Software Requirements Specification

Version 1.0

August 15, 2018

Library Management System

Shovan Kumar Behera

Rupsita Parida

Shakti Prasad Mishra

Submitted in partial fulfillment

of the requirements of

Software Engineering

# Table of Contents

[Table of Contents i](#_Toc77487619)

[1.0. Introduction 1](#_Toc77487621)

1.1. Project Description 1

1.2. List of Actors 1

[1.3. Scope of Projects 2](#_Toc77487624)

[1.4. References 2](#_Toc77487625)

[2.0. Overall Description 4](#_Toc77487627)

[2.1 Functional Requirements 5](#_Toc77487629)

[2.1.1 Use Case 5](#_Toc77487630)s

2.1.2 Entity Relationship Diagrams

2.1.3 Data Flow Diagrams

[2.2 Non-Functional Requirements 15](#_Toc77487649)

2.2.1 System Requirements

2.2.2 Performance Requirements

2.2.3 Security Requirments

2.2.4 Database Requirments

2.2.5 Safety Requirments

1. **Introduction**

**1.1 Project Description**

The purpose of this document is to present a detailed description of the Library Management System. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both the users and administers of the system and will be proposed to the educational institutions for its approval.

**1.2 List of Actors**

Admin (Librarian), Issuer (Student/ Teacher)

**1.3 Scope of Projects**

This software system will be a Library Management System for a university. This system will be designed to manage the issuing of books efficiently by providing tools to assist in automating the book issue, return date and generation of fine, if any, which would otherwise have to be performed manually. By maximizing the librarian’s work efficiency the system will meet the users’ needs while remaining easy to understand and use.

More specifically, this system will keep a track of what and how many books a person issues on a regular basis. It’ll also keep a record of the number of copies of individual books that the library owns and currently has available for issuing. The system will also make it easy for the users to find their choice of books by categorising the books under authors, subjects and editions. The system also contains a relational database for all issuers and books.

**1.4 References**

Example: IEEE. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.

1. **Overall Description**

**2.1 Product Functions**

There will be two different users who can use this product in different way:

a) **Librarian :**

Librarian will have access to any information and main control.

Librarian can add, modify or delete resources.

Librarian can register new members.

Librarian can check in and check out the resources.

Librarian has control to the payment system.

b) **Registered Members** :

Member can search the availability of needed resources.

Members can renew their resources.

Members can view their membership information, rental history information,

payment information, due dates, etc. online.

**2.2 User Characteristics**

There are 2 users in our Library Management System product:

a) **Librarian**: They have administrator control over their portion of library system. They can

access any of their member’s information. They can manage the resource and perform

operation on resource such as adding, removing or modifying the resource information that

are in library database server. They checks in/checks out resource for members. Librarian are

expected to have basic knowledge of on how to use computer system/software. They need to

take a week training on how to use Library Management System.

b) **Registered Members**: Users are supposed to be registered first in the Library Management System and pay the membership fee to be registered member of the library. Then, they will have access to library resources. They can search for the available resources. They can also view the account with

the information such as rental history, due dates, payment information, and membership information. Members are at least required to know how to use the Library Management System software.

**2.3 General Constraints**

1. **Storage Constraint**: Every Library subscribed to LMS will have 20 GB of Cloud Storage

allocated.

2. **High Level Language Requirement**: Software should be in English language.

3. **Reliability Constraint**: System should be updated to backup server frequently in order to

provide fault tolerance capability.

4. **Implementation Constraints**: Implementation of application should be in Python.

**2.4 Functional Requirements**

**2.4.1 Use Case**

Example: Search Book

Diagram:

RETURN

Issuer

ISSUE

**Brief Description:**

The Issuer searches for a book in the library and issues it if available.

**Initial Step-By-Step Description:**

**For Issue:**

The Issuer searches for a book by book name, author name, subject, and edition.

If available the Issuer goes to the Librarian to issue the book.

The system displays the information of the book i.e., book id, book name, author, subject and total no. of issues to the Librarian.

The system displays the information of the Issuer i.e., Name, Book Id, Date of Issue, Date of return, Fine if any by the Id of the Issuer.

If more than three books haven’t been returned, the system rejects to issue any book else the system allows proceeding further.

The Librarian enters the details in the Issuers database and decrements the availability column of the book in Book database.

The Issuer can now take the book.

**For Return:**

The Issuer goes to the Librarian to return the book.

