Library Management System

By : Maya Chidambaram, Trisha Gundavelli

System Description

Our project is a simple library management system.
Users are organized into students and librarians.
Students can search for various academic files (ie: textbooks).
Librarians can add, edit, and delete files, subjects, publishers, etc.
The files are organized by subject and publisher.
Additionally, users can search based on publication date,
name, etc. after selecting subject and publisher.

Github link -
https://github.com/mayachidambaram/CSE111Proj/tree/main

Implementation Details

- SQLite3 for database
  - To avoid having to set up a server and because it’ll handle smaller sized databases easier
- Python for backend
  - For its simplicity and ease of use for database interaction, logic, and certain library imports.
- Command Line Interface (CLI)
  - The user interacts with the system through a series of command-line prompts and inputs. Lightweight and efficient.
- Visual Studio Code
  - For actual development

Use cases

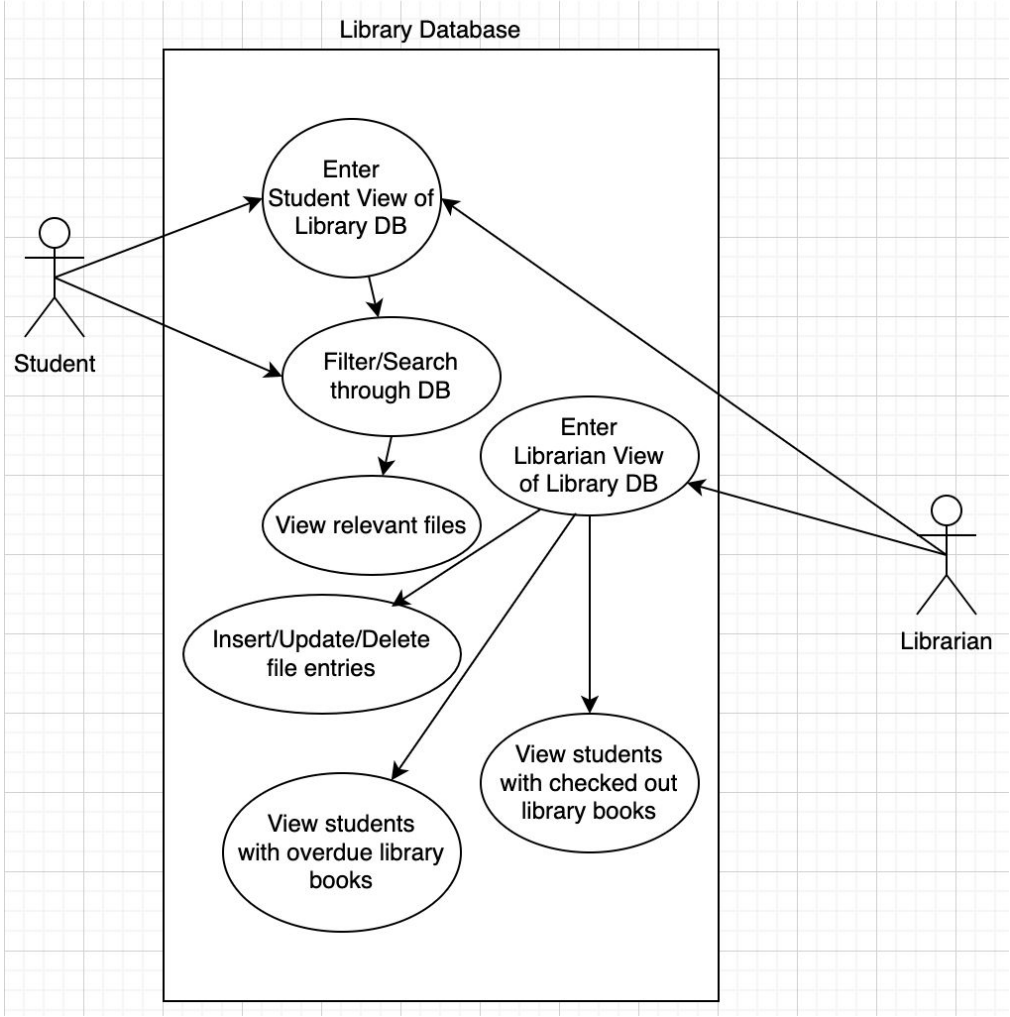
Student:

- Logs into library database
- Filters what kind of file they’re looking for, the publisher, and the subject type, author name, and to view the subject with the most books.

