**Use Case Diagram for CityLibrary Management System**

**Use case diagrams** referred as a Behavior model or diagram. It simply describes and displays the relation or interaction between the users or customers and providers of application service or the system. It describes different actions that a system performs in collaboration to achieve something with one or more users of the system. Use case diagram is used a lot nowadays to manage the system.

Diagram

Description automatically generated

Here, we will understand the designing use case diagram for the library management system. Some scenarios of the system are as follows :

1. *User who registers himself as a new user initially is regarded as staff or student for the library system.*
   * *For the user to get registered as a new user, registration forms are available that is needed to be fulfilled by the user.*
   * *After registration, a library card is issued to the user by the librarian. On the library card, an ID is assigned to cardholder or user.*
2. *After getting the library card, a new book is requested by the user as per there requirement.*
3. *After, requesting, the desired book or the requested book is reserved by the user that means no other user can request for that book.*
4. *Now, the user can renew a book that means the user can get a new due date for the desired book if the user has renewed them.*
5. *If the user somehow forgets to return the book before the due date, then the user pays fine. Or if the user forgets to renew the book till the due date, then the book will be overdue and the user pays fine.*
6. *User can fill the feedback form available if they want to.*
7. *Librarian has a key role in this system. Librarian adds the records in the library database about each student or user every time issuing the book or returning the book, or paying fine.*
8. *Librarian also deletes the record of a particular student if the student leaves the college or passed out from the college. If the book no longer exists in the library, then the record of the particular book is also deleted.*
9. *Updating database is the important role of Librarian(admin).*

***References****:* *https://www.geeksforgeeks.org/use-case-diagram-for-library-management-system/*

Key-Words:

***Librarian****--- admin*

**User**----members(staff/student/other registered )

*Fine—overdue payment*

*Renewed—update books/profile*

**Class Diagram for Library Management System**

In [Object-Oriented modeling](https://www.geeksforgeeks.org/types-of-models-in-object-oriented-modeling-and-design/), the main building block generally represents different objects in a system, their attributes, their different functions, and relationships among objects. These building blocks are known as **Class Diagram**.

Class diagrams are generally used for conceptual modeling of static view of a software application, and for modeling translating models into programming code in a detailed manner. At time of developing or construction software systems, a class diagram is widely used. They are also used for data modeling. It is used to show classes, relationships among them, interface, association, etc. Class in a class diagram simply is a blueprint of an object. It simply describes and explains different type of objects in system, and different types of relationships that exist between them.

**Class Diagram for Library Management System :**  
Aggregation and Multiplicity are two important points that need to take into consideration while designing a Class Diagram. Let us understand in detail.

1. **Aggregation –**  
   Aggregation simply shows a relationship where one thing can exist independently of other thing. It means to create or compose different abstractions together in defining a class. Aggregation is represented as a part of relationship in class diagram. In diagram given below, we can see that aggregation is represented by an edge with a diamond end pointing towards superclass. The “Library Management System” is superclass that consists of various classes.

These classes are User, Book, and Librarian as shown in diagram. Further, for “Account” class, “User” is a superclass. All of these, share a relationship and these relationships are known as aggregate relationships.

1. **Multiplicity –**  
   Multiplicity means that number of elements of a class is associated with another class. These relations can be one-to-one, many-to-many, and many-to-one or one-to-many. For denoting one element we use **1**, for zero elements we use **0**, and for many elements we use **\***. We can see in diagram; many users are associated with many books denoted by **\*** and this represents a **many-to-many** type of relationship. One user has only one account that is denoted by 1 and this represents a **one-to-one** type of relationship.

Many books are associated with one librarian and this represents **many-to-one** or **one-to-many** type of relationship. All these relationships are shown in diagram.

Class Diagram for Library Management System simply describes structure of Library Management System class, attributes, methods or operations, relationship among objects.

**Classes of Library Management System :**

* **Library Management System class –**  
  It manages all operations of Library Management System. It is central part of organization for which software is being designed.
* **User Class –**  
  It manages all operations of user.
* **Librarian Class –** It manages all operations of Librarian.
* **Book Class –**  
  It manages all operations of books. It is basic building block of system.
* **Account Class –**  
  It manages all operations of account.
* **Library database Class –**  
  It manages all operations of library database.
* **Staff Class –**  
  It manages all operations of staff.
* **Student Class –**  
  It manages all operations of student.

**Attributes of Library Management System :**

* **Library Management System Attributes –**  
  UserType, Username, Password
* **User Attributes –**  
  Name, Id
* **Librarian Attributes –**  
  Name, Id, Password, SearchString
* **Book Attributes –**  
  Title, Author, ISBN, Publication
* **Account Attributes –**  
  no\_borrowed\_books, no\_reserved\_books, no\_returned\_books, no\_lost\_books fine\_amount
* **Library database Attributes –**  
  List\_of\_books
* **Staff Class Attributes –**  
  Dept
* **Student Class Attributes –**  
  Class

**Methods of Library Management System :**

* **Library Management System Methods –**  
  Login(), Register(), Logout()
* **User Methods –**  
  Verify(), CheckAccount(), get\_book\_info()
* **Librarian Methods –**  
  Verify\_librarian(), Search()
* **Book Methods –**  
  Show\_duedt(), Reservation\_status(), Feedback(), Book\_request(), Renew\_info()
* **Account Methods –**  
  Calculate\_fine()
* **Library database Methods –**  
  Add(), Delete(), Update(), Display(), Search()

Diagram

Description automatically generated

Reference : https://www.geeksforgeeks.org/class-diagram-for-library-management-system/?ref=rp