DATABASE MANAGEMENT SYSTEM

PROJECT NAME – LIBRARY MANAGEMENT SYSTEM

Submitted by :

Kiran Rani

URN:2203848

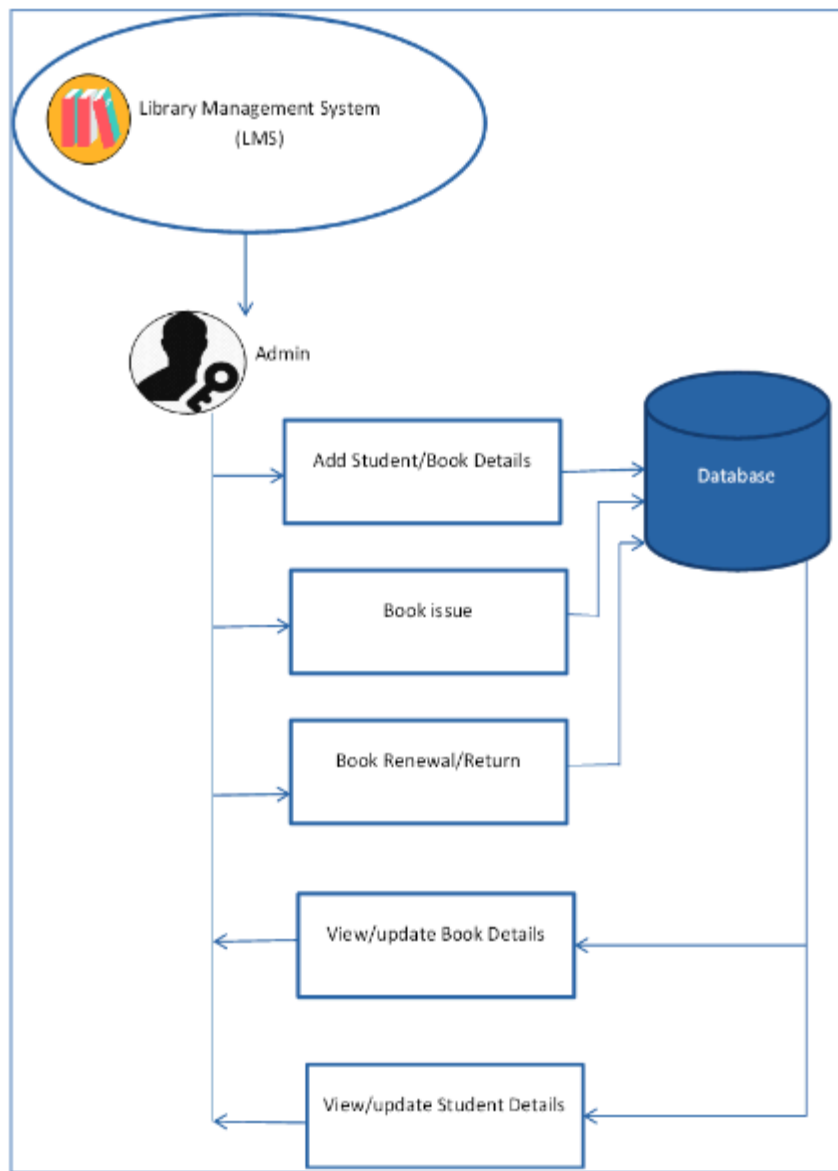CRN:2221068

BRANCH: IT A2

Submitted to :

Dr.Mohanjit Kaur   Kang

1. Introduction

Our Library Management System is like a digital assistant for our library..Library database Management System is **a system that shows all the available books and their count and also books taken by people**, the date on which they took that particular book, expected date of return, late due fees, membership details, and so on. It helps us keep track of all the books and resources we have, and makes it easy for you to find and borrow them.

3.1 Block Schematic

2. Objectives

The primary objectives of the Library Management System are as follows:

Efficient Resource Management: Organize library materials effectively to streamline operations.Ensure easy access to resources for patrons.

Enhanced User Experience: Provide a user-friendly interface for easy navigation.Simplify borrowing and returning processes.

Promotion of Information Access: Facilitate quick and easy retrieval of information.access to diverse resources across subjects.

Data-driven Decision Making: Use data analytics to make informed decisions.Monitor usage trends for better resource allocation.

Automation of Routine Tasks: Automate repetitive tasks to save time.Free up staff for more valuable activities.

