Hackathon-3 Documentation

1. Steps Taken to Build and Integrate Components:

Project Setup:

We started by creating a new Next.js project using the create-next-app command. The project structure was organized into different directories: pages, components, styles, and utils.

Component Development:

We identified key UI components needed for the application, such as Navbar, Footer, and Card components. Each component was developed in isolation, ensuring they were reusable and maintainable. Components were organized in the components directory, and React hooks were used for state management where needed.

API Integration:

For data fetching, we utilized Next.js's getServerSideProps and getStaticProps methods to handle server-side rendering and static generation for better performance. API endpoints were created for backend integration.

Routing and Pages:

We utilized Next.js's file-based routing system. Dynamic routing was implemented where necessary to render pages with dynamic content (e.g., product details).

Styling:

CSS Modules were used for scoped styling, while Tailwind CSS was incorporated for utility-first design to ensure responsive and modern layouts. We also implemented custom styles for specific components to meet the project's branding needs.

Challenges Faced during Hackathone.

Initially, we struggled with rendering dynamic content on the server side. Managing state across components are also challenging for me. We faced some latency issues with API responses. To mitigate this, we implemented caching strategies and added loading states to enhance user experience.

```
import { createClient } from '@sanity/client';
   import axios from 'axios';
   import dotenv from 'dotenv';
   import { fileURLToPath } from 'url';
    import path from 'path';
   const __filename = fileURLToPath(import.meta.url);
 8 const __dirname = path.dirname(__filename);
   dotenv.config({ path: path.resolve(__dirname, '../../.env') });
   const client = createClient({
     projectId: process.env.NEXT_PUBLIC_SANITY_PROJECT_ID || 'clcvif8g',
      dataset: process.env.NEXT_PUBLIC_SANITY_DATASET || 'production',
      token: process.env.SANITY_API_TOKEN | 'skDK2XGQq065u9k9q02o6ERD5ffCBQv9F8r2ZZPBRC1INQWAmAG217eVGDwjUR4s7tn9dIhxX91Nec11m01vGWDq0s4Vt1X3uW1C1fV9FHdY0900PCkCFxeMM8Gge0j2tyHW2Cv6D2PCJ04xX1t0M7rKhZ1UonRPWfpMSk0Pi20JBQtEKUOX',
      apiVersion: '2025-01-15',
      useCdn: false,
   async function uploadImageToSanity(imageUrl) {
     try {
       console.log(`Uploading Image : ${imageUrl}`);
       const response = await axios.get(imageUrl, { responseType: 'arraybuffer' });
       const buffer = Buffer.from(response.data);
       const asset = await client.assets.upload('image', buffer, {
         filename: imageUrl.split('/').pop(),
28
       console.log(`Image Uploaded Successfully : ${asset._id}`);
       return asset._id;
      catch (error) {
       console.error('Failed to Upload Image:', imageUrl, error);
       return null;
38 async function importData() {
     try {
       console.log('Fetching Product Data From API ...');
       const response = await axios.get("https://next-ecommerce-template-4.vercel.app/api/product")
       const products = response.data.products;
44
45
        for (const item of products) {
         console.log(`Processing Item: ${item.name}`);
48
         let imageRef = null;
49
         if (item.imagePath) {
            imageRef = await uploadImageToSanity(item.imagePath);
50
51
         const sanityItem = {
           _type: 'product',
           id:product.id,
           name: item.name,
           category: item.category | null,
           price: item.price,
           description: item.description | '',
59
           discountPercentage: item.discountPercentage || 0,
60
           stockLevel: item.stockLevel | 0,
            isFeaturedProduct: item.isFeaturedProduct,
            image: imageRef
                 _type: 'image',
                 asset: {
                   _type: 'reference',
                   _ref: imageRef,
             : undefined,
          };
         console.log(`Uploading ${sanityItem.category} - ${sanityItem.name} to Sanity !`);
         const result = await client.create(sanityItem);
75
         console.log(`Uploaded Successfully: ${result._id}`);
         console.log("-----")
         console.log("\n\n")
79
       console.log('Data Import Completed Successfully !');
       catch (error) {
       console.error('Error Importing Data : ', error);
84
   importData();
```

```
const Featuredproducts = () => {
