

---

# **Software Requirements Specification**

**for**

# **Library Management System**

**Version 2.0**

**Prepared by Swati S Bhat , 15IT249**

**National Institute of Technology Karnataka , Surathkal**

**9<sup>th</sup> January 2018**

# Table of Contents

<b>Table of Contents.....</b>	<b>ii</b>
<b>Revision History.....</b>	<b>ii</b>
<b>1. Introduction.....</b>	<b>1</b>
1.1 Purpose.....	1
1.2 Document Conventions.....	1
1.3 Intended Audience and Reading Suggestions.....	1
1.4 Product Scope.....	2
1.5 References.....	2
<b>2. Overall Description.....</b>	<b>3</b>
2.1 Product Perspective.....	3
2.2 Product Functions.....	3
2.3 User Classes and Characteristics.....	4
2.4 Operating Environment.....	4
2.5 Design and Implementation Constraints.....	5
2.6 User Documentation.....	5
2.7 Assumptions and Dependencies.....	5
<b>3. External Interface Requirements.....</b>	<b>6</b>
3.1 User Interfaces.....	6
3.2 Hardware Interfaces.....	6
3.3 Software Interfaces.....	6
3.4 Communications Interfaces.....	6
<b>4. System Features.....</b>	<b>7</b>
4.1 Use Cases.....	7
4.2 Functional Requirements.....	7
<b>5. Other Nonfunctional Requirements.....</b>	<b>9</b>
5.1 Performance Requirements.....	9
5.2 Safety Requirements.....	9
5.3 Security Requirements.....	9
5.4 Software Quality Attributes.....	9
5.5 Business Rules.....	9
<b>6. Other Requirements.....</b>	<b>10</b>
<b>Appendix A: Glossary.....</b>	<b>11</b>
<b>Appendix B: Analysis Models.....</b>	<b>11</b>
<b>Appendix C: To Be Determined List.....</b>	<b>18</b>

## Revision History

Name	Date	Reason For Changes	Version

# 1. Introduction

## 1.1 Purpose

The purpose of this Software Requirements Specification document is to provide a description of the requirements, functional and non-functional, of the LMS (Library Management System) software. This document provides a description of the functionalities of the product, the software and hardware requirements and the operating environment. The product is in its first version and there's no existing release.

LMS is an online system intended to provide members with a portal to search for and borrow books available in the library where it is installed. The main functionalities include issuing books, maintaining records of registered users and the books issued to them and updating the LMS database among others.

## 1.2 Document Conventions

The conventions followed in this document include:

1. LMS – Library Management System
2. SRS – Software Requirements Specification
3. UML – Unified Modelling Language
4. GUI – Graphical User Interface

## 1.3 Intended Audience and Reading Suggestions

This SRS document is intended for developers, project managers, testers and stakeholders. The document contains technical terms some of which require domain knowledge and thus may not be comprehensible to the end user.

**Section 1 (Introduction)** of this document gives a brief overview of the rest of the SRS document. It also talks about the scope of the software, the target audience of the software and the references made while preparing this document.

**Section 2 (Overall Description)** of this document provides an overview and description of the entire software. It also contains the UML diagrams (context diagram, dataflow diagram and the entity relationship diagram). This section is intended for all audience including users and developers.

**Section 3 (External Interface Requirements)** provides information about how the software would interact with other processes, users and hardware.

**Section 4 (System Features)** describes the functional requirements of the software. It also contains the use case and domain model diagrams. This section is mainly meant for the developers and stakeholders.

**Section 5 (Non functional requirements)** lists the non functional requirements of the software like performance, safety and security requirements. It is meant for the stakeholders primarily. Any further requirements can be found in **Section 6**.

## **1.4 Product Scope**

LMS is an online system that mitigates the issues associated with the conventional and manual system of reservation and issuing of books. It's an online software that makes looking up books available easier for the registered users of the software. It automates the process of maintaining information about the number of copies of books available in the library. It provides a user friendly environment that makes borrowing and returning books easier and more efficient. It also provides a simple and easy to understand GUI that makes maintaining records of registered users and books issued, easier for the librarian/administrator. LMS is a software intended for readers (students, faculty and other registered users) and the librarian/administrator.

Its a secure and robust online application that addresses the issues of a manual library management system .

## **1.5 References**

[1] IEEE Software Engineering Standards Committee, "IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications", October 20, 1998

[2] Karlsson J, "A Cost-Value Approach for Prioritizing Requirements", Norges Teknisk-Naturvitenskapelige Uni.1997

[3] Davis M A, "Just Enough Requirements Management: Where Software Development Meets Marketing", New York, Dorset House Publishing, 2005.

[4] "Design and Implementation of a Library Management System Based on the Web Service" by Yujun Li, Hao Zheng, Tengfei Yang, Zhiqiang Liu. <http://ieeexplore.ieee.org/document/6405716/>

## 2. Overall Description

### 2.1 Product Perspective

This software is a web application that provides a number of functionalities to a registered user. Since this is a data-centric product, a database is maintained. The administrator/librarian possesses the permission to add/delete books to/from the library database once signed in as the administrator. A new user is required to create their own account if they wish to borrow books from the library, after which they'll be assigned a unique ID. However, a user, registered or not, can query/search for books in the database using the search box provided. The query will display information about the presence or absence of the book in the LMS and if present, displays information about the availability of the book.

The administrator once signed in, holds the right to update and maintain the database. The administrator can add/delete books to/from the database as and when available and also update information about availability of books. The administrator also plays the role of a librarian and can issue books to registered users and change the status of an issued book on return.