Security and Privacy: Ensure the security and privacy of patron information.Implement controls to prevent unauthorized access.

3. Implementation Details : Tables in the Library System:

1. **Books:**
   - Attributes:
     - book_id (Primary Key): Unique identifier for each book.
     - title: Title of the book.
     - author: Author of the book.
     - isbn: International Standard Book Number of the book (unique).
     - publication_year: Year the book was published.
     - genre: Genre of the book.
     - copies_available: Number of copies available for borrowing.
2. **Borrowers:**
   - Attributes:
     - borrower_id (Primary Key): Unique identifier for each borrower.
     - name: Name of the borrower.
     - email: Email address of the borrower (unique).
     - phone: Phone number of the borrower.
     - address: Address of the borrower.
3. **Loans:**
   - Attributes:
     - loan_id (Primary Key): Unique identifier for each loan.
     - book_id (Foreign Key): Identifier of the book being loaned.
     - borrower_id (Foreign Key): Identifier of the borrower taking the loan.
     - loan_date: Date the book was borrowed.
     - due_date: Date by which the book is due to be returned.
     - return_date: Date the book was returned.

4. **MembershipCard:**

   - Attributes:
     - card_id (Primary Key):Unique identifier for each membership card.
     - member_id (Foreign Key):Identifier of the member associated with the card
     - card.card_number:Unique identification number printed on the membership card
     . expiration_date:Description: Date when the membership card expires.

Rules for Managing the App's Databases:

1. Books:
   - Each book must have a unique ISBN.
   - The copies_available column should reflect the number of copies currently available for borrowing.
   - Books can belong to one or more genres.
   - Updating book information should be done carefully to maintain data integrity.
2. Borrowers:
   - Each borrower must have a unique email address.
   - Borrower information should be accurately recorded to ensure communication regarding loans and returns.
   - Managing borrower details such as address and phone number should be done accurately to avoid miscommunication.
3. Loans:
   - Each loan must have a unique identifier.
   - Foreign key constraints ensure that only valid book and borrower IDs are referenced.
   - The due_date should be calculated based on the loan date and predefined loan duration.
   - Proper management of loan statuses (e.g., overdue, returned) is crucial for tracking borrowed books.

4.Data Integrity and Access Control:
   - Ensure that all membership card records maintain data integrity to guarantee accurate and reliable information, including enforcing referential integrity constraints and logging modifications.
   - Restrict access to the membership card database to authorized users only, implementing role-based access control mechanisms to protect sensitive information and maintain privacy.

Create Database and Tables

```
mysql> CREATE TABLE books (
    ->     book_id INT AUTO_INCREMENT PRIMARY KEY,
    ->     title VARCHAR(255) NOT NULL,
    ->     author VARCHAR(255) NOT NULL,
    ->     isbn VARCHAR(20) NOT NULL,
    ->     publication_year INT,
    ->     genre VARCHAR(100),
    ->     copies_available INT DEFAULT 1,
    ->     UNIQUE (isbn)
    -> );
```

```
mysql> CREATE TABLE borrowers (
    ->     borrower_id INT AUTO_INCREMENT PRIMARY KEY,
    ->     name VARCHAR(255) NOT NULL,
    ->     email VARCHAR(255) NOT NULL,
    ->     phone VARCHAR(20),
    ->     address TEXT
    -> );
```

```
mysql> CREATE TABLE loans (
    ->     loan_id INT AUTO_INCREMENT PRIMARY KEY,
    ->     book_id INT,
    ->     borrower_id INT,
    ->     loan_date DATE,
    ->     due_date DATE,
    ->     return_date DATE,
    ->     FOREIGN KEY (book_id) REFERENCES books(book_id),
    ->     FOREIGN KEY (borrower_id) REFERENCES borrowers(borrower_id)
    -> );
```

```
mysql> CREATE TABLE MembershipCards (
    ->     CardID INT PRIMARY KEY,
    ->     MemberID INT,
    ->     CardNumber VARCHAR(20) UNIQUE,
    ->     ExpirationDate DATE,
    ->     IssuedDate DATE,
    ->     IsActive BOOLEAN
    -> );
```

