MIT Art, Design and Technology University
MIT School of Computing, Pune

Department of Information Technology

Lab Manual

Practical - CPAD

Class - T.Y. (SEM-VI), SMAD

Batch - SMAD-TY

Mr.

Vedant Prasad Parulekar

A.Y. 2024 – 2025 (SEM-II)

Experiment No.2

Problem Statement:

To develop a TechKeeda mobile/web app that brings together tech news and tutorials through a blog section, alongside a fully functional tech e-commerce platform — all in one unified application experience.

Objective:

- To create a responsive app interface where users can:
  - Browse tech blog articles (news, reviews, tutorials)
  - Shop for tech gadgets, accessories, and devices
- To implement an intuitive product display and cart management system
- To provide smooth navigation between blog content and product purchases within the same app
- To ensure the app is scalable and user-friendly for both desktop and mobile users

TechKeeda's app vision is to bring this integrated content-commerce experience to mobile in a native, fast, and user-centric form. While the web version attracts new users, the app is focused on retention, personalization, and mobile commerce.

The TechKeeda app (based on the same codebase for now) aims to deliver a mobile-first, app-like experience with enhanced interactivity. Whether deployed as a PWA (Progressive Web App) or converted to a native app using React Native, the goal is to bring TechKeeda's hybrid experience directly to mobile devices.

• On mobile, user behavior differs — users prefer bite-sized content, quick browsing, and fast checkout. The app adapts to these needs by offering touch-optimized UI, persistent login sessions, and push notification potential for blog updates or new product launches.

• The cart and product system is tightly integrated with the UI, supporting swipe gestures, native-like modals, and responsive design. Mobile apps can also integrate device-native features like camera for QR scans, saved payment methods, etc.

• Unlike the SEO-driven web version, the app is focused on user retention, personalized recommendations, and frequent engagement.

🔧 Technical Considerations:

•Can be implemented as:

•PWA (Progressive Web App) — lightweight, installable web app

•React Native App — full-fledged mobile app (for Android & iOS)

- Data sync: Cloud storage or Firebase can be used for user login, saved carts, reading history, and product wishlists.

📲 Mobile User Patterns:

- Users on mobile are task-oriented — they want to:
- Read briefly during commute or breaks
- Buy quickly without distractions
- Stay updated via notifications

📈 Strategic Value:

- App allows push notifications, which drive traffic back to the app for:
- New blog drops
- Exclusive app-only offers
- Restocked or recommended products
- Opportunity to implement gamified experiences:
- Points for reading articles
- Rewards for purchases

🎨 Mobile UX Considerations:

- Swipeable product cards
- Bottom navigation bar (thumb-reachable)
- In-app dark mode for reading
- Native animations for smooth transitions

🔐 Enhanced Features Possible in App:

- One-tap Google/Apple login
- Fingerprint-secured checkout
- Personalized feed of blogs and products
- Offline reading of saved articles

Theory:

The TechKeeda app follows a hybrid platform model combining content (blogs) and commerce (tech products). Users can educate themselves through articles and immediately act on purchase intent, creating a content-driven commerce loop.

The application is likely built using React (or a React framework like Vite + TypeScript), with modular components such as:

- HomePage: Showcases featured blogs and highlighted products
- ShopPage: Allows browsing and selection of products
- CartSidebar: Handles cart interaction and purchasing logic
- BlogPage: Displays all tech articles in a user-friendly format

This architecture enhances engagement by allowing users to interact with both informative and transactional elements without leaving the app ecosystem.

Code:

App File

code:

```
import React, { useState } from 'react';

import Header from './components/layout/Header';

import Footer from './components/layout/Footer';
import HomePage from './pages/HomePage';
import BlogPage from './pages/BlogPage';
import ShopPage from './pages/ShopPage';
import BlogDetail from './components/blog/BlogDetail';
import ProductDetail from './components/products/ProductDetail';
import CartSidebar from './components/cart/CartSidebar';
import { CartProvider } from './contexts/CartContext';
import { BlogPost, Product } from './types';

function App() {
  const [activePage, setActivePage] = useState('home');
  const [selectedBlog, setSelectedBlog] = useState<BlogPost | null>(null);
  const [selectedProduct, setSelectedProduct] = useState<Product | null>(null);
  const [showCart, setShowCart] = useState(false);

  const renderContent = () => {
    if (selectedBlog) {
      return (
        <BlogDetail
          blog={selectedBlog}
          onBack={() => setSelectedBlog(null)}
        />
      );
    }

    if (selectedProduct) {
      return (
        <ProductDetail
          product={selectedProduct}
          onBack={() => setSelectedProduct(null)}
        />
      );
    }

    switch (activePage) {
      case 'blog':
        return <BlogPage setSelectedBlog={setSelectedBlog} />;
      case 'shop':
        return <ShopPage setSelectedProduct={setSelectedProduct} />;
      default:
        return (
          <HomePage
            setActivePage={setActivePage}
            setSelectedBlog={setSelectedBlog}
            setSelectedProduct={setSelectedProduct}
          />
        );
    }
  };

  return (
    <CartProvider>
      <div className="min-h-screen flex flex-col bg-gray-50">
        <Header
          setActivePage={setActivePage}
          activePage={activePage}
          setShowCart={setShowCart}
        />

        <main className="flex-grow">
```

```
         {renderContent()}
       </main>

       <Footer />

       <CartSidebar
         isOpen={showCart}
         onClose={() => setShowCart(false)}
       />
     </div>
   </CartProvider>
 );
}

export default App;
```

