FUNCTIONAL REQUIREMENT DOCUMENT

Library Management System (LMS) Website – EdTech

Document Information

|  |  |
| --- | --- |
| Document Attributes | Description |
| Document ID | Doc001 |
| Author | Sandeep Taksande |
| Contributor | - |

Revision History

|  |  |  |
| --- | --- | --- |
| Version | Date | Changes |
| Draft | 01-October-2024 | New Document |
| V0.1 |  |  |
|  |  |  |

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Reviewer/ Approver | Role | Comments | Date |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Contents

[1. Introduction 4](#_Toc179195888)

[1.1 Purpose 4](#_Toc179195889)

[1.2 Scope 4](#_Toc179195890)

[2. Functional Requirements 4](#_Toc179195891)

[2.1 Book Management 4](#_Toc179195892)

[2.2 Member Management 5](#_Toc179195893)

[2.3 Transaction Management 6](#_Toc179195894)

[2.4 Reporting and Analytics 6](#_Toc179195895)

[3. Non-Functional Requirements 7](#_Toc179195896)

[3.1 Performance 7](#_Toc179195897)

[3.2 Security 7](#_Toc179195898)

[3.3 Scalability 7](#_Toc179195899)

[3.4 Usability 7](#_Toc179195900)

[3.5 Accessibility 7](#_Toc179195901)

[3.6 Integration 7](#_Toc179195902)

[4. Use Cases 7](#_Toc179195903)

# 1. Introduction

## 1.1 Purpose

This Functional Requirements Document (FRD) outlines the detailed functional requirements for the Library Management System (LMS) based on the Business Requirements Document (BRD). The FRD provides a more granular level of detail, specifying the specific features, functionalities, and behaviours of the LMS.

## 1.2 Scope

The FRD covers the same key areas as the BRD, including:

* Book Management
* Member Management
* Transaction Management
* Inventory Management
* Reporting and Analytics

# 2. Functional Requirements

## 2.1 Book Management

* **Cataloguing**
  + **Requirement:** The system shall allow users to create, edit, and delete book records.
  + **Acceptance Criteria:**
    - Users can add new books with required metadata (title, author, ISBN, publication date, subject, keywords).
    - Users can modify existing book records.
    - Users can delete book records permanently.
  + **Additional Details:**
    - The system shall support multiple book formats (e.g., books, e-books, audiobooks).
    - The system shall allow for the import and export of book data in standard formats (e.g., MARC, XML).
* **Circulation**
  + **Requirement:** The system shall manage the circulation of books, including check-ins, check-outs, renewals, and reservations.
  + **Acceptance Criteria:**
    - Users can check out books to members within their borrowing limits.
    - The system automatically calculates due dates and sends overdue notifications.
    - Users can renew books before their due date.
    - Users can place holds on books that are currently checked out.
  + **Additional Details:**
    - The system shall track the location of books (on the shelf, checked out, lost, etc.).
    - The system shall calculate and apply fines for overdue books.
* **Inventory Management**
  + **Requirement:** The system shall track the inventory of books, including stock levels, damage assessment, and disposal.
  + **Acceptance Criteria:**
    - The system maintains accurate stock levels for all books.
    - Users can report damaged books and track their repair status.
    - Users can initiate the disposal process for damaged or outdated books.
  + **Additional Details:**
    - The system shall support periodic inventory audits.
    - The system shall generate reports on inventory levels and usage.

## 2.2 Member Management

* **Registration**
  + **Requirement:** The system shall allow users to register new members and manage their profiles.
  + **Acceptance Criteria:**
    - Users can create new member accounts with required information (name, address, contact details, etc.).
    - Users can update their member profiles.
    - The system shall assign membership types and borrowing limits based on predefined criteria.
  + **Additional Details:**
    - The system shall support online and in-person registration.
    - The system shall integrate with a payment gateway for membership fees.
* **Overdue Notifications**
  + **Requirement:** The system shall send automatic notifications to members with overdue books.
  + **Acceptance Criteria:**
    - The system sends email or SMS notifications to members with overdue books.
    - Notifications are sent at predefined intervals (e.g., daily, weekly).
  + **Additional Details:**
    - The system shall allow users to customize notification preferences.

