# Database Requirements

## Team: DataVerse

(Nora Manolescu, Kaitlyn Clements, Sam Muehlebach, Kyler Luong, Ben Stonestreet)

## Introduction

**Project Overview**

This is a library database that will provide an easy, efficient, user-friendly platform for managing book inventory, tracking checkouts and returns, and administering user access to library resources. The design will be implemented to improve organization and simplify library operations. It will enhance user experience by easily providing information such as book availability, due dates, and user activity. This will be done using MySQL.

**Scope**

Using a MySQL-based relational database, we are able to organize and categorize each book and digital media item by different attributes such as title, author or creator, ISBN, etc. We are able to use it for clients in the database using unique ID, name, contact information, membership type, and account status. We are also using a user-friendly interface for library staff and for clients as well. The UI will consist of searching for items, reserving, for the client side, while for the staff side being able to track transactions between customers and the status for the items.

**Stakeholders:**

Librarians (Administrators): Library staff. Manage book collections, library memberships, checking out books, book returns, and generating records/reports

Library Members (End Users): Users who are members of the library. End users can check out, return, and reserve books, as well as search the libraries catalog of available books

Database Administrator: In charge of managing the database and ensuring security/integrity

# Requirements

## Functional Requirements

- <u>User administration</u>: Add, update, activate, deactivate, and manage user accounts
- <u>Item management</u>: Add new items, update details on an item, track an item's availability, delete items
- <u>Borrowing and Returning</u>: Manage borrowing and returning items, apply borrowing thresholds, and determine late fees
- <u>Reservations</u>: Authorize users to reserve items, notify users when reserved item is available
- <u>Notification System</u>: Send users notifications when a due date is approaching, an item is overdue, or if a reserved item is available
- <u>Report Generation</u>: Create reports such as:
    - List of books, sorted by author or year of publication
    - Membership statuses
    - Outstanding fines by client
    - Item Availability
    - Client Check-Out History
    - Operational Reports concerning overdue items

## Data Entities

- <u>Book</u>
    - Title (char)
    - Author (char)
    - ISBN (char, unique)
    - Year of Publication (year)
    - Genre (char)
- <u>Copy of Book</u>
    - Item ID (unique)
    - ISBN (Foreign Key)
    - Availability (enum: Available, Unavailable)
- <u>Library Members</u>
    - Member ID (int, unique)
    - Name (char)
    - Contact Information (char)
    - Membership Type (char)
    - Account Status (enum: Active, Inactive, Suspended)

- Check-Out Transaction
    - Transaction ID (int, unique)
    - Member ID (int, foreign key)
    - Item ID (int, foreign key)
    - Check-Out Date (datetime)
    - Due Date (datetime)
    - Return Date (datetime, nullable)
    - Fine Amount (decimal, nullable)
- Reservations
    - Reservation ID (int, unique)
    - Member ID (int, foreign key)
    - Item ID (int, foreign key)
    - Date of Reservation (datetime)
    - Status (enum: Active,  Cancelled)

## Hardware and Software Requirements

- Software Requirements:
    - DBMS: MySQL (version 8.0 or higher)
    - OS: Windows 10/11, macOS Ventura (or newer),  Linux (Ubuntu 20.04 LTS or newer)
    - Dev Tools: MySQL Workbench (for database design and management), MySQL Command Line Client (for executing queries), and an IDE (ex: VSCode)
- Hardware Requirements:
    - CPU: Intel Core i3 or AMD Ryzen 3 (or better)
    - RAM: 8GB (16GB recommended)
    - Storage: 10GB of free disk space
    - Network: Stable Internet Connection

## Appendices

N/A