A. Home page:

code:

```
import React from 'react';
import { blogPosts } from '../data/blogs';
import { products } from '../data/products';
import BlogCard from '../components/blog/BlogCard';
import ProductCard from '../components/products/ProductCard';
import { ArrowRight, Laptop, Cpu, ChevronRight } from 'lucide-react';

interface HomePageProps {
 setActivePage: (page: string) => void;
 setSelectedBlog: (blog: any) => void;
 setSelectedProduct: (product: any) => void;
```

```
}


const HomePage: React.FC<HomePageProps> = ({
  setActivePage,
  setSelectedBlog,
  setSelectedProduct
}) => {
  return (
    <div className="pt-20 bg-gradient-to-b from-gray-50 to-white">
      {/* Hero Section */}
      <section className="py-16 md:py-24">
        <div className="container mx-auto px-4 md:px-6">
          <div className="flex flex-col md:flex-row items-center">
            <div className="md:w-1/2 mb-10 md:mb-0 md:pr-12">
              <h1 className="text-3xl md:text-4xl lg:text-5xl font-bold text-gray-900 mb-4 leading-tight">
                Your Ultimate Tech <br />
                <span className="text-blue-700">Resource Hub</span>
              </h1>
              <p className="text-lg text-gray-600 mb-8 leading-relaxed">
                Stay informed with the latest tech insights and shop premium gadgets. Discover expert articles and cutting-edge products all in one place.
              </p>
              <div className="flex flex-col sm:flex-row space-y-3 sm:space-y-0 sm:space-x-4">
                <button
                  onClick={() => setActivePage('blog')}
                  className="bg-blue-700 text-white px-6 py-3 rounded-md hover:bg-blue-800 transition-colors flex items-center"
                >
                  Explore Blogs
                  <ArrowRight className="ml-2 w-5 h-5" />
                </button>
                <button
                  onClick={() => setActivePage('shop')}
                  className="border border-blue-700 text-blue-700 px-6 py-3 rounded-md hover:bg-blue-50 transition-
colors"
                >
                  Shop Products
                </button>
              </div>
            </div>
            <div className="md:w-1/2">
              <img
                src="https://images.pexels.com/photos/1181244/pexels-photo-1181244.jpeg?
auto=compress&cs=tinysrgb&w=1260&h=750&dpr=2"
                alt="Tech devices on desk"
                className="rounded-lg shadow-xl"
              />
            </div>
          </div>
        </div>
      </section>

      {/* Featured Blogs Section */}
      <section className="py-16 bg-gray-50">
        <div className="container mx-auto px-4 md:px-6">
          <div className="flex items-center justify-between mb-10">
            <h2 className="text-2xl md:text-3xl font-bold text-gray-900">Featured Articles</h2>
            <button
              onClick={() => setActivePage('blog')}
              className="text-blue-700 hover:text-blue-900 transition-colors flex items-center text-sm font-medium"
            >
              View All
      <ChevronRight className="ml-1 w-4 h-4" />
     </button>
    </div>

    <div className="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-3 gap-8">
     {blogPosts.slice(0, 3).map((blog) => (
      <BlogCard
       key={blog.id}
       blog={blog}
       setSelectedBlog={setSelectedBlog}
      />
     ))}
    </div>
   </div>
</section>

{/* Featured Products Section */}
<section className="py-16">
   <div className="container mx-auto px-4 md:px-6">
    <div className="flex items-center justify-between mb-10">
     <h2 className="text-2xl md:text-3xl font-bold text-gray-900">Top Products</h2>
     <button
      onClick={() => setActivePage('shop')}
      className="text-blue-700 hover:text-blue-900 transition-colors flex items-center text-sm font-medium"
     >
      View All
      <ChevronRight className="ml-1 w-4 h-4" />
     </button>
    </div>
```

```
        <div className="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-3 gap-8">
          {products.slice(0, 3).map((product) => (
            <ProductCard
              key={product.id}
              product={product}
              setSelectedProduct={setSelectedProduct}
            />
          ))}
        </div>
      </div>
    </section>

    {/* Categories Section */}
    <section className="py-16 bg-gray-50">
      <div className="container mx-auto px-4 md:px-6">
        <h2 className="text-2xl md:text-3xl font-bold text-gray-900 mb-10 text-center">
          Browse by Category
        </h2>

        <div className="grid grid-cols-1 sm:grid-cols-2 lg:grid-cols-4 gap-6">
          <div className="bg-white p-6 rounded-lg shadow-sm hover:shadow-md transition-shadow text-center">
            <div className="w-16 h-16 bg-blue-100 rounded-full flex items-center justify-center mx-auto mb-4">
              <Laptop className="w-8 h-8 text-blue-700" />
            </div>
            <h3 className="text-lg font-semibold text-gray-900 mb-2">Computing</h3>
            <p className="text-gray-600 text-sm mb-4">Laptops, Desktops, Components</p>
            <button
              onClick={() => setActivePage('shop')}
              className="text-blue-700 hover:text-blue-900 transition-colors flex items-center justify-center mx-auto text-sm font-medium"
            >
              View Products
              <ChevronRight className="ml-1 w-4 h-4" />
            </button>
          </div>

          <div className="bg-white p-6 rounded-lg shadow-sm hover:shadow-md transition-shadow text-center">
            <div className="w-16 h-16 bg-blue-100 rounded-full flex items-center justify-center mx-auto mb-4">
              <svg xmlns="http://www.w3.org/2000/svg" className="w-8 h-8 text-blue-700" viewBox="0 0 24 24" 
    fill="none" stroke="currentColor" strokeWidth="2" strokeLinecap="round" strokeLinejoin="round">
                <rect x="5" y="2" width="14" height="20" rx="2" ry="2" />
                <path d="M12 18h.01" />
              </svg>
            </div>
            <h3 className="text-lg font-semibold text-gray-900 mb-2">Mobile Devices</h3>
            <p className="text-gray-600 text-sm mb-4">Smartphones, Tablets, Accessories</p>
            <button
              onClick={() => setActivePage('shop')}
              className="text-blue-700 hover:text-blue-900 transition-colors flex items-center justify-center mx-auto 
    text-sm font-medium"
            >
              View Products
              <ChevronRight className="ml-1 w-4 h-4" />
            </button>
          </div>

          <div className="bg-white p-6 rounded-lg shadow-sm hover:shadow-md transition-shadow text-center">
            <div className="w-16 h-16 bg-blue-100 rounded-full flex items-center justify-center mx-auto mb-4">
              <svg xmlns="http://www.w3.org/2000/svg" className="w-8 h-8 text-blue-700" viewBox="0 0 24 24" 
    fill="none" stroke="currentColor" strokeWidth="2" strokeLinecap="round" strokeLinejoin="round">
                <circle cx="12" cy="12" r="10" />
                <path d="M12 2a14.5 14.5 0 0 0 0 20 14.5 14.5 0 0 0 0-20" />
```

