Day 3 - API Integration Report - Avion

1. Manual Data Addition Process

Overview:
For the Avion project, I opted to manually add data into Sanity CMS instead of integrating APIs. This approach provided greater control over data structure, accuracy, and flexibility for the marketplace.

Steps:

- ❖ **Sanity CMS Schema Creation**:
  - ➢ Designed a schema called main to handle product details with the following structure:
    - ■ **Main Product Image**: To store the primary image for the main product.
    - ■ **Product Cards**: An array of product card objects, each containing details like ID, image, heading, price, description, dimensions, and features.
- ❖ **Data Entry**:
  - ➢ Navigated to the Sanity Studio interface.
  - ➢ Populated data for product cards, including:
    - ■ Product images, headings, and prices.
    - ■ Detailed descriptions and product dimensions.
    - ■ Key features of each product.
- ❖ **Verification**:
  - ➢ Verified the accuracy of data entries by previewing the populated fields in the Sanity CMS dashboard.

Tools Used:

- **Sanity Studio**: For manual data entry.

---

2. Adjustments Made to Schemas

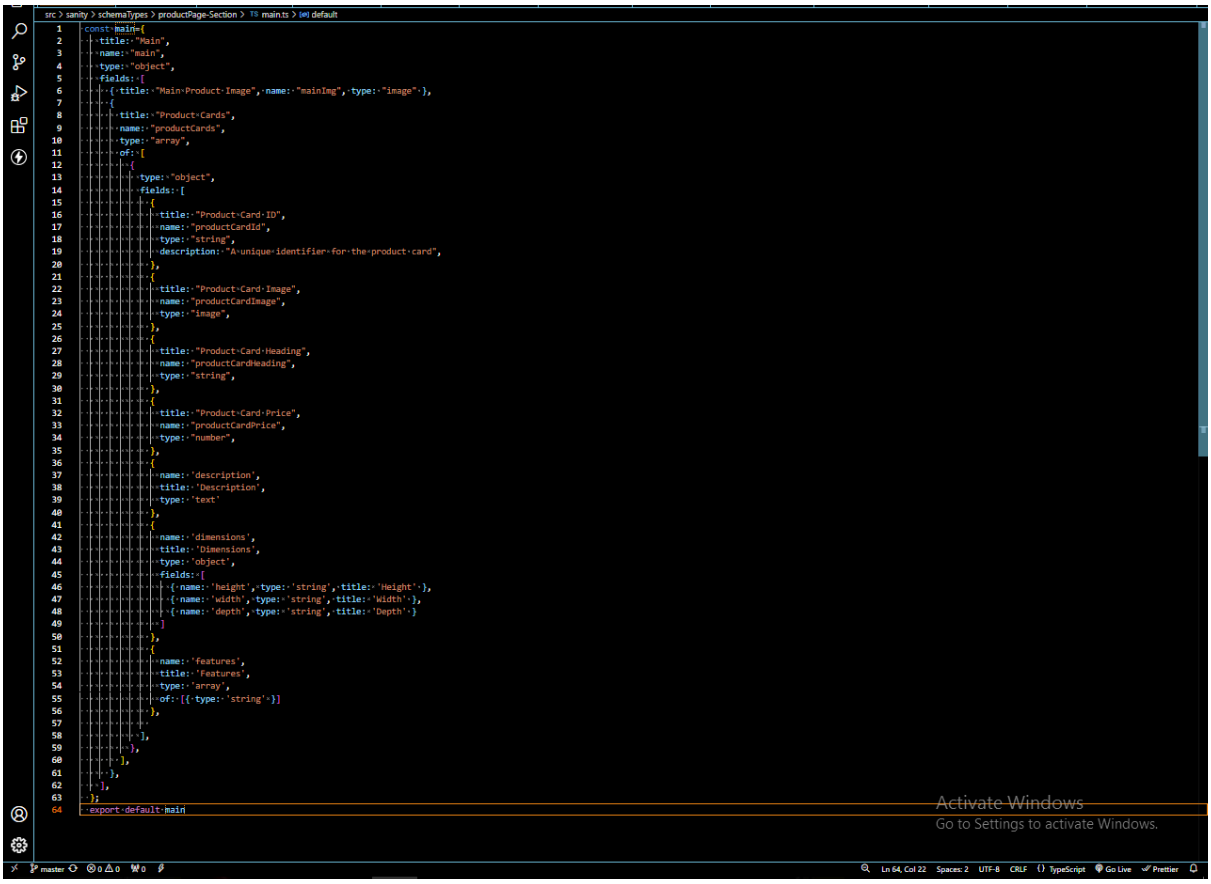
Overview:
To accommodate the manual data entry process, I created a detailed schema (main) tailored to the requirements of the Avion marketplace.

