PROJECT PROPOSAL

ON

Library Management System

Guided By:

Mr. Anuj kumar

Created By:

Anshika Soni

Khushi Krishna

Kanika Chaudhary

AF I'D:

AF04991159

AF04991215

AF04971772

Batch Code:

ANP-D2405

Table Of Contents

Index

PROJECT TITLE:

Library Management System (Terminal-Based)

Using Java, JDBC, MySQL

INTRODUCTION

Libraries are immensely important in supplying knowledge resources among students, teachers, and readers. Conventionally, maintaining a library involves manually keeping a record of books in stock, member records, and tracking of issued and returned books. These manual systems are time-consuming, full of human errors, and difficult to manage as the number of books and users grows.

To overcome these limitations, a library management system is introduced to computerize the entire operations of the library. This project focuses on developing a terminal-based application using Java, JDBC, and MySQL, which ensures efficient management of library resources while maintaining data accuracy and security.

OBJECTIVE:

- The automation of library operations, including the management of books, members, and transactions.
- To reduce paperwork and avoid manual errors in keeping records.
- To store all the information in a secure and well-organized database.
- This will ensure fast searching and retrieval of records of books and members.
- Arranging books for easier access and management.
- To systematically track the movement of books-issued and returned-to understand their availability status.
- This aims to provide a simple interface so admins can easily operate the system.

PROJECT CATEGORY:

- This project belongs to the Database Management System category.
- It is a terminal based software application.
- Core Java is used as the primary programming language.
- Java Database Connectivity (JDBC) is used to connect database with java.
- MySQL is used as the relational database management system.
- The system demonstrate CRUD operations (i.e. create, retrieve, update and delete operations).

ANALYSIS

LIBRARY MANAGEMENT SYSTEM: MODULES

1. Admin Management
2. Member Management
3. Category Management
4. Book Management
5. Issue and Return Management

Module 1: Admin Management

1.1– Admin Login
1.2 – Admin Profile Update
1.3 – View System Overview / Dashboard

Module 2: Member Management

2.1 – Member Registration
2.2 – Member Update
2.3 – Member List
2.4 – Member Details / Profile
2.5 – Member Delete

Module 3: Category Management

3.1 – Add Category
3.2 – Update Category
3.3 – Category List
3.4 – Category Delete

Module 4: Book Management

4.1 – Add New Book
4.2 – Update Book Details
4.3 – Book List
4.4 – Book Details / View Book
4.5 – Delete Book

Module 5: Issue and Return Management

5.1 – Issue Book
5.2 – Return Book
5.3 – Issue List
5.4 – Return List
5.5 – Fine Calculation (If applicable)
5.6 – Due Date Tracking

DATABASE DESIGN

Table 1: Admin

Fields	datatype	properties
admin_id	int	primary key
username	varchar(50)	not null
password	varchar(50)	not null
name	varchar(100)	not null

Table 2: Member

fields	datatype	properties
member_id	int	primary key
name	varchar(100)	not null
email	varchar(100)	not null
phone	varchar(20)	not null
address	varchar(300)	not null

Table 3: Category

fields	datatype	properties
category_id	int	primary key
category_name	varchar(100)	not null

Table 4: Books

fields	datatype	properties
book_id	int	primary key
title	varchar(200)	not null
author	varchar(150)	not null
publisher	varchar(150)	not null
category_id	int	not null, foreign key

Table 5: Issue_Return

fields	datatype	properties
transaction_id	int	primary key
book_id	int	not null, foreign key
member_id	int	not null, foreign key
issue_date	date	not null
return_date	date	null
status	varchar(20)	not null

Relationship:

Books – Category:
One Category can have many Books.

Issue_Return – Books & Members:
Multiple issue-return records can belong to one book or one member.

Table 6: Category_Books_Relation

(One Category → Many Books)

fields	datatype	properties
category_id	int	primary key (Category table)
book_id	int	foreign key (Books table)

Table 7: Member_Transaction_Relation

(One Member → Many Transactions)

