OOP Course Project Presentation Template

---

1. Title Slide

- **Project Title:** Library Management System
- **Student Name**: Sabina Novruzova
- **Student ID**: 61220061

---

2. Project Overview

Project Information

The Library Management System is designed to simplify and digitize library operations. It provides features such as adding books, registering members, borrowing books, and returning books.

The main purpose of the system is to make library management easier for librarians and offer a smooth experience for library members. It supports two types of members: regular and premium, each with their own borrowing limits.

Purpose and Objectives

1. To automate manual library tasks like tracking books and managing member information.
2. To reduce the chances of human error, such as misplacing records or exceeding borrowing limits.
3. To provide a scalable and maintainable solution for managing both books and members.

Users

The system is designed for:

- Librarians who manage book inventory and member data.
- Regular members, with a borrowing limit of 5 books.
- Premium members, who can borrow up to 10 books.

Problem Statement

Libraries often face challenges when managing book records manually, including misplaced data, delays in finding books, and difficulties in tracking members' borrowing histories. This project solves these issues by automating library tasks, ensuring books are accurately tracked and borrowing is managed effectively.

---

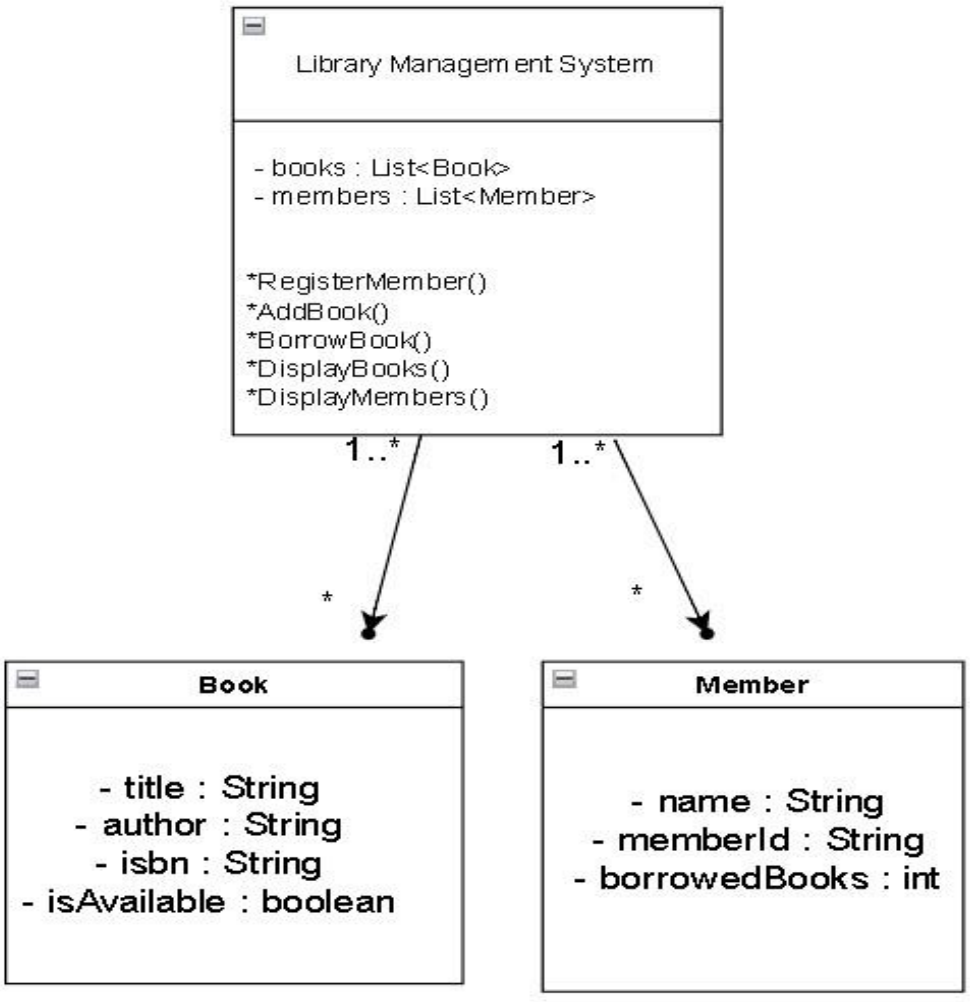
3. System Architecture

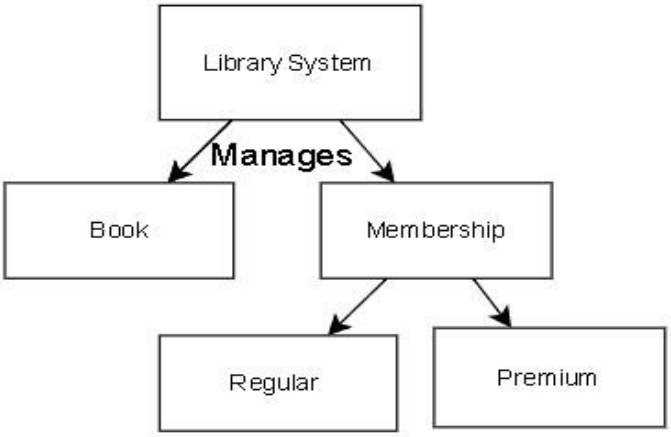
Class Diagram

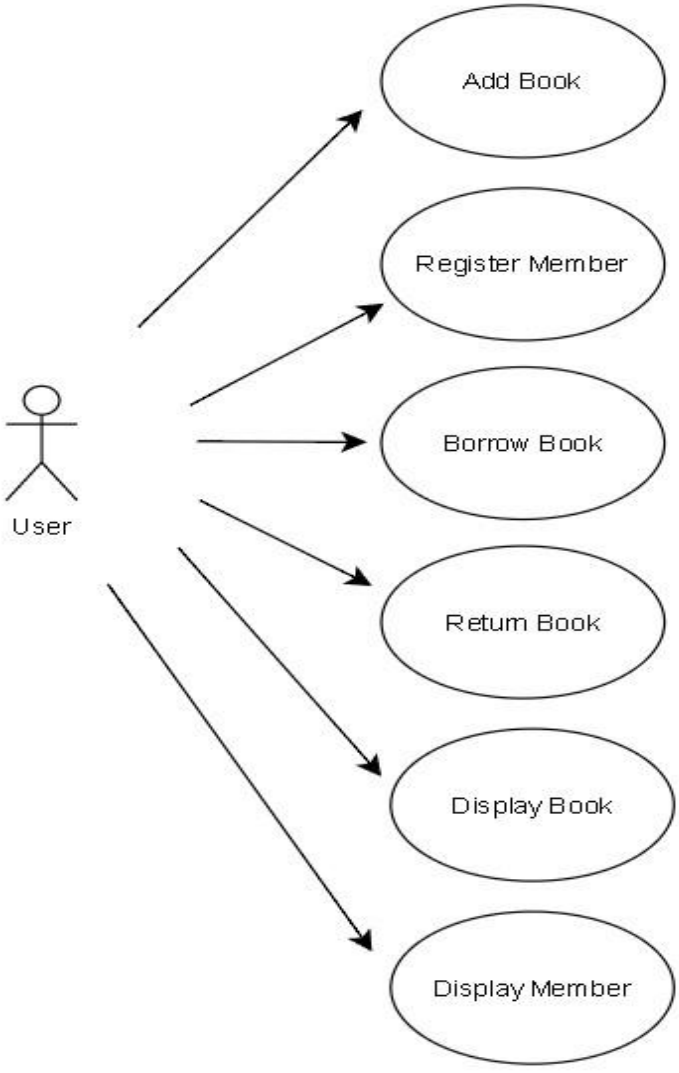
The system architecture revolves around several interconnected classes:

1. Book: Represents books with attributes like title, author, and availability.
2. Member (Base Class): Encapsulates common functionality for all members.