Schema Details:

- ❖ **Main Fields**:
  - ➢ maimIng: Stores the main product image.
  - ➢ productCards: An array of product objects, each with fields for detailed product information.
- ❖ **Product Card Fields**:
  - ➢ productCardId: Unique identifier for each product card.
  - ➢ productCardImage: Image of the product card.

- ➢ productCardHeading: Title or heading of the product.
- ➢ productCardPrice: Price of the product.
- ➢ description: Detailed description of the product.
- ➢ dimensions: Object containing height, width, and depth of the product.
- ➢ features: Array of key product features.

Schema:


src > sanity > schemaTypes > productPage-Section > TS main.ts > default
Activate Windows
Go to Settings to activate Windows.
master
Ln 64, Col 22
Spaces: 2
UTF-8
CRLF
TypeScript
Go Live
Prettier

```
const main={
  title: "Main",
  name: "main",
  type: "object",
  fields: [
    { title: "Main Product Image", name: "mainImg", type: "image" },
    {
      title: "Product Cards",
      name: "productCards",
      type: "array",
      of: [
        {
          type: "object",
          fields: [
            {
              title: "Product Card ID",
              name: "productCardId",
              type: "string",
              description: "A unique identifier for the product card",
            },
            {
              title: "Product Card Image",
              name: "productCardImage",
              type: "image",
            },
            {
              title: "Product Card Heading",
              name: "productCardHeading",
              type: "string",
            },
            {
              title: "Product Card Price",
              name: "productCardPrice",
              type: "number",
            },
            {
              name: 'description',
              title: 'Description',
              type: 'text'
            },
            {
              name: 'dimensions',
              title: 'Dimensions',
              type: 'object',
              fields: [
                { name: 'height', type: 'string', title: 'Height' },
                { name: 'width', type: 'string', title: 'Width' },
                { name: 'depth', type: 'string', title: 'Depth' }
              ]
            },
            {
              name: 'features',
              title: 'Features',
              type: 'array',
              of: [{ type: 'string' }]
            },

          ],
        },
      ],
    },
  ],
};
export default main
```

3. Migration Steps and Tools Used

Overview:
While no API migration was performed, the manual data entries in Sanity CMS are exportable for future updates or migrations if required.

Steps:

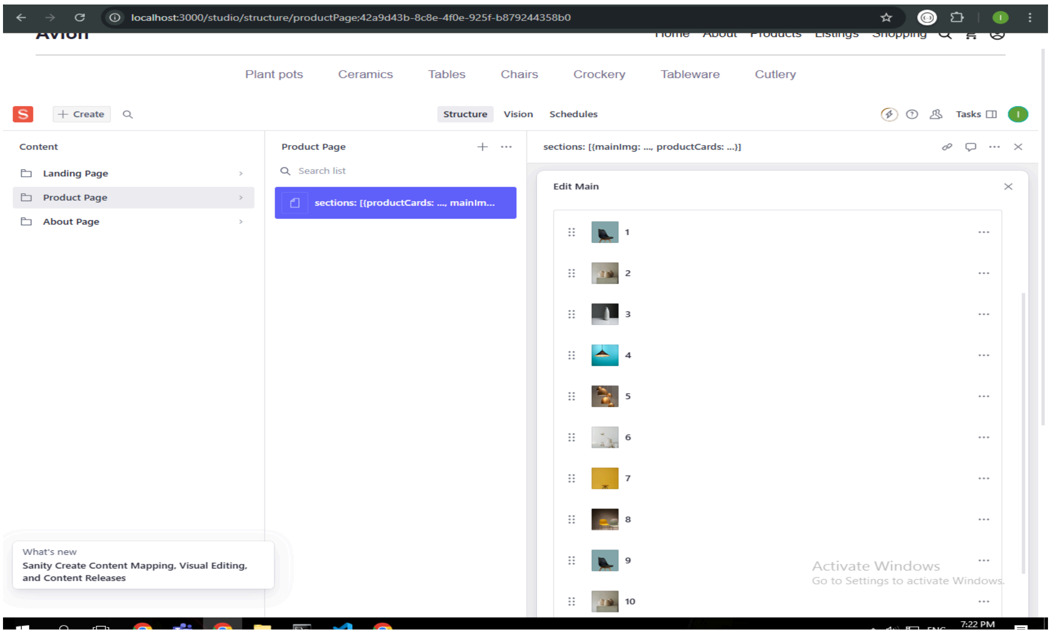
1. Exported data from Sanity CMS using sanity dataset export for backup.
2. Verified the exported JSON structure to ensure all fields were correctly populated.
3. Imported the dataset back to Sanity CMS for testing data consistency.

Tools Used:

- **Sanity CLI**: For dataset export and import.

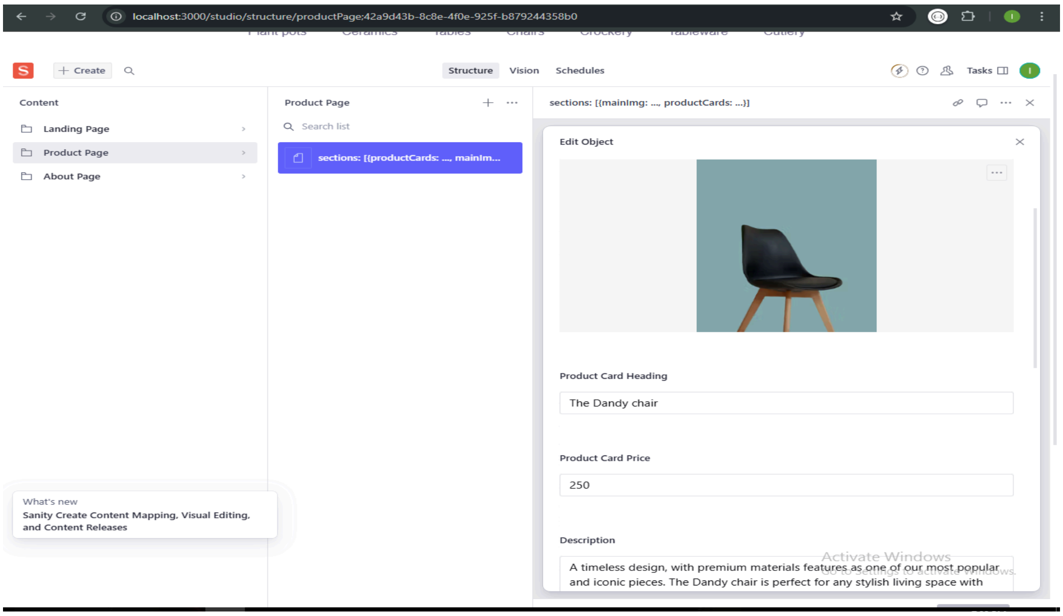
4.screenshots:

1.Sanity CMS Fields: Populated fields in Sanity Studio, showing products and their details.



