Software Requirements Specification document for project Booksmania

Ibrahim Khalid , Mohab Mahmoud
Subervised by : Dr. Marwa Solayman & Eng. Ragda Essam
Nuveber 9 , 2023

1. Introduction

1.1Purpose

The purpose of this document is to outline the functional and non-functional requirements for the "Booksmania" website, an online platform dedicated to selling various books.

1.2 Scope

Booksmania will allow visitors to explore a wide range of books, make purchases, and facilitate book delivery. The website's primary goal is to provide an efficient and user-friendly platform for book buyers and site managers.

1.3 Overview

The "Booksmania" project is an electronic e-commerce website dedicated to selling a diverse collection of books. This platform allows visitors to explore and select books of their choice, facilitating the purchase process and ensuring the timely delivery of purchased books. The system's primary objectives are to provide an efficient and user-friendly online shopping experience for book enthusiasts, while also ensuring dependability, environmental sustainability, and compliance with ethical, regulatory, and legal standards.

1.4 **Business Context**

The 'Booksmania' project operates within the competitive landscape of the online e-commerce industry, serving a diverse audience of book enthusiasts, students, and professionals. Its core objective is to provide a dependable, user-friendly online platform for purchasing books, offering a wide array of book genres and ensuring an efficient and secure shopping experience. The project places strong emphasis on ethical, legal, and regulatory compliance, safeguarding user data and intellectual property rights, while also prioritizing environmental sustainability through energy-efficient practices. In this context, 'Booksmania' aims to differentiate itself by focusing on dependability, user experience, and ethical standards in the rapidly evolving world of online book sales

2. General Description

2.1Product Functions

The "Booksmania" project is designed to serve as an electronic platform for the sale of books. Its primary functions include enabling users to create accounts, browse and search for books, add items to a shopping cart, complete the checkout process, manage orders, and receive book deliveries. Site managers have functions related to overseeing user accounts, processing purchase requests, and assigning delivery representatives. The platform also facilitates user feedback and reporting.

2.2 Similar System Information

"Booksmania" is intended to be a stand-alone e-commerce platform dedicated to book sales. While it operates independently, it may potentially integrate with third-party payment gateways and shipping services to enhance its functionality and deliver an all-encompassing user experience.

2.3 User Characteristics

The user community for "Booksmania" is expected to be diverse, consisting of book enthusiasts, students, professionals, and individuals from various backgrounds. The system is designed to accommodate users with varying levels of expertise in software systems and the domain of online book purchasing.

2.4 User Problem Statement

Users today face challenges in finding a reliable and efficient online platform for purchasing books that meets their diverse needs. Many existing solutions lack the comprehensive book collections or dependable user experiences users seek.

2.5 User Objectives

From the user's perspective, "Booksmania" aims to provide a set of objectives and requirements, including:

Offering a diverse and extensive collection of books across genres.
Providing a seamless, secure, and user-friendly online shopping experience.
Ensuring data privacy and upholding ethical and legal standards.
Delivering responsive customer support services.
Prioritizing environmental responsibility through energy-efficient practices.

2.6 General Constraints

The project constraints include:

Compatibility with modern web browsers (e.g., Chrome, Firefox, Safari).
Reliance on a reliable web hosting service.
Integration with third-party payment gateways and shipping services.
Supporting a maximum of 10,000 concurrent users.

3. Functional requirements

3.1 User Register

Function Name	Enable User Registration
Description	Users must be able to create an account by providing their personal information, including name, email address, username, and password.
Criticality	The system must securely store and manage user data, implement password encryption, and provide mechanisms for email verification.
Technical Issues	The system must securely store and manage user data, implement password encryption, and provide mechanisms for email verification.
Cost and schedule	The development and implementation of secure user registration may impact project timelines and development costs.
Risks	Failure to implement secure registration can lead to data breaches and unauthorized access. Measures must be taken to reduce security risks.
Dependencies with other requirements	This requirement is dependent on a secure authentication and user data management system.
Pre-Condition	The system is available for user registration.
Post-Condition	Users have registered accounts and can log in for access.

