Hackthon Day-3 Report

Prepared By: Jawaid Ali

Reporting Date: Jan 17, 2025

API Integration process

I add the given API to the script that works to import or fetch api data and also migrating the the products data to sanity.

I have already installed sanity in project, you can install according to your need then i just created a folder called script and the in folder created a file called import-data.mjs so in this file i added the logic to perform action.

After just run the command in terminal npm run import-data. As soon as command will run the all the provided data send to sanity one by one.

Note: Also add the import-data in your package.json file to configure the and perform well in terminal.

The script used to fetch and migrate data to sanity:

```
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);

const __dirname = path.dirname(__filename);

dotenv.config({ path: path.resolve(__dirname, '../../.env.local') });

console.log('Sanity Project ID:', process.env.NEXT_PUBLIC_SANITY_PROJECT_ID);

const client = createClient({

  projectId: process.env.NEXT_PUBLIC_SANITY_PROJECT_ID,

  dataset: process.env.NEXT_PUBLIC_SANITY_DATASET,

  token: process.env.SANITY_API_TOKEN,

  apiVersion: '2025-01-17',

  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(),
    });
    console.log(`Image Uploaded Successfully : ${asset._id}`);
    return asset._id;
  }
  catch (error) {
    console.error('Failed to Upload Image:', imageUrl, error);
    return null;
  }
}

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;

    for (const item of products) {
      console.log(`Processing Item: ${item.name}`);

      let imageRef = null;
      if (item.imagePath) {
        imageRef = await uploadImageToSanity(item.imagePath);
      }

      const sanityItem = {
        _type: 'product',
        name: item.name,
        category: item.category || null,
        price: item.price,
        description: item.description || '',
        discountPercentage: item.discountPercentage || 0,
        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);
      console.log(`Uploaded Successfully: ${result._id}`);
      console.log("----------------------------------------------------------")
      console.log("\n\n")
    }

    console.log('Data Import Completed Successfully !');
  } catch (error) {
    console.error('Error Importing Data : ', error);

  }

}
```

Adjustment made to schema

According to data i made some changes in my schema and added all the required field that are mentioned in data like productName, id, description, image, and more.

},

Updated Schema according to data and i used definetype to handle types.

```
import { defineField, defineType } from "sanity";

import { TrolleyIcon } from "@sanity/icons";



export const product = defineType({

  name: "product",

  title: "Product",

  type: "document",

  icon: TrolleyIcon,

  fields: [

    defineField({

      name: "name",

      title: "Name",

      type: "string",

      validation: (Rule) => Rule.required().error("Name is required"),

    }),

    defineField({

      name: "image",

      title: "Image",

      type: "image",

      options: {

        hotspot: true, // Allow hotspot selection for better cropping

      },

      description: "Upload an image of the product.",

    }),

    defineField({
```

```
    name: "price",
    title: "Price",
    type: "string",
    validation: (Rule) => Rule.required().error("Price is required"),
  }),
  defineField({
    name: "description",
    title: "Description",
    type: "text",
    validation: (Rule) =>
      Rule.max(150).warning("Keep the description under 150 characters."),
  }),
  defineField({
    name: "discountPercentage",
    title: "Discount Percentage",
    type: "number",
    validation: (Rule) =>
      Rule.min(0)
        .max(100)
        .warning("Discount must be between 0 and 100."),
  }),
  defineField({
    name: "isFeaturedProduct",
    title: "Is Featured Product",
    type: "boolean",
  }),
  defineField({
    name: "isLatestProduct",
    title: "Is Latest Product",
    type: "boolean",
  }),
  defineField({
    name: "isTrending",
    title: "Is Trending Product",
    type: "boolean",
  }),
  defineField({
    name: "stockLevel",
    title: "Stock Level",
    type: "number",
    validation: (Rule) =>
      Rule.min(0).error("Stock level must be a positive number."),
  }),
  defineField({
    name: "category",
    title: "Category",
    type: "string",
    options: {
      list: [
        { title: "Chair", value: "Chair" },
        { title: "Sofa", value: "Sofa" },
      ],
    },
    validation: (Rule) => Rule.required().error("Category is required"),
  }),
],
preview: {
  select: {
    title: "name",
    media: "image",
    subtitle: "price",
  },
  prepare(selection) {
    return {
      title: selection.title,
      subtitle: `$${selection.subtitle}`,
      media: selection.media,
    };
  },
```

Migration Steps & used Tools

Migration Steps:

1. Data Preparation:

Prepared the data according to the schema defined in Sanity.

