## **Introduction**

This project consists of two parts:

1. **Library Management System**: A console-based application that allows librarians to manage books, patrons, and transactions effectively.
2. **BiblioConnect**: An integrated social platform built on top of the Library Management System that provides users with access to library resources while facilitating social interactions among patrons.

## **System Requirements**

* Java Development Kit (JDK) 8 or higher
* MySQL Database Server
  + User Interface
  + --------------
  + BiblioConnect features a console-based user interface that guides users through various operations:
  + 1. Login Menu:
  + 1. Login
  + 2. Exit
  + 2. Main Menu:
  + 1. Book Services
  + 2. Borrowing Services
  + 3. User Management
  + 4. Search
  + 5. Reports
  + 6. Social Media
  + 7. Logout
  + 3. Book Services Menu:
  + 1. List all books
  + 2. Add a book (Librarians only)
  + 3. Remove a book (Librarians only)
  + 4. Return to Main Menu
  + 4. Borrowing Services Menu:
  + 1. Borrow a book
  + 2. Return a book
  + 3. Reserve a book
  + 4. View my reservations
  + 5. View borrowing history
  + 6. View my overdue books
  + 7. View all overdue books (Librarians only)
  + 8. Return to Main Menu
  + 5. User Management Menu:
  + 1. Update my profile
  + 2. Register new user (Librarians only)
  + 3. List all users (Librarians only)
  + 4. Remove user (Librarians only)
  + 5. Return to Main Menu
  + 6. Search Menu:
  + 1. Search by title
  + 2. Search by author
  + 3. Search by ISBN
  + 4. Return to Main Menu
  + 7. Reports Menu (Librarians and Faculty only):
  + 1. View database content report
  + 2. Return to Main Menu
  + 8. Social Media Menu:
  + (Placeholder for future features)
  + 1. Return to Main Menu
  + Menu Navigation:
  + - Users navigate through menus by entering the number corresponding to their desired action.
  + - Invalid inputs are handled with appropriate error messages, prompting the user to try again.
  + - Certain menu options are only available to users with specific roles (e.g., Librarian).
  + - Users can always return to the previous menu or main menu from any sub-menu.
  + User Prompts:
  + - The system provides clear prompts for each action, guiding the user through the process.
  + - For actions like borrowing or returning books, users are prompted to enter relevant information (e.g., book ID).
  + - Confirmation messages are displayed after successful actions.
  + - Error messages are shown when an action cannot be completed, explaining the reason.
  + Role-Based Access:
  + - The menu options displayed to a user depend on their role (Student, Faculty, or Librarian).
  + - Librarians have access to all features, including administrative functions.
  + - Students and Faculty have access to basic borrowing and searching features.
  + - The system checks user permissions before executing role-specific actions.
  + Session Management:
  + - Users remain logged in and at the main menu until they choose to logout.
  + - Logging out returns the user to the login menu.
  + - The current user's information is maintained throughout their session for personalized interactions.
  + File Descriptions
  + -----------------
  + 1. BiblioConnectApplication.java
  + Main class that initializes and starts the application.
  + 2. LibraryManagementSystem.java
  + Interface defining the core structure of the library management system.
  + 3. UserManagementService.java
  + Interface outlining user management operations like registration and authentication.
  + 4. LibraryManagementService.java
  + Interface defining library operations such as book management and borrowing services.
  + 5. LibraryManagementImpl.java
  + Implements LibraryManagementSystem, coordinating UserManagementService and LibraryManagementService.
  + 6. UserManagementServiceImpl.java
  + Implements UserManagementService, handling user-related database operations.
  + 7. LibraryManagementServiceImpl.java
  + Implements LibraryManagementService, managing book and borrowing related operations.
  + 8. User.java
  + Represents a user in the system, storing personal information and role.
  + 9. UserRole.java
  + Enum defining user roles (STUDENT, FACULTY, LIBRARIAN) with specific borrowing privileges.
  + 10. Book.java
  + Abstract class representing a book, serving as a base for specific book types.
  + 11. PhysicalBook.java
  + Extends Book, representing a physical book with location information.
