**API INTEGRATION REPORT (DAY-03)**

**API INTEGRATION PROCESS:**

* **Wrote code in page.tsx to check API by fetching it by this code.**
* // DATA FETCH FROM API TO WEBSITE
* // "use client"
* // import React, { useEffect, useState } from "react";
* // const HomePage = () => {
* //   const [data, setData] = useState(null);
* //   const [loading, setLoading] = useState(true);
* //   useEffect(() => {
* //     const fetchData = async () => {
* //       const response = await fetch("https://template6-six.vercel.app/api/products");
* //       const result = await response.json();
* //       setData(result);
* //       setLoading(false);
* //     };
* //     fetchData();
* //   }, []);
* //   if (loading) return <p>Loading...</p>;
* //   return (
* //     <div>
* //       <h1>Fetched Data</h1>
* //       <pre>{JSON.stringify(data, null, 2)}</pre>
* //     </div>
* //   );
* // };
* **Created Schema in scemaTypes/products.ts in this file I wrote my schema for products.**

import { defineType } from "sanity"

export const product = defineType({

    name: "product",

    title: "Product",

    type: "document",

    fields: [

        {

            name: "title",

            title: "Title",

            validation: (rule) => rule.required(),

            type: "string"

        },

        {

            name:"description",

            type:"text",

            validation: (rule) => rule.required(),

            title:"Description",

        },

        {

            name: "productImage",

            type: "image",

            validation: (rule) => rule.required(),

            title: "Product Image"

        },

        {

            name: "price",

            type: "number",

            validation: (rule) => rule.required(),

            title: "Price",

        },

        {

            name: "tags",

            type: "array",

            title: "Tags",

            of: [{ type: "string" }]

