**EXPERIMENT 3:**

**CASE STUDY: LIBRARY MANAGEMENT SYSTEM**

1. **Aim: Identify and analyze events**

The flow of events of a usecase contains the most important information derived from usecase modeling work. It should describe the usecases flow of events clearly enough for an outsider to easily understand it. Remember flow of events should present what the system does, not how the system is design to perform the required behavior.

Guidelines for the contents of the flow of events are,

1) Describe how the usecase starts and ends.

2) Describe what data is exchanged between the actor and the usecase

3) Do not describe the details of the user interface, unless it is necessary to understand the behavior of the system.

**b) Aim: Identify usecases**

**Procedure:**

**a)** **Register member: H**ere the members who can access the library are registered based on the id nos.

**b)** **Issue book:** Required book is issued.

**c) Verify member: B**efore issuing book the specific member is verified whether he/she is registered or not.

**d) Check availability or book: A**fter verifying the person then the availability of book which he/she selected is checked whether the book is available or not after verifying & checking the book is issued.

**e) Return book: A**fter the specific time is over the person should return book during returning.

**f) Calculate fine: T**he fine is calculated.

**g) Enquiry: I**n this the member is checked if fine is existed or not.

**h) Maintaining books: T**his is done by the librarian that how many books are present, person or members login id and their registrations etc.

**Graphical representations:**

****

1. **Aim :Develop event table:**

|  |  |
| --- | --- |
| **Usecase** | **Description** |
| Register member | The members who can access the library are registered based on the ids |
| Issue book-verify member | Before issuing the book the specific member is verified whether he/she is registered or not |
| Check availability of book | Check the book is available or not |
| Return book | After the specific time is over the person should return |
| Calculate fine | The fine is calculated if their fine |
| Enquiry | In this the member is checked if fine is existed or not |
| Maintaining books | This is done by the librarian to cross checking the books |

1. **Identify and analyze domain classes:**



1. **Aim: Represent class diagram for library management system**

🡪 Class diagram



**🡪Usecase diagram**



1. **Develop CRUD matrix to represent relationships between usecase and problem domain classes**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Entity/Function | Librarian | Transaction | Book | Journals | Study Books | Magzines | Bill | Memberrecord | Student | faculty |
| Register member | UC,D |  |  |  |  |  |  |  |  |  |
| Enquiry | C |  |  |  |  |  |  | R | R | R |
| IssueBook | C |  | R | R | R | R |  |  |  |  |
| VerifyBook | R |  |  |  |  |  |  |  |  |  |
| VerifyMember |  |  |  |  |  |  |  |  |  |  |
| CheckAvalilability of Book |  |  | R | R | R | R |  | R | R | R |
| ReturnBook | R | R |  |  |  |  |  |  | C | C |
| CalculateFine | C | C |  |  |  |  |  |  | R | R |
| MaintainBooks | C |  |  |  |  |  |  |  |  |  |

**Result:** The Design was successfully completed.