1. **Project Phase 1**

Software Requirements Specification

Revision History

| **Date** | **Revision** | **Description** | **Author** |
| --- | --- | --- | --- |
| 09/11/2025 | 1.0 | Initial Version | Jason Serduke |
| 09/11/2025 | 1.1 | Rough scope, overview, architecture, features | All |
| 09/21/2025 | 1.2 | Rough constraints, assumptions, and module-specific requirements | Jason Serduke |
| 09/20/2025 | 1.3 | Constraints | Thanh Nguyen |
| 09/21/2025 | 1.4 | Security and Privacy Requirements | Thanh Nguyen |
| 09/22/2025 | 1.5 | Environmental Requirements, Assumptions | Danica Pespitro |
| 09/23/2025 | 1.3 | Use Case Specifications | Thanh Nguyen |
| 09/23/2025 | 1.7 | 1.2 - Definitions, 2.3 - Product Functionality, 3.3 - Internal Interface Reqs | Larry Hor |
| 09/23/2025 | 1.8 | 2.1 - Product Perspective, 4.3 - Performance Requirements | Sean Walker |
| 09/23/2025 | 1.9 | Sequence Diagrams | Thanh Nguyen |
| 09/23/2025 | 1.10 | UML Use Case Diagram | Danica Pespitro |
| 09/26/2025 | 1.11 | UML Class Diagram | Jason Serduke |
| 10/01/2025 | 1.12 | Added references | Jason Serduke |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

**1.** **Purpose 4**

1.1. Scope 4

1.2. Definitions, Acronyms, Abbreviations 4

1.3. References 4

1.4. Overview 4

**2.** **Overall Description 5**

2.1. Product Perspective 5

2.2. Product Architecture 5

2.3. Product Functionality/Features 5

2.4. Constraints 5

2.5. Assumptions and Dependencies 5

**3.** **Specific Requirements 6**

3.1. Functional Requirements 6

3.2. External Interface Requirements 6

3.3. Internal Interface Requirements 7

**4.** **Non-Functional Requirements 8**

4.1. Security and Privacy Requirements 8

4.2. Environmental Requirements 8

4.3. Performance Requirements 8

# Purpose

This document outlines the requirements for a public library platform.

## Scope

This document will catalog the user, system, and hardware requirements for the library system. It will not, however, document how these requirements will be implemented.

## Definitions, Acronyms, Abbreviations

MSRP - Manufacturer’s Suggested Retail Price

ISBN - International Standard Book Number

Dewey Decimal System - Library classification system comprising 10 classes separated into 10 divisions with 10 sections.

Call Number - A book’s associated Dewey Decimal categorization system, ie. 100-199 = Philosophy and Psychology

CSV File - “Comma-separated value” file extension for managing databases.

ID - Internal Unique ID used for Library material to reference internally

PIN - Personal Identification Number

GUI - Graphical user interface

RBAC - Role based access control

PII - Personal Identifying Information

## References

Use Case Specification Document – See UseCaseSpecification.docx

UML Use Case Diagrams Document – See UML Use Case Diagram.jpg

Class Diagrams – See LibrarySystemClassesUML.png

Sequence Diagrams – See SequenceDiagrams.docx

## Overview

The public library platform is designed to facilitate the management of library services such as the inventory of physical media, patron self-service, the status of checked out materials, and calculation of fees from interfaces for both members and administrators. Automating these processes will free up more time for librarians to use their skills for more meaningful interactions and tasks.

# Overall Description

## Product Perspective

This public library platform is a client-server application that will support members and staff across multiple libraries to track the loan and return of library materials. Members will be able to log in, search, and check out and return media items such as books, magazines, and DVDs. The server will store and manage all information while the client provides a user interface.

## Product Architecture

The system will be organized into five major modules: the media information module, the accounts module, the searching module, the history/records module, and the fees module.

## Product Functionality/Features

The high-level features of the system are as follows (see section 3 of this document for more detailed requirements that address these features):

* Store library inventory data.
* Search and browse inventory.
* Add and remove library items from inventory.
* Check out and return items.
* Calculate late fees.
* Support multiple users at the same time.
* Graphical user interface.
* Support user and admin account with permission levels.
* Maintain checkout and return history.

## Constraints

2.4.1 The system must be implemented in Java. This applies to all server and application logic.

2.4.2 We can’t use a database; all data must be stored in server side text files (CSV/TXT). Schemas must stay consistent across files.

