

Hackathon 03

Day 03 Task

MUHAMMAD AMSAL 482160

This is my Day 3 GIAIC Hackathon task.

Sanity.io is a powerful headless CMS that allows you to structure content and serve it through an API. Here's how I set up my project:

- **Install Sanity in the Project**

First, I initialized a new Sanity project within my Next.js setup by running the following command:

```
E:\hackathon-giaic>npm create sanity@latest
Need to install the following packages:
  create-sanity@3.70.0
Ok to proceed? (y) y
✔ You are logged in as muhammadamsal520@gmail.com using Google
✔ Fetching existing projects

? Create a new project or select an existing one Create new project
? Your project name: temp0api
Your content will be stored in a dataset that can be public or private, depending on
whether you want to query your content with or without authentication.
The default dataset configuration has a public dataset named "production".
? Use the default dataset configuration? Yes
✔ Creating dataset
? Would you like to add configuration files for a Sanity project in this Next.js folder? Yes
? Do you want to use TypeScript? Yes
? Would you like an embedded Sanity Studio? Yes
? What route do you want to use for the Studio? /studio
? Select project template to use Clean project with no predefined schema types
? Would you like to add the project ID and dataset to your .env.local file? Yes
Added http://localhost:3000 to CORS origins
Running 'npm install --legacy-peer-deps --save @sanity/vision@3 sanity@3 @sanity/image-url@1 styled-components@6'
npm WARN EBADENGINE Unsupported engine {
npm WARN EBADENGINE   package: '@sanity/diff-match-patch@3.1.2',
npm WARN EBADENGINE   required: { node: '>=18.18' },
npm WARN EBADENGINE   current: { node: 'v18.17.0', npm: '9.6.7' }
npm WARN EBADENGINE }
npm WARN EBADENGINE Unsupported engine {
npm WARN EBADENGINE   package: 'rimraf@6.0.1',
npm WARN EBADENGINE   required: { node: '20 || >=22' },
npm WARN EBADENGINE   current: { node: 'v18.17.0', npm: '9.6.7' }
npm WARN EBADENGINE }
npm WARN EBADENGINE Unsupported engine {
npm WARN EBADENGINE   package: 'rimraf@6.0.1',
npm WARN EBADENGINE   required: { node: '20 || >=22' },
npm WARN EBADENGINE   current: { node: 'v18.17.0', npm: '9.6.7' }
npm WARN EBADENGINE }
npm WARN EBADENGINE Unsupported engine {
```

```

npm WARN EBADENGINE Unsupported engine {
npm WARN EBADENGINE   package: 'path-scurry@2.0.0',
npm WARN EBADENGINE   required: { node: '20 || >=22' },
npm WARN EBADENGINE   current: { node: 'v18.17.0', npm: '9.6.7' }
npm WARN EBADENGINE }
npm WARN EBADENGINE Unsupported engine {
npm WARN EBADENGINE   package: 'rimraf@6.0.1',
npm WARN EBADENGINE   required: { node: '20 || >=22' },
npm WARN EBADENGINE   current: { node: 'v18.17.0', npm: '9.6.7' }
npm WARN EBADENGINE }
npm WARN EBADENGINE Unsupported engine {
npm WARN EBADENGINE   package: 'next-sanity@9.8.38',
npm WARN EBADENGINE   required: { node: '>=18.18' },
npm WARN EBADENGINE   current: { node: 'v18.17.0', npm: '9.6.7' }
npm WARN EBADENGINE }
npm WARN EBADENGINE Unsupported engine {
npm WARN EBADENGINE   package: '@sanity/next-loader@1.2.15',
npm WARN EBADENGINE   required: { node: '>=18.18' },
npm WARN EBADENGINE   current: { node: 'v18.17.0', npm: '9.6.7' }
npm WARN EBADENGINE }

added 16 packages, and audited 1292 packages in 15s

239 packages are looking for funding
  run `npm fund` for details

1 high severity vulnerability

To address all issues, run:
  npm audit fix --force

Run `npm audit` for details.

Success! Your Sanity configuration files has been added to this project

E:\hackathon-giaic>

```

Make a Project in Sanity

GI

muhammad amsal
giaic_hackathon

PLAN

STATUS

PROJECT ID

Growth Trial

Active

qc9ei68n

[Getting started](#)
[Overview](#)
[Members](#)
[Studios](#)
[Datasets](#)
[Access](#)
[Activity](#)
[Usage](#)
[Plan](#)
[API](#)
[Settings](#)

Webhooks

CORS origins

Tokens

new


Create Content Mapping, Visual Editing, Content Releases

GROQ-powered webhooks

+ Create webhook

HTTP callbacks to a given URL triggered by changes in your content lake

Learn more about webhooks



There are no GROQ-powered webhooks in this project

Made by creating a new webhook?

