Bookstore Requirements Specification

Version 2.5

April 11, 2023

**Table of Contents**

[**BOOKSTORE REQUIREMENTS SPECIFICATION 1**](#_heading=h.19c6y18)

[**VERSION 1.0 1**](#_heading=h.3tbugp1)

[**APRIL 10, 2023**](#_heading=h.28h4qwu)

[**1.**](#_heading=h.nmf14n) **EXECUTIVE SUMMARY 3**

[1.1](#_heading=h.37m2jsg) Project Overview 3

[1.2](#_heading=h.1mrcu09) Purpose and Scope of this Specification 3

[**2.**](#_heading=h.46r0co2) **PRODUCT/SERVICE DESCRIPTION 3**

[2.1](#_heading=h.2lwamvv) Product Context 3

[2.2](#_heading=h.111kx3o) User Characteristics 3

[2.3](#_heading=h.3l18frh) Assumptions 3

[2.4](#_heading=h.206ipza) Constraints 3

[2.5](#_heading=h.4k668n3) Dependencies 4

[**3.**](#_heading=h.2zbgiuw) **REQUIREMENTS 4**

[3.1](#_heading=h.1egqt2p) Functional Requirements 5

[3.2](#_heading=h.3ygebqi) Non-Functional Requirements 5

[*3.2.1*](#_heading=h.2dlolyb) *Product Requirements 5*

[**3.2.1.1**](#_heading=h.44sinio) **User Interface Requirements** 6

[**3.2.1.2**](#_heading=h.2jxsxqh) **Usability** 6

[**3.2.1.3**](#_heading=h.z337ya) **Efficiency** 6

[3.2.1.3.1](#_heading=h.3j2qqm3) Performance Requirements 6

[3.2.1.3.2](#_heading=h.1y810tw) Space Requirements 6

[**3.2.1.4**](#_heading=h.4i7ojhp) **Dependability** 6

[**3.2.1.5**](#_heading=h.2xcytpi) **Security** 7

[*3.2.2*](#_heading=h.sqyw64) *Organizational Requirements 7*

[**3.2.2.1**](#_heading=h.3whwml4) **Environmental Requirements** 7

[**3.2.2.2**](#_heading=h.2bn6wsx) **Operational Requirements** 7

[**3.2.2.3**](#_heading=h.qsh70q) **Development Requirements** 7

[*3.2.3*](#_heading=h.3cqmetx) *External Requirements 7*

[**3.2.3.1**](#_heading=h.1pxezwc) **Regulatory Requirements** 7

[**3.2.3.2**](#_heading=h.49x2ik5) **Ethical Requirements** 7

[**3.2.3.3**](#_heading=h.2p2csry) **Legislative Requirements** 7

[3.2.3.3.1](#_heading=h.147n2zr) Accounting Requirements 7

[3.2.3.3.2](#_heading=h.3o7alnk) Security Requirements 7

[3.3](#_heading=h.1rvwp1q) Domain Requirements 7

[**4.**](#_heading=h.4bvk7pj) **USER SCENARIOS/USE CASES 7**

# Executive Summary

## Project Overview

The focus of this software will be to manage all important steps of a bookstore. The software can keep data about all the books in the bookstore, such us ISBN of the book, title, category of the book, selling price, author, stock. At the same time, it keeps track of the sold products by bill number, their sold quantities, prices, and date of transaction.

The application will have a three-level user system: Librarian, Manager and Administrator. Each will have different views and usage of the software. Each of the users has a username and a password to enter in the software.

**Librarian** -> Has the right to check out books that a customer may need from the bookstore. The librarian creates a bill and enters the data of the bought books, such as ISBN of the book and its quantity. If the book is out of stock or does not exist, the librarian is alerted. The software provides them the total amount of the bill, which is in printable format. The librarian can also take book requests from the store’s clients.

**Manager** ->The manager has the right to supply the bookstore with the needed books. So, they can enter in the stock the new book category, and/or add books of the same category to the stock of the bookstore. The manager is also informed if there are few items of a book in the bookstore stock. The manager can also check the performance of the librarians by checking their total number of bills, books sold, and the total amount of money made for a certain date or between a certain period of time.

**Administrator** ->The administrator has the right to manage almost everything that librarian and manager does. Beside them they have the right to manage the employees (librarian and manager), by registering, modifying, and deleting them. The administrator keeps info about employees such as: name, birthday, phone, email, salary, access level. The software also provides statistics about books sold the same way it does for managers.

## Purpose and Scope of this Specification

The purpose of this software specification is to define the scope of the project and clarify which requirements will be addressed in the development of the software. The scope of this specification includes defining the features and functionalities of the software, as well as outlining the three-level user system (Librarian, Manager, and Administrator) and the specific tasks that each user is able to perform.

In Scope:

* Definition of the functional and non-functional requirements for the Bookstore Management Software
* Identification of the user roles and associated permissions for the software
* Definition of user scenarios and use cases to illustrate the software's intended usage
* Documentation of regulatory, ethical, and legislative requirements related to the software
* Identification of any constraints or dependencies that may affect the development or implementation of the software
* Identification of assumptions made during the requirements gathering process
* Identification of any organizational, environmental, operational, or development requirements that may affect the software's implementation or use

Out of Scope:

* Detailed technical specifications or implementation details for the software
* User interface design or graphical elements of the software
* Hardware or infrastructure requirements for the software
* Marketing or branding strategies for the software
* Customization or configuration of the software for specific clients or use cases

# Product/Service Description

The main focus of this software is to help librarians and manager exchange data in order to keep the transactions and stocks organized and transparent at all times. The manager will first supply the bookstore with new books and enter their information in the bookstore stock organized by their genres. Afterward, the librarian can search for a certain book title and check its stock status and then perform the sales transactions.

