DBMS Mini Project Report on
“Library Management System”

Submitted By

Rajat Belgundi ERP 1032180203

Kartik Bhutada ERP 1032180229

Amey Bhide ERP 1032180301

Divyang Bagla ERP 1032180739

Under the Guidance of

PROF. Pradnya Kulkarni

At

MIT-WPU

॥ विश्वशान्तिर्ध्रुवं ध्रुवा ॥

Dr. Vishwanath Karad

MIT WORLD PEACE UNIVERSITY | PUNE

TECHNOLOGY, RESEARCH, SOCIAL INNOVATION & PARTNERSHIPS

Contents

Abstract

Library management system is a project which aims in developing a computerized system to help perform daily work of the library. This application provides the facility of user and librarian login. Librarian can keep a track of the details of books issued and returned by the user. This project is developed in java which mainly focuses on basic operations in library like addition of new members, books, updating user information, searching books and members and facility to borrow and return book.

The Library Management System provides exact information about the number of books that are taken by the users, the number of books that are left, date of issue etc and it even calculates the fine automatically, thereby reducing the risk present in the manual management of libraries.

Overall this project is being developed to help the users and library staff to maintain the library in the best way possible and also reduce the human efforts.

Chapter -1

Introduction

1.1 Motivation :

Few reasons to select this project of Library Management System :-

- Library is an essential part of society as a whole.
- To overcome flaws in manual library management.
- To facilitate quick and efficient searching of books.
- To establish a systematic, neat and well organised library.

1.2 Objectives:

The main aims and objectives are:

- To have an efficient book issue and return facility.
- To provide information regarding authors and publishers.
- To display the late return and penalty information to the user.
- To facilitate easy addition and deletion of books to the Librarian.
- To provide monitoring of members to the Librarian.

Chapter -2

Problem Definition

2.1 Problem Statement:

The library management system is a software developed for monitoring and controlling the transactions in the library. The system maintains the record of books available in the library. It mainly focuses on adding new members, books, author and publisher details. The system also facilitates easy searching, issuing and returning of books. The system also shows the borrow details like lend date, return date, penalty / fine. The system also provides the facility to search for books based on authors and publishers as well. The system also provides the facility to monitor the members of the library.

The librarian details involve librarian id, name, password, email, contact details, address (street, city, zipcode). The member details involve member id, name, password, email, contact details, address (street, city, zipcode). The book details involve book id, name, aisle, genre. The author details involve author id, name, email. The Publisher details involve Publisher id, name, email. The contact details of librarian and members are assumed to be single valued.

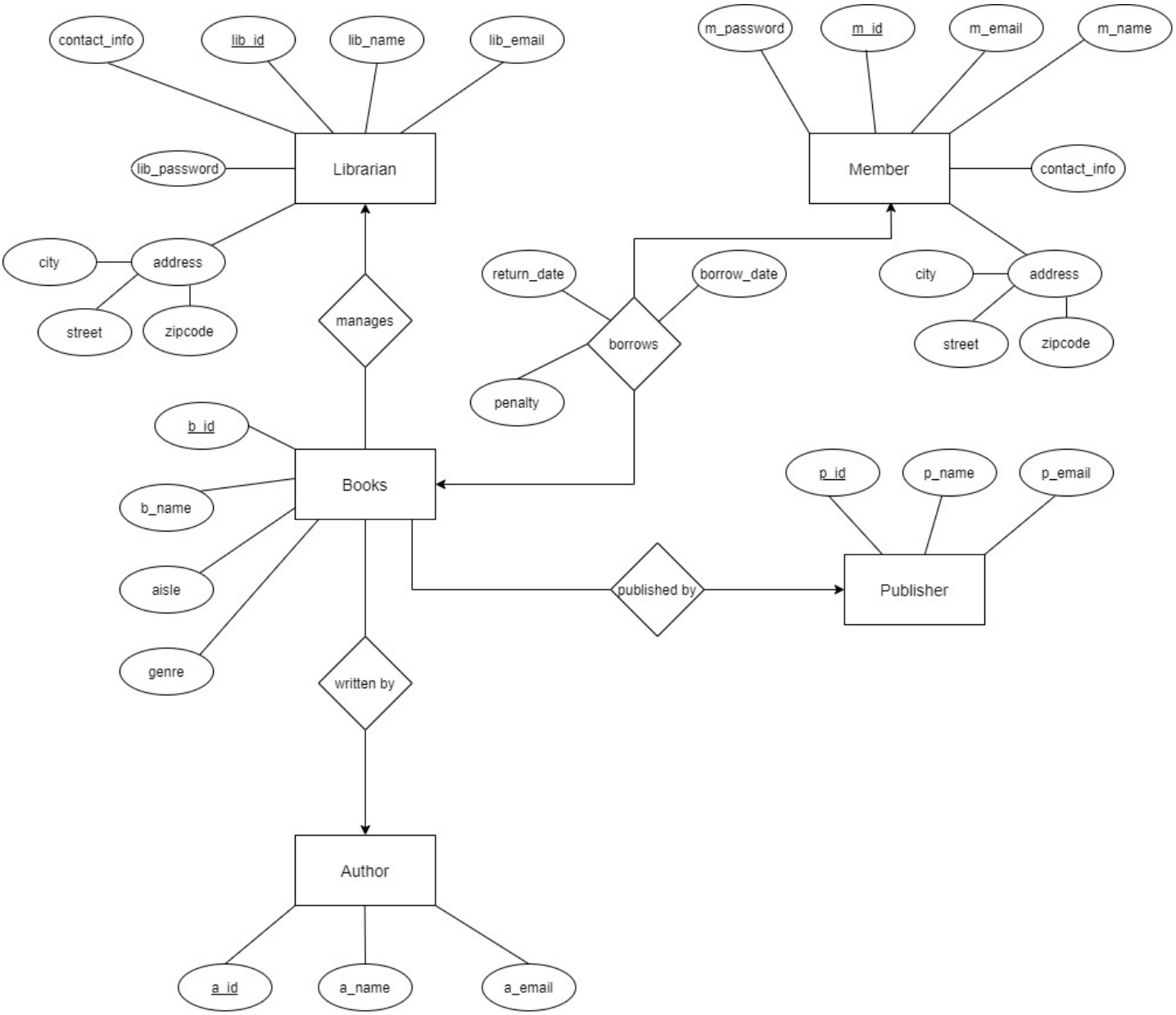
A member can issue a single book at a time. One author can write many books. One publisher can publish many books. A librarian can add / delete more than one books.

