**Functional Requirements Specification (FRS)**

**Project Name: Dapperly – B2B eCommerce Platform**

**Type: B2B eCommerce Platform**

**Tech Stack: React, HTML5, CSS, C#, .NET Core, Web API, SQL-based Database**

**Version: 1.0**

**Date: May 2, 2025**

**Prepared by: Somanath Kasta**

**1. Introduction**

Dapperly is a B2B eCommerce web application designed to facilitate product purchases between businesses. It provides a platform where vendors can showcase their products and buyers can search, browse, and order them in bulk.

**2. Target Audience**

* **Vendors/Sellers:** Businesses looking to sell products wholesale to other businesses.
* **Buyers/Customers:** Businesses (retailers, distributors) seeking bulk purchases.
* **Admins:** Platform managers who oversee user accounts, products, and transactions.

**3. User Roles & Authentication**

**3.1 User Types**

* **Buyer**
* **Vendor**
* **Admin**

**3.2 Authentication**

* User registration (vendor/buyer)
* Secure login/logout (JWT or cookie-based)

**4. Functional Modules**

**4.1 Product Management (Vendor Role)**

* Add/Edit/Delete products
* Upload product images
* Set product descriptions, prices, categories, and stock levels
* Bulk upload via CSV/Excel

**4.2 Product Search & Filters (All Users)**

* Keyword search
* Filters:
  + Category
  + Price Range
  + Vendor
  + Rating
  + Availability
* Sort options (price, popularity, newest, etc.)

**4.3 Product Details Page**

* Product name, images, and description
* Pricing, available stock, vendor info
* Customer ratings and reviews
* Add to cart button

**4.4 Shopping Cart & Checkout**

* Add/remove items from cart
* Quantity management
* Save cart for later
* Checkout process:
  + Billing/Shipping info
  + Order summary
  + Dummy payment gateway
  + Order confirmation screen

**4.5 Dummy Payment Gateway Integration**

* Simulates payment (no real transactions)
* Payment success/failure mock
* Post-payment order confirmation logic

**4.6 Search & Filter Functionality (Global)**

* Full-text search across products

**5. Technical Requirements**

**5.1 Frontend**

* **ReactJS** with JSX and Hooks
* Responsive design using **HTML5/CSS**
* Axios for API calls

**5.2 Backend**

* **C# .NET Core Web API**
* RESTful API endpoints for all modules
* Entity Framework

**5.3 Database**

* SQL Server
* Tables: Users, Products, Orders, Payments, Cart etc.

**5.4 Payment Gateway (Dummy)**

* Simulated payment endpoint (/api/payment/process)
* Status: Success, Failed, Timeout
* No integration with real banks

**5.5 Security**

* HTTPS enforced
* JWT Authentication

**6. Non-Functional Requirements**

* **Performance:** Fast response time (avg < 2s)
* **Scalability:** Scalable backend with modular API architecture
* **Usability:** Intuitive UI for business users
* **Maintainability:** Code structured with SOLID principles
* **Availability:** Uptime target: 99.9% (host-dependent)

**Conclusion**

The "Dapperly" B2B e-commerce application aims to deliver a seamless experience for businesses to list and sell products, manage inventory, and track orders. With a responsive frontend built using React, a backend API powered by .NET Core, and a simple dummy payment gateway for simulation, the platform will serve as an ideal solution for business transactions at scale.