**Project Participants:** Evan Flink

**Title:** Music Library Database Management System

**Executive Summary:**

The Music Library Database Management System is a web-based application designed to manage music-related data. The application allows users to create, view, update and delete entities such as songs, artists, and albums. The system also supports one-to-many and many-to-many relationships between these entities, allowing for complex queries to be performed. The goal of the project is to provide a user-friendly interface to manage music data.

**Entities:** User, Artist, Song, Album, Genre

**Initial Features:**

Create a new song (POST on songs)

View all songs (GET on songs)

Update an existing song (PUT on songs/{id})

Delete a song (DELETE on songs/{id})

View all albums (GET on albums)

Create a new artist (POST on artists)

View all artists (GET on artists)

Update an existing artist (PUT on artists/{id})

Delete an artist (DELETE on artists/{id})

View all songs by an artist (GET on artists/{id}/songs)

Create a new album (POST on albums)

View all albums by an artist (GET on artists/{id}/albums)

Update an existing album (PUT on albums/{id})

Delete an album (DELETE on albums/{id})

Add a song to an album (POST on albums/{id}/songs)

Remove a song from an album (DELETE on albums/{id}/songs/{song\_id})

View all songs in an album (GET on albums/{id}/songs)

Search for songs by title or artist (GET on songs?query={search\_term})

**API Endpoints:**

• POST /songs - create a new song

• GET /songs - retrieve all songs

• PUT /songs/{id} - update a song with a specific ID

• DELETE /songs/{id} - delete a song with a specific ID

• POST /artists - create a new artist

• GET /artists - retrieve all artists

• PUT /artists/{id} - update an artist with a specific ID

• DELETE /artists/{id} - delete an artist with a specific ID

• GET /artists/{id}/songs - retrieve all songs by an artist with a specific ID

• POST /albums - create a new album

• GET /albums - retrieve all albums

• PUT /albums/{id} - update an album with a specific ID

• DELETE /albums/{id} - delete an album with a specific ID

• POST /albums/{id}/songs - add a song to an album with a specific ID

• DELETE /albums/{id}/songs/{song\_id} - remove a song from an album with a specific ID

• GET /albums/{id}/songs - retrieve all songs in an album with a specific ID

• GET /songs?query={search\_term} - search for songs by title or artist