The librarian will be easily informed if the book is out of stock or not found at all. Simultaneously the manager will be alerted of what books are running out of stock and supply the bookstore with the needed books. Based on how many books are sold overall by each librarian, the manager can determine their performance and check the most demanded books by the public and what categories are preferred.

## Product Context

The Bookstore Management Software is a mostly independent and self-contained product designed to manage the operations of a bookstore. It does not interface with other systems or products, and is designed to be a standalone solution for bookstores.

This software is a helpful tool for every bookstore to keep track of their transactions, measure employee productivity, form a revenue strategy and organize the overall stock efficiently. Our application only focuses on this institution as a for-profit business, meaning it will only be available to the employed staff of the bookstore, not open to the public.

Without disrupting the app’s effectiveness, its priority is to be easily operable by every employee regardless of their qualifications or knowledge of computer use. It also provides great data backup and organization, avoiding the use of on-paper documentation of numerous book ISBN (book serial numbers). This way data is reached faster and easier in case of a transaction.

To store this data, the app is operating on cloud storage, thought as the best option appropriate to the size of this business. This storage type assures cost-effectiveness, accessibility for authorized personnel, security and protection of data, and automatic backup/recovery of data.

## User Characteristics

Librarian:

* Checks out information about a book that a customer is interested in
* Creates a bill with the data of the book of interest
* Checks for book availability
* Gets notified if the book is out of stock or does not exist
* Prints the bill for the customer if the book is in stock
* Bills are stored to provide statistics regarding books sold whenever requested by manager
* Basic computer skills required to navigate GUI and print bills

Manager:

* Supplies the bookstore with the needed books
* Enters book information to the bookstore stock
* Categorizes books and groups them together accordingly
* Gets notified when there are fewer than 5 available copies per book in bookstore stock
* Checks librarian’s performance based on the:

1. Number of books sold
2. Amount of money made

All are reflected in the created bills over time

* Basic computer skills required

Administrator:

* Can manage everything that the employees can
* Manages the employees by registering, modifying, and deleting them
* Keeps data about the aforementioned such as name, birthday, phone number, salary, access level, etc.
* Receives data about the total income, meaning the total of books sold and the total cost, being the total of items bought and staff salaries in a given period
* Revoke’s permission access from employees
* Basic computer skills required

## Assumptions

* It is assumed that this software will be available only to the employed staff
* In order to use the app, it is crucial to have a good internet connection at all times
* It is assumed that the staff members understand English and know how a computer works
* It is expected that the computers have an operating system that supports Java
* The employees have to be equipped with the essential knowledge on how to use the app and, if required, update its data
* A user browses and/or validates, depending on his or her permissions
* The administrator will presumably have unrestricted access to all the relevant data of this bookstore
* User maintenance: adds, updates, and deletes system users and their associated information
* Book maintenance: adds, updates, and deletes books that will be used by the organization using the system

## Constraints

The purpose of constraints is to help prevent the bookstore app from becoming complex and confusing.

* Resource constraints are given by the relation between the demands for processing the data and the capacity of the computing machinery. The resources are runtime, memory, facilities, communication and energy. Hence, the system is optimized with regard to minimal resource consumption.
* The system is strictly used by the authorized individuals (administrator, manager and librarian). The software will need to have access control mechanisms to ensure that only authorized personnel can access and modify sensitive data. Management of user accounts and privileges will also need to be included. Additionally, the software will need to be designed with security best practices in mind to ensure that customer data is protected.
* The software will be a critical component of the bookstore's operations. Any downtime or issues with the software could result in lost sales and revenue. As such, the software will need to be designed with high availability and reliability in mind.
* The users must have an active internet connection in order for the app to work properly.

## Dependencies

* Integration with the store's existing database system to ensure seamless data transfer and avoid any discrepancies or data duplication.
* Availability of reliable and stable internet connection to ensure cloud storage is always accessible and up-to-date.
* Compatibility with the store's existing hardware such as computer devices or printers, that may be required to use the software.
* Completion of any necessary training or educational programs for the store employees to learn how to use the software effectively.
* Timely delivery of hardware, software, and other components required to implement the software in a timely and efficient manner.
* Compliance with any regulatory or legal requirements, such as data protection laws and regulations, and software licensing agreements.
* The software requires Java and JavaFX to be installed on the user's system in order to run properly.

# Requirements

* Describe all system requirements in enough detail for designers to design a system satisfying the requirements and testers to verify that the system satisfies requirements.
* Organize these requirements in a way that works best for your project. See Appendix DAppendix D, Organizing the Requirements for different ways to organize these requirements.
* Describe every input into the system, every output from the system, and every function performed by the system in response to an input or in support of an output. (Specify what functions are to be performed on what data to produce what results at what location for whom.)
* Each requirement should be numbered (or uniquely identifiable) and prioritized.

See the sample requirements in Functional Requirements, and System Interface/Integration, as well as these example priority definitions:

**Priority Definitions**

The following definitions are intended as a guideline to prioritize requirements.

* Priority 1 – The requirement is a “must have” as outlined by policy/law
* Priority 2 – The requirement is needed for improved processing, and the fulfillment of the requirement will create immediate benefits
* Priority 3 – The requirement is a “nice to have” which may include new functionality

It may be helpful to phrase the requirement in terms of its priority, e.g., "The value of the employee status sent to DIS **must be** either A or I" or "It **would be nice** if the application warned the user that the expiration date was 3 business days away". Another approach would be to group requirements by priority category.