0 of 2 webhooks
(2 included in plan)

Get more webhooks

Webhooks

CORS origins

Tokens

Tokens

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

Name

Examples: "Employee import", "Website preview" or "PDF generator".

day3

Permissions

Choose the access privileges for the token.

Contributor

☐ Read and write access to draft content within all datasets, with no access to project settings. (Tokens: read+write drafts)

Deploy Studio (Token only)

☐ Access to deploy Sanity Studio and GraphQL APIs to our hosted service.

Developer

☒ Read and write access to all datasets, with access to project settings for developers. (Tokens: read+write)

Editor

☐ Read and write access to all datasets, with limited access to project settings.

new

Create Content Mapping, Visual Editing, Content Releases

Generate a Token

Webhooks

CORS origins

Tokens

Tokens

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

NAME	PERMISSIONS	CREATED
day3	Developer	0 seconds

Copy the token below – this is your only chance to do so!

skz24mJdbUITh1q93afhfAWh9J80Evks7IFiO03DjYJ3TbGMVDtCkEosI5P718fVD3Lo7bzT8LzzIfU7ITtn17bgwFARJ5LXsLa2XY1XkaYdb5ZY58sbNpbrIHfputGeUgU4d32Q6vdr36gyVKt4q0bZUvqxcQ93hNhVPN2ZwSxiS7R8ICQr

● MIGRATING DATA TO SANITY

Once the project was initialized, I proceeded to migrate existing content into Sanity. I created schemas to define the content structure:

Create a Schema

HACKATHON-GIAIC

next

node_modules

public

scripts

data-migration.mjs

src

api

fetch-and-insert.js

app

sanity

lib

schemaTypes

index.ts

product.ts

env.ts

structure.ts

env.local

eslintrc.json

gitignore

next-env.d.ts

next.config.js

package-lock.json

package.json

postcss.config.js

README.md

sanity.cli.ts

sanity.config.ts

tailwind.config.ts

tsconfig.json

OUTLINE

TIMELINE

src > sanity > schemaTypes > TS product.ts > default

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

```
export default {
  name: 'product',
  type: 'document',
  title: 'Product',
  fields: [
    {
      name: 'name',
      type: 'string',
      title: 'Name',
    },
    {
      name: 'image',
      type: 'image',
      title: 'Image',
      options: {
        hotspot: true, // Enable image cropping
      },
    },
    {
      name: 'price',
      type: 'number',
      title: 'Price',
    },
    {
      name: 'description',
      type: 'text',
      title: 'Description',
    },
    {
      name: 'discountPercentage',
      type: 'number',
      title: 'Discount Percentage',
    },
    {
      name: 'isFeaturedProduct',
      type: 'boolean',
      title: 'Featured Product'
    }
  ]
}
```

```

HACKATHON-GIAIC  scripts > JS data-migration.mjs > client > token
> .next
> node_modules
> public
> scripts
  JS data-migration.mjs U
    > src
    > api
    JS fetch-and-insert.js U
    > app
    > sanity
    > lib
    > schemaTypes
      TS index.ts U
      TS product.ts U
      TS env.ts U
      TS structure.ts U
    $ .env.local
    .eslintrc.json
    .gitignore
    TS next-env.d.ts
    JS next.config.js M
    {} package-lock.json M
    {} package.json M
    JS postcss.config.js
    ① README.md
    TS sanity.d.ts U
    TS sanity.config.ts U
    TS tailwind.config.ts
    tsconfig.json M
  > OUTLINE
  > TIMELINE

1 import { createClient } from '@sanity/client';
2 import fetch from 'node-fetch';
3
4 // Initialize Sanity client
5 const client = createClient({
6   projectId: "qc9ei68n",
7   dataset: "production",
8   useCdn: false, // Set to true if you want faster reads
9   apiVersion: '2025-01-13',
10   token: "skz24m3dbUITH1q93afhFAWH9380Evks7IF1003DJY3TbGMYDTcKEos15P718fV03Lo7bzT8LzzIfu7ITtnI7bgwfArJ5LXstLa2XY1XkaYdb5ZY58sbNpbrTH";
11 });
12
13 // Function to upload an image to Sanity
14 Tabnine | Edit | Test | Explain | Document
15 async function uploadImageToSanity(imageUrl) {
16   try {
17     console.log('uploading image: ${imageUrl}');
18
19     const response = await fetch(imageUrl);
20     if (!response.ok) {
21       throw new Error('Failed to fetch image: ${imageUrl}');
22     }
23
24     const buffer = await response.arrayBuffer();
25     const bufferImage = Buffer.from(buffer);
26
27     const asset = await client.assets.upload('image', bufferImage, {
28       filename: imageUrl.split('/').pop(),
29     });
30
31     console.log('Image uploaded successfully: ${asset._id}');
32     return asset._id;
33   } catch (error) {
34     console.error('Failed to upload image:', imageUrl, error);
35     return null;
36   }
37 }
```

```

HACKATHON-GIAIC  scripts > JS data-migration.mjs > client > token
> .next
> node_modules
> public
> scripts
  JS data-migration.mjs U
    > src
    > api
    JS fetch-and-insert.js U
    > app
    > sanity
    > lib
    > schemaTypes
      TS index.ts U
      TS product.ts U
      TS env.ts U
      TS structure.ts U
    $ .env.local
    .eslintrc.json
    .gitignore
    TS next-env.d.ts
    JS next.config.js M
    {} package-lock.json M
    {} package.json M
    JS postcss.config.js
    ① README.md
    TS sanity.d.ts U
    TS sanity.config.ts U
    TS tailwind.config.ts
    tsconfig.json M
  > OUTLINE
  > TIMELINE

39 Tabnine | Edit | Test | Explain | Document
40 async function uploadProduct(product) {
41   try {
42     const imageId = await uploadImageToSanity(product.imagePath);
43
44     if (imageId) {
45       const document = {
46         _type: 'product',
47         id: product.id,
48         name: product.name,
49         image: {
50           _type: 'image',
51           asset: {
52             _ref: imageId,
53           },
54         },
55         price: parseFloat(product.price), // Ensure the price is a number
56         description: product.description,
57         discountPercentage: product.discountPercentage,
58         isFeaturedProduct: product.isFeaturedProduct,
59         stockLevel: product.stockLevel,
60         category: product.category,
61       };
62
63       const createdProduct = await client.create(document);
64       console.log('Product "${product.name}" uploaded successfully:', createdProduct);
65     } else {
66       console.log('Product "${product.name}" skipped due to image upload failure.');
```



```

HACKATHON-GIAIC
├── .next
├── node_modules
├── public
├── scripts
│   └── JS data-migration.mjs
├── src
│   ├── api
│   │   ├── JS fetch-and-insert.js
│   │   ├── app
│   │   │   ├── sanity
│   │   │   └── lib
│   │   ├── schemaTypes
│   │   │   ├── TS indexes
│   │   │   ├── TS products.ts
│   │   │   ├── TS env.ts
│   │   │   └── TS structure.ts
│   │   ├── .env.local
│   │   ├── .eslintrc.json
│   │   ├── .gitignore
│   │   ├── TS next-env.d.ts
│   │   ├── JS next.config.js
│   │   ├── {} package-lock.json
│   │   ├── {} package.json
│   │   ├── JS postcss.config.js
│   │   ├── README.md
│   │   ├── TS sanity.d.ts
│   │   ├── TS sanity.config.ts
│   │   ├── TS tailwind.config.ts
│   │   └── tsconfig.json
└── ...

scripts > JS data-migration.mjs > client > token
39  async function uploadProduct(product) {
67  } catch (error) {
68  | console.error('Error uploading product:', error);
69  }
70  }
71
72  // Function to fetch products from the provided API and upload them to Sanity
73  Tabnine | Edit | Test | Explain | Document
74  async function migrateProducts() {
75  | try {
76  |   const response = await fetch('https://template-0-beta.vercel.app/api/product');
77  |   if (!response.ok) {
78  |     throw new Error(`HTTP error! Status: ${response.status}`);
79  |   }
80  |   const products = await response.json();
81  |   for (const product of products) {
82  |     await uploadProduct(product);
83  |   }
84  | } catch (error) {
85  |   console.error('Error fetching products:', error);
86  | }
87  }
88  }
89
90  // Start the migration
91  migrateProducts();
92