2. Script Creation:

Created a custom script for transforming and structuring the data to match Sanity's requirements.

3. Command Execution:

Used the sanity dataset import command or ran the custom script to import the data into Sanity.

4. Validation:

Verified the migrated data in Sanity Studio to ensure everything was successfully imported.

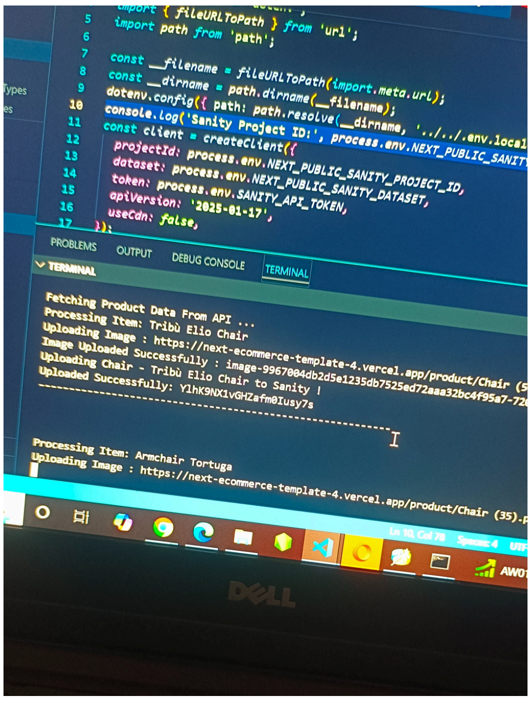
Tools Used:

1. Sanity CLI: For running commands and managing the dataset migration.

2 Sanity Studio: To validate and test the migrated data.

This explanation directly addresses the

✓ Pictures of completion

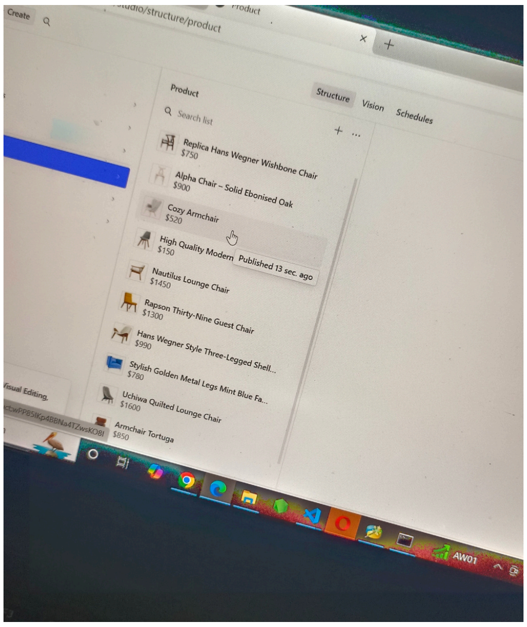


```
import { fileURLToPath } from 'url';
import path from 'path';

const __filename = fileURLToPath(import.meta.url);
const __dirname = path.dirname(__filename);
dotenv.config({ path: path.resolve(__dirname, '../../.env.local
console.log('Sanity Project ID:', process.env.NEXT_PUBLIC_SANITY
const client = createClient({
  projectId: process.env.NEXT_PUBLIC_SANITY_PROJECT_ID,
  dataset: process.env.NEXT_PUBLIC_SANITY_DATASET,
  token: process.env.SANITY_API_TOKEN,
  apiVersion: '2025-01-17',
  useCdn: false,
});
```

```
Fetching Product Data From API ...
Processing Item: Tribù Elio Chair
Uploading Image : https://next-ecommerce-template-4.vercel.app/product/Chair (5
Image Uploaded Successfully : image-9967004db2d5e1235db7525ed72aaa32bc4f95a7-72
Uploading Chair - Tribù Elio Chair to Sanity !
Uploaded Successfully: YlhK9NX1vGHZafm0Iusy7s
------------------------------------------------------------------------


Processing Item: Armchair Tortuga
Uploading Image : https://next-ecommerce-template-4.vercel.app/product/Chair (35).p
```