* A good requirement is:
* Correct
* Unambiguous (all statements have exactly one interpretation)
* Complete (where TBDs are absolutely necessary, document why the information is unknown, who is responsible for resolution, and the deadline)
* Consistent
* Ranked for importance and/or stability
* Verifiable (avoid soft descriptions like “works well”, “is user friendly”; use concrete terms and specify measurable quantities)
* Modifiable (evolve the Requirements Specification only via a formal change process, preserving a complete audit trail of changes)
* Does not specify any particular design
* Traceable (cross-reference with source documents and spawned documents).

## Functional Requirements

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Req# | Requirement | Comments | Priority | Data reviewed | SME reviewed/  approved |
| R\_01 | Users should be able to log in with a username and password. | This ensure secure access to the system. | 1 | 13.05.2023 | Elisa Gjuraj  Nevina Dervishi |
| R\_02 | The app will offer different views for different user level(librarian,manager and administrator). | Each user will have a unique login with corresponding access level ,the app will provide different views. | 1 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_03 | The app shall allow librarians to check out books for customers . | To fulfill this requirement, the app should provide a checkout feature where librarians can select the books requested by customers and complete the checkout process. | 1 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_04 | The app shall allow librarians to create bills with book details and prices. | To fulfill this requirement, the app should provide a user-friendly interface where librarians can input the necessary book information, such as ISBN, title, author, and quantity. The system should then calculate the total price based on the entered book prices and quantitie | 1 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_05 | The app should alert the librarian when books are out of stock or unavailable for purchase. | Timely alerts help the librarian manage stock and inform customers about book availability. | 2 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_06 | The app shall provide a total amount for bills that is in printable format for librarians. | The bill should include all relevant information such as book details, prices and total amount due | 3 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_07 | The app shall allow librarians to take book requests from customers. | The librarian should be able to record the book request and contact the customer when the book becomes available. | 2 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_08 | The app shall allow managers to add ,update and delete book categories in the system. | Proper categorization helps organize books and improve search capabilities. | 1 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_09 | The manager should be able to update the stock quantity of books in the bookstore. | Accurate stock management ensures timely replenishment and prevents overselling or stockouts. | 1 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_10 | The app shall alert managers when the stock of a book is low. | The alert should be triggered when the stock of a book falls below a specified threshold. | 2 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_11 | The app shall allow managers to check the performance of librarians by checking their total number of bills, book sold, and total amount of money made for a certain date or between a certain period of time. | The manager should be able to filter the performance data by data or time period. | 3 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_12 | The app shall allow administrator to register,modify,and delete librarian and manager accounts. | Efficient employee management allows for proper role assignment and maintenance of employee records. | 1 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_13 | The app shall allow the administrator to store employee information such as name, birthday,  phone, email, salary, and access level. | The administrator should be able to view and edit the employee information at any time. | 1 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_14 | The app shall provide statistics about books sold for administrators in the same way it does for managers. | The administrators should be able to view book sales data by date or time period. | 3 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_15 | The app shall allow administrator to generate sales report for certain date or between a certain period of time. | The administrator should be able to generate sales reports for specific dates or periods of time, providing valuable information for decision-making purposes. | 2 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_16 | The app shall provide the ability for librarians and managers a search function to find books based on ISBN, title, category or author. | This requirements will provide librarians and managers the ability to search for books based on various criteria, helping them to locate books quickly and efficiently. | 1 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_17 | The app shall keep track of the total stock of each book and alter managers when the stock of a book falls below a certain threshold | This requirement will ensure that managers are alerted when the stock of a book falls below a certain threshold, allowing them to restock before running out of stock**.** | 2 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_18 | The app shall allow managers to update book details such as title, author, category and price. | The managers will be able to update book details such as title, author, category and price ensuring that the data is accurate and up-to-date. | 2 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_19 | The app shall allow administrator to set access level for each employee. | The administrator can set appropriate levels for each employee ,ensuring that employees only have access to the necessary features and data. | 1 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_20 | The app should generate reports on various aspects, such as sales, inventory and employee performance ,in printable formats. | Reports assist in monitoring bookstore operations and evaluating performance. | 2 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_21 | The app should allow the librarian to register and update customer information, including name, contact details, and purchase history. | The requirement emphasizes the importance of maintaining a purchase history for each customer. This means that the system should record and track the books that each customer has purchased in the past. This purchase history data can be useful for several purposes, including personalized services, such as recommending books based on the customer's preferences and purchase history. | 2 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_22 | The app should provide personalized book recommendations to customers based on their purchase history and preferences. | This requirements allows books to be recommended based on the customer's purchase history | 2 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_23 | The app should track sales by recording the bill number, sold quantities, prices and transaction dates. | This requirement enables accurate sales tracking is essential for effective inventory management, enabling bookstores to monitor stock levels, identify popular books, and make informed purchasing decisions. | 1 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_24 | The app shall allow managers to create and manage purchase orders for restocking books. | This requirement can be fulfilled by providing the manager with a user-friendly interface with the system. The system should allow the manager to generate and send purchase orders to suppliers electronically. | 2 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_25 | The app should store supplier information, including contact details and payment terms. | This requirement ensure effective supplier management by providing a centralized database for storing supplier information. | 2 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_26 | The system should send notifications and reminders to librarians managers, and administrators for pending tasks and low stock. | This requirement plays a vital role in keeping stakeholders informed and facilitating timely actions. By sending notifications and reminders, the system ensures that librarians, managers, and administrators stay updated on pending tasks. Additionally, notifications about low stock. | 2 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_27 | The system should allow the manager to create and manage discount,  promotions and special offers for books. | This requirement empowers the manager to effectively implement marketing strategies by offering discounts and promotions on selected books. | 2 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_28 | The system should provide an online platform for customers to place book orders and make payments. | This requirement aims to offer customers the convenience of ordering books online. By providing a user-friendly online platform, the system allows customers to browse and select books, place orders, and securely complete payments. | 1 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_29 | The app should enable customers to track the status of their online book orders. | This requirement focuses on providing customers with updates on their orders. By including order tracking, the system lets customers know where their books are during the shipping process. This increases customer satisfaction, reduces support inquiries, and builds trust. | 2 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_30 | The app should handle returns and refunds ,including tracking return requests, processing refunds and updating inventory accordingly. | This requirement emphasizes the importance of managing returns and refunds effectively. By having a system in place to handle customer returns, process refunds, and update inventory, the bookstore can provide a seamless experience for customers who need to return or exchange books. | 2 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_31 | This app shall allow customers to make payments using their credit or debit cards. | This requirements enables customers to complete transactions seamlessly, expanding their payment choices and enhancing overall user experience. | 1 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_32 | The app should enable customers to reserve books for specific duration before making a purchase. | This requirements can provide a feature where customers can select the desired book and choose a reservation duration . | 2 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_33 | The app should allow customers to provide reviews and ratings for books they have purchased. | The app should include a feature where customers can provide their feedback, write reviews, and rate the books they have purchased. | 2 | 13.05.2023 | E.Gjuraj  N.Dervishi |
| R\_34 | The app should provide a feature to identify and display the bestseller book(s). | This requirements enables book recommendations for librarians assist in book ordering and stocking decision. | 2 | 13.05.2023 | E.Gjuraj  N.Dervishi |

