

GALGOTIAS UNIVERSITY

Sector 17A Yamuna Expressway, Greater Noida,
Gautam Buddha Nagar, Uttar Pradesh 203201



GALGOTIAS
UNIVERSITY

Academic Year: 2024-2025

School of Computer Science and Engineering

(AI & ML)

Project Report

"LIBRARY MANAGEMENT SYSTEM"

Semester: 2

Submitted By

Aman Kumar Gupta

Manish

Amrit Shekhar

Sumit Pandey

TEAM : (MAAS CODER)

(24SCSE1180579)

(24SCSE1180560)

(24SCSE1180302)

(24SCSE1180506)

1. Core Feature Implementation

Description

This section describes the main features of the Library Management System that were implemented successfully.

Core Features

- **Book Management** : Add, update, delete, and search books.
- **Student Management** : Register students and assign IDs.
- **Issue/Return Books** : Track which student has borrowed which book, and when it was returned.
- **Fine Calculation** : Calculates fines for late returns automatically.
- **Search Functionality** : Allows searching books by name, author, or ISBN.
- **Admin Panel** : For managing books and users.
- **Login Authentication** : Secure login for admin/staff.

2. Error Handling and Robustness

Description

This section explains how the system handles errors and ensures stability.

Implementation

- **Try-Catch Blocks** used in Java to prevent crashes.
- Validation checks to handle:
 - Invalid input (e.g., empty book name, wrong ISBN format)
 - Non-existent user ID or book ID
 - Book already issued/returned
- Error messages are displayed to the user with helpful guidance.
- **Robust file/database operations** : File not found, connection errors, etc., are caught and logged.

3. Integration of Components

Description

Details how different parts of the system work together.

Examples

- **Login System** connects to User Database for authentication.
- **Book Issue Module** interacts with both **Book Records** and **Student Data**.
- **Fine System** integrates with Date and Time module to calculate overdue charges.
- **GUI Interface** (if present) is connected to the backend database.
- Frontend, Backend, and Database are all connected for seamless operation.

4. Event Handling and Processing

Description

This section explains how the system responds to user actions.

Examples

- **Clicking “Issue Book” triggers a check on**
 - Book availability
 - Student’s borrowing limit
 - **Returning a book triggers**
 - Due date check
 - Fine calculation if late
 - Button clicks and form submissions are handled through event listeners or actionPerformed() methods
- Java
- Real-time updates after book issue/return.

5. Data Validation

Description

Explains how data inputs are checked before being processed or stored.

Validation Implemented

- **Student Registration** : Name not empty, valid roll number.
- **Book Entry** : Valid ISBN, title, and quantity > 0.
- **Issue Book** : Book must be available, student must exist.
- Prevents duplicate book IDs or user IDs.
- UI form fields have constraints (e.g., numeric only, required fields).

6. Code Quality and Innovative Features

Code Quality

- Code is modular, with functions/classes separated by logic.
- Proper indentation and naming conventions.
- Comments added to explain logic.

Innovative Features (if added)

- **Search with Filters** (author, title, year).
- **Email Notification** for due date reminder (if applicable).
- **Dark Mode** for UI.
- **Dashboard Analytics** showing issued books count, most borrowed book, etc.

7.Project Documentation

Contents of Final Report

- Title Page
- Abstract
- Objective
- Tools & Technologies Used
- System Design (Flowchart or UML)
- Implementation Screenshots
- Code Snippets
- Testing and Validation Results
- Future Scope
- Conclusion
- References

TABLE OF CONTENTS

1. DECLARATION.....	4
2. ACKNOWLEDGEMENT.....	5
3. SUMMARY.....	6
4. OBJECTIVE AND PURPOSE....	7
4.1. Purpose.....	7
5	
4.2. Objective.....	7
4.3. Project Scope.....	8
4.4. Advanced Technology.....	8
5. INTRODUCTION.....	9
5.1. Add Book.....	9
5.2. Home.....	9
5.3. Issue Book.....	9
5.4. Return Book.....	9
5.5. Student Registration.....	10
6. HARDWARE AND SOFTWARE.....	11
6.1. Minimum Hardware Required.....	11
6. 2.Minimum Software Required.....	11
6.3. Operating System.....	11
6.4. Other Non Function Requirement.....	11
7. PROGRAMMING LANGUAGE AND DEVELOPMENT TOOLS.....	12
7.1. Java.....	12
7.2. Servral New Features of java.....	13
7.3. Advantages of Java.....	14
7. 4.Disadvantage of Java.....	14
7.5DATABASE:MYSQL QUERRY BROWSER.....	14

