# I. Introduction

- Purpose

The school library currently operates without a digital system for managing resources, relying on a manual check-in and check-out process. This has led to inefficiencies, errors, and difficulties in tracking the availability of resources. To address these challenges, our group initiated a database management system project for the library. The project's primary objective is to improve the efficiency and effectiveness of the library's operations, with specific goals that include automating resource management, tracking resource availability, and ensuring data security. By achieving these goals, the system will enable the library to operate more efficiently and effectively, improving the overall experience for library staff and students.

# II. System Overview

- Description of the system's features

The library database management system is a comprehensive software application that provides a range of features designed to automate and streamline key library management processes. Here are some key features of the school library database management system:

- Resource Management: The system allows for the efficient tracking and organization of resources within the library's collection. This includes the creation and maintenance of a comprehensive catalog of resources, with fields for essential information such as author, title, category, and cost of book.

- Student & Staff Management: The system enables the management of borrowing history of students and staff, and overdue materials. This allows library staff to enforce lending policies and ensure that resources are available to all users.

- Availability Tracking: The system allows for the tracking of resource availability, with functionality for checking in and out materials, marking materials as lost or damaged, and generating reports on the status of resources in the collection.

- Security and Data Protection: Using a database management system (DBMS) such as Postgres, the database is kept secure and intact and data is not corrupted. The database is not harmed by physical factors unlike a logbook which can be destroyed by physical means.

## III. Database Design

The database is designed to manage a library system that contains books, students, and staff. The Entity-Relationship (ER) diagram for the database shows that the system has four main entity sets: books, students, staff, and book_issue.

The books entity set has attributes such as the International Standard Book Number (ISBN), title, author, category, price, and number of copies. The primary key of the books table is the ISBN attribute, which is not nullable.

The students entity set contains information about the students who are members of the library, including their student ID, username, email, department, and balance. The primary key of the students table is the student ID attribute.

The staff entity set stores information about the staff members who work in the library, including their staff ID, username, job type, and email. The primary key of the staff table is the staff ID attribute.

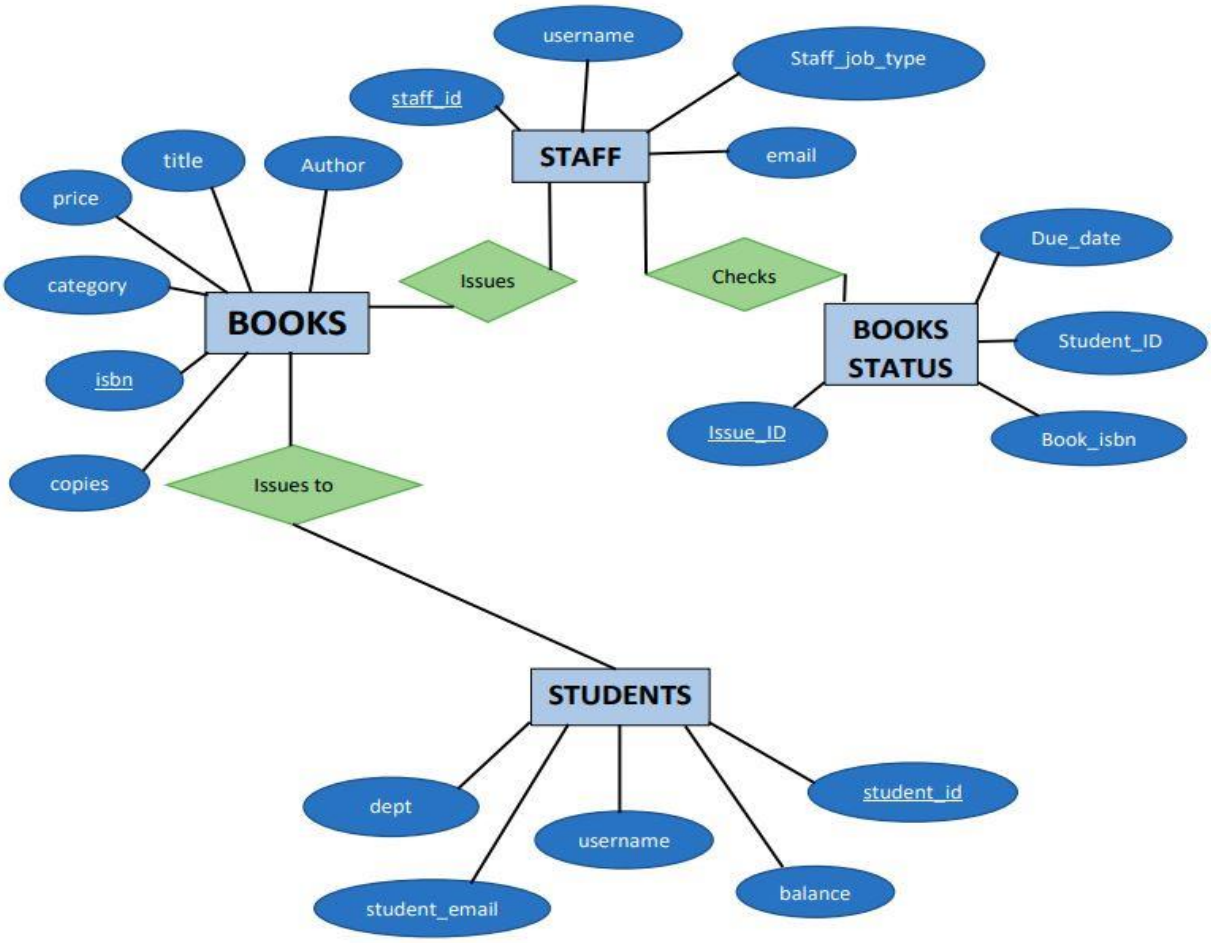
The book_issue entity set records information about books that have been borrowed by students, including the issue ID, the student ID, the ISBN of the book, and the due date. The primary key of the book_issue table is the issue ID attribute.