      const [product, setProduct] = useState<Product[]>([])
      useEffect(() => {
        async function fetchProduct() {
          const fetchedProduct: Product[] = await client.fetch(allProducts)
          setProduct(fetchedProduct)
        fetchProduct()
10
      }, [])
11
      return (
13
        <div className="py-16 px-4">
          <div className=" max-w-6x1 px-4">
14
            <h1 className="text-4x1 font-bold text-center text-black mb-12">
15
              Latest Products
16
            </h1>
18
              {product.map((product) => (
                <Link href={`/id/${product.slug.current}`}>
20
                  <div className="m-[100px] w-full max-w-sm bg-white border border-gray-200 rounded-lg shadow dark:bg-gray-800 dark:border-gray-700">
21
                    {product.image && (
22
23
                         <Image
                         src={urlFor(product.image).url()}
24
                         alt={product.name}
25
26
                         width={300}
27
                         height={300}
                         className="object-cover w-full h-full"
28
29
30
                    <div className="p-4 text-center">
31
                      <div className="text-x1 font-semibold tracking-tight text-gray-900 dark:text-white">{product.name}
32
                      <div className="flex items-center mt-2.5 mb-5">
33
34
          <div className="flex items-center space-x-1 rtl:space-x-reverse">
35
            <svg
36
              className="w-4 h-4 text-yellow-300"
              aria-hidden="true"
37
              xmlns="http://www.w3.org/2000/svg"
38
              fill="currentColor"
39
              viewBox="0 0 22 20"
40
```

```
1 'use client'
      2 import { allProducts } from '@/lib/queries'
      3 import { client } from '@/sanity/lib/client'
      4 import { urlFor } from '@/sanity/lib/image'
        5 import React, { useEffect, useState } from 'react'
        6 import Image from 'next/image'
         7 import Link from 'next/link'
   10 //<section className="py-16 px-4 ">
   //<h2 className="text-3xl font-bold text-center mb-8">Top Categories</h2>
    12 //</section><div className="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-4 gap-6"></div>
    17 const Featuredproducts = () => {
                const [product, setProduct] = useState<Product[]>([])
                  useEffect(() => {
                       async function fetchProduct() {
                           const fetchedProduct: Product[] = await client.fetch(allProducts)
                           setProduct(fetchedProduct)
                       fetchProduct()
                  },[])
                  return (
                        <div className="py-16 px-4">
                             <div className=" max-w-6x1 px-4">
                                 <h1 className="text-4x1 font-bold text-center text-black mb-12">
                                     Latest Products
                                  <div className="w-[1440px] grid grid-cols-1 sm:grid-cols-2 md:grid-cols-3 lg:grid-cols-4 gap-8">
                                     {product.map((product) => (
                                          <Link href={`/id/${product.slug.current}`}>
                                               <div className="m-[100px] w-full max-w-sm bg-white border border-gray-200 rounded-lg shadow dark:bg-gray-800 dark:border-gray-700">
                                                   {product.image && (
                                                                 <Image</pre>
                                                               src={urlFor(product.image).url()}
                                                               alt={product.name}
                                                               width={300}
                                                             height={300}
                                                              className="object-cover w-full h-full"
                                                    <div className="p-4 text-center">
                                                         <div className="text-x1 font-semibold tracking-tight text-gray-900 dark:text-white">{product.name}
                                                         <div className="flex items-center mt-2.5 mb-5">
                             <div className="flex items-center space-x-1 rtl:space-x-reverse">
                                     className="w-4 h-4 text-yellow-300"
                                     aria-hidden="true"
