Software Requirements Specification

for

Book Management

Version 1.0 approved

Ankul Jogi

Harsh Chauhan

Harsh Kholiya

Rk University

Date:-24-02-2024

Table of Contents

Table of Contents 2
Revision History 2

1. Introduction 1
   1.1 Purpose 1
   1.2 Document Conventions 1
   1.3 Intended Audience and Reading Suggestions 1
   1.4 Project Scope 1
   1.5 References ..... Error! Bookmark not defined.
2. Overall Description 1
   2.1 Product Perspective 1
   2.2 Product Features 1
   2.3 User Classes and Characteristics 2
   2.4 Operating Environment 2
   2.5 Design and Implementation Constraints 2
   2.6 User Documentation 2
   2.7 Assumptions and Dependencies 2
3. System Features 2
   3.1 System Feature 1 2
   3.2 System Feature 2 (and so on) 3
4. External Interface Requirements 4
   4.1 User Interfaces 4
   4.2 Hardware Interfaces 4
   4.3 Software Interfaces 4
   4.4 Communications Interfaces 4
5. Other Nonfunctional Requirements 4
   5.1 Performance Requirements 4
   5.2 Safety Requirements 5
   5.3 Security Requirements 5
   5.4 Software Quality Attributes 5
   6. Other Requirements ..... 6 Appendix A: Glossary 6
   Appendix B: Analysis Models ..... 6 Appendix C: Issues List 6

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

The purpose of this document is to provide a detailed description of the requirements for the Book Management System. It will outline the functional and non-functional requirements, constraints, and interfaces of the system.

1.2 Document Conventions

Use of UML diagrams for visual representation of system components.
Use of specific terminology defined in the glossary section.

1.3 Intended Audience and Reading Suggestions

This document is intended for developers, testers, project managers, and stakeholders involved in the development and maintenance of the Book Management System. It is recommended to read the document thoroughly to understand the requirements and scope of the project.

1.4 Project Scope

The Book Management System is designed to manage the inventory, sales, and customer information of a bookstore. It will include features such as adding/editing books, managing customer orders, generating reports, and tracking inventory.

2. Overall Description

2.1 Product Perspective

The book management system is a standalone application designed to help manage books in a library or similar setting. It will provide features for librarians to manage the library's collection, track book loans, and provide information to library users.

2.2 Product Features

- Book Catalog: Allows librarians to add, edit, and remove books from the catalog.
- Book Search: Provides a search function for users to find books based on title, author, or category.

- User Management: Allows librarians to manage user accounts, including registration and authentication.
- Loan Management: Tracks book loans, including due dates and renewals.
- Reporting: Generates reports on book availability, user activity, and overdue books.
- Reservation System: Allows users to reserve books that are currently checked out.
- Notifications: Sends notifications to users for overdue books and reservation availability.

2.3 User Classes and Characteristics

- Librarian: Manages the library catalog, user accounts, and book loans.
- Library User: Searches for books, borrows books, and manages their account.

2.4 Operating Environment

The system will be web-based and will require a modern web browser to access. It will be developed using ASP.NET Core for the backend and React for the frontend. The system will be hosted on a server running Windows Server with IIS.

2.5 Design and Implementation Constraints

- The system must be user-friendly and intuitive for both librarians and users.
- The system must be able to handle a large number of books and users efficiently.
- The system must comply with relevant privacy and data protection regulations.

2.6 User Documentation

The system will include comprehensive user documentation that covers how to use the system, including how to search for books, borrow books, and manage user accounts. The documentation will be available online and will include step-by-step instructions and screenshots.

2.7 Assumptions and Dependencies

- The system assumes that users have basic computer literacy skills.
- The system depends on a reliable internet connection for users to access the system.
- The system depends on a database server for storing book and user information.

3. System Features

3.1 System Feature 1

3.1.1 Description and Priority

- Description: Allows users to add, edit, and delete books from the system.
- Inputs: Book details such as title, author, ISBN, genre, and publication date.
- Processing: Validation of inputs, storage in the database.
- Outputs: Confirmation message, error messages if applicable.

