
Software Requirements Specification

for

Smart Library Management System

Version 1.2 approved

Prepared by: Constantin Monica-Andreea

Popa Oana-Maria

Ruieneanu Dorian

University of Craiova,

Faculty of Automation, Computers and Electronics

09.03.2025

Table of Contents

1. Introduction.....	1
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
2. Overall Description.....	3
2.1 Product Perspective.....	3
2.2 Product Functions.....	3
2.3 User Classes and Characteristics.....	3
2.4 Operating Environment.....	3
2.5 Design and Implementation Constraints.....	3
2.6 User Documentation.....	4
2.7 Assumptions and Dependencies.....	4
3. External Interface Requirements.....	4
3.1 User Interfaces.....	4
3.2 Hardware Interfaces.....	11
3.3 Software Interfaces.....	12
3.4 Communications Interfaces.....	12
4. System Features.....	12
4.1 Authentication and Account Management.....	12
4.1.1 Description and Priority.....	12
4.1.2 Stimulus/Response Sequences.....	12
4.1.3 Functional Requirements.....	12
4.2 Library Catalog.....	13
4.2.1 Description and Priority.....	13
4.2.2 Stimulus/Response Sequences.....	13
4.2.3 Functional Requirements.....	13
4.3 Loan and Reservation Management.....	14
4.3.1 Description and Priority.....	14
4.3.2 Stimulus/Response Sequences.....	14
4.3.3 Functional Requirements.....	14
4.4 User Management.....	14
4.4.1 Description and Priority.....	14
4.4.2 Stimulus/Response Sequences.....	14
4.4.3 Functional Requirements.....	14
4.5 Notifications and Alerts.....	15
4.5.1 Description and Priority.....	15

4.5.2 Stimulus/Response Sequences.....	15
4.5.3 Functional Requirements.....	15
4.6 Report Generation.....	15
4.6.1 Description and Priority.....	15
4.6.2 Stimulus/Response Sequences.....	15
4.6.3 Functional Requirements.....	16
4.7 Library Resource Management.....	16
4.7.1 Description and Priority.....	16
4.7.2 Stimulus/Response Sequences.....	16
4.7.3 Functional Requirements.....	16
4.8 User Interface.....	17
4.8.1 Description and Priority.....	17
4.8.2 Stimulus/Response Sequences.....	17
4.8.3 Functional Requirements.....	17
5. Other Nonfunctional Requirements.....	17
5.1 Performance Requirements.....	17
5.2 Safety Requirements.....	17
5.3 Security Requirements.....	18
5.4 Software Quality Attributes.....	18
5.5 Business Rules.....	19
6. Other Requirements.....	19
Appendix A: Glossary.....	19
Appendix B: Analysis Models.....	20
Appendix C: To Be Determined List.....	20

Revision History

Name	Date	Reason For Changes	Version
Constantin Monica Andreea	06.03.2025	Completed 1.Introduction and 2. Overall Description	1.0
Popa Oana Maria	07.03.2025	Completed 3. External Interface Requirements and 4. System Features	1.1
Ruieneanu Dorian	09.03.2025	Completed 5. Other Nonfunctional Requirements and 6. Other Requirements	1.2

1. Introduction

1.1 Purpose

This document specifies the software requirements for the **Smart Library Management System** (SLMS). The system is designed to facilitate the management of library operations, providing an efficient way for clients to borrow books, track due dates and manage their accounts.

1.2 Document Conventions

This document follows a set of formatting and typographical conventions to ensure clarity and consistency. The key conventions are as follows:

1. **Text Formatting:**
 - **Bold text** is used to highlight section headers, key terms, and important points.
 - *Italicized text* is used for emphasis or to denote document-specific terminology.
2. **Requirement Numbering and Prioritization:**
 - Requirements are uniquely identified using the format **FREQ-XXX**, where "XXX" is a sequential number.
 - Priorities are categorized as follows:
 - **High (H)** – Critical for system functionality and must be implemented.
 - **Medium (M)** – Important but can be deferred if necessary.
 - **Low (L)** – Optional or nice-to-have features.
 - Higher-level requirement priorities are inherited by their detailed sub-requirements unless explicitly stated otherwise.
3. **Diagrams and Figures:**
 - All diagrams and figures are labeled sequentially (e.g., **Figure 1: System Architecture**).

