

Day 3 - API Integration Report – HEKTO

1. Sanity Project Initialization

- Created a new Sanity project.
- Added `NEXT_PUBLIC_SANITY_PROJECT_ID`, `SANITY_API_TOKEN`, and `NEXT_PUBLIC_SANITY_DATASET` in `.env.local`.



CORS origins

[+ Add CORS origin](#)

Hosts that can connect to the project API.

ORIGIN	CREDENTIALS	CREATED	
http://localhost:3000	Allowed	5 days	
http://localhost:3333	Allowed	5 days	

Tokens

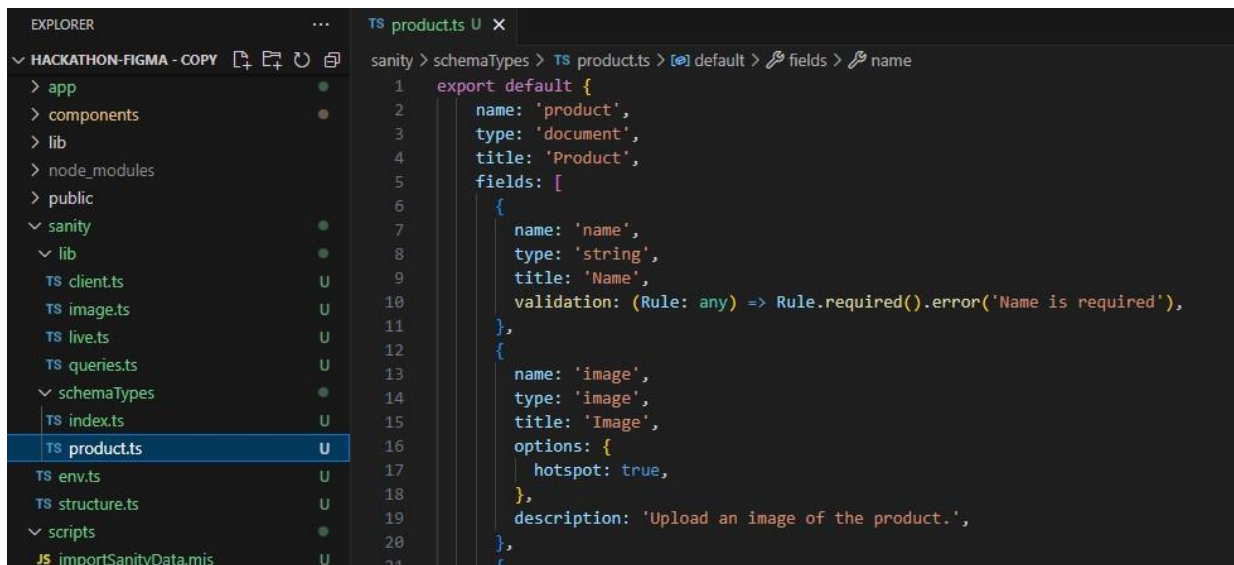
[+ Add API token](#)

Tokens are used to authenticate apps and scripts to access project data.

