Software Requirements Specification (SRS)

Full Stack .NET eBook Platform Web Application

1. Introduction:

eBook Platform for Digital Reading

1.1 Purpose

This document provides a comprehensive Software Requirements Specification (SRS) for the development of a .NET Full-Stack eBook Platform Web Application. The system aims to facilitate online reading by allowing users to browse, purchase, download, and read digital books efficiently.

1.2 Scope

The eBook platform will be developed using ASP.NET Core MVC 9 (backend) with Entity Framework Core and a SQL Server database. The system will provide features such as:

- User registration and authentication
- eBook listing and search
- Shopping cart and checkout
- Library management
- Payment integration
- Reader interface
- Admin panel for content management

1.3 Intended Audience and Usage

This document is intended for developers, testers, project managers, and stakeholders involved in the development and deployment of the system.

1.4 Definitions and Acronyms

- ASP.NET Core MVC: A cross-platform framework for building web applications.
- Entity Framework Core: Object-Relational Mapping (ORM) framework.
- SQL Server: A relational database management system.
- Session: Temporary server-side storage for cart management.
- Identity: Authentication framework for user management.

2. Overall Description

2.1 Product Perspective

The eBook platform web application is a standalone system designed to manage an online digital bookstore. It will have both customer-facing interfaces for reading and purchasing, as well as admin-facing interfaces for content management.

2.2 Product Features

1. **User Management**
   - User registration and login (Identity authentication)
   - Profile management
   - Reading preferences and history
2. **Book Management**
   - Categories and genres
   - Author profiles
   - Book search and filtering
3. **Shopping Cart & Checkout**
   - Add, remove, and update items in the cart
   - Session-based cart management
   - Checkout process with payment integration
4. **Library Management**
   - Access to purchased books
   - Reading progress tracking
   - Bookmarks and annotations
5. **Reader Interface**
   - In-browser eBook reader
   - Customizable reading experience
   - Offline reading capabilities
6. **Admin Dashboard**
   - Manage users, books, and orders
   - Content curation and publishing
   - Sales analytics
7. **Security & Performance**
   - Secure application architecture
   - DRM for eBook content
   - Scalability considerations

2.3 Assumptions and Dependencies

- The system assumes a stable internet connection for initial book downloads.
- Third-party payment gateways will be used.
- Digital Rights Management (DRM) system for protecting content.

3. Functional Requirements

3.1 User Management

- Users should be able to register and log in securely.
- Passwords should be stored using encryption.
- Users can manage their profile and reading preferences.

3.2 Book Listing & Search

- Users can browse books by genre, price, author, and publication date.
- The system should support keyword-based search with autocomplete.
- Book previews should be available for sample reading.

3.3 Shopping Cart & Checkout

- Users can add/remove books from the cart.
- Session management for cart persistence.
- The checkout process should integrate a payment gateway.

3.4 Library & Reading Experience

- Users should have access to their purchased books in their personal library.
- The system should track reading progress across devices.
- Users should be able to bookmark pages and make annotations.

3.5 Admin Panel

- Only authorized admins should have access.
- Admins should be able to add/edit/delete books and manage content.
- Analytics dashboard for sales and reader engagement.

4. Non-Functional Requirements

4.1 Performance Requirements

- The system should handle at least 500 concurrent users.
- Page loading times should not exceed 2 seconds.
- Book downloads should be optimized for speed.

4.2 Security Requirements

- All sensitive data should be encrypted.
- Users should be authenticated using Identity framework.
- DRM protection for eBook content to prevent unauthorized distribution.

4.3 Usability Requirements

- The UI should be mobile-friendly and responsive.
- The reading interface should be comfortable for extended reading.
- Navigation should be intuitive with minimal clicks to access books.

4.4 Scalability Requirements

- The system should be scalable to support increasing book catalog and user traffic.
- Storage considerations for growing eBook library.

5. System Design

5.1 Architecture

- Frontend: ASP.NET Core MVC with Razor Views
- Backend: ASP.NET Core Controllers
- Database: SQL Server with Entity Framework Core
- Authentication: Identity framework
- Hosting: IIS

5.2 Key Controllers and Actions

Controller	Action	Method	Description
HomeController	Index	GET	Display featured books
BooksController	Browse	GET	Browse book catalog
BooksController	Details	GET	View book details
CartController	AddToCart	POST	Add book to cart
CartController	ViewCart	GET	View shopping cart
AccountController	Register	POST	Register new user
AccountController	Login	POST	Authenticate user
LibraryController	MyBooks	GET	View purchased books
ReaderController	Read	GET	Open eBook reader
AdminController	Dashboard	GET	Admin control panel

6. Appendix

- Technology Stack: ASP.NET Core MVC 9, Entity Framework Core, SQL Server
- Development Tools: Visual Studio, Git
- Third-Party Integrations: Payment gateway, PDF/ePub rendering

This document serves as a foundation for the development and implementation of the eBook platform system.