COLLEGE CODE : 9530

COLLEGE NAME : ST.Mother Theresa Engineering College

NM-ID ROLL NO :7E218C3BFA487AFB48C4D7838988A650

DATE : 1 3 /10/2025

Completed the project named phase 5

PROJECT NAME : PRODUCT CATALOG WITH FILTERS

SUBMITTED BY,

NAME : R.Nithish kumar

MOBILE NO: 8015958116

IBM-FE-Product Catalog with Filters

Final Demo Walkthrough

Thisproject demonstrates an interactive and responsive Product Catalog web application that allows users to view, filter, and search products based on various criteria such as category, price range, and rating. The purpose of the demo is to showcase frontend development skills using modern web technologies and clean UI/UX principles.

The home page displays a grid of products fetched from an API or local JSON file. Filters and search bars dynamically update the product list without reloading the page. The user interface is fully responsive and optimized for both desktop and mobile screens.

Key Features:

- Product filtering by category, price, and rating
- Dynamic search functionality
- Responsive and clean design
- Lightweight frontend-only application
- Can be deployed easily on platforms like Netlify or GitHub Pages

Project Report

The 'Product Catalog withFilters' is designed as a modern frontend web application built primarily using HTML, CSS, and JavaScript (or React). It focuses on providing an easy-to-navigate catalog interface for users to explore items visually and apply dynamic filters to narrow down their search results.

The application is structured with modular components, separating product data, UI logic, and event handling. It uses CSS flexbox and grid layouts for better visual alignment, and JavaScript for DOM manipulation and filter control. If built with React, functional components and hooks are utilized for state management.

The data source can either be static JSON files or fetched from a mock API (such as FakeStoreAPI). This helps simulate a real-world e-commerce front-end interface without requiring a backend server.

Screenshots / API Documentation

ScreenshotsSection(to beadded by student):
→ Screenshot 1: Home page with all products displayed
→ Screenshot 2: Filter applied (e.g., Category - Electronics)
→ Screenshot 3: Search bar usage example
→ Screenshot 4: Responsive layout preview (Mobile View)

API Documentation:

If the application uses a live or mock API, the data structure follows the below format:

```
{ "id": 1,
"title": "Product Name",
"price": 999.99,
"category": "Electronics",
"rating": 4.5,
"image": "image_url_here" }
```

Challenges & Solutions

1.Challenge:Handling multiplefilterstogether dynamically.
Solution: Implemented combined filtering logic using array methods and conditional checks. 2. Challenge: Maintaining responsiveness on all devices. Solution: Used CSS media queries and grid system for adaptable layout. 3. Challenge: Optimizing performance when dealing with many products. Solution: Used lazy loading for images and efficient rendering with minimal DOM updates.

GitHub README & Setup Guide

1. Clone the repository:
git clone https://github.com/yourusername/product-catalog-with-filters
2. Navigate to the project folder:
cd product-catalog-with-filters
3. Open index.html in browser (for plain HTML version)
or run `npm start` (for React version).
4. Deploy to GitHub Pages or Netlify for online access.

Final Submission:

• GitHub Repository Link: ____________________________ •
Deployed Link: ____________________________
Submitted by:
• College Name: ____________________________
Date: ____________________________