NAME	PERMISSIONS	CREATED	
DataImport	Developer	5 days	

2. Schema Setup

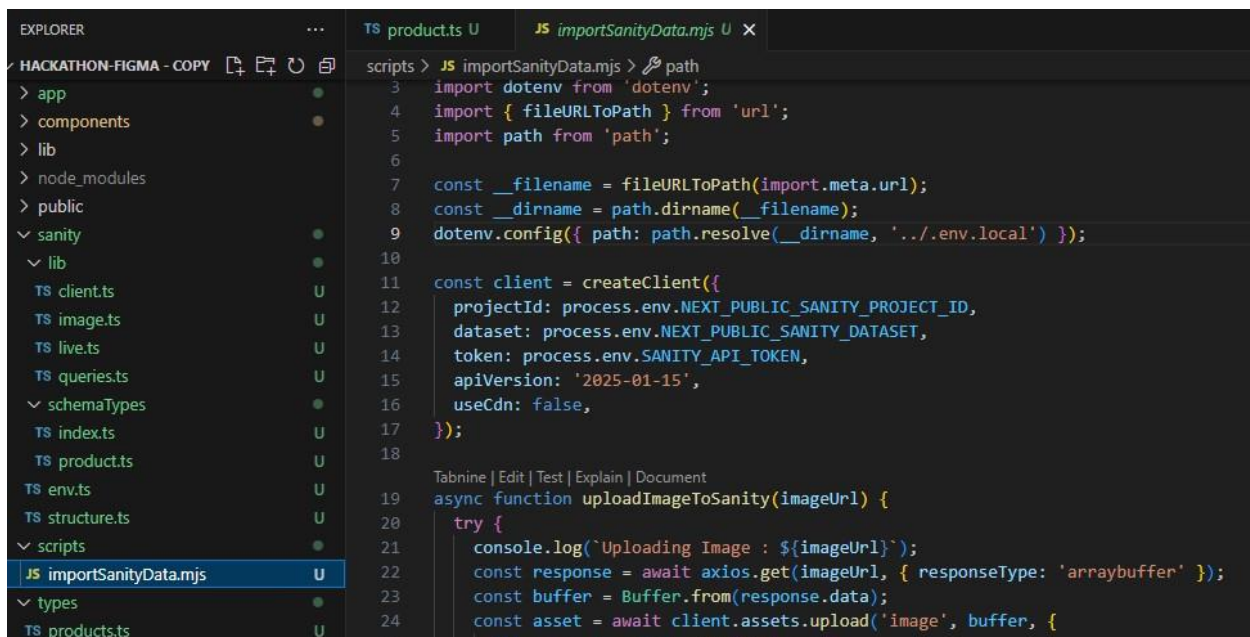
- Created `product.ts` in the schema folder.
- Defined the product schema with fields like `name`, `price`, `image`, `category`, `stockLevel`.
- Imported the schema into `index.ts` in the schema folder.



```
1 export default {
2   name: 'product',
3   type: 'document',
4   title: 'Product',
5   fields: [
6     {
7       name: 'name',
8       type: 'string',
9       title: 'Name',
10      validation: (Rule: any) => Rule.required().error('Name is required'),
11    },
12    {
13      name: 'image',
14      type: 'image',
15      title: 'Image',
16      options: {
17        hotspot: true,
18      },
19      description: 'Upload an image of the product.',
20    },
21  ],
22 }
```

3. Data Import

- Created `importSanityData.mjs` in the `scripts` folder.
- Wrote a script to fetch product data from an external API and upload it to Sanity CMS.

