CHAPTER-1

INTRODUCTION

CHAPTER 1

INTRODUCTION

_____

1.1 Introduction

A library requires a proper arrangement and placement of books. That makes it easy for the user to find a particular book. But in very large libraries which have a huge collection, locating a particular book becomes a very difficult task.

Though all books are arranged in categories, but the location of category and books must be known at first so that the user gets to know where that particular category of books is placed. A user also needs to know about books availability, borrowed and returned books status etc. They may need to see any book preview.

In such a condition there needs to be a way through user can precisely locate the location of any book seeing graphically by just typing its name and get all the library facilities in a server base android application. Here we are proposing a server based system using an android application to achieve this task using internet or Wi-Fi technology.

1.2 Objectives

The objectives of this project is to providing a server based android app, improving the library management system features and reducing the access time of users. As this is a server based application, the users will remain updated about his/her activities of library such as books availability, graphical book location, borrowed/returned book status and penalty/fine status, book reservation, notice of library, yearly book generate reports, etc. The library operator or librarian just needs to maintain the data server. For example: "The librarian can add or remove the book details, user details, issue/returned books details, etc. and the location of a book in the system such as "3rd row | 5th column" and a map of the book along with its status will show whether it is located." All this data is stored in the server. New members are provided an android application with an account which will be used for library related purpose. Users having internet or Wi-Fi connection will use the application and will get most of the library facilities in the app. The data server queries the database and show necessary

things in the app what users wants to know. Thus it allows to automate the library book finding, availability checking, and the others functionalities of a library.

1.3 Existing System

There are two existing systems of our project. They are-

1. Traditional Library System

Fig 1.1: Traditional Library System (1.3.1)

Generally libraries are the place to preserve and distribute the physical forms of resources, such as books and magazines, journals, periodicals to maintain these resources with cataloguing and classification. Physical Searching method is using to retrieve the resources. Information is stored in physical format. The users may be borrowed the resources and make use of it. A traditional library consist details of available stock in books and subscription of periodicals.
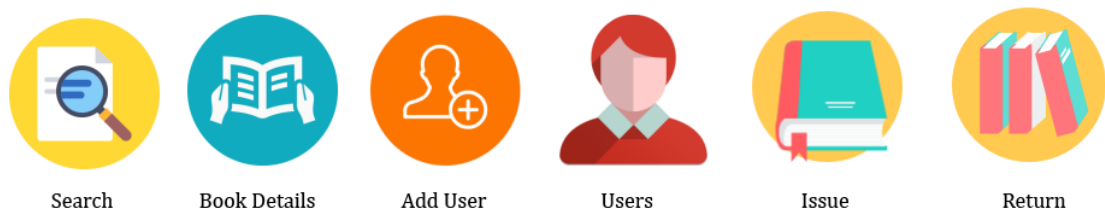
2. Web Based Library Management System

Search   Book Details   Add User   Users   Issue   Return

Fig 1.2: Web Based Library Management System (1.3.2)

In Web Based Library Management System, there are different things which are maintained. They are - Keep the record of different categories like; Books, Journals, Newspapers, Magazines, etc. Classify the books subject wise. Keep the record of complete information of a book like; Book name, Author name, Publisher's name, Date/ Year of publication, Cost of the book, Book purchasing date/ Bill No., Fine

calculation for late returns, Different criteria for searching a book. Different kind of reports like; total no. of books, no. of issued books, no. of journals, etc. How many books are issued to a particular student, etc.

1.4 Problems of Existing System

There are some problems of existing system which are mentioned below.

1. The First one is user interface problem. The user interface should be more responsive.
2. A user can't find specific book location easily.
3. The book searching process is so time-consuming.
4. Book Issuing and Returning process is not better.
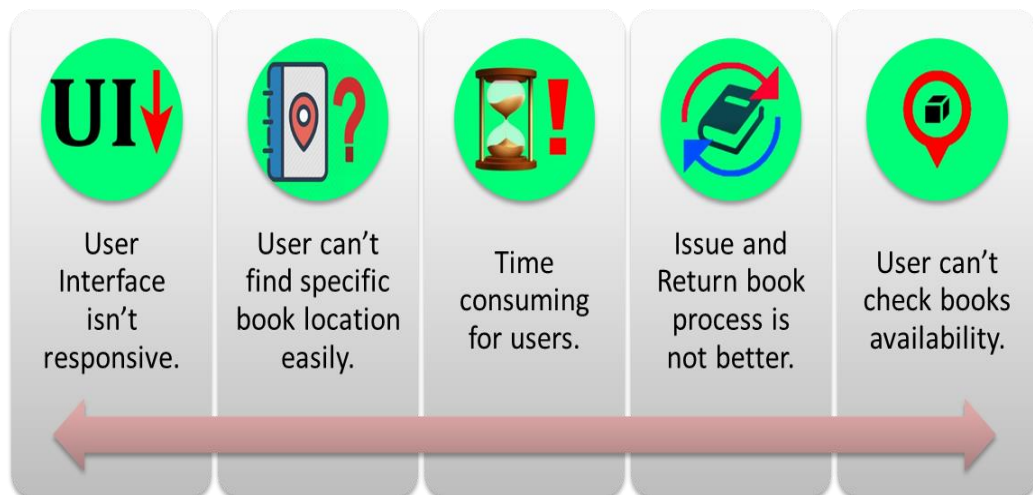5. User can't check book availability.

Fig 1.3: Drawbacks of existing system (1.4.1)

1.5 Proposed System

Proposed system aims at developing a Server based mobile application called "Library Book Tracker Android App" which can make the library system better than previous for the users. As this is a server based application, the users will remain updated about his/her activities of library such as graphical book location, borrowed/returned book status and penalty/fine status etc. The library operator just needs to maintain the data server.

Fig 1.4: Library Book Tracker Android App Icon (1.5.1)

1.6 Features of Proposed System

The key features of the proposed system are-

1. User can see book preview.

2. User can check book availability.

3. User can see the graphical book location.

4. User can reserve books and see status.

5. User can view borrowed book, return book (notification) and penalty status.

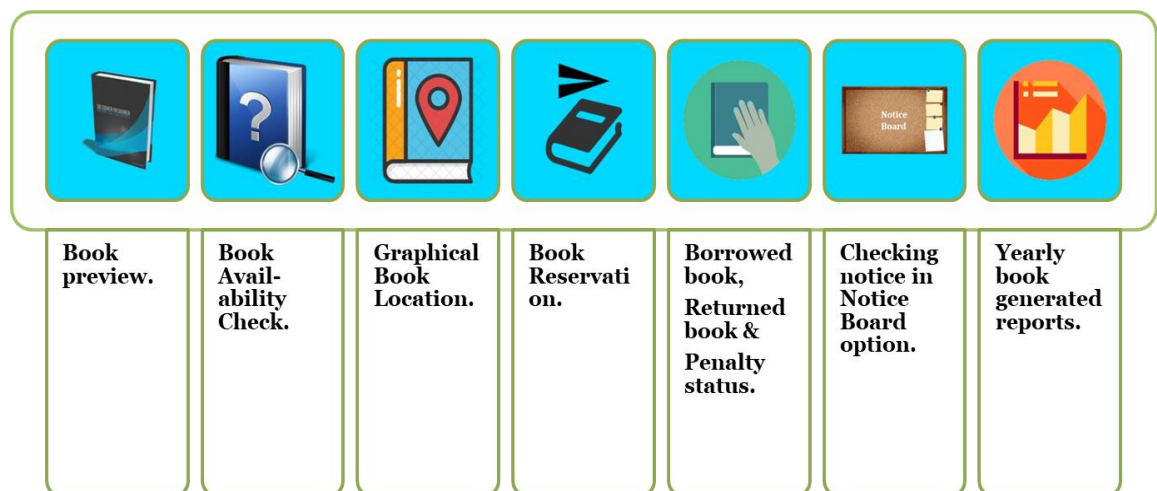6. User can check notice of library.

Book preview.	Book Avail-ability Check.	Graphical Book Location.	Book Reservati on.	Borrowed book, Returned book & Penalty status.	Checking notice in Notice Board option.	Yearly book generated reports.

Fig 1.5: Features of Proposed System (1.6.1)

1.7 Scope and Limitations of the Study

A. **System Features and Functionality**

This Library Book Tracker Android App is composed two major modules such as the **Admin Librarian Module** which indicates the **data server** and **Student/User Module** which indicates the **app**.

A.1. Admin Librarian Module

Account Information – allows admin librarian to edit personal information such as name, gender, username and password, etc.

Book Information – allows admin librarian to edit book information such as book id, name, author name, edition, book location, etc.

View Issued Book and Returned Book – allows the admin librarian to monitor borrowed books that due every day and which are returned every day. It will also allow the admin librarian to return and renew borrowed books.

View Reserved Book – allows the admin librarian to monitor requested/reserved books that due every day.

Open/Close Student/User Reservation – enables the admin librarian to close or open the book reservation in the student/user module.

View Statistics (yearly generated books report) – allows the admin librarian to view the statistics of borrowed book every month per institution year.

Notice Board – allows the admin to provide important notices.

Log-out – allows the admin librarian to sign-out of the admin librarian module.