Insert Data into Tables

```
mysql> INSERT INTO books (title, author, isbn, publication_year, genre, copies_available)
    -> VALUES
    -> ('The Great Gatsby', 'F. Scott Fitzgerald', '9780743273565', 1925, 'Classic', 5),
    -> ('To Kill a Mockingbird', 'Harper Lee', '9780061120084', 1960, 'Fiction', 3),
    -> ('1984', 'George Orwell', '9780451524935', 1949, 'Dystopian', 2);
```

```
mysql> INSERT INTO borrowers (name, email, phone, address)
    -> VALUES
    -> ('Alice Johnson', 'alice@example.com', '123-456-7890', '123 Oak St, City'),
    -> ('Bob Smith', 'bob@example.com', '987-654-3210', '456 Maple Ave, Town');
Query OK, 2 rows affected (0.01 sec)
Records: 2  Duplicates: 0  Warnings: 0
```

```
mysql> -- Borrower Alice borrows "The Great Gatsby" on 2024-04-26, due on 2024-05-10
mysql> INSERT INTO loans (book_id, borrower_id, loan_date, due_date)
    -> VALUES
    -> (1, 1, '2024-04-26', '2024-05-10');
Query OK, 1 row affected (0.01 sec)
```

```
mysql> INSERT INTO MembershipCards (CardID, MemberID, CardNumber, ExpirationDate, IssuedDate, IsActive)
    -> VALUES
    -> (1, 101, 'LIB-001', '2024-12-31', '2024-04-28', true),
    -> (2, 102, 'LIB-002', '2024-12-31', '2024-04-29', true),
    -> (3, 103, 'LIB-003', '2024-12-31', '2024-04-30', true);
Query OK, 3 rows affected (0.04 sec)
Records: 3  Duplicates: 0  Warnings: 0
```

TABLES Retrieve Data

```
mysql> -- Retrieve all books
mysql> SELECT * FROM books;
+---------+--------------------+--------------------+----------------+------------------+-------------+------------------+
| book_id | title              | author             | isbn           | publication_year | genre       | copies_available |
+---------+--------------------+--------------------+----------------+------------------+-------------+------------------+
|       1 | Book Title 1       | Author 1           | ISBN1          |             2022 | Fiction     |                3 |
|       2 | Book Title 2       | Author 2           | ISBN2          |             2020 | Non-Fiction |                2 |
|       3 | The Great Gatsby   | F. Scott Fitzgerald| 9780743273565  |             1925 | Classic     |                5 |
|       4 | To Kill a Mockingbird | Harper Lee      | 9780061120084  |             1960 | Fiction     |                3 |
|       5 | 1984               | George Orwell      | 9780451524935  |             1949 | Dystopian   |                2 |
+---------+--------------------+--------------------+----------------+------------------+-------------+------------------+
5 rows in set (0.00 sec)

mysql>
mysql> -- Retrieve all borrowers
mysql> SELECT * FROM borrowers;
+-------------+--------------+-----------------------+--------------+----------------------+
| borrower_id | name         | email                 | phone        | address              |
+-------------+--------------+-----------------------+--------------+----------------------+
|           1 | Borrower 1   | borrower1@example.com | 123-456-7890 | 123 Main St, City     |
|           2 | Borrower 2   | borrower2@example.com | 987-654-3210 | 456 Elm St, Town      |
|           3 | Alice Johnson| alice@example.com     | 123-456-7890 | 123 Oak St, City      |
|           4 | Bob Smith    | bob@example.com       | 987-654-3210 | 456 Maple Ave, Town   |
+-------------+--------------+-----------------------+--------------+----------------------+
4 rows in set (0.00 sec)

mysql>
mysql> -- Retrieve all loans
mysql> SELECT * FROM loans;
+---------+---------+-------------+------------+------------+-------------+
| loan_id | book_id | borrower_id | loan_date  | due_date   | return_date |
+---------+---------+-------------+------------+------------+-------------+
|       1 |       1 |           1 | 2024-04-26 | 2024-05-10 | NULL        |
|       2 |       2 |           2 | 2024-04-27 | 2024-05-11 | NULL        |
|       3 |       3 |           1 | 2024-04-28 | 2024-05-12 | NULL        |
+---------+---------+-------------+------------+------------+-------------+
3 rows in set (0.00 sec)
```

```
mysql> select*from MembershipCards;
+--------+----------+------------+----------------+------------+----------+
| CardID | MemberID | CardNumber | ExpirationDate | IssuedDate | IsActive |
+--------+----------+------------+----------------+------------+----------+
|      1 |      101 | LIB-001    | 2024-12-31     | 2024-04-28 |        1 |
|      2 |      102 | LIB-002    | 2024-12-31     | 2024-04-29 |        1 |
|      3 |      103 | LIB-003    | 2024-12-31     | 2024-04-30 |        1 |
+--------+----------+------------+----------------+------------+----------+
3 rows in set (0.00 sec)
```

