Library Information System

LIBRERIA
LIBRARY INFORMATION SYSTEM
Group Number : 52
Name of group members:

- Ashrujit Ghoshal (14CS10060)
- Sayan Ghosh (14CS10060)

Contents

1 INTRODUCTION

1.1 Purpose

This Software Requirement Specification (SRS) document provides a complete list and description of all the functions and specifications of the Library Information System (LIS).

The software is meant to hold the records of the books and the users and to provide the functionality of issuing and returning books. Apart from that the users can also request to reserve a required book if available in the library.

The expected audience of this document are Library clerk , librarian of an institute and the developer of the software.

Problem Statement

Different activities of the library of our institute pertaining to the issue and return of the books by the members of the library and various queries regarding books as listed below are to be automated.

- The library has 10,000 books. Each book is assigned a unique identification number (called ISBN number). The Library clerk should be able to enter the details of the book into the LIS through a suitable interface.
- There are four categories of members of the library: undergraduate students, post graduate students, research scholars, and faculty members.
- Each library member is assigned a unique library membership code number.
- Each undergraduate student can issue up to 2 books for 1 month duration.
- Each postgraduate student can issue up to 4 books for 1 month duration.
- Each research scholar can issue up to 6 books for 3 months duration.
- Each faculty member can issue up to 10 books for six months duration.
- The LIS should answer user queries regarding whether a particular book is available. If a book is available, LIS should display the rack number in which the book is available and the number of copies available.
- LIS registers each book issued to a member. When a member returns a book, LIS deletes the book from the member's account and makes the book available for future issue.
- Members should be allowed to reserve books which have been issued out. When such a reserved book is returned, LIS should print a slip for the concerned member to get the book issued and should disallow issue of the book to any other member for a period of seven days or until the member who has reserved the books gets it issued.
- When a member returns a book, LIS prints a bill for the penalty charge for overdue books. LIS calculates the penalty charge by multiplying the number of days the book is overdue by the penalty rate.
- LIS prints reminder messages for the members against whom books are over due, upon a request by the Librarian.
- LIS should allow the Librarian to create and delete member records.

- The Librarian periodically needs to check if there are any books which have not been issued even once during the last five years. He uses this information in planning to dispose off unused and old books. For this purpose it is necessary for LIS to maintain the statistics regarding issuance of different books.
- When books are disposed off, the Library clerk should be able to delete the book from the list of books in the Library and when a book is procured the system should support entering the details.

Q and A with the client

Question Answer with the client:

- Q : Is the ISBN number same for all the copies of a book.If yes then how do we differentiate between copies?
  A : Yes,ISBN number is same for all copies of a book.They are differentiated by a seperate copy ID
- Q : Is the month mentioned in the problem satement 30 days or from a date of one month to the sme date of the next?
  A : It is not 30 days.It is from a date of a month to the same date of another.For instance, one month from 5th February is 5th March.
- Q : Can an user reserve books if he has exhausted his reseve quota?
  A : No, he has to return his issued books before he can reserve books.
- Q : When asked to display rack number of a book,do we need to display rack number of all copies of a book?
  A : Yes

1.2 Document Conventions

The conventions used in the document are as per the IEEE standard provided for formatting the SRS document of a software

1.3 Definitions

The following terms are used in this document to refer to various componensts of the system :

Librarian	Person-in-charge (human administrator) of the software
Member	Members of the library who can issue a book
Issue	The act of borrowing a book by a user against his/her account
Return	The act of returning the book issued by an user previously
Reserve	The act of reserving a book currently unavailabele but for issuing in future
Fine	The penalty to be paid by a user in case he/she fails the deadline to return the book
ISBN	International Standard Book Number, a unique ID for each book

1.4 Intended Audience

The intended readers of this document are the developers of the software, testers, library owners and managers and coordinators.
These people are meant to understand the details of the software architecture modelled and used to understand the proper functioning of the software and their individual roles in the software also.

Any suggested changes on the requirements listed on this document should be included in the last version of it so it can be a reference to developing and validating teams.

1.5 Scope

LIS is a GUI tool that enhances the manual process of keeping records in a library.This is an appication which has been designed keeping in mind that it is meant to run on Library computers and to allow the library to keep a track of the books present.It allows the users to check availabilty whenever they want. It also allows the users to reserve a book in advance if needed. The librarian is able to keep a record of the overdue books, add new books, dispose old data, add new member or delete a member. It is a powerful but still functionally simple library management software for big libraries and can provide a free easy-to-use system for rising libraries.
Presently the maximum number of books has been kept constant to ensure a fruitful implementation of the software architecture. In future the list of books may be made to increase dynamically to make it more realistic.
Also since the software is very useful and lightweight in design it may be further incorporated in a bigger software like school automation system or an university automation system of which this will be a small but a necessary module.
The software may be futher modified to make a web interface also later in order to enable a remote access feature thus generalizing it further for common use of the public.