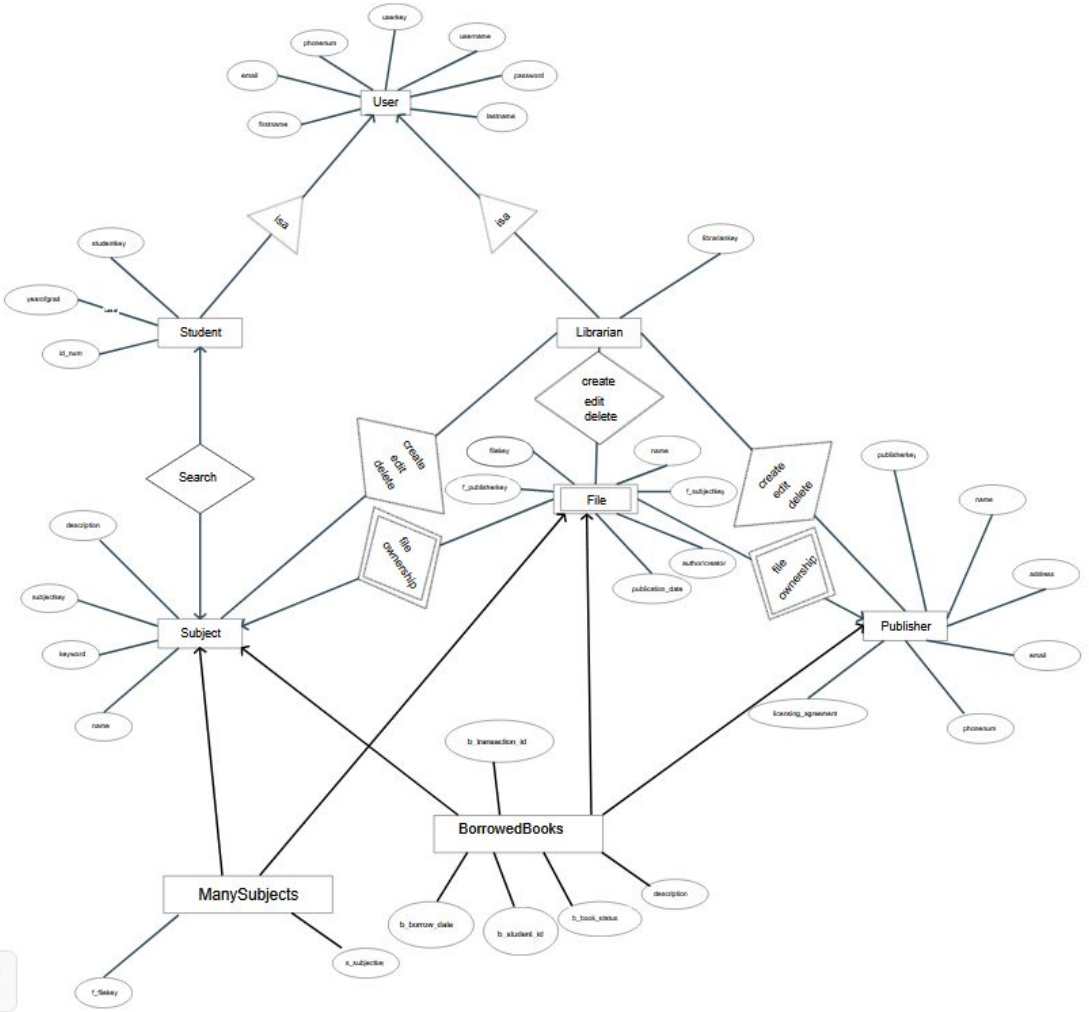
Librarian

- Create new files in the database
- Add/delete existing information in the database
- Look up students with checked out books and who has overdue library books

Use-Case Diagram



ER Diagram


User
Student
Librarian
Search
create
edit
delete
File
Subject
Publisher
BorrowedBooks
ManySubjects
file ownership

Relational Schema

```
CREATE TABLE Files (
        f_filekey INT PRIMARY KEY, f_title VARCHAR(255), f_author VARCHAR(100),, f_publicationYear DATE ,f_publisherkey INT,  f_subjectkey INT, FOREIGN KEY (f_publisherkey) REFERENCES Publisher(p_publisherkey), FOREIGN KEY (f_subjectkey) REFERENCES Subjects(s_subjectkey));
CREATE TABLE User (
        u_userkey INT PRIMARY KEY,u_username VARCHAR(50), u_pass_word VARCHAR(50), u_firstname VARCHAR(50), u_lastname VARCHAR(100), u_email VARCHAR(100), u_phonenum INT);
CREATE TABLE Student(
        s_studentkey INT PRIMARY KEY,s_yearofgrad INT,s_idnumber INT);
CREATE TABLE Librarian(
        l_librariankey INT PRIMARY KEY, l_idnumber INT);
CREATE TABLE ManySubjects (
        f_filekey INT, s_subjectkey INT, PRIMARY KEY (f_filekey, s_subjectkey), FOREIGN KEY (f_filekey) REFERENCES Files(f_filekey), FOREIGN KEY (s_subjectkey) REFERENCES Subjects(s_subjectkey));
CREATE TABLE Publisher(
        p_publisherkey INT PRIMARY KEY, p_publishername VARCHAR(100),p_licensingagreement INT );
CREATE TABLE BorrowedBooks (
        b_transaction_id INT PRIMARY KEY, b_borrow_date DATE, b_student_id INT, b_book_id INT, b_status CHAR, FOREIGN KEY (b_student_id) REFERENCES Student(s_studentkey), FOREIGN KEY (b_book_id) REFERENCES Files(f_filekey));
CREATE TABLE Subjects(
        s_subjectkey INT PRIMARY KEY, s_description VARCHAR(255), s_subjectname VARCHAR(255));
```