2022 - 2024

# Library management system


Submitted By

NAME: ARKA DAS
ROLL: 002210503046

NAME: MD NAIMUR RAHAMAN
ROLL: 002210503040

MCA 1ST YEAR 2ND SEMESTER
JADAVPUR UNIVERSITY

# Library Management System

## Problem Description:

The Library Management System is a software application designed to streamline and automate the operations of a library. It helps librarians maintain the database of new books and the books that are borrowed by members along with their due dates. Libraries face several challenges in managing their resources efficiently and providing seamless services to their users. Manual processes for book management, borrower records, and administrative tasks often lead to inefficiencies, errors, and a lack of real-time information.

The basic purpose of using a library management system is to reduce the amount of human work involved. Automating various library operations makes the entire library process seamless for the librarians. It reduces the risk of paper work such as file lost, file damaged and time consuming.

## Problem Statement:

The problems known before having computerized system and databases:

1. **Inefficient Book Tracking**: Traditional paper-based systems make it difficult to track the status and location of books within the library. This can result in misplaced or lost books, delays in retrieving requested items, and inaccurate inventory records.
2. **Tedious Borrowing and Return Processes**: Manual handling of borrower registration, book issuance, and return procedures can be time-consuming and prone to errors. Long queues and delays in processing transactions can frustrate library users and affect their overall experience.
3. **Lack of Centralized Information**: Libraries often struggle with fragmented data stored across various systems or paper records. This leads to challenges in accessing up-to-date information about available books, borrower history, fines, and other administrative details.
4. **Limited Reporting and Analytics**: Generating reports and analyzing library usage patterns manually can be arduous and time-intensive. Library administrators require timely and accurate data insights to make informed decisions about resource allocation, purchasing decisions, and optimizing library services.

The Library Management System aims to address these challenges by providing a comprehensive and automated solution that simplifies library operations, enhances user experience, and improves overall efficiency. By leveraging modern technology, the system enables libraries to manage their collections, streamline borrowing processes, maintain accurate records, and gain valuable insights into library usage patterns.

By providing an effective solution to these problem areas, the Library Management System will empower libraries to deliver an enhanced experience to their users, optimize resource utilization, and streamline administrative tasks.

## System objectives:

The project vision is to come up with a library management system database application that does the jobs of the librarian like maintaining book records, maintaining user records, due dates, fines, etc. efficiently and in a quick time without errors. These objectives serve as guiding principles for the development and implementation of the system

1. **Efficient Book Management**: The system should provide efficient book tracking, including real-time availability status, accurate location information, and streamlined processes for adding, updating, and removing books from the library's collection.
2. **Simplified Borrowing and Returns**: The system should simplify the borrowing and return processes for library users, reducing waiting times, minimizing errors, and automating tasks such as book issuance, due date reminders, and fine calculations.
3. **Centralized and Accurate Data**: The system should ensure that library information, such as book catalog, borrower records, and administrative details, is stored in a centralized database, enabling easy access, accurate reporting, and efficient data management.
4. **User-Friendly Interface**: The system should provide an intuitive and user-friendly interface for both library staff and patrons. It should be easy to navigate, with clear instructions and visual cues to assist users in performing tasks and finding information.
5. **Security and Privacy**: The system should incorporate robust security measures to protect sensitive data, including user information, financial transactions, and administrative records. It should adhere to privacy regulations and ensure secure access control to prevent unauthorized access.
6. **Scalability and Flexibility**: The system should be designed to accommodate the evolving needs of the library, allowing for scalability as the collection and user base grow. It

should also be flexible enough to adapt to changing technologies, integration with other systems, and future enhancements.

7. **Reliability and Performance**: The system should be reliable, ensuring consistent availability and minimal downtime. It should also be designed for optimal performance, handling concurrent user requests, and providing a responsive experience even during peak usage periods.

## Functional requirements:

These requirements are outlines of what the system should be able to handle and how it should behave.