All requirements are numbered and clearly labeled. Any technical term is defined in the Glossary section.

1.3 Intended Audience and Reading Suggestions

This document is intended for librarians, IT administrators, developers and end users of the system. It is recommended to read the introductory section to understand the purpose and structure of the document, then explore the requirements in detail.

This document is intended for various stakeholders involved in the development, management, and usage of the system. The primary audiences include:

- developers - to understand functional and non-functional requirements for system implementation.
- testers - to design test cases and validate system functionality based on the specified requirements
- system architects - to align the system design with the defined requirements.

Reading Suggestions

This Software Requirements Specification (SRS) is structured to facilitate a clear understanding of the system. It is recommended that readers follow this sequence:

1. **Introduction** – Provides an overview of the document’s purpose, scope, and key definitions. Recommended for all readers.
2. **Overall Description** – Describes the general factors affecting the system, including assumptions, dependencies, and constraints. Essential for project managers, business analysts, and system architects.
3. **External Interfaces Requirements** – Details interactions with other systems, user interfaces, and features. Important for UI/UX designers, developers, and testers.
4. **System Features**– Includes additional supporting information.
5. **Other Nonfunctional Requirements**

1.4 Product Scope

The scope of the system is to modernize the library by introducing a **smart and efficient book management system**. The final product will integrate **QR code technology** to enhance user experience, allowing clients to quickly access **book reviews, author information, and availability status** by scanning a QR code on each book.

The library information system will allow users to search for books, make reservations, borrow volumes, and receive notifications regarding return deadlines. The system also supports library management, including book inventory, user management, loans and returns, reports, and analysis. Additionally, administrators will have control over the catalog and users.

1.5 References

www.aman.ro
<https://app.diagrams.net/>
<https://github.com/shieldui/shieldui-lite/tree/master/qrcode/js>
https://www.w3schools.com/html/html_css.asp
<https://github.com/topics/library-management-system>
<https://github.com/DeveloperWide/HTML-CSS-Projects>
<https://www.geeksforgeeks.org/spring/>

2. Overall Description

2.1 Product Perspective

The system will be a distributed software accessible through a web application. It will allow users to perform specific actions based on their role, whether they are readers, librarians, or administrators.

2.2 Product Functions

The following functions make up the core of the web app:

1. Authentication and account management
2. Book search and filtering
3. Book reservation and borrowing
4. Automatic notifications
5. User and catalog management
6. Report and statistics generation

2.3 User Classes and Characteristics

There are three types of users that interact with the web app:

1. **Readers:** Search and borrow books, receive notifications about return deadlines.
2. **Librarians:** Manage the catalog, monitor loans and returns.
3. **Administrators:** Configure the system, create accounts, generate reports.

2.4 Operating Environment

The system will run on a PostgreSQL database server with the logic being handled by a Java administration program running under Windows 10 . The system will be accessible via the internet using any widely used browser platform and the device camera will be used to scan the QR code for the review and preview functionalities.

Any person who will use this application must have the following:

- Internet access 10.0.56
- Microsoft Windows 8 at least installed
- One of the next modern browsers: Google Chrome (version 133.0.6943.142), Microsoft Edge (version 134.0.3124.51)

2.5 Design and Implementation Constraints

Constraints:

IDE: IntelliJ Idea Ultimate 2024.2.4, Webstorm 2024.3.4

Back-end development: Java 23.0.1

Framework: Spring Boot 3.4.3

Database: PostgreSQL 17

Front-end development: HTML/ CSS, JavaScript

2.6 User Documentation

The Smart Library Management System (SLMS) is designed to be **intuitive and self-explanatory**. This word document represents the software requirements documentation for the project. This document will replace the user guide and other documentation.

2.7 Assumptions and Dependencies

It needs an internet connection and must be usable on various devices.