        },

        {

            name:"discountPercentage",

            type:"number",

            title:"Discount Percentage",

        },

        {

            name:"isNew",

            type:"boolean",

            title:"New Badge",

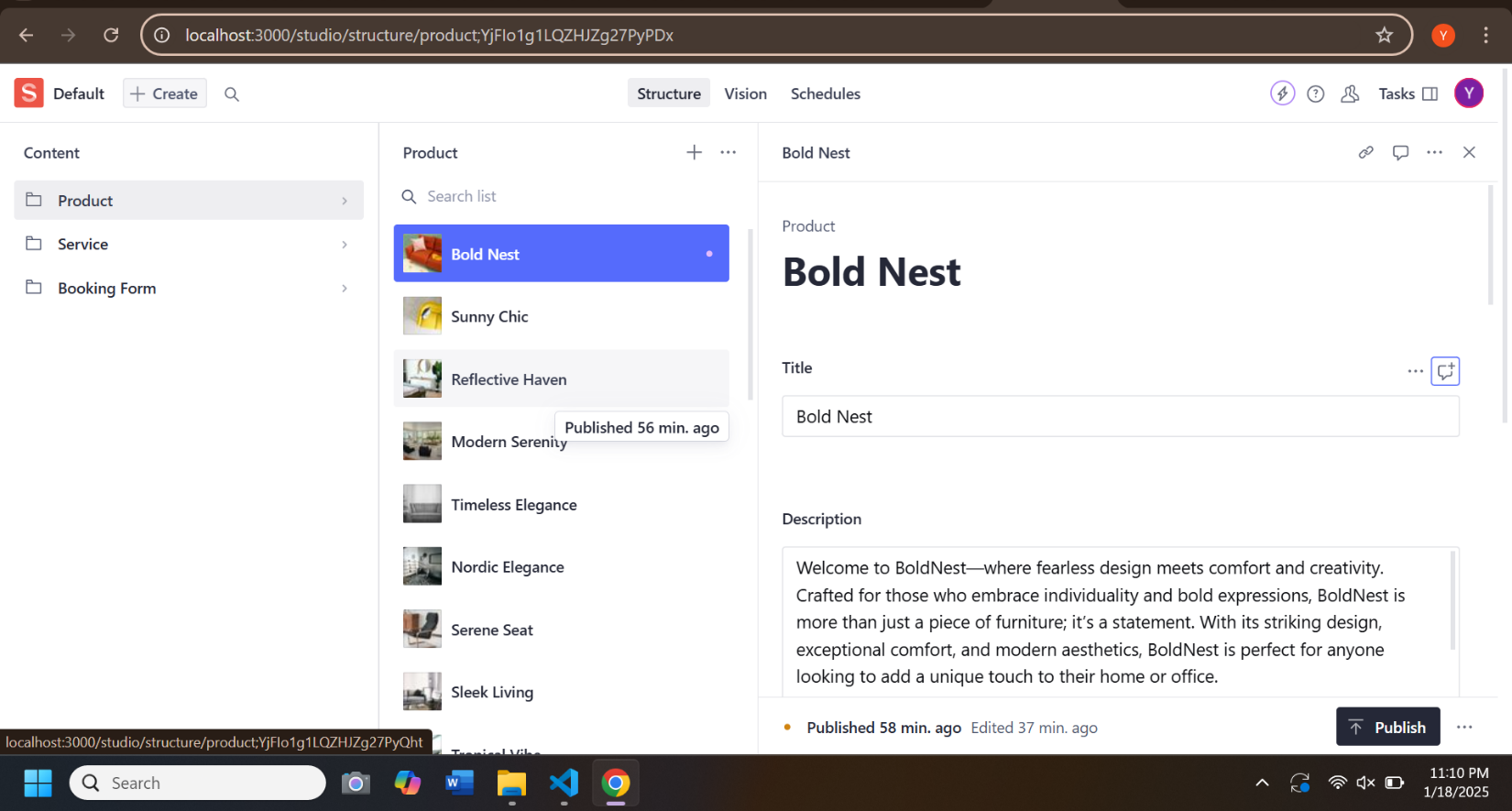
        }

    ]

})

* **Made file named scripts/migrate.mjs and wrote this below code. This code is for migrating data from External API to Sanity .**
* import { createClient } from '@sanity/client';
* import axios from 'axios';
* import dotenv from 'dotenv';
* import { fileURLToPath } from 'url';
* import path from 'path';
* // Load environment variables from .env.local
* const \_\_filename = fileURLToPath(import.meta.url);
* const \_\_dirname = path.dirname(\_\_filename);
* dotenv.config({ path: path.resolve(\_\_dirname, '../.env.local') });
* // Create Sanity client
* const client = createClient({
* projectId: process.env.NEXT\_PUBLIC\_SANITY\_PROJECT\_ID,
* dataset: process.env.NEXT\_PUBLIC\_SANITY\_DATASET,
* useCdn: false,
* token: process.env.NEXT\_PUBLIC\_SANITY\_TOKEN,
* apiVersion: '2021-08-31'
* });
* 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()
* });
* console.log(`Image uploaded successfully: ${asset.\_id}`);
* return asset.\_id;
* } catch (error) {
* console.error('Failed to upload image:', imageUrl, error);
* return null;
* }
* }
* async function uploadProduct(product) {
* try {
* const imageId = await uploadImageToSanity(product.imageUrl);
* if (imageId) {
* const document = {
* \_type: 'product',
* title: product.title,
* price: product.price,
* productImage: {
* \_type: 'image',
* asset: {
* \_ref: imageId,
* },
* },
* tags: product.tags,
* dicountPercentage: product.discountPercentage, // Typo in field name: dicountPercentage -> discountPercentage
* description: product.description,
* isNew: product.isNew,
* };
* const createdProduct = await client.create(document);
* console.log(`Product ${product.title} uploaded successfully:`, createdProduct);
* } else {
* console.log(`Product ${product.title} skipped due to image upload failure.`);
* }
* } catch (error) {
* console.error('Error uploading product:', error);
* }
* }
* async function importProducts() {
* try {
* const response = await fetch('https://template6-six.vercel.app/api/products');
* if (!response.ok) {
* throw new Error(`HTTP error! Status: ${response.status}`);
* }
* const products = await response.json();
* for (const product of products) {
* await uploadProduct(product);
* }
* } catch (error) {
* console.error('Error fetching products:', error);
* }
* }
* importProducts();
* **Before running below command run : npm install dotenv and then make file named .env and wrote same code as .env.local**
* **Now run command : npm run migrate**

**SCREENSHOTS OF DISPLAYED DATA:**

****