CS 3200 Database Design
Project Proposal

PriviteraDRizzoMRizzoN

Project Description

The database system designed by PriviteraRizzoRizzo will serve as a book management system for libraries in a library network, i.e. all public libraries in the City of Boston. This system will allow a library to locate where books are located amongst its branches, and to keep track of what users have checked out what books.

The data being tracked includes each library branch in a network, users who have library cards in the network, employees of each library, books owned by each library, and the location of each book in a particular library.

Members of libraries within a system that uses the application will be able to search for books available at the library, request a book for any library in the system, place a hold for any book in the system, track the due date of a given loaned book, track the availability of a book, and other functionalities of that nature.

The project will allow library users the ability to perform these commands completely via a web page which will display information to them, and allow for inputs as needed.

Storage

The group will be using SQL to store data for the project.

Required Components:

Software

The software will primarily be a Python based backend web application that interfaces with the MySql database.

Apps

At this time, a mobile application is not under development. The group will create a web application for the user to interact with. The web application will facilitate the interface between the user's inputs (like borrowing a book) and updating the data management system - the database.

Languages

The database management system employed for the project is MySQL, as was done for a large portion of the course. The programming language used to interface the database and the application will be Python. This can be referred to as the "backend". The group will consider having the python code execute queries on the database (especially if user input is used by a WHERE clause), using "parameterized queries" with prepared statements; rather than string formatted arbitrary "execute()" commands. The reason for this is the prevention of SQL injections via the Web Application. This process was recommended by various cyber security classes taken by group members.

The Front End User Interface will be implemented in Javascript, CSS, and HTML. Javascript for handling events, form submissions, and determining where to place information retrieved from the backend. HTML will be used to create the structure / layout of the various

web pages. When information is retrieved from the backend using Javascript and calls to Rest API's, it will be placed in a given HTML section (probably a tag). CSS will be used to improve abstraction and modularity of the styling of the webpage.

SQL queries will be employed (through python library calls) to actually interact with the data stored in the database.

Libraries

The python web application will be based on the "Flask" library to enable easy creation of http endpoints that enable users to query the app for information and allow the backend to perform the corresponding sql query.

To interface with the database within the flask app in python the group will use the Flask extension library called "flask-mysql". This library can connect to the database and make use of strings and tuples to construct sql queries to update and select data from the database.

The Javascript front end code will mainly use the Jquery library in order to facilitate HTTP requests to REST API endpoints exposed by the python web application.

Hardware

The backend can be deployed to any machine which can install python and MySql. The Javascript code is run on a client's browser and is supported by all browsers.

Machine Restrictions

This database and application are being designed for Windows. It is possible the backend will be cross compatible with Linux and Mac devices if the python modules and MySql Workbench are installed, but that will not be guaranteed / thoroughly verified as with Windows.

Interest in Data Domain

Each member of this group is an avid reader, with members being on the Executive Board of the Northeastern Book Club. Some of the group members read upwards of 75 books a year and would have spent exorbitant amounts of money to fund this hobby without the use of libraries. Being able to check if the library system has a book and check it out or place a hold has been fortuitous in gaining access to reading material. Over time the sheer frequency of library utilization has made the group members interested in reconstructing and learning about, albeit a smaller scoped version, library system applications and databases.

