
Software Requirements Specification

for

E-bookstore

Version 1.0 approved

Prepared by EWB

09/11/2024

Table of Contents

Table of Contents

Revision History

i

1. Introduction

- 1.1 Purpose
- 1.2 Intended Audience and Reading Suggestions
- 1.3 Project Scope

2. Overall Description

- 2.1 Product Perspective
- 2.2 Product Features
- 2.3 User Classes and Characteristics
- 2.4 Operating Environment
- 2.5 Design and Implementation Constraints
- 2.6 User Documentation
- 2.7 Assumptions and Dependencies

3. System Features

- 3.1 Search and Filter
 - 3.1.1 Description and Priority
 - 3.1.2 Stimulus/Response Sequences
 - 3.1.3 Functional Requirements
- 3.2 Add to Wishlist
 - 3.2.1 Description and Priority
 - 3.2.2 Stimulus/Response Sequences
- 3.3 Log in / Sign up
 - 3.3.1 Description and Priority
 - 3.3.2 Stimulus/Response Sequences
 - 3.3.3 Functional Requirements
- 3.4 Add to Basket & My Books
 - 3.4.1 Description and Priority
 - 3.4.2 Stimulus/Response Sequences

4. External Interface Requirements⁴

- 4.1 User Interfaces
- 4.2 Hardware Interfaces
- 4.3 Software Interfaces
- 4.4 Communications Interfaces

5. Other Nonfunctional Requirements

- 5.1 Performance Requirements
 - 5.1.1 Performance Requirements
 - 5.1.2 Search Performance
 - 5.1.3 Cart Processing
- 5.2 Security Requirements
- 5.3 Software Quality Attributes
 - 5.3.1 Maintainability
 - 5.3.2 Portability
 - 5.3.3 Flexibility

- 6. Other Requirements
 - 6.1 Responsive Design
- 7. Glossary

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

This document's goal is to provide a thorough overview of the E-Bookstore Website. It will outline the system's characteristics, goals, and interfaces as well as what it will perform, the limitations it must work under, and how it will respond to outside stimuli. This document is intended to our instructor Albert Ozkohen and group members.

1.2 Intended Audience and Reading Suggestions

Instructor:

The instructor can know the main goals of the project, what it includes, and how well it meets the requirements with this document. Detailed information are on sections Purpose, Project Scope, and Requirements Overview.

Group Members:

Each group member should understand different parts of the project to work well together. They can read 1.1 - Purpose to learn the project's goals, 1.3- Project Scope to see what it includes, and 3- Requirements Overview to understand the features and limits. This will help the team work smoothly.

1.3 Project Scope

This software system will be a Bookstore website. The system will be designed to create an enhanced experience for users who wants to buy books. This system will allow users to search by book and author names using a search bar, filter the books by genre and language and access their purchase history by creating an account.

2. Description

2.1 Product Perspective

This project is developed as an independent application. It will be built with potential for future integration with payment systems.

2.2 Product Features

The application offers an organized main page that shows a grid of e-books and a list for access to categories. A bar is placed for searching for e-books by title or author. The e-books can be added to cart. The application also comprises user functionality, such as register, log in and personal information showing which books a user has purchased. Users can engage with the community with reviews and ratings, helping other customers.

2.3 User Classes and Characteristics

The e-bookstore will have two primary user classes: users and admins. Users will browse and purchase e-books, with an interface designed to be simple and enjoyable. Admins will manage the platform, using their higher technical skills and privileges to perform tasks like content management, user account administration.

2.4 Operating Environment

The software is intended to run on a VPS/VDS host machine in the future. But it will operate on localhost in the time being.

2.5 Design and Implementation Constraints

The project must be complete by the end of 2024-Fall semester. We will use specific technologies and tools, such as MySQL for database management and a programming language like PHP. Programming standards will be followed to ensure maintenance and flexibility.

2.6 User Documentation

A tutorial file with images and a readme file as text will be delivered along with the project.