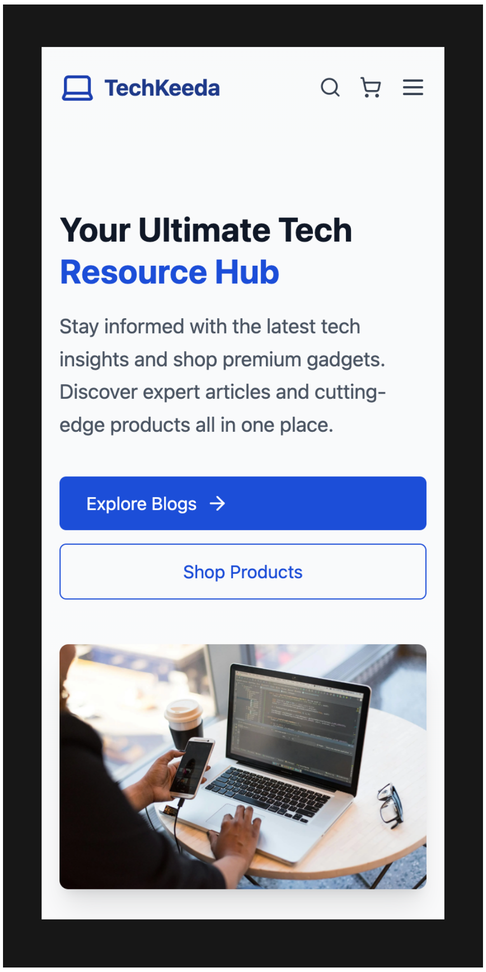
```
                <path d="M2 12h20" />
              </svg>
            </div>
            <h3 className="text-lg font-semibold text-gray-900 mb-2">Networking</h3>
            <p className="text-gray-600 text-sm mb-4">Routers, Switches, Wi-Fi Systems</p>
            <button
              onClick={() => setActivePage('shop')}
              className="text-blue-700 hover:text-blue-900 transition-colors flex items-center justify-center mx-auto text-sm font-medium"
            >
              View Products
              <ChevronRight className="ml-1 w-4 h-4" />
            </button>
          </div>

          <div className="bg-white p-6 rounded-lg shadow-sm hover:shadow-md transition-shadow text-center">
            <div className="w-16 h-16 bg-blue-100 rounded-full flex items-center justify-center mx-auto mb-4">
              <Cpu className="w-8 h-8 text-blue-700" />
            </div>
            <h3 className="text-lg font-semibold text-gray-900 mb-2">Components</h3>
            <p className="text-gray-600 text-sm mb-4">CPUs, GPUs, Memory, Storage</p>
            <button
              onClick={() => setActivePage('shop')}
              className="text-blue-700 hover:text-blue-900 transition-colors flex items-center justify-center mx-auto text-sm font-medium"
            >
              View Products
              <ChevronRight className="ml-1 w-4 h-4" />
            </button>
          </div>
        </div>
```

```
      </div>
    </section>
  }

    {/* Newsletter Section */}
    <section className="py-16 bg-blue-700">
      <div className="container mx-auto px-4 md:px-6">
        <div className="max-w-3xl mx-auto text-center">
          <h2 className="text-2xl md:text-3xl font-bold text-white mb-4">
            Stay Updated with TechKeeda
          </h2>
          <p className="text-blue-100 mb-8">
            Subscribe to our newsletter for the latest tech news, product releases, and exclusive offers.
          </p>
          <div className="flex flex-col sm:flex-row space-y-3 sm:space-y-0 max-w-lg mx-auto">
            <input
              type="email"
              placeholder="Enter your email"
              className="flex-grow px-4 py-3 rounded-l-md sm:rounded-r-none focus:outline-none focus:ring-2
focus:ring-blue-500"
            />
            <button className="bg-gray-900 text-white px-6 py-3 rounded-md sm:rounded-l-none hover:bg-gray-800
transition-colors">
              Subscribe
            </button>
          </div>
        </div>
      </div>
    </section>
  </div>
 );
};
```

```
export default HomePage;
```

Output:

A. Index/Home page output:



TechKeeda

Your Ultimate Tech Resource Hub

Stay informed with the latest tech insights and shop premium gadgets. Discover expert articles and cutting-edge products all in one place.

