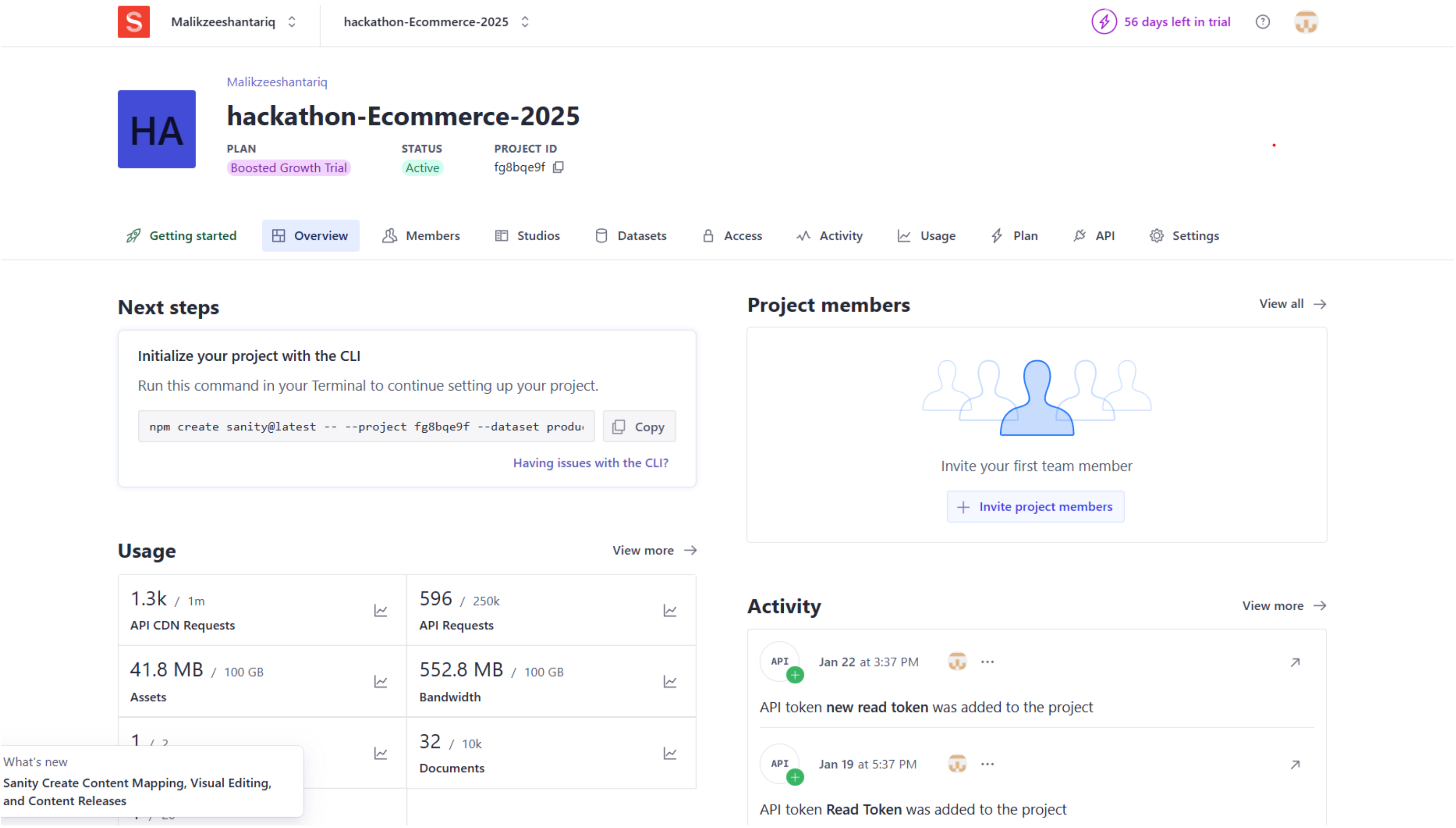
Document Title: "Day 3 - API Integration Report - AVION

Create Sanity Project



Malikzeeshantariq

hackathon-Ecommerce-2025

PLAN: Boosted Growth Trial

STATUS: Active

PROJECT ID: fg8bqe9f

Next steps

Initialize your project with the CLI

Run this command in your Terminal to continue setting up your project.

```
npm create sanity@latest -- --project fg8bqe9f --dataset produ
```