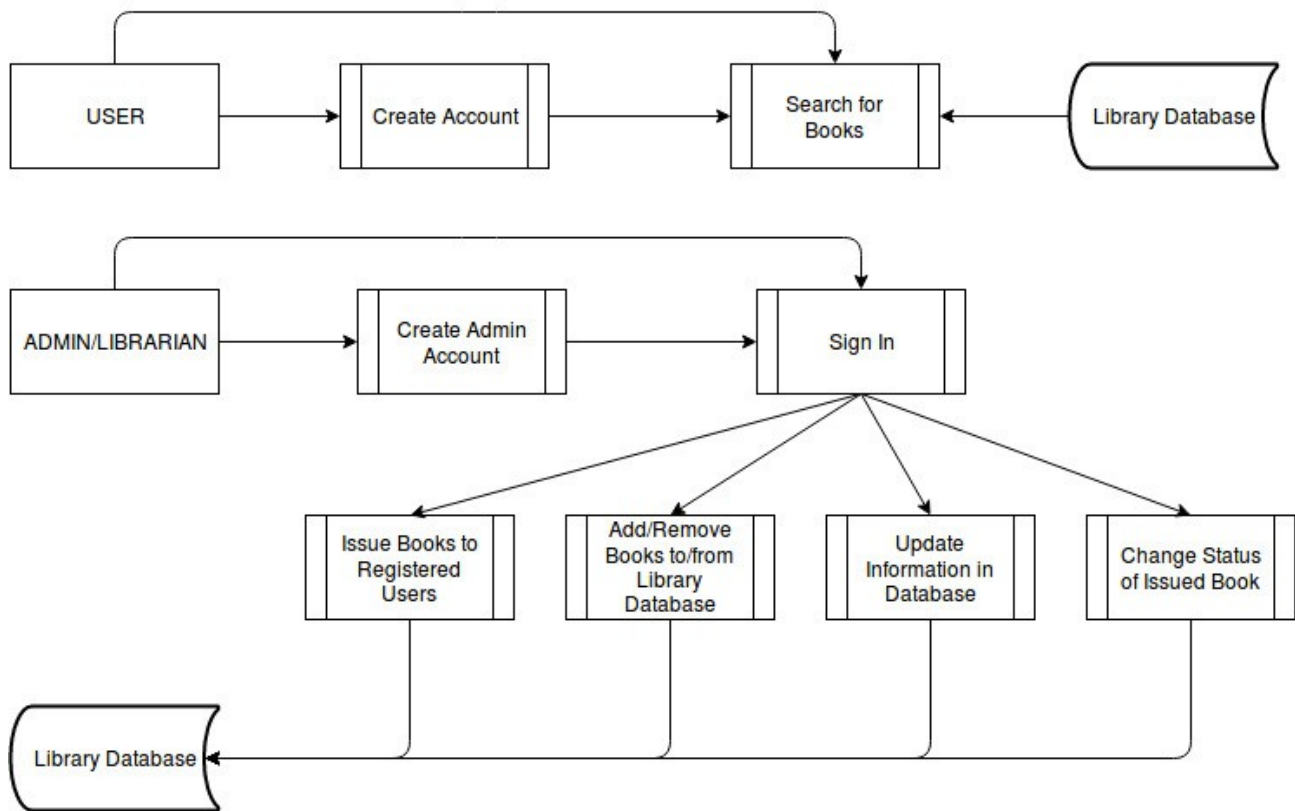
The administrator has access to the library database. Any other user does not.

### 2.2 Product Functions

The LMS software should provide the following functionalities:

1. Functionalities available to users:
  - Registration/sign up functionality for users to set up an account
  - Search functionality for users(registered or non-registered) to look up books available in the library
2. Functionalities available to the librarian/administrator:
  - Functionality that facilitates issue and return of books to registered user
  - Functionality to add/remove books to/from the library database
  - Functionality to update information in the database
  - Functionality to update number of copies of books available in the library

Here's a data flow diagram to understand how the functions are intended to be provided.



## 2.3 User Classes and Characteristics

The software should provide support for two types of user privileges – user and administrator. Users should have access to user functions, and the administrator should have access to both user and administrator functions. The administrator of the system is expected to have more knowledge about the internal working of the system and possess the ability to rectify problems that may arise due to disk crashes, power failures or other system failures. The GUI must be user friendly and sufficient to educate the users on how to use the system without any issues and hence do not require any technical expertise/domain knowledge.

## 2.4 Operating Environment

The software is developed to run on any \*nix platform or Windows 8+ platform with XAMPP installed. All devices that support these versions of the operating system with XAMPP up and running will be able to run the software. The software is developed using Django, a python framework to develop web applications with ease. Since the product is data-centric, a database that satisfies the required storage needs is used. LMS software thus uses MySQL database to store data.

## **2.5 Design and Implementation Constraints**

The software will be developed using the Django framework for Python. MySQL database is used for storage. However the amount of information that can be stored for the LMS will be constrained by the capacity of the database. The TCP/IP protocol is followed for internet communication. For working on the coding phase of the LMS - Python is used as the primary programming language and supplemented by HTML,CSS,jQuery among others for the web development part of the software development process.

## **2.6 User Documentation**

The software does not provide any user documentation and assumes that the user is qualified enough to navigate through the application without any external help.

## **2.7 Assumptions and Dependencies**

It is assumed that the user machine has enough RAM to run the web application and a working internet connection.

## **3. External Interface Requirements**

### **3.1 User Interfaces**

The GUI is user friendly and the users interact with the system through the application portal. The web portal contains a home page with search functionality with GUI buttons to navigate to the registration page or the administrator login page. The user should be able to search for and view information about books in the home page. Upon successful login, the administrator should be redirected to a page which has GUI buttons to add/remove books from the database. The administrator should also be able to issue books and change status of issued books on return . The information about books present in the database is listed in the form of a table. The administrator home page on logging in should have a GUI button to facilitate the logout function.

