

## Part I

# System Analysis

## 1 Feasability Study

### 1.1 Understanding the problem

In this section, we make an effort to understand the purpose of the software. The LIS (Library Information Software) is intended to ensure a systematic manner of maintaining the database of a library and easing the job of a librarian. The LIS will help maintain a record of books in the library and members registered with the library. The software aims at hassle-free handling of basic library functions like issue/reserve/return books as well as giving the freedom to the librarian to add and delete books and members to the database. Extending further the software can be used by the librarian to decide on disposing of books which have not been issued for a long time. This software also helps to keep track of statistics of issue of a particular book over a period of time. Since everything is done on the computer, it is easy to record all data and there are minimal chances of inconsistencies or ambiguities. Also use of a secure database support enhances the security aspects of the software thus enabling to create a more robust and secure software.

### 1.2 Scope of the Problem

In this section, we list the various functions performed by the software:

- Add books to the database
- Modify existing records of books
- Delete unused books from database
- Add users to the database
- Modify existing details of users
- Remove user from the database
- Query regarding books on ISBN number or name
- Query regarding user on user id or name
- Record issue, return and reservation of books
- Calculation of penalty on overdue books
- Maintenance of statistics of all books

### 1.3 Analysing stakeholders

The various stakeholders are:

- Librarian
- Library Clerk
- Member / Library user
  - Faculty
  - Research Scholar
  - PG Student
  - UG Student

These stakeholders have been clearly described with their attributes and methods in the SRS.

### 1.4 Defining Alternatives

#### 1.4.1 Connection between software and librarian

The librarian is treated as the administrator of the software who has the sole capability to create and delete the members like the faculty or the students (including research scholars,UG and PG students).

The librarian can send a print notification if required if a penalty is required to be paid in case of overdue of a book

Also the librarian can study the book statistics and can send a notification to the clerk to dispose off old and unused books from the library database

#### 1.4.2 Connection between software and member

The members can perform the following operations:

- Can issue a book if available in the library
- Can

#### 1.4.3 Hardware Infrastructure

1. Processor Pentium II processor or higher
2. Hard Disk space 500MB
3. RAM 512 MB
4. Network/Internet access

#### **1.4.4 Software Infrastructure**

1. Operating system
  - Windows 7 or later
  - Linux distributions like Ubuntu 14.04.03 or other
2. MySQL
3. Java JDK platform 1.7 or higher

#### **1.4.5 Technology Used**

The technologies used are as follows :

- Use of the platform independent JVM using JDK 1.8
- MySQL support to provide DBMS features to the software

#### **1.4.6 Security**

The security of the software has been taken care of in a very cautious and judicious manner. The protocol followed to ensure data security and software safety are as follows :

- User can access the books only after a successful login with a valid username and password.
- Check has been done during creation of new user to ensure a unique user id.
- Also check has been done to set a minimum length of 8 characters and use of alphanumeric characters to make a password without storing any value (to prevent publicity of password) so that users can give a safe and efficient password to safeguard their account.
- Unsuccessful attempts may occur if the user gives wrong username or wrong password or even wrong user type.

### **1.5 Defining Criteria to evaluate**

The software is evaluated on the basis of the following criterias:

- Only the librarian can be created without any preprocessing. But only one librarian must be present.
- Addition and deletion of books can be performed by library clerk in this software
- Creation of users by the librarian can be performed effectively

- Users should be able to successfully login into the software
- Creation of effective users who can perform all the operations like issue, reserve and returning of a book

## 1.6 Report

The software designed is able to perform all the above operations mentioned above. One can create a single librarian at first who hold the admin access of the software. The librarian is able to create the users or the library members who can perform all the required operations as mentioned in the SRS.

The library clerk is able to add books into the library and can remove the un-required books from the database.

The login feature has also been created, tested and is working perfectly. Only the authenticated users can log into the software to perform the designated operations.

A background cron job takes care of all the background processes and helps to maintain all the records as required like in the case of issuing and returning. It also takes care of the issued books and takes care if a penalty is to be issued against any user account.

All the process as mentioned in the SRS document provided have been implemented and tested successfully.

## 2 UML Diagrams

### 2.1 Refinement of Use Case Diagram

### 2.2 Refinement of Class Diagrams

### 2.3 Sequence Diagram

### 2.4 Collaboration Diagram

### 2.5 Statechart Diagram

### 2.6 Activity Diagram

## 3 System Parameters

### 3.1 Platform

The software requires the following platforms to run : Minimum system requirements :

- Hardware Requirements :
  1. Processor Pentium II processor or higher

2. Hard Disk space 500MB
  3. RAM 512 MB
- Software Requirements :
    1. Operating system
      - Windows 7 or later
      - Linux distributions like Ubuntu 14.04.03 or other
    2. MySQL
    3. Java JDK platform 1.7 or higher

### **3.2 Language**

The entire software is written using the language Java. The object oriented programming paradigm of Java has been followed throughout.

### **3.3 Build System**

### **3.4 Libraries**

The libraries used for the software are as follows :

1. Java Swing (java.awt.\*)
2. Java AWT (javax.swing.\*)
3. Java Exception (to be included)

### **3.5 Sizing**

### **3.6 Performance**

## **4 Limitations and Exceptions**

The number of books which can be incorporated into the system has been kept to be constant in this case to ensure a simpler design. In case of a dynamically ncreasing book set the algorithms and the data structures used to mange have to be optimized further to keep the running time of the software as fast as possible.

Exceptions which are present or handled in this software are as follows :

## 5 Other Information about analysis

### Part II

## System Design

### 6 Refined System Parameters

#### 6.1 Global System Architecture

The overall system architecture is a 2-tier architecture which includes client at one end and the database at the other. There is no server based middle tier in the software being designed.

#### 6.2 Platform

##### 6.2.1 Hardware

The hardware platform required in this software is :

1. Processor Pentium II processor or higher
2. Hard Disk space 500MB
3. RAM 512 MB

##### 6.2.2 Software

The software platform of the software is fully developed in Java. For suitable execution Java JDK 1.7 or later is required. The database management has been done using a DBMS software like MySQL.

##### 6.2.3 Networking

The GUI support and the database support are interlinked via a tcp network which facilitates the flow of information between the two ends. It provides scope for dividing the space requirements by distributing the load. The use of tcp network ensures secure data flow between the two interfaces.

The tcp network identifies the two ends of the network and establishes a secured connection between the two ends. Keeping the data at a separate location can improve the safety of the software as it prevents data loss to a huge extent in case of a glitch in the GUI end.

Thus we can ensure a safe, secure, reliable and fast software design.

## **6.3 Software Architecture**

Object-oriented architecture forms the basis of the LIS. In this style data representations and their associated primitive operations are encapsulated in an abstract data type or object. The components of this style are the objects—or instances of the abstract data types. Objects interact through function and procedure invocations. Two important aspects of this style are A. that an object is responsible for preserving the integrity of its representation (usually by maintaining some invariant over it), and B. that the representation is hidden from other objects. Thus the aspects of OOA mentioned justify our choice.

### **6.3.1 Details**

### **6.3.2 Justification**

## **7 Database Design**

## **8 Design Details**

### **8.1 Refinement of UML diagrams**

### **8.2 Prototype Design**

### **8.3 Design I/O procedures and user interfaces**

### **8.4 Design of classes in target language**

### **8.5 Exception Design**

## **9 Adoptable Practices**

## **10 Any Other Information Of previous Stages**