Project members

Invite your first team member

Usage

1.3k / 1m
API CDN Requests

596 / 250k
API Requests

41.8 MB / 100 GB
Assets

552.8 MB / 100 GB
Bandwidth

1 / 2

32 / 10k
Documents

What's new

Sanity Create Content Mapping, Visual Editing, and Content Releases

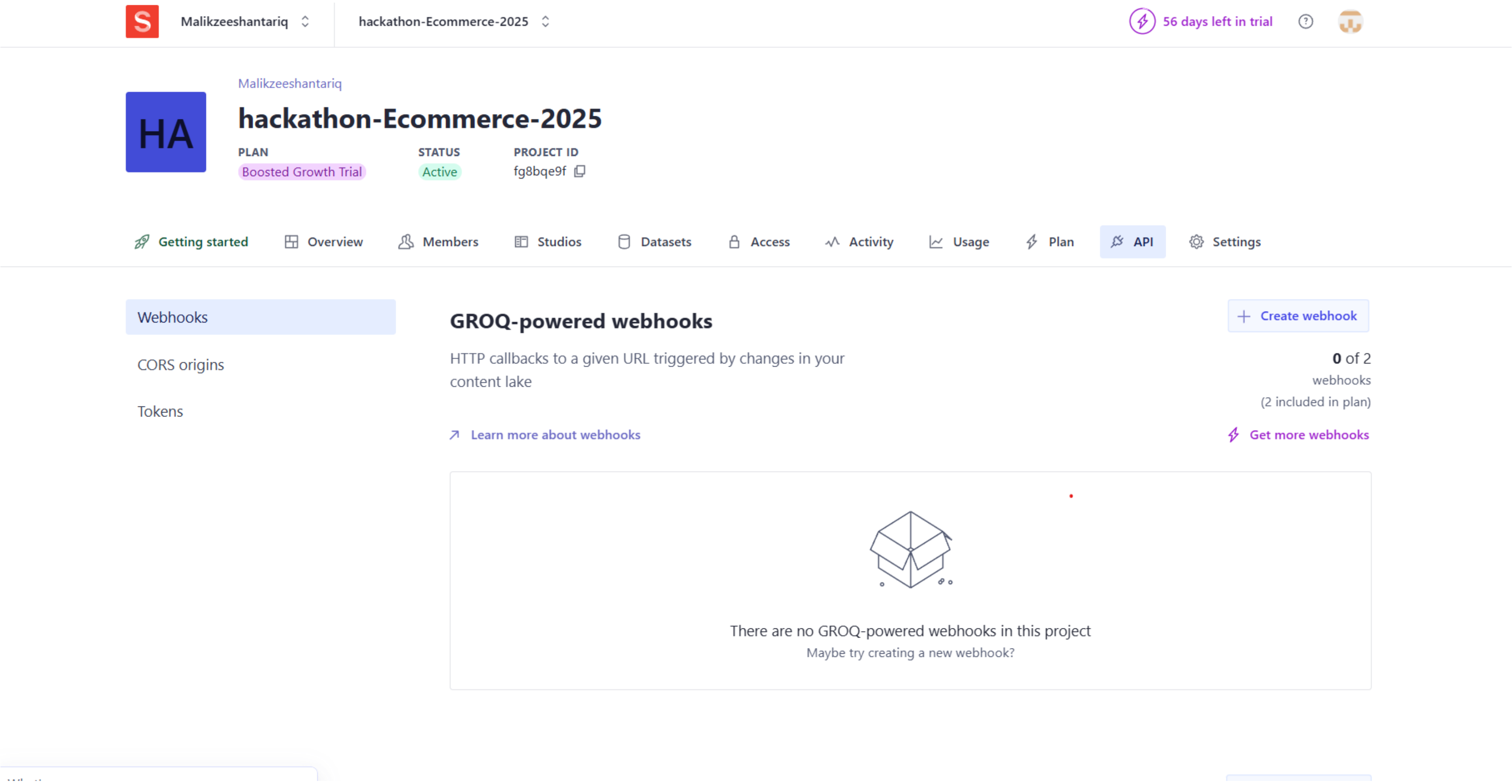
Activity

API — Jan 22 at 3:37 PM

API token **new read token** was added to the project

API — Jan 19 at 5:37 PM

API token **Read Token** was added to the project



Malikzeeshantariq

hackathon-Ecommerce-2025

PLAN: Boosted Growth Trial

STATUS: Active

PROJECT ID: fg8bqe9f

GROQ-powered webhooks

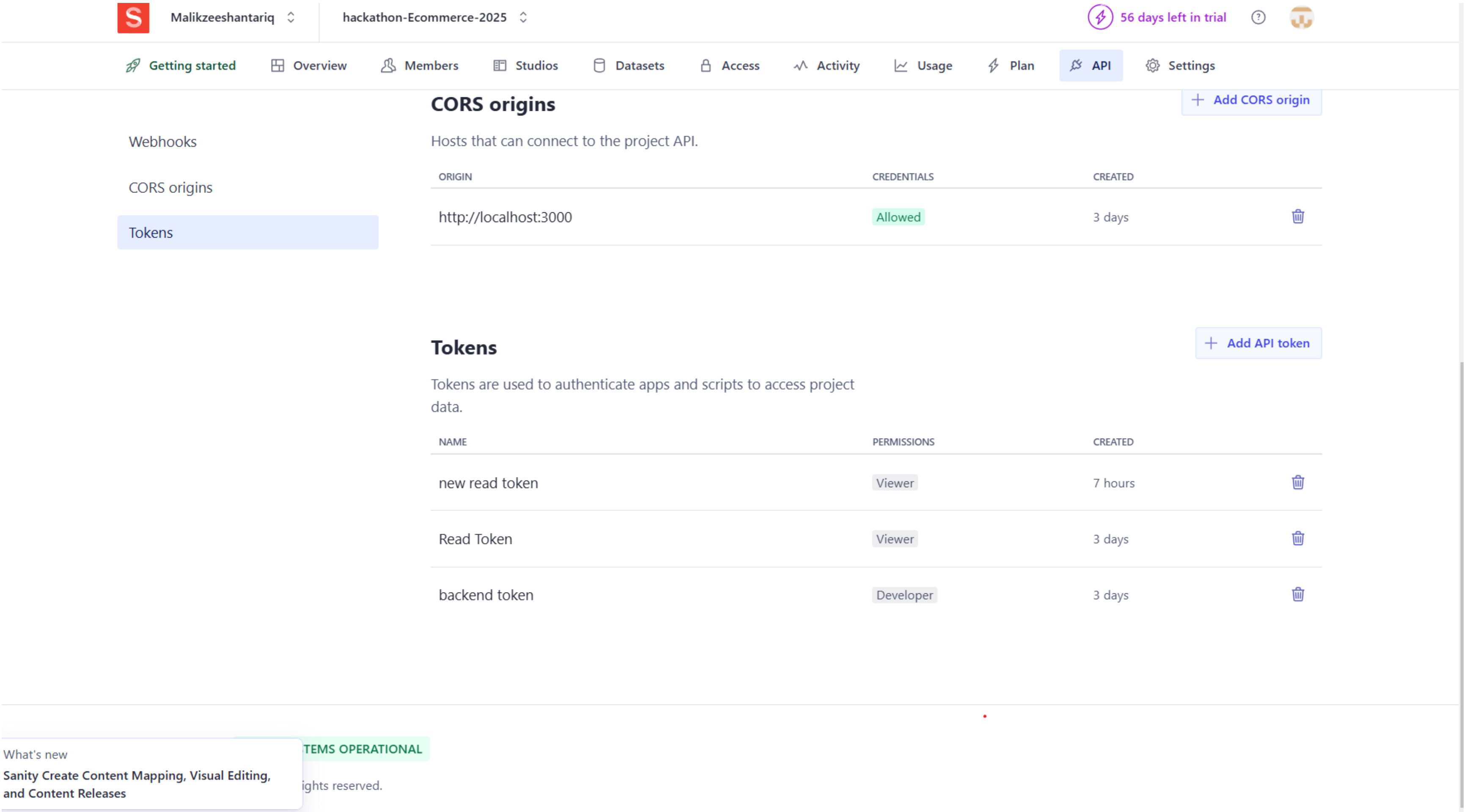
HTTP callbacks to a given URL triggered by changes in your content lake

0 of 2 webhooks (2 included in plan)

There are no GROQ-powered webhooks in this project

Maybe try creating a new webhook?

Added Token



CORS origins

Hosts that can connect to the project API.

