Project Overview: Our project aims to create a database system for a library. This will allow us to track and loan an assortment of books, magazines and digital media. This system also allows us to manage users, apply borrowing rules and generate reports on total fees, client activity, inventory, and overdue items.

* Scope: Our project covers the design, implementation, and management of a relational database system for a mock local library. This allows us to manage a diverse collection of loanable items, track various types of memberships, enforce borrowing rules, and generate meaningful reports. This project does not cover communication with other library databases, instead focusing on its own processes.
* Glossary:
  + ER Model: Graphical representation of the database that uses entities that consist of describing attributes and relations between those entities to plan how data will be organized.
  + SQL: (Structured Query Language): Programming language that allows users to communicate with relational databases to store, retrieve, and manipulate data.
  + ISBN: (International Standard Book Number): An ISBN is a unique book identifier that is assigned to each separate edition/variation of a published book and is consistent internationally.
  + Client:The clients are the people with a membership to the library, they are able to check out books, magazines, and digital media for a period of time for free.
  + Digital Media: For our library digital media means non-physical copies of magazines or books.
  + Book: The books will be physical with each having its own unique ISBN and will be able to be checked out by clients with the title.
  + Magazine: These will be physical items that have their own unique title and issue number to be checked out by clients.
  + Fee: The fees will have an amount stored in the fee entity that is tied to a client entity. The amount is determined by checkout date and return date (if applicable) from any of the items available for checkout which is tied to the client entity.
  + Borrowed: This will have the check out date of the item that the client checks out attached to the relationship.

Review Requirements:

For our database since we will need an entity for each item we are allowing people to check out this includes books, digital media and magazines. Additionally we will need a client entity to store which client has which book, which will be connected by a borrow relationship for each respective item type (item being anything that can be checked out). Fees belong to clients and are given an amount determined by the clients membership type. To account for multiple copies of the same book, where ISBN wouldn’t distinguish two of the same books, a Book\_copy entity exists to keep track of all of the copies that exist of the same book within our system, which maintains the integrity of our database.

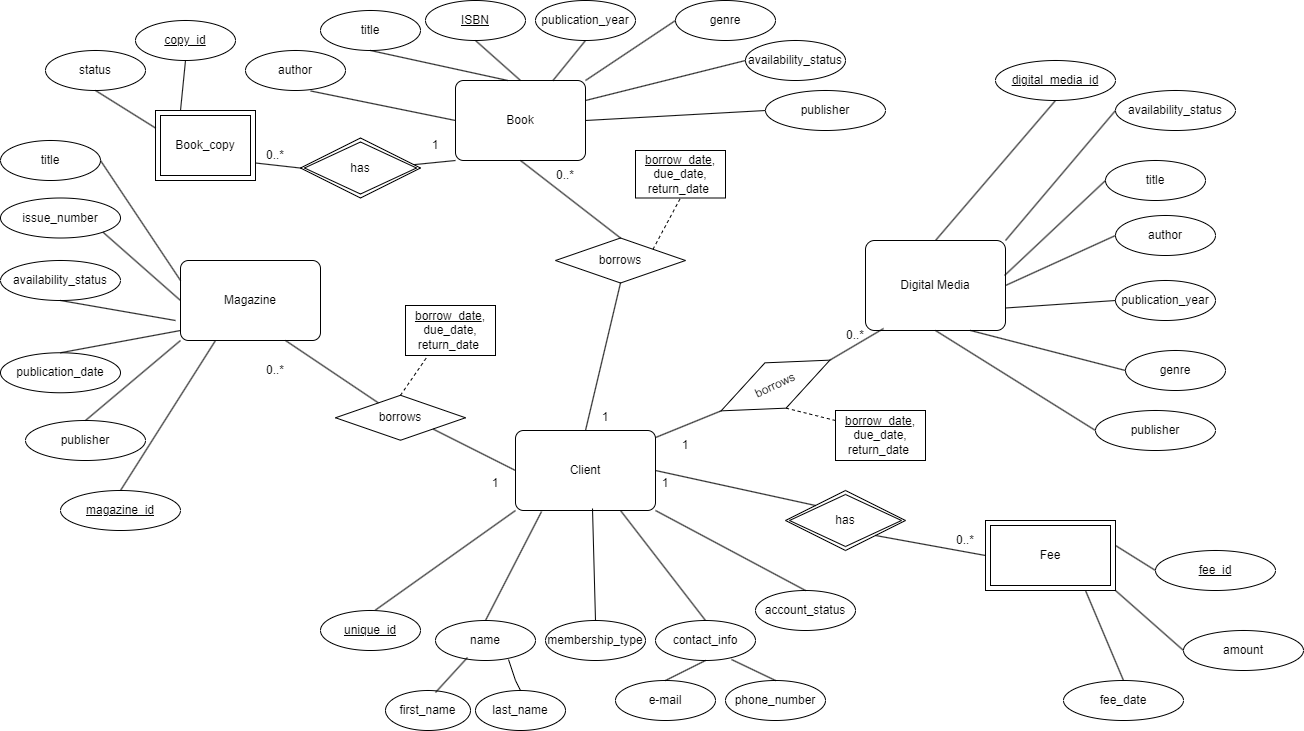
Entities/Data types/Constraints:

* Book: {title (varchar), author (varchar), ISBN (varchar), publication\_year (int), genre (varchar), availability\_status (bool), publisher (varchar}
* Digital Media: {digital\_media\_id (int), title (varchar), author (varchar), publisher (varchar), publication\_year (int), genre (varchar), availability\_status (varchar)}
* Magazine: {magazine\_id (int), issue\_number (int), title (varchar), publisher (varchar), publication\_date (date), availability\_status (varchar)}
* Client: {unique\_id (int), name (varchar), contact\_info (varchar), membership\_type (varchar), account\_status (varchar)}
* Fee: {fee\_id (int), amount (decimal), fee\_date (date)}
* Book\_copy: {copy\_id (int), status (varchar)}

Relationships:

* A client (1) borrows books (0..\*)
* A client (1) can borrow digital media (0..\*)
* A client (1) borrows magazines (0..\*)
* A client (1) can receive 0 or more fees on overdue books, magazines, and digital media.
* A Book (1) can have 0 or more book copies

ER Diagram:



Appendices:

* There is currently no addition needed here, but it will be updated as required.

Github Repository Management:

* A new path was added for the Conceptual Modeling to easily separate this document from previous documents
* The file was also uploaded into the documents of the repository.