3.1.2 Stimulus/Response Sequences

- Description: Enables administrators to manage user accounts.
- Inputs: User details such as username, password, email, and role (admin or regular user).
- Processing: Validation of inputs, storage in the database.
- Outputs: Confirmation message, error messages if applicable. 3.1.3 Functional Requirements

REQ-1:User Registration:

- Users can create a new account by providing a username, email, and password.
- The system should verify the uniqueness of the username and email.
- Passwords must be securely stored and encrypted.

REQ-2: Book Listing:

- The system should display a list of available books with details such as title, author, and availability status.
- Users should be able to sort and filter the list based on different criteria.

User Management:

- Administrators can add, edit, and delete user accounts.
- Users can update their profile information.

Book Return:

- Users can return borrowed books.
- The system should update the book's availability status and remove the borrower's information.

3.2 System Feature 2 (and so on)

- User Management
- Description: Enables administrators to manage user accounts.
- Inputs: User details such as username, password, email, and role (admin or regular user).
- Processing: Validation of inputs, storage in the database.
- Outputs: Confirmation message, error messages if applicable.

Administration

- Description: Provides administrative functions such as system configuration and user management.
- Inputs: Configuration settings, user management actions.
- Processing: Validation of inputs, updating the database.
- Outputs: Confirmation message, error messages if applicable.

4. External Interface Requirements

4.1 User Interfaces

- The system shall provide a user-friendly web interface for users to interact with.
- The user interface shall include:
- A search feature to find books based on title, author, genre, or ISBN.
- A list of available books with details such as title, author, and availability status.
- User registration and login forms.
- Book checkout and return functionality.
- User profile management.
- Administrative functions for user and book management.
- The user interface shall be responsive and accessible on different devices and screen sizes

4.2 Hardware Interfaces

- The system shall be compatible with standard hardware configurations, including desktops, laptops, and mobile devices.
- No specific hardware interfaces are required beyond standard web browsing capabilities.

4.3 Software Interfaces

- The system shall be developed using web technologies such as HTML, CSS, and JavaScript. The system shall use a backend database to store book and user information.
- The system shall integrate with third-party services for features such as book purchasing and notifications.
- The system shall be compatible with common web browsers such as Chrome, Firefox, and Safari.

4.4 Communications Interfaces

- The system shall use HTTP/HTTPS protocols for communication between the client and server.
- The system shall support secure communication for user authentication and data transfer.
- The system shall integrate with email and SMS services for sending notifications to users.
- The system shall handle communication with external bookstores for book purchasing functionality.

5. Other Non-functional Requirements

5.1 Performance Requirements

- The system shall be able to handle a minimum of 1000 concurrent users without significant performance degradation.
- Search operations shall return results within 2 seconds.

- The system shall be able to process book checkout and return operations within 5 seconds.

5.2 Safety Requirements

- The system shall have automated backup procedures to ensure data safety.
- User passwords shall be stored securely using encryption techniques.
- The system shall have error handling mechanisms to prevent data loss or corruption.

5.3 Security Requirements

- The system shall enforce user authentication for all user interactions.
- User passwords shall be stored securely and not be visible in plain text.
- The system shall use HTTPS to encrypt data transmitted between the client and server.
- The system shall have role-based access control to restrict user access to certain features or data.

5.4 Software Quality Attributes

- The system shall be maintainable, with well-documented code and modular design.
- The system shall be reliable, with a low rate of system failures or errors.
- The system shall be scalable, able to accommodate growth in user base and data volume.
- The system shall be user-friendly, with intuitive interfaces and clear error messages.