A.2 Student/User Module

Search book by title, author, category – enables the students to search for the book by its title, author, category, etc.

Book Preview – enables the students to see book previews.

Book Availability – enables the students to check book preview.

Book Location – enables the students to see the book location map.

Reserve Book – enables the students to make a book reservation. Students can only perform this process only if he or she is logged-in in the app.

View reserved book – enables the students to view all the books that was being reserved.

View borrowed book and Returned Book Status– enables the students to view all the borrowed books and returned books status.

View Penalty Status – enables the students to view penalty status.

View Notice Board – enables the students to view notices.

Changing Password – enables the students to change password of their accounts.

Log-out – enables the students to sign-out of the student/user module.

B. Limitations

Due to the limited time given in the development of this project, some features are excluded in this system. Limitation of the system will be discussed below.

B.1 Student Module

1. Book reservation is only for the students/users. Instructors, faculties and staffs of the school has no access to make book reservation, they can only search for books in this module.

2. Book contents are not visible in this module; only the details of the books are available and can be viewed inside such as the author, title, subject, category, publisher and edition of the book.

1.8 Motivation

From the view of current system of library management system, there is no efficient software or solution which can easily manage all the library activities at a platform. This free Android application will make the library system better hopefully. You will have all the major functions of library in this app. This app will save the time of librarian as well as the students/users.

1.9 Conclusion

So, we have known about the Objectives, Existing System, Problems of Existing System, Proposed System, Features of Proposed System, Scope and Limitations of the Study and Motivation of our project in this chapter.

CHAPTER-2
FEASIBILITY STUDY &
PROPOSED SYSTEM

CHAPTER 2
FEASIBILITY STUDY & PROPOSED SYSTEM

_____

2.1 Introduction

In this chapter, we will know about the Feasibility Study and we will also know about whether the proposed system is feasible or not.

2.2 Feasibility Study

A feasibility study is an analysis of how successfully a project can be completed, accounting for factors that affect it such as economic, technological, legal and scheduling factors. Here we use feasibility studies to determine potential positive and negative outcomes of a project before investing a considerable amount of time and money into it.

The key considerations of feasibility are:

1. Economic Feasibility

2. Technical Feasibility

3. Operational Feasibility

2.3 Economic Feasibility

It looks at the financial aspect of the project. It determines whether the management has enough resources and budget to invest in the proposed system and the estimated time for the recovery of cost incurred. It also determines whether it is worthwhile to invest the money in the proposed project. Economic feasibility determines by the means of cost benefit analysis. The proposed system is economically feasible because the cost minimizes in hardware purchasing. The operating environment costs are marginal. the less time involved also helped in its economic feasibility. The Business Case provides an analysis of the business environment including, but not limited to, a description of who the expected customers are, the nature of the business, how the payment is currently being processed, if applicable, and the current and expected volume and timing of transactions. The Business Case also presents the benefits of the proposed project.

2.4 Technical Feasibility

Technical feasibility is one of the first studies that must be conducted after a project has been identified. In large engineering projects consulting agencies that have large staffs of engineers and technicians conduct technical studies dealing with the projects. This project is strongly technically feasible. Because it needs less hardware & technical equipment. Technical feasibility takes of the all issues concerned with the design and development part of the project. It concerns itself with the software, hardware and the platform related issues.

2.5 Operational Feasibility

Operational feasibility is the measure of how well a proposed system solves the problems, and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development. The system will be used if it is developed well then be resistance for users that undetermined:

- No major training and new skills are required of users.
- It will help in the time saving and fast processing.
- Faster systematic processing of user application approval. User had greater chances of error due to wrong information.
- Easy UI for controlling.

2.6 Proposed System, It's Features and Feasibility

Proposed system aims at developing a **Server based mobile application** which can make the **library** system better than previous for the users. As this is a server based application, the users will remain updated about his/her activities of library such as graphical book location, borrowed/returned book status and penalty/fine status etc. The library operator just needs to maintain the data server.

The features of the proposed system are-

1. Book preview.

2. Availability.

3. Graphical book location.

4. Book request for reservation.

5. Borrowed book, returned book and penalty status.

6. Notice of library.

For this project, we need android studio pc software to develop the app and the students/users need an android smartphone to use the app which is available to all now-a-days. So the App is feasible for all the users.

2.7 Conclusion

So, we have known about the Feasibility Study, Economic Feasibility, Technical Feasibility, Operational Feasibility, Proposed System and it's Features & Feasibility in this chapter.

CHAPTER-3
SYSTEM ANALYSIS

CHAPTER 3
SYSTEM ANALYSIS

_____

3.1 Introduction

In this chapter, we will know about the Feasibility Study and we will also know about whether the proposed system is feasible or not.

3.2 Problem Statement

Nowadays, people are using Smart phones so much than a Computer. So it is time consuming for those who are using web based library management system. The users need to start the computer and use the web based library management system. Yes, users can run this web based library management system in mobile browser but the interface is not seemed suitable as like in a computer. As Smart phones are always near to hand of users, they can easily get all the library facilities in this single app. The main purpose of the app is to facilitate interactions between library and the users of the app.

3.3 Requirement Analysis

Requirements analysis, also called requirements engineering, is the process of determining user expectations for a new or modified product. These features, called requirements, must be quantifiable, relevant and detailed. In software engineering, such requirements are often called functional specifications.
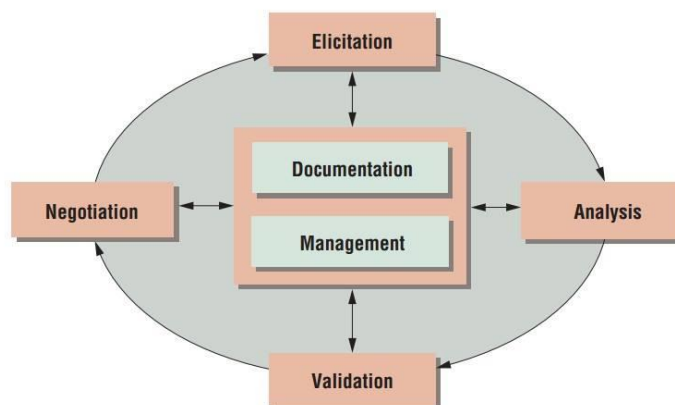

Fig 3.1: Requirement Analysis (3.3.1)

3.4 Functional and Non-functional Requirements

The functional requirement is describing the behavior of the system as it relates to the system's functionality. The non-functional requirement elaborates a performance characteristic of the system. The functional requirements of our project are the librarian and the app users. The non-functional requirements are accessibility, efficiency, usability, performance, reliability, maintainability, availability, portability, safety requirements, security requirements etc.

3.5 Software Requirements

A software requirements specification (SRS) is a description of a software system to be developed. It lays out functional and non-functional requirements, and may include a set of use cases that describe user interactions that the software must provide. The software requirements for our project are:

- ❑ Operating System- Windows 7, 8 or 10 and Linux Ubuntu 16.04.
- ❑ Android Studio IDE (Integrated Development Environment) v2.2.2 or more.
- ❑ Android SDK (Software Development Kit) v22.0 or more.
- ❑ Android Emulator or Android Virtual Device (AVD).
- ❑ JDK (Java Development Kit) v8.0+.
- ❑ Android API (Application programming Interface)-19 (4.4 Kit Kat) or more.

3.6 Hardware Requirements

- ❑ Memory of 4 GB RAM or more
- ❑ Monitor resolution of 1024 x 768 or higher
- ❑ Intel Pentium 4th generation processor or higher.
- ❑ 200GB (or more) available hard disk space.

3.7 Programming Language & Others

JAVA & Android programming languages, XML for app design, MySQL or SQL for database, phpMyAdmin for server. These are used to develop the App.

3.8 App User Requirements

- ❑ Android Smartphone (Minimum Android Version: Kit Kat 4.4.2 or Higher).
- ❑ User Profile/Account.
- ❑ Internet or Wi-Fi Connection.

3.9 Process Model Used

The ultimate objectives if software engineering is to produce good quality maintainable software within reasonable time frame and at affordable cost. This is achievable only if we have matured processes to produce it. For a mature process, it should be possible to determine in advance how much time, cost and effort will be required to produce the final product. This can only be done using data from past experience, which requires that we must measure the software process. A key component of any software development process is the life cycle model on which the process is based. Life cycle of the software starts from concept exploration and ends at the retirement of the software.
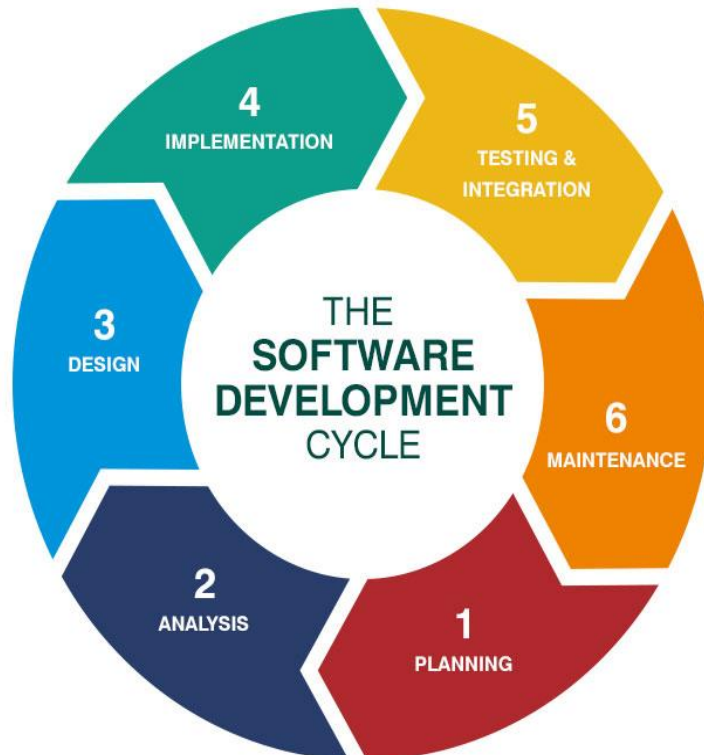

Fig 3.2: SDLC - The Software Development Life Cycle (3.9.1)

The system life cycle is the period of time that starts when a software product is conceived and ends when the products are no longer available for use. The software life cycle typically includes a requirement phase, design phase, implementation phase, test phase, installation and check out phase, operation and maintenance phase, and sometimes retirement phase. This system is developed mainly following Agile Software Development Methodology.