Code:

B. menu/product page:

code:

```
import React, { useState } from 'react';

import { products } from '../data/products';

import ProductCard from '../components/products/ProductCard';

import { Search, Filter, ShoppingBag } from 'lucide-react';


interface ShopPageProps {

 setSelectedProduct: (product: any) => void;

}


const ShopPage: React.FC<ShopPageProps> = ({ setSelectedProduct }) => {

 const [searchTerm, setSearchTerm] = useState('');
 const [selectedCategory, setSelectedCategory] = useState('All');
 const [showInStock, setShowInStock] = useState(false);
 const [sortBy, setSortBy] = useState('featured');
 const [isFilterOpen, setIsFilterOpen] = useState(false);

 const categories = ['All', ...Array.from(new Set(products.map(product => product.category)))];

 const filteredProducts = products.filter(product => {
   const matchesSearch = product.name.toLowerCase().includes(searchTerm.toLowerCase()) ||
                         product.description.toLowerCase().includes(searchTerm.toLowerCase());

   const matchesCategory = selectedCategory === 'All' || product.category === selectedCategory;

   const matchesStock = showInStock ? product.inStock : true;

   return matchesSearch && matchesCategory && matchesStock;
 }).sort((a, b) => {
   if (sortBy === 'price-low') {
     return a.price - b.price;
   } else if (sortBy === 'price-high') {
     return b.price - a.price;
   } else if (sortBy === 'rating') {
     return b.rating - a.rating;
   }
   return 0; // 'featured' - no sorting
 });

 return (
   <div className="pt-20 pb-16">
     {/* Shop Hero */}
     <section className="bg-blue-700 py-16">
       <div className="container mx-auto px-4 md:px-6 text-center">
         <div className="max-w-3xl mx-auto">
           <div className="inline-block bg-blue-800 p-2 rounded-full mb-4">
             <ShoppingBag className="w-6 h-6 text-white" />
           </div>
           <h1 className="text-3xl md:text-4xl font-bold text-white mb-4">
             Tech Products Shop
           </h1>
           <p className="text-blue-100 text-lg mb-8">
             Premium tech gadgets and accessories for professionals and enthusiasts
           </p>
           <div className="relative max-w-2xl mx-auto">
             <div className="absolute inset-y-0 left-0 pl-3 flex items-center pointer-events-none">
               <Search className="h-5 w-5 text-gray-400" />
             </div>
             <input
               type="text"
               placeholder="Search products..."
               value={searchTerm}
               onChange={(e) => setSearchTerm(e.target.value)}
               className="block w-full pl-10 pr-3 py-3 rounded-md bg-white text-gray-900 placeholder-gray-500 focus:outline-none focus:ring-2 focus:ring-blue-500"
             />
           </div>
         </div>
       </div>
     </section>

     {/* Shop Content */}
     <section className="py-12">
```

```
<div className="container mx-auto px-4 md:px-6">
 <div className="flex flex-col md:flex-row">
  {/* Sidebar - Desktop */}
  <div className="hidden md:block w-64 mr-8">
   <div className="bg-white rounded-lg shadow-sm p-6 sticky top-24">
    <h2 className="text-lg font-semibold text-gray-900 mb-4">Filters</h2>

    <div className="mb-6">
     <h3 className="text-sm font-medium text-gray-900 mb-2">Categories</h3>
     <div className="space-y-2">
      {categories.map((category) => (
       <div key={category} className="flex items-center">
        <input
         id={`category-${category}`}
         type="radio"
         checked={selectedCategory === category}
         onChange={() => setSelectedCategory(category)}
         className="h-4 w-4 text-blue-700 focus:ring-blue-500"
        />
        <label htmlFor={`category-${category}`} className="ml-2 text-sm text-gray-700">
         {category}
        </label>
       </div>
      ))}
     </div>
    </div>

    <div className="mb-6">
     <h3 className="text-sm font-medium text-gray-900 mb-2">Availability</h3>
     <div className="flex items-center">
```

```
        <input
         id="in-stock"
         type="checkbox"
         checked={showInStock}
         onChange={() => setShowInStock(!showInStock)}
         className="h-4 w-4 text-blue-700 focus:ring-blue-500"
        />
        <label htmlFor="in-stock" className="ml-2 text-sm text-gray-700">
         In Stock Only
        </label>
       </div>
      </div>

      <div>
       <h3 className="text-sm font-medium text-gray-900 mb-2">Sort By</h3>
       <select
        value={sortBy}
        onChange={(e) => setSortBy(e.target.value)}
        className="block w-full bg-white border border-gray-300 rounded-md py-2 px-3 text-sm focus:outline-none focus:ring-blue-500 focus:border-blue-500"
       >
        <option value="featured">Featured</option>
        <option value="price-low">Price: Low to High</option>
        <option value="price-high">Price: High to Low</option>
        <option value="rating">Highest Rated</option>
       </select>
      </div>
     </div>
    </div>

    {/* Mobile Filter Button */}
```

```
<div className="md:hidden mb-4">
  <button
    onClick={() => setIsFilterOpen(!isFilterOpen)}
    className="w-full bg-white border border-gray-300 rounded-md py-2 px-4 flex items-center justify-center text-gray-700 hover:bg-gray-50 transition-colors"
  >
    <Filter className="w-5 h-5 mr-2" />
    Filters & Sorting
  </button>
</div>

{/* Mobile Filters */}
{isFilterOpen && (
  <div className="md:hidden mb-6 bg-white rounded-lg shadow-sm p-4">
    <div className="mb-4">
      <h3 className="text-sm font-medium text-gray-900 mb-2">Categories</h3>
      <div className="flex flex-wrap gap-2">
        {categories.map((category) => (
          <button
            key={category}
            onClick={() => setSelectedCategory(category)}
            className={`px-3 py-1 rounded-full text-xs ${
              selectedCategory === category
                ? 'bg-blue-700 text-white'
                : 'bg-gray-100 text-gray-700'
            }`}
          >
            {category}
          </button>
        ))}
      </div>
```