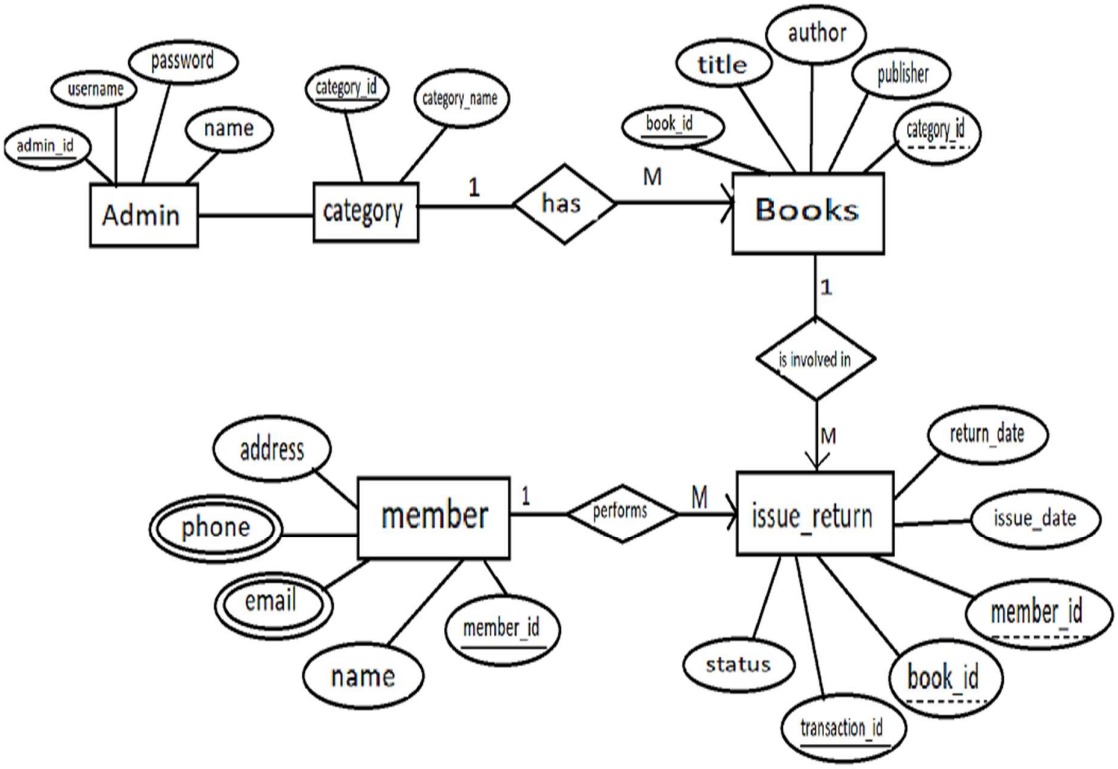
fields	datatype	properties
member_id	int	primary key (Member table)
transaction_id	int	foreign key (Issue_Return table)

Table 8: Book_Transaction_Relation

(One Book → Many Transactions)

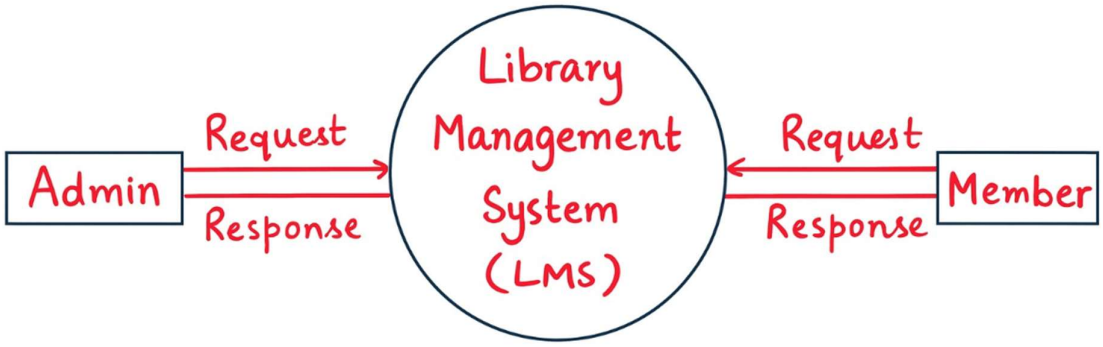
fields	datatype	properties
book_id	int	primary key (Books table)
transaction_id	int	foreign key (Issue_Return table)