ORIGIN	CREDENTIALS	CREATED
http://localhost:3000	Allowed	3 days

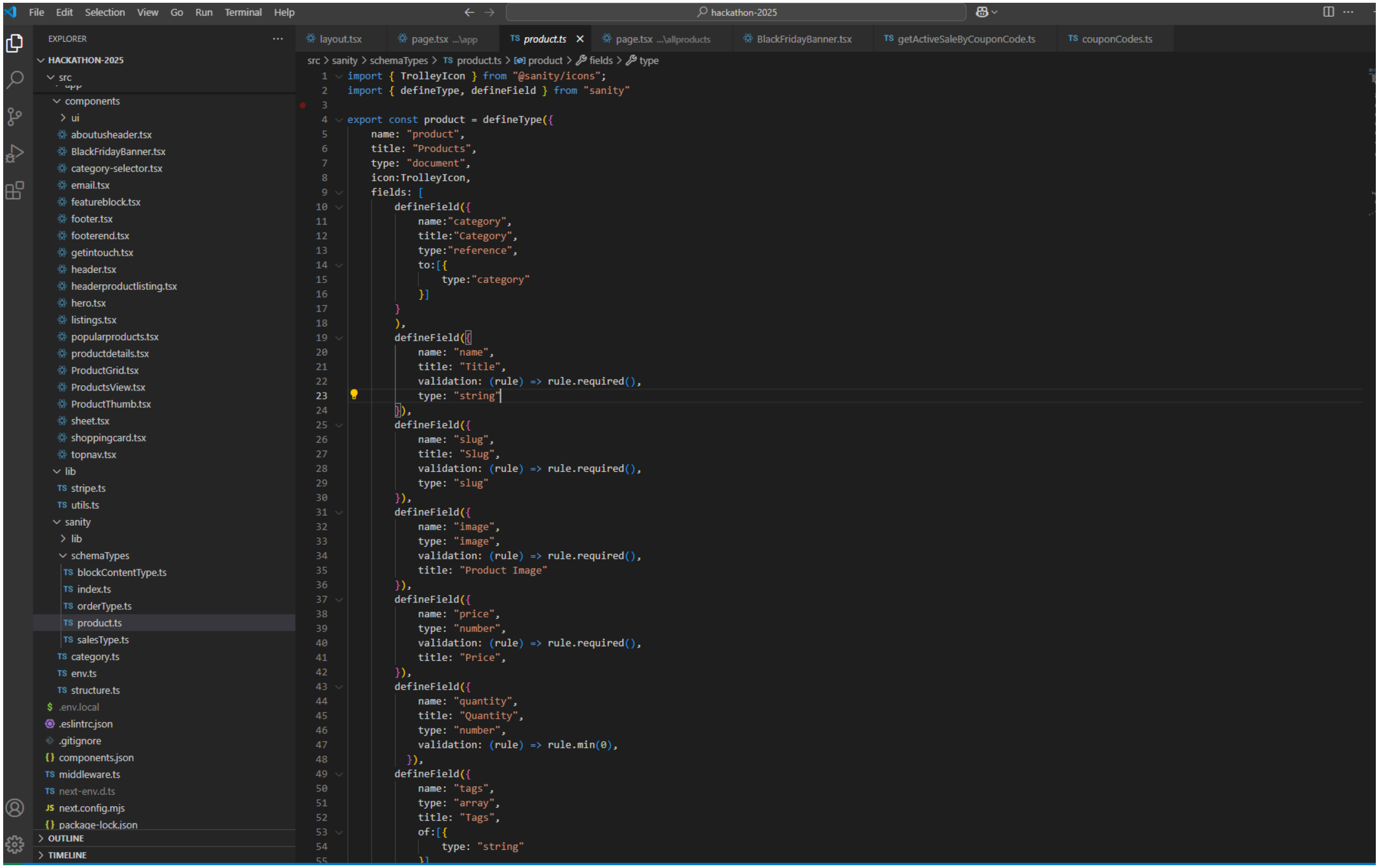
Tokens

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

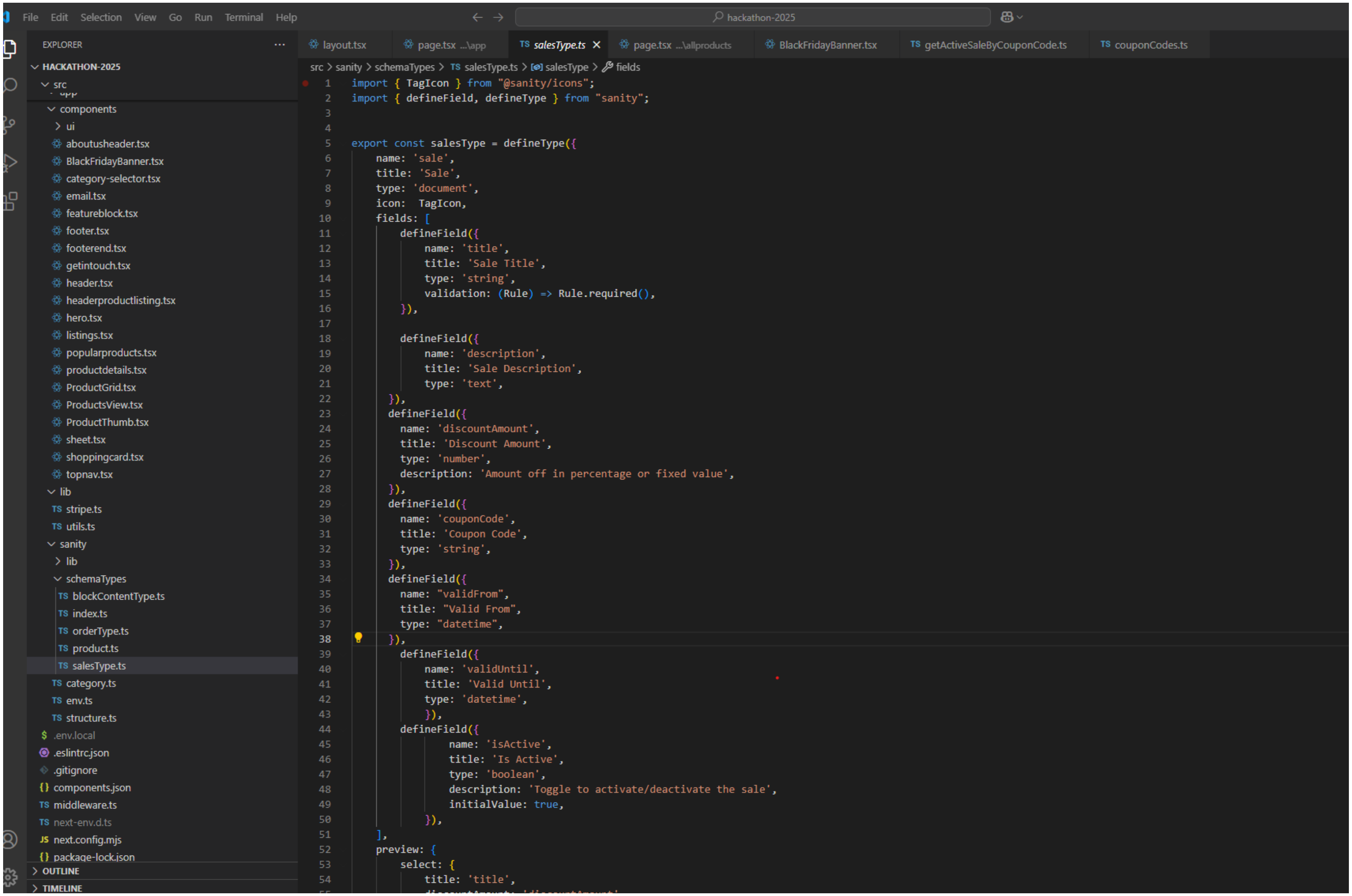
NAME	PERMISSIONS	CREATED
new read token	Viewer	7 hours
Read Token	Viewer	3 days
backend token	Developer	3 days