2.4.3 No browser specific features are allowed, so the client must work on any standards compliant browser.

2.4.4 The system is expected to run only when internet access is available; design features assuming an online environment.

2.4.5 We plan for up to roughly ten simultaneous users; file I/O must avoid corruption under light concurrency. Use safe read/write patterns.

2.4.6 Target machines will share the same OS image; do not require additional software being a browser for users.

## Assumptions and Dependencies

2.5.1 It is assumed that the system will only be capable of functioning when there is access to the Internet.

2.5.2 It is assumed that the maximum number of users at any given time will be ten.

2.5.3 It is assumed that the bandwidth of the network connection will be sufficient to handle up to ten users.

2.5.4 It is assumed that the machines that the library system software will be installed on will all be running the same operating system.

2.5.5 It is assumed that the computers will not have any other software running on them at the same time as the library platform.

2.5.6 It is assumed that all users will access the system through a standards-complaint web browser.

2.5.7 It is assumed that CSV/TXT files will remain intact and not be altered outside of the system

# Specific Requirements

## Functional Requirements.

### Common Requirements:

3.1.1.1 All users (staff and members) must log in with username and password before performing any actions involved with their account.

3.1.1.2 The system must give clear error messages if users enter wrong login details.

3.1.1.3 Every action done by a user must be checked to make sure they have the right permissions.

3.1.1.4 The system must always save data safely so information is not lost when multiple users are working.

3.1.1.5 Users must be able to log out at any time, and their session will end right away.

3.1.1.6 The system must be simple to use with clear buttons and text so members and staff can find what they need.

### Media Information Module Requirements:

3.1.2.1 Each title in the library should be assigned an internal ID that will be unique among all other media.

3.1.2.2 The information for any type of media that is stored should include its title, publisher, genre, MSRP, total quantity, and quantity available.

3.1.2.3 In addition to the information for all media, the author, ISBN, and Dewey Decimal call number should be stored for books.

3.1.2.4 In addition to the information for all media, the rating and total duration should be listed for all DVDs.

3.1.2.5 Information for books should be stored in a CSV file on the server side, with the fields in the following order:  
internal ID, title, author, publisher, genre, Dewey Decimal call number, ISBN, total quantity, quantity available

3.1.2.6 Information for DVDs should be stored in a CSV file on the server side, with the fields in the following order:  
internal ID, title, publisher, genre, rating, duration, MSRP, total quantity, quantity available

3.1.2.7 Book information will need to be able to be accessed and modified from other modules.

### Accounts Module Requirements:

3.1.3.1 Library patrons should be able to register for a new library account with their full name, birthday, and street address (PII). Next, they must enter an email address that will also function as their username and then finally choose a password of at least seven alphanumeric characters.

3.1.3.2 Upon registration of a new member, they should be assigned a unique ID number for internal purposes.

3.1.3.3 Library members with an account should be able to log in to a management portal with their email address and password.

3.1.3.4 While logged in, and having less than five materials already checked out, library members should be able to check out a book by entering its author and title or ISBN.

3.1.3.5 While logged in, library members should be able to see a list of media that they have checked out, when they checked out each one, when each one is due, and how many days from the current day until each one is due.

3.1.3.6 While logged in, library members should be able to return a piece of media by clicking the corresponding button in their list of checked out works.

3.1.3.7 A librarian or volunteer should be able to log in to a management portal with their email address and password.

3.1.3.8 When logged in, a librarian should be able to enter the email address of a new librarian or volunteer so that they may register their account on the library platform.

3.1.3.9 Librarians and volunteers should be able to register for a new library account with their full name, birthday, and street address. Next, they must enter the approved email address and finally choose a password of at least seven alphanumeric characters.

3.1.3.10 When logged into their portal, librarians should be able to search for and view the information of any registered patron. From there, they should be able to view the member’s information and their checked out materials with the associated status information.

3.1.3.11 When logged in, a member should be able to view an invoice for any accrued fees, with information about each charge including which material it relates to and why they have received it (e.g., late or lost).

3.1.3.12 When logged in, a librarian or volunteer should be able to add new supplies of a current material to the library’s inventory by entering its internal ID and then choosing how many to add.

