

Library Management System

Problem Statement

Develop a Library Management System from a librarian's perspective with a graphical user interface (GUI). The system should manage books, student records, and transaction logs through CSV files while providing functionalities for book issuance, returns, and inventory management.

System Requirements

1. Data Storage

The system must maintain three CSV files:

1. **books.csv**: Store book information
2. **students.csv**: Store registered student information
3. **logs.csv**: Maintain transaction records of book issues and returns

2. Required Classes

2.1 Book Class

Required Methods:

- Constructor to initialize book attributes
- Getter/setter methods for attributes
- Method to convert book data to CSV format

2.2 Student Class

Required Methods:

- Constructor to initialize student attributes
- Methods to add/remove borrowed books
- Method to convert student data to CSV format
- Method to check borrowing eligibility

2.3 Librarian Class

Core class handling all library operations. **Required Methods:**

- **check_stock()**: Display available books
- **search_book(query)**: Search books by title/author/ID
- **search_student(query)**: Search student records
- **issue_book(book_id, student_id)**: Process book issuance
- **return_book(book_id, student_id)**: Handle book returns
- **calculate_penalty(issue_date)**: Calculate late fees
- **load_data()**: Load data from CSV files
- **save_data()**: Save updates to CSV files

- `update_logs(transaction_type, book_id, student_id)`: Record transactions

3. GUI Requirements

3.1 Main Window

- Title bar showing "Library Management System"
- Clean, professional interface
- Proper error handling and user feedback

3.2 Required Components

1. **Book Management Section:**
 - Search bar for books
 - Button to check available stock
 - Display area for search results/stock information
2. **Student Management Section:**
 - Search bar for students
 - Display area for student information
3. **Transaction Section:**
 - Book issue form/button
 - Book return form/button
 - Due date display
 - Penalty calculation display

4. Functional Requirements

4.1 Book Search and Display

- Implement real-time or button-triggered search
- Display book details: ID, Title, Author, Available Copies
- Show appropriate message for no results

4.2 Student Verification

- Verify student registration before issue
- Display student borrowing history
- Show current borrowed books

4.3 Book Issuance

- Check book availability
- Verify student eligibility
- Update available copies
- Record transaction in logs
- Generate due date

4.4 Book Return

- Verify book belongs to returning student
- Calculate penalty if returned late

- Update available copies
- Record return in logs

4.5 Penalty Calculation

- Base penalty rate: \$1 per day
- Maximum penalty cap: \$50
- Grace period: 2 days
- Weekend exemption (optional feature)