7.6. Introduction to Mysql query browser.....	15
8. METHODOLOGY.....	16
8.1. Home Page.....	16
8.2. Login Page.....	16
8.3. Student Registration.....	17
8.4. Add Book.....	17
8.5. Issue Book.....	18
8.6. Return Book.....	18
8.7. Logout form.....	19
9. REFERENCES.....	

DECLARATION

We, The Team MAAS CODER, here by certify that the project entitled "LIBRARY MANAGEMENT SYSTEM" has been undertaken and successfully completed as part of our academic/work requirements for the award of degree of Btech (computer science and engineering Specialization AI&ML).Submitted in the department of computer science and engineering at Galgotias University, Greater Noida.

Project Duration: [01 May 2025] to [10 June 2025]

Team Name: MAAS CODER

Team Members	Admission Number
Amrit Shekhar	24SCSE1180302
Sumit Pandey	24SCSE1180506
Aman Kumar Gupta	24SCSE1180579
Manish	24SCSE1180560

Institution/Organization:

Galgotias University / Guvi Team

ACKNOWLEDGEMENT

We, the team MAAS CODER, would like to express our sincere gratitude to all those who supported us in the successful completion of our project entitled "Library Management System."

First and foremost, we extend our heartfelt thanks to [Prabhakaran M / Guvi Team], whose valuable guidance, encouragement, and constructive feedback were crucial at every stage of this project. Their support helped us overcome various challenges and kept us focused on our goals.

We would also like to thank the faculty and staff of Galgotias University for providing us with the resources and learning environment necessary for the development of this project.

Special thanks to our friends and classmates for their constant support, and for being a source of motivation throughout the journey.

Lastly, we express our deepest appreciation to our families for their unwavering support, patience, and encouragement throughout the course of this work.

This project would not have been possible without the collective effort and contribution of everyone involved.

Team Members

Aman Kumar Gupta

Manish

Amrit Shekhar

Sumit Pandey

SUMMARY

This is the project report on "LIBRARY MANAGEMENT SYSTEM". During the making/ developing of this project we explored new ideas and functionalities behind the working of notepad.

This project is output of our planning, schedule, programming skill and the hard work. This report reflects our steps taken at various levels of programming skill, planning and schedule.

We have learnt during this project and liked the improvement in our testing skill and deep concepts related to this kind of project.

Our project is "LIBRARY MANAGEMENT SYSTEM".

Library Management System can be used by education institutes to maintain the record of library easily.

Achieving this objective is difficult using a manual system, as the information is scattered, may be redundant, and collecting relevant information can be very time-consuming. This project aims to solve all these problems effectively.

PURPOSE:

5. The Library Management System is an application designed to:
6. Allow the Administrator to add new books to the library database.
7. Enable the Administrator to view the list of all books, available books, and issued books.
8. Allow the Administrator to update student/member information and delete book records from the database.
9. Add new members or students to the database.
10. Enable the Administrator to issue books to students/members, update their information, and delete their records from the database.
11. Provide a student login section where students can view and update their personal details.
12. Allow students to check the books issued to them.

OBJECTIVE:

13. The key objectives of this project are:
14. Efficient handling of all books within the system.
15. Simplified registration of new books.
16. Easy searching of books and issuing them to students/members.
17. Convenient updating and deletion of book records.
18. Effective registration of new members/students.
19. Easy search, update, and deletion of member details.
20. Generation of results in a user-friendly and interactive manner.

PROJECT SCOPE

The scope of this project is extensive. The primary objective of this document is to analyze the current management system and define the scope and objectives of the proposed automated Library Management System. It also explores alternative solutions to various existing challenges. This document serves as a guide for developing a prototype of the proposed system and aims to improve efficiency, accuracy, and ease of use.