3.1.3.13 When logged in, a librarian or volunteer should be able to add new supplies of a new material to the library’s inventory by entering all of its information and then choosing how many to add.

### Searching Module Requirements:

3.1.4.1 Library patrons (registered or not), should be able to search for materials in the library by specifying the title, author, ISBN, or publisher.

3.1.4.2 Upon making a search, a list of results should be displayed which correspond to the provided search term(s).

3.1.4.3 Upon clicking on a particular search result, the stored data for that search result should be displayed along with whether it is currently available or not.

### History/Records Module

3.1.5.1 All of the permanent additions and removals of the library’s inventory should be recorded in a CSV file.

3.1.5.2 All checkouts of library materials should be recorded in a CSV file with the internal ID and the date that it was checked out.

3.1.5.3 All returns of library materials should be recorded in a CSV file with the internal ID, the date that it was checked out, and the date that it was returned. The corresponding record that was recorded during checkout should be deleted.

### Fees Module

3.1.6.1 A late fee should be calculated by multiplying the number of days late for an item by the fee accrued per day.

3.1.6.2 A lost fee should be equal to the MSRP of the lost item.

3.1.6.3 The total late fees should be the sum of the late fees for each late item.

3.1.6.4 The total lost fees should be equal to the lost fees for each lost item.

3.1.6.5 The total amount of fees should be equal to the sum of the total late fees and the total lost fees.

## External Interface Requirements

3.2.1 The system must provide an interface for exporting reports of checkout out and returned books. These reports will be saved in csv files for use by staff.

3.2.2 The system must allow members to receive email notifications for due dates, late fees, and account changes.

3.2.3 The system must support a barcode scanner so staff can check out and return items quickly.

3.2.4 The system must provide simple file export so administrators can back up user and inventory data.

## Internal Interface Requirements

3.3.1 The system will keep track of all associated library inventory that can be accessed by other libraries in the system.

3.3.2 The system will be able to generate a report of all inventories of the associated libraries and their current status ie. (checked out, reserved, missing)

3.3.3 The system will have a GUI for users to interact with the system for searching, checking out media, etc.

3.3.4 Libraries will be able to access, read, edit, and generate .csv files detailing inventories of selected libraries.

3.3.5 Critical library inter-calls must be under 100ms response time.

3.3.6 Shared resource precautions best practices followed.

3.3.7 The catalog database (.csv) must provide bibliographic records (title, author, ISBN, availability) to the catalog system

3.3.8 If a patron is blocked, circulation functions must return an error code preventing checkout

# Non-Functional Requirements

## Security and Privacy Requirements

4.1.1 All information sent between the user and the system must be protected so no one can read it while it is moving the network.

4.1.2 Users must log in with an email and password. Passwords must be at least seven characters long and never stored in plain text.

4.1.3 The system must use roles(Member and Admin) so that only the right people can do certain tasks. Each user should have the permissions they need.

4.1.4 The system must manage sessions safely. This means using secure session IDs, logging users out after time of no activity, and ending the session right away when they log out.

4.1.5 The system must check all information entered by users, like forms and search boxes, to make sure it is safe and cannot be used for attacks.

4.1.6 All files with library data must be stored in safe folders on the server. These files must have restricted permissions and be saved in a way that avoids corruption.

## Environmental Requirements

4.2.1 Administrators and members will be able to access the system directly through a web browser without the need to download or install other applications.

4.2.2 The platform will run on the library’s existing server and will not require additional hardware or third-party software.

4.2.3 The system will function only when there is an active internet connection since features such as searching, checkouts, and account access requires connectivity.

4.2.4 All system data will be stored in server as CSV/TXT files and file formats will remain consistent across all modules

4.2.5 The platform will operate across multiple operating systems, including Windows, macOS, Linux, with a responsive interface for desktops.

4.2.6 The system will support up to multiple users at the same time without risking data corruption by using safe file read and write methods.

## Performance Requirements

4.3.1 The system should respond to staff and member inquiries such as searching, logging in/out and checking in/out without delay.

4.3.2 The platform must be able to handle multiple concurrent users at the same time.

4.3.3 In cases of system errors or crashes, the platform should have error handling to handle common errors and be able to recover from a crash or restart.

4.3.4 The system should remain operational even if an internet connection is not available and connection to other libraries is lost.

4.3.5 The client will attempt to reconnect if the TCP connection drops.