3.2 User authenticate

Function Name	Enable User Authentication
Description	Enables users to log in securely with unique credentials to access the system.
Criticality	The system must securely store and manage user data, implement password encryption, and provide mechanisms for email verification.
Technical Issues	Security protocols, encryption, database management

Cost and schedule	Medium cost, early implementation
Risks	Security breaches, user data vulnerability
Dependencies with other requirements	Dependent on database setup and encryption algorithms
Pre-Condition	User has valid login credentials
Post-Condition	User gains access to system features upon successful authentication

3.3 Book Search

Function Name	Implement Book Search Functionality
Description	Users should be able to search for books by title, author, genre, or keywords to facilitate book discovery.
Criticality	This requirement is crucial for users to find and select books efficiently.
Technical Issues	Implementing an effective search algorithm to retrieve relevant book listings.
Cost and schedule	The development of a robust search function may require additional resources and time.
Risks	Inefficient search may lead to user frustration and hinder book selection.
Dependencies with other requirements	This requirement is dependent on the availability of a book database and user interface.
Pre-Condition	The user initiates a book search.
Post-Condition	The system displays search results based on user queries.

3.4 Make Order

Function Name	Allow Order Placement
Description	Users should be able to place orders for selected books after providing shipping and payment information.
Criticality	This requirement is essential for the primary function of the system.
Technical Issues	Secure handling of payment information and order processing.

Cost and schedule	Implementing secure payment processing may have associated costs and development time.
Risks	Inadequate payment security could lead to financial risks for users.
Dependencies with other requirements	This requirement relies on the availability of selected books, shopping cart functionality, and user data management.
Pre-Condition	The user has items in the shopping cart and is ready to place an order.
Post-Condition	An order confirmation is provided to the user, and the order is processed for delivery.

3.5 Return order

Function Name	Return Book
Description	Allows users to return purchased or rented books
Criticality	High
Technical Issues	Integration with database for book inventory updates
Cost and schedule	Estimated implementation time: 2 weeks
Risks	Potential delays if database integration is complex
Dependencies with other requirements	Depends on book purchase/rental and user authentication
Pre-Condition	Book must be in acceptable condition for return
Post-Condition	Book inventory updated; user record reflects return

3.6 Adjust book inventory

Function Name	Making book Inventory adjustment
Description	Manages real-time updates of book quantities, locations, and availability within the bookstore.
Criticality	This requirement is essential for the primary function of the system.
Technical Issues	Bar-code scanning integration, database synchronization
Cost and schedule	Moderate cost, early to mid-stage implementation
Risks	Inventory discrepancies, data synchronization issues
Dependencies with other requirements	Relies on accurate book data and system synchronization
Pre-Condition	Availability of book inventory information
Post-Condition	Accurate tracking of book quantities and locations in the system

3.7 Write Review

Function Name	Write Review for a Book
Description	Allows registered users to submit reviews and ratings for specific books available in the bookstore's catalog.
Criticality	This requirement is essential for the primary function of the system.
Technical Issues	User interface design, database integration
Cost and schedule	Moderate cost, mid-stage implementation
Risks	Fake/malicious reviews, data storage
Dependencies with other requirements	Depends on user profile and book availability
Pre-Condition	User intends to review a book.
Post-Condition	Review and rating are successfully recorded for the specified book in the system

3.8 Send Announcement

Function Name	Send Announcement
Description	Allows administrators to broadcast messages to users
Criticality	Medium
Technical Issues	Integration with email service or notification system
Cost and Schedule	Low cost; estimated implementation time: 1 week
Risks	Spamming concerns if not regulated properly
Dependencies with other requirements	Admin access and user notification settings
Pre-Condition	Admin authentication and authorized access
Post-Condition	Announcement sent and visible to intended recipients

3.9 write report

Function Name	Write Report
Description	Allows users or administrators to generate reports
Criticality	Medium
Technical Issues	Report generation tools or APIs integration
Cost and Schedule	Moderate cost; estimated implementation time: 3 weeks
Risks	Data accuracy issues if reporting tools are faulty
Dependencies with other requirements	Access to necessary data and user permissions
Pre-Condition	User/admin logged in and access to report features
Post-Condition	Report generated and accessible/downloadable for user

3.10 Edit Profile

Function Name	Edit Profile
Description	Allows users to modify their profile information
Criticality	High
Technical Issues	User interface design for profile editing
Cost and Schedule	Moderate cost; estimated implementation time: 2 weeks
Risks	Data security risks if not properly authenticated
Dependencies with other requirements	User authentication and profile management system
Pre-Condition	User logged in and access to profile settings
Post-Condition	Updated profile information saved and visible

