

---

# DATA AND APPLICATIONS

---

## Project Phase - 1 MUS3RHQ Label

### **Author**

#### **Team No: 13**

Aadi Deshmukh

Abraham Paul Elenjical

Manasa Kalaimalai

Sumit Kumar

Vidhi Rathore

# 1 Introduction

The database "MUS3RHQ Label" is designed for a recording label agency and encompasses a comprehensive system for managing and storing information related to the recording label and its associated entities. It includes details about artists, albums, tours, employees, departments, investors, and more. The database serves various users, including recording label management, artists, and employees.

It allows for functionalities such as tracking financial performance, artist management, marketing campaign management, and more. Additionally, the database supports various relationships between entities, enabling a wide range of queries and analyses.

# 2 Purpose

The purpose of the database "MUS3RHQ" is to:

- Manage and store information related to a recording label and its associated entities, including artists, albums, tours, employees, departments, investors, and more.
- Conduct Analyses on the data and return specific reports.

# 3 Users of the database

This database has the following users:

- **Recording Label Management:** Executives, directors, and managers of the recording label may use the database to oversee the label's overall operations, track financial performance, and make strategic decisions.
- **Artists:** Artists signed to the recording label can access their own details and track their album releases, tours, and royalties.
- **Employee Management:** HR personnel and department heads use the database to manage employee information, including hiring, salaries, and department-specific data.

# 4 Database Requirements

MUS3RHQ is a **RECORDING LABEL** with a unique recording ID. **RECORDING LABEL** can be the parent of another **RECORDING LABEL**. **RECORDING LABEL** has **DEPARTMENTS**. Each **DEPARTMENT** has an ID, name, ongoing projects, etc., and a department head. The database stores each **ARTIST**'s details such as ID, Name, etc. **ALBUM** is sung by **ARTIST** and contains details such as a tracklist, release date, etc. **ARTIST** goes on **TOUR**, which contains information such as the start date, etc. **ALBUM** has an **ALBUM COVER** which stores information like a photo, cover artists, etc. **RECORDING LABEL** is invested in by **INVESTOR**, which stores details like Investor ID, amount invested, etc.

## 4.1 Strong Entities:

**Note: Attributes in bold represent primary key.**

### 1. Recording Label

- **Attributes**

- **Label ID** (*Type: Integer*) - **Single Attribute**
- Recording Label name (*Type: Varchar*) - Composite Attribute
- Warden name (*Type: Varchar*) - Single Attribute
- Number of employees (*Type: Integer*) - Single Attribute
- Number of Songs (*Type: Integer*) - Single Attribute
- Charts Ranking (*Type: Varchar*) - Multivalued composite attribute
- Distribution Channels (streaming platforms, physical distribution, online retailers) (*Type: Varchar*) - Multivalued composite attribute
- Revenue (*Type: Integer*) - Single Attribute
- Legal Contracts (*Type: Varchar*) - Multivalued composite attribute

### 2. Employees (Superclass)

- **Attributes**

- **Employee ID** (*Type: Integer*) - **Single Attribute**
- Employee Name (*Type: Varchar*) - Single Attribute
- Phone Number (*Type: BIGINT*) - Multi-valued Attribute
- Joining Date (*Type: Date*) - Single Attribute
- Date of birth (*Type: Date*) - Single Attribute
- Salary (*Type: Integer*) - Single Attribute

- **Subclasses**

**Note: Will inherit all the attributes of the superclass, which is Prison Locations**

#### (a) Designers

**Attributes**

- **Designer ID** (*Type: Integer*) - **Single Attribute**
- Number of Covers Designed (*Type: Integer*) - Composite Attribute (Artist who the cover was designed for)
- Total Revenue Generated (*Type: Integer*) - Single-valued Attribute
- Artistic Style (*Type: Varchar*) - Multi-valued Attribute

#### (b) Producers

**Attributes**

- **Producer ID** (*Type: Integer*) - **Single Attribute**
- Artists they handle (*Type: Varchar*) - Composite Attribute