2.2 Tools and Technologies Used:

- MySQL Shell
- MySQL Workbench
- Eclipse IDE for Java Developers
- JDK
- JDBC
- Java Swing

Chapter – 3

Database Design (Entity Relationship Diagram)



Chapter - 4

Database Schema

LIBRARIAN

lib_id	lib_name	lib_email	lib_password	contact_info	street	city	zipcode

MEMBER

m_id	m_name	m_email	m_password	contact_info	street	city	zipcode

BOOKS

b_id	b_name	lib_id	a_id	p_id	aisle	genre

AUTHOR

a_id	a_name	a_email

PUBLISHER

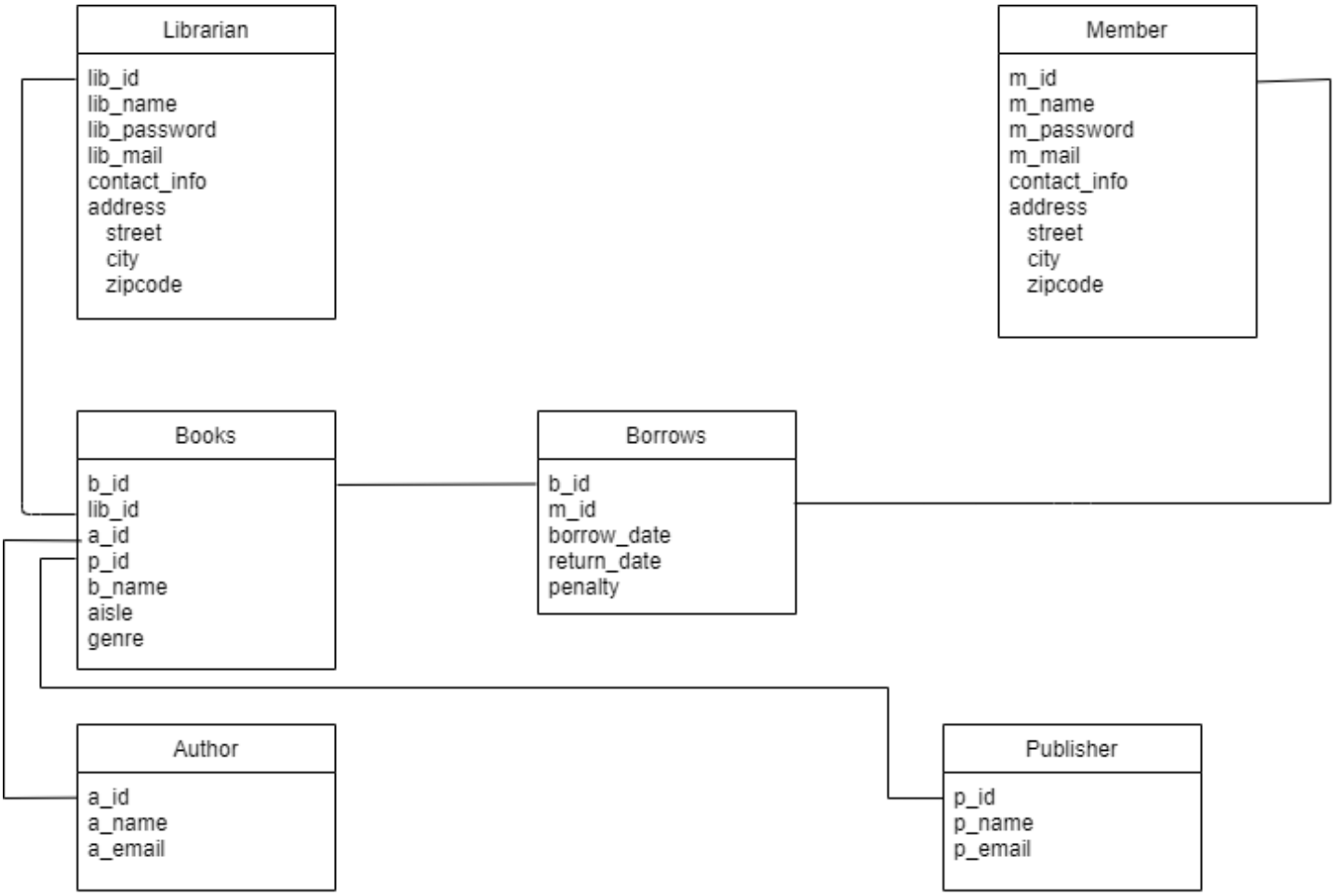
p_id	p_name	p_email

BORROWS

b_id	m_id	returndate	borrow_date	penalty

Chapter – 5

Relational Database Design (Schema Diagram)



Chapter – 6

Normalization

Primary Keys :-

Librarian(lib_id)

Member(m_id)

Books(b_id)

Author(a_id)

Publisher(p_id)

Borrows(b_id,m_id) – Composite primary keys

1st Normal Form :-

1st NF is concerned with multivalued attributes.

In this case, the entities do not have any multi valued attributes.

2nd Normal Form :-

For the tables to be in 2nd NF the non key attributes of the table should not be partially dependent on any single element of composite primary key.

In this case, borrows table has composite primary key (b_id, m_id)

All the attributes are dependent on composite primary key so, there is no partial dependency in this case.

3rd Normal Form :-

For the table to be in 3rd NF the non key attributes shouldn't have the transitive dependency on the primary key.

In this case, the entities do not have any transitive dependencies.

Chapter 7

SQL Commands Used

7.1 DDL

Create Database

```
MySQL localhost:33060+ ssl SQL > create database library;
Query OK, 1 row affected (0.0158 sec)
MySQL localhost:33060+ ssl SQL > show databases;
+--------------------+
| Database           |
+--------------------+
| company            |
| information_schema |
| library            |
| mysql              |
| performance_schema |
| sakila             |
| sys                |
| world              |
+--------------------+
8 rows in set (0.0028 sec)
MySQL localhost:33060+ ssl SQL > use library;
Default schema set to `library`.
Fetching table and column names from `library` for auto-completion... Press ^C to stop.
MySQL localhost:33060+ ssl library SQL > show tables;
Empty set (0.0026 sec)
```

Create Table Librarian

```
MySQL localhost:33060+ ssl library SQL > create table librarian(
                                      -> lib_id int primary key,
                                      -> lib_name varchar(30) not null,
                                      -> lib_password varchar(30) not null,
                                      -> lib_mail varchar(20) not null,
                                      -> contact_info char(10) not null,
                                      -> street varchar(30) not null,
                                      -> city varchar(20) not null,