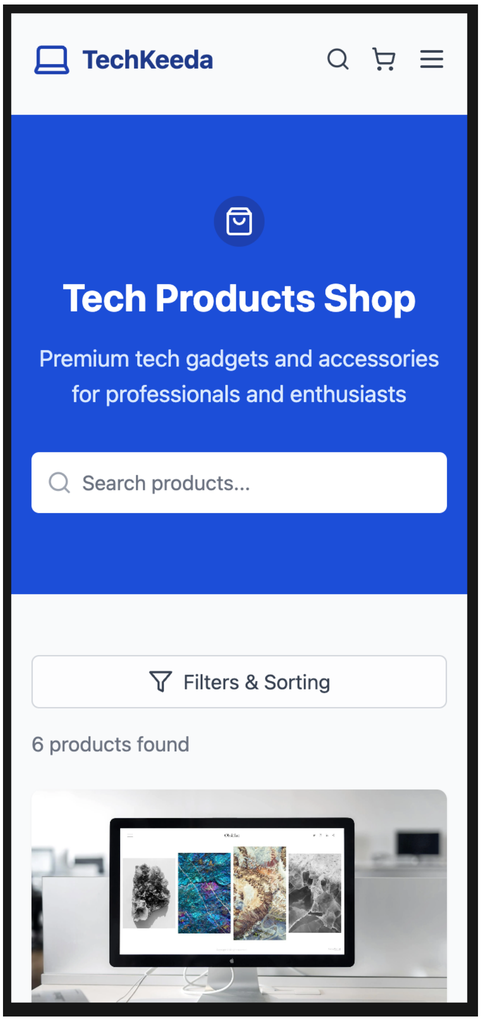
```
            </div>


            <div className="mb-4">
              <div className="flex items-center">
                <input
                  id="mobile-in-stock"
                  type="checkbox"
                  checked={showInStock}
                  onChange={() => setShowInStock(!showInStock)}
                  className="h-4 w-4 text-blue-700 focus:ring-blue-500"
                />
                <label htmlFor="mobile-in-stock" className="ml-2 text-sm text-gray-700">
                  In Stock Only
                </label>
              </div>
            </div>


            <div>
              <h3 className="text-sm font-medium text-gray-900 mb-2">Sort By</h3>
              <select
                value={sortBy}
                onChange={(e) => setSortBy(e.target.value)}
                className="block w-full bg-white border border-gray-300 rounded-md py-2 px-3 text-sm focus:outline-none focus:ring-blue-500 focus:border-blue-500"
              >
                <option value="featured">Featured</option>
                <option value="price-low">Price: Low to High</option>
                <option value="price-high">Price: High to Low</option>
                <option value="rating">Highest Rated</option>
              </select>
            </div>
```

```
              </div>
            )}

            {/* Product Grid */}
            <div className="flex-1">
              {filteredProducts.length === 0 ? (
                <div className="text-center py-12">
                  <p className="text-gray-500 text-lg">No products found. Try a different search term or filter.</p>
                </div>
              ) : (
                <>
                  <p className="text-gray-500 mb-6">{filteredProducts.length} products found</p>
                  <div className="grid grid-cols-1 sm:grid-cols-2 lg:grid-cols-3 gap-6">
                    {filteredProducts.map((product) => (
                      <ProductCard
                        key={product.id}
                        product={product}
                        setSelectedProduct={setSelectedProduct}
                      />
                    ))}
                  </div>
                </>
              )}
            </div>
          </div>
        </div>
      </section>
    </div>
  );
};
```

```
export default ShopPage;
```

Output:

B. menu/product page output:



TechKeeda

Tech Products Shop

Premium tech gadgets and accessories for professionals and enthusiasts

Search products...

Filters & Sorting

6 products found

Code:

C. cart page:

code:

```
import React from 'react';
import { useCart } from '../../contexts/CartContext';
import { X, Plus, Minus, ShoppingBag, ArrowRight } from 'lucide-react';
import { formatCurrency } from '../../utils/formatters';

interface CartSidebarProps {
  isOpen: boolean;
  onClose: () => void;
}

const CartSidebar: React.FC<CartSidebarProps> = ({ isOpen, onClose }) => {
  const { cart, removeFromCart, updateQuantity, getCartTotal, clearCart } = useCart();

  if (!isOpen) return null;

  return (
    <div className="fixed inset-0 z-50 overflow-hidden">
      <div
        className="absolute inset-0 bg-black bg-opacity-50 transition-opacity"
        onClick={onClose}
      />

      <div className="absolute inset-y-0 right-0 max-w-full flex">
        <div className="relative w-screen max-w-md">
          <div className="h-full flex flex-col bg-white shadow-xl overflow-y-auto">
            {/* Header */}
            <div className="px-4 pt-5 pb-4 sm:px-6 flex items-center justify-between">
              <h2 className="text-lg font-medium text-gray-900 flex items-center">
    <ShoppingBag className="w-5 h-5 mr-2" />
    Your Cart
  </h2>
  <button
    className="text-gray-400 hover:text-gray-500 transition-colors"
    onClick={onClose}
    aria-label="Close"
  >
    <X className="w-6 h-6" />
  </button>
</div>

{/* Cart Items */}
<div className="flex-1 px-4 sm:px-6">
  {cart.length === 0 ? (
    <div className="py-12 text-center">
      <ShoppingBag className="w-16 h-16 mx-auto text-gray-300" />
      <p className="mt-4 text-gray-500">Your cart is empty</p>
      <button
        className="mt-4 text-blue-600 hover:text-blue-800 transition-colors flex items-center mx-auto"
        onClick={onClose}
      >
        Continue Shopping
        <ArrowRight className="w-4 h-4 ml-1" />
      </button>
    </div>
  ) : (
    <ul className="divide-y divide-gray-200">
      {cart.map((item) => (
        <li key={item.product.id} className="py-4 flex">
```

```
<div className="h-24 w-24 flex-shrink-0 overflow-hidden rounded-md border border-gray-200">
  <img
    src={item.product.imageUrl}
    alt={item.product.name}
    className="h-full w-full object-cover object-center"
  />
</div>

<div className="ml-4 flex flex-1 flex-col">
  <div className="flex justify-between">
    <div>
      <h3 className="text-sm font-medium text-gray-900">
        {item.product.name}
      </h3>
      <p className="mt-1 text-sm text-gray-500">
        {item.product.category}
      </p>
    </div>
    <p className="text-sm font-medium text-gray-900">
      {formatCurrency(item.product.price)}
    </p>
  </div>

  <div className="flex flex-1 items-end justify-between text-sm">
    <div className="flex items-center border border-gray-200 rounded">
      <button
        className="px-2 py-1 text-gray-600 hover:text-blue-600 transition-colors"
        onClick={() => updateQuantity(item.product.id, item.quantity - 1)}
        aria-label="Decrease quantity"
      >
```

