**COSC 31093**

**Enterprice Software Design and Architecture**

**Milestone 2:**

**REST API Design and Implementation**

# Group Number :33

Group Members:

EC/2021/047-K.M.S.M Kulathunga

PS/2021/082-W.D.K.R Anjula

PS/2021/032-M.T.S.S Perera

## Proposal: Library Management System (LMS)

This system is designed to automate and streamline the operations of a library, including book cataloging, member management, loan tracking, and inventory control. It will provide librarians, members, and administrators with an efficient digital platform to manage library resources and enhance user experience.

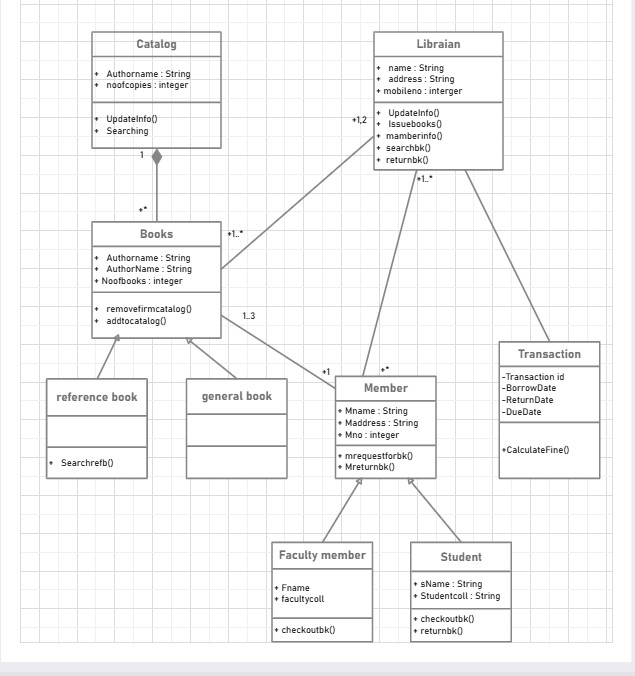
**Key Features:**

* **Book Management**: Catalog, categorize, and manage books, journals, and digital resources.
* **Member Management**: Handle user registrations, profiles, and membership renewals.
* **Loan & Return System**: Track book checkouts, due dates, returns, and late fees.
* **Search & Reservation**: Allow users to search for books and reserve available copies.
* **Reporting & Analytics**: Generate reports on book circulation, overdue items, and user activity.
* **Fine & Payment Processing**: Manage late fees and payment transactions.

**Intended Users:**

* **Librarians:** Manage inventory, process loans/returns, and oversee daily operations.
* **Members (Students/Patrons):** Browse, borrow, and reserve books; manage accounts.
* **Administrators:** Monitor system performance, generate reports, and configure settings.

**Class Diagram**



**This class diagram models the core structure of a Library Management System, capturing the main entities, their attributes, and relationships. Here’s an analysis of the design:**

**Key Components**

* 1. **Catalog Class:**

The Catalog class manages collections of books, tracking attributes like author name and the number of copies. It provides methods for updating catalog information and searching, reflecting its central role in organizing library resources.

* 1. **Books Class and Specializations:**

The Books class represents individual book records, containing details such as author name and the number of books. It includes methods for catalog management (add/remove). The diagram also shows specialization with Reference Book and General Book as subclasses, demonstrating inheritance and allowing for specific behaviors like searching for reference books.

* 1. **Member Class and Subtypes:**

The Member class holds user details (name, address, member number) and methods for requesting and returning books. It is further specialized into Faculty Member and Student subclasses, each with unique attributes and methods (e.g., faculty collection, student ID, checkout/return methods), showcasing the use of inheritance to capture different user roles.

* 1. **Librarian Class:**

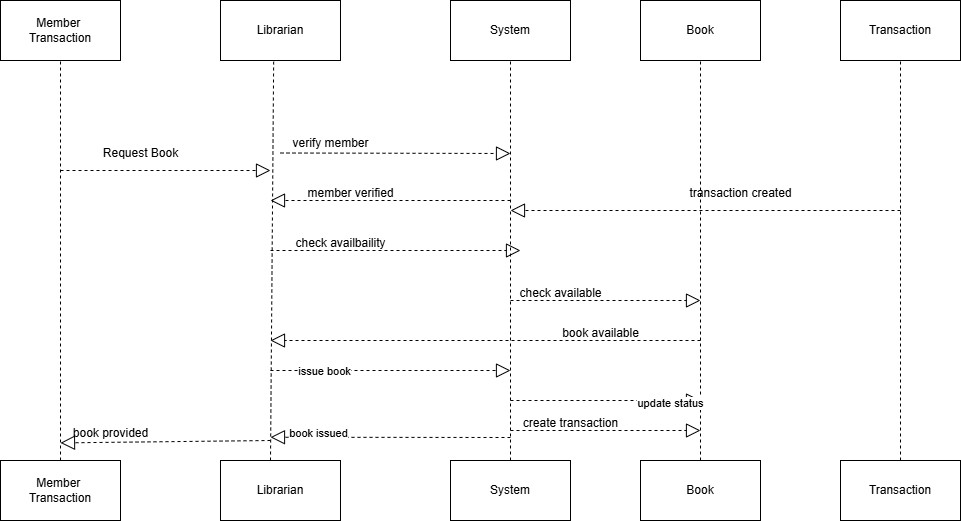
The Librarian class manages library operations, including issuing and returning books, updating member information, and searching the catalog. It interacts with both Members and Transactions, underlining the librarian’s central role in system workflows.

* 1. **Transaction Class:**

The Transaction class records borrowing events, capturing transaction ID, borrow/return dates, and due dates, with a method to calculate fines. This supports accountability and fine management for overdue books1. **Relationships**

* + The diagram depicts associations between Catalog, Books, Members, Librarians, and Transactions, reflecting real-world interactions such as cataloging, borrowing, and returning books.
  + Inheritance is used for both book types and member roles, promoting code reuse and clarity.
  + Multiplicities (e.g., 1..\*, 1..3) indicate how many instances of each class can participate in relationships, ensuring accurate modeling of the system’s constraints

**Sequence Diagram**



The sequence diagram for the "Book Borrowing" process in a Library Management System visually represents the interaction between the Member, Librarian, System, Books, and Transaction components involved in the borrowing workflow. It shows the chronological flow of messages and operations among these participants:

**Key Components:**

* + **Member:** Initiates the book borrowing request.
  + **Librarian:** Acts as the intermediary, verifying member eligibility and managing the borrowing process.
  + **System:** Handles verification, availability checking, transaction creation, and status updates.
  + **Books:** Represents the book item being borrowed and its availability status.
  + **Transaction:** Records the borrowing event and manages due dates and fines.

**Purpose:**

This diagram clarifies the flow of data and responsibilities among system components and users, helping developers and stakeholders understand the system behavior, improve design decisions, and debug workflows. It highlights how user requests are processed step-by-step through the system to successfully issue a book.