1.6 References

The following references have been consulted to make a suitable and effective software for library information system.

- IEEE standard 830-1998 recommended practice for Software Requirements Specifications-Description
- Bourque, P.; Fairley, R.E. (2014). "Guide to the Software Engineering Body of Knowledge (SWEBOK)". IEEE Computer Society. Retrieved 17 July 2014.
- Pressman, Roger (2010). Software Engineering: A Practitioner's Approach. Boston: McGraw Hill. p. 123. ISBN 9780073375977.
- "DI-IPSC-81433A, DATA ITEM DESCRIPTION SOFTWARE REQUIREMENTS SPECIFICATION (SRS)". everyspec.com. 1999-12-15. Retrieved 2013-04-04.
- Stellman, Andrew and Greene, Jennifer (2005). Applied software project management. O'Reilly Media, Inc. p. 308. ISBN 0596009488.

2 OVERALL DESCRIPTION

2.1 Product Perspective

The product is meant to automate the manual work of a librarian and library clerk to increase the efficiency of the process. It will facilitate the users to check a book's availability status . One can reserve, issue and return a book via this software. The software is designed to automatically update the details in the account of the user thus minimizing the laboriuos work earlier used to be performed by the clerks in the manual process.
One can genrate the reports and penalty bills via this software directly. It is designed to make the proces of issuing and searching books more easy.

It is designed to make the process of issuing, reserving and returning of a book more swift and efficient with minimum manual effort.The background cron job takes care of updation of the user account in case any of the above functions are performed.

2.2 Product Features

The product provides the features to create a librarian , library clerks and a set of users comprising of UG student, PG student, Faculty and Research Scholar.

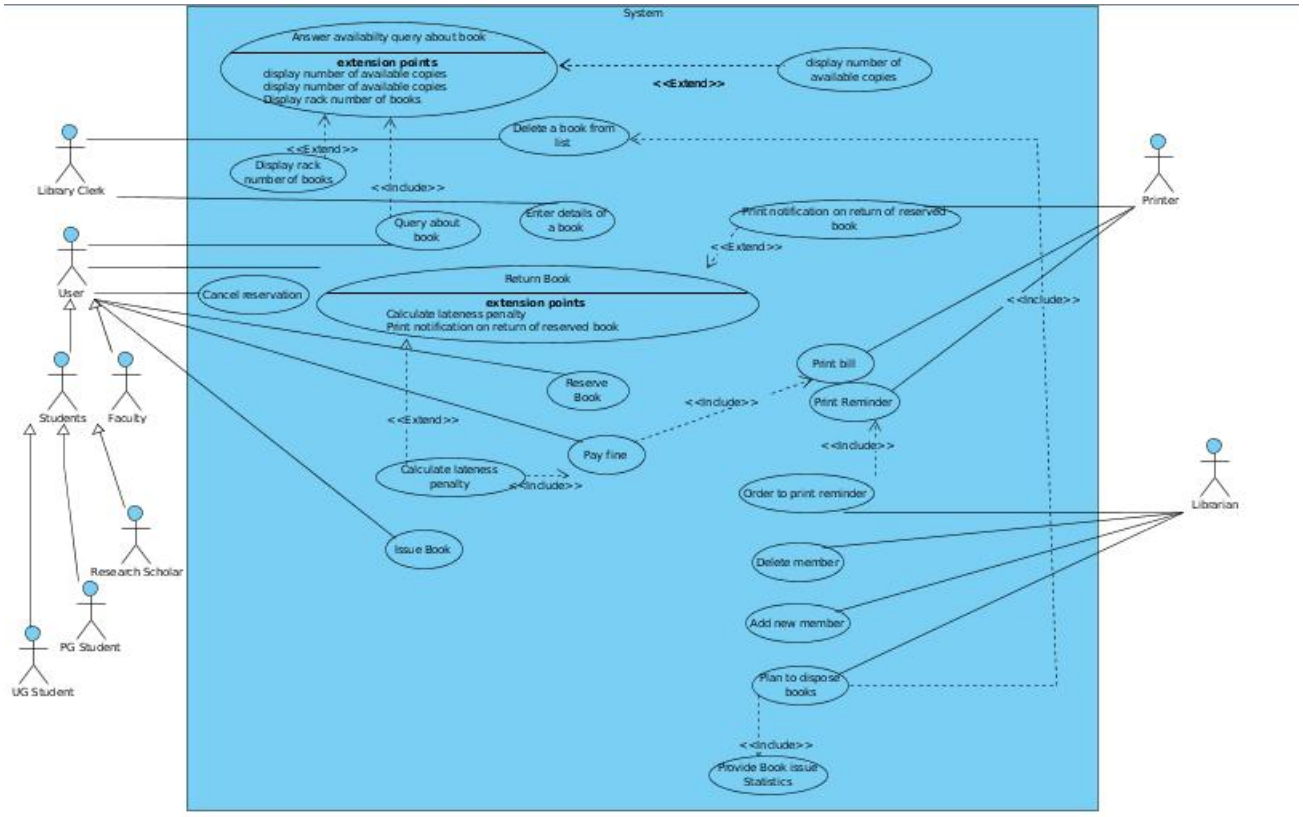
All the users irrespective of their type can issue and return a book.The librarian has some special functions like creation of a library member/user. The librarian can have access to account of any user and can look into it in case need arises.