Product

Replica Hans Wegner Wishbone Chair
$750

Alpha Chair – Solid Ebonised Oak
$900

Cozy Armchair
$520

High Quality Modern
$150

Nautilus Lounge Chair
$1450

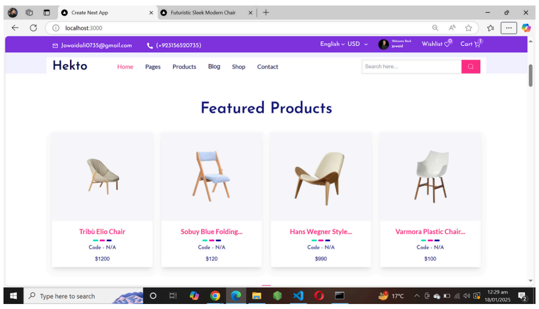
Rapson Thirty-Nine Guest Chair
$1300

Hans Wegner Style Three-Legged Shell...
$990

Stylish Golden Metal Legs Mint Blue Fa...
$780

Uchiwa Quilted Lounge Chair
$1600

Armchair Tortuga
$850



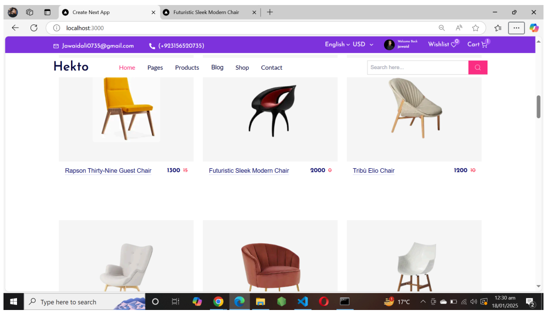
Featured Products

Tribù Elio Chair
Code - N/A
$1200

Sobuy Blue Folding...
Code - N/A
$120

Hans Wegner Style...
Code - N/A
$990

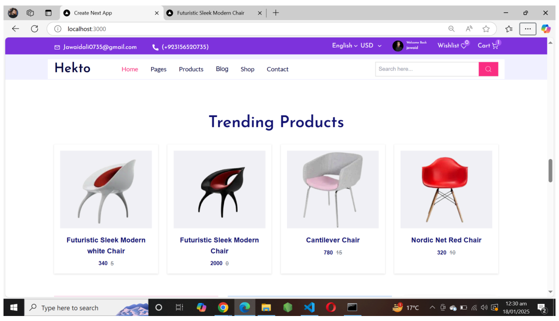
Varmora Plastic Chair...
Code - N/A
$100

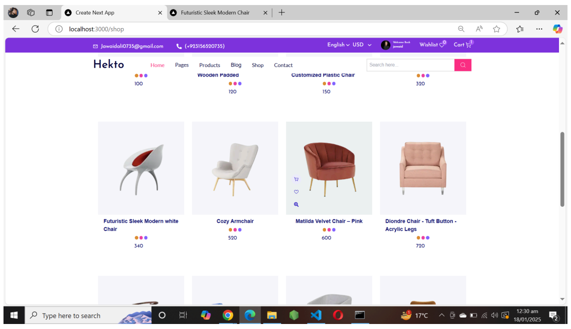


Rapson Thirty-Nine Guest Chair 1300 15

Futuristic Sleek Modern Chair 2000 0

Tribù Elio Chair 1200 10




Hekto
Home
Pages
Products
Blog
Shop
Contact
Futuristic Sleek Modern white Chair
340
Cozy Armchair
520
Matilda Velvet Chair – Pink
600
Diondre Chair - Tuft Button - Acrylic Legs
720

Progress Overview

Understanding	Check	Due Date	Status
Api Fetch	✓	Jan 17, 2025	Done
Migration	✓	Jan 17, 2025	Done
Schema adjust	✓	Jun 20, 2030	Done
Sanity to frontend	✓	Jan 17, 2025	Done

Notes

- Data feilds and schema feild are align together
- Add import data in package file to run the command.