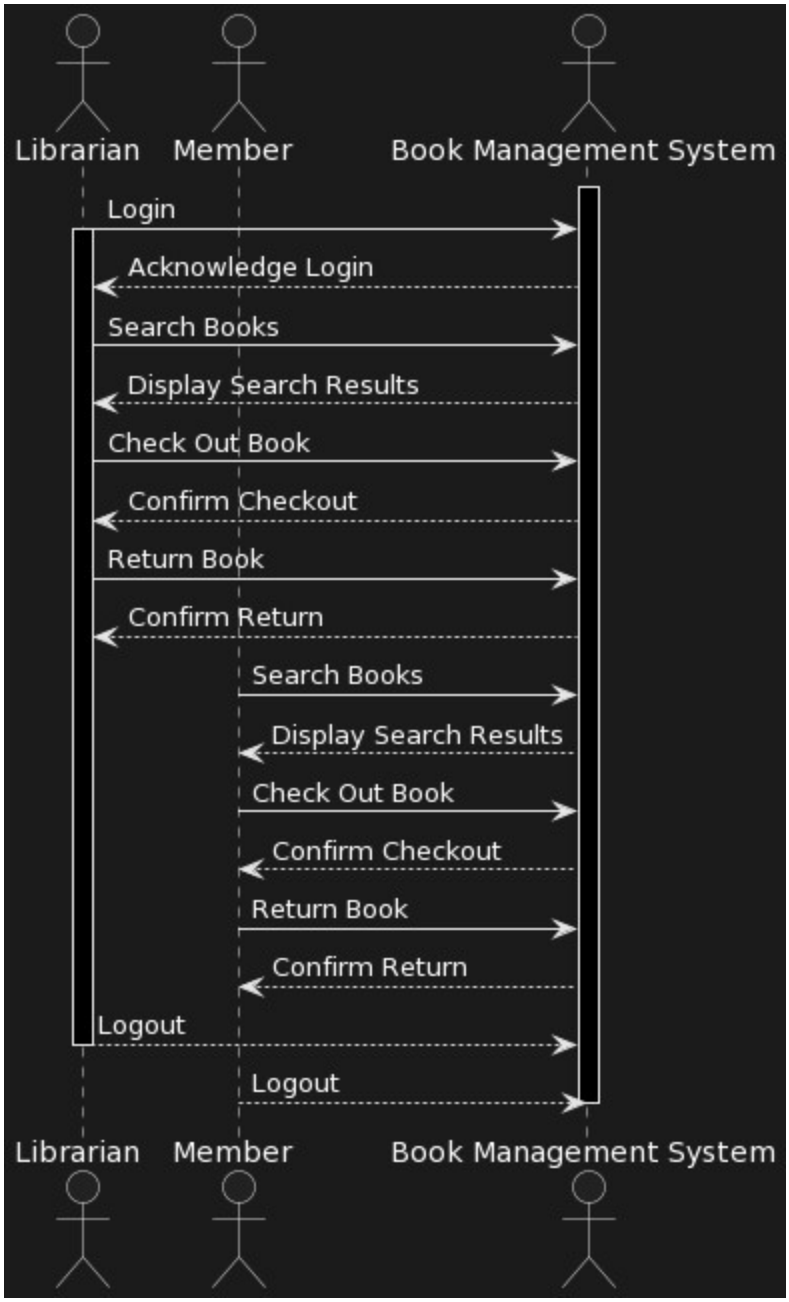
6. Other Requirements

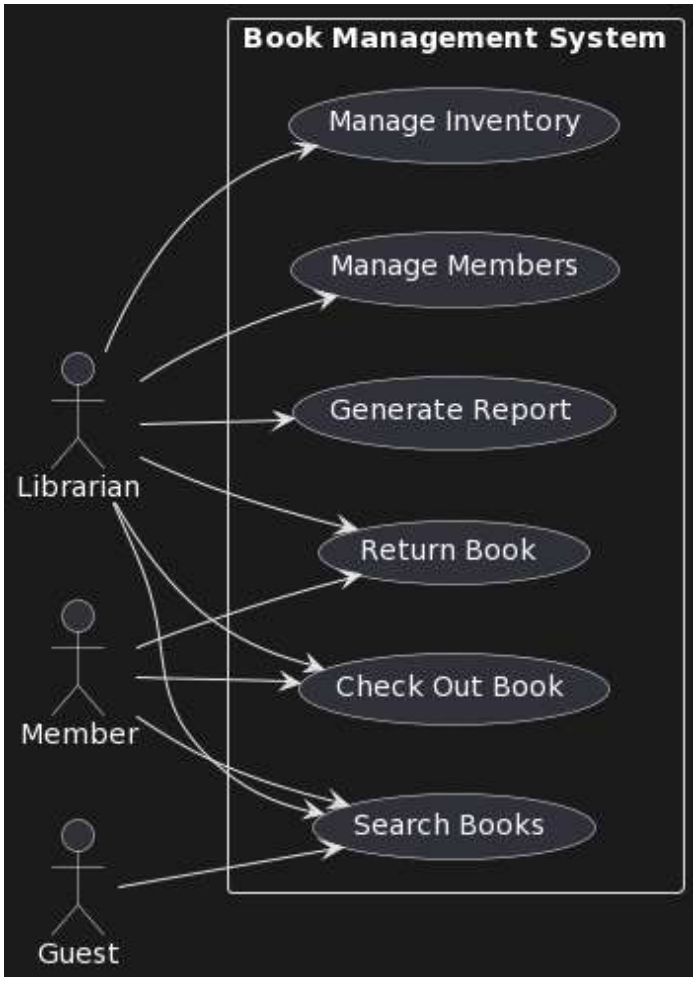
Appendix A: Glossary

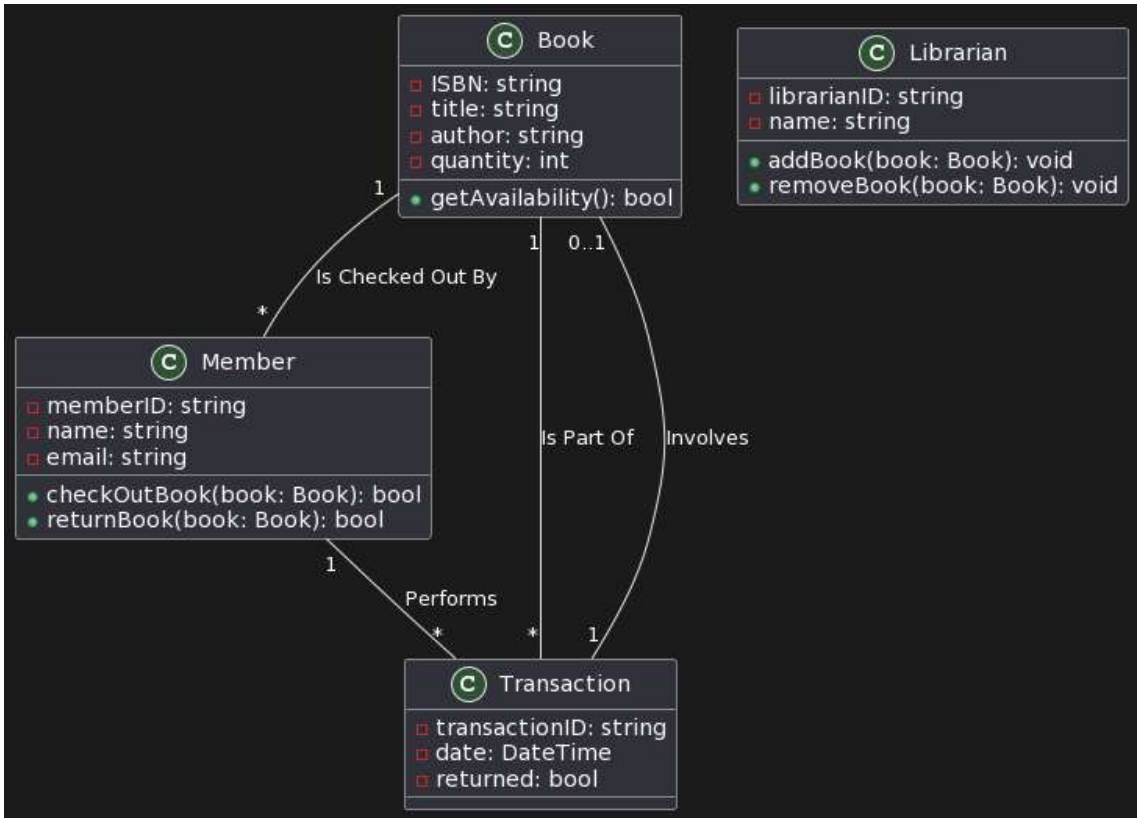
- User: A person who interacts with the book management system.
- Administrator: A user with special privileges to manage the system.
- ISBN: International Standard Book Number, a unique identifier for books.
- API: Application Programming Interface, a set of rules that allows different software applications to communicate with each other.
- UI: User Interface, the visual part of a computer application or system.
- HTTP: Hypertext Transfer Protocol, the protocol used for transmitting data over the internet.