3.11 Support Request

Function Name	Support Request
Description	Enables users to submit inquiries or issues for assistance
Criticality	High
Technical Issues	Implementation of a ticketing system or support interface
Cost and Schedule	Moderate cost, estimated implementation time: 3 weeks
Risks	Overwhelming volume of requests; inadequate response time
Dependencies with other requirements	User authentication and support team infrastructure
Pre-Condition	User access to support/request feature
Post-Condition	Support ticket logged or acknowledgment to the use

3.12 Give Shipping Order

Function Name	Give Shipping Order
Description	Allows staff/admins to generate and assign shipping orders
Criticality	High
Technical Issues	Integration with delivery management system or API
Cost and Schedule	Moderate cost; estimated implementation time: 2 weeks
Risks	Incorrect shipping details; delays in delivery or lost packages
Dependencies with other requirements	Inventory management, order processing, and user authentication
Pre-Condition	Staff/admin logged in and access to shipping management features
Post-Condition	Shipping order generated and assigned to delivery representative

3.13 Profits Calculate

Function Name	Profits Calculation
Description	Calculates and tracks profits generated from book sales
Criticality	High
Technical Issues	Integration with accounting system or calculation algorithms
Cost and Schedule	Moderate cost; estimated implementation time: 4 weeks
Risks	Inaccurate calculations affecting financial reports
Dependencies with other requirements	Sales data tracking, inventory management, payment systems
Pre-Condition	Access to sales and financial data
Post-Condition	Accurate profit figures generated and accessible

3.1 Interface Requirements

3.5 User Interfaces

4.1.1 GUI (Graphical User Interface)
The "Booksmania" platform provides a user-friendly graphical user interface for customers and site managers. It includes the following features:

Home Page: Displays book listings, featured books, and search options.
Book Listings: Shows book details, including title, author, price, and description.
Shopping Cart: Allows users to manage their selected items before checkout.
Checkout: Facilitates the input of shipping and payment information.
User Account: Enables users to manage their profiles, orders, and settings.
(Mock-ups and screen dumps illustrating the GUI features are available in the design documentation.)

4.1.2 CLI (Command-Line Interface)
A command-line interface is not applicable to the "Booksmania" project, as the system primarily relies on a graphical user interface for user interactions.

4.1.3 API (Application Programming Interface)
The "Booksmania" project offers an API for potential integration with external services and applications. It includes functions for accessing book listings, user accounts, and order data. The API documentation provides details such as function names, arguments, return values, example invocations, and interactions with other functions.

4.1.4 Diagnostics or ROM

The system provides diagnostic capabilities for troubleshooting and debugging. Users or administrators can access diagnostic data through the admin dashboard. This information includes error logs, system status, and user activity logs.

3.6 Hardware Interfaces

The "Booksmania" system does not have direct interfaces with hardware devices, as it primarily operates as a web-based application. However, it relies on standard web server hardware and network infrastructure for hosting and data transmission.

3.7 Communications Interfaces

The system includes communication interfaces for online transactions and data exchange. These interfaces support secure communication between the user's browser and the system's web server, ensuring the privacy and integrity of user data during interactions such as user registration, login, and order processing.

3.8 Software Interfaces

Apart from the user and communication interfaces, the "Booksmania" system interfaces with the following software components:

Payment Gateway: The system integrates with a third-party payment gateway to process payments securely. The interface ensures that payment information is transmitted and verified without compromising security.
Shipping Services: To facilitate book delivery, the system interfaces with external shipping services to coordinate order shipments and provide delivery tracking.

Database Management System: The system interacts with a relational database management system (DBMS) to store and retrieve book listings, user profiles, and order information

4. Performance Requirements

The "Booksmania" platform must meet specific performance criteria to ensure its efficient and responsive operation. Performance requirements are critical to delivering a satisfactory user experience.

4.1 Response Time

Page Load Time: The system shall ensure that web pages load within a maximum of 2 seconds under normal operating conditions. This applies to all pages, including the home page, book listings, shopping cart, and checkout.

Search Response Time: The search functionality should return relevant results within 1 second, considering a typical query for book listings.