### **3.2 Hardware Interfaces**

Since the web portal does not have any designated hardware, it does not have any direct hardware interfaces. The hardware connection to the database server is managed by the underlying operating system on the web server.

### **3.3 Software Interfaces**

The software interfaces include:

- MySQL version 5.7.19 or greater
- \*nix based operating system
- XAMPP server

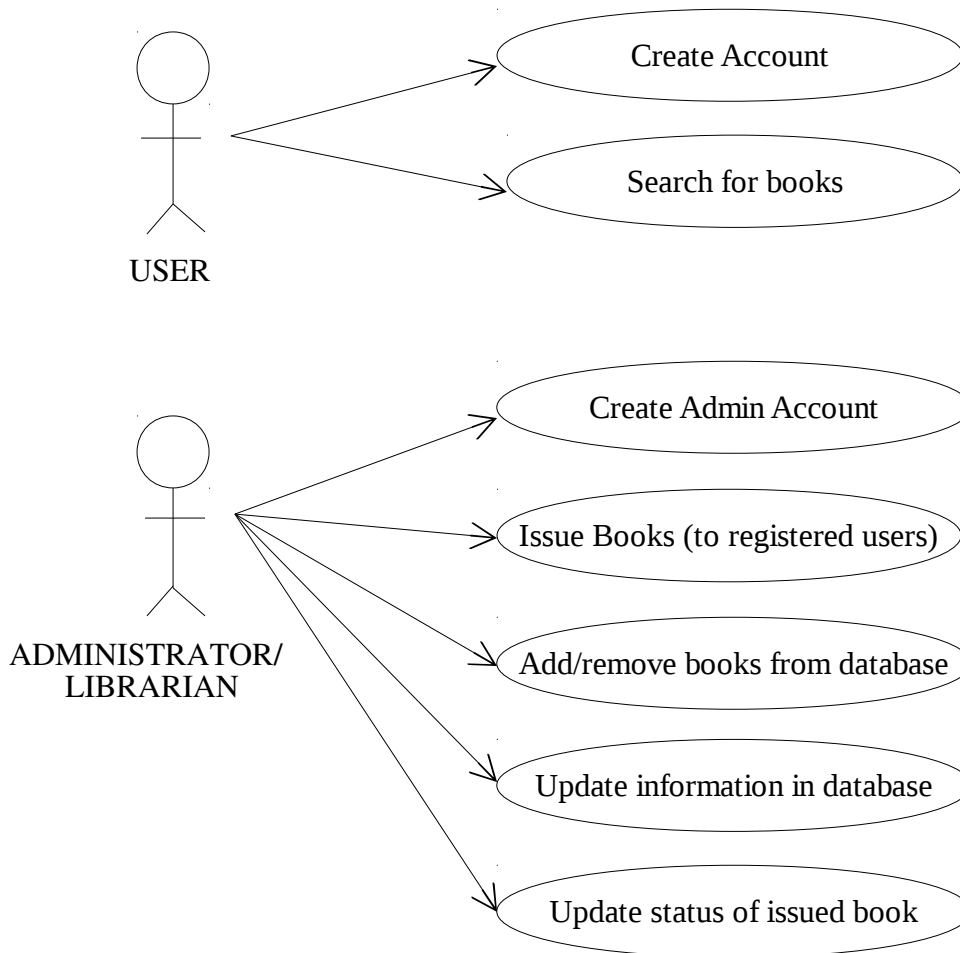
### **3.4 Communications Interfaces**

The communication of the software will be through HTTP.



## 4. System Features

### 4.1 Use Cases



### 4.2 Functional Requirements

In this subsection, the functional requirements of the software are specified. The functional requirements of this software are:

#### 4.2.1 Functional requirements associated with functionalities available to users:

1. Allow creation of user account  
The user should be able to register to set up an account of their own. The necessary details should be filled in by the user to create a profile and set up an account.
2. Allow user to search for books available in the library  
The user irrespective of whether they've registered or not, should be able to search for books in the library database by giving a search keyword in the search box provided.

**4.2.2 Functional requirements associated with functionalities available to the librarian/administrator:**

1. Allow creation of librarian/administrator account  
The librarian/administrator should set up an account to be able to make use of the functionalities available to the administrator of the software.
2. Allow librarian/administrator to issue books to registered users  
The librarian/administrator should be able to issue a book to a registered user by entering the UID of the user and BID of the book being issued. The librarian/administrator should also be able to specify the return date of the book being issued. The librarian/administrator should be signed in to be able to carry out any of these operations.
3. Allow librarian/administrator to add/remove books from the library database  
The librarian/administrator on being signed in , should be able to add books or remove books from the database with arrival/availability of new books in the library.
4. Allow librarian/administrator to update certain necessary information in the database  
The librarian/administrator on being signed in , should be able to update the number of copies of a book available in the library, in the database.
5. Allow librarian/administrator to change status of an issued book once it's returned .

## 5. Other Nonfunctional Requirements

### 5.1 Performance Requirements

The software functionalities should be easy to understand and use for the end users since it's a system which is intended for an audience that may have varying level of domain knowledge/technical knowledge. The speed of the system should be good enough to retrieve information requested by the user and also to make the process of borrowing and returning books easier, less time consuming and more efficient.

### 5.2 Safety Requirements

The database backup should be performed on a regular basis so that recovery becomes easier in case of system failure. There are no other safety requirements with this application.

### 5.3 Security Requirements

Database access rights depend on the category of the end user of the software. Database should be accessible only to the administrator. The software should not give database access to the other users (registered or non-registered). The application should be robust, the database should be secure and the contents should not be lost or manipulated by unauthorized user.

### 5.4 Software Quality Attributes

#### **Portability**

The software must be platform independent that is it should be portable. The database contents should not be lost in the process.