The ER diagram also shows the relationships between the entity sets. The books entity set has a one-to-many relationship with the book_issue entity set because multiple students can borrow the same book. The book_issue entity set has a one-to-many relationship with the students entity set because a student can borrow multiple books. Finally, the staff entity set has a one-to-many relationship with the book_issue entity set because multiple staff members can be involved in issuing and returning books.

The data model for the database is a Relational Model, which consists of tables that store data for each entity set. The schema of the database includes the CREATE TABLE statements for each entity set and the corresponding attributes. In addition, the schema includes the CREATE TRIGGER statements for the two functions that control how data recorded into and deleted from the book_issue table affects the students and books tables.

Overall, the database schema is designed to provide a comprehensive solution for managing the library system by recording information about books, students, staff, and book borrowing, and it ensures data integrity and consistency by enforcing referential integrity constraints through foreign keys and triggers.

## ER DIAGRAM




IV. Implementation

- Choice of database management system

For the implementation of the school library database management system, PostgreSQL has been chosen as the database management system due to its robustness, scalability, SQL compliance, and security features. PostgreSQL offers strong support for complex data types, built-in indexing, full-text search, and extensibility through custom functions and procedures, making it an ideal choice for managing the diverse data associated with the library's resources and users. Additionally, PostgreSQL has a large and active community of developers and users, providing extensive documentation, tutorials, and support resources to ensure the ongoing evolution and success of the system.

- System architecture and setup

The system architecture and setup for the school library database management system will involve installing and configuring the PostgreSQL database management system. This setup will provide a robust and flexible environment for developing, testing, and refining the system, while ensuring that it meets the needs and requirements of the library staff and users.

- Data population and migration

For data population and testing purposes, mockaroo will be used to generate mock data that will be used to test the functionality and performance of the school library database management system. Mockaroo is a powerful data generation tool that allows for the creation of custom data sets with a wide range of data types and formats. The tool is designed to produce realistic and diverse data that accurately reflects real-world scenarios and use cases.

Mockaroo will be used to generate data for various entities in the library database, including books, authors, category, students, and transactions. The generated data will be imported into the PostgreSQL database using appropriate tools and techniques, ensuring that the data is accurately mapped to the database schema and is consistent with the system's requirements and specifications. The mock data generated by mockaroo will be used to test the system's functionality, performance, and scalability under different scenarios and conditions. The tests will be designed to identify and address any issues or bottlenecks in the system, and to ensure that the system meets the needs and requirements of the library staff and patrons.

Overall, using mockaroo to generate mock data for testing purposes will ensure that the school library database management system is thoroughly tested and refined, with any issues or concerns being addressed before the system is deployed in a production environment.

- Background Operation of Database

When a user (student) borrows a book, the staff (librarian) inserts the school id of the student, isbn of the borrowed book, and the date the book was borrowed. In our case, we set the maximum number of days to return a borrowed book to twenty (20). In our source code, we implemented two (2) trigger functions each with a trigger when data is inserted or deleted from the book issue table. With the first trigger function, issue_book_func(), which is implemented before data is inputted unto the book issue table, it updates the due_date field of book_issue table by adding 20 days to the date inputted. It also updates the balance field of students by adding the price of the book to the initial value of balance. It reduces the number of copies in the books table by one (1) too. With the second trigger function, return_book_func(), which is implemented before data is deleted from the book issue table, it updates the balance field of students by subtracting the price of the book from the initial value of balance. It also increases the number of copies in the books table by one (1) too. These two trigger functions and their triggers effectively manage the database and data within it such that the library resources are managed effectively.

## V. Conclusion

- Summary of the project and its outcomes

The project is a database management system for a school library, which aims to provide an online platform for managing the library's collection, borrowers, debts of users, and other related information. The system is built using PostgreSQL as the database management system and is designed to be accessible. The project objectives include improving the efficiency of the library's operations, providing an accurate record of the library's collection and its usage, and enabling students and staff to easily search for and borrow books from the library. The outcomes of the project include a fully functional database system, the ability to generate reports on the library's collection and usage, and improved management of the library's resources. Additionally, the project provides a foundation for future enhancements to the system, such as integration with other library systems or the implementation of more advanced features like recommendations and user reviews. Overall, the project is expected to have a positive impact on the library's operations and the user experience of the library's users.