4.2 Scalability

User Scalability: The system must be able to handle a minimum of 10,000 concurrent users without a significant degradation in performance. It should be capable of dynamically scaling to accommodate increased user loads.

Inventory Scalability: As the inventory of books grows, the system must handle a minimum of 100,000 book listings efficiently without impacting search and listing response times

4.3 Security Performance

Authentication Response Time: User authentication and login processes should have a response time of less than 1 second to prevent delays in accessing user accounts.

Payment Processing: Payment processing should be completed within 5 seconds, ensuring quick and secure financial transactions.

4.4 Database Performance

Database Query Response Time: Database queries to retrieve book listings, user data, and order information should have a response time of less than 2 seconds.
5.5 Concurrent Order Processing
Order Processing: The system should be able to handle concurrent order placements by multiple users without errors. It should efficiently process a minimum of 100 orders per minute during peak usage.

4.5 Error Handling

Error Response Time: The system should provide error messages and notifications to users in case of issues. Error messages must be delivered within 3 seconds to guide users on issue resolution

4.6 Reporting Performance

Report Generation: The generation of sales reports and analytic should be completed within a maximum of 5 seconds to provide site managers with timely business insights.These performance requirements are crucial to ensuring that the "Booksmania" platform operates efficiently, even under heavy user loads, and delivers a responsive and reliable user experience.

5. Design Constraints

This section outlines specific constraints and limitations that the design team must consider when developing the "Booksmania" project.

5.1 Standards Compliance

Web Browser Compatibility: The system design must comply with established web standards to ensure compatibility with modern web browsers, including but not limited to Chrome, Firefox, and Safari. Adherence to web standards is crucial for consistent functionality and appearance across browsers.
Data Protection Regulations: The system must strictly adhere to relevant data protection and e-commerce regulations, ensuring that user data is handled in accordance with applicable legal standards. Compliance with these regulations may impact the system's design and functionality.

5.2 Hardware Limitations

Web Hosting Service: The system's design must accommodate the hosting infrastructure provided by the selected web hosting service. Hardware limitations and capabilities of the hosting environment may influence system performance and availability.

5.3 Others as Appropriate

Third-Party Integration: The system design should account for the integration with third-party services, including a payment gateway and external shipping services. Any constraints, limitations, or updates imposed by these services must be considered during system design to ensure seamless operations.

Maximum User Load: The design team must adhere to the specified constraint of supporting a maximum of 10,000 concurrent users. The system design should prioritize responsiveness and stability under this user load.

Technology Stack: The project's design should align with the chosen technology stack, encompassing specific technologies, frameworks, and programming languages. Any changes or updates to these components may necessitate corresponding adjustments in the system's design and implementation.

Mobile Responsiveness: The system's design should prioritize a responsive and mobile-friendly user experience. Ensuring mobile accessibility and usability is essential for users accessing the platform on various mobile devices.

6. Non-Functional Requirements

The "Booksmania" project must meet specific non-functional attributes to ensure the system's overall quality, performance, and reliability. These attributes are as follows:

6.1 Security

User Data Security: The system must employ robust encryption techniques to safeguard user data, including personal information and payment details.

Access Control: User access must be controlled through secure authentication mechanisms, and authorization must be based on user roles.

Protection Against Cyber Threats: The system should include security measures to guard against common cyber threats such as SQL injection, cross-site scripting (XSS), and Distributed Denial of Service (DDoS) attacks.

6.2 Binary Compatibility

Browser Compatibility: The system's web interface must be compatible with various web browsers, including Chrome, Firefox, Safari, and Edge.

Operating System Compatibility: The system should be compatible with major operating systems, such as Windows, macOS, and Linux.

6.3 Reliability

Mean Time to Failure (MTTF): The system's mean time between failures should be at least 30 days, ensuring its reliable operation.

Availability: The system should be available 24/7 with a maximum downtime of 1 hour per month for scheduled maintenance.

6.4 Maintainability

Code Maintainability: The system's code must be well-documented and follow coding standards to ensure ease of maintenance and future development.

Scalability: The system should be designed to scale easily, accommodating an increasing number of books and users.

6.5 Portability

Cross-Platform Compatibility: The system's web application must be responsive and compatible with a wide range of devices, including desktop computers, tablets, and smartphones.