3.10 Agile Software Development

Agile software development describes a set of values and principles for software development under which requirements and solutions evolve through the collaborative effort of self-organizing cross-functional teams.
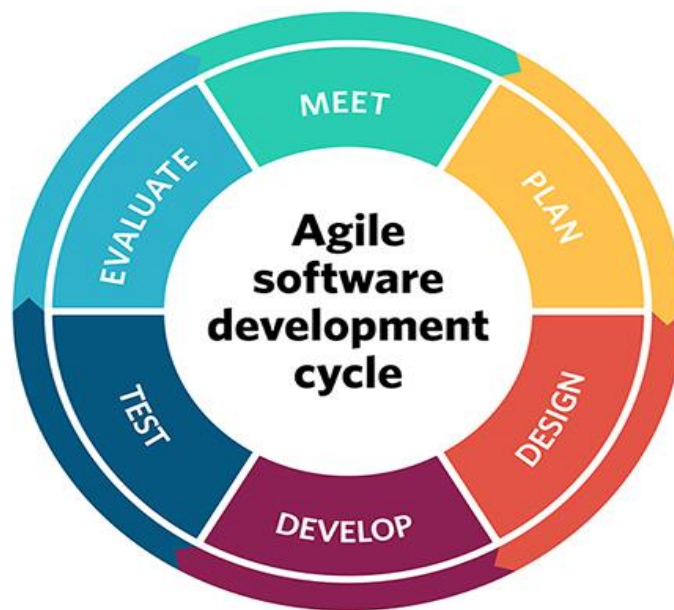

Fig 3.3: Agile Software Development Cycle (3.5.1)

Main features of an Agile Software Development life cycle model:

1. Individuals and interactions

In the Agile software development life cycle model, self-organization and motivation takes precedence over delegation of authority and following the "seniority" hierarchy. Team members are encouraged to take an active part in the development and planning activities. They are also "empowered" to take certain decisions on their own. The

Agile team has to collaborate and share ideas to develop the product "as a whole" unit i.e. each member should support a common vision.

2. Working software

Agile concentrates upon delivering sustained "working" product releases through product incremental cycles over documentation and working protocols. The main objective is to develop, and deliver, bug free product feature releases in a continuous and sustained manner until the entire product is developed.

3. Customer/user collaboration

Since all the requirements pertaining to product development may not be available, or "acquirable", at the project start up time owing to various factors, development should commence almost "immediately", and presented to clients for verification purposes. Stakeholders and project owners "clear" the product features developed through the sprint cycles. A lot of time is saved through customer collaboration, and as a result, the project proceeds in a successful manner as the client always Okays the development keeping in mind the current market trends.

4. Responding to changes

Agile focuses upon incorporating dynamic changes in the product development cycle. Changes in the product features can be easily and effortlessly carried out by developing "user stories" – product functionality or features as defined in the product backlog. Changes can be carried out at any time while the features are being developed – even late in the product development cycle.

Advantages of Agile model:

- ❑ Customer satisfaction by rapid, continuous delivery of useful software.
- ❑ People and interactions are emphasized rather than process and tools. Customers, developers and testers constantly interact with each other.
- ❑ Working software is delivered frequently (weeks rather than months).
- ❑ Face-to-face conversation is the best form of communication.
- ❑ Close daily cooperation between business people and developers.

- ❑ Continuous attention to technical excellence and good design.
- ❑ Regular adaptation to changing circumstances.
- ❑ Even late changes in requirements are welcomed.

Disadvantages of Agile model:

- ❑ In case of some software deliverables, especially the large ones, it is difficult to assess the effort required at the beginning of the software development life cycle.
- ❑ There is lack of emphasis on necessary designing and documentation.
- ❑ The project can easily get taken off track if the customer representative is not clear what final outcome that they want.
- ❑ Only senior programmers are capable of taking the kind of decisions required during the development process. Hence it has no place for newbie programmers, unless combined with experienced resources.

Reason for Use

- The customer has frequent and early opportunities to see the work being delivered, and to make decisions and changes throughout the development project.

- The customer gains a strong sense of ownership by working extensively and directly with the project team throughout the project.

- If time to market for a specific application is a greater concern than releasing a full feature set at initial launch, Agile can more quickly produce a basic version of working software which can be built upon in successive iterations.

- Development is often more user-focused, likely a result of more and frequent direction from the customer.

3.11 Used Technology & Tools

This project requires the following technology & tools in order to function properly.

JAVA:

Java is a general-purpose computer-programming language that is concurrent, class-based, object-oriented, and specifically designed to have as few implementation dependencies as possible. It is intended to let application developers "write once, run

anywhere" (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation. Java applications are typically compiled to "byte code" that can run on any Java virtual machine (JVM) regardless of the underlying computer architecture. The language derives much of its original features from Small Talk, with a syntax similar to C and C++, but it has fewer low-level facilities than either of them. As of 2016, Java was one of the most popular programming languages in use, particularly for client-server web applications, with a reported 9 million developers.

Java was originally developed by Canadian James Gosling at Sun Microsystems (which has since been acquired by Oracle) and released in 1995 as a core component of Sun Microsystems' Java platform. The original and reference implementation Java compilers, virtual machines, and class libraries were originally released by Sun under proprietary licenses. As of May 2007, in compliance with the specifications of the Java Community Process, Sun had relicensed most of its Java technologies under the GNU General Public License. Meanwhile, others have developed alternative implementations of these Sun technologies, such as the GNU Compiler for Java (byte code compiler), GNU Class path (standard libraries), and Iced Tea-Web (browser plugin for applets).

The latest version is Java 11, released on September 25, 2018, which is a currently supported long-term support (LTS) version by Oracle. Since Java 9 is no longer supported, Oracle advises its users to "immediately transition" to Java 11. Oracle released the last public update for the legacy Java 8 LTS, which is free for commercial use, in January 2019. Java 8 will be supported with public updates for personal use up to at least December 2020. Oracle and others "highly recommend that you uninstall older versions of Java" because of serious risks due to unresolved security issues.

Android OS:

Android is a mobile operating system developed by Google. It is based on a modified version of the Linux kernel and other open source software, and is designed primarily for touchscreen mobile devices such as smartphones and tablets. In addition, Google has further developed Android TV for televisions, Android Auto for cars and Wear OS for wrist watches, each with a specialized user interface. Variants of Android are also used on game consoles, digital cameras, PCs and other electronics.

Initially developed by Android Inc., which Google bought in 2005, Android was unveiled in 2007, with the first commercial Android device launched in September 2008. The operating system has since gone through multiple major releases, with the current version being 9 "Pie", released in August 2018. Google released the first Android Q beta on all Pixel phones on March 13, 2019. The core Android source code is known as Android Open Source Project (AOSP), and is primarily licensed under the Apache License.

Android is also associated with a suite of proprietary software developed by Google, called Google Mobile Services (GMS) that very frequently comes pre-installed in devices, which usually includes the Google Chrome web browser and Google Search and always includes core apps for services such as Gmail, as well as the application store and digital distribution platform Google Play, and associated development platform. These apps are licensed by manufacturers of Android devices certified under standards imposed by Google, but AOSP has been used as the basis of competing Android ecosystems, such as Amazon.com's Fire OS, which use their own equivalents to GMS.

Android has been the best-selling OS worldwide on smartphones since 2011 and on tablets since 2013. As of May 2017, it has over two billion monthly active users, the largest installed base of any operating system, and as of December 2018, the Google Play store features over 2.6 million apps. As of 2019, Android has about 500 million monthly active users, and it has the largest installed base of any operating system.

<u>Android Studio</u> -

Android Studio is the official integrated development environment (IDE) for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, macOS and Linux based operating systems. It is a replacement for the Eclipse Android Development Tools (ADT) as the primary IDE for native Android application development.

Android Studio was announced on May 16, 2013 at the Google I/O conference. It was in early access preview stage starting from version 0.1 in May 2013, then entered beta stage starting from version 0.8 which was released in June 2014. The first stable build was released in December 2014, starting from version 1.0. The current stable version is 3.3, which was released in January 2019.

phpMyAdmin Server:

Here, phpMyAdmin is used for data server. All the databases are stored in the data server.