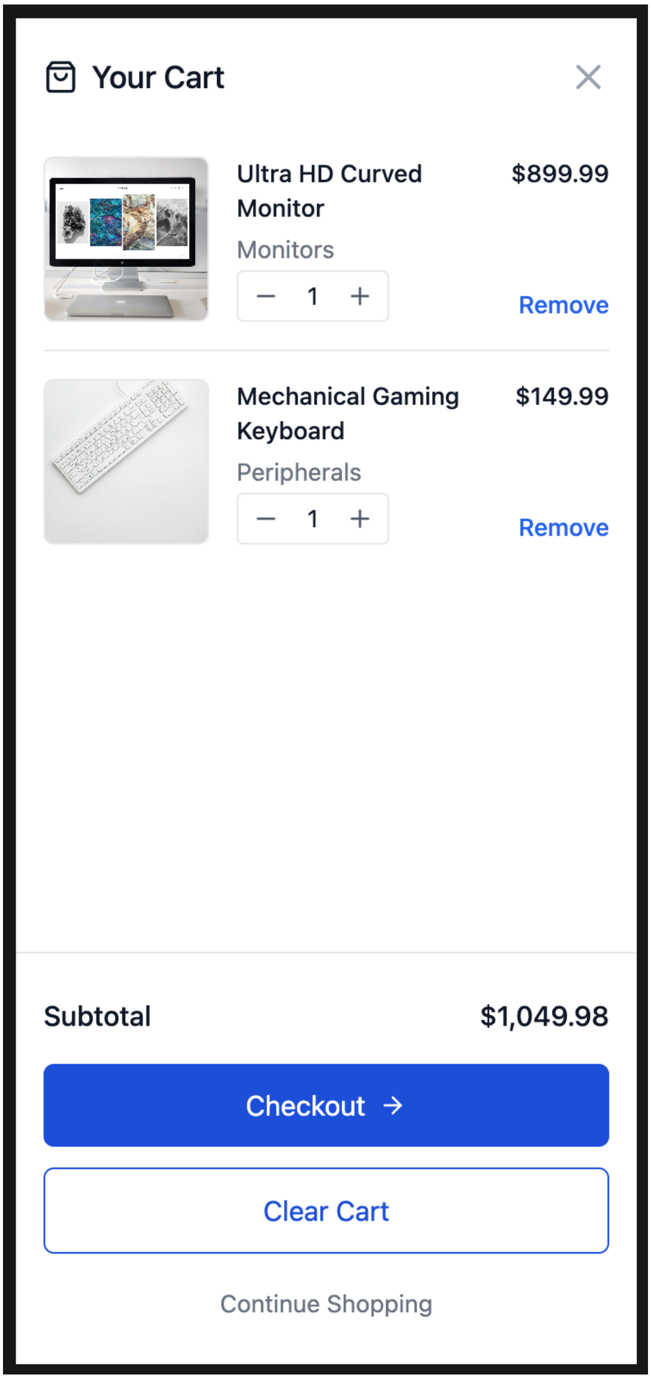
```
                  <Minus className="w-4 h-4" />
                </button>
                <span className="px-2 py-1 text-gray-900">{item.quantity}</span>
                <button
                  className="px-2 py-1 text-gray-600 hover:text-blue-600 transition-colors"
                  onClick={() => updateQuantity(item.product.id, item.quantity + 1)}
                  aria-label="Increase quantity"
                >
                  <Plus className="w-4 h-4" />
                </button>
              </div>

              <button
                type="button"
                className="font-medium text-blue-600 hover:text-blue-800 transition-colors"
                onClick={() => removeFromCart(item.product.id)}
              >
                Remove
              </button>
            </div>
          </div>
        </li>
      ))}
    </ul>
  )}
</div>

{/* Footer */}
{cart.length > 0 && (
  <div className="border-t border-gray-200 px-4 py-6 sm:px-6">
```

```
              <div className="flex justify-between text-base font-medium text-gray-900 mb-4">
                <p>Subtotal</p>
                <p>{formatCurrency(getCartTotal())}</p>
              </div>
              <div className="flex flex-col space-y-3">
                <button
                  className="w-full bg-blue-700 text-white py-3 px-4 rounded-md hover:bg-blue-800 transition-colors flex items-center justify-center"
                >
                  Checkout
                  <ArrowRight className="w-4 h-4 ml-2" />
                </button>
                <button
                  className="w-full text-blue-700 border border-blue-700 py-3 px-4 rounded-md hover:bg-blue-50 transition-colors"
                  onClick={clearCart}
                >
                  Clear Cart
                </button>
              </div>
              <div className="mt-4 text-center">
                <button
                  className="text-sm text-gray-500 hover:text-gray-700 transition-colors"
                  onClick={onClose}
                >
                  Continue Shopping
                </button>
              </div>
            </div>
          )}
        </div>
```

```
      </div>
    </div>
  </div>
 );
};

export default CartSidebar;
```