  + 12. EBook.java
  + Extends Book, representing an electronic book with format and download information.
  + 13. AudioBook.java
  + Extends Book, representing an audiobook with narrator and duration details.
  + 14. BorrowingRecord.java
  + Represents a borrowing transaction, tracking dates and reservation status.
  + 15. DatabaseConnection.java
  + Manages database connections and provides methods for database operations.
  + 16. BookFactory.java
  + Factory class for creating appropriate Book subclass instances.
  + 17. DataInitializer.java
  + Utility class for initializing the database with sample data.
  + 18. LibraryUI.java
  + Manages the user interface, handling user inputs and displaying information.
  + MySQL Database Setup
  + --------------------
  + To set up the MySQL database for BiblioConnect, follow these steps:
  + 1. Install MySQL:
  + If you haven't already, download and install MySQL from the official website:
  + https://dev.mysql.com/downloads/mysql/
  + 2. Create the Database:
  + Open MySQL command line client or a tool like MySQL Workbench and run the following commands:
  + CREATE DATABASE project\_db;
  + CREATE USER 'project\_user'@'localhost' IDENTIFIED BY 'Luna123!';
  + GRANT ALL PRIVILEGES ON project\_db.\* TO 'project\_user'@'localhost';
  + FLUSH PRIVILEGES;
  + 3. Create Tables:
  + Use the following SQL script to create the necessary tables:
  + USE project\_db;
  + CREATE TABLE Users (
  + userId VARCHAR(6) PRIMARY KEY,
  + name VARCHAR(100) NOT NULL,
  + email VARCHAR(100) UNIQUE NOT NULL,
  + username VARCHAR(50) UNIQUE NOT NULL,
  + password VARCHAR(255) NOT NULL,
  + role ENUM('STUDENT', 'FACULTY', 'LIBRARIAN') NOT NULL
  + );
  + CREATE TABLE Books (
  + bookId VARCHAR(6) PRIMARY KEY,
  + title VARCHAR(255) NOT NULL,
  + author VARCHAR(100) NOT NULL,
  + isbn VARCHAR(13) UNIQUE NOT NULL,
  + bookType ENUM('Physical', 'EBook', 'AudioBook') NOT NULL,
  + location VARCHAR(50),
  + format VARCHAR(50),
  + fileSize INT,
  + downloadLink VARCHAR(255),
  + narrator VARCHAR(100),
  + duration INT,
  + isAvailable BOOLEAN NOT NULL DEFAULT TRUE
  + );
  + CREATE TABLE BorrowingRecords (
  + recordId INT AUTO\_INCREMENT PRIMARY KEY,
  + userId VARCHAR(6),
  + bookId VARCHAR(6),
  + borrowDate DATETIME NOT NULL,
  + returnDate DATETIME,
  + isReservation BOOLEAN NOT NULL DEFAULT FALSE,
  + dueDate DATETIME NOT NULL,
  + FOREIGN KEY (userId) REFERENCES Users(userId),
  + FOREIGN KEY (bookId) REFERENCES Books(bookId)
  + );
  + Table Schema Explanation:
  + 1. Users Table:
  + - userId: A unique 6-character identifier for each user.
  + - name: The full name of the user.
  + - email: The user's email address (unique).
  + - username: The user's chosen username (unique).
  + - password: The user's password (should be hashed in a real-world scenario).
  + - role: The user's role in the system (STUDENT, FACULTY, or LIBRARIAN).
  + 2. Books Table:
  + - bookId: A unique 6-character identifier for each book.
  + - title: The title of the book.
  + - author: The author of the book.
  + - isbn: The International Standard Book Number (unique).
  + - bookType: The type of book (Physical, EBook, or AudioBook).
  + - location: The physical location of the book (for Physical books).
  + - format: The file format (for EBooks and AudioBooks).
  + - fileSize: The size of the file in KB (for EBooks and AudioBooks).
  + - downloadLink: The URL to download the book (for EBooks and AudioBooks).
  + - narrator: The name of the narrator (for AudioBooks).
  + - duration: The duration of the audiobook in minutes (for AudioBooks).
  + - isAvailable: A boolean indicating if the book is currently available for borrowing.