```

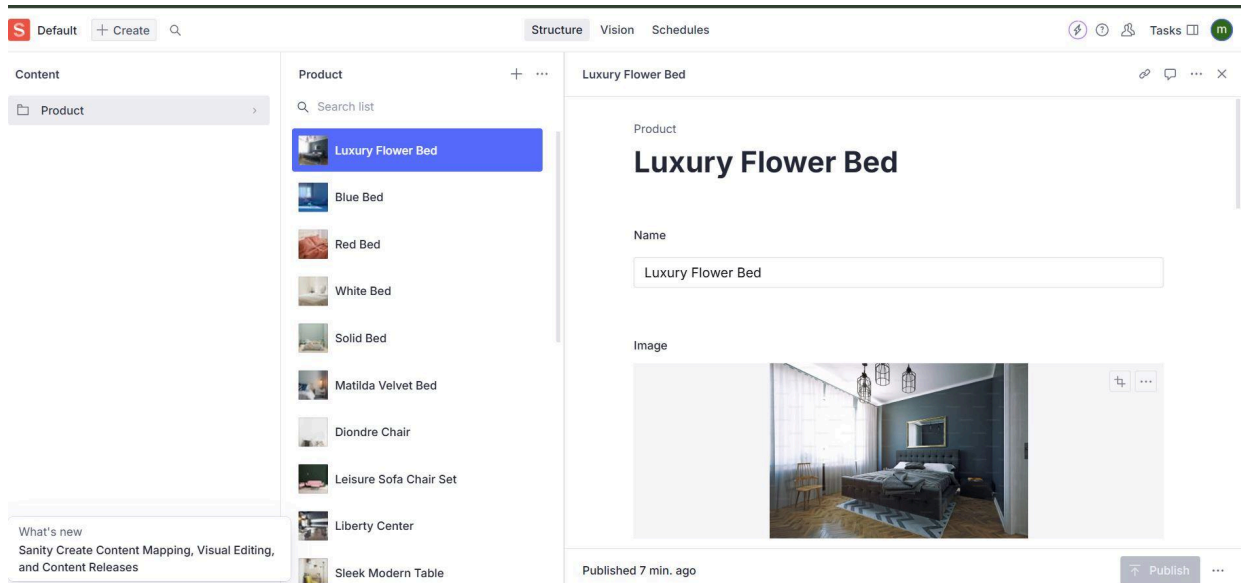
```

category: 'Bed',
description: 'A modern cantilever chair with a unique floating effect.',
discountPercentage: 15,
id: '20',
image: {
  _type: 'image',
  asset: {
    _ref: 'image-c02efe31cc7d519664e3b4c4d4608a8b812109e2-1932x1087-jpg'
  }
},
isFeaturedProduct: true,
name: 'Blue Bed',
price: 780,
stockLevel: 5
}

Uploading image: https://plus.unsplash.com/premium_photo-1661886455585-03ddf4347796?q=80&w=2059&auto=format&fit=crop&ixl=1
Image uploaded successfully: image-432c20d55e31514d09985305e657cb342317d57b-2059x1391-jpg
Product "Luxury Flower Bed" uploaded successfully: {
  _createdAt: '2025-01-19T20:23:54Z',
  _id: 'AJVJVObFUPWewZSbi6TYfx',
  _rev: 'AJVJVObFUPWewZSbi6TYf6',
  _type: 'product',
  _updatedAt: '2025-01-19T20:23:54Z',
  category: 'Bed',
  description: 'A luxurious shell-shaped chair with gold brass metal legs.',
  discountPercentage: 0,
  id: '21',
  image: {
    _type: 'image',
    asset: {
      _ref: 'image-432c20d55e31514d09985305e657cb342317d57b-2059x1391-jpg'
    }
  },
  isFeaturedProduct: true,
  name: 'Luxury Flower Bed',
  price: 2500,
  stockLevel: 2
}

E:\hackathon-giaic>

```



● FETCHING DATA FROM SANITY IN NEXT JS

```

HACKATHON-GIAIC
├── .next
├── node_modules
├── public
├── scripts
├── JS data-migration.mjs
├── src
│   ├── api
│   ├── app
│   │   ├── about
│   │   ├── account
│   │   ├── cart
│   │   ├── checkout
│   │   ├── components
│   │   ├── contact
│   │   ├── images
│   │   ├── product
│   │   └── shop
│   │       ├── page.tsx
│   │       ├── studio
│   │       ├── globals.css
│   │       ├── layout.tsx
│   │       ├── page.tsx
│   │       └── sanity
│   └── lib
└── ...

src > app > shop > page.tsx > page > products.map() callback
21
22
23 type Product={
24   _id:string;
25   name:string;
26   description:string;
27   price:number;
28   imageUrl:string;
29 }
30
31
32
33 Tabnine | Edit | Test | Explain | Document
34 export default async function page() {
35
36   const products:Product[] = await sanityFetch({query:allproducts})
37
38   console.log(products)
39
40
41 > // const products=[ ...
114 // ]
115
116
117
118 return (
119   <div className='mt-[60px] bg-white' >

```