ENTITY RELATIONSHIP DIAGRAM

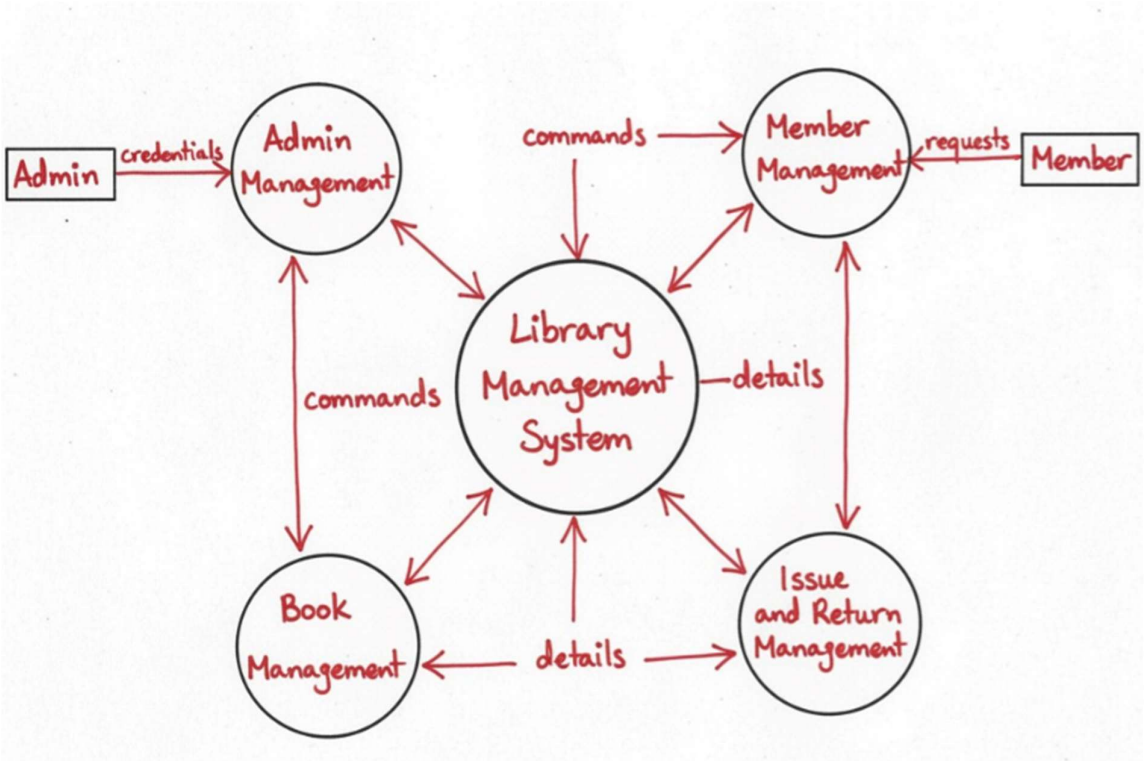


DATA FLOW DIAGRAM

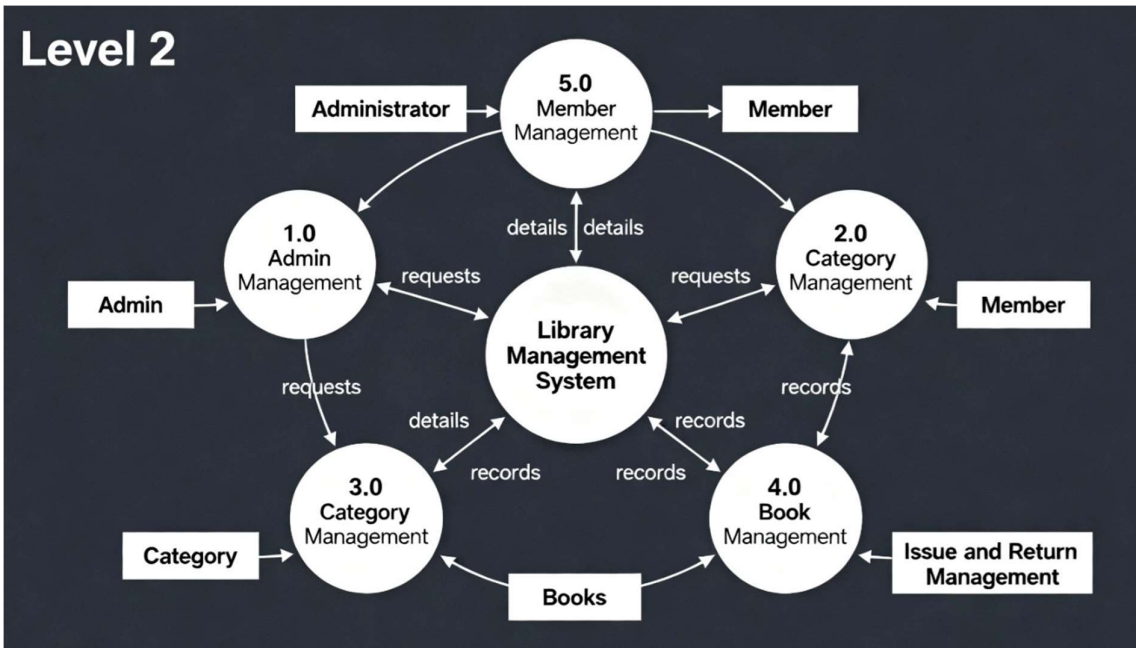
ZERO LEVEL DFD



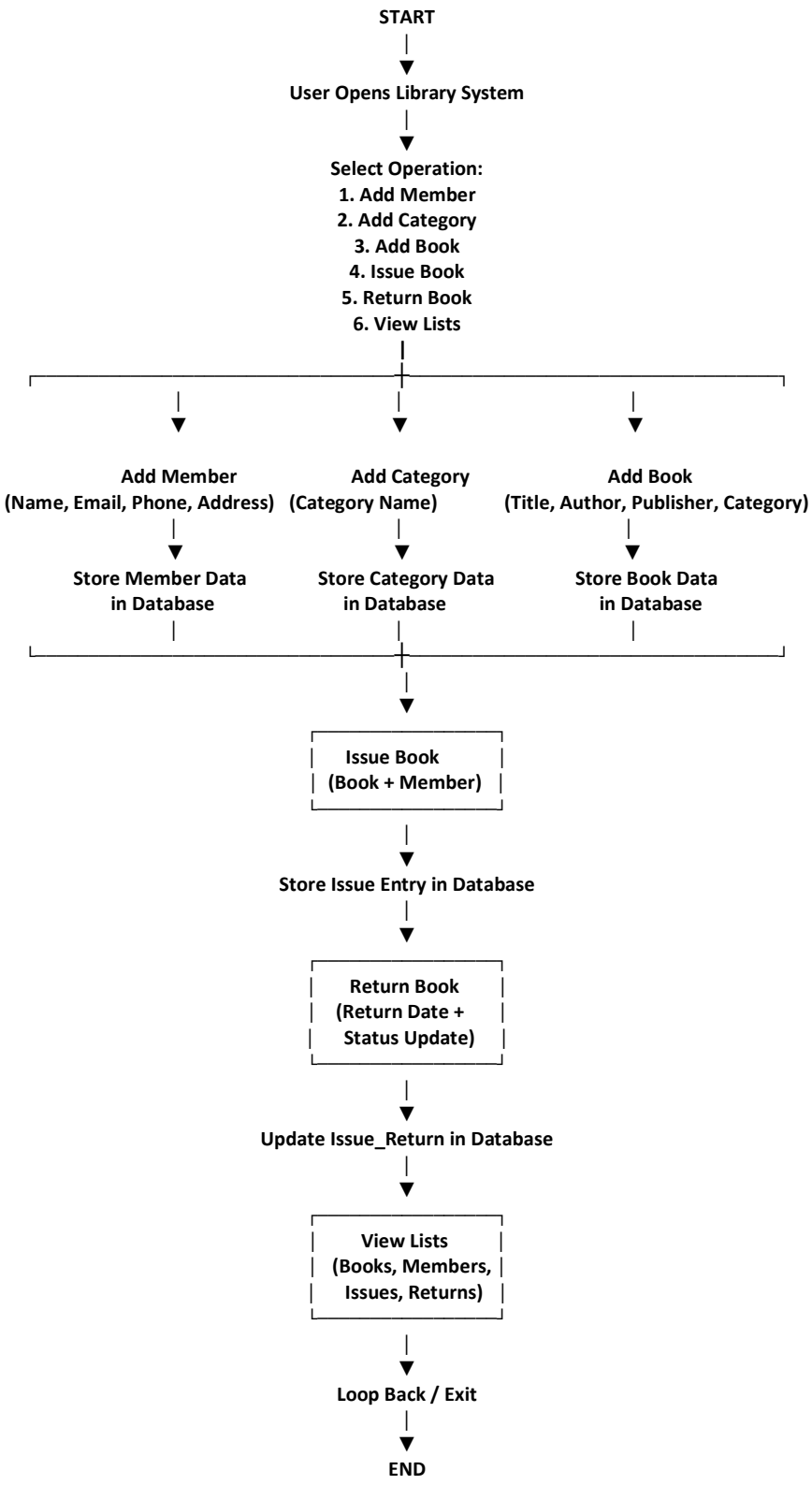
ONE LEVEL DFD



TWO LEVEL DFD



PROCESS LOGICAL DIAGRAM



PLATFORM USED

a) Hardware Requirements

- Processor: Intel i3 or higher
- RAM: Minimum 4GB
- Hard Disk: 500MB free space
- Operating System: Windows/Linux/Mac

b) Software Requirements

- JDK 8 or above
- MySQL Server
- IDE: Eclipse / IntelliJ IDEA / NetBeans
- JDBC Driver
- OS: Windows 10/11

FUTURE SCOPE

- Adding a **Graphical User Interface (GUI)** for better user experience.
- Implementing **online access** for remote library users.
- Adding **fine management** for overdue books.
- Generating **detailed reports** with analytics for library usage.
- Integrating **barcode or QR code scanning** for books.

BIBLIOGRAPHY

- Oracle Java Documentation
- MySQL Official Documentation
- JDBC API Guide
- Reference books on DBMS and Java Programming
- Online educational resources (GeeksforGeeks, TutorialsPoint, JavaTPoint)