                                      -> zipcode varchar(6) not null);
Query OK, 0 rows affected (0.0934 sec)
```

Create Table Member

```
MySQL  localhost:33060+ ssl  library  SQL > create table member(
                                        -> m_id int primary key,
                                        -> lib_id int,
                                        -> m_name varchar(20) not null,
                                        -> m_email varchar(20) not null,
                                        -> m_password varchar(20) not null,
                                        -> contact_info char(10) not null,
                                        -> street varchar(30) not null,
                                        -> city varchar(20) not null,
                                        -> zipcode varchar(6) not null,
                                        -> foreign key(lib_id) references librarian(lib_id) ON DELETE
set null);
Query OK, 0 rows affected (0.1671 sec)
```

Create Table books

```
MySQL  localhost:33060+ ssl  library  SQL > create table books(
                                        -> b_id int primary key,
                                        -> b_name varchar(30) not null,
                                        -> lib_id int,
                                        -> a_id int not null,
                                        -> p_id int,
                                        -> foreign key(lib_id) references librarian(lib_id) ON DELETE
set null,
                                        -> foreign key(a_id) references author(a_id) ON DELETE cascade
,
                                        -> foreign key(p_id) references publisher(p_id) ON DELETE set
null);
Query OK, 0 rows affected (0.0697 sec)
```

Create Table borrows

```
MySQL  localhost:33060+ ssl  library  SQL > create table borrows( b_id int not null, m_id int not null,
 primary key(b_id,m_id), borrow_date date not null, return_date date not null, penalty int, foreign key(
b_id) references books(b_id) ON DELETE CASCADE, foreign key(m_id) references member(m_id) ON DELETE CASC
ADE );
Query OK, 0 rows affected (0.1019 sec)
```

Create Table author

```
MySQL  localhost:33060+ ssl  library  SQL > create table author(
                                        -> a_id int primary key,
                                        -> a_name varchar(30) not null,
                                        -> a_email varchar(30) not null);
Query OK, 0 rows affected (0.0527 sec)
```

Create Table Pubisher

```
MySQL  localhost:33060+ ssl  library  SQL > create table publisher(
                                          -> p_id int primary key,
                                          -> p_name varchar(30) not null,
                                          -> p_email varchar(30) not null);
Query OK, 0 rows affected (0.0732 sec)
```