#### **Correctness**

The software should be correct in terms of its functionality i.e. the application should adhere to functional requirements.

#### **Completeness**

The software should cover all necessary requirements that it is expected out of it. In addition, it will also adhere to other non-functional requirements. The target audience of the software have varying level of domain/technical knowledge and thus the software should be user friendly and simple to use.

### 5.5 Business Rules

The librarian plays the role of the administrator and should be given database access and modification rights. Other users have no access to the database and can make use of other functionalities available to registered and non-registered users.

## **6. Other Requirements**

There are no other requirements associated with this software.

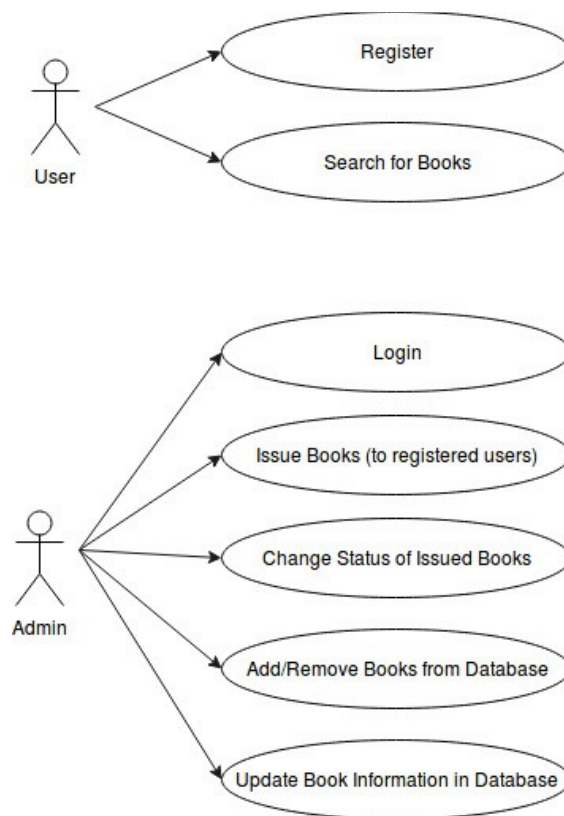
## Appendix A: Glossary

**HTTP:** HTTP stands for Hyper Text Transfer Protocol which is an application protocol for distributed, collaborative, and hypermedia information systems.

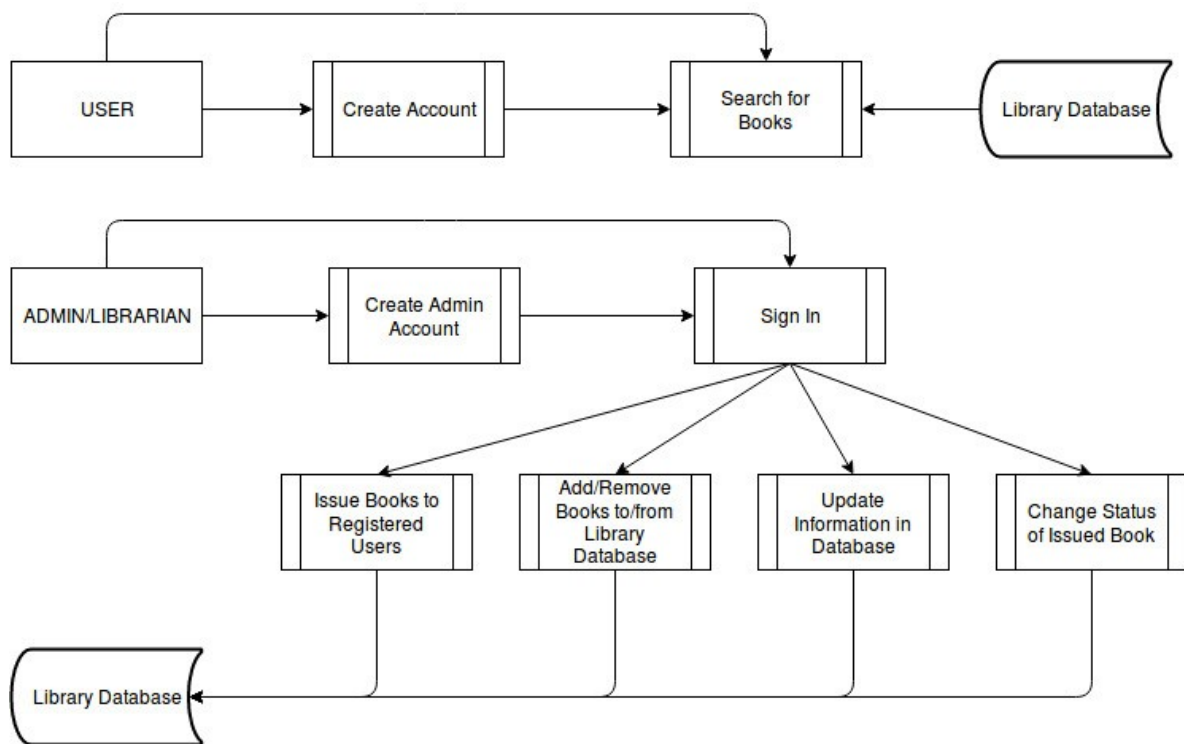
**TCP/IP:** TCP/IP stands for Transmission Control Protocol/Internet Protocol , which is a set of networking protocols that allow two or more computers to communicate.