## Non-Functional Requirements

**In here try to use the Structure given at slide 13 in Requirements Engineering Lecture Slides, with main categories of:**

### Product Requirements

#### **User Interface Requirements**

1. User Roles and Access Levels:

* The login screen should provide a secure authentication mechanism to verify the user's identity.
* The system should support three user roles: Librarian, Manager, and Administrator, each with their respective screens depending on their access level and permissions.
* The interface should clearly indicate the role and access level of the currently logged-in user.

1. Dashboard and Navigation:

* The main dashboard should provide an overview of relevant information depending on the logged-in user’s access level, such as sales statistics, inventory status, and employee performance.
* The navigation menu should be intuitive and organized, allowing users to easily access different functions and features.
* The interface should maintain consistency in layout and design across different screens to ensure a cohesive user experience.

1. Book Management:

* The interface should include a book list with the ability to sort based on various criteria, such as title, author, ISBN, or category.
* The system should display comprehensive book details, including title, author, publication date, price, quantity in stock, and any additional attributes.
* Managers should be able to add new books to the inventory, update existing book information, and remove books from the system.

1. Sales and Checkout:

* The interface should enable the Librarian to process book checkouts efficiently.
* The system should automatically update the inventory upon successful book checkout, reflecting the reduced stock quantity.
* A printable receipt or bill should be generated for each transaction, containing details such as the book(s) purchased and total cost.

1. Employee Performance Monitoring:

* The Manager should have access to an interface that displays employee performance metrics, such as number of transactions and total money made.
* The system should generate reports or visualizations to help evaluate employee performance over time.

1. Reporting and Statistics:

* The interface should allow users to set specific date ranges for generating reports.

1. Error Handling and Messages:

* The system should provide informative and user-friendly error messages when users encounter issues, such as incorrect login credentials or data validation errors.
* Warning messages should be displayed when critical actions are about to be performed, ensuring that users are aware of potential consequences.

#### **Usability**

* Accessibility

The software will be designed to be accessible only locally through a personal computer device at the work site.

* Responsiveness

The software will be designed to be highly responsive and any changes made by the user will be updated in real time and saved into the database appropriately.

* Flexibility

The software will be designed in a way that future updates and changes by the developers at the request of the client/s can be implemented with ease and low risk of major errors.

* Effectiveness

The software will be designed in a way that allows users to complete their tasks seamlessly and effortlessly in a timely manner while also limiting the amount of errors that may hinder the software's efficiency and efficacy.

#### **Efficiency**

##### Performance Requirements

The software will be application based and accessible locally on a personal computer device at the work site for all users and all user levels. Even though application will only be accessible locally, internet connection is required to ensure the function the central database.

* + - * 1. ***Responsiveness***

In the event of external interrupts, the application should save the state it was in prior to said interrupts and return to it after they have been handled in order to maintain the software's integrity and to avoid compromising any crucial data.

* + - * 1. ***Errors***

In the event that users will face errors while operating the application, contingencies will be put in place in order to handle said errors and maintain the integrity of the software and its data.

#### **Dependability**

**Availability**

The software must ensure availability to its users at all times.

The software must function seamlessly without frequent hindrance or interruption for the convenience of users.

It is vital to remember that there could be limitations in accessing the content offline and some features may need an internet connection

**Integrity**

The software should ensure the integrity of the stored data, including book information, sales transactions, and user details.

Measures should be in place to prevent data corruption, unauthorized access, and ensure data consistency and accuracy.

**User Satisfaction**

Ensure that the software is designed with a user-centric approach, focusing on providing a seamless and intuitive user experience.

**Error Handling**

Robust error handling mechanisms should be in place to handle unexpected scenarios, prevent critical failures, and maintain the dependability of the system.

**Internet Connection Dependency**

The application relies on an internet connection for its functioning.

This implies that the availability and performance of the software may be affected by the quality and stability of the internet connection.

**Payment Transaction Dependency**

Certain functionalities, such as the checkout process or online payments, depend on successful payment transactions.

There needs to be secure and reliable payment processing mechanisms to ensure the dependability of financial transactions within the bookstore software.

**Order Fulfillment Dependency**

The fulfillment of book orders is dependent on various factors, such as book availability, proper coordination between the bookstore staff, and accurate order details provided by the customer.