The librarian has the sole administrator access to the software and can also issue a notification to the clerk to remove a book from the library if need arises.

Library clerks on the other hand can add and remove a book from the database of the library. During the return of the book the background cron job automatically checks if there is a need to impose fine in case the book has been overdue.

Finally there is a non-physical member called as LIS (syatem admin) which is autonomous and monitors all the finctioning of the software. It automatically notifies about the delays to the clerks and helps to update and keep track of all the activities of the software.

A class diagram showing the relationship is given below to enhance the understanding of the workflow.



Use cases

1. User use cases:

   - Query about Book

     – Preconditions:
     1. The user must be logged in .
     2.The book must exist in the library

     – Postcondition:
     If the book exists in the library,the availablity status of the book is returned

     – Failure Situations:
     The library does not have the book

     – Postcondition in case of failure:
     A message to user about the same

     – Actors:
     User communiates with the system

     – Trigger:
     User chooses the option to search books

     – Main Success Scenario:
     The library has a copy of the book and it is available for issue.

     – Extensions/Variations:
     The library has a copy of the book but currently none of the copies are available.The book maybe reserved by the user.

   - Issue Book

     – Preconditions:
     1. The user must be logged in .
     2.The book must exist in the library
     3.It must be available for issue.
     4.The user must not have exhausted his quota of number of books

     – Postcondition:
     After successful issue the user account is updated

     – Failure Situations:
     1. The library does not have the book
     2.The library has the book and it is not avaiable for issue.
     3.The user has exhausted his quota of maximum number of books

- Postcondition in case of failure:
  In failure case 2. the user may choose to reserve the book if he has not exhausted his quota

- Actors:
  User communiates with the system

- Trigger:
  User chooses the option to issue books

- Main Success Scenario:
  The library has a copy of the book and it is available for issue.

- Return Book

  - Precondition:
    1.User must be logged in.
    2.User must have previously issued the book.

  - Postcondition:
    1.If the book was overdue the penalty is calculated and a bill is printed
    2.In case the book was reserved by some other user, a notification is sent out to the other user.
    3.The user account is updated

  - Failure Situations:
    The user has not issued any book

  - Postcondition in case of failure:
    A message is give to the user about the same

  - Actors:
    User communiates with the system

  - Trigger:
    User chooses the option to return issued books

  - Main Success Scenario:
    The user had previously issued the book

- Reserve Book

  - Preconditions:
    1. The user must be logged in .
    2.The book must exist in the library
    3.It must not be available for issue.
    4.The user must not have exhausted his quota of number of books

- Postcondition:
  1.After successful issue the user account is updated
  2.When the book is returned a notification is sent to the user.

- Failure Situations:
  1. The library does not have the book
  2.The library has the book and it is avaiable for issue.
  3.The user has exhausted his quota of maximm number of books

- Postcondition in case of failure:
  In failure case 2. the user may choose to issue the book if he has not exhausted his quota

- Actors:
  User communiates with the system

- Trigger:
  User chooses the option to reserve book

- Main Success Scenario:
  The library has a copy of the book and it is not available for issue.

• Cancel Reservation

- Preconditions:
  1. The user must be logged in .
  2.The user must have reserved the book

- Postcondition:
  1.After successful issue the user account is updated
  Failure Situations:
  The user has not issued any book

- Actors:
  User communiates with the system

- Trigger:
  1.User chooses the option to cancel reservation of a book
  2.User does not issue the reserved book within 7 days of return

- Main Success Scenario:
  The library has a copy of the book and the user must have reserved it previously

• Pay Fine

- Precondition:

(a) User must be logged in.

(b) User must have previously issued the book.
The book must be overdue

- Postcondition:
  The book was overdue the penalty is calculated and a bill is printed
- Failure Situations:
  (a) The user has not issued any book
  (b) No returned books are overdue
- Actors: User communiates with the system
- Trigger: User chooses the option to return issued books
- Main Success Scenario: The user had previously issued the book and the book is overdue

2. Library Clerk Use Cases

- Enter details of a book
  - Preconditions:1. The clerk must be logged in.
    2.The book must not be previously entered in the system
  - Failure Situations:
    The book is already in the system
  - Postcondition in case of failure:
    A message to clerk about the same
  - Actors:
    library clerk communicates with the system
  - Trigger:
    Clerk chooses the option to enter new books
  - Main Success Scenario:
    The library does not have the book and the book is newly entered in the system
  - Extensions/Variations:
    The library hasthe book and the umber of copies is increased
- Delete a book
  - Preconditions:
    1. The clerk must be logged in.
    2.The book must be previously entered in the system
    3.The librarian has decided to dispose the book
  - Failure Situations:
    The book is not in the system
  - Postcondition in case of failure:
    A message to clerk about the same
  - Actors:
    library clerk communicates with the system

- Trigger:
  Clerk chooses the option to delete books

- Main Success Scenario:
  The library has the book and it is removed from the system.