What's new
Sanity Create Content Mapping, Visual Editing, and Content Releases
TEMS OPERATIONAL
ights reserved.

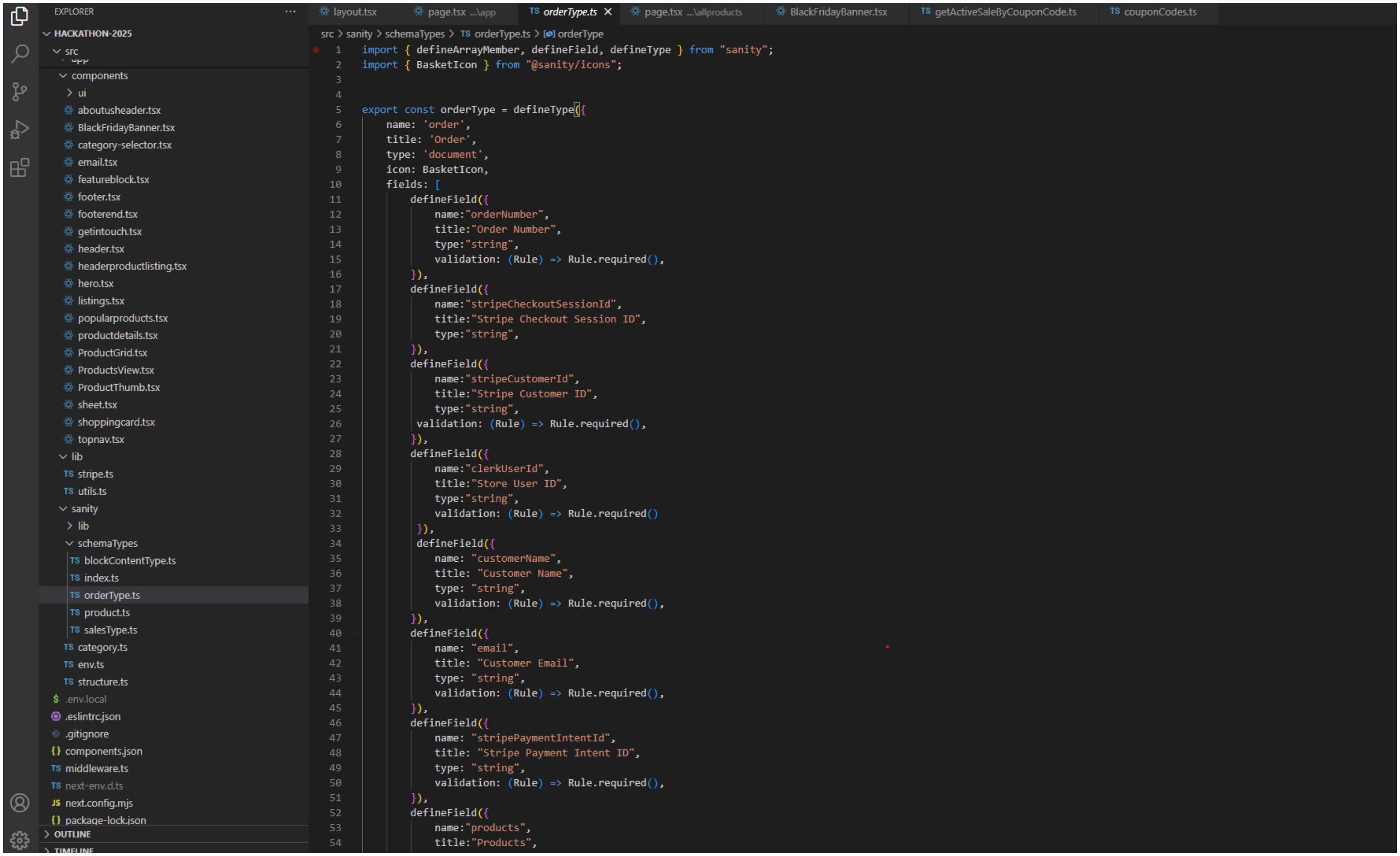
Product Schema Type



```
import { TrolleyIcon } from "@sanity/icons";
import { defineType, defineField } from "sanity"

export const product = defineType({
    name: "product",
    title: "Products",
    type: "document",
    icon:TrolleyIcon,
    fields: [
        defineField({
            name:"category",
            title:"Category",
            type:"reference",
            to:[{
                type:"category"
            }]
        }
        ),
        defineField({
            name: "name",
            title: "Title",
            validation: (rule) => rule.required(),
            type: "string"
        }),
        defineField({
            name: "slug",
            title: "Slug",
            validation: (rule) => rule.required(),
            type: "slug"
        }),
        defineField({
            name: "image",
            type: "image",
            validation: (rule) => rule.required(),
            title: "Product Image"
        }),
        defineField({
            name: "price",
            type: "number",
            validation: (rule) => rule.required(),
            title: "Price",
        }),
        defineField({
            name: "quantity",
            title: "Quantity",
            type: "number",
            validation: (rule) => rule.min(0),
          }),
        defineField({
            name: "tags",
            type: "array",
            title: "Tags",
            of:[{
                type: "string"
            }]
```