- Production Credits (list of albums/tracks they’ve produced) (*Type: Varchar*) - Composite Multivalued Attribute
- (c) Marketers
  - Attributes**
    - **Marketer ID (*Type: Integer*) - Single Attribute**
    - Artists they handle (*Type: Varchar*) - Composite Attribute
    - Number of Campaigns Involved In (*Type: Integer*) - Single Attribute
    - Advertising Budget (*Type: Integer*) - Single Attribute
- (d) Song Writers
  - Attributes**
    - **Writer ID (*Type: Integer*) - Single Attribute**
    - Song Catalog (songs they’ve written) (*Type: Varchar*) - Composite Attribute
    - Royalties Earned (*Type: Integer*) - Single Attribute
    - Genre Expertise (*Type: Varchar*) - Multi-valued Attribute
- (e) Managers
  - Attributes**
    - **Manager ID (*Type: Integer*) - Single Attribute**
    - Artist Roster (ID’s of artists they’re managing) (*Type: Integer*) - Multi-valued Attribute
    - Tour and Event Management (*Type: Varchar*) - Composite Multi-valued Attribute
- (f) A&R Personnel
  - Attributes**
    - **A&R ID (*Type: Integer*) - Single Attribute**
    - Number of New Artists Signed (*Type: Integer*) - Single Attribute
    - Success Rate of Matching Artist with Producer (*Type: Float*) - Single Attribute
    - Industry Connections (*Type: Varchar*) - Composite Multivalued Attribute

### 3. Department (Superclass)

- **Attributes**
  - **Department ID (*Type: Integer*) - Single Attribute**
  - Number of employees (*Type: Integer*) - Single Attribute
  - Department head (*Type: Varchar*) - Single Attribute
  - Location (*Type: Varchar*) - Single Attribute
  - Budget (*Type: Integer*) - Single Attribute
  - Ongoing Projects (*Type: Varchar*) - Multivalued composite attribute

- Subclasses

**Note: Will inherit all the attributes of the superclass, which is Employee**

- (a) Marketing

- Attributes**

- **Head ID (*Type: Integer*) - Single Attribute**
- Marketing Campaigns (*Type: Varchar*) - Multivalued composite attribute
- Social Media Channels (*Type: Varchar*) - Multivalued composite attribute
- Advertising Partners (*Type: Varchar*) - Multivalued composite attribute
- Artist Promotion Plans (*Type: Varchar*) - Multivalued composite attribute

- (b) Finance

- Attributes**

- **Head ID (*Type: Integer*) - Single Attribute**
- Financial reports (*Type: Varchar*) - Multivalued composite attribute
- Budget allocations (*Type: Varchar*) - Multivalued composite attribute
- Invoice tracking (*Type: Varchar*) - Multivalued composite attribute
- Payroll Data (*Type: Varchar*) - Multivalued composite attribute
- Revenue streams (album sales, streaming royalties, concert revenue) (*Type: Varchar*) - Multivalued composite attribute

#### 4. Artist

- Attributes

- **Artist ID (*Type: Integer*) - Single Attribute**
- Artist Name (*Type: Varchar*) - Single Attribute
- Signing Date (*Type: Date*) - Single Attribute
- Number of Released Albums (*Type: Integer*) - Single Attribute
- Date of Birth (*Type: Date*) - Single Attribute
- Phone Number (*Type: Integer*) - Multi-Valued Attribute

#### 5. Investor

- Attributes

- **Investor ID (*Type: Integer*) - Single Attribute**
- Investor Name (*Type: Varchar*) - Composite Attribute
- Contact Information (*Type: Integer*) - Multi-valued composite Attribute
- Investment Amount (*Type: Integer*) - Single Attribute
- Investment Date (*Type: Date*) - Composite Attribute
- Ownership Stake (*Type: Integer*) - Single Attribute
- Return on Investment (ROI) (*Type: Integer*) - Single Attribute
- Duration of Investment (*Type: Varchar*) - Derived Single Attribute