1. Staff login:
    - The system should allow staffs to create new accounts by collecting their personal information and contact details.
    - Staffs should be able to choose a unique username and password for authentication purposes.
    - Staffs will manage the whole system – such as adding new user or reader, adding new book in inventory, buying more copies of book etc.
2. Borrowing and Returns:
    - Staffs should be able to lend books by selecting them from the catalog.
    - The system should enforce borrowing limits and due dates.
    - Users should be able to return books, and the system should update the availability status and handle any associated fines or penalties.
3. User Management:
    - The system should provide administrative functionalities for managing user accounts, including the ability to add, update, and deactivate user profiles.
    - Library staff should have access to view and edit user details, including borrowing history, fines, and contact information.

## Non-Functional requirements:

Non-functional requirements focus on the qualities and characteristics of your library management system

1. Performance: The system should provide fast response times, ensuring that users can perform tasks efficiently without experiencing significant delays.
2. Usability: The system should have a user-friendly interface with intuitive navigation and clear instructions.
3. Security: User authentication and access control mechanisms should be in place to ensure that only authorized individuals can perform certain actions or access sensitive information.
4. Scalability: The system should be designed to handle increasing amounts of data, users, and transactions without significant degradation in performance.
5. Reliability: The system should be reliable, minimizing the occurrence of errors, crashes, or data inconsistencies.

# Database design:

A library management system requires a large database of users, book, transactions and a lot to be managed.

**Important parts of the database to be managed:**

1. **Staffs**: Records of staffs are to be managed. Staffs are main part of this whole application. They manage other parts of the database and thus they will have administrative rights to perform other database operations.
2. **Members**: Users are to be managed by Staffs. Date related to name, contact details etc. are to be stored. Staffs can add or remove users.
3. **Books**: Books are fundamental part of the application. This will hold data related to book line title, author name, ISBN, number of copies available etc. Staffs will manage books whereas users can view and borrow books. Borrowing of book will be processed by Stuffs. Stuffs can add or remove book and can also update book records.

## Relationship among the relations:

1. Relationship among Staff and Members
    a. Staffs keeps track of Members. This is a many-to-many relationship. There will be multiple Members and multiple stuffs. Multiple staffs can manage multiple Members.
2. Relationship among Staffs and Books
    a. Staffs manages Book records. This is also a many-to-many relationship.
3. Relationship among Books and Members
    a. Users can reserve books. Though reservation is processed by Staffs but for simplicity Members and books are directly in a one-to-many relationship.
    b. A Members can borrow multiple books and a book can be borrowed by multiple users. So thus it results in a one-to-many relationship.
4. Relationship among Staff and Authentication system:
    a. Each stuff must log-in to the system in order to use it. Every user will have their own Staff ID and password.
    b. Each stuff logs in through a single authentication system where as each authentication system logs in multiple stuffs. Thus it is a many-to-one relationship.

