

EXPERIMENT 10

Question:

Draw a use case diagram for Library Management System is an automation system used to manage a library and the different resource management required in it like cataloguing of books, allowing check out and return of books, invoicing, user management, etc. The user can search for books details using few book properties (Book ID, Title, Author, and Publisher). Searching should return details about all the book copies that match the search query.

Aim:

The objective of this **Use Case Diagram for Library Management System** is to **visualize** the various interactions between the system and its users. The system automates key library functionalities such as **book cataloging, searching, borrowing, returning, invoicing, and user management** to streamline operations efficiently.

Procedure:

1. Identify Actors:

- Librarian: Manages books, users, and invoices.
- Member: Searches, views, checks out, and returns books.
- System: Facilitates operations like invoice generation.

2. Identify Use Cases:

- Manage Books: Librarian manages the book collection.
- Search Books: Members can search books by ID, Title, Author, or Publisher.
- View Book Details: Members can check book availability and details.
- Catalog Books: Librarian adds new books to the system.
- Check Out Book: Members borrow books from the library.
- Return Book: Members return borrowed books.
- Manage Users: Librarian manages user records.
- Generate Invoice: System generates invoices for overdue books.

3. Draw Use Case Diagram:

- Use ovals to represent use cases (functionalities).

- Use stick figures to represent actors (users).
- Use lines to connect actors with their respective use cases.

4. Define Relationships:

- Librarian is linked to managing books, cataloging, user management, and generating invoices.
- Members are linked to searching, viewing, checking out, and returning books.
- System handles invoicing.

Output:

Usecase Diagram



Result

This **use case diagram** provides a structured view of the **Library Management System**, showcasing how different users interact with key functionalities. It helps in understanding the system flow and user roles efficiently.