#### **Security**

**1.** Implement access control mechanisms to ensure that each user, such as librarians, managers, and administrators, can only access the functionalities and data relevant to their roles and responsibilities within the bookstore software.

2. Ensure that personal data, including customer information and employee details, is protected and only accessible to authorized individuals. Each user should be able to view and modify only their own personal data, preventing unauthorized access or modification.

3. Implement a permission-based system where a customer's personal data can only be accessed by a librarian or manager with the explicit consent of the customer. This ensures that access to sensitive information is limited to authorized personnel and protects the privacy of the customers.

4. Apply encryption techniques to sensitive information such as passwords for user account logins and financial transactions. Utilize secure encryption algorithms and follow best practices to protect data during transmission and storage, reducing the risk of unauthorized access or data breaches.

3. Clearly communicate to users that they are responsible for maintaining the authenticity and accuracy of the personal data they enter into the bookstore software

6. Enforce a password policy that requires users to create strong and secure passwords.

7. Implement secure mechanisms for online transactions within the bookstore software.

#### 

### Organizational Requirements

Requirements which are a consequence of organizational policies and procedures e.g. process standards used, implementation requirements, etc.

#### **Environmental Requirements**

When considering environmental requirements there are a few elements to think about. Some of them include:

* **Data privacy and protection-** Implementing security measures such as: multi factor authentication, which reduces data theft, physical and cybersecurity technicians to protect the system from vandalism and intrusion and ensuring that a backup process is in place, in order to avoid possible data loss.
* **Device Compatibility-** The software should work with a variety of devices, including older ones. The usefulness of older devices can be extended and electronic waste can be avoided by configuring the app to function effectively on them.
* **Data storage-** The app needs to store user data such as bills or purchase history. Effective data storage strategies need to be used, and database optimization can help reduce the amount of storage needed.
* **Recycling programs-** We can implement and promote book recycling initiatives in collaboration with local recycling facilities.
* **Efficient Inventory Management**- Implementing inventory management strategies to decrease overstocking and the possibility of unsold or outdated books.

#### **Operational Requirements**

The software should support user operations based on normal and special scenarios, including:

* User approval processes for major transactions, such as user registration or access to sensitive information.
* Administrator approval requirement for operations related to data integrity and critical system changes.
* Handling idle and unattended periods within the application, such as automatic logouts or session timeouts for security purposes.
* Implementing constraints or limitations on user operations that are not approved or require specific permissions.
* The software should be accessible to users 24/7, allowing them to perform necessary tasks and operations at any time.
* Access to the system and information should be restricted to registered users who have the appropriate permissions and access levels.
* Ability to create, read, update, and delete user accounts within the software.
* Ability to create, read, update, and delete book information, including stock management and category updates.
* Generation of periodic reports, such as sales reports, inventory reports, or financial statements.
* Creation of book requests, purchases, and sales transactions.
* Updating and managing the bookstore's agenda, including scheduling book events or author signings.

#### **Development Requirements**

1. User Roles and Permissions:

* Implement a three-level user system with roles for Librarian, Manager, and Administrator.
* Define different views and permissions for each role to restrict access to specific functionalities.
* Ensure that each role has access to the functions of lower-level roles.

1. User Authentication and Security:

* Implement a secure user authentication system to validate user credentials and prevent unauthorized access.
* Encrypt sensitive data such as user passwords and personal information.
* Implement role-based access control (RBAC) to enforce appropriate access rights for each user role.

1. Inventory Management:

* Develop a system to track and manage bookstore inventory.
* Allow Librarians and Managers to view and update inventory information.
* Implement functionalities for adding new books, updating book details (title, author, ISBN, etc.), and removing books from stock.
* Implement inventory search and filtering capabilities for efficient management.

1. Sales Management:

* Develop features to record and track sales transactions.
* Allow Librarians and Managers to process book sales, including checkout and payment handling.
* Generate printable bills or receipts for customers.
* Provide statistics and reports on sales performance, such as total sales, top-selling books, and revenue generated.

1. Employee Performance Tracking:

* Develop a system to monitor and evaluate employee performance.
* Enable Managers and Administrators to track metrics like sales performance, customer satisfaction, and task completion rates.
* Implement features to assign tasks, set targets, and measure employee progress.
* Generate reports and analytics on employee performance to aid in decision-making.

1. Financial Management:

* Develop functionality to track income and cost data for the bookstore.
* Enable Administrators to monitor financial metrics such as revenue, expenses, and profit margins.
* Implement features for expense tracking, including inventory purchases, employee salaries, and operational costs.
* Provide financial reports and analytics to aid in financial planning and decision-making.

1. Reporting and Analytics:

* Implement reporting capabilities to generate custom reports based on user requirements.
* Provide graphical representations and visualizations of data for better insights.
* Allow users to export reports in various formats (e.g., PDF, CSV) for further analysis.

1. Usability and User Interface:

* Design an intuitive and user-friendly interface for easy navigation and interaction.
* Ensure the system is responsive and accessible across different devices and screen sizes.
* Conduct usability testing to identify and address any usability issues.

1. Performance and Scalability:

* Optimize the software for performance to handle a large amount of data and concurrent users.
* Implement efficient database design and indexing strategies.
* Consider scalability requirements to accommodate future growth and increased user load.

1. Security and Data Protection:

* Implement secure coding practices to prevent common vulnerabilities (e.g., SQL injection, cross-site scripting).
* Apply appropriate encryption mechanisms for sensitive data storage and transmission.
* Regularly update and patch the software to address security vulnerabilities.

1. Integration and APIs:

* Allow integration with external systems or APIs, such as payment gateways or book distributors, if required.
* Define clear and well-documented APIs for potential future integrations.