## ER - Diagram of the system:

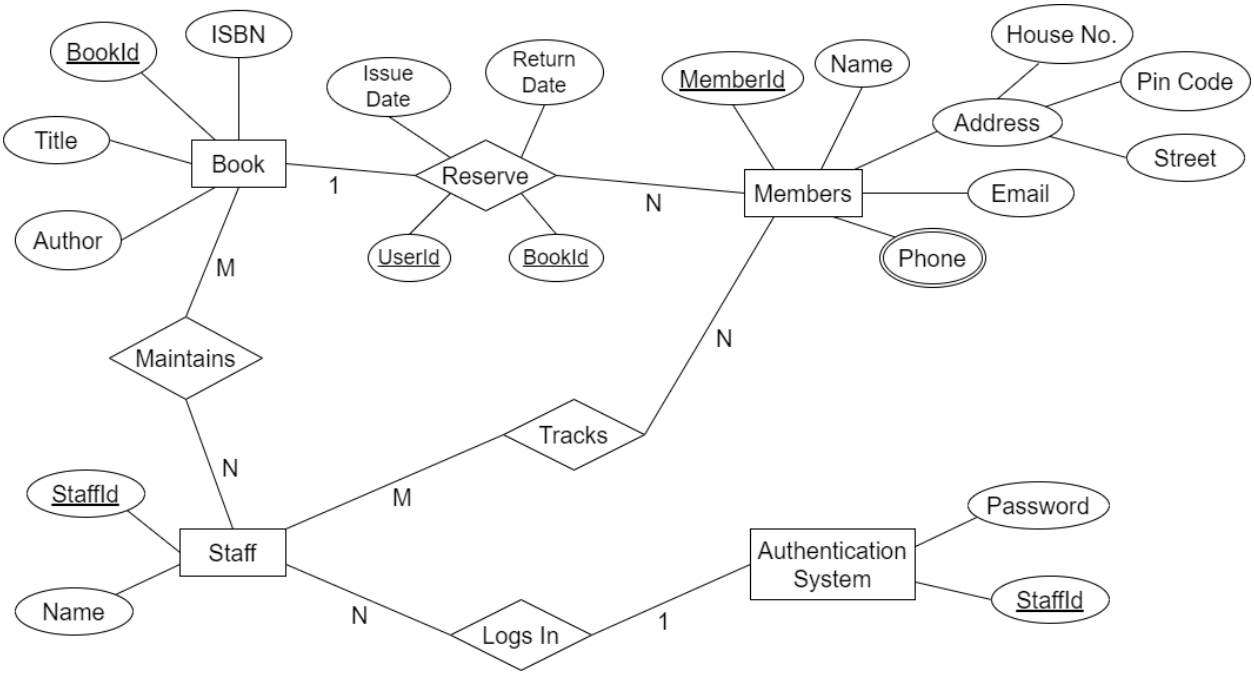
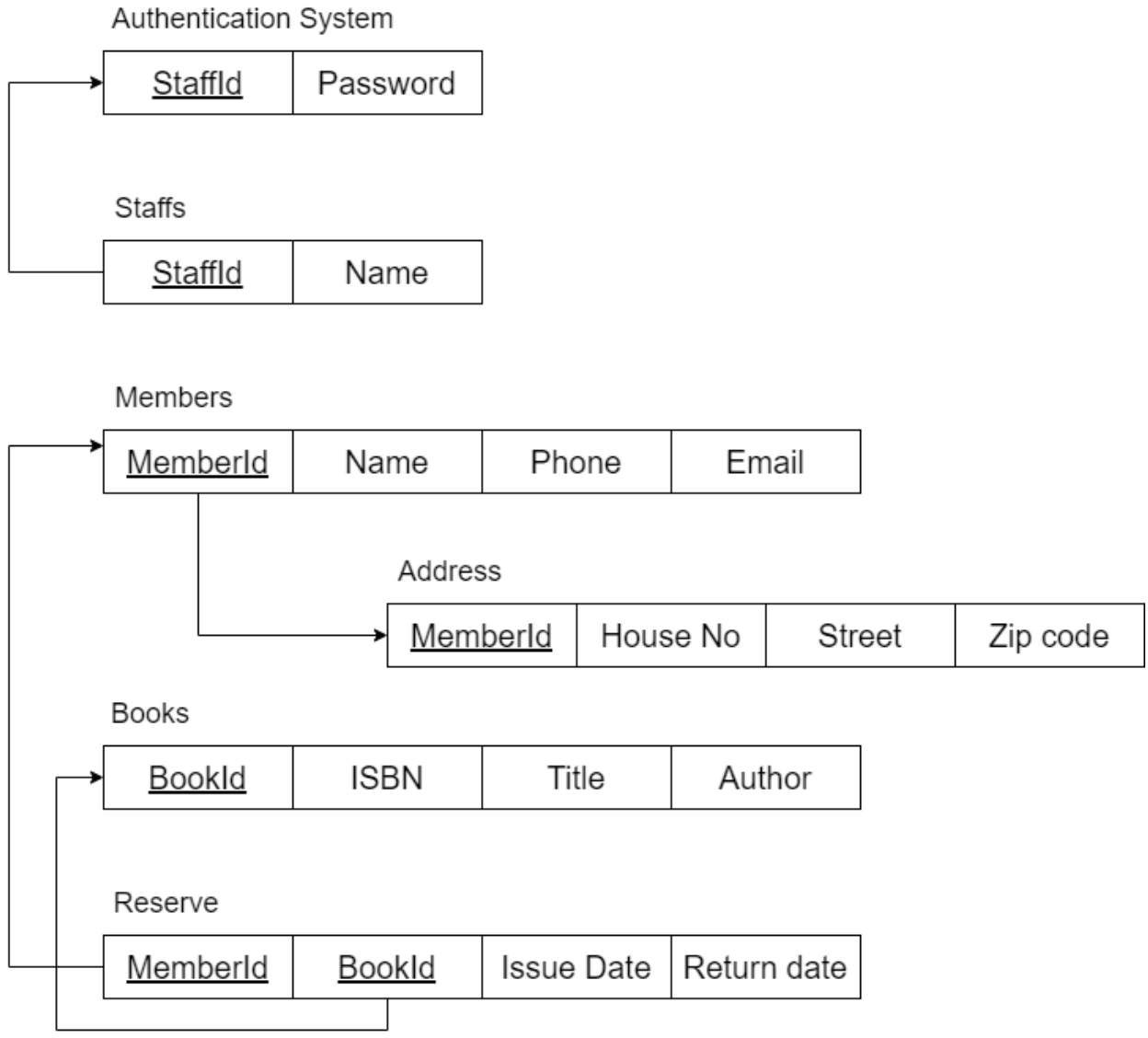