Output:

C. cart page output:



Your Cart

Ultra HD Curved Monitor $899.99

Monitors

Remove

Mechanical Gaming Keyboard $149.99

Peripherals

Remove

Subtotal $1,049.98

Code:

D. Blog card details:

code:

```
import React from 'react';

import { BlogPost } from '../../types';

import { ArrowLeft, Clock, User } from 'lucide-react';


interface BlogDetailProps {
  blog: BlogPost;
  onBack: () => void;
}

const BlogDetail: React.FC<BlogDetailProps> = ({ blog, onBack }) => {
  return (
    <div className="container mx-auto px-4 py-8 max-w-4xl">
      <button
        onClick={onBack}
        className="mb-6 flex items-center text-blue-700 hover:text-blue-900 transition-colors"
      >
        <ArrowLeft className="w-4 h-4 mr-2" />
        Back to Blogs
      </button>

      <div className="bg-white rounded-lg shadow-md overflow-hidden">
        <div className="relative h-[400px]">
          <img
            src={blog.imageUrl}
            alt={blog.title}
            className="w-full h-full object-cover"
          />
          <div className="absolute bottom-0 left-0 right-0 p-6 bg-gradient-to-t from-black to-transparent">
            <div className="inline-block bg-blue-700 text-white text-xs font-semibold px-3 py-1 rounded-full mb-3">
              {blog.category}
            </div>
            <h1 className="text-3xl md:text-4xl font-bold text-white mb-2">{blog.title}</h1>
          </div>
        </div>
```