XAMPP:

XAMPP is a free and open source cross-platform web server solution stack package developed by Apache Friends consisting mainly of the Apache HTTP Server, Maria DB database, and interpreters for scripts written in the PHP and Perl programming languages XAMPP stands for Cross-Platform (X), Apache (A), Maria DB (M), PHP (P) and Perl (P). It is a simple, lightweight Apache distribution that makes it extremely easy for developers to create a local web server for testing and deployment purposes. Everything needed to set up a web server – server application (Apache), database (Maria DB), and scripting language (PHP) – is included in an extractable file. XAMPP is also cross-platform, which means it works equally well on Linux, Mac and Windows. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server extremely easy as well.

SQL:

SQL (pronounced "ess-que-el") stands for Structured Query Language. SQL is used to communicate with a database. According to ANSI (American National Standards Institute), it is the standard language for relational database management systems. SQL statements are used to perform tasks such as update data on a database, or retrieve data from a database. Some common relational database management systems that use SQL are: Oracle, Sybase, Microsoft SQL Server, Access, Ingres, etc. Although most database systems use SQL, most of them also have their own additional proprietary extensions that are usually only used on their system. However, the standard SQL commands such as "Select", "Insert", "Update", "Delete", "Create", and "Drop" can be used to accomplish almost everything that one needs to do with a database. This tutorial will provide you with the instruction on the basics of each of these commands as well as allow you to put them to practice using the SQL Interpreter.

3.12 Conclusion

So, we have known about the Problem Statement, Requirement Analysis, Functional and Non-functional Requirements, Software Requirements, Hardware Requirements, Programming Language & Others, App User Requirements, Process Model Used, Agile Software Development, Advantages of Agile model, Disadvantages of Agile model, Reason for Use and Technology & Tools in this chapter.

CHAPTER-4
DESIGN

CHAPTER 4
DESIGN

_____

4.1 Introduction

This chapter contains the designing part of the application. It describes all the design architectures of this app. For Example: the flow chart, the use case diagram, the E-R diagram, the database designs as the table diagrams.

4.2 Flow Chart

There are 4 flow charts for our project app. They are given below:
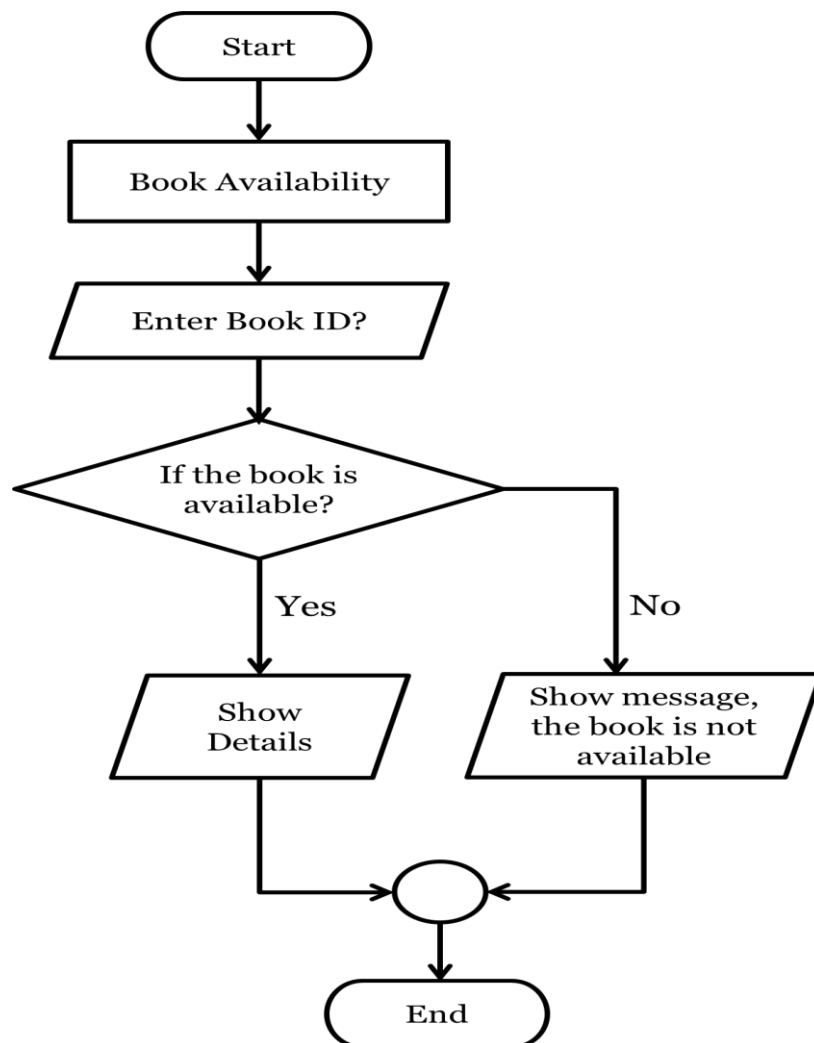
1. Book Availability flow chart.

Fig 4.1: Book Availability flow chart (4.2.1)
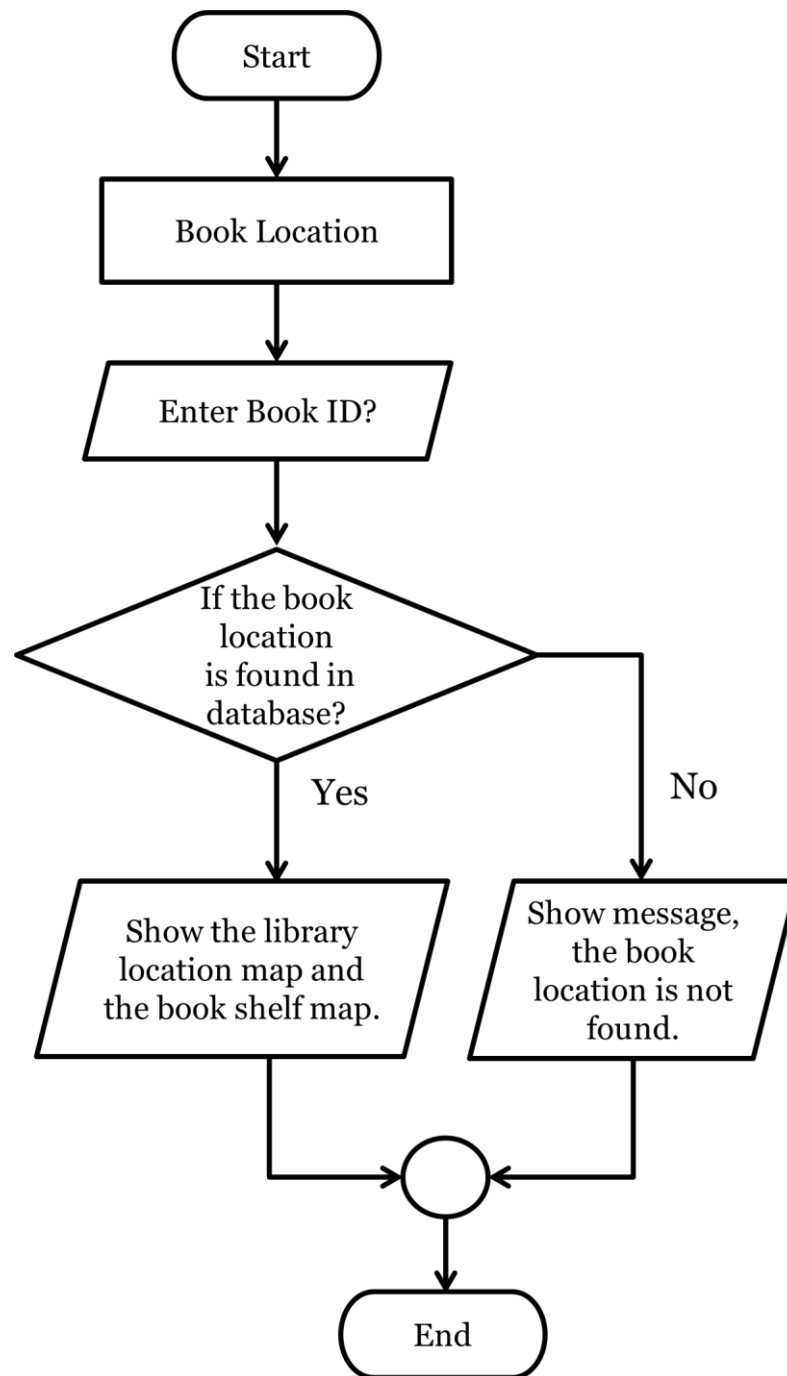
2. Book Location Finding flow chart.