ADVANCED TECHNOLOGY (Before Automation):

Before automation, all tasks were performed manually, resulting in a high probability of errors due to the extensive paperwork involved. The major problems encountered with the manual system included:

- **Manual Registration:** All registrations were done by hand, requiring separate registers, which was tedious and inefficient.
- **Difficulty in Retrieval:** Searching for any data was time-consuming.
- **Difficulty in Updating:** Making updates to records was a lengthy and error-prone process.
- **Difficulty in Verification:** Verifying information (e.g., account details) took a considerable amount of time.
- **Insecurity:** Physical books or registers were less secure and prone to damage or loss.
- **Data Redundancy:** A large amount of data was duplicated and often inconsistent.

INTRODUCTION

As mentioned earlier, the project is all about storing and managing the data related to books and student. Also the project helps admin to view the details at any time and at any place. About the working of our project, we plan to include the few basic options which deal with recording, retrieving, registration or updating.

There are two sections

- Administrator Login
- Student Login

ADD BOOKS

Administrator can add new books to the database at any time. When new book is added into the database a Unique Book Id is generated for that book.

ISSUED BOOKS

Administraator can also Vies the List of those books which are currently not available i.e. which are currently issued by students/members.

RETURN BOOKS

The Return Book feature allows users (borrowers) to return a previously borrowed book back to the library. When a book is returned, the system updates the book's status to indicate it is now available for others to borrow.

HOME PAGE

The Home Page serves as the central navigation hub for the Library Management System. users can easily access all major features of the system. The home page provides a user-friendly interface that displays all available options clearly, allowing users (like librarians or admin) to quickly perform any library-related tasks.

Student Registration

Student Registration in a Library Management System is the process of collecting and storing student information into the system so that they can access library services such as borrowing books, checking book availability, and receiving notifications. It involves inputting student details like name, ID, course, and contact information into the system database for identification and management purposes.

Key Features

Students can register themselves using their following details :-

- Student ID – Unique ID
- Full Name
- Course/Department
- Year/Semester

HARDWARE AND SOFTWARE

Minimum Hardware Requirements:

- Processor: P3 500 MHz or above.
- Memory: 256 MB RAM.
- Hard Disk: 100 MB free hard disk space.
- Monitor: CRT or LCD monitor.

2.2. Minimum Software Requirements:

- Operating system with Java installed.
- MySQL Query Browser.

2.3. Operating System:

Microsoft Windows XP Service Pack 2 or higher versions.

2.4. Other Nonfunctional Requirements:

2.4.1 Performance Requirements

- The system should provide accurate information as per user requirements.
- Data retrieval should be fast.
- The system should be efficient, precise, and easy to maintain.

2.4.2 Safety Requirements

- The database should be backed up routinely to prevent data loss.

2.4.3. Security Requirements

- System should be capable of handling all types of malicious attacks made by intruders. System
- should not provide access to any unauthorized

PROGRAMMING LANGUAGE AND DEVELOPMENT TOOLS 3.1 JAVA

JAVA is a programming language originally developed by James Gosling at Sun Microsystems (which has since merged into Oracle Corporation) and released in 1995 as a core component of Sun Microsystems' Java platform. The language derives much of its syntax from C and C++ but has a simpler object model and fewer low-level facilities. Java applications are typically compiled to byte code (class file) that can run on any Java Machine (JVM) regardless of computer architecture. Java is a general-purpose, concurrent, class-based, object-oriented language that is specifically designed to have as few implementation dependencies as possible. It is intended to let application developers "write once, run anywhere" (WORA), meaning that code that runs on one platform does not need to be recompiled to run on another. Java is as of 2012 one of the most popular programming language in use, particularly for client-Query Browser web applications, with a reported 10 million users. Object-oriented programming is at the core of Java. In fact, all Java programs are objectoriented—this isn't an option the way that it is in C++, for example. OOP is so integral to Java that you must understand its basic principles before you can write even simple Java programs. To manage increasing complexity, the second approach, called object-oriented programming, was conceived. Object-oriented programming organizes a program around its data (that is, objects) and a set of well-defined interfaces to that data. An object-oriented program can be characterized as data controlling access to code.