The system displays the information of the book i.e., book id, book name, author, subject and total no. of issues to the Librarian.

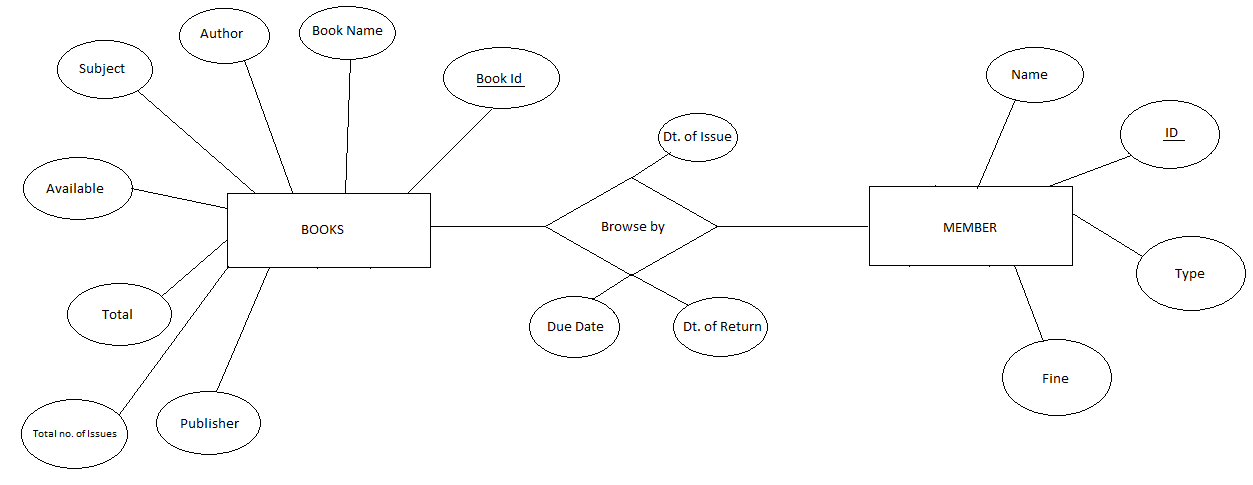
The system displays the information of the Issuer i.e., Name, Book Id, Date of Issue, Date of return, Fine if any by the Id of the Issuer.

If the Issuer is finned with some amount, he/she has to return the book along with the challan.

The Librarian enters the return date in the Issuers database and increments the availability column of Book in Book database.

The Issuer can now issue some other books.

**2.4.2 Entity Relationship Diagrams**



**Member Data Entity**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| Member Id | Test | Id of the Member | Key value |
| Name | Text | Name of the Member |  |
| Type | Text | Student/Teacher | Issuer is Student or Teacher |
| Book Id | Text | Id of the Book | Unique Id for each book |
| Date of Issue | Date | Date when the book was issued |  |
| Due Date | Date | Date till no fine | After due date the member would be charged with some fine |
| Date of Return | Date | Date when the book was returned |  |
| Fine | Integer | Amount to be paid as fine due to late return |  |

**Book Data Entity**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| ID | Text | ID of the book | Used as key |
| Name | Text | Name of the book |  |
| Author | Text | Author of the book |  |
| Publisher | Text | Publisher of the book |  |
| Subject | Text | Book belongs to which subject |  |
| Availability | Boolean | If the book is available or not | True or False |
| Total | Integer | Total no of books | Total no of books available in library of specified type |
| Total no. of issues | Integer | Total no. of issues till day | Total no of issues till day of specified type |

**2.1.3 Data Flow Diagram**

1. **Specific Requirements**

**3.1 Functional Requirements**

**3.1.1 Members :**

**3.1.1.1 Logging In**

The System shall verify valid ID and Password.

The System shall not allow to enter the system with invalid ID or Password

The System shall allow member to enter with valid ID and Password

**3.1.1.2 Search**

The System shall allow members to be able to search for available resource by title,

Resource ID, Author Name.

The System shall display the search results with the details of preferred Resource.

**3.1.1.3 Renew Resource**

The System shall allow members to renew the resource within 72 hours before due date.

**3.1.1.4 Manage/View Account**

The System shall allow members to view history containing information such as list of

resource rented, due date, renewal date, and status.

The System shall allow update their personal information such as phone number, address,

email address, password.

**3.1.1.5 Payment Management**

The System shall display the due payment, description and deadline to the customer.

The System shall provide member payment facility that will store payment information

and data.