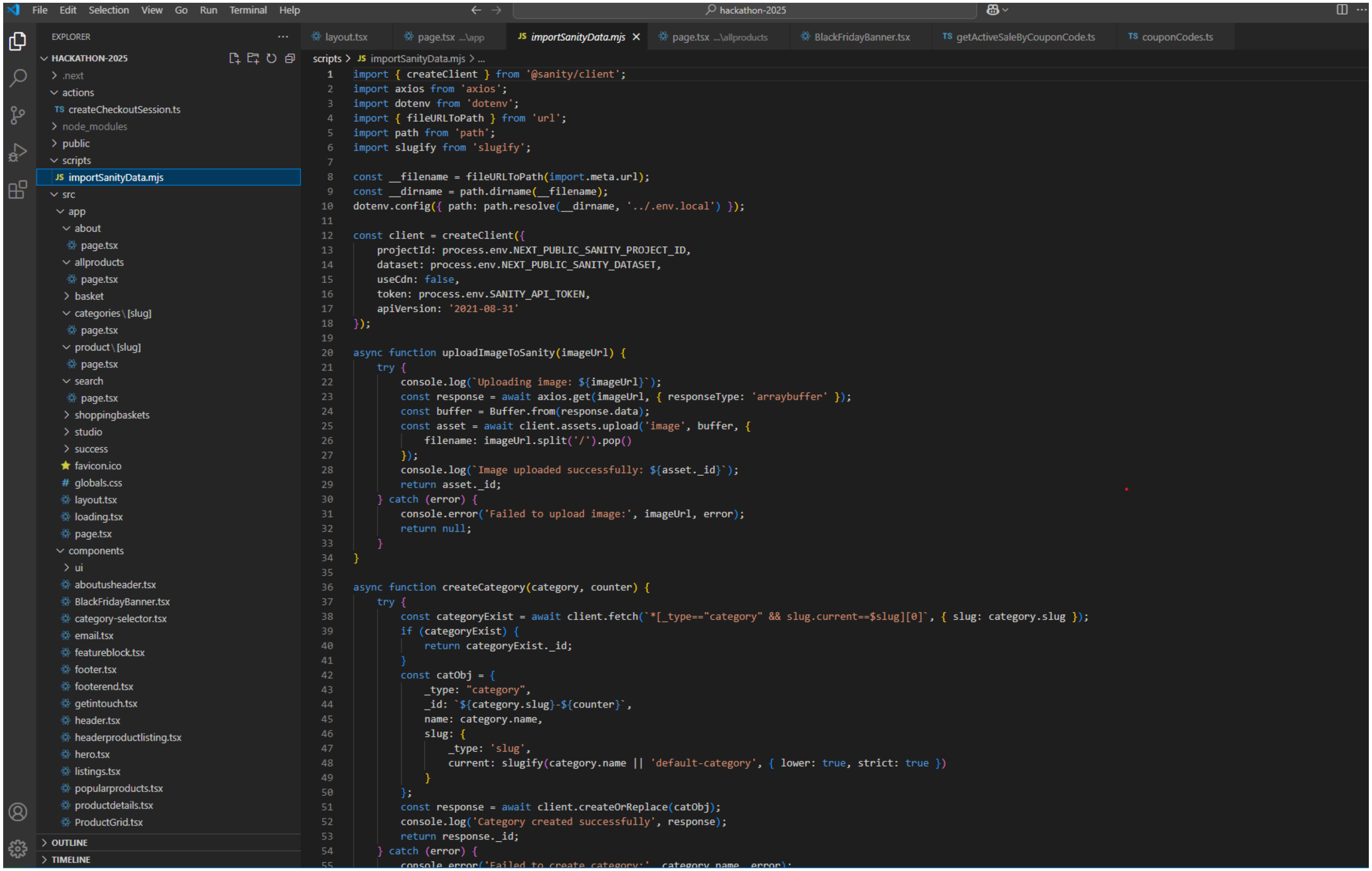


```
import { TagIcon } from "@sanity/icons";
import { defineField, defineType } from "sanity";


export const salesType = defineType({
    name: 'sale',
    title: 'Sale',
    type: 'document',
    icon:  TagIcon,
    fields: [
        defineField({
            name: 'title',
            title: 'Sale Title',
            type: 'string',
            validation: (Rule) => Rule.required(),
        }),

        defineField({
            name: 'description',
            title: 'Sale Description',
            type: 'text',
      }),
      defineField({
        name: 'discountAmount',
        title: 'Discount Amount',
        type: 'number',
        description: 'Amount off in percentage or fixed value',
      }),
      defineField({
        name: 'couponCode',
        title: 'Coupon Code',
        type: 'string',
      }),
      defineField({
        name: "validFrom",
        title: "Valid From",
        type: "datetime",
      }),
        defineField({
            name: 'validUntil',
            title: 'Valid Until',
            type: 'datetime',
            }),
        defineField({
                name: 'isActive',
                title: 'Is Active',
                type: 'boolean',
                description: 'Toggle to activate/deactivate the sale',
                initialValue: true,
            }),
    ],
    preview: {
        select: {
            title: 'title',
            discountAmount: 'discountAmount'
```