## 4.2 Weak Entity types

**Note:** Attributes in bold represent partial keys.

### 1. Album

- **Attributes**

- **Album ID** (*Type: Integer*) - **Single Attribute**
- **Track List** (*Type: Varchar*) - **Multivalued Composite Attribute**
- Chart Position (*Type: Varchar*) - Multivalued Composite Attribute
- Duration (minutes) (*Type: Integer*) - Derived Attribute
- Certifications (*Type: Varchar*) - Multivalued Composite Attribute
- Release Date (*Type: Date*) - Single Attribute
- Genre (*Type: Varchar*) - Multivalued Attribute

### 2. Tours

- **Attributes**

- **Tour ID** (*Type: Integer*) - **Single Attribute**
- Tour Name (*Type: Varchar*) - Single Attribute
- Start Date (*Type: Date*) - Composite Attribute
- End Date (*Type: Date*) - Composite Attribute
- Locations (*Type: Varchar*) - Multi-valued Attribute
- Artists Participating in Tour (*Type: Varchar*) - Composite Attribute
- Ticket Sales (*Type: Integer*) - Single Attribute
- Promotional Campaigns (*Type: Varchar*) - Multi-valued Composite Attribute

### 3. Album Cover

- **Attributes**

- **Album Cover ID** (*Type: Varchar*) - **Single Attribute**
- **Image Link** (*Type: Varchar*) - **Composite Attribute**
- Creation Date (*Type: Date*) - Single Attribute
- Album Name (*Type: Varchar*) - Single Attribute
- Cover Artist(s) (*Type: Varchar*) - Multi-valued Attribute

## 5 Relationships

### 5.1 Binary

Recording Label **RELEASES** Albums

- Cardinality Ratio - 1 : N
- Participation: Total - Total

#### **Albums WRITTEN BY Songwriters**

- Cardinality Ratio - N : M
- Participation: Total - Partial

#### **Designers DESIGN Album Cover**

- Cardinality Ratio - N : M
- Participation: Partial - Total

#### **Marketers MANAGE Marketing Department**

- Cardinality Ratio - N : 1
- Participation: Partial - Total

#### **Marketers PROMOTE Albums**

- Cardinality Ratio - N : M
- Participation: Total - Total

#### **AR Personnel SCOUTS Artists**

- Cardinality Ratio - M : N (multiple agents can scout several artists at once)
- Participation: Total - Partial

#### **Artist MANAGED BY Producer**

- Cardinality Ratio - N : 1 (An artist is managed by only one producer, but a producer can manage multiple artists)
- Participation: Total - Partial

#### **Albums SUNG BY Artist**

- Cardinality Ratio - N : 1
- Participation: Total - Partial

#### **Artist is UNDER Recording Label**

- Cardinality Ratio - N : 1
- Participation: Total - Total

#### **Investor INVESTS IN Recording Label**

- Cardinality Ratio - N : M (multiple investors can invest in multiple labels)
- Participation: Partial - Partial

### **Employee WORKS IN Department**

- Cardinality Ratio - N : 1
- Participation: Total - Total

### **Artist GOES ON Tours**

- Cardinality Ratio - 1 : N
- Participation: Partial - Total

## **5.2 Degree > 2 Relationship Types**

### **Recording Label RELEASES Albums SUNG BY Artists**

- Cardinality Ratio - 1 : N : 1
- Participation: Total - Total - Partial

### **Recording Label RELEASES Albums WRITTEN BY Songwriters**

- Cardinality Ratio - 1 : N : M
- Participation: Total - Total - Partial

## **5.3 Greater than 3**

### **Recording label RELEASES albums SUNG BY Artists who GO ON Tours**

- Cardinality Ratio - 1 : N : 1 : M
- Participation: Total - Total - Partial - Partial

## **5.4 Recursive**

### **Recording Label has a Parent-Subsidiary recursive relationship**

- Cardinality Ratio - 1 : M (a label has multiple subsidiaries)
- Participation: Partial - Total (subsidiary has to have a parent recording label, but a parent does not have to have a subsidiary recording label)

# **6 Functional Requirements**

## **6.1 Modifications**

### **1. Insert:**

- **Adding New Song/Album Data:** Incorporating Fresh Songs, Collaborations, and Teasers for Upcoming Albums. Enhance the album database by introducing new content, including songs, collaborations, and teasers as they become available with album releases.