2.7 Assumptions and Dependencies

The website will be accessed through a modern browser that is HTML5 compatible. The website will be hosted on localhost, with single user access for testing and development . Apache and mySQL for backend and database management is required for a local server environment. PHP is needed for server backend logic. For the frontend, other libraries and frameworks are implemented on top of the CSS-HTML-JS foundation.

3. System Features

As the project is about getting a book from a book store, the main functions are searching for the intended book and filtering the results to make it easier to find the intended book, after finding the book there are sum sub-functions such as add to wish list(read later), purchase the book (to get the book added to my-book list) and so on that will be discussed below.

3.1 Search And Filter:

3.1.1 Description and Priority

Finding the right or intended book is the highest priority feature, to implement that there is a necessity for a search bar and Filters to be applied, there for a search bar in the home page and a filter will be the number 9 priority, Search will be by entering the name of the book or the name of the author, while Filtering will be by Date of release(year only), Number of pages, Language of the book, Price And genre.

3.1.2 Stimulus/Response Sequences

When the users opens the home page, the name of the book or the name of the author should be inserted to show the results of the intended book, after that, the user can apply filters(not necessarily) according to the information they have about this book, eg. If they know the year of release, language , genre and/or any other information that will filter the unneeded results and make it easier for the users to find the book they intend.

If the name of the book/author entered is not in the database, a message will be delivered to the users showing that the book you are searching for is not available yet at our e-bookstore or the name of the book/author entered incorrectly, also if the there was some results after entering a book/author name and applying some filters that filtered all the results a message will be shown to the users saying that “the book you are looking for is not at our e-bookstore or you might applied some wrong filters”.

3.1.3 Functional Requirements

1. An organized database, each book will be associated with the author name if found, or the writer name will be unknown, Tables for Some Filtering criteria (Language of the book, genre and author) and coulombs for some Filtering criteria (Number of pages, Date of release and Price).
2. Applying A search method that will be discussed with the team members at the phase of Implementing the Database.

3.2 Add to Wishlist:

3.2.1 Description and Priority

After finding the intended book, the users might want not to start reading it directly, or may want to process the payment later especially if it was a paid book, Because of that adding a wish list is likely important as adding a my-book list, the system may work without this feature, but to make the system user friendly, adding wishlist is a medium Priority feature that has to be considered.

3.2.2 Stimulus/Response Sequence

When finding the intended book, an option when clicking on the book as “add to my wishlist” or similar syntax will be shown for the users, when clicking on “add to my wishlist” the book will be added to that list and from there, you can decide if you want to purchase that book, leave it for later, or remove it from the list if you are no longer interested about it.

3.3 Log in /Sign up

3.3.1 Description and Priority

If the users have a list of books added to them wishlist or have purchased some books, to protect their information from getting lost, or to access their account from other devices they must have an account that saves all this information, that is why it is a must to have a login/signup feature as a high priority of number 7.

3.3.2 Stimulus/Response Sequence

When the users tries to add some book to wish list or to basket, if they are not already signed in; they will be asked to sign in or sign if they don't have an account, an information will be asked for user like name, sure name, email address and password, After the session ends the user may sign out of the device to sign in with deftest account or for ending the session.

3.3.3 Functional Requirements

Authentication: Validate users credentials against stored data so if the username and password matches the user will have access to the account, otherwise, a syntax showing “password or email address is not correct”.

3.4 Add to basket & my books

3.4.1 Description and Priority

If the users want to get book they found, or a book they already added to them wishlist, they can add it to their basket, once bought, it will be automatically added to my-books list, this is a high priority function of number 7, especially if some books are paid and to make it easier to process the payment of many books at the once.

3.4.2 Stimulus/Response Sequence

When finding the intended book, the user will add it to their basket (they must be already signed in to do so), then, after adding all the books but at least one, they can process a payment, when payment is processed, the books will be automatically added to my-books list relevant with the account signed in with

4. External Interface Requirements

4.1 User Interfaces

The application will serve as a friendly UI for enthusiastic readers to find, check, and purchase books conveniently.