The system shall collect and verify payment information made by member.

The System shall allow member to see their payment history containing information

such as purpose, payment ID, date of payment.

**3.1.2 Librarian :**

**3.1.2.1 Admin Access**

The System shall verify admin login info to provide all access privilege.

**3.1.2.2 Register Members**

The System shall store the registered id in the membership database.

**3.1.2.3 Update Resource**

The System shall allow add, delete or edit resource with their details such as resource

id, resource name, description of resource, location, category and author.

**3.1.2.4 Check In/Check Out**

The System shall record the transaction details such as Resource ID, Member ID, Due

Date, Check out Date, Checked in Date, Status.

**3.2 Non- Functional Requirements**

**3.2.1 Performance Requirements**

Database should be updated within a second.

Search results should be displayed within a second.

User Interface shall not take more than 3 seconds to load.

Login should be validated within 2 seconds.

**3.2.2 Security Requirements**

Every external communication between data server and end user takes place through VPN.

Payment Information are protected and encrypted.

Payment transaction is done with HTTP over Secure Protocol.

**3.2.3 Safety Requirements**

In the event of failure, there should be another data server be on standby to provide fault

tolerance capability.

**3.2.4 Capacity Requirements**

Not more than 10,000 members to be registered.

**3.3 User Interface Requirement**

The UI is loaded from the server to any web browser. So our UI is compatible to any browser such

as Mozilla Firefox, Google Chrome, Safari, Internet Explorer etc. It also provide responsive design

so it can be viewed or operated from the Mobile Browsers as well.

**3.4 Communication Interface**

We will be using HTTPS/HTTP protocol for the communication over the server.

**3.5 Hardware and Software Specifications**

|  |  |  |  |
| --- | --- | --- | --- |
|  | Standard Client | Standard  Web Application  Server | Standard  Database Server |
| Hardware | 200 GB HDD  Dual Core Processor  1 GHz  Monitor 1024x768  1 GB RAM | 1TB HDD  Quad Core  Processor  3 GHZ | 20TB HDD  RAID  Quad Core  Processor |
| Operating  System | Any | Linux | Linux |
| Software | Any Web Browser | Apache/Python | Mongo DB |
| Network | 1 Mbps Ethernet | Dual 100 Mbps  Ethernet | Dual 500  Mbps Ethernet |

1. **Use Case Analysis**

|  |  |  |
| --- | --- | --- |
| **Use Case Name:** Register the member | **ID:** UC1 | **Priority:** High |
| **Actor:** Librarian | | |
| **Description:** Librarian collects the information of the person who wishes to get membership, fill up the information in the portal and registers for the library. | | |
| **Trigger:** A person requests librarian to register for the library. | | |
| **Preconditions:**   1. The librarian collects required credentials and information to be registered as member. | | |
| **Normal Course:**    1. The librarian opens the membership database portal.  2. Fills Up the required information such as name, address, contact, ID of the member.  3. Submits the filled information to the registered member.  4. Server verifies and validate the filled information.  5. Librarians post the payment for the registered member.  6. Server verifies the authenticity of the payment for membership registration.  7. Server creates the profile for registered member. | | |
| **Post conditions:** Newly registered member is activated and fully unlocked to use LMS service. | | |

|  |  |  |
| --- | --- | --- |
| **Use Case Name:** Add a resource | **ID:** UC2 | **Priority:** High |
| **Actor:** Librarian | | |
| **Description:** It adds the available information and description of the resource. | | |
| **Trigger:** 1. Resource is newly available | | |
| **Preconditions:**  1. Librarian logs in the librarian portal.  2. Librarian identity is verified.  3. Generates resource id, and collects the information on new resource | | |
| **Normal Course:**  1. Librarian enters into resource database portal  2. Librarian clicks on add on.  3. System requests the information on resource.  4. Librarian enters the details of resource.  5. System updates the resource details on the resource database. | | |
| **Post conditions:** System updates the newly added resource’s status as available. | | |

|  |  |  |
| --- | --- | --- |
| **Use Case Name:** Delete a resource | **ID:** UC3 | **Priority:** |
| **Actor:** Librarian | | |
| **Description:** When a resource is no more available, resource needs to be deleted and librarian has access to perform the delete resource operation. | | |
| **Trigger:** Resource is no more available. | | |
| **Preconditions:**  1. Librarian logs in the librarian portal.  2. Librarian identity is verified.  3. Identifies the resource to be removed from the database. | | |
| **Normal Course:**  1. Librarian enters into resource database portal  2. Librarian search for the resource in the database.  3. System shows the details on the searched resource.  4. Librarian selects the remove operation on resource  5. System asks for the confirmation.  6. Librarian confirms the operation.  7. System deletes the resource from database. | | |