6.6 Extensibility

Modular Design: The system should have a modular architecture that allows for easy integration of additional features and services in the future.

6.7 Re-usability

Code Re-usability: Code components should be designed with re-usability in mind to minimize redundancy and enhance development efficiency.

6.8 Application Affinity/Compatibility

Third-Party Integration: The system must be compatible with and support integration with third-party services, such as payment gateways and shipping services.

6.9 Resource Utilization

Optimized Resource Usage: The system should be designed to use system resources efficiently, minimizing CPU and memory usage.

6.10 Serviceability

Customer Support: A dedicated customer support team should be available to address user inquiries and issues, aiming for a response time of 24 hours.

7. Preliminary Object-Oriented Domain Analysis

This section provides a structural view of the fundamental objects that must be modeled within the system to satisfy its requirements. It also includes information on inheritance

relationships, class descriptions, and their purposes within the system.

8.1 Inheritance Relationships

This section should contain a set of graphs that illustrate the primary inheritance
hierarchy (is-kind-of) for the system.

There is no Inheritance Relationships in this software system .

8.2 Class Descriptions

8.2.1 Class: Book

Abstract or Concrete: Concrete
List of Superclasses: None
List of Subclasses: None
Purpose: Represents a book available for sale on the platform. Contains book details, such as title, author, description, and price.
Collaborations: Collaborates with the "Shopping Cart" class to allow users to add books for purchase.

Attributes:
Title (String)
Author (String)
Description (String)
Price (Float)
ISBN (String)

Operations:
getDetails(): Returns book details.
setPrice(newPrice: Float): Sets a new price for the book.
updateDescription(newDescription: String): Updates the book's description.

8.2.2 Class: User

Abstract or Concrete: Concrete
List of Superclasses: None
List of Subclasses: None
Purpose: Represents a registered user of the "Booksmania" platform. Contains user information, including name, email, and password.
Collaborations: Collaborates with the "Shopping Cart" class for users to manage their shopping carts.
Attributes:
Name (String)
Email (String)
Password (String)
AccountType (Enum: Admin, Customer)
Operations:
getUserInfo(): Returns user information.
changePassword(newPassword: String): Allows the user to change their password.
viewOrders(): Retrieves a list of the user's past orders.

8.3.3 Class: Shopping Cart

Abstract or Concrete: Concrete
List of Superclasses: None
List of Subclasses: None
Purpose: Represents a shopping cart for users to add and manage books they wish to purchase.
Collaborations: Collaborates with the "Book" class and the "User" class.

Attributes:
CartItems (List of Book)
TotalPrice (Float)
Operations:
addToCart(book: Book): Adds a book to the shopping cart.

removeFromCart(book: Book): Removes a book from the shopping cart.
checkout(): Initiates the checkout process, updating the order and payment details.

8.2.4 Class: Order

Abstract or Concrete: Concrete
List of Superclasses: None
List of Subclasses: None
Purpose: Represents an order placed by a user for one or more books. Contains order details, shipping information, and order status.
Collaborations: Collaborates with the "User" class for user information and the "Book" class for book details.
Attributes:
OrderID (String)
User (User)
OrderedBooks (List of Book)
ShippingAddress (String)
OrderStatus (Enum: Pending, In Progress, Delivered)
Operations:
getOrderDetails(): Returns order details, including book titles, shipping address, and status.
updateStatus(newStatus: Enum): Updates the order status.

8.2.5 Class: Delivery Representative

Abstract or Concrete: Concrete
List of Superclasses: None
List of Subclasses: None
Purpose: Represents a delivery representative responsible for delivering orders to customers.
Collaborations: Collaborates with the "Order" class for order assignment and status updates.

Attributes:
RepresentativeID (String)
AssignedOrders (List of Order)

Operations:
viewAssignedOrders(): Retrieves a list of orders assigned to the delivery representative.
updateOrderStatus(order: Order): Updates the order status upon successful delivery.

8.2.6 Class: Admin

Abstract or Concrete: Concrete
List of Superclasses: None
List of Subclasses: None
Purpose: Represents an administrator with additional privileges to manage the platform.
Collaborations: Collaborates with the "User" class for user information and the "Book" class for book management.
Attributes:
AdminID (String)
ManagedBooks (List of Book)
ManagedUsers (List of User)
Operations:
addBook(newBook: Book): Adds a new book to the platform.
removeBook(book: Book): Removes a book from the platform.
manageUsers(user: User): Manages user accounts, including granting admin privileges.