Conceptual Design

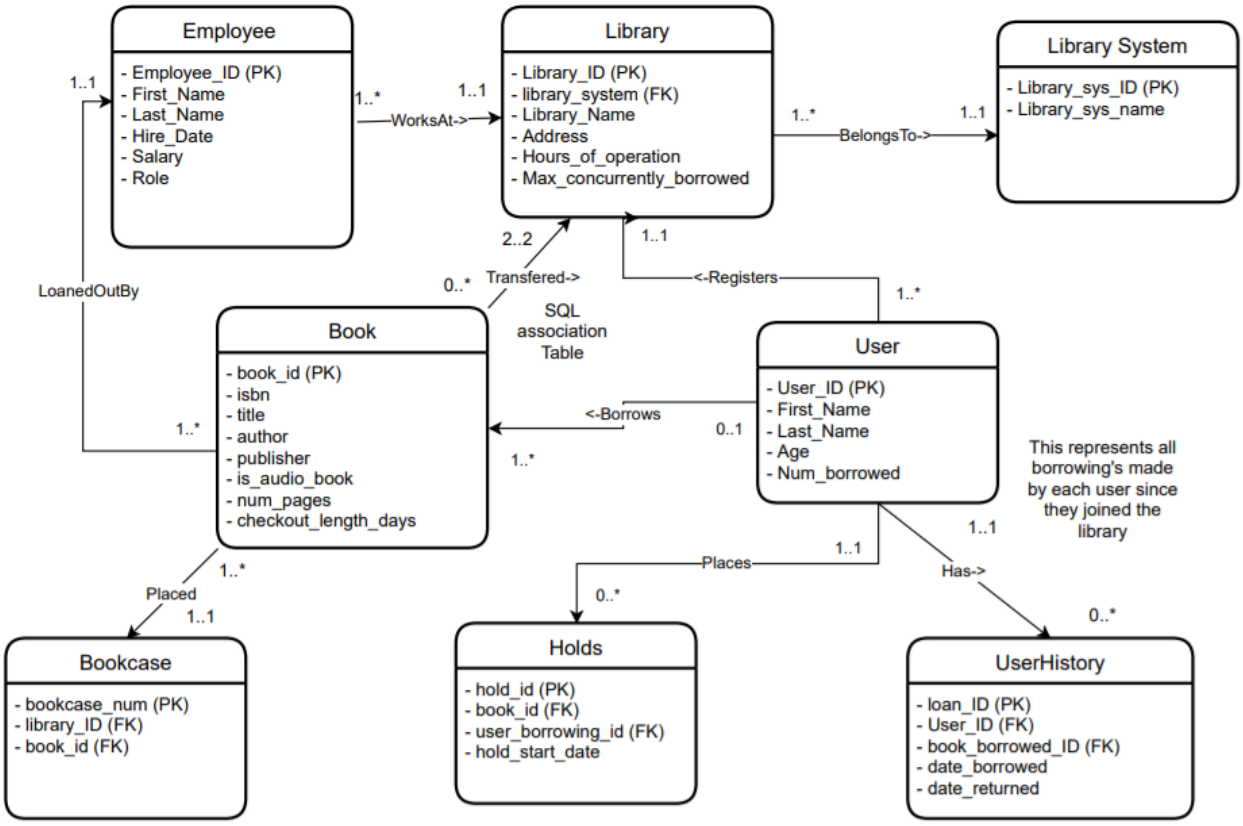

Figure 1: UML diagram of the library management system

User Interactions with the Application

Users will be able to check if any library within the given library system (not just the one they registered at) has copies of any given book. Additionally, the application will display the location (ie the bookcase) of where the book is found in the library. Moreover, users will be able to check out available books or place holds on unavailable books. If a book is at another library within the system, a user can place a request to have the book be transported to their local library, and then borrowed by said user. Users can also see their borrowing history since they registered at a library. Moreover, if a book is listed as an e-book/digital copy, a user can borrow it directly from the webpage, without having to go in person. Future functionality will be a recommendation system to users based on the past books they borrowed.

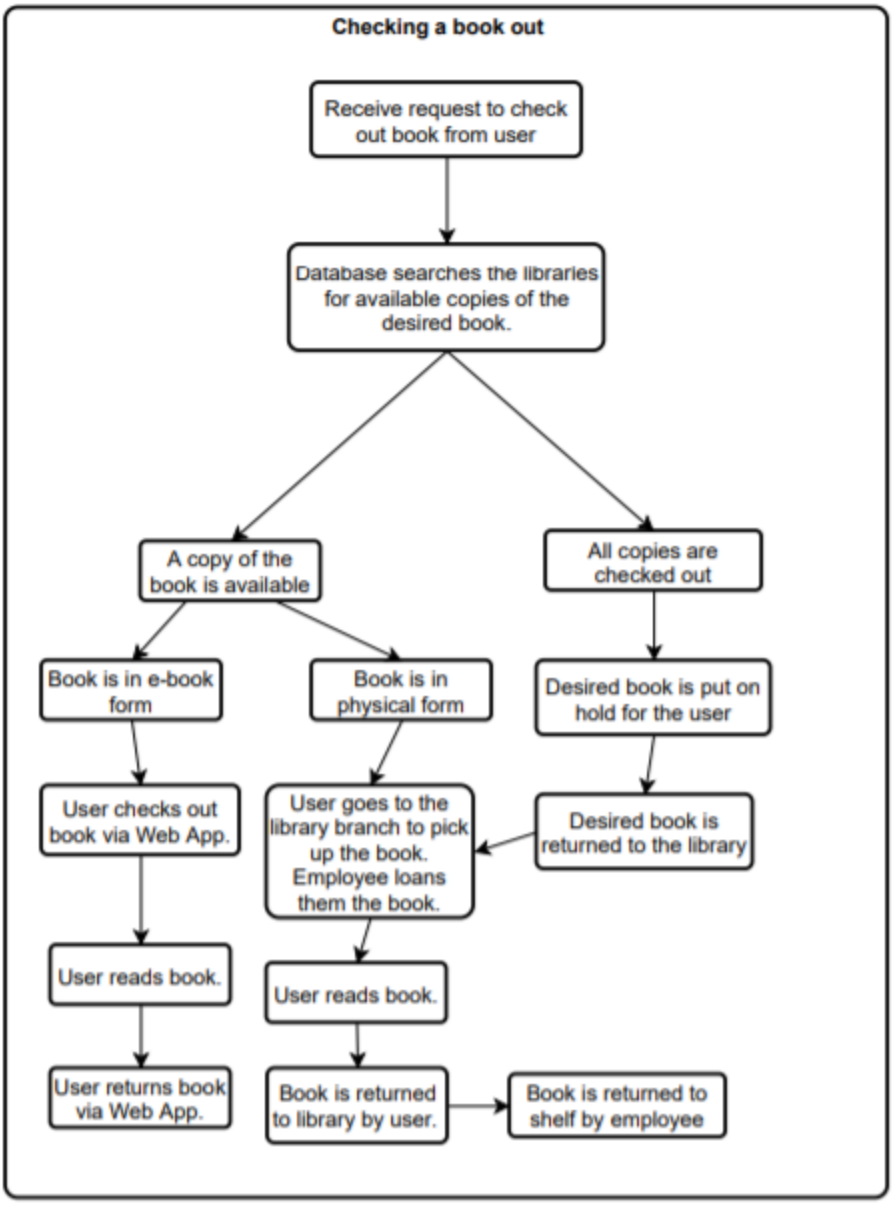

Figure 2: Flowchart of user checking a book out of the library