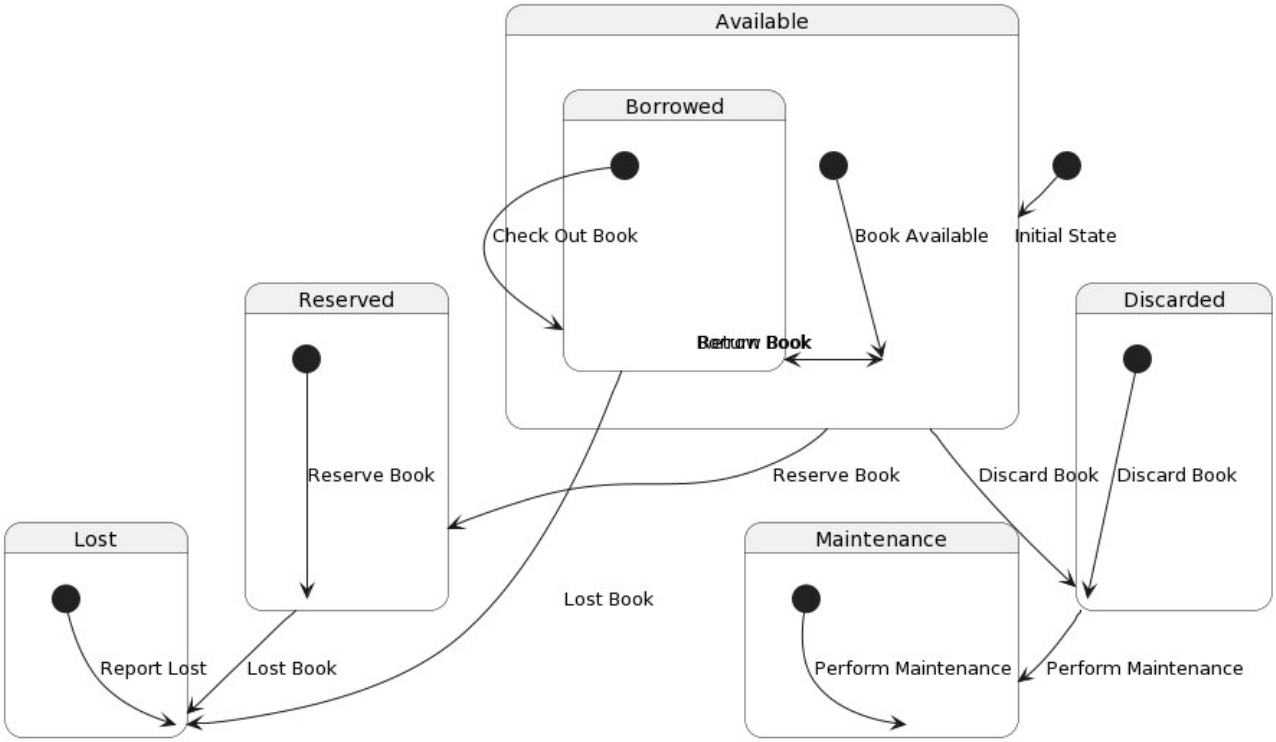
Appendix B: Analysis Models

- Use Case Diagram: A diagram showing the interactions between users and the system.
- Class Diagram: A diagram showing the classes and relationships in the system.
- Sequence Diagram: A diagram showing the sequence of interactions between objects in the system.
- State Diagram: A diagram showing the different states that an object in the system can be in.


Librarian
Member
Book Management System
Login
Acknowledge Login
Search Books
Display Search Results
Check Out Book
Confirm Checkout
Return Book
Confirm Return
Search Books
Display Search Results
Check Out Book
Confirm Checkout
Return Book
Confirm Return
Logout
Logout
Librarian
Member
Book Management System


Book Management System
Manage Inventory
Manage Members
Generate Report
Return Book
Check Out Book
Search Books
Librarian
Member
Guest


Book
ISBN: string
title: string
author: string
quantity: int
getAvailability(): bool
Librarian
librarianID: string
name: string
addBook(book: Book): void
removeBook(book: Book): void
Member
memberID: string
name: string
email: string
checkOutBook(book: Book): bool
returnBook(book: Book): bool
Transaction
transactionID: string
date: DateTime
returned: bool
1
*
Is Checked Out By
1
0..1
Is Part Of
Involves
1
Performs
*
*
1



Appendix C: Issues List

Issue 1: Search functionality does not return accurate results.
Issue 2: User registration form does not validate email addresses correctly. Issue 3: Book checkout process is slow during peak usage times.