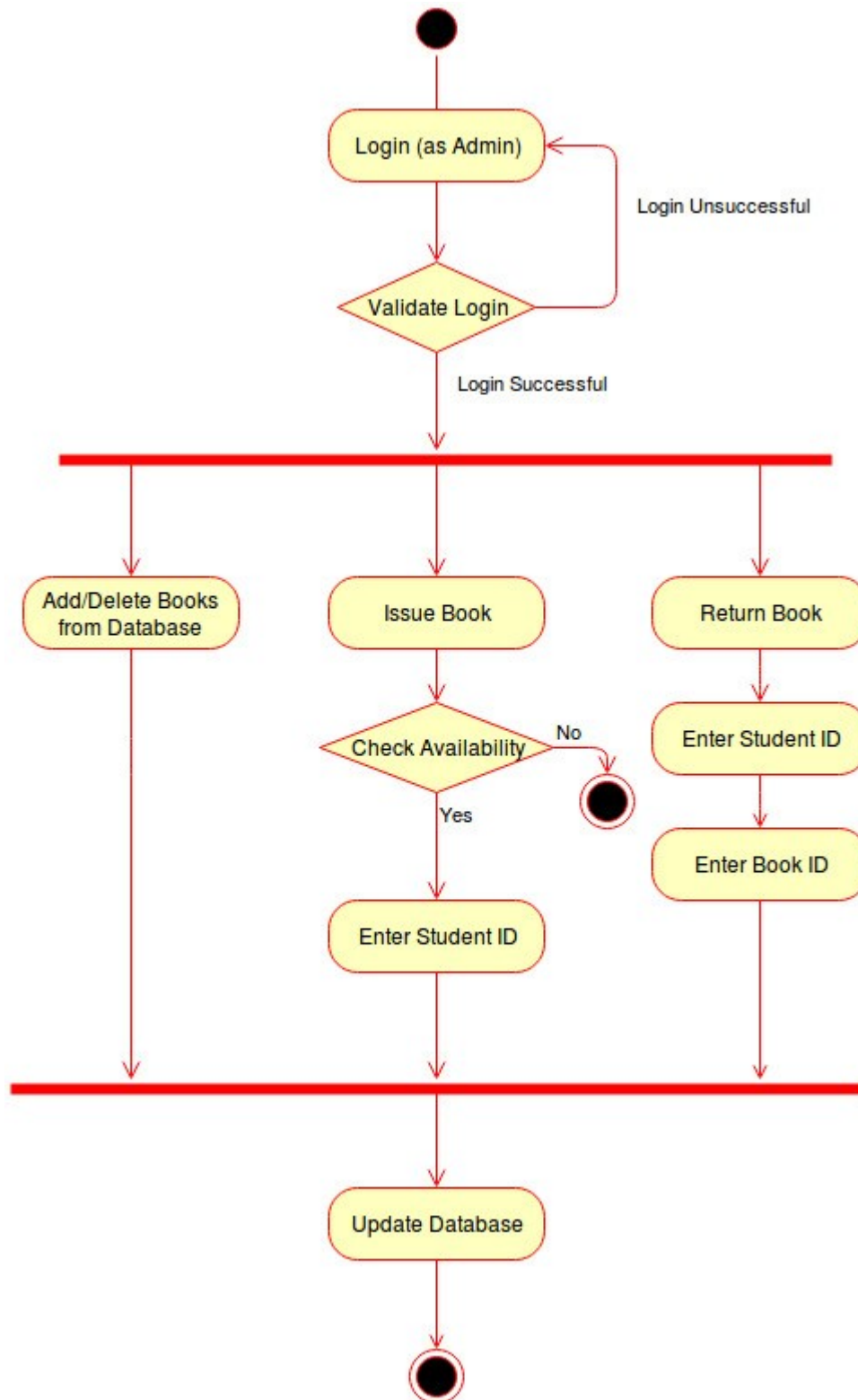
**RAM:** RAM stands for Random Access Memory and is a form of computer data storage that stores data and machine code currently being used.