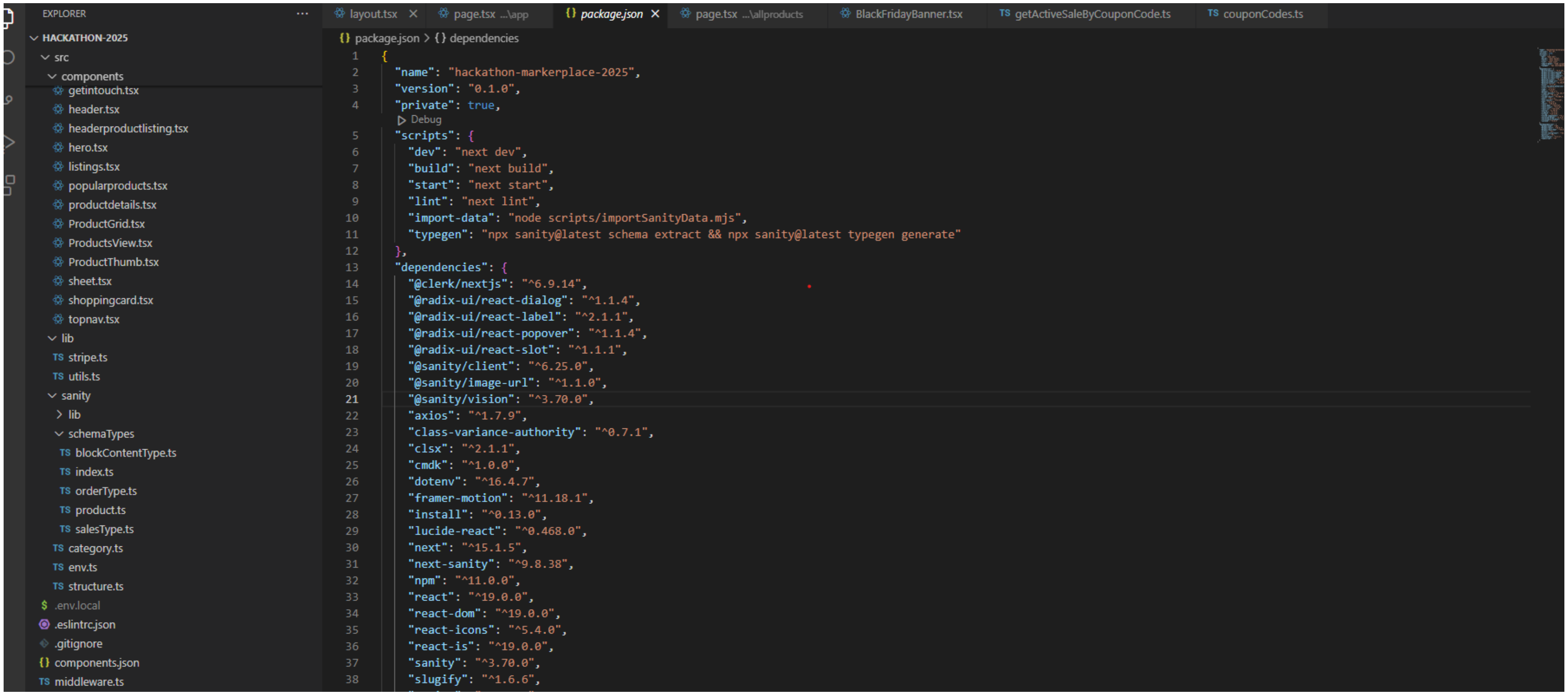


```ts
import { defineArrayMember, defineField, defineType } from "sanity";
import { BasketIcon } from "@sanity/icons";


export const orderType = defineType({
    name: 'order',
    title: 'Order',
    type: 'document',
    icon: BasketIcon,
    fields: [
        defineField({
            name:"orderNumber",
            title:"Order Number",
            type:"string",
            validation: (Rule) => Rule.required(),
        }),
        defineField({
            name:"stripeCheckoutSessionId",
            title:"Stripe Checkout Session ID",
            type:"string",
        }),
        defineField({
            name:"stripeCustomerId",
            title:"Stripe Customer ID",
            type:"string",
         validation: (Rule) => Rule.required(),
        }),
        defineField({
            name:"clerkUserId",
            title:"Store User ID",
            type:"string",
            validation: (Rule) => Rule.required()
         }),
         defineField({
            name: "customerName",
            title: "Customer Name",
            type: "string",
            validation: (Rule) => Rule.required(),
        }),
        defineField({
            name: "email",
            title: "Customer Email",
            type: "string",
            validation: (Rule) => Rule.required(),
        }),
        defineField({
            name: "stripePaymentIntentId",
            title: "Stripe Payment Intent ID",
            type: "string",
            validation: (Rule) => Rule.required(),
        }),
        defineField({
            name:"products",
            title:"Products",
```

