

Software Requirements Specification (SRS) for E-Commerce Web Application

Prepared by .

Mohamed Arfa Abdelrady 202220296

Ahmed Ali Abdelrady 202220295

Instructor: DR Mohamed Reda

Course : Software Engineering

Teaching Assistant :Mohamed karam

Table of Contents

- 1. **Introduction**
 - 1.1 Purpose
 - 1.2 Scope
 - 1.3 Definitions, Acronyms, and Abbreviations
 - 1.4 References
 - 1.5 Overview
 - 2. **System Overview**
 - 3. **Functional Requirements**
 - 3.1 User Registration and Authentication
 - 3.2 Product Catalog
 - 3.3 Shopping Cart
 - 3.4 Checkout and Payment
 - 3.5 Order Management
 - 3.6 Search Functionality
 - 3.7 User Profile Management
 - 3.8 Admin Panel
 - 3.9 Notifications and Alerts
 - 4. **Non-Functional Requirements**
 - 4.1 Performance
 - 4.2 Security
 - 4.3 Scalability
 - 4.4 Usability
 - 4.5 Availability
 - 4.6 Backup and Recovery
 - 5. **System Architecture and Design**
 - 5.1 Architecture Overview
 - 5.2 Technology Stack
 - 5.3 Database Design
 - 5.4 User Interface Design
 - 5.5 External Interfaces
 - 6. **Appendices**
-

1. Introduction

1.1 Purpose

This Software Requirements Specification (SRS) document provides a detailed description of the functional and non-functional requirements for the E-Commerce Web Application. The document aims to outline the necessary features, design specifications, and constraints that must be met during the development process.

1.2 Scope

The E-Commerce Web Application allows users to browse products, add them to the shopping cart, and complete transactions online. It includes functionalities such as user authentication,

order management, product management, and admin controls. This web application will cater to both customers and administrators. Customers will use it to shop for products, while administrators will manage product listings, users, and orders.

1.3 Definitions, Acronyms, and Abbreviations

- **User:** A customer of the e-commerce platform.
- **Admin:** A user with elevated privileges who manages the platform.
- **Shopping Cart:** A temporary list of items a user intends to purchase.
- **SRS:** Software Requirements Specification.

1.4 References

- **IEEE 830-1998:** IEEE Recommended Practice for Software Requirements Specifications.
- **Web Content Accessibility Guidelines (WCAG).**

1.5 Overview

This document outlines the full specifications for an E-Commerce Web Application. The sections will cover detailed functional requirements for various user roles, including customers and administrators. Non-functional requirements such as performance, security, and scalability will also be described.

2. System Overview

The E-Commerce Web Application is a multi-platform solution that allows users to interact with the website for browsing products, placing orders, and managing personal information. It will include various roles:

- **Customer Role:** Can browse, search, add products to the cart, place orders, view order history, and manage their profile.
- **Admin Role:** Can manage users, products, categories, orders, and site settings.

Key features include product listings, search and filtering, shopping cart, user authentication, and an integrated payment gateway.

3. Functional Requirements

3.1 User Registration and Authentication

- Users must be able to register with their email and a password.
- User authentication should support login via email/password.
- Users must be able to reset their password via email.
- Admins should have the ability to create, update, or delete user accounts.

3.2 Product Catalog

- The system will display products by categories, with filtering options (price, brand, rating, etc.).
- Each product will include details such as name, description, price, images, and availability.
- Users can view product details by clicking on product names or images.

3.3 Shopping Cart

- Users can add/remove products from their cart.
- The cart will update automatically to reflect the quantity and price of items.
- Users can view a summary of the items in their cart, including subtotal, taxes, and shipping charges.

3.4 Checkout and Payment

- Users will be able to proceed to checkout after reviewing their cart.
- The system will require users to enter shipping information (name, address, contact number).
- Users will be able to select payment methods (credit card, PayPal, etc.).
- The system should provide an order summary before confirming the purchase.
- Payment gateway integration will be provided for secure transactions.
- Users will receive an order confirmation after the payment is processed.