7.2 DML

Librarian Table

```
MySQL  localhost:33060+ ssl  library  SQL > select * from librarian;
+--------+---------------+-------------+-------------------+--------------+------------+-----------+---------+
| lib_id | lib_name      | lib_password | lib_mail          | contact_info | street     | city      | zipcode |
+--------+---------------+-------------+-------------------+--------------+------------+-----------+---------+
|      1 | kiran         | kiran123    | kiran@gmail.com   | 9876543218   | laxmi Rd   | Pune      | 411021  |
|      2 | Chandler Bing | cb123       | chandler@gmail.com| 6579234211   | Yemen Rd   | Yemen     | 569004  |
|      3 | Joey Tribbiani| howyoudoing | kenadams@gmail.com| 6556234211   | Bajirao Rd | New York  | 890675  |
|      4 | Jon Snow      | iknownothing| myqueen@gmail.com | 9874563218   | Tilak Rd   | Winerfell | 539504  |
+--------+---------------+-------------+-------------------+--------------+------------+-----------+---------+
4 rows in set (1.3877 sec)
```

Member Table

```
MySQL  localhost:33060+ ssl  library  SQL > select * from member;
+------+----------------+--------------------+---------------+--------------+---------------+--------+---------+
| m_id | m_name         | m_email            | m_password    | contact_info | street        | city   | zipcode |
+------+----------------+--------------------+---------------+--------------+---------------+--------+---------+
|    1 | sam            | sam@gmail.com      | sam123        | 8761893214   | karve Rd      | Pune   | 123455  |
|    2 | Ajinkya        | ajinkya@gmail.com  | ajinkya123    | 4567893214   | karve Rd      | Pune   | 123455  |
|    3 | Aniruddha      | aniruddha@gmail.com| aniruddha123  | 4564593214   | Paud Rd       | Pune   | 411496  |
|    4 | Sherlock       | sherlock@gmail.com | sherlocked123 | 7267593214   | Bakers Street | Pune   | 411030  |
|    5 | Parth          | parth@gmail.com    | parth123      | 4567925432   | Koregaon Park | Pune   | 411546  |
|    6 | Virat          | virat@gmail.com    | virat123      | 9864597251   | Sinhgad Rd    | Pune   | 411056  |
|    7 | Anushka        | anushka@gmail.com  | anushka123    | 9864363684   | Karolbaug     | Pune   | 411596  |
|    8 | Rhea           | rhra@gmail.com     | sushant123    | 9864563214   | Kelkar Rd     | Pune   | 456325  |
|    9 | Prabhas        | prabhas@gmail.com  | prabhas123    | 876455674    | SantaCruz     | Pune   | 411023  |
|   10 | Shraddha       | shraddha@gmail.com | shraddha123   | 9665044488   | Kakade Rd     | Pune   | 411011  |
|   11 | Regina Philange| phoebe@gmail.com   | philange123   | 9730879304   | Times square  | Pune   | 411758  |
|   12 | Monica         | monica@gmail.com   | monica123     | 9135679304   | Cental Perk   | Pune   | 411758  |
|   13 | Jennifer       | jennifer@gmail.com | jennifer123   | 9149879304   | Cental Perk   | Pune   | 411758  |
|   14 | Arya           | noone@gmail.com    | noone         | 9149875786   | Cental Perk   | Pune   | 411758  |
|   15 | Sansa          | sansa@gmail.com    | sansa123      | 9147899304   | Downing St    | Pune   | 411758  |
|   16 | jgkuen         | njkefnkj           | as123         | 8921278123   | nvjn          | nvfekj | 718627  |
+------+----------------+--------------------+---------------+--------------+---------------+--------+---------+
16 rows in set (0.4272 sec)
```

Books Table

```
MySQL  localhost:33060+ ssl  library  SQL > select * from books;
+------+------------------------------+------------------+-------+--------+------+------+
| b_id | b_name                       | genre            | aisle | lib_id | a_id | p_id |
+------+------------------------------+------------------+-------+--------+------+------+
|    1 | A Song of Ice and Fire       | Political Fiction |    17 |      1 |    7 |    2 |
|    3 | Harry Potter 1               | Fiction          |     5 |      1 |    1 |    1 |
|    4 | Alchemist                    | Philosophy       |     3 |      1 |    4 |    6 |
|    5 | Inferno                      | Science Fiction  |     6 |      1 |    5 |    5 |
|    6 | Murder on the Orient Express | Crime-Thriller   |     3 |      1 |    6 |    2 |
|    7 | A Secret of Nagas            | Fiction          |     3 |      1 |    2 |    1 |
|    8 | Immortals of Meluha          | Fiction          |     3 |      1 |    2 |    1 |
|    9 | Oath of the Vayuputras       | Fiction          |     3 |      1 |    2 |    1 |
|   10 | Scion of Ikshavaku           | Fiction          |     3 |      1 |    2 |    1 |
|   11 | Eleven Minutes               | Philosophy       |     3 |      1 |    4 |    6 |
|   12 | Harry Potter 2               | Fiction          |     1 |      3 |    1 |    1 |
|   13 | Harry Potter 3               | Fiction          |     1 |      3 |    1 |    1 |
|   14 | Harry Potter                 | Fiction          |     1 |      3 |    1 |    1 |
|   15 | Angels and Demons            | Historic Fiction |     6 |      2 |    5 |    5 |
|   16 | Origin                       | Fiction          |     6 |      2 |    5 |    5 |
|   17 | The Winds of Winter          | Fiction          |    17 |      1 |    7 |    2 |
|   18 | A Dance of Dragons           | Fiction          |    17 |      1 |    7 |    2 |
|   19 | A Storm of Swords            | Fiction          |    17 |      1 |    7 |    2 |
|   20 | A Feast for Crows            | Fiction          |    17 |      1 |    7 |    2 |
|   21 | A Dream of Spring            | Fiction          |    17 |      1 |    7 |    2 |
|   22 | A Clash of Kings             | Fiction          |    17 |      1 |    7 |    2 |
|   23 | The Zahir                    | Philosophy       |    17 |      1 |    4 |    6 |
|   24 | The Deception Point          | Fiction          |    17 |      1 |    5 |    5 |
|   25 | The Hobbit                   | Fiction          |    17 |      1 |    8 |    1 |
|   26 | hgidf                        | ijegn            |    42 |      1 |    3 |    3 |
+------+------------------------------+------------------+-------+--------+------+------+
25 rows in set (0.2328 sec)
```