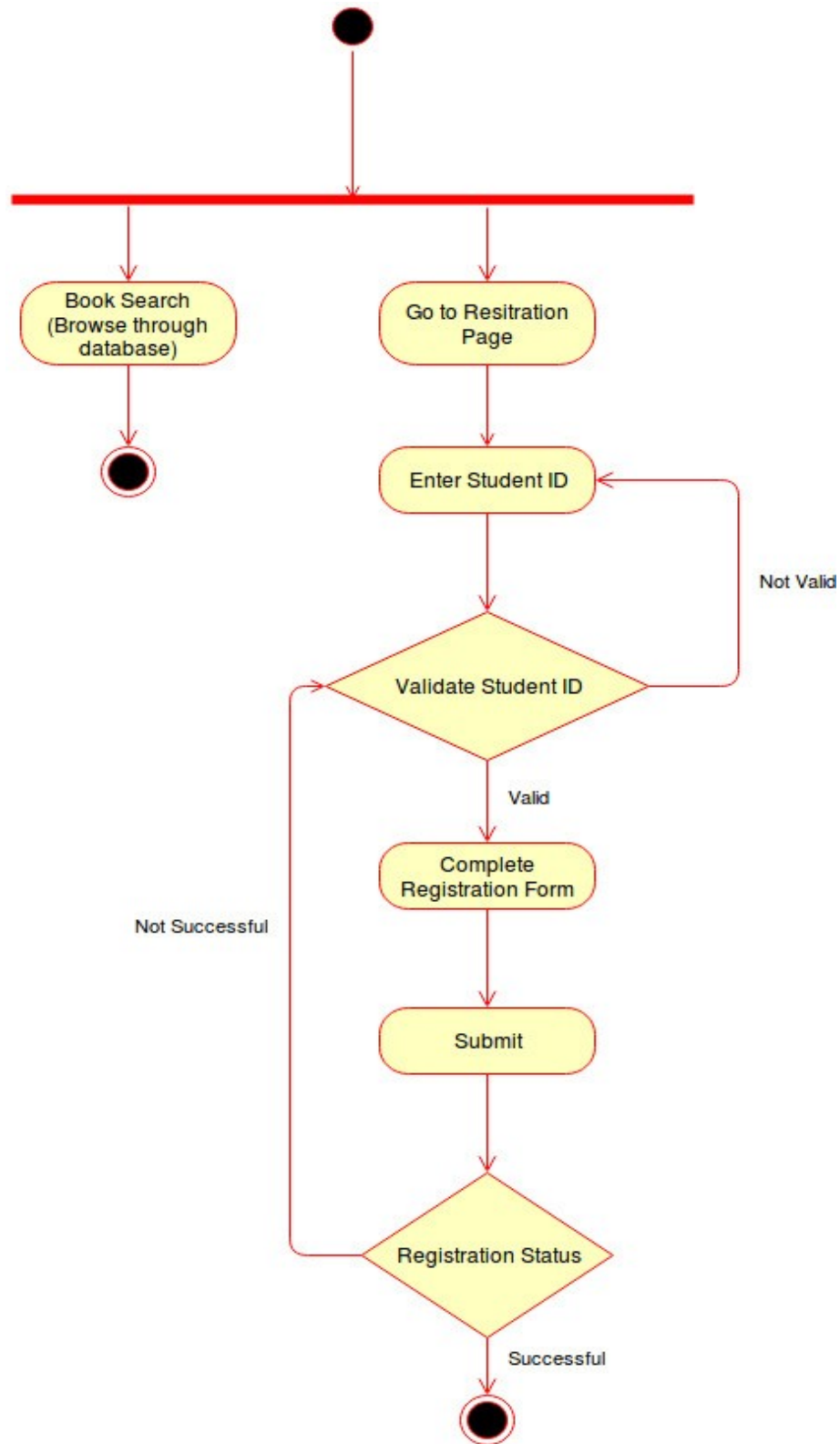
## Appendix B: Analysis Models



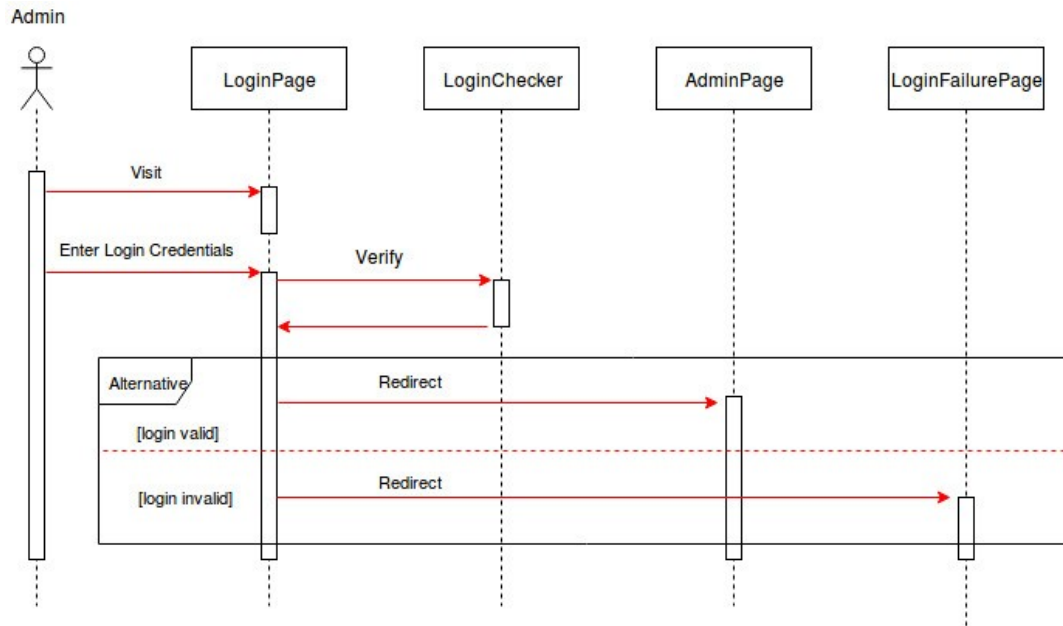
**Fig 1: Use Case Diagram**

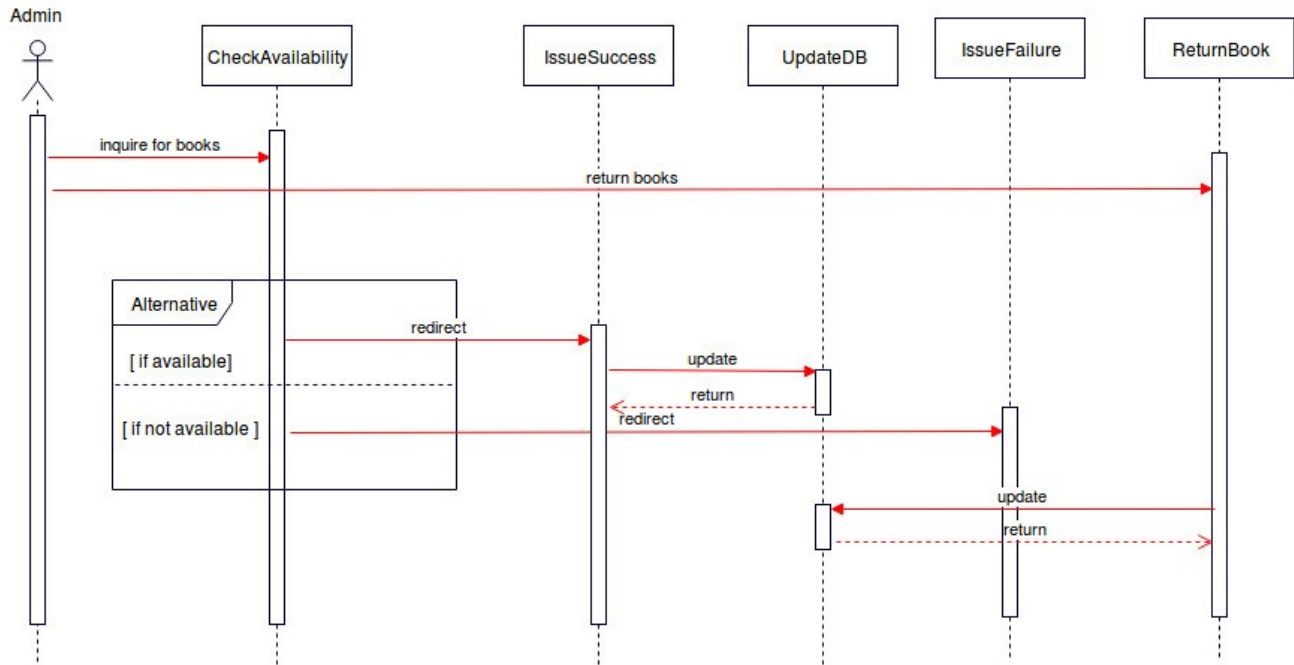
**Fig 2: Flow Diagram**

**Fig 3: Admin Activity Diagram**