- Extensions/Variations:
  The library has the book and the number of copies is reduced.

3. Librarian Use Cases:

- Add new member
  - Preconditions:1.Librarian must be logged in
    2. A person must apply for membership

  - Postcondition:
    A new member account is created

  - Failure Situations:
    The user is already registered

  - Postcondition in case of failure:
    A message to librarian about the same

  - Actors:
    Librarian communicates with the system

  - Trigger:
    Librarian chooses the option to add member

  - Main Success Scenario:
    The user is not previously registered

- Delete member

  - Preconditions:
    1.Librarian must be logged in
    2. A person must apply for cacellation membership

  - Postcondition:
    The member account is deleted

  - Failure Situations:
    The user has no account

- Postcondition in case of failure:
  A message to librarian about the same

- Actors:
  Librarian communicates with the system

- Trigger:
  Librarian chooses the option to delete member

- Main Success Scenario:
  The user previously has an account

- Order to print reminder

  - Preconditions:
    1.Librarian must be logged in
    2. A book issued by a member must be overdue

  - Postcondition:
    A message is sent to the user.

  - Failure Situations:
    There are no overdue books

  - Postcondition in case of failure:
    A message to librarian about the same

  - Actors:
    Librarian communicates with the system

  - Trigger:
    Librarian chooses the option to print reminder

  - Main Success Scenario :
    There are some overdue books

- Plan to dispose books

  - Preconditions:
    1.Librarian must be logged in
    2. The book must not have been issued even once for 5 years

  - Postcondition:
    The book is disposed with a message to the library clerk to delete it.

- Actors:
  Librarian communicates with the system

- Trigger:
  Librarian chooses the option to dispose book

- Main Success Scenario:
  The book has not been issue for 5 years

4. System Use Cases:

- Answer availibilty Query about Book

  - Preconditions:
    1.An user makes a query

  - Postcondition:
    If the book is available, use cases display rack number and number of copies are called

  - Actors:
    System communiates with the user

  - Trigger:
    User chooses the option to search books

- Display rack number of book

  - Preconditions:
    1.If the book is available the use case answer availibility query invokes this

  - Postcondition:
    Rack numbers are displayed

  - Actors:
    System communiates with the user

  - Trigger:
    answer availibity query triggers this

- Display number of copies of book

  - Preconditions:
    1.If the book is available the use case answer availibility query invokes this

- Postcondition:
  the number of copies of a book are displayed

- Actors:
  System communiates with the user

- Trigger:
  answer availibity query triggers this

- Calculate lateness penalty

  - Preconditions:
    1.If the book is overdue, return book invokes this

  - Postcondition:
    Penalty is calculated and print bill is invoked

  - Actors:
    System communiates with the user

  - Trigger:
    return book query triggers this

- Provide book issue statistics
  - Preconditions:
    1.The use case is invoked by plan to dispose books

  - Postcondition:
    Statistics of books is displayed

  - Actors:
    system communicates with librarian

  - Trigger:
    Plan dispose book is invoked

5. Printer use cases:

   - Print bill of penalty

     - Preconditions:
       Some user must have returned the issued book later than his designated return date.