Fig 4.2: Book Location Finding flow chart (4.2.2)
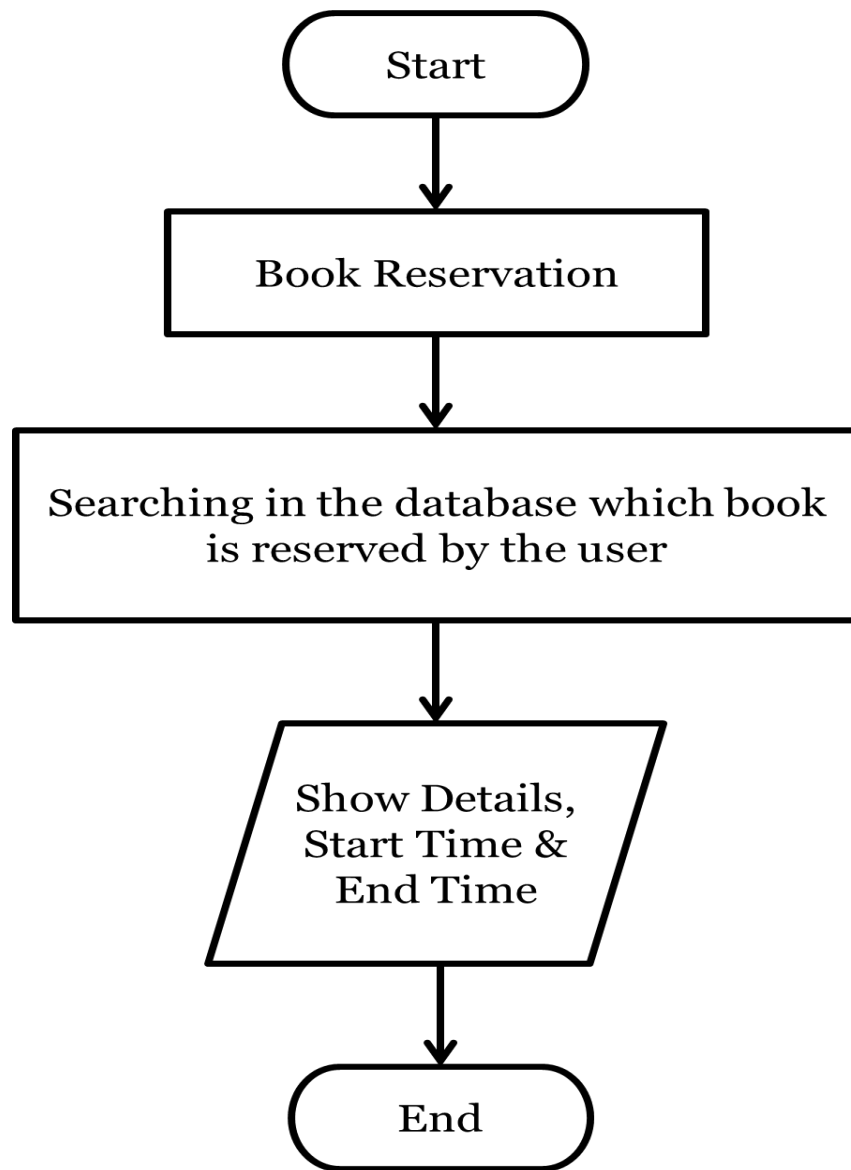
3. Book Reservation flow chart.

```
                    ╭──────────────╮
                    │    Start     │
                    ╰──────────────╯
                           │
                           ▼
          ┌───────────────────────────────┐
          │      Book Reservation         │
          └───────────────────────────────┘
                           │
                           ▼
   ┌─────────────────────────────────────────┐
   │  Searching in the database which book    │
   │        is reserved by the user           │
   └─────────────────────────────────────────┘
                           │
                           ▼
              ╱─────────────────────╲
             ╱    Show Details,       ╲
            ╱     Start Time &          ╲
           ╱       End Time              ╲
          ╱─────────────────────────────╱
                           │
                           ▼
                    ╭──────────────╮
                    │     End      │
                    ╰──────────────╯
```

Fig 4.3: Book Reservation flow chart (4.2.3)
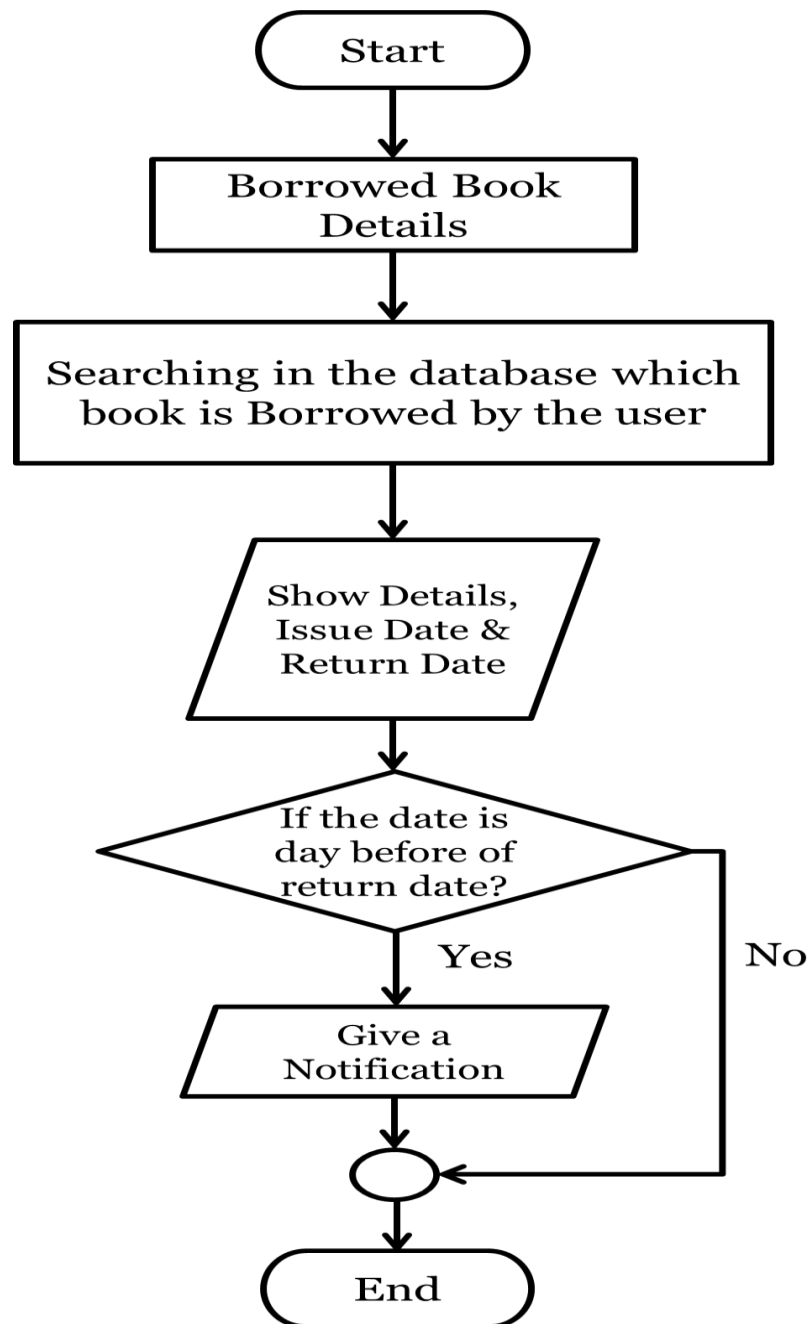
4. Borrowed Book Details flow chart.

Fig 4.4: Borrowed Book Details flow chart (4.2.4)

4.3 Database Design

Here are the Database Designs for our project app. They are given below:

User ID Primary Key int(20)	Password varchar(20)	Name varchar(20)	Mobile Number varchar(20)	Address varchar(50)	Penalty varchar(20)

Table 4.1: User Details (4.3.1)

Book Id (Primary Key) int(20)	Book Name varchar (20)	Book Writer varchar (20)	Book Edition varchar (20)	Book Quantity varchar (20)	Book Category varchar (20)	Book Location int(20)

Table 4.2: Book Details (4.3.2)

Reserve Id (Primary Key) int(20)	Book Id (Foreign Key) int(20)	User Id (Foreign Key) int(20)	Reserve Start Time (Time)	Reserve End Time (Time)

Table 4.3: Reserved Books (4.3.3)

Issue Id (Primary Key) int(20)	Book Id (Foreign Key) int(20)	User Id (Foreign Key) int(20)	Issue Date (Date)	Return Date (Date)	When Returned (Date)

Table 4.4: Issued Books (4.3.4)

Notice Id (Primary Key) int(20)	Notice varchar(50)

Table 4.5: Notices (4.3.5)

Statistics Id (Primary Key) int(20)	Book Id (Foreign Key) int(20)	Book Name (Foreign Key) varchar(20)	Book Writer (Foreign Key) varchar(20)	Book Edition (Foreign Key) varchar(20)	Number of Issue int(20)

Table 4.6: Yearly Generated Books Statistics/Report (4.3.6)