Import Data into Sanity Template-02 using ImportScript

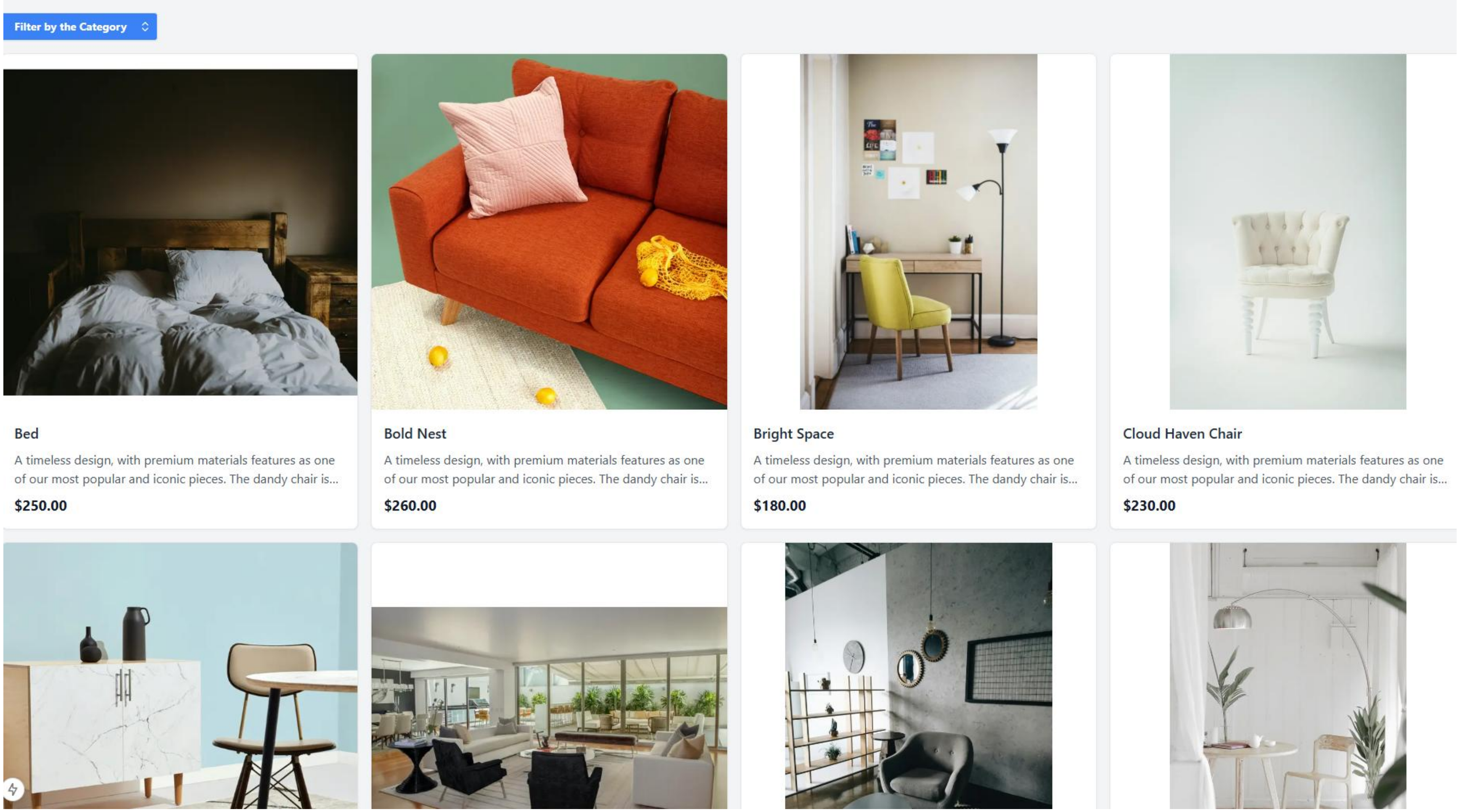


```javascript
import { createClient } from '@sanity/client';
import axios from 'axios';
import dotenv from 'dotenv';
import { fileURLToPath } from 'url';
import path from 'path';
import slugify from 'slugify';

const __filename = fileURLToPath(import.meta.url);
const __dirname = path.dirname(__filename);
dotenv.config({ path: path.resolve(__dirname, '../.env.local') });

const client = createClient({
    projectId: process.env.NEXT_PUBLIC_SANITY_PROJECT_ID,
    dataset: process.env.NEXT_PUBLIC_SANITY_DATASET,
    useCdn: false,
    token: process.env.SANITY_API_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 createCategory(category, counter) {
    try {
        const categoryExist = await client.fetch(`*[_type=="category" && slug.current==$slug][0]`, { slug: category.slug });
        if (categoryExist) {
            return categoryExist._id;
        }
        const catObj = {
            _type: "category",
            _id: `${category.slug}-${counter}`,
            name: category.name,
            slug: {
                _type: 'slug',
                current: slugify(category.name || 'default-category', { lower: true, strict: true })
            }
        };
        const response = await client.createOrReplace(catObj);
        console.log('Category created successfully', response);
        return response._id;
    } catch (error) {
        console.error('Failed to create category:', category.name, error);
```



```json
{
  "name": "hackathon-markerplace-2025",
  "version": "0.1.0",
  "private": true,
  "scripts": {
    "dev": "next dev",
    "build": "next build",
    "start": "next start",
    "lint": "next lint",
    "import-data": "node scripts/importSanityData.mjs",
    "typegen": "npx sanity@latest schema extract && npx sanity@latest typegen generate"
  },
  "dependencies": {
    "@clerk/nextjs": "^6.9.14",
    "@radix-ui/react-dialog": "^1.1.4",
    "@radix-ui/react-label": "^2.1.1",
    "@radix-ui/react-popover": "^1.1.4",
    "@radix-ui/react-slot": "^1.1.1",
    "@sanity/client": "^6.25.0",
    "@sanity/image-url": "^1.1.0",
    "@sanity/vision": "^3.70.0",
    "axios": "^1.7.9",
    "class-variance-authority": "^0.7.1",
    "clsx": "^2.1.1",
    "cmdk": "^1.0.0",
    "dotenv": "^16.4.7",
    "framer-motion": "^11.18.1",
    "install": "^0.13.0",
    "lucide-react": "^0.468.0",
    "next": "^15.1.5",
    "next-sanity": "^9.8.38",
    "npm": "^11.0.0",
    "react": "^19.0.0",
    "react-dom": "^19.0.0",
    "react-icons": "^5.4.0",
    "react-is": "^19.0.0",
    "sanity": "^3.70.0",
    "slugify": "^1.6.6",
```

Fetching data into Next. Js Application



Bed

A timeless design, with premium materials features as one of our most popular and iconic pieces. The dandy chair is...

$250.00

Bold Nest

A timeless design, with premium materials features as one of our most popular and iconic pieces. The dandy chair is...

$260.00

Bright Space

A timeless design, with premium materials features as one of our most popular and iconic pieces. The dandy chair is...