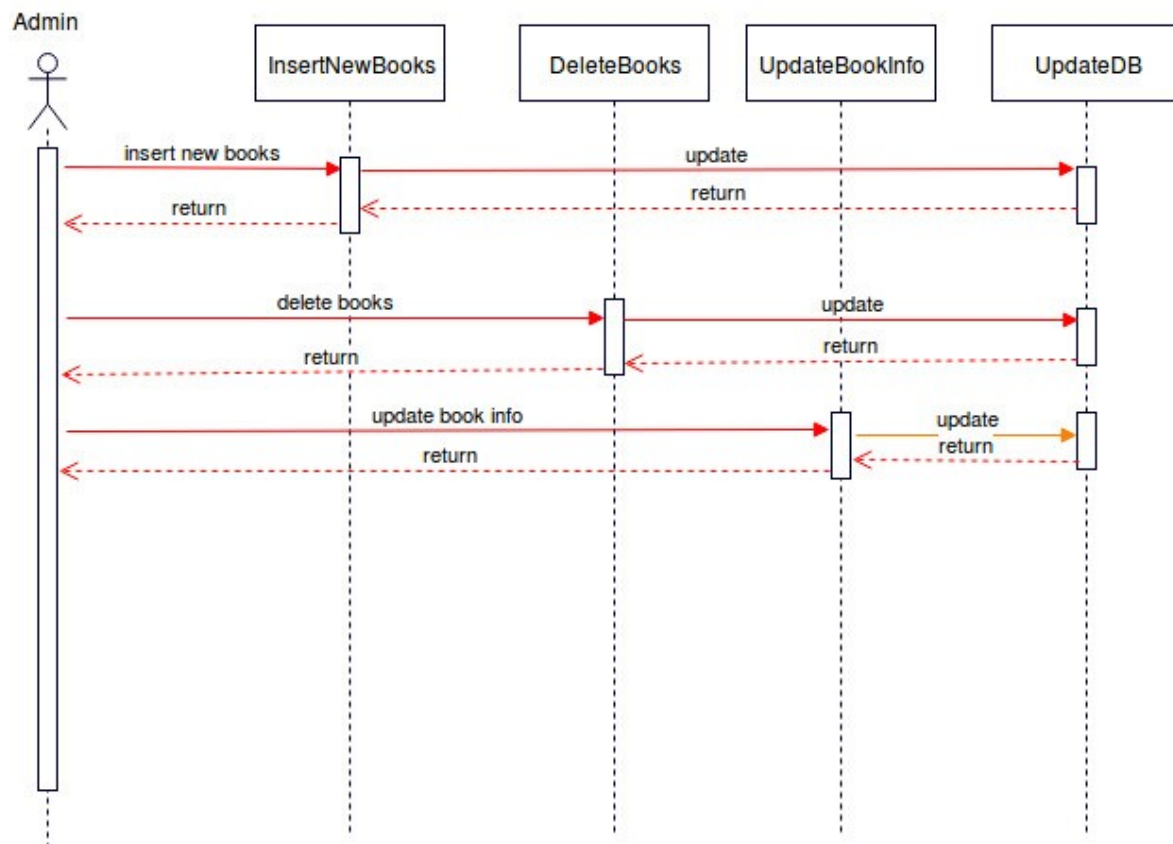
**Fig 4: User Activity Diagram**



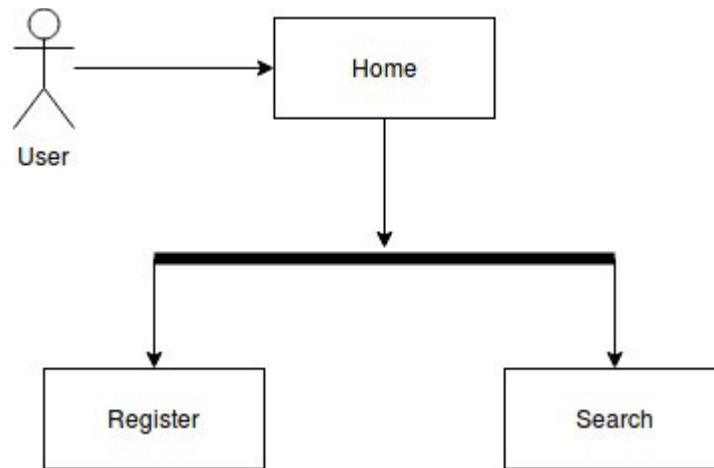
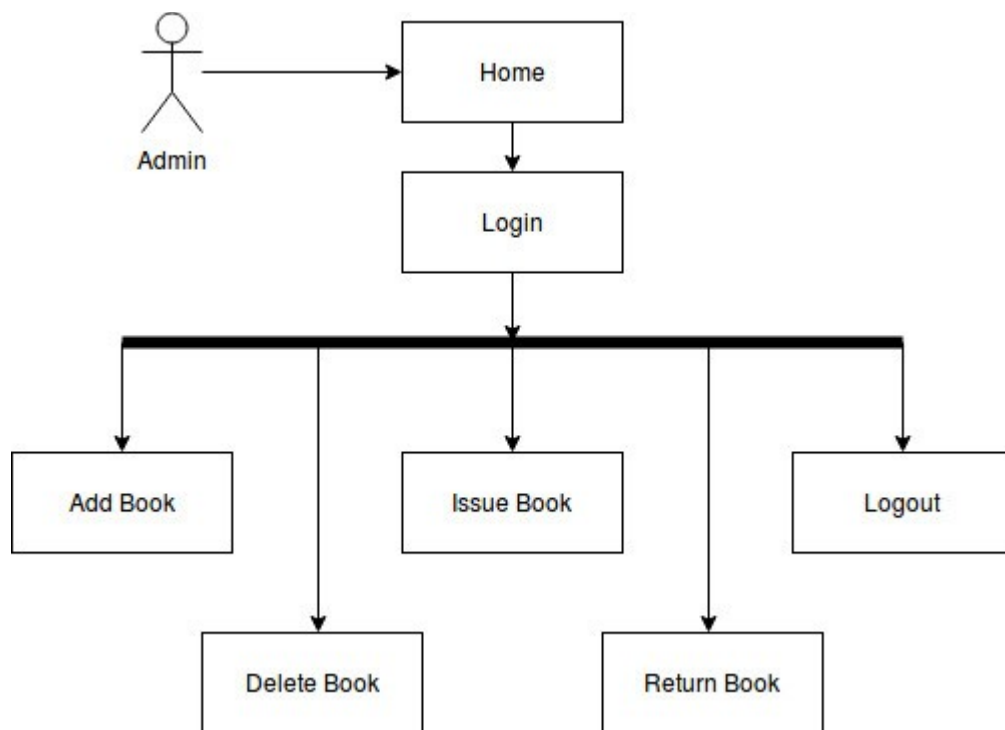
**Fig 5: Login Sequential Diagram**



**Fig 6: Admin Sequential Diagram for Issue/Return