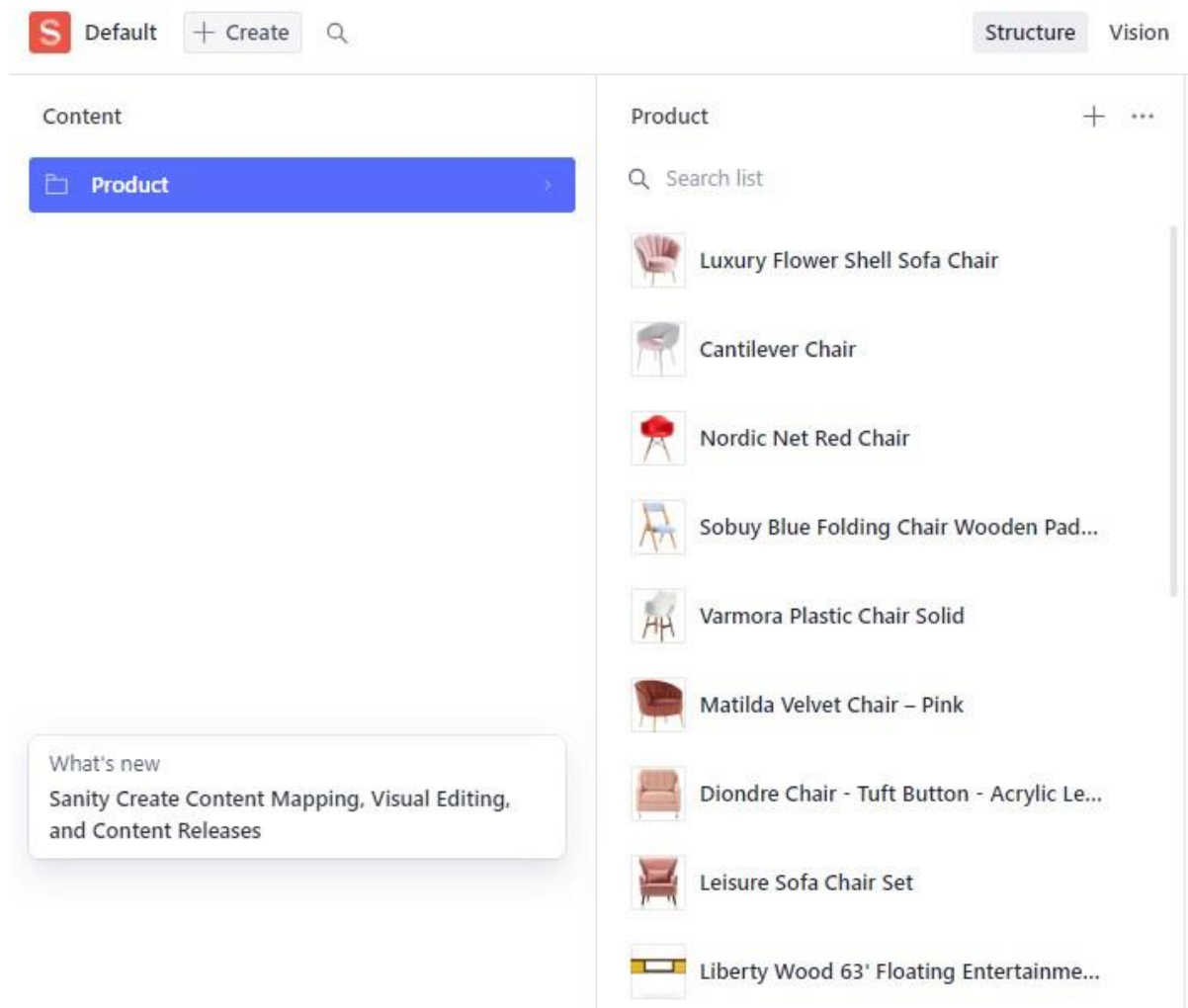


```
3 import dotenv from 'dotenv';
4 import { fileURLToPath } from 'url';
5 import path from 'path';
6
7 const __filename = fileURLToPath(import.meta.url);
8 const __dirname = path.dirname(__filename);
9 dotenv.config({ path: path.resolve(__dirname, '../.env.local') });
10
11 const client = createClient({
12   projectId: process.env.NEXT_PUBLIC_SANITY_PROJECT_ID,
13   dataset: process.env.NEXT_PUBLIC_SANITY_DATASET,
14   token: process.env.SANITY_API_TOKEN,
15   apiVersion: '2025-01-15',
16   useCdn: false,
17 });
18
19 async function uploadImageToSanity(imageUrl) {
20   try {
21     console.log(`Uploading Image : ${imageUrl}`);
22     const response = await axios.get(imageUrl, { responseType: 'arraybuffer' });
23     const buffer = Buffer.from(response.data);
24     const asset = await client.assets.upload('image', buffer, {
```

Added the following script in `package.json`:

```
TS product.ts U  {} package.json M X
{} package.json > {} scripts
1  {
2    "name": "hackathon-figma",
3    "version": "0.1.0",
4    "private": true,
5    "type": "module",
6    "scripts": {
7      "dev": "next dev --turbo",
8      "build": "next build",
9      "start": "next start",
10     "lint": "next lint",
11     "import-data": "node scripts/importSanityData.mjs"
12   },
```

npm run import-data to import data into Sanity Studio.



Sanity Studio interface showing a GROQ query and its result.

Sanity Studio Header: Default + Create Search Structure Vision Schedules

Configuration:

- DATASET: production
- API VERSION: Other
- CUSTOM API VERSION: v2025-01-30
- PERSPECTIVE: raw
- QUERY URL [COPY TO CLIPBOARD]: https://70kmkve2.api.sanity

QUERY:

```
1 *[type==product]
```

RESULT:

```
[...] 52 items
  0: {...} 13 properties
    _updatedAt: 2025-01-30T20:36:59Z
    _type: product
    stockLevel: 20
    price: 1300
    _createdAt: 2025-01-30T20:36:59Z
    name: Rapson Thirty-Nine Guest Chair
    _id: 5u5qSfeSP8Fhinu4Rp6MSm
    isFeaturedProduct: true
```


4. Fetching Data in Next.js

- Created `product.ts` in the `types` folder to define the `Product` interface.
- Created `queries.ts` in the `lib` folder and wrote a GROQ query to fetch product data:
- Updated `next.config.js` to allow images from Sanity:

5. Displaying Products in Next.js

- Used `useEffect` in a component to fetch products from Sanity:


Our Furniture



Rapson Thirty-Nine Guest Chair

A mid-century modern chair with clean lines and durable materials.


Price: \$1300



High Quality Modern Customized Plastic Chair

A simple and lightweight chair ideal for dining or casual use.

Price: \$150



Alpha Chair – Solid Ebonised Oak

A sturdy oak chair with a sleek and minimalist design.

Price: \$900