     - Postcondition:
       Bill of penalty is printed

  - Actors:
    Printer communiates with the system

  - Trigger:
    Calculate lateness penalty triggers print bill

- Print reminder

  - Preconditions:
    Some user must have exceeded the due date

  - Postcondition:
    Reminder to user is printed

  - Actors:
    Printer communiates with the system

  - Trigger:
    order to print reminder triggers this

- Print notification on return of reserved book

  - Preconditions:
    Some user must have returned a reserved book

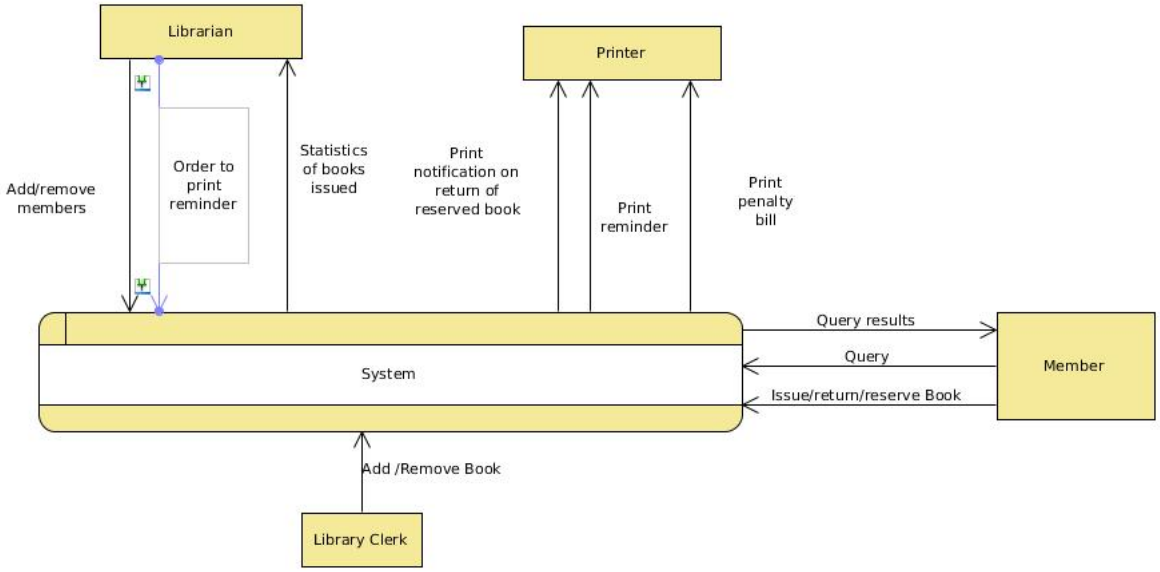
  - Postcondition:
    A notification is printed to the user who reserved the book

  - Actors:
    Printer communiates with the system which communicates with user

  - Trigger:
    return book may trigger this use case

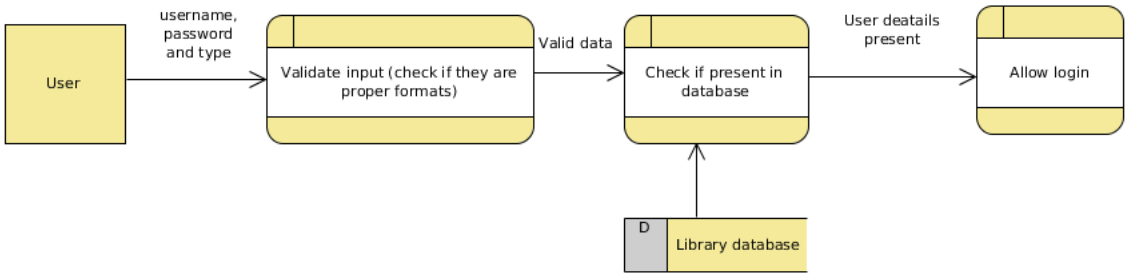
Data Flow Diagram

- Context Diagram

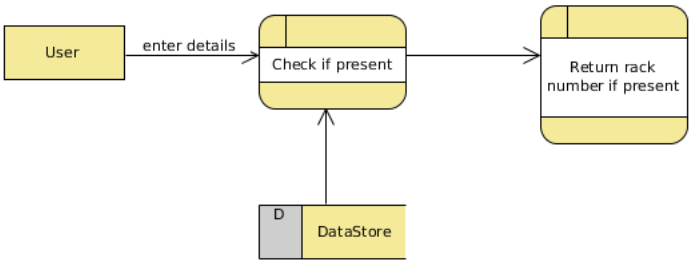


- Level 1 DFD

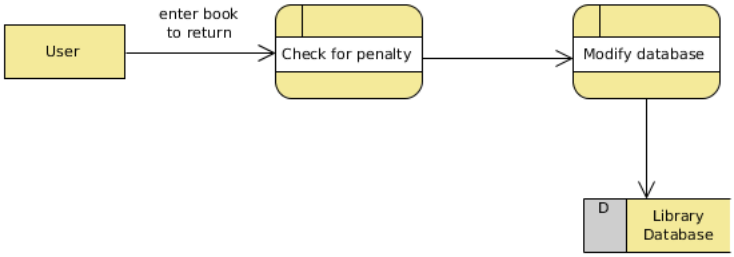
  – Diagram for login :



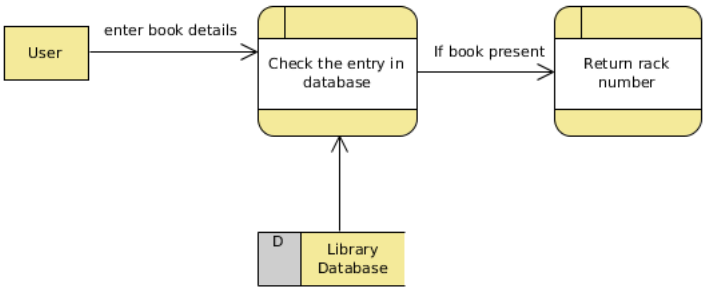
  – Diagram for issuing books :