Borrows Table

```
MySQL  localhost:33060+ ssl  library  SQL > select * from borrows;
+------+------+-------------+-------------+---------+
| b_id | m_id | borrow_date | return_date | penalty |
+------+------+-------------+-------------+---------+
|    3 |    3 | 2020-08-20  | 2020-09-04  |      80 |
|    4 |    4 | 2020-08-22  | 2020-09-06  |      60 |
|    6 |    6 | 2020-08-13  | 2020-08-28  |     150 |
|    7 |    7 | 2020-08-21  | 2020-09-05  |      70 |
|    8 |    8 | 2020-08-21  | 2020-09-05  |      70 |
|    9 |    9 | 2020-08-28  | 2020-09-13  |       0 |
|   10 |   10 | 2020-09-01  | 2020-09-15  |       0 |
|   11 |   11 | 2020-09-05  | 2020-09-19  |       0 |
|   12 |   12 | 2020-09-06  | 2020-09-20  |       0 |
|   14 |   14 | 2020-09-07  | 2020-09-21  |       0 |
|   15 |   15 | 2020-09-08  | 2020-09-22  |       0 |
+------+------+-------------+-------------+---------+
11 rows in set (0.0976 sec)
```

Author Table

```
MySQL  localhost:33060+ ssl  library  SQL > select * from author;
+------+-----------------+----------------------+
| a_id | a_name          | a_email              |
+------+-----------------+----------------------+
|    1 | J.K.Rowlings    | jkrowlings@gmail.com |
|    2 | Amish Tripathi  | amish@gmail          |
|    3 | Dan Brown       | dan@gmail.com        |
|    4 | Paulo Coelho    | paulo@gmail.com      |
|    5 | Dan Brown       | dan@gmail.com        |
|    6 | Agatha Christie | agatha@gmail.com     |
|    7 | G.R.R.Martin    | grrmartin@gmail.com  |
|    8 | J.R.R.Tolkien   | jrr@gmail.com        |
+------+-----------------+----------------------+
8 rows in set (0.6443 sec)
```

Publisher Table

```
MySQL  localhost:33060+ ssl  library  SQL > select * from publisher;
+------+----------------------------+----------------------+
| p_id | p_name                     | p_email              |
+------+----------------------------+----------------------+
|    1 | Rajdeep                    | rajdeep@gmail.com    |
|    2 | Bloomsbury Publishing      | blooms@gmail.com     |
|    3 | Arihant                    | arihant@gmail.com    |
|    4 | HarperCollins Publications | harper@gmail.com     |
|    5 | Mackintosh                 | makintosh@gmail.com  |
|    6 | Rupa publishing            | rupa@gmail.com       |
+------+----------------------------+----------------------+
6 rows in set (1.0345 sec)
```

7.3 DCL

```
MySQL  SQL > \connect root@localhost
Creating a session to 'root@localhost'
Fetching schema names for autocompletion... Press ^C to stop.
Your MySQL connection id is 34 (X protocol)
Server version: 8.0.21 MySQL Community Server - GPL
No default schema selected; type \use <schema> to set one.
MySQL  localhost:33060+ ssl  SQL > create user 'lib_member'@'localhost' identified by 'member123';
Query OK, 0 rows affected (0.0532 sec)
MySQL  localhost:33060+ ssl  SQL > grant insert,select on library.borrows to 'lib_member'@'localhost';
Query OK, 0 rows affected (0.0209 sec)
MySQL  localhost:33060+ ssl  SQL > grant execute on procedure library.calc_penalty to 'lib_member'@'localhost';
Query OK, 0 rows affected (0.0244 sec)
MySQL  localhost:33060+ ssl  SQL > grant execute on function library.chech_user_inBorrows to 'lib_member'@'localhost';
ERROR: 1305: FUNCTION chech_user_inborrows does not exist
MySQL  localhost:33060+ ssl  SQL > grant execute on function library.check_user_inBorrows to 'lib_member'@'localhost';
Query OK, 0 rows affected (0.0101 sec)
MySQL  localhost:33060+ ssl  SQL > grant select on library.books to 'lib_member'@'localhost';
Query OK, 0 rows affected (0.0100 sec)
MySQL  localhost:33060+ ssl  SQL > grant select on library.author to 'lib_member'@'localhost';
Query OK, 0 rows affected (0.0178 sec)
MySQL  localhost:33060+ ssl  SQL > grant select on library.publisher to 'lib_member'@'localhost';
Query OK, 0 rows affected (0.0116 sec)
MySQL  localhost:33060+ ssl  SQL > grant select,update,insert on library.member to 'lib_member'@'localhost';
Query OK, 0 rows affected (0.0118 sec)
MySQL  localhost:33060+ ssl  SQL >
```