The application is available for use as soon as users log in with their username and password. Once they are logged in, the users will be automatically redirected to the home page full of popular books, categories, and recommended books. The oversized search box in the center of the page will enable them to search by book title or author's name.

All a user needs to do is to filter and sort books via the each criteria such as genre, author, language, and popularity. For every book, there will be a page, which gives a detailed description of its information, including the author, language, genre, and popularity. Moreover, users will have a buyer where they will be able to choose the books to be added to the cart, place a rating, and complete the shopping.

Standard Buttons and Functions:

Navigation Bar: Quick access to homepage, search, categories and cart.

Help Button: An area where users can get help while using the basic features of the application.

Error Messages: Standard error messages that direct users to incorrect login information or invalid search results.

4.2 Hardware Interfaces

The application will be supported by desktops and mobile devices with internet connection. Design adaptability across web browsers will be ensured.

4.3 Software Interfaces

The main interfaces of this application will involve the use of code interactively working with PHP, MySQL, JavaScript, HTML, and CSS. PHP will operate the backend tasks including user authentication, search algorithms, and database calls. A MySQL database shall store data such as books, users, and transactions, and relations between them, with initial data imported from Excel. Dynamic uploads of content and user interactions are to be managed by JavaScript while AJAX operations will ensure that the interface is updated instantaneously for real-time data updates. Also, the MySQL API will provide CRUD (Create, Read, Update, Delete) operations to read and write the data about the books and users.

4.4 Communications Interfaces

Communications between the frontend and the backend would be HTTP/HTTPS. There will be continuous updating between the MySQL database and the frontend in order to allow users to always get real time information such as book statuses and baskets; therefore, at any given time, updating data will always be available to the users. This data synchronization will be provided by AJAX and other similar tools. All these operations and data access will require internet connection.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

5.1.1

Page Load Time: To improve user experience, the homepage and product pages should load within 3 seconds or less. This speed is crucial for encouraging users to stay on the site and navigate smoothly.

5.1.2

Search Performance: To provide users with fast and accurate results, the search function should respond within 4 seconds. This time frame helps users find books quickly, even when using various filters.

5.1.3

Cart Processing: Adding to the cart, removing items should be updated in real-time (within 0.5 seconds). This ensures users can complete their shopping experience securely and seamlessly.

5.2 Security Requirements

When the project goes live, data encryption methods will be employed at the database level to protect user data.

5.3 Software Quality Attributes

5.3.1

Maintainability: The platform will have a codebase that allows for easy updates and the rapid addition of new features

5.3.2

Portability: The system will be designed to be easily transferable to different server infrastructures and operating systems. This feature will facilitate quick migration to a more robust infrastructure if needed.

5.3.3

Flexibility: The software should have a flexible architecture that allows easy integration of features.

6. Other Requirements

6.1

Responsive design: The website UI will adapt to the screen size and resolution of the client device's screen.

7: Glossary

- **Admin:** A user with special permissions to manage website content and user accounts.
- **Authentication:** The process of confirming a user's identity.
- **Cart:** A feature that lets users save items they plan to buy.
- **CRUD:** Basic database operations: Create, Read, Update, Delete.
- **Encryption:** A method of securing data by converting it into code.

- **Frontend:** The part of the website users interact with, made with HTML, CSS, and JavaScript.
- **Genre:** A category of books (e.g., Fiction, Science) to help narrow down searches.
- **HTTPS:** A secure internet protocol that protects data during transfer.
- **Interface:** A system's connection point for user interactions or data exchange.
- **JavaScript:** A language for adding interactive features to a website.
- **Localhost:** Refers to the local computer used for testing the website.
- **MySQL:** A database system that stores and organizes data.
- **PHP:** A backend language for managing server-side logic.
- **Portability:** The system's ability to be moved to different servers easily.
- **Responsive Design:** Ensures the website looks good on all screen sizes.
- **Search Bar:** A tool where users type keywords to find books.
- **User:** Anyone using the site to search, view, or buy books.
- **VPS / VDS:** Types of hosting servers used for running a website.