localhost:3000/studio/structure/productPage;42a9d43b-8c8e-4f0e-925f-b879244358b0

Avion
Home About Products Listings Shopping
Plant pots Ceramics Tables Chairs Crockery Tableware Cutlery
Create
Structure Vision Schedules
Tasks
Content
Landing Page
Product Page
About Page
Product Page
Search list
sections: [{productCards: ..., mainIm...
sections: [{mainImg: ..., productCards: ...}]
Edit Main
1
2
3
4
5
6
7
8
9
10
What's new
Sanity Create Content Mapping, Visual Editing, and Content Releases
Activate Windows
Go to Settings to activate Windows.
7:22 PM



localhost:3000/studio/structure/productPage;42a9d43b-8c8e-4f0e-925f-b879244358b0

Plant pots Ceramics Tables Chairs Crockery Tableware Cutlery
Create
Structure Vision Schedules
Tasks
Content
Landing Page
Product Page
About Page
Product Page
Search list
sections: [{productCards: ..., mainIm...
sections: [{mainImg: ..., productCards: ...}]
Edit Object
What's new
Sanity Create Content Mapping, Visual Editing, and Content Releases
Activate Windows

Product Card Heading

The Dandy chair

Product Card Price

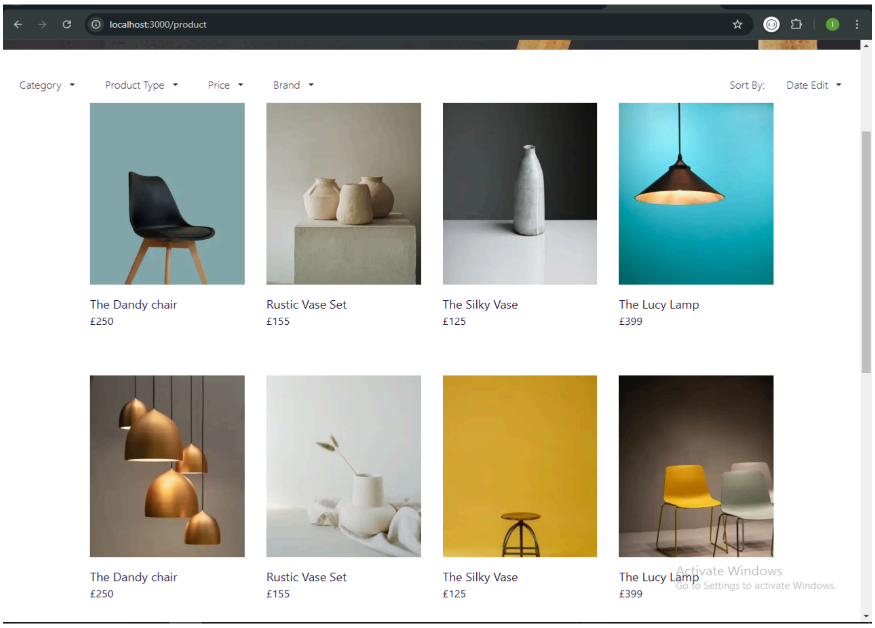
250

Description

A timeless design, with premium materials features as one of our most popular and iconic pieces. The Dandy chair is perfect for any stylish living space with

2.Frontend Display:Data fetched from Sanity CMS displayed on the Avion frontend.

Product page:



localhost:3000/product

Category Product Type Price Brand Sort By: Date Edit

The Dandy chair
£250

Rustic Vase Set
£155

The Silky Vase
£125

The Lucy Lamp
£399

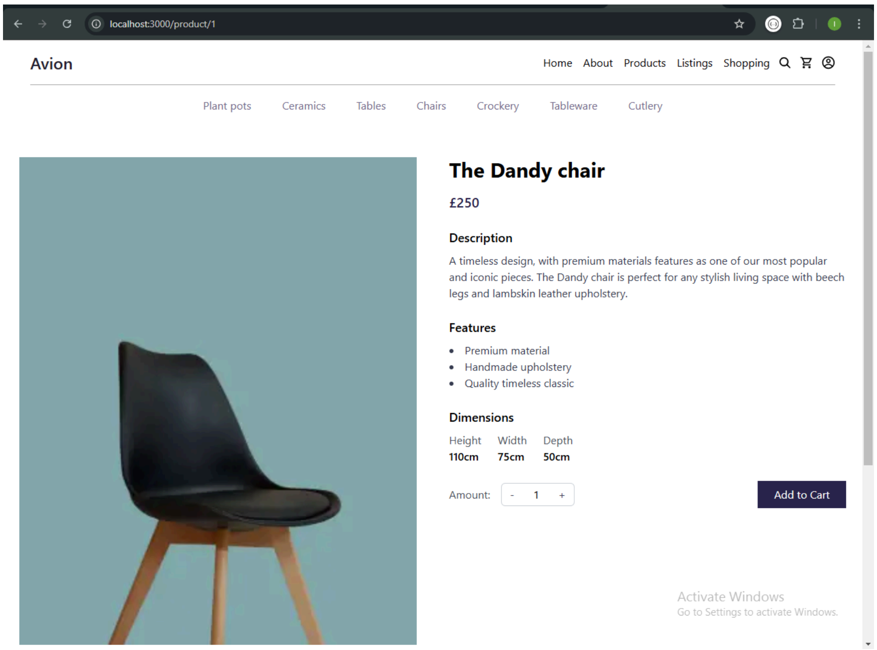
The Dandy chair
£250

Rustic Vase Set
£155

The Silky Vase
£125

The Lucy Lamp
£399

Dynamic page:



localhost:3000/product/1

Avion

Home About Products Listings Shopping

Plant pots Ceramics Tables Chairs Crockery Tableware Cutlery

The Dandy chair

£250

Description

A timeless design, with premium materials features as one of our most popular and iconic pieces. The Dandy chair is perfect for any stylish living space with beech legs and lambskin leather upholstery.

Features

- Premium material
- Handmade upholstery
- Quality timeless classic

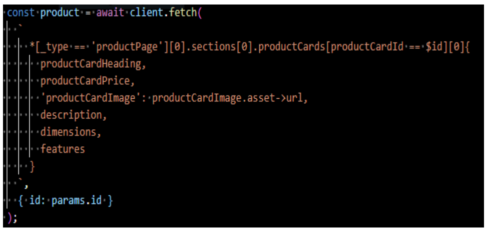
Dimensions

Height	Width	Depth
110cm	75cm	50cm

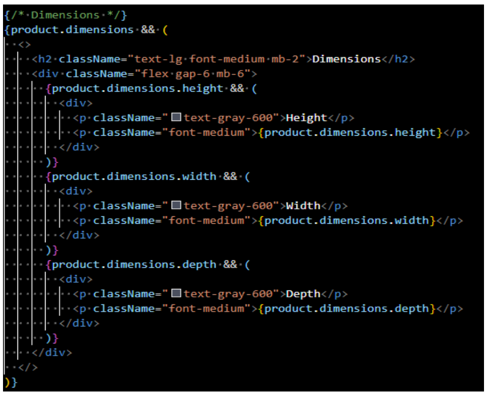
Amount: - 1 +

Add to Cart

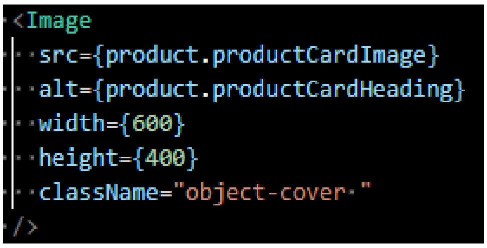
5.Code Snippets:



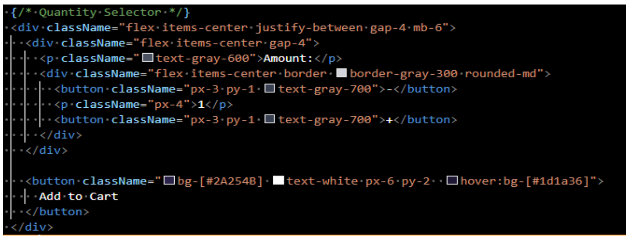
```
const product = await client.fetch(
  `
    *[_type == 'productPage'][0].sections[0].productCards[productCardId == $id][0]{
      productCardHeading,
      productCardPrice,
      'productCardImage': productCardImage.asset->url,
      description,
      dimensions,
      features
    }
  `,
  { id: params.id }
);
```



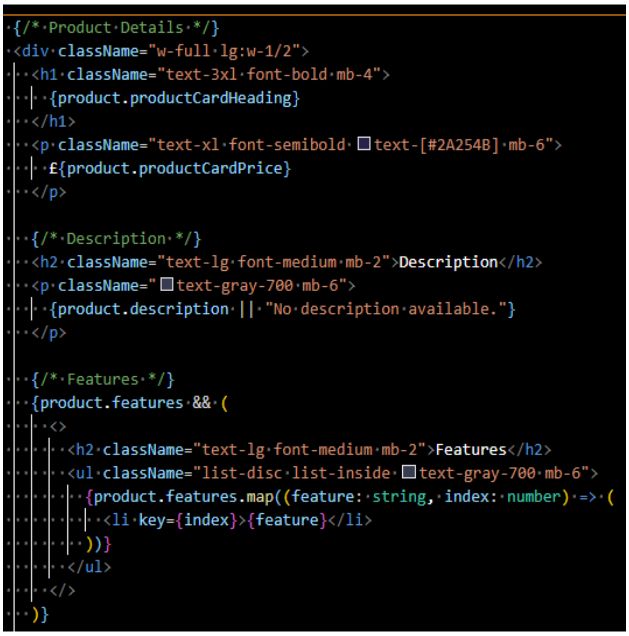
```
{/* Dimensions */}
{product.dimensions && (
  <>
    <h2 className="text-lg font-medium mb-2">Dimensions</h2>
    <div className="flex gap-6 mb-6">
      {product.dimensions.height && (
        <div>
          <p className="text-gray-600">Height</p>
          <p className="font-medium">{product.dimensions.height}</p>
        </div>
      )}
      {product.dimensions.width && (
        <div>
          <p className="text-gray-600">Width</p>
          <p className="font-medium">{product.dimensions.width}</p>
        </div>
      )}
      {product.dimensions.depth && (
        <div>
          <p className="text-gray-600">Depth</p>
          <p className="font-medium">{product.dimensions.depth}</p>
        </div>
      )}
    </div>
  </>
)}
```



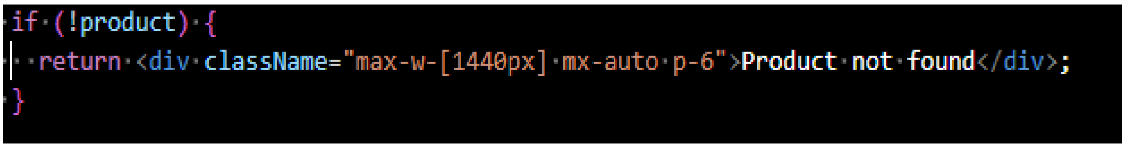
```
<Image
  src={product.productCardImage}
  alt={product.productCardHeading}
  width={600}
  height={400}
  className="object-cover "
/>
```



```
{/* Quantity Selector */}
<div className="flex items-center justify-between gap-4 mb-6">
  <div className="flex items-center gap-4">
    <p className="text-gray-600">Amount:</p>
    <div className="flex items-center border border-gray-300 rounded-md">
      <button className="px-3 py-1 text-gray-700">-</button>
      <p className="px-4">1</p>
      <button className="px-3 py-1 text-gray-700">+</button>
    </div>
  </div>

  <button className="bg-[#2A254B] text-white px-6 py-2  hover:bg-[#1d1a36]">
    Add to Cart
  </button>
</div>
```