## 2.3 Transaction Management

* **Transaction History**
  + **Requirement:** The system shall record and track all transactions, including check-ins, check-outs, renewals, and fines.
  + **Acceptance Criteria:**
    - The system maintains a complete transaction history for each member.
    - Users can search for and view specific transactions.
    - The system generates transaction reports in various formats (e.g., PDF, CSV).
  + **Additional Details:**
    - The system shall support integration with a financial management system.

## 2.4 Reporting and Analytics

* **Customizable Reports**
  + **Requirement:** The system shall generate customizable reports on library usage, circulation statistics, inventory levels, and financial performance.
  + **Acceptance Criteria:**
    - Users can create custom reports based on their specific needs.
    - Reports can be filtered and sorted by various criteria.
    - Reports can be exported in different formats (e.g., PDF, CSV).
  + **Additional Details:**
    - The system shall provide pre-defined report templates for common use cases.
* **Data Visualization**
  + **Requirement:** The system shall provide data visualization tools to help users understand and analyse library data.
  + **Acceptance Criteria:**
    - Users can create charts and graphs to visualize data.
    - Charts and graphs can be customized with different styles and formats.
  + **Additional Details:**
    - The system shall support integration with external data visualization tools.

# 3. Non-Functional Requirements

## 3.1 Performance

* The LMS should be responsive and handle high transaction volumes efficiently.
* Load testing should be conducted to ensure system performance under peak usage.

## 3.2 Security

* Implement robust security measures to protect sensitive data (e.g., member information, financial records).
* Adhere to industry best practices for data privacy and protection.

## 3.3 Scalability

* The LMS should be scalable to accommodate future growth in the library's collection and membership.

## 3.4 Usability

* The user interface should be intuitive and easy to navigate for both staff and members.
* Provide clear instructions and help documentation.

## 3.5 Accessibility

* Ensure the LMS is accessible to users with disabilities, complying with relevant accessibility standards (e.g., WCAG).

## 3.6 Integration

* If applicable, consider integrating the LMS with other systems (e.g., library catalog, financial management software).

# 4. Use Cases

**Use Case 1: Member Registration**

* **Actor:** Potential Member
* **Preconditions:** None
* **Postconditions:** A new member account is created in the system.
* **Steps:**
  1. The potential member visits the library or accesses the online registration portal.
  2. The member provides personal information (name, address, contact details, etc.).
  3. The system validates the input data and creates a new member account.
  4. The system assigns a membership type and borrowing limits based on predefined criteria.
  5. The system sends a confirmation email or message to the member.

**Use Case 2: Book Checkout**

* **Actor:** Member
* **Preconditions:** The member has a valid membership and the book is available.
* **Postconditions:** The book is checked out to the member.
* **Steps:**
  1. The member selects the book they want to borrow.
  2. The librarian scans the book's barcode and the member's barcode.
  3. The system verifies the member's borrowing limits and checks if the book is available.
  4. The system calculates the due date and records the check-out transaction.
  5. The librarian hands the book to the member.

**Use Case 3: Book Return**

* **Actor:** Member
* **Preconditions:** The book is checked out to the member.
* **Postconditions:** The book is checked in to the library.
* **Steps:**
  1. The member brings the book to the library.
  2. The librarian scans the book's barcode and the member's barcode.
  3. The system verifies that the book is checked out to the member.
  4. The system records the check-in transaction and updates the book's availability.
  5. The librarian returns the book to the shelves.

**Use Case 4: Book Search**

* **Actor:** Member or Librarian
* **Preconditions:** None
* **Postconditions:** The search results are displayed.
* **Steps:**
  1. The user enters search criteria (e.g., title, author, subject, keywords).
  2. The system searches the catalogue for matching books.
  3. The system displays the search results, including book details and availability.