  + 3. BorrowingRecords Table:
  + - recordId: An auto-incrementing unique identifier for each borrowing record.
  + - userId: The ID of the user who borrowed the book (foreign key to Users table).
  + - bookId: The ID of the borrowed book (foreign key to Books table).
  + - borrowDate: The date and time when the book was borrowed.
  + - returnDate: The date and time when the book was returned (NULL if not yet returned).
  + - isReservation: A boolean indicating if this is a reservation (TRUE) or a current borrowing (FALSE).
  + - dueDate: The date by which the book should be returned.
  + 4. Configure Database Connection:
  + Update the DatabaseConnection.java file with the following details:
  + private static final String URL = "jdbc:mysql://localhost:3306/project\_db";
  + private static final String USER = "project\_user";
  + private static final String PASSWORD = "Luna123!";
  + 5. Initialize Data:
  + When you run the application for the first time, it will use the DataInitializer class to populate the database with sample data.
  + This includes creating some initial users, books, and borrowing records.
  + Note: Ensure that your MySQL server is running before starting the BiblioConnect application. If you encounter any connection issues, verify that the
  + server is accessible and that the user credentials are correct.
  + Security Note: The database credentials provided here are for demonstration purposes only. In a production environment, use strong,
  + unique passwords and consider using environment variables or a secure configuration file to store sensitive information.

## **Testing**

### **Manual Testing**

* **Test Case 1: Search for Existing Book by Title**
  + **Steps**:
    1. Log in as any user.
    2. From the main menu, select option 4. Search.
    3. Enter a known book title.
    4. Verify that the book details are displayed.
    5. Verify that the book type (Physical/Ebook/Audiobook) is clearly indicated in the results.
    6. Select the option to return to the main menu.
  + **Test Result**: Passed
  + :
* **Test Case 2: Search for Non-Existing Book**
  + **Steps**:
    1. Log in as any user.
    2. From the main menu, select option 4. Search.
    3. Enter a non-existing book title.
    4. Verify that "No books found" message is displayed.
    5. Select the option to return to the main menu.
  + **Test Result**: Passed
  + :
* **Test Case 3: Borrow Available Book**
  + **Steps**:
    1. Login as a student user.
    2. Select option 2. Borrowing Services from the main menu.
    3. Select option 1. Borrow a book from the Borrowing Services menu.
    4. Enter the ID of an available book.
    5. Verify that "Book borrowed successfully" message is displayed.
  + **Test Result**: Passed
  + :
* **Test Case 4: Attempt to Borrow Unavailable Book**
  + **Steps**:
    1. Login as a student user.
    2. Select option 2. Borrowing Services from the main menu.
    3. Select option 1. Borrow a book from the Borrowing Services menu.
    4. Enter the ID of an unavailable book.
    5. Verify that an error message is displayed.
  + **Test Result**: Passed
  + :
* **Test Case 5: Return Borrowed Book**
  + **Steps**:
    1. Login as a user who has borrowed a book.
    2. Select option 2. Borrowing Services from the main menu.
    3. Select option 2. Return a book from the Borrowing Services menu.
    4. Enter the ID of a book the user has borrowed.
    5. Verify that "Book returned successfully" message is displayed.
  + **Test Result**: Passed
  + :
* **Test Case 6: Attempt to Return Non-Borrowed Book**
  + **Steps**:
    1. Login as a student user.
    2. Select option 2. Borrowing Services from the main menu.
    3. Select option 2. Return a book from the Borrowing Services menu.
    4. Enter the ID of a book the user has not borrowed.
    5. Verify that an error message is displayed.
  + **Test Result**: Passed
  + :
* **Test Case 7: View Non-Empty Borrowing History**
  + **Steps**:
    1. Login as a user with borrowing history.
    2. Select option 2. Borrowing Services from the main menu.
    3. Select option 5. View borrowing history from the Borrowing Services menu.
    4. Verify that the list of borrowed books is displayed.
  + **Test Result**: Passed
  + :
* **Test Case 8: View Empty Borrowing History**
  + **Steps**:
    1. Login as a new user with no borrowing history.
    2. Select option 2. Borrowing Services from the main menu.
    3. Select option 5. View borrowing history from the Borrowing Services menu.
    4. Verify that "No borrowing history" message is displayed.
  + **Test Result**: Passed
  + :
* **Test Case 9: Reserve Unavailable Book**
  + **Steps**:
    1. Login as a student user.
    2. Select option 2. Borrowing Services from the main menu.
    3. Select option 3. Reserve a book from the Borrowing Services menu.
    4. Enter the ID of an unavailable book.
    5. Verify that "Book reserved successfully" message is displayed.
  + **Test Result**: Passed
  + :
* **Test Case 10: Attempt to Reserve Available Book**
  + **Steps**:
    1. Login as a student user.
    2. Select option 2. Borrowing Services from the main menu.
    3. Select option 3. Reserve a book from the Borrowing Services menu.
    4. Enter the ID of an available book.
    5. Verify that an error message is displayed.
  + **Test Result**: Passed
  + :
* **Test Case 11: View Active Reservations**
  + **Steps**:
    1. Login as a user with active reservations.
    2. Select option 2. Borrowing Services from the main menu.
    3. Select option 4. View my reservations from the Borrowing Services menu.
    4. Verify that the list of reserved books is displayed.
  + **Test Result**: Passed
  + :
* **Test Case 12: View Empty Reservations**
  + **Steps**:
    1. Login as a user with no reservations.
    2. Select option 2. Borrowing Services from the main menu.
    3. Select option 4. View my reservations from the Borrowing Services menu.
    4. Verify that "You have no active reservations" message is displayed.
  + **Test Result**: Passed
  + :
* **Test Case 13: View Overdue Books**
  + **Steps**:
    1. Login as a user with overdue books.
    2. Select option 2. Borrowing Services from the main menu.
    3. Select option 6. View my overdue books from the Borrowing Services menu.
    4. Verify that the list of overdue books is displayed.
  + **Test Result**: Passed
  + :
* **Test Case 14: View No Overdue Books**
  + **Steps**:
    1. Login as a user with no overdue books.
    2. Select option 2. Borrowing Services from the main menu.
    3. Select option 6. View my overdue books from the Borrowing Services menu.
    4. Verify that "You have no overdue books" message is displayed.
  + **Test Result**: Passed
  + :
* **Test Case 15: List All Books**
  + **Steps**:
    1. Login as any user.
    2. Select option 1. Book Services from the main menu.
    3. Select option 1. List all books from the Book Services menu.
    4. Verify that all books in the library are displayed.
  + **Test Result**: Passed
  + : Too many screenshots but all books showed.
* **Test Case 16: Update User Profile - Name**
  + **Steps**:
    1. Login as any user.
    2. Select option 3. User Management from the main menu.
    3. Select option 1. Update my profile from the User Management menu.
    4. Enter a new name and press enter for other fields.
    5. Verify that "Profile updated successfully" message is displayed.
  + **Test Result**: Passed
  + :
* **Test Case 17: Update User Profile - Email**
  + **Steps**:
    1. Login as any user.
    2. Select option 3. User Management from the main menu.
    3. Select option 1. Update my profile from the User Management menu.
    4. Press enter for name, enter a new email, and press enter for other fields.
    5. Verify that "Profile updated successfully" message is displayed.
  + **Test Result**: Passed
  + :
* **Test Case 18: Logout**
  + **Steps**:
    1. Login as any user.
    2. Select option 6. Logout from the main menu.
    3. Verify that "Logged out successfully" message is displayed.
    4. Verify that the login menu is shown.
  + **Test Result**: Passed
  + :
* **Test Case 19: Add Physical Book as Librarian**
  + **Steps**:
    1. Login as a librarian.
    2. Select option 1. Book Services from the main menu.
    3. Select option 2. Add a book from the Book Services menu.
    4. Enter book details for a physical book.
    5. Verify that "Book added successfully" message is displayed.
  + **Test Result**: Passed
  + :
* **Test Case 20: Add Ebook as Librarian**
  + **Steps**:
    1. Login as a librarian.
    2. Select option 1. Book Services from the main menu.
    3. Select option 2. Add a book from the Book Services menu.
    4. Enter book details for an ebook.
    5. Verify that "Book added successfully" message is displayed.
  + **Test Result**: Passed
  + :
* **Test Case 21: Add Audiobook as Librarian**
  + **Steps**:
    1. Login as a librarian.
    2. Select option 1. Book Services from the main menu.
    3. Select option 2. Add a book from the Book Services menu.
    4. Enter book details for an audiobook.
    5. Verify that "Book added successfully" message is displayed.
  + **Test Result**: Passed
  + :
* **Test Case 22: Remove Existing Book as Librarian**
  + **Steps**:
    1. Login as a librarian.
    2. Select option 1. Book Services from the main menu.
    3. Select option 3. Remove a book from the Book Services menu.
    4. Enter the ID of an existing book.
    5. Verify that "Book removed successfully" message is displayed.
  + **Test Result**: Passed
  + :
* **Test Case 23: View All Overdue Books as Librarian**
  + **Steps**:
    1. Login as a librarian.
    2. Select option 2. Borrowing Services from the main menu.
    3. Select option 7. View all overdue books from the Borrowing Services menu.
    4. Verify that a list of all overdue books is displayed.
  + **Test Result**: Passed
  + :
* **Test Case 24: Register New Student User as Librarian**
  + **Steps**:
    1. Login as a librarian.
    2. Select option 3. User Management from the main menu.
    3. Select option 2. Register new user from the User Management menu.
    4. Enter details for a new student user.
    5. Verify that "User registered successfully" message is displayed.
  + **Test Result**: Passed
  + :
* **Test Case 25: List All Users as Librarian**
  + **Steps**:
    1. Login as a librarian.
    2. Select option 3. User Management from the main menu.
    3. Select option 3. List all users from the User Management menu.
    4. Verify that a list of all users is displayed.
  + **Test Result**: Passed
  + : Too long for all screenshots.
* **Test Case 26: Remove Existing User as Librarian**
  + **Steps**:
    1. Login as a librarian.
    2. Select option 3. User Management from the main menu.
    3. Select option 4. Remove user from the User Management menu.
    4. Enter the username of an existing user.
    5. Verify that "User removed successfully" message is displayed.
  + **Test Result**: Passed
  + :
* **Test Case 27: Attempt to Register User with Existing Email**
  + **Steps**:
    1. Log in as a librarian.
    2. Select option 3. User Management from the main menu.
    3. Select option 2. Register new user from the User Management menu.
    4. Enter details for a new user, using an email that already exists in the system.
    5. Verify that the system shows an error message and doesn't register the duplicate user.
    6. Select the option to return to the main menu.
  + **Test Result**: Passed
  + :
* **Test Case 28: Search for Book with Empty String**
  + **Steps**:
    1. Login as any user.
    2. Select option 4. Search from the main menu.
    3. Enter an empty string as the search query.
    4. Verify that the system handles this appropriately (e.g., shows an error or lists all books).
  + **Test Result**: Passed
  + :
* **Test Case 29: Update User Profile with Existing Username**
  + **Steps**:
    1. Login as any user.
    2. Select option 3. User Management from the main menu.
    3. Select option 1. Update my profile from the User Management menu.
    4. Try to change the username to one that already exists in the system.
    5. Verify that the system shows an error message and doesn't update the profile.
  + **Test Result**: Passed
  + :
* **Test Case 30: Add Book with Duplicate ISBN as Librarian**
  + **Steps**:
    1. Login as a librarian.
    2. Select option 1. Book Services from the main menu.
    3. Select option 2. Add a book from the Book Services menu.
    4. Enter details for a new book, using an ISBN that already exists in the system.
    5. Verify that the system shows an error message and doesn't add the duplicate book.
  + **Test Result**: Passed
  + :
* **Test Case 31: View Database Content Report as Librarian**
  + **Steps**:
    1. Login as a librarian.
    2. Select option 5. Reports from the main menu.
    3. Select option 1. View database content report from the Reports menu.
    4. Verify that the database content report is displayed correctly.
  + **Test Result**: Passed
  + :