4.4 E-R Diagram

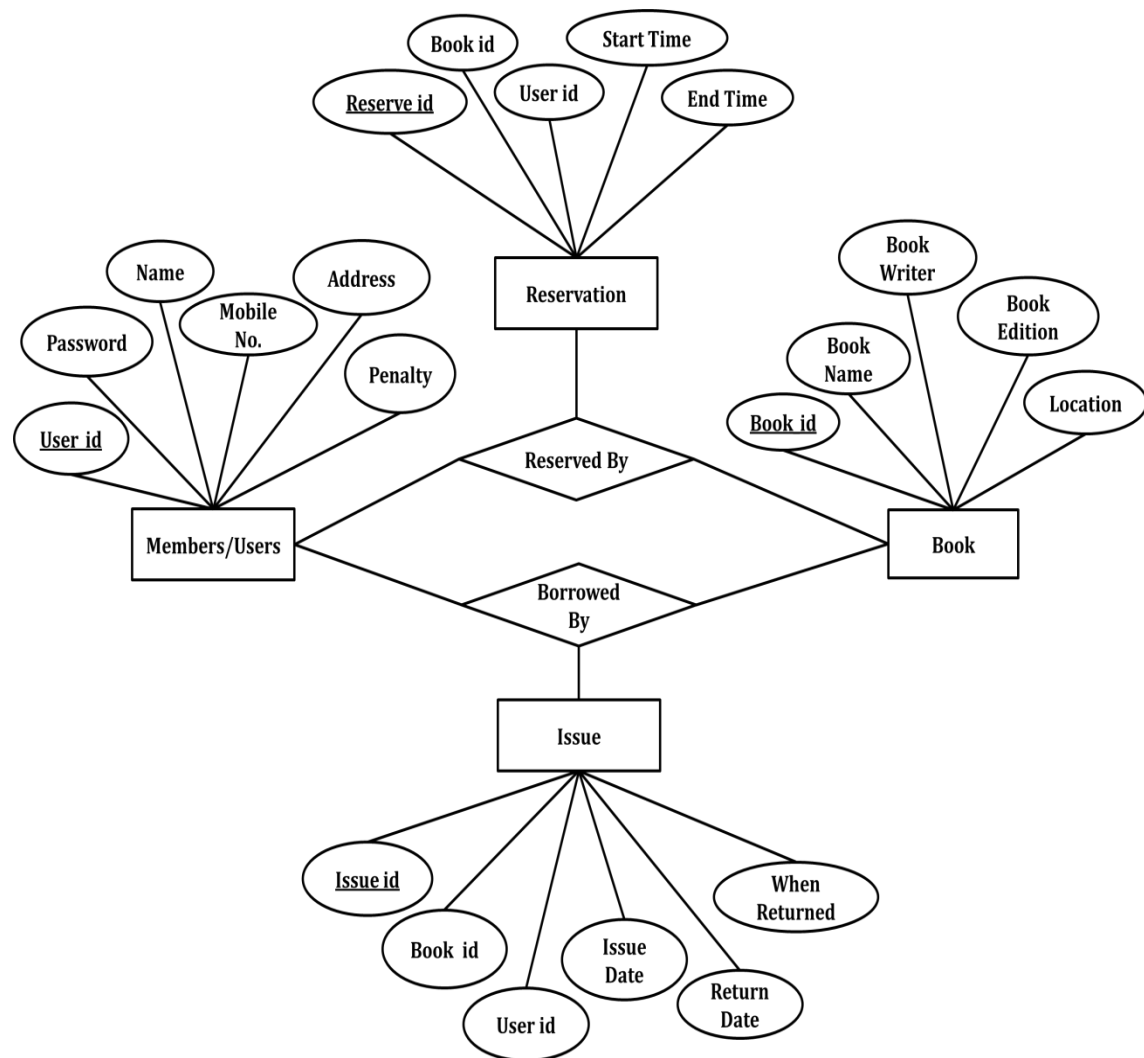Here is E-R Diagram for our project app. This is given below:

Fig 4.5: E-R Diagram (4.4.1)

4.5 Use Case Diagram

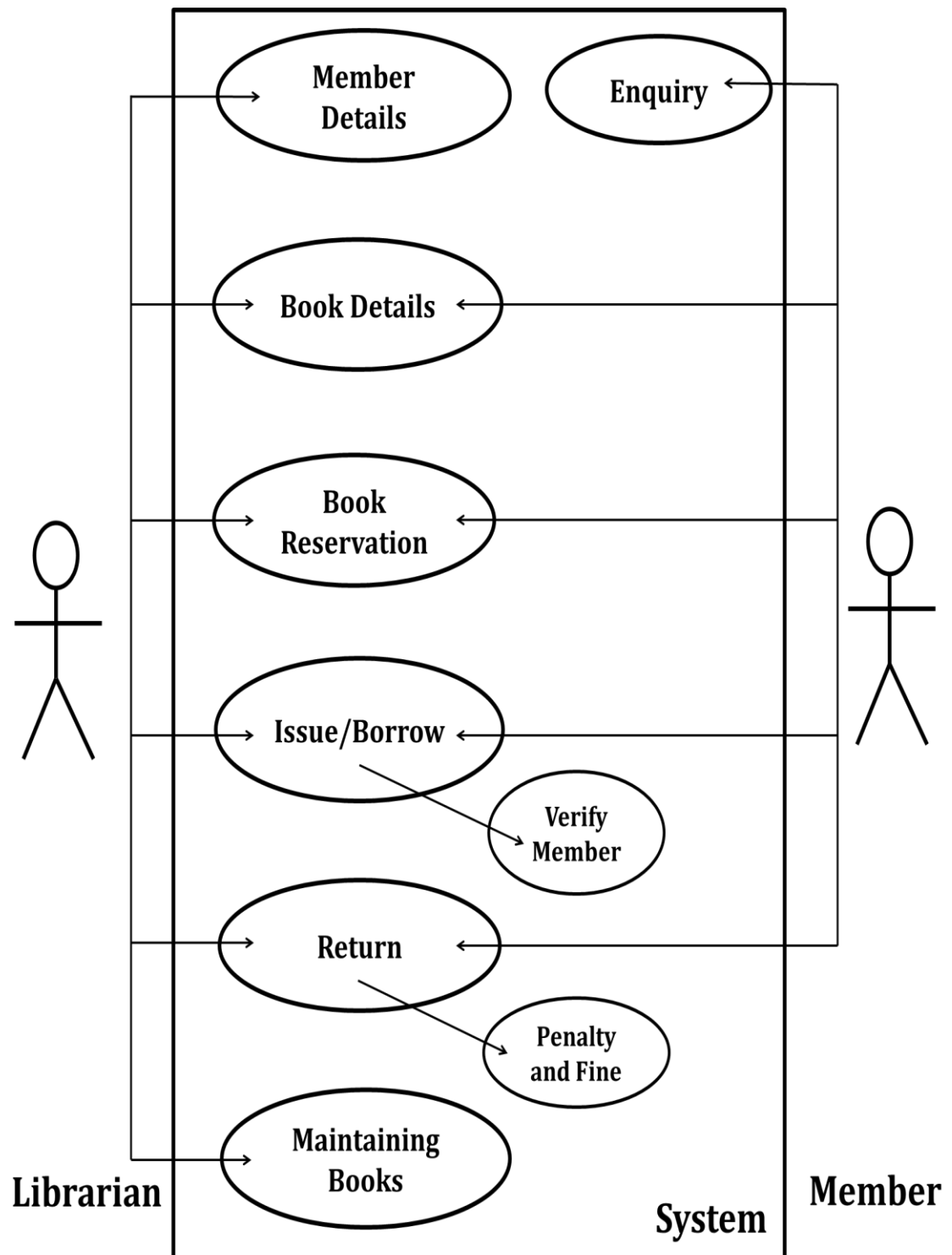Here is Use Case Diagram for our project app. This is given below:

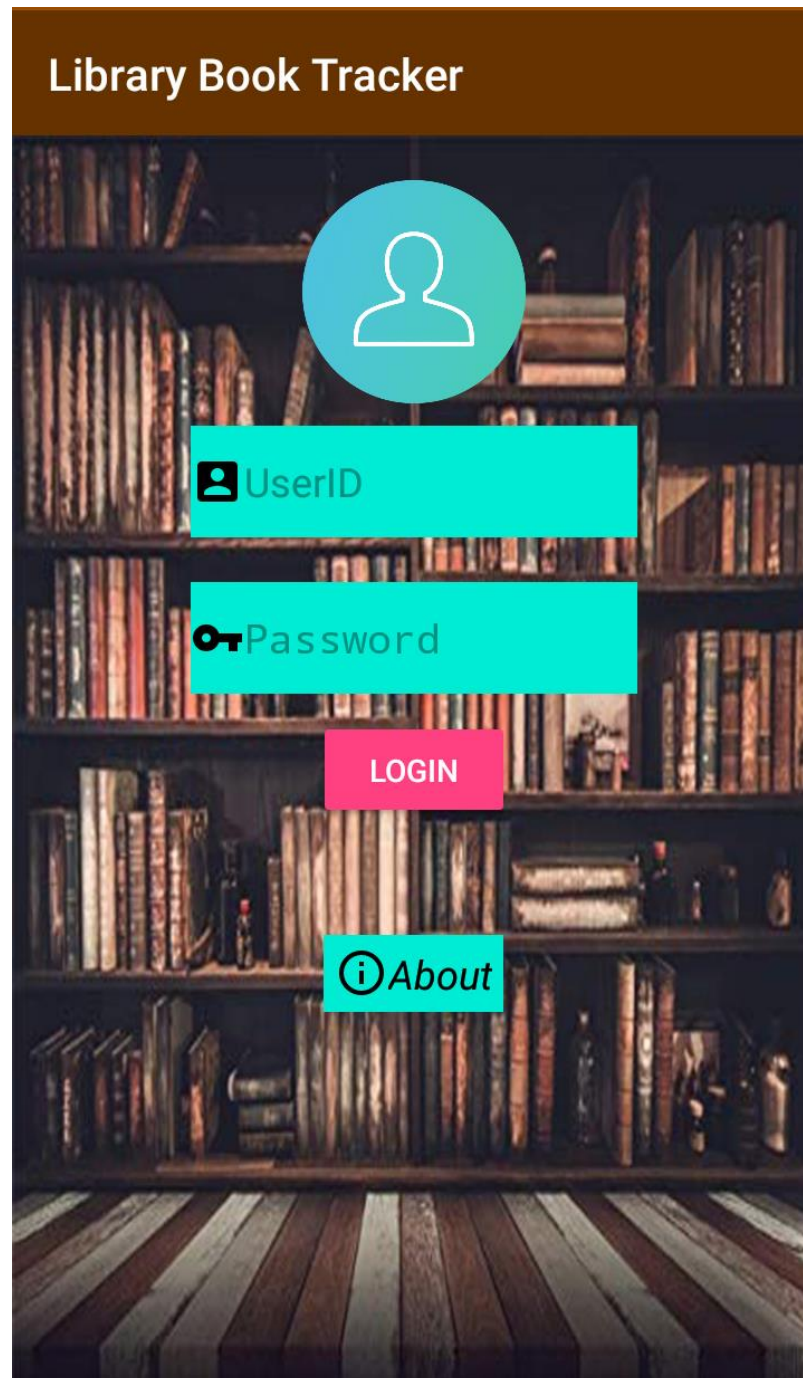**Fig 4.6: Use Case Diagram (4.5.1)**

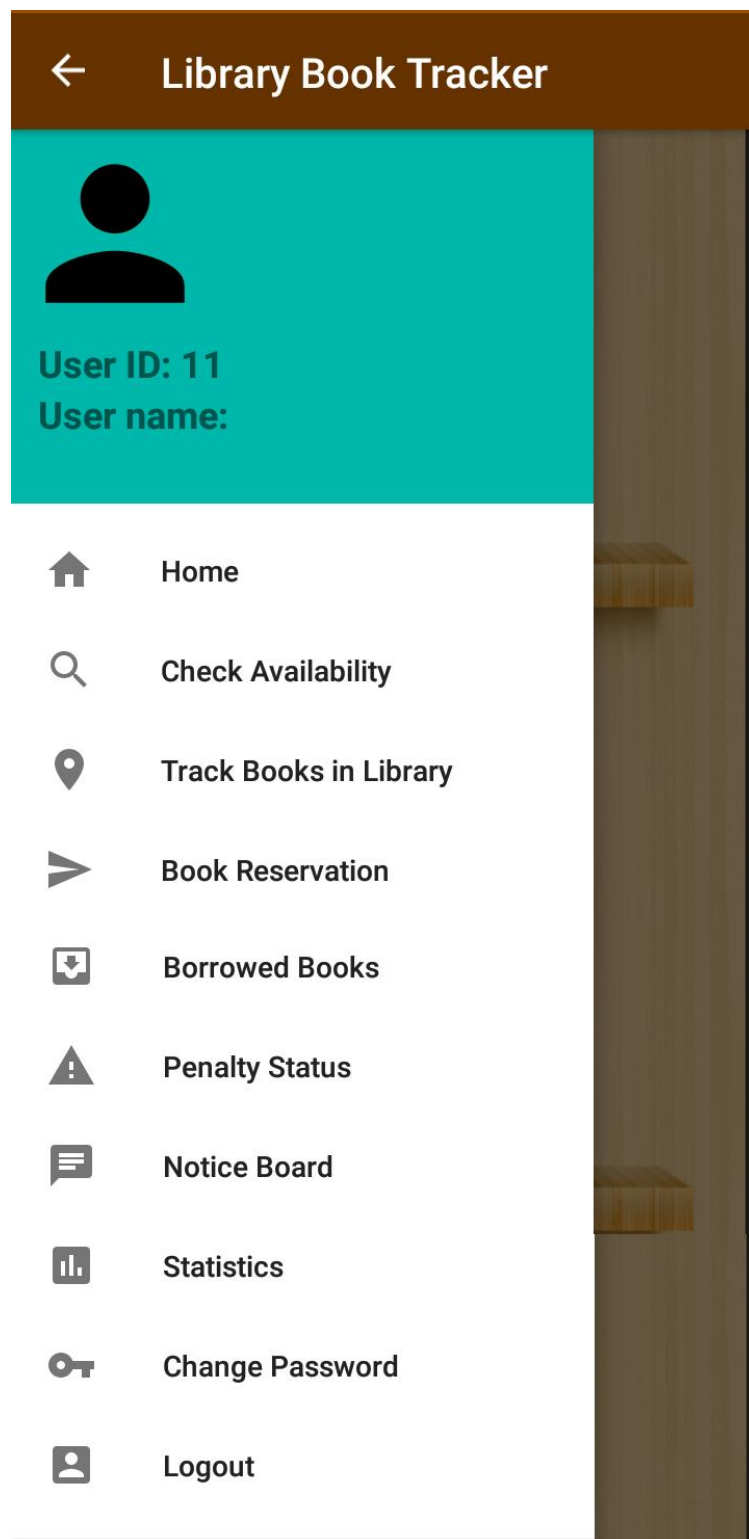## 4.6 App User Interface

Fig 4.7: Login Interface (4.6.1)

Fig 4.8: App Options Interface (4.6.2)
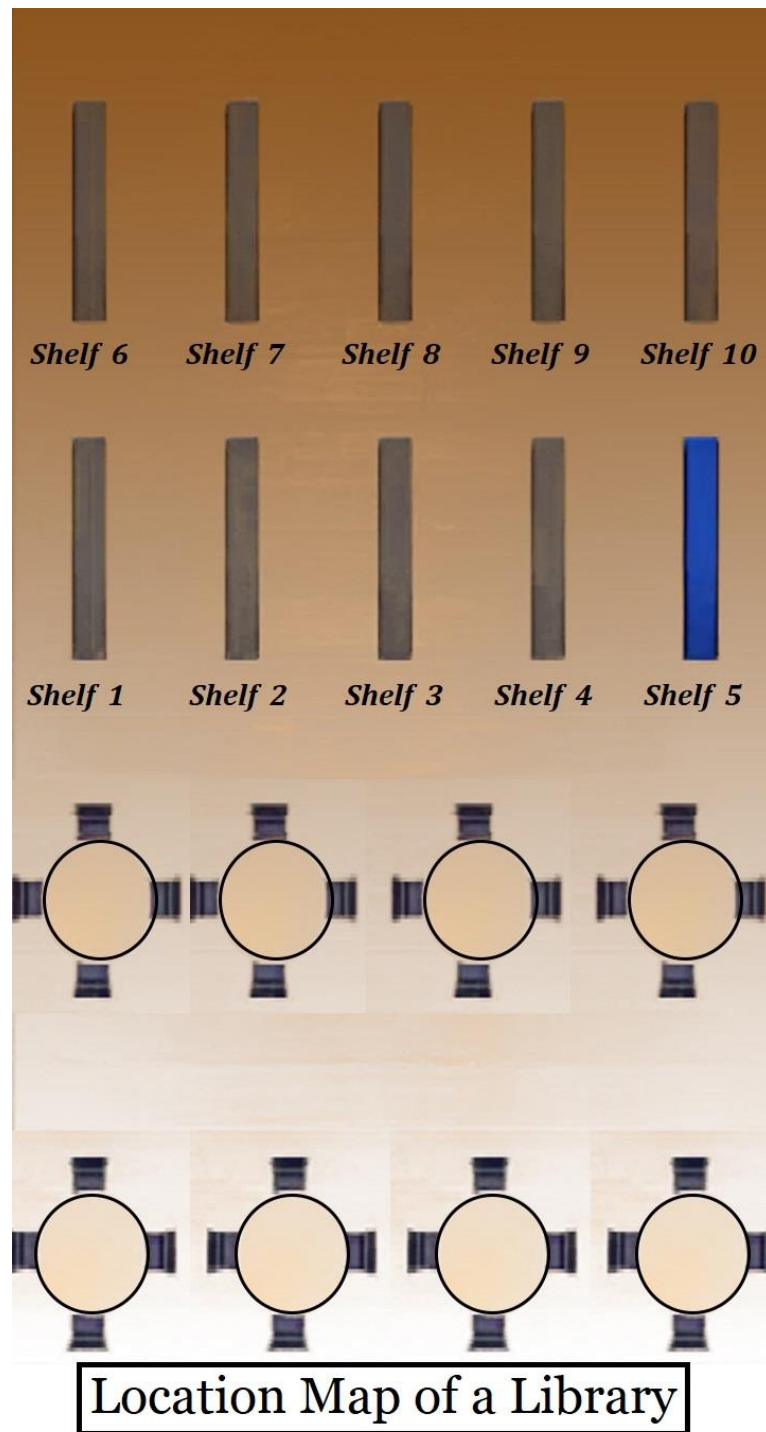
Fig 4.9: Book Preview Interface (4.6.3)