3. External Interface Requirements

3.1 User Interfaces

- Modern, intuitive, and responsive design
- Features accessible on desktop

Signup

Signup

First Name	Last Name
<input type="text" value="first name"/>	<input type="text" value="last name"/>
Date of Birth	Gender
<input type="text" value="date of birth"/>	<input type="text" value="gender"/>
Phone Number	Adress
<input type="text" value="phone number"/>	<input type="text" value="adress"/>
User Name	Email
<input type="text" value="user name"/>	<input type="text" value="email"/>
Password	Confirm Password
<input type="text" value="....."/>	<input type="text" value="....."/>

Data processing agreement

☐ By creating an account you agree with the processing of personal data

Signup

Already have an account? Login

Login page

Login

Email

Password

[Forgot Password?](#)

[Login](#)

[Don't have an account? Signup now](#)

Reader Menu

Personal Informations

Reader Profile

[Profile Details](#)[Reading Worksheet](#)[Delete Profile](#)

Utilities

[Update Account Informations](#)[Reset Password](#)[Physical Headquarters Schedule](#)[About Our Headquarters](#)[Book Search](#)[Most Borrowed Books](#)[New Book Reservation](#)[Notifications](#)

Librarian Menu

Personal Informations

Librarian Profile

[Profile Details](#)

Utilities

[Update Account Informations](#)[Reset Password](#)[Physical Headquarters Schedule](#)[About Our Headquarters](#)[Book Search](#)[Most Borrowed Books](#)[Books Evidence](#)[Readers Evidence](#)[Add Reader](#)[Remove Reader](#)[Unwanted Customers Table](#)[Notifications](#)

Administrator Menu

Personal Informations			Admin Profile	
<div>Profile Details</div>				
Utilities				
<div>Update Account Informations</div>	<div>Reset Password</div>	<div>Physical Headquarters Schedule</div>	<div>About Our Headquarters</div>	
<div>Add Reader</div>	<div>Remove Reader</div>	<div>Add Librarian</div>	<div>Remove Librarian</div>	
<div>Books Evidence</div>	<div>Readers Evidence</div>	<div>Librarians Evidence</div>	<div>Administrators Evidence</div>	
<div>Recovery Account</div>	<div>Book Search</div>	<div>Add New Book</div>	<div>Most Borrowed Books</div>	
<div>Unwanted Customers Table</div>	<div>Notifications</div>			

Profile Details

Profile Details	
First Name	<div>first name</div>
Last Name	<div>last name</div>
Date of Birth	<div>xx-xx-xxxx</div>
Gender	<div>gender</div>
Phone Number	<div>xxxx-xxx-xxx</div>
Adress	<div>adress</div>
User Name	<div>user name</div>
ID	<div>xxxxxx</div>
Email	<div>email</div>
Password	<div>xxxxxxxxxx</div>
<div>Exit</div>	

Update Profile Details

Update Profile Details

First Name	<input type="text" value="first name"/>
Last Name	<input type="text" value="last name"/>
Date of Birth	<input type="text" value="xx-xx-xxxx"/>
Gender	<input type="text" value="gender"/>
Phone Number	<input type="text" value="xxxx-xxx-xxx"/>
Adress	<input type="text" value="adress"/>
User Name	<input type="text" value="user name"/>
ID	<input type="text" value="xxxxxx"/>
Email	<input type="text" value="email"/>
Password	<input type="text" value="xxxxxxxxxx"/>

Reset Password

Reset Password

Enter the current password	<input type="text"/>
Enter the new password	<input type="text"/>
Confirm the new password	<input type="text"/>

Delete Profile

Delete Profile

After deleting your profile, you will not be able to log in anymore. All personal data will be deleted.
This is irreversible.

Enter password to confirm:

Profile recovery

Profile recovery

ID

xxxxxx

Recovery

Cancel

Reading Worksheet

Reader username:							Reader ID:			
Crt. No.	Title	Author	Publishing house	Price (lei)	Inventory Number	U.D.C Classification	Loan Date (day, month, year)	Return Date (day, month, year)	The branch where the loan was made	Book status
1										
2										
3										
4										
5										
6										
7										
8										
9										
10										

New Book Reservation

New Book Reservation

Title

Author

Loan Date (day, month, year)

Return Date (day, month, year)

The branch where you want to make the loan

Availability

Reserve

Exit

Add New Book

Add New Book

Title	<input type="text" value="title"/>
Author	<input type="text" value="author"/>
Publishing house	<input type="text" value="publishing house"/>
Price (lei)	<input type="text" value="price"/>
Inventory Number	<input type="text" value="inventory number"/>
U.D.C Classification	<input type="text" value="U.D.C classification"/>
The branch it's added to	<input type="text" value="branch"/>
Number of copies	<input type="text" value="number of copies"/>