- **Tour Schedule with Date Validity:** Add tour dates to the database, verifying that the dates are valid and do not overlap with other events or tours.
- **Album Cover Design with Compatibility Check:** Add an album cover design record, confirming that it matches the associated album and follows design specifications.

## 2. Update:

- **Chart Position Updates:** Enhancing the Chart Positions of Albums through Updates. Improve the ranking of albums on the charts by updating their chart positions.
- **Marketing Campaign Reallocation:** Reallocate marketing campaign budgets to optimise resources based on performance data and changing marketing priorities.
- **Producer Contract Extension:** Extend the contract of a producer, updating the contract terms and end date while preserving historical data.

## 3. Delete:

- **Managing Artist Transitions:** Deleting Departing Artists: Preserving Database Consistency by Removing Artist Records Upon Departure. Maintain data consistency by removing artist records from the database when they leave the label.
- **Employee Deletion:** Deleting employee records for individuals who have left the company to maintain historical data while ensuring privacy.
- **Album Retrospective Removal:** Delete records of albums that were initially marked for release but were cancelled before they were published, preserving a historical record of album plans.

## 6.2 Retrieval

### 1. Selection:

- **Recent Album Releases by Artists:** Enumerate the artists who have launched an album in the previous year.
- **Discover Artists Under the Guidance of a Specific Producer:** Users have the ability to explore the database and unveil a catalogue of artists under the direction of a particular producer.

### 2. Projection:

- **Searching by Genre:** Allow users to search the database for the names of all artists in a specific genre (e.g., "Retrieve a list of all artists in the 'Rock' genre").
- **Searching tours in a certain Location:** Allow users to specify a certain location, and retrieve the tours planned in the location

### 3. Aggregate:

- **Analysing Artists' Collective Experience:** Calculate the mean (AVG) years of experience among all artists within the database.
- **Exploring Album Prolificacy:** Tabulating the Total Count (SUM) of Albums Released by the Entire Artist Ensemble.
- **Discovering Chart-Topping Hits:** Identifying the Song within an Album with the Highest Chart Rating Achieved by Any Artist (MAX).
- **Spotlighting the Minimalist Artist:** Uncover the Artist with the Least Number of Albums Released (MIN).
- **Determine the Most Controversial Artist:** Assess which artist has the highest number of legal disputes or controversies associated with their work by checking which artist has used the most legal services, providing insight into the artist's level of controversy within the music industry.

#### 4. Search:

- **Artist Name Lookup** - Example: Searching for "Post" to find "Post Malone"
- **Album Name Search** - Example: Searching for "Dream" to find "Dreamland" - Glass Animals"

## 6.3 Functional Analysis

### 6.3.1 Average Album Chart Position of an Artist:

Find the average Chart position of an Artist (across all their albums)

### 6.3.2 Fan Engagement Index:

Develop a metric to assess artist and album success based on fan engagement. (No. of listeners (streams across platforms), No. of albums bought, Average turnout to tours, No. of social media followers).

### 6.3.3 Financial Harmony:

Investigate the correlation between the financial performance of albums and the allocation of marketing budget.

### 6.3.4 Artist-Producer Compatibility:

Assess which artists and producers have the most successful collaborations, resulting in high-charting albums. Returns the pairs of Artist-Producers who have the highest selling albums.

## 7 Bonus Analysis

### Data-Driven Marketing:

While labels use data for marketing, there's room for more sophisticated data-driven marketing strategies. Labels could invest in advanced analytics and AI to better target audiences

and personalise marketing campaigns. From this database we can find out artists with substantial fan followings and a history of controversies tend to receive a larger marketing budget allocation.