Design Document

The **SQL Programming Project** is implemented by me using java programming language has 5 source files attached to it

1. Library_Search
2. CheckIn
3. CheckOut
4. BorrowerAdd
5. Fines

Library_Search program is for displaying the search results given the Home page with initial search using either of ISBN, Author and Title.

Further I isolated the functionalities mentioned in the project into different source files for a modular approach and ease of debugging.

Library_Search:

This gives us an Initial GUI which displays the search criteria and the search results using a table. I used the Swing API (which comes in javax.swing package in JAVA SE) for displaying the results using GUI.

I used the Border Layout Design for the frame used in showing the search results. I used three regions of this Border Layout

North Region – for the search criteria

Text fields for each of ISBN, Author and Title for taking the input from the user.

East Region - for other options and Functionalities

I gave buttons for accessing each of different functions which are asked to be implemented in the project namely Check-in, Check-out, Fines and Adding a new Borrower. Upon clicking each button, I create a new instance of this source and call the required methods in the sources. To know whether a button has been clicked, I used ActionListeners at different points.

Central Region– for displaying search results using table

I used the built-in JTable object which is a part of Swing API for showing the results in the tabular format.

Table result has the following columns ISBN, Branch_id, Copies, Book Title, Branch Name, Book Author(s) and Available Copies.

Program searches the tables Book, Book_Authors, Authors, LIBRARY_BRANCH and BOOK_COPIES for all the columns except the Available copies.

The Available Copies is obtained from the book_loans table by checking the due_date for the book at the given branch.

Checking out a book is done by using a model selected Listener and passing the ISBN, Branch ID and other ends.

Check-Out:

I used single frame for displaying the checkout functionality, which shows the Book Title, ISBN, Branch_id and Available Copies. It prompts the user to enter the Card_no in the given entries.

The Check-out functionality is written using the checkoutBook() function and here the program checks whether the user has three or more books outstanding on his name by checking the BOOK_LOANS table for the user Card_no. This is done with the help of query checking the Date_in field for the taken books and comparing it with current books.

Check-In:

For check-In, I used two frames

1. For search criteria using any of Borrower name or card_no or book_id.
2. For displaying the results using the search criteria.
   Check-In is done upon querying the other tables.

Borrower-Add:

I added new borrowers to the system using this functionality, this is based on a single borrower table and I used a single Frame comprising of key fields for entry into the Borrower table.

The basic assumption here is that no user will have more than one SSN assigned to him so if the system sees more existing SSN for a new entry, it rejects the entry and shows the result.

Fines:

Fines are calculated on the existing entries and also the previous entries are shown with the help of querying the tables. I used a frame to display results using the Card_no and also normal entries in the fines table.

Database Tables Implementation:

The Book Table is implemented with two fields ISBN field as Character with a length of 10 and Title with a Variable Character extending length upto 100.

The Book_Authors Table is implemented using ISBN field as a foreign key referring to book and AuthorId field as a forign key referring to Authors Table.

Authors Table is created with the AuthorId as the primary key which is 10 length Integer field and Title, First,Middle and Last names. In addition to these, I added a new field fullname so as to make it convenient for the application o search using one field instead of all 4 fields.

BOOK_COPIES table was implemented taking Book_Id as the foreign key to ISBN in BOOK table and Branch_Id as the foreign key to LIBRARY_BRANCH table. No_of_copies Field is 10 length Integer.

LIBRARY_BRANCH table was implemented with Branch_Id as Integerof length 10, Branch name is a character of length 30 and Address is of length 50.

BORROWER table is implemented with fields Card_no varchar(10) , Ssn char(11), Fname varchar(20),Lname varchar(20) ,Addres varchar(50), Phone varchar(20). SSN has 11 characters with 2 '-'s in between them. Here Card_no is the primary key.

Book_Loans is implemented with Loan_id with primary key. Isbn, Card_no, Branch_Id are foreign keys with Book, BORROWER and LIBRARY_Branch tables. Additionally, as there are 3 date fields for Date_in, Date_out and Due_date of which Date_out and Due_date are not NULL.

Fines is implemented with Loan_id as the primary key and it is also a foreign key to BOOK_LOANS. We have other two fields Fine_Amt field which is a decimal with two decimal points.