Shelf 6   **Shelf 7**   **Shelf 8**   **Shelf 9**   **Shelf 10**

Shelf 1   **Shelf 2**   **Shelf 3**   **Shelf 4**   **Shelf 5**

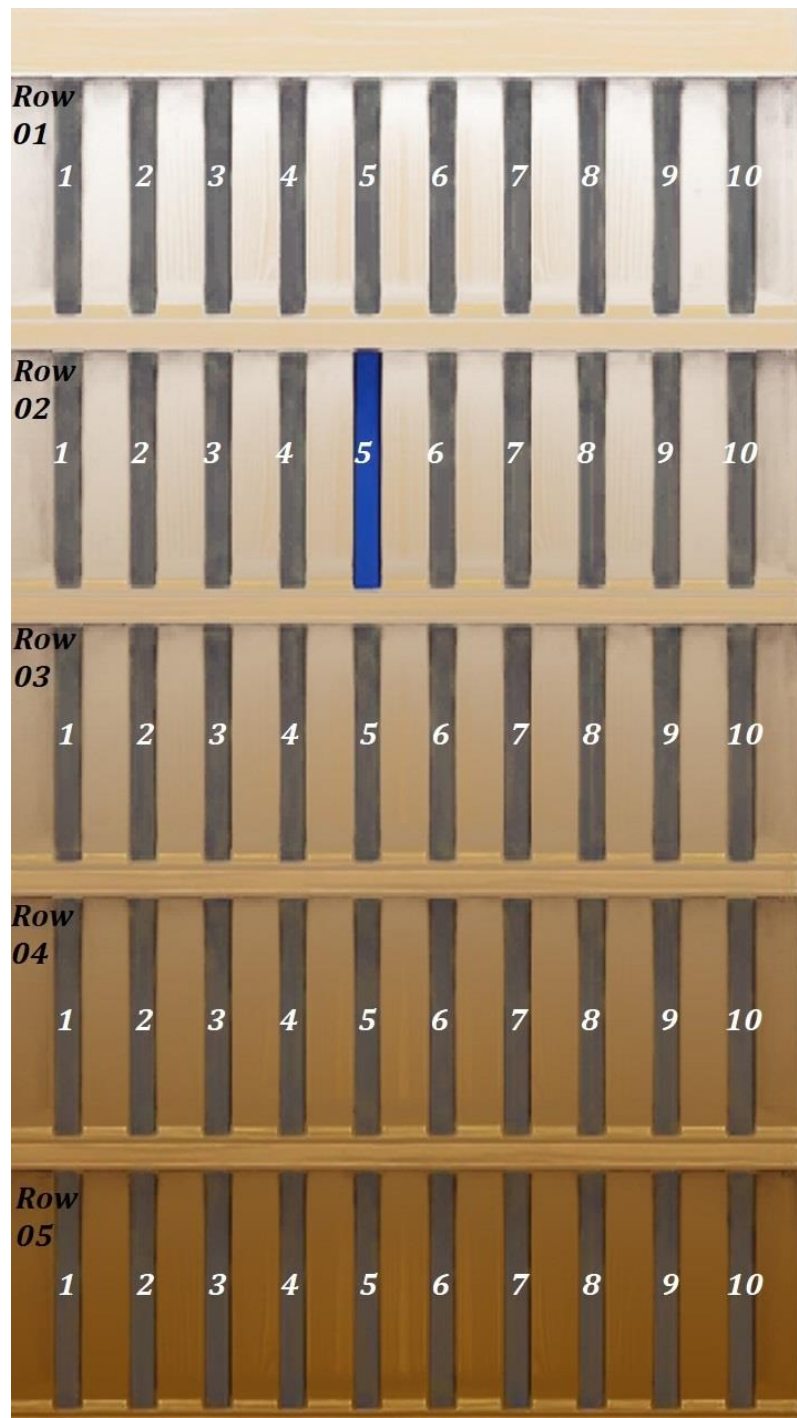Location Map of a Library

Fig 4.10: Map of a Library Room (4.6.4)

Location Map of a Shelf

Fig 4.11: Location of a Book in a Shelf (4.6.5)

4.7 Conclusion

So, we have known about the Flow Chart, Database Design, E-R Diagram, Use Case Diagram and App User Interface of our project in this chapter.

CHAPTER-5

CONCLUSION

CHAPTER 5
CONCLUSION

_____

5.1 Conclusion

So, at conclusion we can say that the project of "Library Book Tracker Android App" will help the library members/users to get most of the library facilities. We have just finished our project works till the design phase. Our next steps will be giving focus on the Implementation, Testing, Maintaining and Releasing the Final Stable Version of the app. Soon, we will start our work on that.

Abbreviations

AOSP - Android Open Source Project

API - Application Programming Interface

ADT - Android Development Tools

AVD - Android Virtual Device

DB - Database

GNU - GNU's Not UNIX

GMS - Google Mobile Services

IDE - Integrated Development Environment

JDK - JAVA Development Kit

JVM - Java virtual machine

LTS - Long Term Support

OS - Operating System

PHP - Personal Home Page

SDLC - Software Development Life Cycle

SQL - Structured Query Language

SDK - Software Development Kit

UI - User Interfaces

Wi-Fi - Wireless Fidelity

XAMPP - Cross-Platform (X), Apache (A), Maria DB (M), PHP (P) and Perl (P)

XML - eXtensible Markup Language

References

[1]. Ashutosh Tripathi& Ashish Srivastava, Online Library Management System, IOSR Journal of Engineering (IOSRJEN), Vol. 2 Issue 2, Feb.2012, pp. 180-186.

[2]. http://en.wikipedia.org/wiki/Integrated_library_system.

[3]. C. Srujana, B. Rama Murthy, K .Tanveer Alam, U. Sunitha, Mahammad D.V, P.Thimmaiah,Development of RFID Based Library Management System Using MATLAB, International Journal of Engineering and Advanced Technology (IJEAT)ISSN:2249 – 8958, Volume-2, Issue-5, June 2013.

[4]. http://en.wikipedia.org/wiki/Mobile_technology.

[5]. Hui li, Zhao-quan cai2, Design and Implementation Of the Mobile library app based on Smart Phone, Proceedings of the 2016 International Conference on Machine Learning and Cybernetics, Jeju, South Korea, 10-13 July, 2016.

[6]. http://en.wikipedia.org/wiki/Android_application_package.

[7]. R.Dinesh, S.R.Arun Pravin, M .Aravindhan, D .Rajeswari, Library Access System Smartphone Application using Android, R. Dinesh *et al*, International Journal of Computer Science and Mobile Computing, vol.4 issue.3, march- 2015, pg. 142-149.

[8]. http://www.webopedia.com/TERM/A/android_app.html

[9]. http://developer.android.com/sdk/index.html.

[10]. http://developer.android.com/tools/help/adt.html.

[11]. https://www.techopedia.com/definition/25099/android-app.

[12] Archana, S. N., Padmakumar, P. K. and Beena, C. (2014). Catalogue interfaces of Integrated Library Management System (ILMS): Experiences in a proprietary and open source software. DESIDOC Journal of Library and Information Technology, 34(1), 16-22.

[13] Deddens, M. (2002). Overview of integrated library system. EDUCAUSE Evolving Technologies Committee, 1-6.

[14] Gunawardena, D. N. T. (2016). Impact of the Library Management System on Academic Activities of Undergraduate Students of the University of Moratuwa, Sri Lanka.

[15] Horsfall, K. (1992). The human impact of library automation. Retrieved from htto://websimmons,edu/--cheninit/NIT'92/195-hor.htm

[16] Khan, M. T., Zahid, A., and Rafiq, M. (2016). Journey from Library Management System (LMS) to KOHA by Government College University Libraries,

Kumar, A. and Mandal, S. (2013). Development of cloud computing in integrated library management and retrieval system. International Journal of Library and Information Science, 5(10), 394-400.

[17] Muller, T. (2011). How to choose an free and open source integrated library system. OCLC Systems and Services : International Digital Library Perspective, 27(1), 57-78.

[18] Ping, F. and Fitzgerald, M. (2013). A comparative analysis of the effect of the integrated library system on staffing models in academic libraries. Information Technology and Libraries, 47-58.

[19] Reddy, T. R. and Kumar, K. (2013). Open source software's and their impact on library and information centre : An overview. International Journal of Library and Information Science , 5(4), 90-96.

[20] Ryan, C. E...[et al.] (2012). NISO Wcbinar : The future of integrated library systems : Part 1 : RDA and Cataloging. Serials Review, 38, 48-65.

[21] Shailendra, K. and Namrata, R. (2011). Web interface in library management software system. DESIDOC Journal of Library and Information Technology , 31(4), 262-274.

[22] Vaughan, J. and Costello, K. (2011). Management and support of shared integrated library system. Information Technology and Libraries, 62-70.

[23] Yang, S. (2013). From integrated library system to library management service : Time for change?. Library Hi Tech News, 30(2), 1-8.

[24] Yongming, W. and Dawes, T. A. (2012). The next generation integrated library system: A promise fulfilled. Information Technology and Libraries, 76-84.