# Part 3: Conceptual Modeling

## Project Overview

- Purpose
  - This database is designed to replace outdated, handwritten record-keeping systems for tracking library assets. By automating the process, librarians and staff can reduce the time spent on administrative tasks and focus more on assisting students. It also minimizes human errors, ensures accurate record-keeping, and improves overall efficiency in managing library materials.

- Intended Use
  - The database serves as an automated system for tracking library assets, making it easier to manage inventory, lending, and returns. It will be used by students, teachers, librarians, school administrators, and the IT department. Students will primarily use it to search the catalog, check out books, and track their borrowing history. Librarians and administrators will oversee inventory, manage book conditions, and handle late fees or lost book charges. By streamlining these processes, the database ensures smoother library operations and better access to educational resources.

## Project Scope

The School Library Database will serve as a centralized system for efficiently managing library resources, user accounts, and borrowing activities. The system will track books, their availability, and condition while maintaining comprehensive borrowing records. It will support core library functions such as inventory management, book reservations, loan processing, and overdue fee calculations. The database will also enable librarians and administrators to monitor user activity, enforce borrowing policies. The Entity-Relationship (ER) Diagram will illustrate key entities, relationships, and constraints, providing a visual representation of how data is structured and interconnected within the system. This ensures seamless operations while maintaining scalability and accuracy in tracking library assets and user interactions.

## Glossary

- **ISBN:** (International Standard Book Number) is a unique identifier assigned to books for tracking and cataloging purposes.
- **Medium:** The format in which a library item is available, such as DVD, VHS, Blu-ray, audiobook, eBook, or print.
- **Unix Timestamp:** A numerical representation of time, counting the number of seconds that have elapsed since January 1, 1970 (UTC). It is commonly used in databases and programming for date/time storage and calculations.

- **VARCHAR:** (Variable Character) A data type in SQL that stores variable-length text strings. It allows storing strings with a defined maximum length, making it more efficient than fixed-length types like char

### ER Modeling Components

**Items** - item_id **[INT, PRIMARY KEY]**, title**[VARCHAR(255)]**, price**[DECIMAL(8,2)]**, condition**[VARCHAR(255)]**

Defines an Item which can either be of three subtypes: Book, Digital Media, or Magazines.

**Books** - item_id**[INT, PRIMARY KEY, FOREIGN KEY REFERENCES Items(Item_id)]**, author_id**[INT, FOREIGN KEY REFERENCES Authors(author_id)]**, isbn[**VARCHAR(13)**], genre[**VARCHAR(255)**], publisher_name[**VARCHAR(255)**], edition[**VARCHAR(255)**], publication_date[**DATE**], faculty_only[**BOOLEAN**]

A book is a special type of item. If there are two entries with the same item_id, one in Items and one in Books, it is a book.

For every book, there is always one author.

**Digital Media** - item_id**[INT, PRIMARY KEY, FOREIGN KEY REFERENCES Items(Item_id)]**, medium**[VARCHAR(255)]**, publication_date**[DATE]**, publisher_name**[VARCHAR(255)]**

Digital media is a special type of item. If there are two entries with the same item_id, one in Items and one in Digital Media, it is digital media.

**Magazines** - item_id**[INT, PRIMARY KEY, FOREIGN KEY REFERENCES Items(Item_id)]**, publisher_name**[VARCHAR(255)]**, issue_number**[INT]**

A magazine is a special type of item. If there are two entries with the same item_id, one in Items and one in Magazines, it is a magazine.

**Authors** - author_id**[INT, PRIMARY KEY]**, first_name**[VARCHAR(255)]**, middle_name**[VARCHAR(255)]**, last_name**[VARCHAR(255)]**

For every author there is at least one book.

**Users** - user_id**[INT, PRIMARY KEY]**, email**[VARCHAR(255)]**, address**[VARCHAR(255)]**, phone**[VARCHAR(10)]**, first_name**[VARCHAR(255)]**, last_name**[VARCHAR(255)]**, middle_name**[VARCHAR(255)]**

Defines a user which can be of two subtypes: Student or Faculty.

**Students** - user_id**[INT, PRIMARY_KEY, FOREIGN KEY REFERENCES Users(user_id)]**

A student is a special type of user. If there is an entry in Users and Students with the same user_id, it is a student.

All students may have any amount of fees applied towards them, but not all students have fees.

All students may reserve or loan something, but they do not have to loan or reserve anything.

**Faculty** - user_id**[INT, PRIMARY_KEY, FOREIGN KEY REFERENCES Users(user_id)]**, role**[VARCHAR(255)]**

Faculty members are a special type of user. If there is an entry in Users and Faculty with the same user_id, it is a Faculty member.

All faculty may have any amount of fees applied towards them, but not all faculty have fees. All faculty may apply fees to any user, but not all faculty must apply fees.

All faculty may reserve or loan something, but they do not have to loan or reserve anything.

**Fees - fee_id[INT,PRIMARY_KEY]**, **amount[DECIMAL(8,2)**], fee_applier_user_id**[INT, FOREIGN KEY REFERENCES Users(user_id)]**, fee_acceptor_user_id**[INT, FOREIGN KEY REFERENCES Users(user_id)]**, reason**[VARCHAR(255)]**

Defines a fee that is applied by a given faculty user id: fee_applier_user_id, and received by a given user id: fee_acceptor_user_id.

