uploaded songs)

Statistics/Insights

- Site-wide statistics page
- Artist's 10 ten most played/liked songs
- User's most streamed songs

Data Reports

Monthly Site Analytics

- Total new user registrations
- Song upload statistics by genre
- Most active users and artists

Realtime Analytics

- Trending songs and albums
- Artist growth metrics (followers, plays)
- Genre popularity statistics

Triggers

Update Timestamp

- Automatically update `updated_at` when a record in `songs`, `albums`, or `playlists` is modified.

History Cleanup

- Automatically delete rows from `user_history` older than 30 days whenever the table is updated.

Schema

