

Software Requirements Specification (SRS) for Demoblaze Website

1. Introduction

The purpose of this document is to specify the software requirements for the Demoblaze E-commerce Website. It defines both functional and non-functional requirements to ensure that the system meets user expectations and business needs.

2. Overall Description

Demoblaze is an online shopping platform that allows users to view products, add them to a cart, and make purchases. It is a browser-based application designed for simplicity and usability.

3. Functional Requirements (User Interface)

FR-UI-1: User Account Management

FR-UI-1.1 (Signup): The system shall allow a new user to register via the "Sign up" modal using a unique username and a password.

FR-UI-1.2 (Login): The system shall allow a registered user to log in via the "Log in" modal.

FR-UI-1.3 (Logout): The system shall allow a logged-in user to log out. The "Log out" link must be visible only after logging in.

FR-UI-1.4 (Welcome Message): Upon successful login, the header must display a "Welcome [username]" message.

FR-UI-2: Product Catalog

FR-UI-2.1 (View All Products): The homepage must display a paginated list of all available products.

FR-UI-2.2 (Filter by Category): The user must be able to filter the product list by selecting a category (Phones, Laptops, Monitors).

FR-UI-2.3 (Pagination): The user must be able to navigate through the product list using "Previous" and "Next" buttons.

FR-UI-2.4 (View Product Details): Clicking a product must navigate the user to its dedicated details page, showing its image, title, price, and description.

FR-UI-3: Shopping Cart

FR-UI-3.1 (Add to Cart): The user must be able to add a product to the cart from the product detail page. A confirmation alert ("Product added") must be displayed.

FR-UI-3.2 (View Cart): The user must be able to view the contents of their cart by clicking

the "Cart" link in the header.

FR-UI-3.3 (Cart Details): The cart page must display a table of added items with their picture, title, price, and a "Delete" option.

FR-UI-3.4 (Delete from Cart): The user must be able to remove an item from the cart.

FR-UI-3.5 (Total Price): The cart page must display the correct total price of all items.

FR-UI-4: Checkout Process

FR-UI-4.1 (Place Order): The user must be able to initiate checkout by clicking "Place Order" on the cart page.

FR-UI-4.2 (Order Form): The "Place Order" modal must contain fields for Name, Country, City, Credit Card, Month, and Year.

FR-UI-4.3 (Purchase Confirmation): Upon submitting the order form, a "Thank you for your purchase!" confirmation modal must appear with the purchase details.

4. Functional Requirements (API)

FR-API-1 (Signup): POST /signup

Accepts: JSON with username, password.

Success Response: HTTP 200 OK.

Error Response: HTTP 200 OK with {"errorMessage": "User already exists."} for duplicate users.

FR-API-2 (Login): POST /login

Accepts: JSON with username, password.

Success Response: HTTP 200 OK with {"Auth_token": "..."}.

Error Response: HTTP 200 OK with {"errorMessage": "Wrong password."} or {"errorMessage": "User does not exist."}.

FR-API-3 (Get All Products): GET /entries

Success Response: HTTP 200 OK with a JSON body containing an "Items" array. Each item must have id, title, price, img, etc.

FR-API-4 (Get Products by Category): POST /bycat

Accepts: JSON with {"cat": "phone"} (or laptop, monitor).

Success Response: HTTP 200 OK with a filtered "Items" array.

FR-API-5 (Get Product Details): POST /view

Accepts: JSON with {"id": "<product-id>"}

Success Response: HTTP 200 OK with a JSON object of the specific product.

FR-API-6 (Add to Cart): POST /addtocart

Accepts: JSON with id, cookie, flag.
Success Response: HTTP 200 OK.

FR-API-7 (View Cart): POST /viewcart

Accepts: JSON with cookie, flag.

Success Response: HTTP 200 OK with an "Items" array of products in the cart.

FR-API-8 (Delete from Cart): POST /deletcart

Accepts: JSON with {"id": "<cart-item-id>"}

Success Response: HTTP 200 OK.

FR-API-9 (Place Order): POST /order

Accepts: JSON with cart details and customer info.

Success Response: HTTP 200 OK.

5. Non-Functional Requirements

NFR-1 (Performance): All pages must load within 4 seconds. API responses must be under 800ms.

NFR-2 (Security): The application must use HTTPS. Passwords must not be stored in plain text.

NFR-3 (Usability): Navigation must be intuitive, and workflows (e.g., checkout) must be simple to complete.

6. Future Enhancements

- Integration with real payment gateways (e.g., PayPal, Visa).
- Add user order history and invoice generation.
- Implement an admin dashboard for product and order management.
- Enable product reviews and ratings from users.
- Support multi-language and multi-currency options.

7. Appendices

- Appendix A: Use Case Diagram (Login, Product Browsing, Checkout).
- Appendix B: Sample UI Mockups for Home, Cart, and Checkout pages.
- Appendix C: API Endpoint Summary Table.
- Appendix D: Glossary of terms used in the SRS document.