of Books**



**Fig 7: Admin Sequential Diagram to update database**

**Fig 8: User Navigation Diagram****Fig 9: Admin Navigation Diagram**

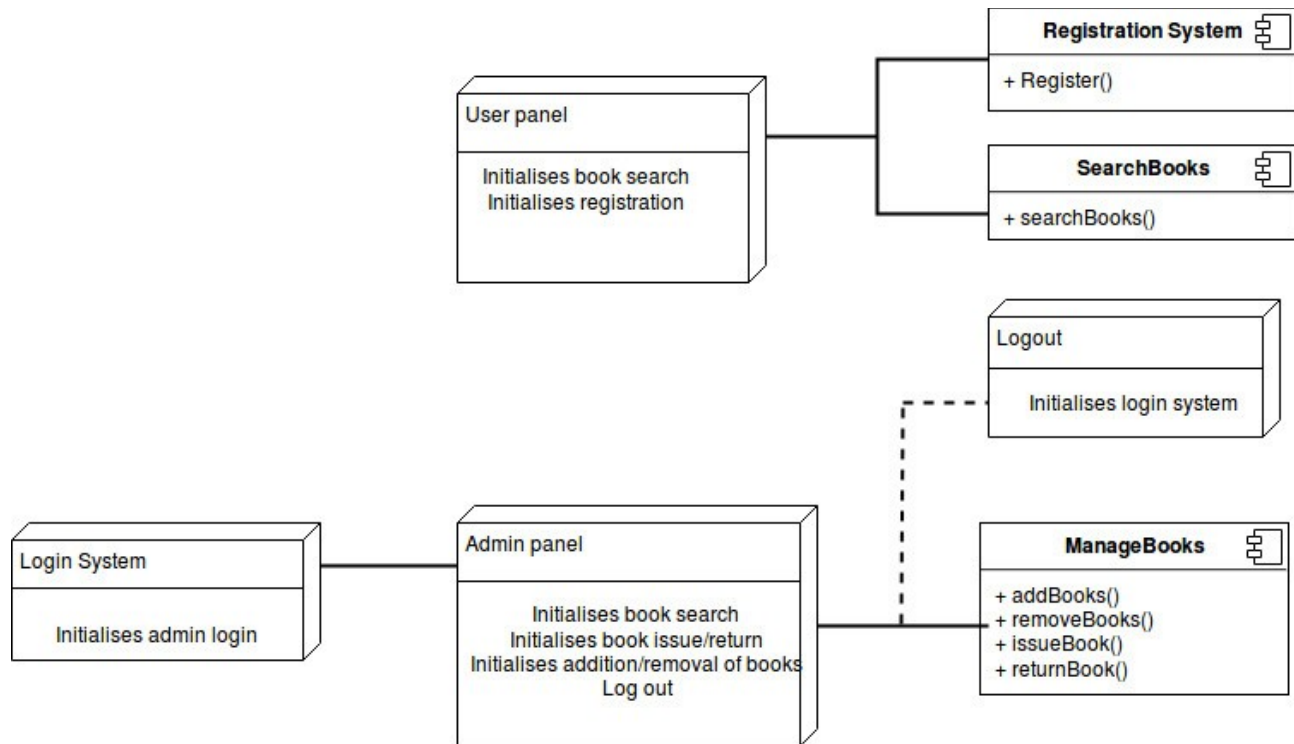


Fig 10: Deployment Diagram

## Appendix C: To Be Determined List