8.2.7 Class: Report

Abstract or Concrete: Concrete
List of Superclasses: None
List of Subclasses: None
Purpose: Represents a report generated by site managers to track sales and user analytics.

Collaborations: Collaborates with the "Admin" class for report generation.
Attributes:
ReportID (String)
ReportType (Enum: Sales, User Analytics)
GeneratedBy (Admin)
ReportData (String)
Operations:
getReportDetails(): Retrieves report details, including data and report type.
updateReportData(newData: String): Updates the report data.

8.2.8 Class: Feedback

Abstract or Concrete: Concrete
List of Superclasses: None
List of Subclasses: None
Purpose: Represents user feedback and ratings provided for purchased books.
Collaborations: Collaborates with the "User" class for user feedback submission and the "Book" class for book ratings.
Attributes:
FeedbackID (String)
User (User)
Book (Book)
Rating (Integer)
Comment (String)

Operations:
getFeedbackDetails(): Retrieves feedback details, including user comments and ratings.
addFeedback(rating: Integer, comment: String): Allows users to submit feedback for a book.

8.2.9 Class: Customer Support

Abstract or Concrete: Concrete
List of Superclasses: None
List of Subclasses: None
Purpose: Represents the customer support service for user inquiries and issue resolution.
Collaborations: Collaborates with the "User" class for user inquiries and the "Admin" class for issue resolution.
Attributes:
SupportID (String)
User (User)
Admin (Admin)
Inquiry (String)
Resolution (String)
Operations:
createInquiry(inquiry: String): Allows users to create inquiries for support.
resolveInquiry(admin: Admin, resolution: String): Permits administrators to resolve user inquiries.

9. Operational Scenarios

Scenario 1: User Registration

Use Case: User Registration

Description: A new user, Sarah, wants to join the "Booksmania" platform. She navigates to the registration page.
Steps:
Sarah accesses the "Booksmania" website and clicks on the "Sign Up" button.
She fills in her personal details, including her name, email address, chosen username, and password.
After completing the registration form, Sarah clicks the "Register" button.

The system validates her input, sends a verification email, and redirects her to the login page.

Sarah receives a verification email, clicks the verification link, and her account is confirmed.She can now log in using her credentials.

Scenario 2: Book Search and Purchase

Use Case: Book Search and Purchase

Description: John, a registered user, wants to find and purchase a book on the platform.

Steps:

John logs in to his "Booksmania" account.

He uses the search bar to find a specific book or explores categories and genres to browse the collection.

After finding the desired book, John clicks on it to view details, including title, author, price, and description.

He decides to purchase the book and adds it to his shopping cart.

John proceeds to checkout, enters shipping and payment information, and places the order.

The system confirms the order, and John receives an order confirmation.

The book is prepared for delivery to John's address.

Scenario 3: Site Manager Order Processing

Use Case: Site Manager Order Processing

Description: A site manager, Alex, receives a list of pending orders and needs to process them for delivery.

Steps:

Alex accesses the admin dashboard by logging in with managerial credentials.

In the dashboard, he can view a list of pending orders, including order details and customer information.

Alex reviews the orders and assigns them to available delivery representatives for fulfillment.

He confirms the assignment and updates the order status to "In Progress."

The system notifies the assigned delivery representative about the new order. After the delivery is complete, the order status is updated to "Delivered."

Scenario 4: User Feedback and Support

Use Case: User Feedback and Support

Description: Lisa, a user, has read a book and wants to provide feedback and request support for an inquiry.

Steps:

Lisa logs in to her "Booksmania" account.

She navigates to the book's page and finds an option to rate and review the book.

Lisa rates the book and writes a review, which she submits.

Later, Lisa has a question about her recent order. She clicks on the "Contact Support" option.

Lisa submits her inquiry, including a description of the issue and any relevant order details.

The system generates a support ticket for her inquiry.

A customer support representative responds to her inquiry within 24 hours.