Update Data

```
mysql> -- Update book information
mysql> UPDATE books
    -> SET genre = 'Thriller'
    -> WHERE title = 'Book Title 1';
Query OK, 0 rows affected (0.08 sec)
Rows matched: 1  Changed: 0  Warnings: 0

mysql>
mysql> -- Update borrower information
mysql> UPDATE borrowers
    -> SET address = '789 Oak St, Village'
    -> WHERE name = 'Borrower 1';
Query OK, 0 rows affected (0.01 sec)
Rows matched: 1  Changed: 0  Warnings: 0
```

```
mysql> select*from books;
+---------+----------------------+----------------------+---------------+------------------+-------------+------------------+
| book_id | title                | author               | isbn          | publication_year | genre       | copies_available |
+---------+----------------------+----------------------+---------------+------------------+-------------+------------------+
|       1 | Book Title 1         | Author 1             | ISBN1         |             2022 | Thriller    |                3 |
|       2 | Book Title 2         | Author 2             | ISBN2         |             2020 | Non-Fiction |                2 |
|       3 | The Great Gatsby     | F. Scott Fitzgerald  | 9780743273565 |             1925 | Classic     |                5 |
|       4 | To Kill a Mockingbird| Harper Lee           | 9780061120084 |             1960 | Fiction     |                3 |
|       5 | 1984                 | George Orwell        | 9780451524935 |             1949 | Dystopian   |                2 |
+---------+----------------------+----------------------+---------------+------------------+-------------+------------------+
5 rows in set (0.00 sec)

mysql> select*from borrowers;
+-------------+---------------+----------------------+--------------+---------------------+
| borrower_id | name          | email                | phone        | address             |
+-------------+---------------+----------------------+--------------+---------------------+
|           1 | Borrower 1    | borrower1@example.com | 123-456-7890 | 789 Oak St, Village |
|           2 | Borrower 2    | borrower2@example.com | 987-654-3210 | 456 Elm St, Town    |
|           3 | Alice Johnson | alice@example.com    | 123-456-7890 | 123 Oak St, City    |
|           4 | Bob Smith     | bob@example.com      | 987-654-3210 | 456 Maple Ave, Town |
+-------------+---------------+----------------------+--------------+---------------------+
4 rows in set (0.00 sec)
```

Delete related records from the `loans` table:

```
mysql> DELETE FROM loans
    -> WHERE book_id IN (SELECT book_id FROM books WHERE title = 'Book Title 2');
Query OK, 1 row affected (0.07 sec)

mysql>
mysql> DELETE FROM loans
    -> WHERE borrower_id IN (SELECT borrower_id FROM borrowers WHERE name = 'Borrower 2');
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> select*from loan;
ERROR 1146 (42S02): Table 'library_system.loan' doesn't exist
```

SELECT STATMENT

```
mysql> select bookauother from books;
+--------------------+
| bookauother        |
+--------------------+
| Harper Lee         |
| George Orwell      |
| J.D. Salinger      |
| F. Scott Fitzgerald |
+--------------------+
4 rows in set (0.11 sec)
```

COLUMN ALIANCE

```
mysql> select bookauother as"Book_auother"
    -> from books;
+--------------------+
| Book_auother       |
+--------------------+
| Harper Lee         |
| George Orwell      |
| J.D. Salinger      |
| F. Scott Fitzgerald |
+--------------------+
4 rows in set (0.00 sec)
```

CONCATION OPERATION

```
mysql> select concat(bookid,
    -> '
    '> ;
    '>
    -> ,booktitle )as BOOKCOMBINATIONS
    -> from books;
+----------------------------+
| BOOKCOMBINATIONS           |
+----------------------------+
| 23
,
To Kill a Mockingbird   |
| 34
,
1984                    |
| 223
,
The Catcher in the Rye  |
| 4
,
The Great Gatsby        |
+----------------------------+
4 rows in set (0.01 sec)
```