7.4 Triggers

```
delimiter $$

create trigger history

after delete on borrows

for each row

BEGIN

insert into history values(OLD.m_id,OLD.b_id,OLD.borrow_date,curdate());

END $$
```

7.5 Procedures & Functions

Procedure

```
delimiter $$

create procedure calc_penalty(IN mem_id int)

BEGIN

declare r_date date;

declare diff_days int;

set r_date = (select return_date from borrows where m_id=mem_id);

set diff_days = DATEDIFF(curdate(),r_date);

if(diff_days > 0) then

UPDATE borrows

set penalty=diff_days*10

WHERE m_id=mem_id;

end if;

END $$
```

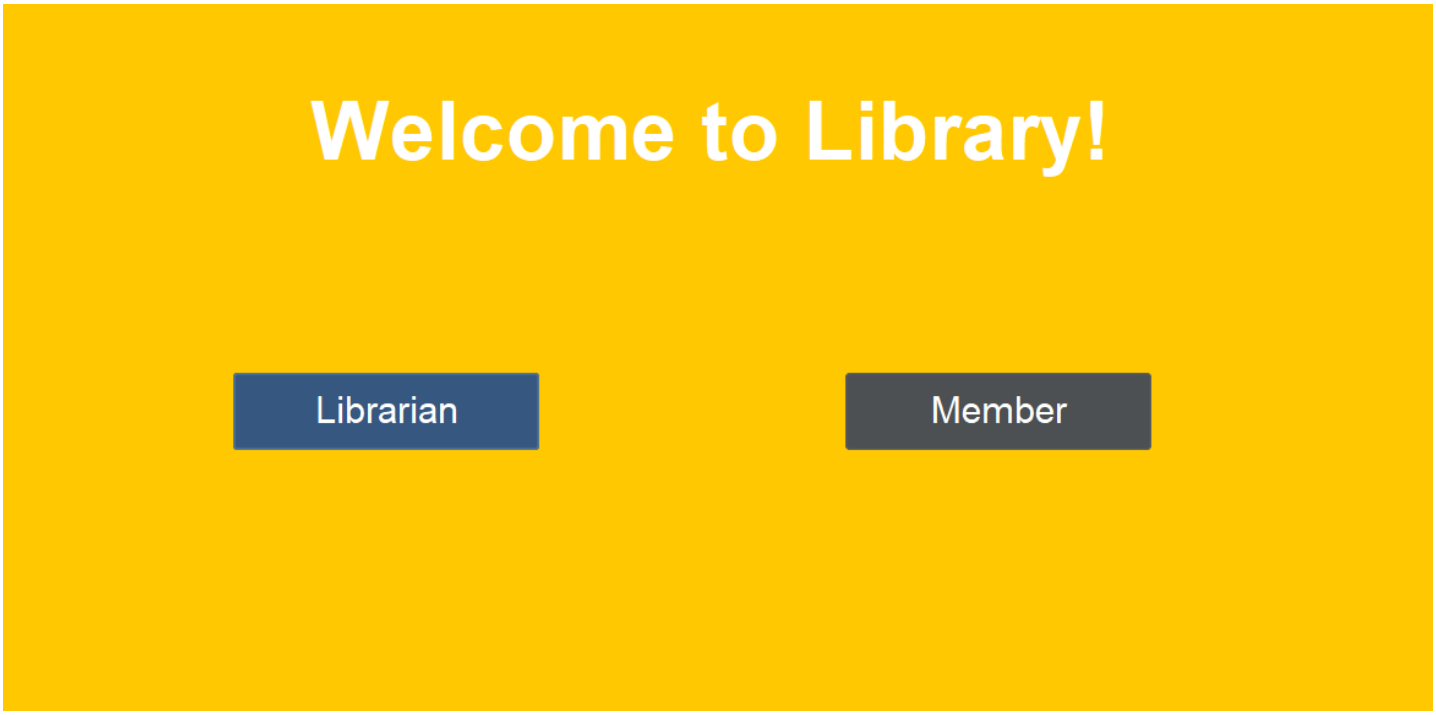
Function

```
delimiter $$

create function check_user_inBorrows(mem_id int)

returns int

DETERMINISTIC

BEGIN

declare mem_check int default 0;

set mem_check = (select m_id from borrows where m_id=mem_id);

if(mem_check != 0) then

return 1;

else

return 0;

end if;

END $$
```

