



High Level Design Document

Introduction

This High Level Design (HLD) document outlines the architecture and core components for **ShopEase - E-commerce Product Catalog**. The project aims to deliver a user-friendly, responsive product catalog with advanced filtering, sorting, and dynamic rendering, built using React and Tailwind CSS.

1. System Architecture Overview

Architecture Description:

ShopEase is a single-page frontend application structured around modular React components. The system fetches product data (mocked or via API), manages state for filtering/sorting, and renders responsive product cards.

Module/Component	Role/Responsibility
App Shell	Root component, layout, and routing (if needed)
Product Catalog	Fetches and displays product list
Product Card	Renders individual product details
Filter Panel	UI for filtering products (category, price, etc.)
Sort Controls	UI for sorting products (price, popularity, etc.)
State Management	Manages filter/sort state and product data
Styles (Tailwind/CSS)	Provides responsive and modern UI styling

2. Component Interactions

Source Component	Target Component	Interaction Description
App Shell	Product Catalog	Renders and passes state/props
Filter Panel	State Management	Updates filter state on user input
Sort Controls	State Management	Updates sort state on user input
State Management	Product Catalog	Supplies filtered/sorted product data
Product Catalog	Product Card	Maps product data to individual cards

Sequence Flow:

- User interacts with Filter Panel or Sort Controls.
 - State Management updates filter/sort state.
 - Product Catalog receives updated data and re-renders Product Cards.
-



3. Data Flow Overview

Data Source	Flow Direction	Data Type/Description
Product Data (API/mock)	→ State Management	Raw product list
State Management	→ Product Catalog	Filtered and sorted product list
User Input (UI)	→ State Management	Filter/sort parameters
Product Catalog	→ Product Card	Individual product details

4. Technology Stack

Layer/Area	Technology/Framework
UI Framework	React (Functional Components)
Styling	Tailwind CSS, advanced CSS
Language	JavaScript (ES6+), HTML
State Management	React useState/useContext
Data Source	Mock data or REST API
Tooling	Node.js, npm/yarn

5. Scalability & Reliability

- **Scalability:**
 - Modular React components enable easy extension (e.g., new filters, pagination).
 - Stateless frontend; can integrate with scalable backend APIs.
- **Reliability:**
 - UI state is managed locally for responsiveness.
 - Responsive design ensures usability across devices.
- **Security:**
 - No sensitive data handled; standard frontend security practices apply.

End of Document