                                     xmlns="http://www.w3.org/2000/svg"
                                     fill="currentColor"
                                     viewBox="0 0 22 20"
                                      <path d="M20.924 7.625a1.523 1.523 0 0 0 -1.238-1.0441-5.051-.734-2.259-4.577a1.534 0 0 0 -2.752 0L7.365 5.8471-5.051.734A1.535 1.535 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.535 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.532 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.532 0 0 0 0 1.463 9.213.656 3.563-.863 5.031a1.532 1.532 0 0 0 0 1.463 9.213 5.031a1.532 1.532 0 0 0 0 1.463 9.213 5.031 5.031 5.031 5.031 5.031 5.031 5.031 5.031 5.031
                                </svg>
                                     className="w-4 h-4 text-yellow-300"
                                     aria-hidden="true"
                                     xmlns="http://www.w3.org/2000/svg"
                                     fill="currentColor"
                                     viewBox="0 0 22 20"
                                      <path d="M20.924 7.625a1.523 0 0 0-1.238-1.0441-5.051-.734-2.259-4.577a1.534 0 0 0 2.226-1.6171-.863-5.031a1.532 1.535 0 0 0 2.226-1.6171-.863-5.031a1.532 1.535 0 0 0 2.226-1.6171-.863-5.031a1.535 1.535 0 0 0 2.226-1.6171-.863-5.031a1.532 1.535 0 0 0 0 2.226-1.61611 17.03314.518 2.375a1.534 0 0 0 0 2.226-1.6171-.863-5.031a1.532 1.535 0 0 0 0 2.226-1.6171-.863-5.031a1.532 1.536 0 0 0 0 2.226-1.6171-.863-1.536 0 0 0 0 2.226-1.6171-.863-1.536 0 0 0 0 2.226-1.6171-.863-1.536 0 0 0 0 2.226-1.6171-.863
                                     className="w-4 h-4 text-yellow-300"
                                     aria-hidden="true"
                                     xmlns="http://www.w3.org/2000/svg"
                                     fill="currentColor"
                                    viewBox="0 0 22 20"
                                      </svg>
                                     className="w-4 h-4 text-yellow-300"
                                     aria-hidden="true"
                                     xmlns="http://www.w3.org/2000/svg"
                                     fill="currentColor"
                                     viewBox="0 0 22 20"
                                     <path d="M20.924 7.625a1.523 1.523 0 0 0-1.238-1.0441-5.051-.734-2.259-4.577a1.534 0 0 0 2.226-1.6171-.863-5.031a1.532 1.535 0 0 0 0 2.226-1.6171-.863-5.031a1.532 1.532 0 0 0 0 2.226-1.6171-.863-5.031a1.532 0 0 0 0 2.226-1.6171-.863-5.031a1.532 0 0 0 0 2.22
                                </svg>
                                     className="w-4 h-4 text-gray-200 dark:text-gray-600"
                                     aria-hidden="true"
                                     xmlns="http://www.w3.org/2000/svg"
                                     fill="currentColor"
                                     viewBox="0 0 22 20"
                                      <path d="M20.924 7.625a1.523 0 0 0 -1.238-1.0441-5.051-.734-2.259-4.577a1.534 0 0 0 2.226-1.6171-.863-5.031a1.532 1.535 0 0 0 -2.752 0L7.365 5.8471-5.051.734A1.535 1.535 0 0 0 2.226-1.6171-.863-5.031a1.532 1.535 0 0 0 2.226-1.6171-.863-5.031a1.532 1.535 0 0 0 0 2.226-1.6171-.863-5.031a1.535 1.535 0 0 0 0 2.226-1.6171-.863-5.031a1.532 1.532 0 0 0 0 2.226-1.6171-.863-5.031a1.532 1.532 0 0 0 0 2.226-1.6171-.863-5.031a1.532
                                </svg>
                             </div>
                             <span className="bg-blue-100 text-blue-800 text-xs font-semibold px-2.5 py-0.5 rounded dark:bg-blue-200 dark:text-blue-800 ms-3">
                                 5.0
                            </span>
                        </div>
                        <div className="flex items-center justify-between">
                             <span className="text-3x1 font-bold text-gray-900 dark:text-white">
                           ${product.price}
                            </span>
                                href="#"
                                className="text-white bg-blue-700 hover:bg-blue-800 focus:ring-blue-800 focus:ring-blue-800" className="text-white bg-blue-600 dark:hover:bg-blue-700 dark:focus:ring-blue-800"
                                Add to cart
                       </div>
                                                        </div>
                                                   </div>
                                              </div>
                                          </Link>
  123 export default Featuredproducts
```