3.2 SIGNIFICANT NEW FEATURES OF JAVA

3.2.1 Platform Independence

Many types of computers and operating systems are in use throughout the world — and many are connected to the Internet. For programs to be dynamically downloaded to all the various types of platforms connected to the Internet, some means of generating portable executable code is needed. As you will soon see, the same mechanism that helps ensure security also helps create portability. Indeed, Java’s solution to these two problems is both elegant and efficient.

3.2.2 Object Oriented

Although influenced by its predecessors, Java was not designed to be source-code compatible with any other language. This allowed the Java team the freedom to design with a blank slate. One outcome of this was a clean, usable, pragmatic approach to objects. Borrowing liberally from many seminal object-software environments of the last few decades, Java manages to strike a balance between the purist’s “everything is an object” paradigm and the pragmatist’s “stay out of my way” model. The object model in Java is simple and easy to extend, while simple types, such as integers, are kept as high-performance nonobjects.

3.2.3 Multithreaded

Java was designed to meet the real-world requirement of creating interactive, networked programs. To accomplish this, Java supports multithreaded programming, which allows you to write programs that do many things simultaneously. The Java run-time system comes with an elegant yet sophisticated solution for multiprocess synchronization that enables you to construct smoothly running interactive systems. Java’s easy-to-use approach to multithreading allows you to think about the specific behavior of your program, not the multitasking subsystem.

Advantages of Java:

- Platform Independent
- Object-Oriented
- Simple and Easy to Learn
- Secure
- Robust

Disadvantage of java

- Performance Overhead
- Verbose Syntax
- Lack of Low-Level Programming Features
- GUI Development is Cumbersome
- Memory Management Issues
- No Support for Unsigned Integers •High CPU and Memory Usage

3.6 DATABASE: MYSQL Query Browser

3.6.1 What is MYSQL?

MYSQL stands for Structural Query Language.

MYSQL lets you access and manipulate databases.

MYSQL is an ANSI (American National Standards Institute) standard.

3.6.2 What can MYSQL do?

MYSQL can execute queries against a database.

MYSQL can retrieve data from a database.

MYSQL can insert records in a database.

MYSQL can update records in a database.

MYSQL can delete records from a database.

MYSQL can create new databases.

MYSQL can create new tables in a database.

MYSQL can create stored procedures in a database.

MYSQL can create views in a database.

MYSQL can set permissions on tables, procedures, and views.

3.7 INTRODUCTION TO MYSQL QUERY BROWSER:

MYSQL Query Browser is an MYSQL-compliant RDBMS. MYSQL-compliant means it uses the ANSI (American National Standard Institute) version of Structured Query Language or 'MYSQL'. Structured Query Language is a command that allows us to modify or retrieve information from the database. It works on the client Query Browser environment. Client Query Browser means that MYSQL Query Browser is designed to store data in the central location (the Query Browser) and deliver it on demand to numerous other locations.

Microsoft MYSQL Query Browser is a relational database management system (RDBMS) produced by Microsoft. Its primary query language is Transact-MYSQL, an implementation of the ANSI/ISO standard Structured Query Language (MYSQL). An RDBMS stores and retrieves data for multiple sources.

At the center of any MYSQL relational database implementation is the MYSQL Query Browser. This is the software platform (usually multi-user) that manages the database itself. It has much in common with traditional Query Browser implementations in that the user model consists of an administrative 'super user', various user groups, and users all with authentication.

