Library Management System Project Report

# 1. Project Overview

The Library Management System is a software application developed to manage the operations of a library. It provides functionalities for adding and removing users and books, borrowing and returning books, and searching for books by title or author. The system is implemented using Java and demonstrates core concepts of object-oriented programming and data structures.

# 2. Objectives

• To design and implement a system that efficiently manages the library's resources.  
• To provide an easy-to-use interface for library administrators to manage users and books.  
• To enable users to borrow and return books seamlessly.  
• To allow quick search and retrieval of book information.

# 3. Tools and Technologies Used

• Programming Language: Java  
• IDE: Visual Studio Code  
• Version Control: Git

# 4. Project Implementation

The project consists of four main classes: Main, Library, User, and Book.

## Main Class

• Function: Acts as the entry point of the application.  
• Responsibilities: Displays the menu, accepts user input, and interacts with the Library class to perform various operations.

## Library Class

• Function: Manages the collection of books and users.  
• Responsibilities: Adding/removing users and books, handling book borrow/return operations, and searching books.

## User Class

• Function: Represents a user in the library system.  
• Responsibilities: Stores user information and manages the list of borrowed books.

## Book Class

• Function: Represents a book in the library system.  
• Responsibilities: Stores book information and maintains the book's availability status.

# 5. Features

• Add User: Allows the administrator to add a new user to the system.  
• Remove User: Allows the administrator to remove an existing user from the system.  
• Add Book: Allows the administrator to add a new book to the library.  
• Remove Book: Allows the administrator to remove a book from the library.  
• Borrow Book: Enables a user to borrow a book, updating the book's availability status.  
• Return Book: Allows a user to return a borrowed book, updating the book's availability status.  
• Search Book by Title: Allows users to search for books by their title.  
• Search Book by Author: Allows users to search for books by the author's name.

# 6. Data Structures Used

## HashMap

• Purpose: Store and manage users and books efficiently.  
• Usage:   
 - Library class: Two HashMap instances for storing users and books by their unique identifiers (user ID and ISBN).

## ArrayList

• Purpose: Maintain a list of borrowed books for each user.  
• Usage:   
 - User class: An ArrayList for storing books borrowed by the user.

## Stream API

• Purpose: Efficiently search and filter books based on criteria.  
• Usage:   
 - Library class: Used in methods to search for books by title and author.

# 7. Challenges and Solutions

• Challenge: Ensuring efficient search and retrieval of books.  
 - Solution: Utilized HashMap for O(1) time complexity on average for add/remove/search operations and used the Stream API for efficient filtering.  
  
• Challenge: Managing the state of borrowed books.  
 - Solution: Implemented methods in the User class to handle borrowing and returning of books, updating their availability status.

# 8. Future Enhancements

• User Authentication: Implement user login and authentication for added security.  
• GUI: Develop a graphical user interface to enhance user experience.  
• Notifications: Add a notification system to remind users of due dates for borrowed books.  
• Database Integration: Integrate with a database to handle larger volumes of data more efficiently.

# 9. Conclusion

The Library Management System project provided an excellent opportunity to apply Java programming skills and understand the use of data structures in real-world applications. The project demonstrates efficient management of library operations and can be expanded with additional features and improvements in the future.