3. PremiumMember & RegularMember: Subclasses of Member that define borrowing limits.
4. LibrarySystem: Handles the core operations like adding books, registering members, borrowing, and returning books.

UML Diagram




Library System
Manages
Book
Membership
Regular
Premium


Add Book
Register Member
Borrow Book
Retum Book
Display Book
Display Member
User

Highlights:

- Inheritance: PremiumMember and RegularMember inherit from Member.
- Association: LibrarySystem interacts with both Book and Member.

---

4. Key Features

The main features of the Library Management System are:

1. Book Management: Librarians can add books by entering the title, author, and ISBN. The system automatically tracks their availability.
2. Member Registration: Users can be registered as either regular or premium members based on their borrowing needs.
3. Borrow and Return Books: The system ensures that books are only borrowed by eligible members (within their borrowing limits) and tracks whether books are returned on time.
4. Search Functionality: Books and members can be searched using their unique identifiers (ISBN and member ID, respectively).

---

5. OOP Concepts Implemented

Key Concepts:

1. **Classes and Objects:**
    - Classes like Book, Member, and LibrarySystem represent real-world entities.
    - Objects are created to represent specific books and members.
2. **Encapsulation:**
    - Attributes like title, isbn, and borrowedBooks are private. Access to these attributes is restricted through getter and setter methods to maintain data integrity.
3. **Inheritance:**
    - The Member class serves as the parent class for PremiumMember and RegularMember. This allows reusability of common attributes and methods, while each subclass defines specific behaviors like borrowing limits.
4. **Polymorphism**:

- The canBorrow method is overridden in PremiumMember and RegularMember to define custom borrowing rules for each member type.