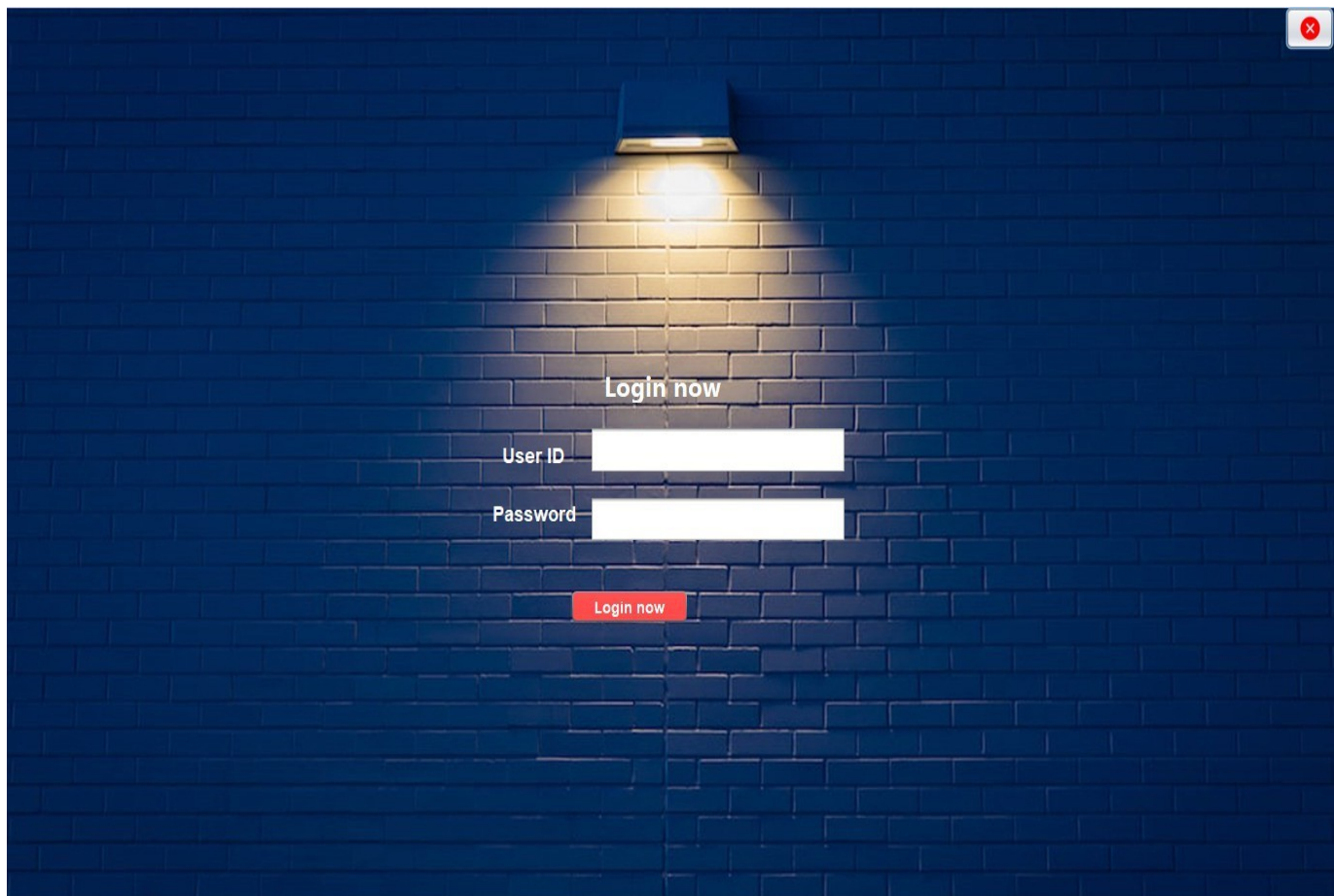
Microsoft MYSQL Query Browser and the Oracle family are the two best-known commercial MYSQL implementations.

METHODOLOGY

HOME PAGE



LOGIN PAGE



STUDENT REGISTRATION

 Student Registration

Student ID

Student Name

Course Name

Branch Name

Semester

Save





Student Registration

add book

issue books

reutrn book

 Logout

ADD BOOK

 Add Book Details

Book ID

Book Name

Publisher

Price

Publisher Year

Save





Student Registration

add book

issue books

reutrn book

 Logout

ISSUE BOOK

**Issue Book**

Book ID

Search

Student ID

Book Name

Issue Date

Due Date

Issue Book





Student Registration


add book

issue books

reutrn book

LOG OUT

RETURN BOOK

Return Book

Book ID

Student ID

Student Name

Book Name

Issue Date

Due Date

Search


Return

Student Registration

add book

issue books

reutrn book

LOG OUT

LOGOUT

Student Registration

add book

issue books

reutrn book

LOG OUT

Logut

Are you really Logout ?

Yes

No

References

Books

- Beginning Java Objects: From Concepts to Code – by Jacquie Barker
- The Complete Reference Java (McGrawHill; Herbert Schildt – reprint 2008) • Gary Cornell and Cay S. Horstmann, Core Java, second ed., SunSoft Press, 1997.