1. Testing and Quality Assurance:

* Conduct comprehensive testing, including unit tests, integration tests, and system tests.
* Implement automated testing where applicable to ensure software reliability.
* Perform thorough quality assurance to identify and fix bugs or usability issues.

1. Documentation and Training:

* Create comprehensive documentation for system installation, configuration, and usage.
* Develop user manuals or guides for different user roles.
* Provide training resources and conduct training sessions to familiarize users with the software.

1. Maintenance and Support:

* Plan for regular software maintenance to address bug fixes, security updates, and feature enhancements.
* Provide ongoing technical support and assistance to address user queries and issues.

### External Requirements

1. Hardware Requirements:

* Identify the hardware infrastructure needed to run the software effectively.
* Determine the minimum system requirements for client machines (computers, tablets, etc.) accessing the software.
* Consider the hardware requirements for the server hosting the software, such as processing power, memory, and storage capacity.

1. Operating System Compatibility:

* Determine the supported operating systems for both the client-side and server-side components of the software.
* Ensure compatibility with popular operating systems like Windows, macOS, and Linux.

1. Database Management System:

* Select a suitable database management system (DBMS) for storing and managing bookstore data.
* Consider factors such as data volume, performance requirements, scalability, and ease of maintenance.
* Determine whether an existing DBMS is already in use within the organization and ensure compatibility or plan for migration if necessary.

1. Integration with Existing Systems:

* Identify any existing systems or software within the bookstore's infrastructure that the new software needs to integrate with.
* Determine the required data exchange formats, protocols, or APIs for seamless integration.
* Plan for data migration or data synchronization between the new software and existing systems if needed.

1. Compliance and Legal Requirements:

* Understand and comply with relevant legal and regulatory requirements, such as data protection and privacy laws.
* Ensure the software adheres to industry standards and best practices.
* Consider any specific requirements related to the bookstore's location or jurisdiction.

1. Localization and Internationalization:

* Determine if the software needs to support multiple languages or regional preferences.
* Consider localization requirements, such as date and time formats, currency symbols, and language translations.
* Ensure the software can handle international characters and encoding if necessary.

1. Backup and Disaster Recovery:

* Develop a robust backup strategy to protect critical data in case of system failures or data loss.
* Implement regular backup procedures and establish backup storage solutions.
* Create a disaster recovery plan outlining steps to recover from system failures or major disruptions.

1. User Training and Support:

* Plan for user training sessions to familiarize bookstore staff with the software's functionalities and features.
* Provide comprehensive user documentation, manuals, and guides to assist users in using the software effectively.
* Establish a support system to address user queries, troubleshoot issues, and provide ongoing technical assistance.

1. Budget and Resource Allocation:

* Define a budget for the software development, including hardware, licensing, and personnel costs.
* Allocate resources, both human and financial, to ensure successful implementation and maintenance of the software.

1. User Acceptance Testing:

* Involve end-users in the testing process to gather feedback and ensure the software meets their requirements.
* Conduct user acceptance testing (UAT) to validate that the software functions as expected and aligns with user expectations.

1. Scalability and Future Growth:

* Consider the potential growth of the bookstore and the software's ability to scale accordingly.
* Design the software to handle increased data volume, user load, and additional features in the future.

1. Vendor Selection:

* If the software is being developed by an external vendor, perform a thorough evaluation and selection process.
* Consider factors such as the vendor's reputation, experience, support capabilities, and financial stability.

#### **Ethical Requirements**

1. Clearly state that users' personally identifiable information, such as their email address, name, surname, phone number, and other information.

2. Specify that the bookstore software will provide personally identifiable information about users to third parties only if users have given their explicit permission for such dissemination. Users should have control over their data and be able to manage their consent preferences within the software.

3. Specify that the administrator may disclose user information in order to protect property rights and ensure the security of the bookstore premises, employees, customers, and other individuals. This includes situations where there is a need to safeguard against theft, fraud, or physical harm.

4. Emphasize the importance of transparency and user control over their personal information. Users should be informed about the data collected, the purpose of its collection, and how it will be used.

Provide users with options to manage their data, review and update their personal information, and control the level of information shared with third parties.

#### **Legislative Requirements**

1. Ensure compliance with applicable laws and regulations, including consumer protection, data protection, intellectual property, and any specific regulations related to the sale of books.

2. Address potential legal scenarios where vendors, bookstores, and employees could be held liable for harm or injuries caused to customers or other parties due to software usage or negligence.

3. Respect and comply with intellectual property laws, including copyright laws, to avoid infringement and protect the rights of authors and publishers.

4. Fulfill legal obligations related to consumer protection, such as providing accurate product information, fair pricing practices, and clear refund or return policies.

5. Comply with data protection and privacy laws to safeguard customer information, implement security measures, and obtain necessary consent for data processing.

6. Seek guidance from legal professionals to ensure full compliance with all applicable laws and regulations.

##### Accounting Requirements

1. Revenue Recognition:

* Implement a system to accurately record and track revenue from book sales.
* Ensure proper recognition of revenue based on the applicable accounting principles (e.g., accrual basis or cash basis).

1. Expense Tracking:

* Develop functionality to record and track various expenses incurred by the bookstore, such as inventory purchases, employee salaries, rent, utilities, and other operating costs.
* Categorize expenses appropriately for financial reporting purposes.

1. Chart of Accounts:

* Define a suitable chart of accounts that aligns with the bookstore's financial reporting requirements.
* Assign appropriate account codes to different types of transactions and financial activities within the software.

1. General Ledger:

* Develop a general ledger module that captures all financial transactions and updates the relevant accounts.
* Ensure accurate posting of revenue, expenses, and other financial activities to the general ledger.