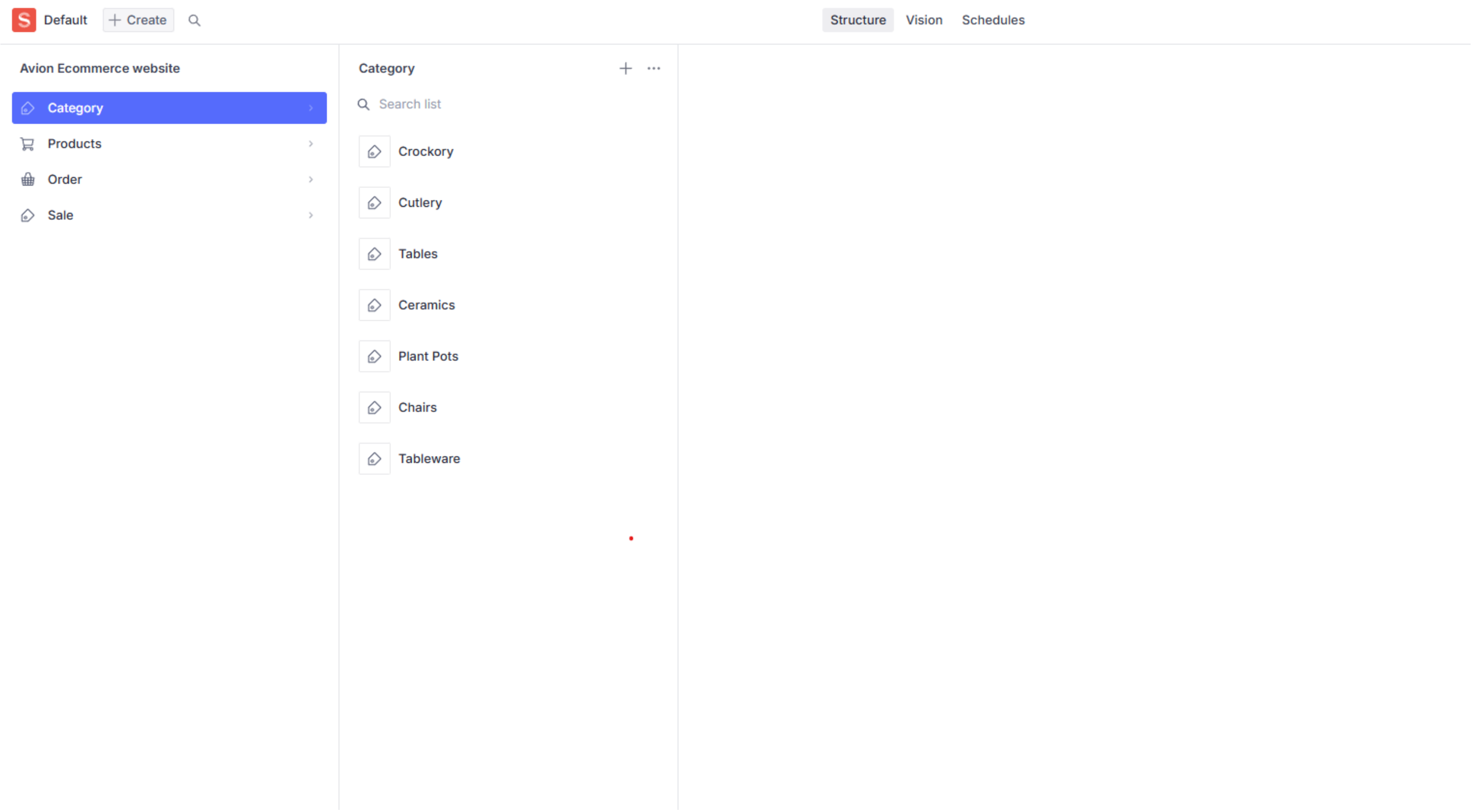
$180.00

Cloud Haven Chair

A timeless design, with premium materials features as one of our most popular and iconic pieces. The dandy chair is...

$230.00

Display Data in Sanity



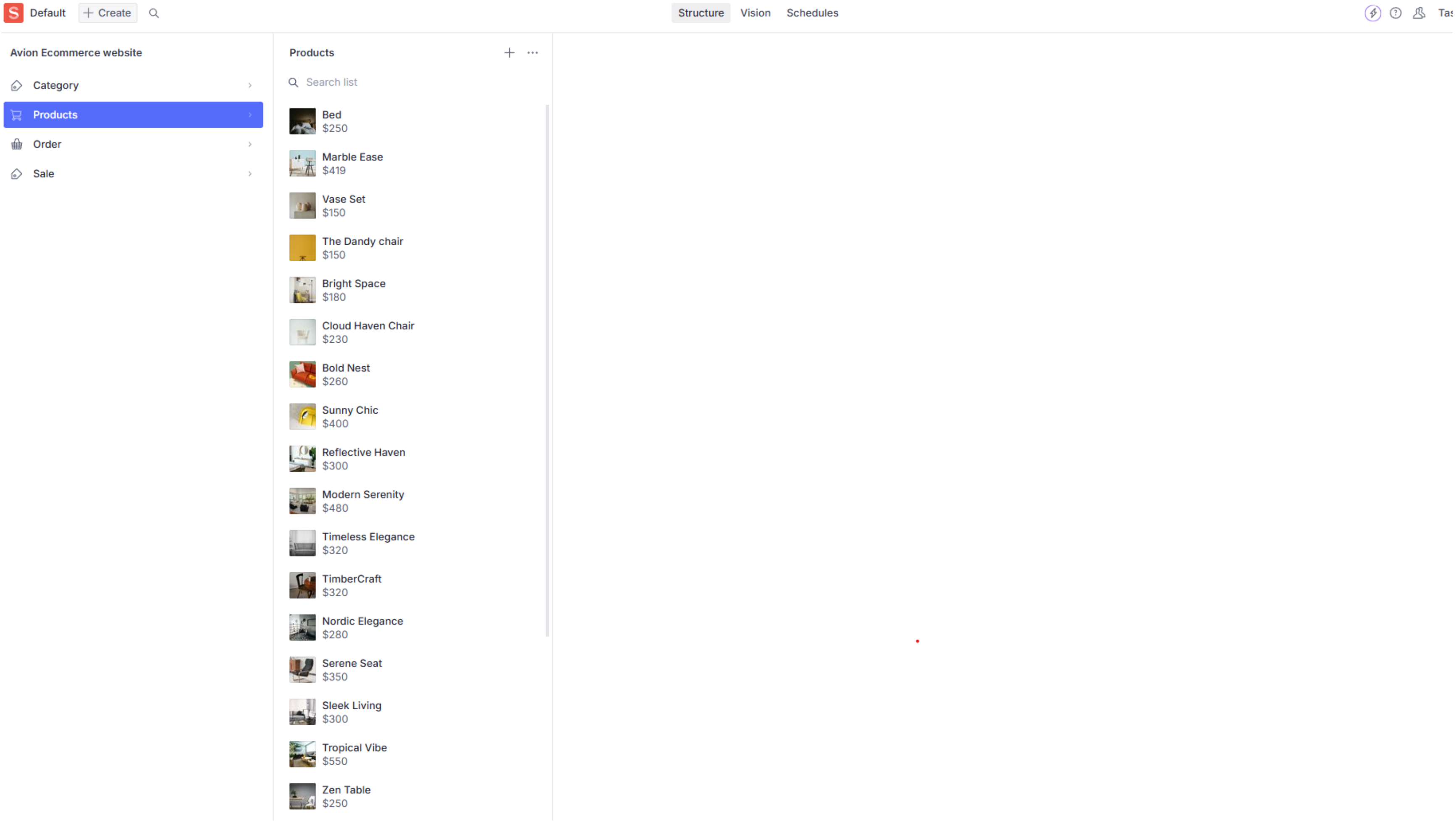
Avion Ecommerce website

- Category
- Products
- Order
- Sale

Category

Search list

- Crockory
- Cutlery
- Tables
- Ceramics
- Plant Pots
- Chairs
- Tableware



Avion Ecommerce website

- Category
- Products
- Order
- Sale

Products

Search list

Bed
$250

Marble Ease
$419

Vase Set
$150

The Dandy chair
$150

Bright Space
$180

Cloud Haven Chair
$230

Bold Nest
$260

Sunny Chic
$400

Reflective Haven
$300

Modern Serenity
$480

Timeless Elegance
$320

TimberCraft
$320

Nordic Elegance
$280

Serene Seat
$350

Sleek Living
$300

Tropical Vibe
$550

Zen Table
$250