5. **Abstraction**:

   - While the Member class is not abstract, it provides a base structure that hides implementation details, offering a higher-level interface for its subclasses.

---

6. Evaluation of OOP Concepts

1. **Classes and Objects:**

   - A clear class structure is used to separate concerns. For instance, Book handles book-related data, while Member handles user-specific data.

2. **Encapsulation:**

   - Used throughout the project. For example, book availability (isAvailable) is private and can only be modified through specific methods like setAvailable.

   - Simplifies the implementation of different member types, avoiding duplication of code.

3. Polymorphism:

   - Demonstrated in the borrowing rules, where the system handles both member types (RegularMember and PremiumMember) seamlessly using the same method call (canBorrow).

4. Abstraction:

   - The Member class abstracts shared functionality, making the system easier to maintain and extend.

---

7. Challenges Faced

1. Challenge: Designing the borrowing rules for different member types.
   Solution: Used inheritance and polymorphism to define borrowing behavior for each member type.

2. Challenge: Ensuring book availability updates correctly when books are borrowed or returned. Solution: Implemented checks for book status and updated attributes like isAvailable to ensure proper tracking.

3. Challenge: Debugging runtime errors, especially when invalid inputs were provided.
   Solution: Added input validation to handle unexpected cases, like entering invalid membership types.

4. Challenge: Visualizing and maintaining the relationships between classes.
   Solution: Created a UML diagram during the planning phase to ensure a clear design.

8. Project Screenshots

Here are some screenshots of the system in action:

```
Library Management System
1. Add Book
2. Register Member
3. Borrow Book
4. Return Book
5. Display Books
6. Display Members
0. Exit
Enter your choice: 5

Available Books:
*Anna Karenina  L. Tolstoy      ISBN-112233     Available-true
*Jane Eyre      Ch. Brontë      ISBN-445566     Available-true
*Harry Potter   J. K. Rowling   ISBN-778899     Available-true
*Hamlet W. Shakespeare  ISBN-123456     Available-true

Library Management System
1. Add Book
2. Register Member
3. Borrow Book
4. Return Book
5. Display Books
6. Display Members
0. Exit
Enter your choice: 3
Enter member ID: 001
Enter book ISBN: 112233
Book borrowed successfully!

Library Management System
1. Add Book
2. Register Member
3. Borrow Book
4. Return Book
5. Display Books
6. Display Members
0. Exit
Enter your choice: 3
Enter member ID: 003
Enter book ISBN: 112233
Book is already borrowed.
```

```
Library Management System
1. Add Book
2. Register Member
3. Borrow Book
4. Return Book
5. Display Books
6. Display Members
0. Exit
Enter your choice: 6

Registered Members:
*ID-001 Name-Sabrina    BorrowedBooks-1
*ID-002 Name-David      BorrowedBooks-0
*ID-003 Name-Celine     BorrowedBooks-0
*ID-004 Name-Anna       BorrowedBooks-0

Library Management System
1. Add Book
2. Register Member
3. Borrow Book
4. Return Book
5. Display Books
6. Display Members
0. Exit
Enter your choice: 5

Available Books:
*Anna Karenina  L. Tolstoy      ISBN-112233     Available-false
*Jane Eyre      Ch. Brontë      ISBN-445566     Available-true
*Harry Potter   J. K. Rowling   ISBN-778899     Available-true
*Hamlet W. Shakespeare  ISBN-123456     Available-true
```

```
Library Management System
1. Add Book
2. Register Member
3. Borrow Book
4. Return Book
5. Display Books
6. Display Members
0. Exit
Enter your choice: 4
Enter book ISBN: 112233
Enter member ID: 001
Book returned successfully!
```

```
Library Management System
1. Add Book
2. Register Member
3. Borrow Book
4. Return Book
5. Display Books
6. Display Members
0. Exit
Enter your choice: 5

Available Books:
*Anna Karenina  L. Tolstoy      ISBN-112233     Available-true
*Jane Eyre      Ch. Brontë      ISBN-445566     Available-true
*Harry Potter   J. K. Rowling   ISBN-778899     Available-true
*Hamlet W. Shakespeare  ISBN-123456     Available-true

Library Management System
1. Add Book
2. Register Member
3. Borrow Book
4. Return Book
5. Display Books
6. Display Members
0. Exit
Enter your choice: 0
Exiting...
PS C:\Users\hp\Desktop\practice>
```