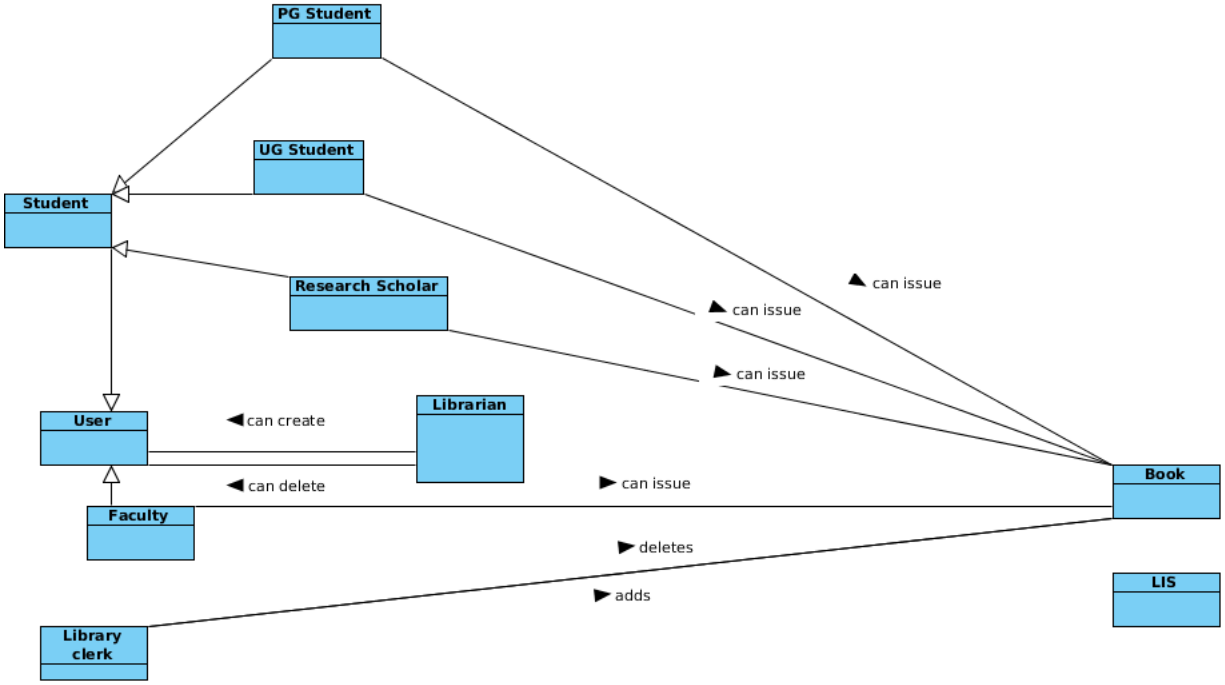
– Diagram for returning books :



– Diagram for searching books :



2.3 User Classes and Characteristics



- Book
  - Class Attributes:

    Name:Store the name of book
    No_of_copies:Stores the number of copies of a book
    Penalty_Rate:the rate of penalty for book.A static member as it is common for all the books
    Author:stores name author of book
    LastIssueDateDate:Stores the last issued date of book
    NoOfCopiesIssued:Stores the number of copies of the book that have been issued out
    NoOfCopiesReserved:Stores the number of issued books that have been reserved
    NoOfBooks:A static variable storing the total number of books
    IDsofIssuedMembers[]:An array of integers storing the ids of all the members who have issued copies of the book
    IDsOfMembersReserved[]:An array of integers storing the ids of all the members who have issued copies of the book
    RackNo:stores the rack number where the book is kept
  - Operations:
    Getter functions for name,No_of_copies,ISBN number,Penalty_rate,LastIssueDate,author,NoOfCopiesIssued,NoOfC
    We have used getters for these as we need to view these attributes from outside
    Setters for No_of_copies,LastIssueDate,NoOfCopiesIssued,NoOfCopiesReserved,IDsOfIssuedMembers, IDSOfMembersReserved,RackNo.
    We have used setters for these as these are changeable with time and need to be changesd at a later point of time.These being private members setters ar only way to modify them

- Library Member
  - Attributes:
    - Name:Name of the user
    - ID_number:Login id of the user
    - Number of books in account:Total number of issued books
    - Number of reserved books:Total number of books issued by the user
    - NoOfUsers:A static variable storuing total number of users
    - ISBNnosOfIssuedBooks[]:an array storing the ISBN number of all books issued by the user
    - ISBNnosOfReservedBooks[]:an array storing the ISBN number of all books reserved by the user
    - maximumNoOfBooksIssued:the maximum number of books that the user can issue
    - Pasword:The login password of the user which is necessary for login authentication
  - Operations:
    - Constructor to initialize member
    - Getters for NumberOfBooksinAccount,No ofReservedBooks,NoOfUsers. We have used getters for these as we need to view these attributes from outside
    - Setters for NumberOfUsers,Number of books in account,No Of reserved Books.
      We have used setters for these as these are changeable with time and need to be changesd at a later point of time.These being private members setters ar only way to modify them
    - IssueBook:called when member tries to issue a book
    - SearchBook:Called when member tries to search fo a book
    - Reserve Book:Called when member tries to reserve a book
    - ReturnBook:called when member tries to return a book
    - PayFine:Called when member returns an overdue book