3.5 Order Management

- Users will be able to view their past orders, including status (processing, shipped, delivered).
- Admins can view and manage customer orders, update statuses, and process refunds or cancellations.

3.6 Search Functionality

- Users can search for products using keywords or product attributes.
- The system will support advanced filters, including product type, price range, and brand.

3.7 User Profile Management

- Users can update their personal information, including name, email, password, and shipping address.
- Users can view their order history and save favorite items.

3.8 Admin Panel

- Admins will have access to a dashboard to manage products, users, and orders.
- Admins can add, update, or delete products.
- Admins can manage categories and subcategories of products.

3.9 Notifications and Alerts

- Users will receive notifications about order status (shipped, delivered, etc.).
 - Admins will receive alerts about low stock or user activity.
 - Email notifications will be sent for account registration, password reset, and order confirmations.
-

4. Non-Functional Requirements

4.1 Performance

- The application must support 10,000 simultaneous users without performance degradation.
- Page load times should be under 2 seconds for an optimal user experience.

4.2 Security

- User data (passwords, payment info) must be encrypted using industry-standard encryption techniques (e.g., SSL/TLS).
- Admin functions should be protected by multi-factor authentication (MFA).
- Sensitive information, such as credit card numbers, should never be stored on the server.

4.3 Scalability

- The system should be scalable to support increased traffic, especially during sales or promotions.
- The database should be able to handle high volumes of transactions efficiently.

4.4 Usability

- The user interface should be intuitive and responsive on both desktop and mobile devices.
- The application should be accessible, meeting WCAG 2.1 standards.

4.5 Availability

- The application must be available 99.9% of the time, with minimal downtime for maintenance.
- Backup systems should be in place for database redundancy.

4.6 Backup and Recovery

- Regular backups of data (orders, user profiles, inventory) should be taken every 24 hours.
 - The application should have a disaster recovery plan to restore the system in case of failure.
-

5. System Architecture and Design

5.1 Architecture Overview

The application will follow a **three-tier architecture**:

1. **Presentation Layer:** Web interface that interacts with users.
2. **Business Logic Layer:** Handles all processing logic, such as order management, user management, and payment processing.
3. **Data Layer:** The database and data access layer, responsible for storing user and product data.

5.2 Technology Stack

- **Frontend:** React.js, HTML5, CSS3, Bootstrap
- **Backend:** Node.js with Express.js
- **Database:** MySQL or PostgreSQL
- **Payment Gateway:** Stripe, PayPal API
- **Authentication:** JWT (JSON Web Token) for secure token-based authentication
- **Hosting:** AWS, Azure, or similar cloud platforms

5.3 Database Design

- **Users Table:** user_id, email, password_hash, shipping_address, order_history
- **Products Table:** product_id, name, description, price, stock_quantity, image_url
- **Orders Table:** order_id, user_id, product_ids, total_price, shipping_address, payment_status
- **Categories Table:** category_id, name, parent_category_id

5.4 User Interface Design

- The web interface should be responsive and mobile-friendly.
- The homepage will feature product categories, popular products, and user login options.
- Users should be able to easily add items to the shopping cart and view the cart.
- The checkout page will have a clean, step-by-step process for order completion.

5.5 External Interfaces

- **Payment Gateway API:** For processing payments securely.
 - **Email Service:** For sending account registration, order confirmation, and password reset emails.
-

6. Appendices

A. Glossary of Terms

- **SKU:** Stock Keeping Unit.
 - **API:** Application Programming Interface.
 - **JWT:** JSON Web Token, used for secure user authentication.
-

This concludes the Software Requirements Specification (SRS) document for the E-Commerce Web Application. The requirements outlined in this document will guide the development and implementation of the system, ensuring all key features are implemented to meet user needs.