1. Financial Statements:

* Generate financial statements, including the income statement (profit and loss statement), balance sheet, and cash flow statement.
* Ensure the accuracy and integrity of financial data when generating these statements.

1. Accounts Receivable and Accounts Payable:

* Implement functionality to track and manage accounts receivable (customer payments) and accounts payable (vendor invoices and payments).
* Provide features to generate invoices, record customer payments, and track outstanding balances.

1. Tax Calculation and Reporting:

* Incorporate tax calculation capabilities based on the applicable tax regulations and rates.
* Generate reports or provide data required for tax filings, such as sales tax or value-added tax (VAT) reports.

1. Financial Analysis and Reporting:

* Provide features for financial analysis, such as generating financial ratios or performance indicators (e.g., gross margin, return on investment).
* Enable the generation of custom financial reports to meet specific reporting requirements.

1. Audit Trail and Financial Controls:

* Implement an audit trail functionality to track and record changes made to financial data within the software.
* Incorporate financial controls, such as user permissions and approval workflows, to ensure data integrity and prevent unauthorized modifications.

1. Integration with Accounting Systems:

* If required, enable integration with external accounting systems (e.g., QuickBooks, SAP, or other enterprise resource planning systems) to facilitate seamless data transfer and reconciliation.

1. Compliance with Accounting Standards:

Ensure compliance with relevant accounting standards and regulations (e.g., Generally Accepted Accounting Principles or International Financial Reporting Standards) applicable to the bookstore's operations and jurisdiction

## Domain Requirements

1. Book Information Management:

* The system should allow the storage and management of book information, including title, author, ISBN, publisher, publication date, price, and other relevant attributes.
* It should support the categorization and classification of books based on genres.
* The system should have the ability to handle a large volume of book records efficiently.

1. Inventory Management:

* The system should provide functionality to track the quantity of books in stock and automatically update the inventory upon sales or additions to the stock.
* It should alert the user when the stock of a particular book falls below a predefined threshold.

1. Sales and Checkout:

* The system should support the process of book checkouts, including handling multiple book purchases in a single transaction.

1. Employee Management:

* The system should provide features to manage employee information, including personal details, roles, and permissions.
* The system should allow for the tracking of employee performance metrics, such as sales or total money made.

1. Security and Privacy:

* The system should implement robust security measures to protect sensitive data, such as customer information and financial transactions.
* It should ensure proper access control and authentication mechanisms to prevent unauthorized access to the system.
* The system should adhere to privacy regulations and guidelines.

# User Scenarios/Use Cases

Provide a summary of the major functions that the product will perform. Organize the functions to be understandable to the customer or a first time reader. Include use cases and business scenarios, or provide a link to a separate document (or documents). A business scenario:

* Describes a significant business need
* Identifies, documents, and ranks the problem that is driving the scenario
* Describes the business and technical environment that will resolve the problem
* States the desired objectives
* Shows the “Actors” and where they fit in the business model
* Is specific, and measurable, and uses clear metrics for success

APPENDIX

The appendixes are not always considered part of the actual Requirements Specification and are not always necessary. They may include

* Sample input/output formats, descriptions of cost analysis studies, or results of user surveys;
* Supporting or background information that can help the readers of the Requirements Specification;
* A description of the problems to be solved by the system;
* Special packaging instructions for the code and the media to meet security, export, initial loading, or other requirements.

When appendixes are included, the Requirements Specification should explicitly state whether or not the appendixes are to be considered part of the requirements.

1. **Definitions, Acronyms, and Abbreviations**

Define all terms, acronyms, and abbreviations used in this document.

1. **References**

List all the documents and other materials referenced in this document.

1. **Requirements Traceability Matrix**

The following trace matrix examples show one possible use of naming standards for deliverables (FunctionalArea-DocType-NN). The number has no other meaning than to keep the documents unique. For example, the Bargaining Unit Assignment Process Flow would be BUA-PF-01.

For example (1):

| **Business Requirement** | **Area** | **Deliverables** | **Status** |
| --- | --- | --- | --- |
| BR\_LR\_01  The system should validate the relationship between Bargaining Unit/Location and Job Class.---Comments: Business Process = "Assigning a Bargaining Unit to an Appointment" (Priority 1) | BUA | BUA-CD-01  Assign BU Conceptual Design | Accepted |
| BUA-PF-01  Derive Bargaining Unit-Process Flow Diagram | Accepted |
| BUA-PF-01  Derive Bargaining Unit-Process Flow Diagram | Accepted |
| BR\_LR\_09  The system should provide the capability for the Labor Relations Office to maintain the job class/union relationship.---Comments: Business Process = "Maintenance" (Priority 1) | BUA | BUA-CD-01  Assign BU Conceptual Design | Accepted |
| BUA-PF-02  BU Assignment Rules Maint Process Flow Diagram | ReadyForReview |

For example (2):