**Reserves** - reserve_id**[INT PRIMARY_KEY]**, item_id**[INT, FOREIGN KEY REFERENCES Items(item_id)]**, user_id**[INT FOREIGN KEY REFERENCES Users(user_id)]**, reserve_unix_timestamp**[TIMESTAMP]**

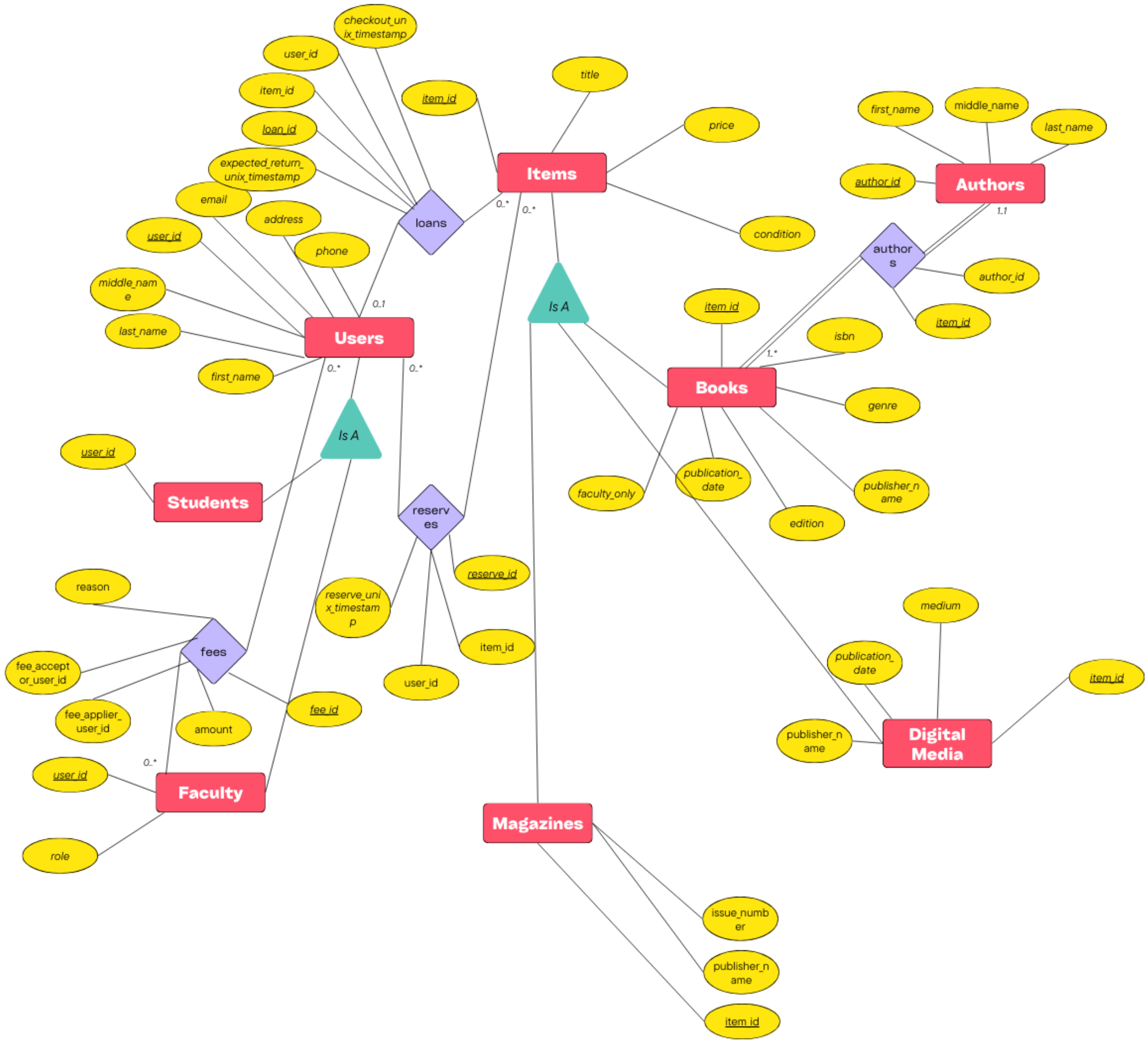
Defines a reservation for a book. The earliest reservation made on a book will be awarded that book upon receiving it back. The reservation will be deleted when the person who made the reservation receives the book.

**Loans** - loan_id**[INT PRIMARY KEY]**, item_id**[INT, FOREIGN KEY REFERENCES Items(Item_id)]**, user_id**[INT FOREIGN KEY REFERENCES Users(user_id)]**, checkout_unix_timestamp**[TIMESTAMP]**, expected_return_unix_timestamp**[TIMESTAMP]**

Defines a loan for a book. There may not be multiple loans for the same book at a time. Loans are deleted from the system once a book is returned, thus, if the current timestamp is greater than expected_return_unix_timestamp, a book is late.

## ER Model

*see github for higher resolution image.*


checkout_unix_timestamp
user_id
item_id
loan_id
expected_return_unix_timestamp
item_id
title
price
Items
loans
0..*
0..*
email
address
phone
user_id
middle_name
last_name
first_name
Users
0..1
0..*
0..*
condition
first_name
middle_name
last_name
author_id
Authors
1..1
authors
author_id
item_id
Is A
item_id
isbn
1..*
Books
genre
faculty_only
publication_date
publisher_name
edition
user_id
Students
Is A
reserves
reserve_id
reserve_unix_timestamp
item_id
user_id
reason
fees
fee_acceptor_user_id
fee_applier_user_id
amount
fee_id
0..*
user_id
Faculty
role
medium
publication_date
item_id
publisher_name
Digital Media
Magazines
issue_number
publisher_name
item_id