|  |  |  |
| --- | --- | --- |
| **Use Case Name:** Modify Resource | **ID:** UC4 | **Priority:** High |
| **Actor:** Librarian | | |
| **Description:** When a resource is information needs to be changed, resource details on database has to be updated. | | |
| **Trigger:** Resource information has changed. | | |
| **Preconditions:**  1. Librarian logs in the librarian portal.  2. Librarian identity is verified.  3. Identifies the resource to be modified. | | |
| **Normal Course:**  1. Librarian enters into resource database portal  2. Librarian search for the resource in the database.  3. System shows the details on the searched resource.  4. Librarian selects the modify operation on resource  5. System enables the modification operation and allows to modify the previous details.  6. Librarian confirms the operation.  7. System updates the resource information. | | |

**Use Case Name:** Check Out **ID:** UC5 **Priority:** High

**Actor:** Librarian

**Description:** When a member picks up the resource from library to rent, then librarian need to

check out the books.

**Trigger:** Member wants to rent resource from library

**Preconditions:**

1. Librarian logs in the librarian portal.

2. Librarian identity is verified.

**Normal Course:**

1. Librarian scans the resource barcode.

2. Resource details is displayed.

3. Librarian clicks on check out.

4. System request the member ID to whom resource is being rented.

5. Librarian provides the member ID.

6. System updates the status of resource to ‘Rented’

7. System updates the member’s entry of list of rents.

8. System updates the due date on the respective entry.

**Use Case Name:** Check In **ID:** UC6 **Priority:** High

**Actor:** Librarian

**Description:** When a member drops up the resource, then librarian have to check in the book.

**Trigger:** Member brings the resource back to library.

**Preconditions:**

1. Librarian logs in the librarian portal.

2. Librarian identity is verified.

**Normal Course:**

1. Librarian scans the resource barcode.

2. Resource details is displayed.

3. Librarian clicks on check in.

4. System updates the status of resource to ‘Available’

5. System removes the entry from the member’s list of rents.

**Use Case Name:** View Member Information **ID:** UC7 **Priority:** High

**Actor:** Librarian

**Description:** When Librarian need to get details of member

**Trigger:** Someone authorized requests the information on member, it can be member itself as well

**Preconditions:**

1. Librarian logs in the librarian portal.

2. Librarian identity is verified.

**Normal Course:**

1. Librarian search for member by ID or Name

2. Members display icons shows

3. Librarian clicks member icon.

4. System Displays member account window

5. System display member’s personal information, rental status and history, and payment

history/info.

**Use Case Name:** Search for Resource **ID:** UC8 **Priority:** High

**Actor:** Member

**Description:** When members require to check whether the necessary resource is available in the

library or not, they need to search in the resource database.

**Trigger:** Members looks for the resource

**Preconditions:**

1. Registered member logs into LMS.

**Normal Course:**

1. Members clicks on search window.

2. System requests any information on resource such as Resource ID, Name and Author.

3. Member enters the information and clicks on search.

4. System response with list of available resource that matches the search.

**Use Case Name:** Renew Resource **ID:** UC9 **Priority:** High

**Actor:** Member

**Description:** Resources that are within 72 hrs. Before due date needs to be renewed by member.

**Trigger:** 72 hrs. Before the due date.

**Preconditions:**

1. Registered member logs into LMS.

**Normal Course:**

1. Members are notified it’s time to renew their book or check in.

2. Member’s click on the resource entry that’s to be renewed.

3. Members renew the resources.

4. System extends the due date.

**Use Case Name:** Pay the due **ID:** UC10 **Priority:** High

**Actor:** Member

**Description:** Dues and the description are displayed on the members account and need to be paid

through the payment portal.

**Trigger:**

**1.** Membership Renewal

**2.** Fine after due date of resource rental

**3.** When member rents the resource that costs for rent.

**Preconditions:**

1. Registered member logs into LMS.

**Normal Course:**

1. Amount to be paid are displayed on member’s account.

2. Members click on the payment portal.

3. System request the payment information.

4. Member provides the payment information.

5. System process the payment

6. System provides the payment receipt

7. Updates the payment information in the data server.