| **BizReqID** | **Pri** | **Major Area** | **DevTstItems DelivID** | **Deliv Name** | **Status** |
| --- | --- | --- | --- | --- | --- |
| BR\_LR\_01 | 1 | BUA | BUA-CD-01 | Assign BU Conceptual Design | Accepted |
| BR\_LR\_01 | 1 | BUA | BUA-DS-02 | Bargaining Unit Assignment DB Modification Description | Accepted |
| BR\_LR\_01 | 1 | BUA | BUA-PF-01 | Derive Bargaining Unit-Process Flow Diagram | Accepted |
| BR\_LR\_01 | 1 | BUA | BUA-UCD-01 | BU Assign LR UseCase Diagram | ReadyForReview |
| BR\_LR\_01 | 1 | BUA | BUA-UCT-001 | BU Assignment by PC UseCase - Add Appointment and Derive UBU | Reviewed |
| BR\_LR\_01 | 1 | BUA | BUA-UCT-002 | BU Assignment by PC UseCase - Add Appointment (UBU Not Found) | Reviewed |
| BR\_LR\_01 | 1 | BUA | BUA-UCT-006 | BU Assignment by PC UseCase - Modify Appointment (Removed UBU) | Reviewed |
| BR\_LR\_09 | 1 | BUA | BUA-CD-01 | Assign BU Conceptual Design | Accepted |
| BR\_LR\_09 | 1 | BUA | BUA-DS-02 | Bargaining Unit Assignment DB Modification Description | Accepted |
| BR\_LR\_09 | 1 | BUA | BUA-PF-02 | BU Assignment Rules Maint Process Flow Diagram | Accepted |
| BR\_LR\_09 | 1 | BUA | BUA-UCD-03 | BU Assign Rules Maint UseCase Diagram | Reviewed |
| BR\_LR\_09 | 1 | BUA | BUA-UCT-045 | BU Assignment Rules Maint: Successfully Add New Assignment Rule | Reviewed |
| BR\_LR\_09 | 1 | BUA | BUA-UCT-051 | BU Assignment Rules MaintUseCase: Modify Rule | Reviewed |
| BR\_LR\_09 | 1 | BUA | BUA-UCT-053 | BU Assignment Rules MaintUseCase - Review Assignment Rules | Reviewed |
| BR\_LR\_09 | 1 | BUA | BUA-UCT-057 | BU Assignment Rules MaintUseCase: Inactivate Last Rule for a BU | Reviewed |
| BR\_LR\_09 | 1 | BUA | BUA-UI-02 | BU AssignRules Maint UI Mockups | ReadyForReview |
| BR\_LR\_09 | 1 | BUA | BUA-TC-021 | BU Assignment Rules Maint TestCase: Add New Rule (Associated Job Class Does Not Exist) - Success | ReadyForReview |
| BR\_LR\_09 | 1 | BUA | BUA-TC-027 | BU Assignment Rules Maint TestCase: Modify Rule - Success | ReadyForReview |
| BR\_LR\_09 | 1 | BUA | BUA-TC-035 | BU Assignment Rules Maint TestCase: Add New Rule (Associated Job Class Does Not Exist) - Error Condition | ReadyForReview |
| BR\_LR\_09 | 1 | BUA | BUA-TC-049 | BU Assignment Rules Maint TestCase: Modify Rule - Error Condition | ReadyForReview |

For example (3):

| **BizReqID** | **CD01** | **CD02** | **CD03** | **CD04** | **UI01** | **UI02** | **UCT01** | **UCT02** | **UCT03** | **TC01** | **TC02** | **TC03** | **TC04** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| BR\_LR\_01 |  |  | X |  | X |  | X |  |  | X |  | X |  |
| BR\_LR\_09 | X |  |  | X |  | X |  |  | X |  | X |  | X |
| BR\_LR\_10 | X |  |  | X |  |  |  |  | X |  | X |  |  |
| BR\_LR\_11 |  | X |  |  |  |  |  |  |  |  |  |  |  |

1. **Organizing the Requirements**

This section is for information only as an aid in preparing the requirements document.

Detailed requirements tend to be extensive. Give careful consideration to your organization scheme. Some examples of organization schemes are described below:

**By System Mode**

Some systems behave quite differently depending on the mode of operation. For example, a control system may have different sets of functions depending on its mode: training, normal, or emergency.

**By User Class**

Some systems provide different sets of functions to different classes of users. For example, an elevator control system presents different capabilities to passengers, maintenance workers, and fire fighters.

**By Objects**

Objects are real-world entities that have a counterpart within the system. For example, in a patient monitoring system, objects include patients, sensors, nurses, rooms, physicians, medicines, etc. Associated with each object is a set of attributes (of that object) and functions (performed by that object). These functions are also called services, methods, or processes. Note that sets of objects may share attributes and services. These are grouped together as classes.

**By Feature**

A feature is an externally desired service by the system that may require a sequence of inputs to affect the desired result. For example, in a telephone system, features include local call, call forwarding, and conference call. Each feature is generally described in a sequence of stimulus-response pairs, and may include validity checks on inputs, exact sequencing of operations, responses to abnormal situations, including error handling and recovery, effects of parameters, relationships of inputs to outputs, including input/output sequences and formulas for input to output.

**By Stimulus**

Some systems can be best organized by describing their functions in terms of stimuli. For example, the functions of an automatic aircraft landing system may be organized into sections for loss of power, wind shear, sudden change in roll, vertical velocity excessive, etc.

**By Response**

Some systems can be best organized by describing all the functions in support of the generation of a response. For example, the functions of a personnel system may be organized into sections corresponding to all functions associated with generating paychecks, all functions associated with generating a current list of employees, etc.

**By Functional Hierarchy**

When none of the above organizational schemes prove helpful, the overall functionality can be organized into a hierarchy of functions organized by common inputs, common outputs, or common internal data access. Data flow diagrams and data dictionaries can be used to show the relationships between and among the functions and data.

**Additional Comments**

Whenever a new Requirements Specification is contemplated, more than one of the organizational techniques given above may be appropriate. In such cases, organize the specific requirements for multiple hierarchies tailored to the specific needs of the system under specification.

There are many notations, methods, and automated support tools available to aid in the documentation of requirements. For the most part, their usefulness is a function of organization. For example, when organizing by mode, finite state machines or state charts may prove helpful; when organizing by object, object-oriented analysis may prove helpful; when organizing by feature, stimulus-response sequences may prove helpful; and when organizing by functional hierarchy, data flow diagrams and data dictionaries may prove helpful.