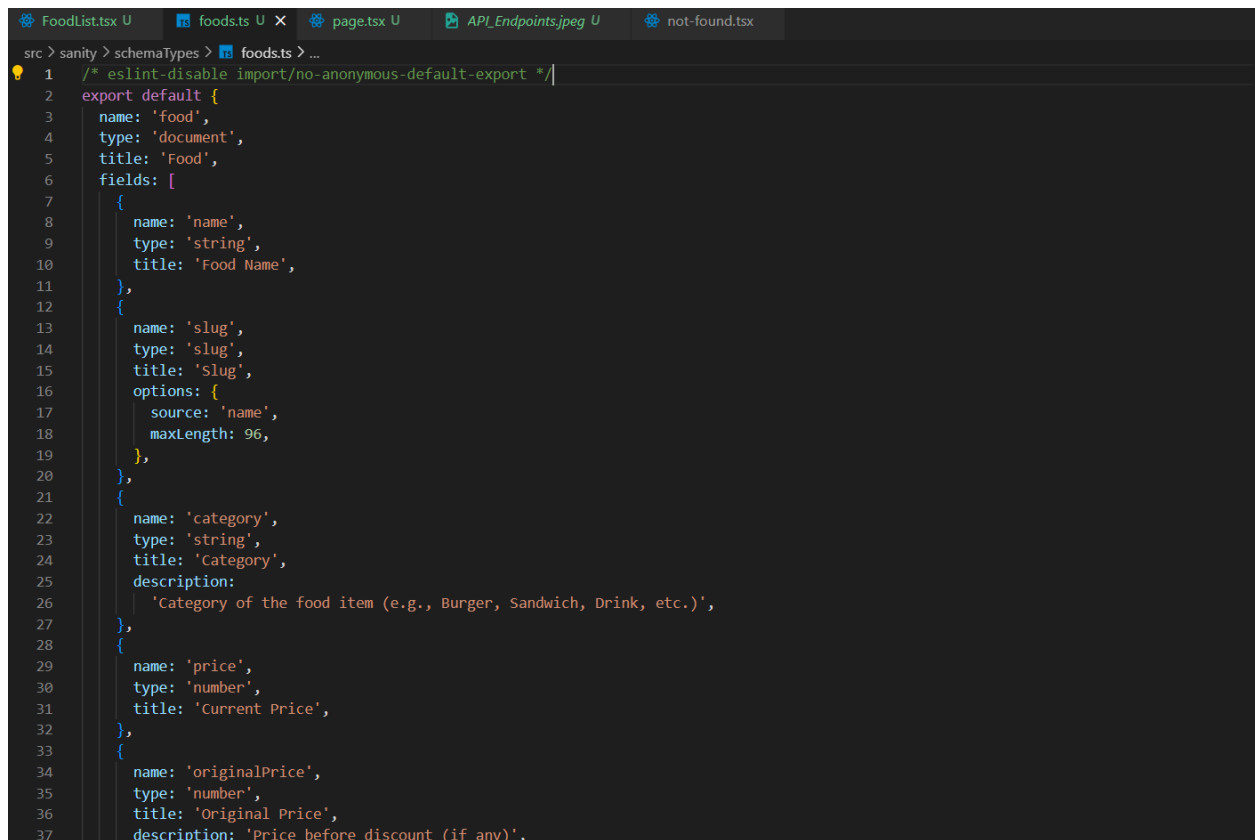


## Day 3 - API Integration Report

This report details the process of integrating an API with Sanity CMS and leveraging the data in a Next.js application. The integration steps cover setting up the Sanity schema, querying data with GROQ, and dynamically rendering the data in Next.js components. Visual aids are included for enhanced clarity.