10. Preliminary Schedule Adjusted

The "Booksmania" project requires an initial project plan to outline the major tasks, their inter-dependencies, and tentative start/stop dates. This preliminary schedule will guide the project's development and ensure that it aligns with the proposed timeline. The project plan is accompanied by a Gantlet chart to visualize the project's timeline.

10.1 Major Tasks

The project comprises several major tasks, each contributing to the successful development and deployment of the "Booksmania" platform. These tasks are as follows:

Project Initiation: Define project goals, objectives, and scope. Assemble the project team and allocate responsibilities.

Requirements Gathering: Conduct in-depth requirements analysis, including functional and non-functional requirements.

Design and Prototyping: Create user interface mock-ups, design the database schema, and develop prototypes.

Software Development: Write and test the code for the platform, including user registration, book listings, shopping cart, and order processing.

Third-Party Integration: Integrate with payment gateway and shipping services.

Quality Assurance: Conduct comprehensive testing, including unit testing, integration testing, and user acceptance testing.

Deployment and Hosting: Deploy the platform on the selected web hosting service and ensure a stable production environment.

Marketing and Promotion: Launch marketing campaigns and user acquisition efforts to promote the platform.

User Support and Feedback: Establish customer support mechanisms and address user inquiries and feedback.

Ongoing Development: Plan for future feature enhancements, updates, and maintenance.

10.2 Inter-dependencies

Several tasks are interdependent, and the project schedule must account for these dependencies:

Requirements gathering must be completed before design and development can begin.
Design and prototyping precede software development.
Software development precedes quality assurance and testing.
Marketing and promotion activities run concurrently with software development and testing.
User support and feedback mechanisms must be in place when the platform is launched.
Ongoing development is continuous and may overlap with other tasks.

10.3 Tentative Start/Stop Dates

The tentative start and stop dates for each major task are as follows:

[Insert Gantlet chart here to visualize the project timeline.]

10.4 Resource Requirements

The successful execution of the project requires the allocation of various resources, including:

Human Resources: A dedicated project team with software developers, designers, QA testers, and marketing specialists.
Hardware: Suitable hardware for development and hosting.
Software: Development tools, project management software, and other software resources.

11. Preliminary Budget Adjusted

The "Booksmania" project requires an initial budget to support its development and deployment. The budget is itemized to provide an overview of the anticipated expenses associated with the project. This preliminary budget is subject to adjustments and refinements as the project progresses.

11.1 Development Costs

Software Development: Initial costs for software development, including salaries, software licenses, development tools, and project management software.

Design and User Experience: Expenses related to user interface design, user experience testing, and mock-up creation.

Server Hosting: Costs associated with web hosting services to host the "Booksmania" platform.

11.2 Integration Costs

Third-Party Services: Fees for integrating third-party services, such as payment gateways and shipping services.
Database Management: Expenses for database management services and tools.

11.3 Marketing and Promotion

Marketing Campaigns: Funds allocated for marketing campaigns, including online advertising, social media promotions, and search engine optimization (SEO).

User Acquisition: Costs for acquiring initial users, which may include referral incentives or promotional offers.

12. Appendices

This section provides additional information to aid in understanding the requirements and content of the SRS document.

12.1 Definitions, Acronyms, Abbreviations

This appendix provides definitions for unfamiliar terms, acronyms, and abbreviations used throughout the SRS document. It serves as a reference to ensure a common understanding of key terminology.

Definitions

SRS (Software Requirements Specification): A document that outlines the functional and non-functional requirements of a software project.
GUI (Graphical User Interface): The visual interface that allows users to interact with software applications.
API (Application Programming Interface): A set of rules and protocols that enable different software applications to communicate with each other.
QA (Quality Assurance): The process of ensuring that a software application meets predefined quality standards and specifications.
DBMS (Database Management System): Software that manages and organizes data in a structured manner within a database.
PERT (Program Evaluation and Review Technique): A project management tool used to schedule, organize, and coordinate tasks within a project.
Gantlet Chart: A visual representation of a project's schedule that displays tasks, their dependencies, and timeline.

12.2 Collected Material

This appendix includes various materials collected during the requirements analysis and documentation process. It may consist of diagrams, charts, mock-ups, or any additional documentation that aids in understanding the project requirements and objectives.

The collected material provides supplementary information that complements the main body of the SRS document, offering a more comprehensive view of the project's scope and requirements.