Add

Cancel

Add Reader

Add Reader

First Name	<input type="text" value="first name"/>
Last Name	<input type="text" value="last name"/>
Date of Birth	<input type="text" value="xx-xx-xxxx"/>
Gender	<input type="text" value="gender"/>
Phone Number	<input type="text" value="xxxx-xxx-xxx"/>
Adress	<input type="text" value="adress"/>
User Name	<input type="text" value="user name"/>
ID	<input type="text" value="xxxxxx"/>
Email	<input type="text" value="email"/>
Password	<input type="text" value="....."/>

Add

Cancel

Remove Reader

Remove Reader

ID	<input type="text" value="xxxxxx"/>
----	-------------------------------------

Remove

Cancel

Add Librarian

Add Librarian

First Name

first name

Last Name

last name

Date of Birth

xx-xx-xxxx

Gender

gender

Phone Number

xxxx-xxx-xxx

Adress

address

User Name

user name

ID

xxxxxxx

Email

email

Password

.....

Add

Cancel

Remove Librarian

Remove Librarian

ID

xxxxxx

Remove

Cancel

Readers Evidence

[illegible]

Librarians Evidence

[illegible]

Administrators Evidence

[illegible]

Unwanted Readers

[illegible]

Books Evidence

[illegible]

Search Book

Search Book	
<input type="checkbox"/> Title	<input type="text" value="title"/>
<input type="checkbox"/> Author	<input type="text" value="author"/>
<input type="checkbox"/> Publishing house	<input type="text" value="publishing house"/>
<input type="checkbox"/> U.D.C Classification	<input type="text" value="U.D.C classification"/>
<input type="checkbox"/> Branch where it is available	<input type="text" value="branch"/>
<div> <input type="button" value="Search"/> <input type="button" value="Cancel"/> </div>	

Most Borrowed Books

Most borrowed books				
Crt. No.	Title	Author	Publishing house	U.D.C Classification
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				

Physical Headquarters Schedule

Schedule	
Monday	08:00 - 19:00
Tuesday	08:00 - 19:00
Wednesday	08:00 - 19:00
Thursday	08:00 - 19:00
Friday	08:00 - 19:00
Saturday	10:00 - 16:00
Sunday	Closed

About our headquarters

About Our Headquarters

ORGANIZATION

DOLJ COUNTY LIBRARY "ALEXANDRU ȘI ARISTIA AMAN"

str. M. Kogălniceanu nr.9, e-mail: btb@aman.ro, website:
http://www.aman.ro

<https://www.facebook.com/bibliotecajudeteanaaman>

ORGANIZATIONAL STRUCTURE

DIRECTOR/MANAGER

MANAGER: Name
PUBLIC RELATIONS SERVICE

Head of Service: Name
email: name.name@yahoo.com

ADULT BOOK LOAN DEPARTMENT
CHILDREN AND YOUTH BOOK LOAN DEPARTMENT
BRANCH DEPARTMENT:
 BRANCH NO. 1: Braza lui Novac Neighborhood, Block G8, ground floor;
 BRANCH NO. 2: Craovița Nouă Neighborhood, Orizont Complex, ground floor;
 BRANCH NO. 3: Valea Rosie Neighborhood, Block 52, room 1, ap. 1;
 BRANCH NO. 4: Ravine Neighborhood, Block E21, room 2, ap. 1.
GENERAL DEPOSIT AND READING ROOMS SERVICE:

Head of Service: Name
 BASIC COLLECTIONS DEPOSIT SECTION
 PERIODICAL PUBLICATIONS AND READING ROOMS SECTION
 MEDIA THEATRE/AMERICAN CORNER SECTION
 ROMANIAN EXILE INSTITUTE/ROMANIAN BOOK AND EXILE MUSEUM

DIRECTOR: Name
 ADMINISTRATIVE SERVICE

Head of Service: Name
 HUMAN RESOURCES SECTION
 LEGAL SECTION
 SECRETARIAT-ADMINISTRATIVE SECTION
 FINANCIAL ACCOUNTING - ACQUISITIONS SECTION
 BOOKBINDING SECTION
 ARCHIVE SECTION
 CHIEF ACCOUNTANT