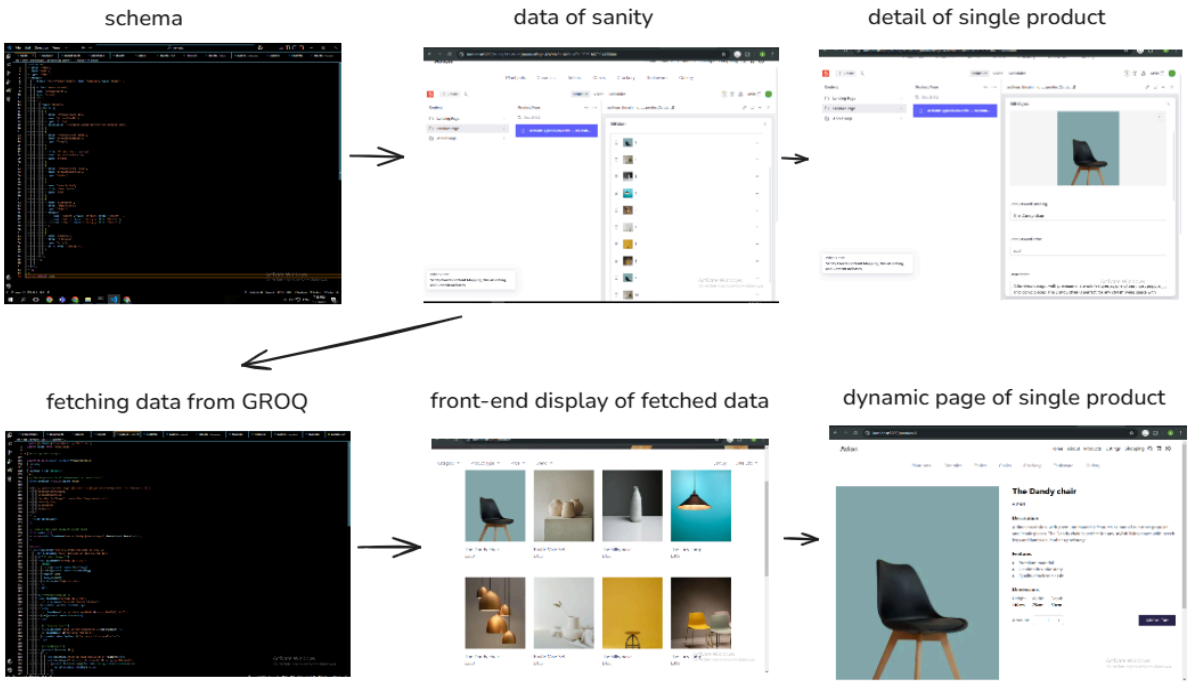


```
{/* Product Details */}
<div className="w-full lg:w-1/2">
  <h1 className="text-3xl font-bold mb-4">
    {product.productCardHeading}
  </h1>
  <p className="text-xl font-semibold text-[#2A254B] mb-6">
    £{product.productCardPrice}
  </p>

  {/* Description */}
  <h2 className="text-lg font-medium mb-2">Description</h2>
  <p className="text-gray-700 mb-6">
    {product.description || "No description available."}
  </p>

  {/* Features */}
  {product.features && (
    <>
      <h2 className="text-lg font-medium mb-2">Features</h2>
      <ul className="list-disc list-inside text-gray-700 mb-6">
        {product.features.map((feature: string, index: number) => (
          <li key={index}>{feature}</li>
        ))}
      </ul>
    </>
  )}
```



```
if (!product) {
  return <div className="max-w-[1440px] mx-auto p-6">Product not found</div>;
}
```

Process of fetching data


schema
data of sanity
detail of single product
fetching data from GROQ
front-end display of fetched data
dynamic page of single product
The Dandy chair