

## Database Design: My Library Online

### Database selection:

'My Library Online' is a web application which will be used by library users to browse as well as search the resources available in library. This is the minimal viable product features. An extension of the same would also provide more features such as login, place hold on resources as well as request for extension. The user dashboard will show at one glance, all the resources checked out by the user.

For this use case we need to maintain various lists of similar data such as resource lists, user list, membership list, checked out resource list, category list, etc. These lists will have various attributes to define the data that they encompass and each list (i.e. entity) would have relationship with other lists. The list mentioned would translate to tables and the attributes would form the columns for the respective table. Each data record becomes a new row in the table.

Considering the data involved, I believe that this project is a perfect use case for a **relational database (such as PostgreSQL)**. The primary benefit of the relational database approach is the ability to create meaningful information by joining the tables, which helps us understand the relationship between data.

SQL includes the ability to count, add, group, and combine queries. SQL can perform basic math and subtotal functions and logical transformations as well as order the results by date, name, or any column.

Also, PostgreSQL is open source. It provides enterprise features such as security, scalability as well as supports stored procedures, which is a more complex programming language built on top of SQL.

### Tables to be created

Following table will need to be created to store and represent all the data elements and operations in the web application.

- 1) Resource table: This table will record a unique resourceId which can be an autogenerated sequence, resource name (title in UI), authorName, description (which can be a text to describe the resource), publicationYear.
- 2) Category table: This table will contain list of all possible categories of resources, the attributes will consist of categoryId (autogenerated sequence) and category name.
- 3) User table: User table will contain data related to all the users of the online library system. It will contain attributes such as userId (autogenerated sequence), firstName, lastName, address, phone, addressProofId, addressProofType
- 4) Membership table: This table will consist of all the library memberships. The attributes for the same would be membershipId (autogenerated sequence), userId (foreign key to user table), startDate, activeFlag.

- 5) Resource loan table: This table will contain list of the loaned resources. The attributes will be loanId (autogenerated sequence), resourceId (foreign key to resource table), membershipId (foreign key to membership table), dueDate, reissueCount
- 6) ResourceCategory table (missed previously): This table will link categories to resources, one resource can have one or more categories. Similarly, one category can have many resources. The attributes will be ResourceCategoryId, ResourceId and CategoryId.

### How the tables contextualize into application elements:

The minimal viable product should consist of following features

- Search by title, author, description
- Browse the books in all the available categories
- View the resource details by clicking on any resource

The additional features which can be included based on bandwidth and available time are placing a hold on resource, request extension/reissue and view user dashboard which showcases a list of all the resources checked out or on hold by the user. It also would require the login and logout functionality to recognize the user for supporting these user specific operations.

### MVP features:

**Search screen (MVP):** This screen would use the resource table to fetch the list of all available resources in the online library system. The list would be filtered by the keyword provided by the user in the search text box. The search criteria would also be provided (set to default as resource title) by the user. Based on this criterion the search API would try to filter the list by matching with respective column in resource table (i.e. title, author, description).

**Browse screen (MVP):** This screen would display various categories of resources (e.g. science, fiction, technology, kids, history, self-help, etc.) under each category header will be provided a list of resources which belong to that category. This screen will use resource table and category table. They will be joined to pull the resources associated with each category.

**Resource details screen (MVP):** This screen would display all the data associated with the resource pulled from resource table (i.e. resource title, publication year, author, description) as well as option to place hold on this resource.

### Stretch features:

**User dashboard:** User dashboard will display list of all the resources checked out by the user. It will use the membership table, resource loan table as well resource table to pull its details.

**Login/logout:** This functionality will enable user to login into the application and be able to perform user specific operations such as view dashboard, place hold and request extension. It will use user table and membership table to verify the user logging in.

**Place hold/request extension:** Placing hold on a resource will require an entry in resource loan table whereas requesting extension will require updating the reissue attempt on an existing checkout. Assumption is that the checkout will happen by an admin module which is out of scope for the given user case.

## ER diagram

Added the ER diagram which was missed previously.