- Library Clerk
  - Attributes:
    - UserId:User Id of the library Clerk
    - Password:Password of the library Clerk
  - Operations:
    - AddBook:Adds a new procured book to database

        * Removebook:Deletes a disposed book from database

- Librarian
  - Attributes:
    * UserId:User Id of the librarian
    * Password:Password of the librarian
  - Operations:
    * AddUser:Adds a new user account
    * DeleteUser:Deletes an existing user account
    * getStatistics: asks for statistics from LIS
    * DisposeBooks:Disposes a book not issued in 5 years
    * OrderReminder:Orders LIS to print reminder on overdue books
- LIS
  - Operations:
    * ApproveIssue:Called when user tries to issue a book
    * ApproveReturn:Called when user tries to return a book
    * ApproveReserve:Called when user tries to reserve a book
    * CalculatePenalty:Calculates penalty on overdue book
    * PrintBill:prints penalty Bill
    * PrintReminder:prints Reminder on overdue books
    * Show Statistics:Display statistics of books issued
- Faculty
  It is a generalization of Library User
- Student
  It is a generalization of Library User
- Research Scholar
  It is a generalization of Student
- UG Student
  It is a generalization of Student
- PG Student
  It is a generalization of Student

2.4 Operating Environment

The software is designed to run on the following environments :

- Windows 7 or later
- Ubuntu 14.04.03 (LTS version)

Any OS with the support of Java JDK and JRE libraries 1.7 and above can be used to execute the software.

2.5 Design and Implementation Constraints

This software is optimized to run on a maximum of 10000 books . Exceeding this number can result in a slower running time of the algorithm and thus the execution may not fruitful enough.

2.6 User Documentation

User manual :

User Login

Any user with a valid username and password will be able to login to the software to access the features. Initially just after the installation of the software only a librarian can be created in the software. Upon thd creation of the librarian the librarian can then create the other users in the library.

- For librarian :
  If a user logs in as librarian he is then shown a librarian home page screen with the functionalities exclusively of a libraraian.
- For clerk :
  If user logs in as clerkhe is then shown the clerk home page with the functions like adding or deleting book etc.
- For others(members) :
  We switch on to a new screen which is the home screen of the member. The member has the options of issuing, reserving and returning a book.

User creation

The librarian can be created in the beginning when the software is first installed. Once a librarian is creates an new librarian can not be created until and unless the old one is removed.
Only the librarian can create a new member of the library like the faculty or the students. During the creation of a member the librarian should also provide the type of the user as the maximim book count is different for different categories of the users.Once the type of a library member is given during creation one does not need to provide it anywhere else during issuing or returning of a book as it will be saved in the database and automatically fetched when required.
There are several rules while creating the user :

- The username can consist of alphanumeric characters along with an underscore (_)
- The password needs to consist atleast one of each type i.e. atleast one alphabet , atleast one number and atleast one special character.
  The password needs to atlest 8 characters long to ensure good security.
- The user type also needs to be provided.

Librarian Home Screen

The librarian hold the administrator access of the library.The feature available to him are as follows :

- Creation of user :
  For this purpose the librian will be directed to a new screen where he will have to provide the necessary details in order to create a new account and he has to provide the member type during account creation.
- Removing user :
  The librarian will provide with the details of the user to remove . If clicked to remove the details of the corresponding user will be removed from the database if and only if all his issued books are returned and there is no penalty due for him.

- Print notifications:
  The librarian can access any user history an can send a request or notification of printing the same
- Notification to print penalty bill :
  The librarian can send message to the printer to print the penalty bill.
- Notification to clerk to remove book :
  The librarian can also issue a request to the clerk to remove a book from the kibrary database is he finds it necessary.

User home screen

- Issue book:
  The user can issue book from this portal if the book is present in the library at that point of time.
- Return book :
  The user can return the book earlier issued via a portal.It also checks if the book is returned in time and if any fine is due.
- Reserve book :
  The user can reserve any book if it is not present at that point of time and the user will not exceed his maximum issue count.
- Search book :
  The user can search for any book in the library database by providing any details of the book.If it is present the software will display the rack number of the location of the book.

Clerk home screen

- Add book :
  The clerk can add a book into the library database with all the relevant details
- Remove book :
  The clerk can remove any old and unused book from the library and remove all its details from the database.

2.7 Assumptions and Dependencies

We have assumed that the user can cancel his reservation any time later if he/she feels that the book is not required for issuing. Also the other things that have been taken care of are :

- ISBN number is same for all copies of a book.They are differentiated by a seperate copy ID
- For penalty date calculation a "month" is not just 30 days.It is from a date of a month to the same date of another.For instance, one month from 5th February is 5th March.
- The user has to return his issued books before he can reserve books.