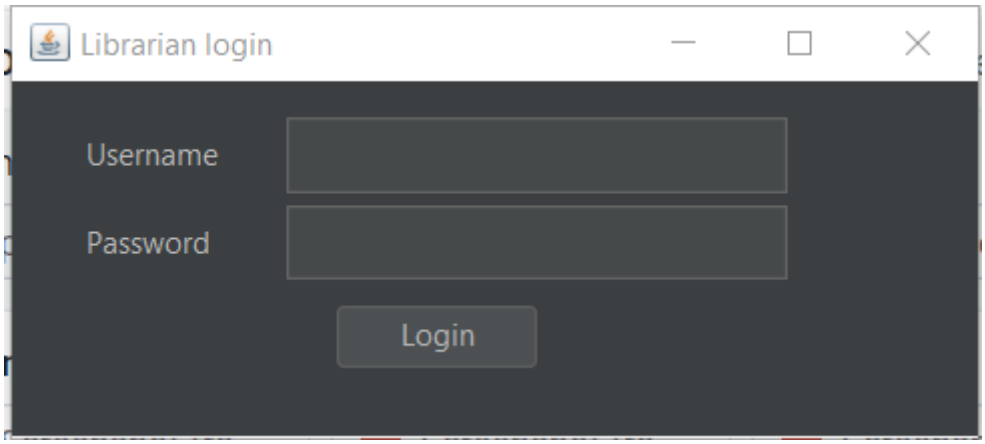
Chapter – 8

Frontend GUI

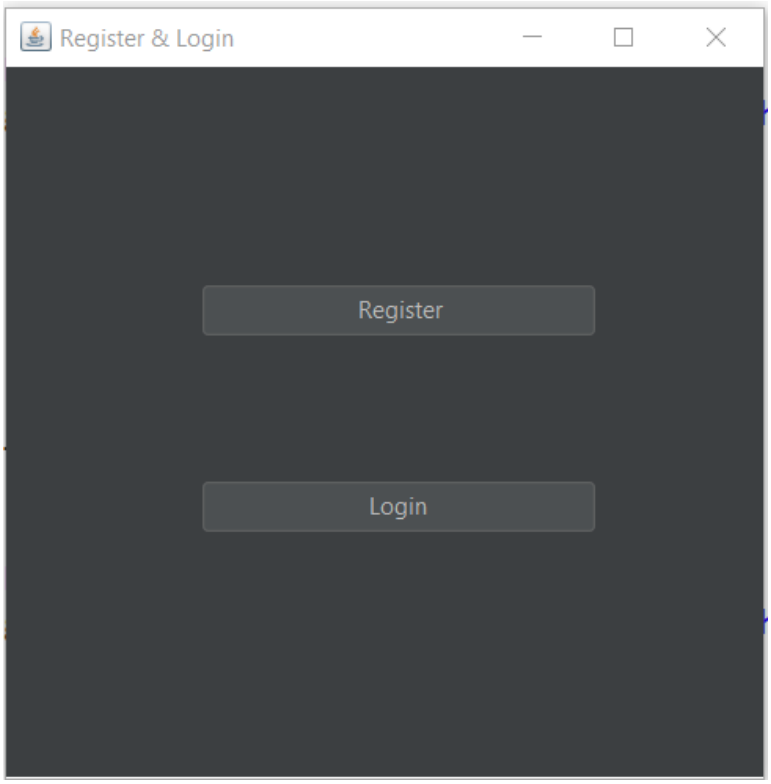
Welcome Window



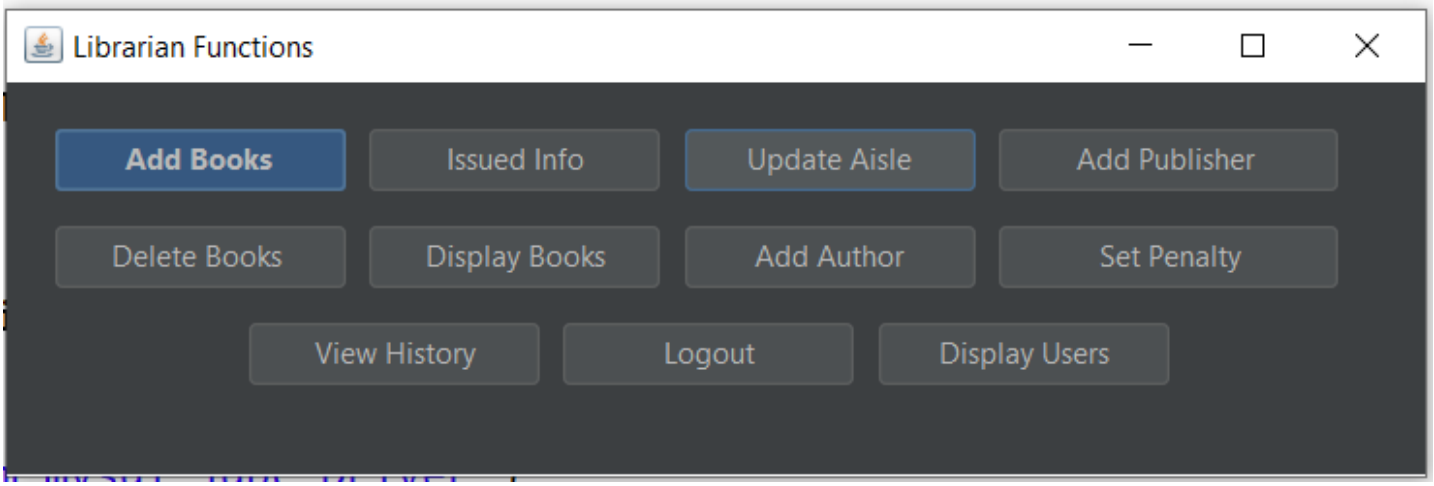
Librarian Login



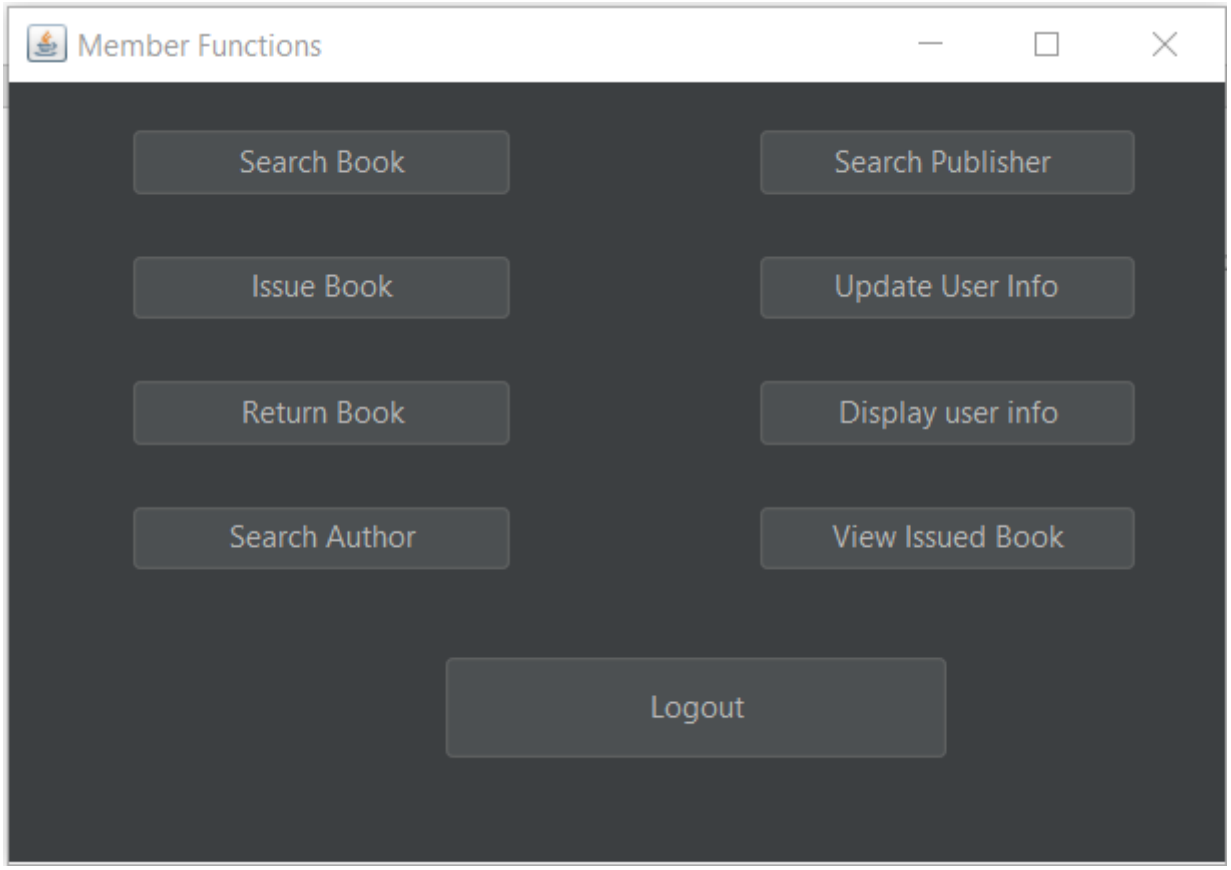
Member Login & Registration



Librarian Functions



Member Functions



Chapter 9

Conclusion

The implemented system automates the tasks involved in managing the library in a efficient way. The system is user friendly and also librarian friendly. The system provides author and publisher information for the avid readers. This system also provides easy tracking of members which makes it easier for the librarian to manage and control the transactions of the library. For the digital world where everything works through mobile and computers, this system is the perfect replacement for the traditional library management system.

Source Code:- **https://github.com/bagladivyang03/Library-Management-System**

References

S. Kappagantula **"How To Create Library Management System Project in Java?"** https://www.edureka.co/blog/library-management-system-project-in-java (5-sept-2020)

Prabhakar Gupta **"Library Management System "** https://github.com/prabhakar267/library-management-system (6-sept-2020)

Open Source Contribution **"Library Management System JAVA"** https://github.com/OSSpk/Library-Management-System-JAVA (8-sept-2020)