*Figure 1: ER-Diagram of the system*

The Entity-Relationship (ER) diagram above provides an overview of the database structure for our library management system. The diagram visually represents the entities, their attributes, and the relationships between them. This diagram serves as a blueprint for organizing and storing data in the system.

By visually representing these entities and their relationships, the ER diagram helps us understand how the data is structured in our library management system. It provides a foundation for designing efficient queries, ensuring data integrity, and facilitating seamless interactions between different components of the system.

## Database Schema design:




This is a simple schema diagram which gives an overview of the system. This shows the interconnection and dependency between the attributes of the relations.

We can now construct the relations / tables in the database using this above schema.

## Creating relations in database and inserting records:

Below, we provide a snapshot of the actual records and tables within the database of our library management system. This section aims to showcase the structure and content of the database, giving a glimpse of the real-world data that the system manages.

The tables listed below correspond to the entities identified in the Entity-Relationship (ER) diagram previously discussed. Each table contains rows that represent individual records or instances of the respective entity. The columns within each table represent the attributes or properties associated with the entity.

### Relation regarding Staff records:

| STAFFID | NAME |
|---|---|
| **S1** | Alice Johnson |
| **S2** | Benjamin Adams |
| **S3** | Chloe Thompson |
| **S4** | Daniel Wilson |
| **S5** | Emma Roberts |

### Relation regarding Member records:

| MEMBERID | NAME | PHONE NUMBER | EMAIL |
|---|---|---|---|
| **M1** | John Smith | 555-123-4567 | john@gmail.com |
| **M2** | Jane Doe | 555-987-6543 | jane@gmail.com |
| **M3** | David Lee | 555-555-5555 | david@gmail.com |
| **M4** | Emily Chen | 555-111-2222 | emily@gmail.com |
| **M5** | Mark Davis | 555-888-9999 | mark@gmail.com |

### Relation regarding Address records:

| MEMBERID | HOUSE NUMBER | STREET | ZIP CODE |
|---|---|---|---|
| **M1** | 123 | Main Street | 12345 |
| **M2** | 456 | Elm Avenue | 23456 |
| **M3** | 789 | Oak Street | 34567 |
| **M4** | 321 | Maple Lane | 45678 |
| **M5** | 654 | Pine Street | 56789 |

## Relation regarding Book records:

| BOOKID | ISBN | TITLE | AUTHOR |
|---|---|---|---|
| **B1** | 9780132350884 | Introduction to Algorithms | Thomas H. Cormen |
| **B2** | 9780262033848 | The Design of Design | Frederick P. Brooks Jr. |
| **B3** | 9780321558237 | Clean Code: A Handbook of Agile | Robert C. Martin |
| **B4** | 9781449355739 | Python Crash Course | Eric Matthes |
| **B5** | 9780596007126 | Head First Design Patterns | Eric Freeman |

## Relation regarding Transaction records:

| MEMBERID | BOOKID | ISSUEDATE | RETURNDATE |
|---|---|---|---|
| **M1** | **B3** | 2023-05-10 | 2023-05-17 |
| **M2** | **B5** | 2023-05-12 | 2023-05-19 |
| **M3** | **B2** | 2023-05-15 | 2023-05-22 |
| **M4** | **B1** | 2023-05-18 | 2023-05-25 |
| **M5** | **B4** | 2023-05-20 | 2023-05-27 |