  The software depends on the java awt and swing packages
  Also for the database support we have used Mysql which is enabled by the java sql libraries.
  So the software depends on these above mentioned packages and without these it wont execute.

3 SYSTEM FEATURES

3.1 User login

The software should allow the user to provide the details of a user to login into the system.

3.2 Add user

The librarian is allowed to create a new user of the library who can have the facilities like issuing , searching and returning of a book.
The librarian gives the valid details in the fields to make the user and adds it to the database

3.3 Remove user

The librarian who hold the administrator acces to the software can remove any user from the user if need arises.All the records of the user will be removed from the databse maintained in the software.

3.4 Search book

A library user(faculty,research scholar,UG or PG student) can search a book to see if it is available in the library for issuing.

3.5 Issue book

A library user(faculty,research scholar,UG or PG student) can issue a book which is available in the library only if he/she has not exceeded the maximum issue count aginst his/her account.

3.6 Return book

A library user(faculty,research scholar,UG or PG student) can return a book which is available in the library only and is liable to pay fines in case of overdue.

3.7 Give penalty

A library user(faculty,research scholar,UG or PG student) can pay the fine if he/she has aprevious overdue dute to late returning of a book.

4 EXTERNAL INTERAFCE REQUIREMENTS

4.1 User Interfaces

4.2 Hardware Interfaces

1. The software is run offline so hard disk with sufficient space of about — is required to store and save the records for further use

2. Keyboard and mouse to interact with the GUI

4.3 Software Interfaces

1. Back end : Built using Java and MySQL (for DBMS)
2. Front end : Using Java

4.4 Communication Interfaces

The databases can be kept in a separeate location other than the source computer itself to make it more space efficient.It can be done by using tcp/ip protocol for network access.
It is not mandatory for this specific design but can be taken up as a feature which can be appiled to make the software more realistic and user friendly.
This will not only make the software more space efficient but at the same time maintain data security since we will be using a secured channel for data communication.

5 OTHER NONFUNCTIONAL REQUIREMENTS

5.1 Performance Requirements

- With the Library Information System the librarian and library clerk can process a book transaction in a faster speed .
- The members will be able to check the status of the checked out items and can also borrow and return a book in a short amount of time.
- Automatic updates in the account of each user will enable them to know about the item they have checked out and all the relevant details due to which they do not need to consult or ask the library clerk each time.
- The background system admin will automatically perform the above mentioned updation in the accounts in the background autonomously without explicit invocation.
- In case of defaulters the fine and other penalties should be automatically reported and saved in the record once the deadline has crossed . This is also performed by the syatem admin (referred as LIS itself) in the background.
- ALl the background tasks should not compromise with the UX of the software.All the background processes should be methodically handled and not affect the runtime functionalities of the software

5.2 Safety Requirements

Only the valid users can change the records that too only in their respective account.In case any invalid user tries to tamper with the data a error message will be flagged thus aborting the action.
In this type of software safety is a necessary thing that needs to be taken care of.For this absolute care needs to be taken to check the validity of user before tampering any information in the database.
Also a seperate copy of the database needs to be maintained so that a backup of the dynamic data is available all the time if required. In case the software breaks down due to a glitch all the data should

not be lost.

5.3 Security Requirements

The sytem must be highly secure in the login part.
No user should be able to access a domain outside his reach.
Care must be taken to keep the user accounts secure and all the book transaction should be properly updated to the corresponding user.

5.4 Software Quality Requirements

5.4.1 Reliability requirements

The software is extremely reliable and has a lightweight design.It uses as minimum resources as possible but still it provides a quality support for the intended software requirements. Multiple copies of the database have been kept at sepearte locations to ensure that the data is not lost in case of a serious software glitch. These backup datasets can be recovered easily by a file reader to get back the lost data. The background database should always be updated so that whenever a member visits he/she can get the latest information required.Also the duplicate or the backup databases must also be updated at the same time so that no discrepancy is there among the parallel databases.
For every process that the software can do the steps are quite clearly framed and unambiguous in nature so that the user will not face any problem in doing the job.

5.4.2 Usability Requirements

The software must be user-friendly so that the user can easily perform all the tasks which the software is meant to do.It must have a soothing UX design and clear instructions to guide the user.
In case of any error suitable error messages must be displayed to assist the user.Also a user manual is provided with the glimpses of the software design to show how the softwar eis supposed to work.
Care has been taken to make the software process very fast and reliable.The running time of the software and the space complexity have been optimized as far as possible to keep the software more realistic and usable

6 OTHER REQUIREMENTS

The other thing which is required to make the back-end of the software is sql support. The sql framework used behind this software is MySQL for JAVA. Using the packages the sql support has been estblished.
Use of sql is there to ensure proper database facilities rather than just using normal text files which might have also served our purpose.Use of proper databse structure greatly reduces the running time making it logarithmic from linear with respect to the number of records.