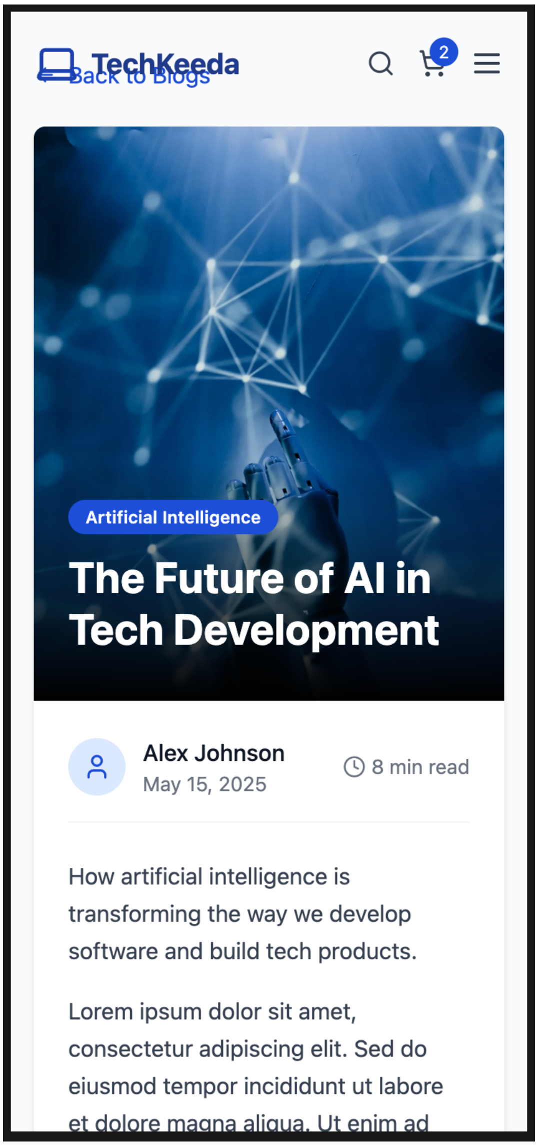
```
<div className="p-6 md:p-8">
  <div className="flex items-center justify-between mb-6 border-b border-gray-100 pb-4">
    <div className="flex items-center">
      <div className="w-10 h-10 rounded-full bg-blue-100 flex items-center justify-center text-blue-700">
        <User className="w-5 h-5" />
      </div>
      <div className="ml-3">
        <p className="text-gray-900 font-medium">{blog.author}</p>
        <p className="text-gray-500 text-sm">{blog.date}</p>
      </div>
    </div>
    <div className="flex items-center text-gray-500">
      <Clock className="w-4 h-4 mr-1" />
      <span className="text-sm">{blog.readTime}</span>
    </div>
  </div>


  <div className="prose prose-blue max-w-none">
    <p className="text-gray-700 leading-relaxed mb-4">
      {blog.excerpt}
    </p>
    <p className="text-gray-700 leading-relaxed mb-4">
      Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.
    </p>
    <h2 className="text-2xl font-semibold text-gray-900 mt-8 mb-4">Key Insights</h2>
    <p className="text-gray-700 leading-relaxed mb-4">
      Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.
    </p>
```

```
          <ul className="list-disc pl-5 mb-6 text-gray-700">
            <li className="mb-2">Innovative technology applications in modern software development</li>
            <li className="mb-2">Best practices for integrating AI into existing workflows</li>
            <li className="mb-2">Future trends and predictions for the tech industry</li>
            <li className="mb-2">How developers can prepare for upcoming challenges</li>
          </ul>
          <p className="text-gray-700 leading-relaxed">
            Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo.
          </p>
        </div>

        <div className="mt-8 pt-6 border-t border-gray-100">
          <h3 className="text-lg font-semibold text-gray-900 mb-4">Share this article</h3>
          <div className="flex space-x-4">
            <button className="bg-blue-600 text-white p-2 rounded-full hover:bg-blue-700 transition-colors">
              <svg className="w-5 h-5" fill="currentColor" viewBox="0 0 24 24" aria-hidden="true">
                <path d="M22 12c0-5.523-4.477-10-10-10S2 6.477 2 12c0 4.991 3.657 9.128 8.438 9.878v-6.987h-2.54V12h2.54V9.797c0-2.506 1.492-3.89 3.777-3.89 1.094 0 2.238.195 2.238.195v2.46h-1.26c-1.243 0-1.63.771-1.63 1.562V12h2.773l-.443 2.89h-2.33v6.988C18.343 21.128 22 16.991 22 12z"></path>
              </svg>
            </button>
            <button className="bg-sky-500 text-white p-2 rounded-full hover:bg-sky-600 transition-colors">
              <svg className="w-5 h-5" fill="currentColor" viewBox="0 0 24 24" aria-hidden="true">
                <path d="M8.29 20.251c7.547 0 11.675-6.253 11.675-11.675 0-.178 0-.355-.012-.53A8.348 8.348 0 0022 5.92a8.19 8.19 0 01-2.357.646 4.118 4.118 0 001.804-2.27 8.224 8.224 0 01-2.605.996 4.107 4.107 0 00-6.993 3.743 11.65 11.65 0 01-8.457-4.287 4.106 4.106 0 001.27 5.477A4.072 4.072 0 012.8 9.713v.052a4.105 4.105 0 003.292 4.022 4.095 4.095 0 01-1.853.07 4.108 4.108 0 003.834 2.85A8.233 8.233 0 012 18.407a11.616 11.616 0 006.29 1.84"></path>
              </svg>
            </button>
            <button className="bg-emerald-600 text-white p-2 rounded-full hover:bg-emerald-700 transition-colors">
```

```
                <svg className="w-5 h-5" fill="currentColor" viewBox="0 0 24 24" aria-hidden="true">
                  <path fillRule="evenodd" d="M12.315 2c2.43 0 2.784.013 3.808.06 1.064.049 1.791.218 2.427.465a4.902
4.902 0 011.772 1.153 4.902 4.902 0 011.153 1.772c.247.636.416 1.363.465 2.427.048 1.067.06 1.407.06
4.123v.08c0 2.643-.012 2.987-.06 4.043-.049 1.064-.218 1.791-.465 2.427a4.902 4.902 0 01-1.153 1.772 4.902
4.902 0 01-1.772 1.153c-.636.247-1.363.416-2.427.465-1.067.048-1.407.06-4.123.06h-.08c-2.643
0-2.987-.012-4.043-.06-1.064-.049-1.791-.218-2.427-.465a4.902 4.902 0 01-1.772-1.153 4.902 4.902 0
01-1.153-1.772c-.247-.636-.416-1.363-.465-2.427-.047-1.024-.06-1.379-.06-3.808v-.63c0-2.43.013-2.784.06-3.808.
049-1.064.218-1.791.465-2.427a4.902 4.902 0 011.153-1.772A4.902 4.902 0 015.45 2.525c.636-.247 1.363-.416
2.427-.465C8.901 2.013 9.256 2 11.685 2h.63zm-.081 1.802h-.468c-2.456
0-2.784.011-3.807.058-.975.045-1.504.207-1.857.344-.467.182-.8.398-1.15.748-.35.35-.566.683-.748
1.15-.137.353-.3.882-.344 1.857-.047 1.023-.058 1.351-.058 3.807v.468c0 2.456.011 2.784.058 3.807.045.975.207
1.504.344 1.857.182.466.399.8.748 1.15.35.35.683.566 1.15.748.353.137.882.3 1.857.344 1.054.048 1.37.058
4.041.058h.08c2.597 0 2.917-.01 3.96-.058.976-.045 1.505-.207 1.858-.344.466-.182.8-.398
1.15-.748.35-.35.566-.683.748-1.15.137-.353.3-.882.344-1.857.048-1.055.058-1.37.058-4.041v-.08c0-2.597-.01-2.9
17-.058-3.96-.045-.976-.207-1.505-.344-1.858a3.097 3.097 0 00-.748-1.15 3.098 3.098 0
00-1.15-.748c-.353-.137-.882-.3-1.857-.344-1.023-.047-1.351-.058-3.807-.058zM12 6.865a5.135 5.135 0 110 10.27
5.135 5.135 0 010-10.27zm0 1.802a3.333 3.333 0 100 6.666 3.333 3.333 0 000-6.666zm5.338-3.205a1.2 1.2 0 110
2.4 1.2 1.2 0 010-2.4z" clipRule="evenodd"></path>
                </svg>
              </button>
            </div>
          </div>
        </div>
      </div>
    </div>
  );
};


export default BlogDetail;
```

Output:

D. about us page output:



Artificial Intelligence

The Future of AI in Tech Development

Alex Johnson
May 15, 2025
8 min read

How artificial intelligence is transforming the way we develop software and build tech products.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad

Conclusion:

The TechKeeda App represents a modern approach to merging tech journalism with product sales in a single platform. With its clean architecture and component-driven design, it offers a scalable, maintainable, and engaging experience. Though a few user account features (like login and registration) are yet to be implemented, the app successfully achieves its core vision — a tech blog plus e-commerce hybrid solution for today's digital users.