**Use Case 5: Overdue Notifications**

* **Actor:** System
* **Preconditions:** A book is overdue.
* **Postconditions:** A notification is sent to the member.
* **Steps:**
  1. The system checks for overdue books on a regular basis.
  2. If a book is overdue, the system sends a notification to the member.
  3. The notification includes the book title, due date, and any applicable fines.

**Use Case 6: Report Generation**

* **Actor:** Librarian or Administrator
* **Preconditions:** None
* **Postconditions:** The report is generated and displayed.
* **Steps:**
  1. The user selects the type of report they want to generate (e.g., circulation statistics, inventory report, financial report).
  2. The system collects the necessary data and generates the report.
  3. The user can view, print, or export the report.

**Use Case 7: Member Renewal**

* **Actor:** Member
* **Preconditions:** The member's membership is about to expire.
* **Postconditions:** The member's membership is renewed.
* **Steps:**
  1. The member visits the library or accesses the online renewal portal.
  2. The system verifies the member's existing membership information.
  3. The member pays the renewal fee (if applicable).
  4. The system updates the member's membership expiration date.
  5. The system sends a confirmation email or message to the member.

**Use Case 8: Book Reservation**

* **Actor:** Member
* **Preconditions:** The book is currently checked out or on hold.
* **Postconditions:** The book is reserved for the member.
* **Steps:**
  1. The member places a hold on the book.
  2. The system records the reservation and notifies the member when the book becomes available.
  3. The member is notified when the book is ready for pickup.
  4. The member has a specified timeframe to pick up the reserved book.

**Use Case 9: Fine Payment**

* **Actor:** Member
* **Preconditions:** The member has overdue fines.
* **Postconditions:** The fines are paid.
* **Steps:**
  1. The member visits the library or accesses the online payment portal.
  2. The system calculates the total amount of fines owed.
  3. The member pays the fines using a preferred payment method.
  4. The system updates the member's account to reflect the payment.
  5. The system sends a confirmation email or message to the member.

**Use Case 10: Damage Assessment**

* **Actor:** Librarian
* **Preconditions:** A book is returned damaged.
* **Postconditions:** The damage is assessed, and appropriate action is taken.
* **Steps:**
  1. The librarian inspects the damaged book.
  2. The librarian determines the extent of the damage and its cause.
  3. The librarian records the damage information in the system.
  4. The librarian decides whether to repair the book or dispose of it.
  5. If the book is repairable, the librarian sends it for repair.
  6. If the book is not repairable, the librarian initiates the disposal process.

**Use Case 11: Inventory Audit**

* **Actor:** Librarian or Administrator
* **Preconditions:** None
* **Postconditions:** The inventory is audited, and any discrepancies are addressed.
* **Steps:**
  1. The librarian or administrator conducts a physical inventory of the library's collection.
  2. The librarian or administrator compares the physical inventory with the system's inventory records.
  3. Any discrepancies (e.g., missing books, damaged books, incorrect stock levels) are identified.
  4. The librarian or administrator investigates the cause of discrepancies and takes corrective action.

**Use Case 12: Staff Management**

* **Actor:** Administrator
* **Preconditions:** The library has staff members.
* **Postconditions:** Staff information is managed, and tasks are assigned.
* **Steps:**
  1. The administrator creates staff accounts and assigns roles and permissions.
  2. The administrator manages staff schedules and time off requests.
  3. The administrator assigns tasks and responsibilities to staff members.
  4. The administrator monitors staff performance and provides feedback.

**Use Case 13: System Administration**

* **Actor:** System Administrator
* **Preconditions:** The LMS is operational.
* **Postconditions:** System settings and configurations are managed.
* **Steps:**
  1. The system administrator configures system settings (e.g., library hours, overdue fines, membership types).
  2. The system administrator manages user accounts and permissions.
  3. The system administrator performs system maintenance tasks (e.g., backups, updates, troubleshooting).
  4. The system administrator monitors system performance and addresses any issues.