Chief Accountant: Name
 RESEARCH, ART, LITERARY HISTORY SECTION

DIGITISATION AND PROJECTS

Head of Service: Name
 DOCUMENTATION DEPARTMENT
 INFORMATION DEPARTMENT
 DIGITISATION DEPARTMENT
 BIBLIOGRAPHICAL INFORMATION DEPARTMENT
 EXILE MUSEUM SERVICE

Head of Service: Name
 LIBRARY/CONSERVATION DEPARTMENT
 EXILE, ART, LITERARY HISTORY DEPARTMENT
 ELENA FARAGO MEMORIAL HOUSE
 WRITTEN TREASURES OF THE LOWER DANUBE

Notifications

Notifications	Received
Compose	<div style="border: 1px solid black; height: 20px;"></div>
Received	<div style="border: 1px solid black; height: 20px;"></div>
Unread	<div style="border: 1px solid black; height: 20px;"></div>
Sent	<div style="border: 1px solid black; height: 20px;"></div>
Exit	<div style="border: 1px solid black; height: 20px;"></div>
	<div style="border: 1px solid black; height: 20px;"></div>
	<div style="border: 1px solid black; height: 20px;"></div>

3.2 Hardware Interfaces

Not compatible with standard web servers./ Dedicated hosting server with appropriate processing capacity.

3.3 Software Interfaces

Interface with SQL databases.

REST API for integration with other systems.

3.4 Communications Interfaces

Notifications via email .

HTTP/HTTPS protocol.

4. System Features

4.1 Authentication and Account Management

4.1.1 Description and Priority

This functionality allows users (readers, librarians, local administrators , executive administrators) to log into the system using predefined credentials.

Priority: High – essential for the security and access to the system.

4.1.2 Stimulus/Response Sequences

The user enters the username and password in the designated fields and presses the login button.

The system verifies the credentials and grants access based on the user's role.

If the authentication is successful, the user is redirected to the page corresponding to their role.

If authentication fails, the system displays an error message.

4.1.3 Functional Requirements

FREQ-1: If the user enters valid credentials, the system grants access.

FREQ-2: If authentication fails, the system displays an error message.

FREQ-3: The system assigns a specific role to the user (Reader, Librarian, Administrator).

FREQ-4: The system logs each authentication attempt for monitoring purposes.

FREQ-5: Access to the application must be protected with a username and password, with differentiated access for different user roles (Administrator, Librarians).

FREQ-6: The system must allow password reset.

FREQ-7: Each user will have an account with different levels of access.

FREQ-8: Authentication and security: Users (readers and employees) must log into the system with a username and password, and personal data must be protected according to GDPR regulations.

4.2 Library Catalog

4.2.1 Description and Priority

This functionality allows users to search and filter books available in the library.

Priority: High – essential for users in locating desired resources.

4.2.2 Stimulus/Response Sequences

The user enters the title, author, or keywords in a search field.

The system returns a list of relevant results.

The user can access the details of each book.

4.2.3 Functional Requirements

FREQ-9: Each book has a unique number, title, authors, and the publisher where it was published.

FREQ-10: The system must allow searching for books by multiple criteria.

FREQ-11: The system will display the available stock for each branch.

FREQ-12: The system will allow adding multiple authors for a single book.

FREQ-13: The system will generate statistics regarding the most borrowed books.

FREQ-14: Users can see the availability of books.

FREQ-15: The system will suggest books similar to those already borrowed by the user.

FREQ-16: If the user performs a search, the system will display relevant results.

FREQ-17: The details of each book are accessible by clicking on the result.

4.3 Loan and Reservation Management

4.3.1 Description and Priority

This functionality allows users to borrow, reserve, and return books.

Priority: High – essential for managing the library's resources.

4.3.2 Stimulus/Response Sequences

The user selects an available book and requests a loan.

The system checks the user's status and the book's availability.

If the loan is approved, the book is registered as "borrowed."

When the loan period expires, the user receives a notification for the return.

4.3.3 Functional Requirements

FREQ-18: The system checks the user's status before approving the loan.

FREQ-19: Users can reserve unavailable books for later borrowing.

FREQ-20: Readers can borrow multiple books from different library branches for a limited period.

FREQ-21: The system records the date a book was borrowed and the date of its return.

FREQ-22: Users can view their borrowing history.

FREQ-23: The system notifies the user about the due date for returning books.

FREQ-24: Inter-branch loan: Readers can request a book from another branch, and the system will manage its transfer.

4.4 User Management

4.4.1 Description and Priority

This functionality allows administrators to manage user accounts.

Priority: High – essential for access control.

4.4.2 Stimulus/Response Sequences

The administrator accesses the user management section.

They can add, edit, or delete accounts.

4.4.3 Functional Requirements

FREQ-25: Administrators can create and manage user accounts, add and edit members.

FREQ-26: Users can update their own account information.

FREQ-27: Administrators can delete accounts of inactive users.

FREQ-28: The system must allow the recovery of accounts deleted accidentally.

FREQ-29: The system will allow setting a timeout period of inactivity after which users are automatically logged out.

4.5 Notifications and Alerts

4.5.1 Description and Priority

This functionality allows users to receive notifications about reservations, loans, and returns.

Priority: High – enhances the user experience.

4.5.2 Stimulus/Response Sequences

The system sends notifications via email for important deadlines.

4.5.3 Functional Requirements

FREQ-30: The system notifies users about loan expirations.

FREQ-31: Users receive notifications for completed reservations.

FREQ-32: Administrators receive alerts for out-of-stock books.

FREQ-33: Readers are notified about new books added to the library.

FREQ-34: The system allows users to personalize their notification preferences.

FREQ-35: Librarians are alerted about overdue returns.

4.6 Report Generation

4.6.1 Description and Priority

This functionality allows administrators to generate detailed reports on the library's activity.

Priority: Medium – helps in analyzing statistics.

4.6.2 Stimulus/Response Sequences

The administrator selects the type of report desired.

The system generates an exportable document.

4.6.3 Functional Requirements

FREQ-36: The system must allow exporting reports in PDF format.

FREQ-37: The system will generate monthly reports on the library's activity.

FREQ-38: Reports on the most borrowed books will be generated.

FREQ-39: Reports will include data on active and inactive users.

FREQ-40: Librarians will be able to view the borrowing history per user.

FREQ-41: Administrators will be able to filter reports by categories (books, users, loans).

FREQ-42: The system will allow generating a detailed report on overdue returns.

FREQ-43: The system will allow comparing statistics across different periods.

FREQ-44: Reports on newly registered users will be generated.

4.7 Library Resource Management

4.7.1 Description and Priority

This functionality allows librarians to monitor and manage both physical and digital resources of the library.

Priority: High – essential for organizing and optimizing the library's inventory.

4.7.2 Stimulus/Response Sequences

The librarian accesses the resource management section.

The system displays a list of all available resources.

The librarian can add, edit, or delete resource entries.

The system updates the inventory based on loans and returns.

4.7.3 Functional Requirements

FREQ-45: Librarians can add and update information about books.

FREQ-46: Librarians can mark certain books as "reference-only."

FREQ-47: The system will highlight damaged or lost books.

FREQ-48: Librarians can set rules for special loans (e.g., limiting the number of simultaneous loans).

FREQ-49: Admins can schedule maintenance sessions for library resources.

4.8 User Interface

4.8.1 Description and Priority

This functionality refers to how users interact with the system.

Priority: Medium – influences the user experience and system adoption.

4.8.2 Stimulus/Response Sequences

The user accesses the web platform.

The system displays a personalized dashboard based on the user's role.

The user navigates through intuitive menus to access the desired functionalities.

4.8.3 Functional Requirements

FREQ-50: The system will have a web interface accessible from desktop.

FREQ-51: The design of the interface will be optimized for accessibility.

FREQ-52: The system will offer contextual suggestions to users.

FREQ-53: The web application will have QR scanning functionality for quick searches.

FREQ-54: The system will allow direct feedback from the users.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The system must meet the following performance criteria under different conditions:

- The average response time for API calls should be under 3 seconds.
- The system should support at least 100 concurrent user sessions without degradation in performance.

5.2 Safety Requirements

Not applicable.

5.3 Security Requirements

User Authentication and Authorization: The security will be ensured by implementing role-based access to the data (three roles: readers, librarians and administrators). Every role has different permissions and privileges.

Data Encryption: All communications will be encrypted via HTTPS .

5.4 Software Quality Attributes

1. Maintainability – The code should be easy to update and modify with minimal effort, following best practices such as modular design, clear documentation, and adherence to coding standards to facilitate long-term maintenance.
2. Reusability – Components, functions, and modules should be designed to be reusable across different parts of the application or even in other projects, reducing development time and improving efficiency.
3. Extensibility – The software should be designed to accommodate future enhancements or modifications with minimal changes to the existing code, using design patterns such as Open/Closed Principle (OCP) and dependency injection.
4. Use of CamelCasing – All local variables and method arguments must be named using CamelCase to ensure readability and maintain a consistent coding convention.
5. Use of PascalCasing – Class names and method names should follow PascalCase formatting to clearly distinguish them from variables and method arguments.
6. Avoiding abbreviations – Variable names should be descriptive and avoid abbreviations, except for well-known acronyms such as "ID," to maintain clarity in the code.
7. Clear naming for boolean variables and methods – Boolean variables and methods should have meaningful names that clearly indicate their purpose, using prefixes such as is, has, or can (e.g., isActive, hasPermission).
8. Consistent naming conventions – All variables, functions, and classes should have clear and descriptive names that accurately reflect their purpose and functionality.
9. Modularity – Code should be well-structured by separating business logic from the user interface and breaking functionality into reusable modules to enhance maintainability and scalability.
10. Avoiding redundancy (DRY – Don't Repeat Yourself principle) – Code duplication should be minimized by using reusable functions and components, reducing complexity and improving maintainability.
11. Data validation – User inputs must always be validated to ensure security and prevent potential vulnerabilities.
12. Performance optimization – Unnecessary database calls should be avoided, and SQL queries must be optimized to improve application efficiency.
13. Code testability and error handling – The code should be written in a way that makes it easy to test, avoiding duplication and implementing proper error-handling mechanisms to ensure robust functionality.
14. Relevant and concise comments – Comments should clearly explain the logic of functions, being concise and useful for a quick understanding of the implementation.

15. Adherence to the "Single Responsibility" principle – Each class and method should have a well-defined single responsibility, following SOLID principles to ensure clarity, maintainability, and scalability of the code.

5.5 Business Rules

This software product is designed for the first implementation in Romania of a smart library.

6. Other Requirements

Not Available.

Appendix A: Glossary

- SRS - Software Requirements Specification: A detailed document that outlines the functional and non-functional requirements of a software system, serving as a foundation for development.
- UI - User Interface: The graphical layout of an application, including buttons, text fields, images, and other elements that users interact with.
- UX - User Experience: The overall experience a user has while interacting with a system, focusing on usability, accessibility, and satisfaction.
- QR - Quick Response: A type of two-dimensional barcode that stores information such as URLs, text, or contact details and can be scanned using a smartphone or QR scanner.
- SLMS - The Smart Library Management System: A software system designed to automate and optimize the management of library operations, providing an efficient way for clients to borrow books, track due dates and manage their accounts.
- UDC - Universal Decimal Classification: A classification scheme that uses Arabic numerals arranged decimally to represent a class.
- DB - Database: A structured collection of data that stores information related to readers, librarians and administrator.
- DBMS - Database Management System: Software that enables the creation, organization, and retrieval of data in a structured manner.
- RDBMS - Relational Database Management System: A type of DBMS that stores data in structured tables with relationships between them.
- API - Application Programming Interface: A set of functions and protocols that allow different software applications to communicate with each other.
- RBAC - Role-Based Access Control: A security model that grants user permissions based on their assigned roles (readers, librarians and administrator).
- PDF - Portable Document Format: A file format used to present and exchange documents in a way that is independent of software, hardware, or operating systems.
- UML - Unified Modeling Language: A standardized modeling language used to visualize, specify, construct, and document the components of a software system.

Appendix B: Analysis Models

In addition to this document, analysis models will be included to better understand the application. The UML models that will be included are use-case, activity diagram, class diagram, relational model for database.

Appendix C: To Be Determined List

Not Available.