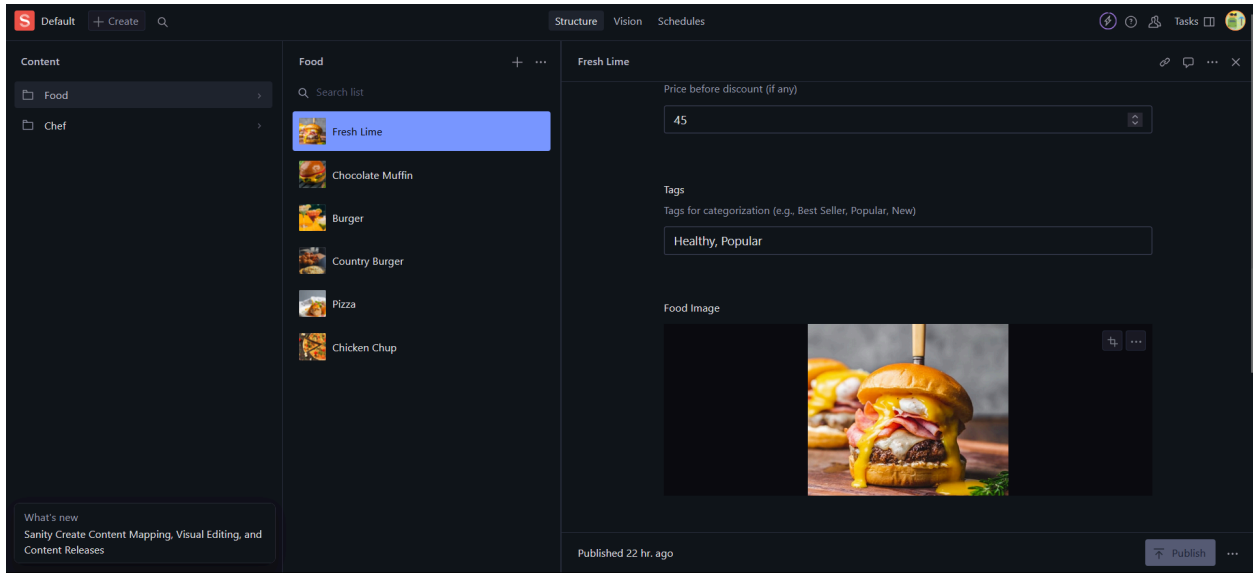
### API Integration with Sanity CMS

#### Step 1: Schemas to food products:

A screenshot of a code editor with a dark theme. The editor shows a file named 'foods.ts' with a Sanity schema for food products. The schema is written in TypeScript and includes fields for name, slug, category, price, and originalPrice. The editor has a tab bar at the top with several files: 'FoodList.tsx', 'foods.ts', 'page.tsx', 'API\_Endpoints.jpeg', and 'not-found.tsx'. The 'foods.ts' tab is active. The code is as follows:

```
src > sanity > schemaTypes > foods.ts > ...  
1  /* eslint-disable import/no-anonymous-default-export */  
2  export default {  
3    name: 'food',  
4    type: 'document',  
5    title: 'Food',  
6    fields: [  
7      {  
8        name: 'name',  
9        type: 'string',  
10       title: 'Food Name',  
11      },  
12      {  
13        name: 'slug',  
14        type: 'slug',  
15        title: 'Slug',  
16        options: {  
17          source: 'name',  
18          maxLength: 96,  
19        },  
20      },  
21      {  
22        name: 'category',  
23        type: 'string',  
24        title: 'Category',  
25        description:  
26        | 'Category of the food item (e.g., Burger, Sandwich, Drink, etc.)',  
27      },  
28      {  
29        name: 'price',  
30        type: 'number',  
31        title: 'Current Price',  
32      },  
33      {  
34        name: 'originalPrice',  
35        type: 'number',  
36        title: 'Original Price',  
37        description: 'Price before discount (if any)',
```

## Step 2. Populating Data in Sanity Studio:




## Step 3. Fetching Data from Sanity using Sanity's GROQ:

```
async function getFoods(): Promise<Food[]> {
  return client.fetch(`*[_type == "food"] {
    _id,
    name,
    slug,
    category,
    price,
    originalPrice,
    tags,
    image,
    description,
    available
  }`);
}


export default async function FoodList() {
  const foods = await getFoods();
}
```

## Step 4. Using the Sanity CMS API in Next.js


Click Below To View Shop Details!



**Burger**  
Sandwich  
Classic country-style burger served with fries.  
**\$21.00** ~~\$45.00~~  
Recommended Available




**Chocolate Muffin**  
Dessert  
Soft and rich chocolate muffin topped with chocolate chips  
**\$28.00** ~~\$30.00~~  
Sell, Sweet Available



**Fresh Lime**  
Drink  
Refreshing fresh lime drink made with natural ingredients.  
**\$38.00** ~~\$45.00~~  
Healthy, Popular Available

## Dynamic Routing:



**Chocolate Muffin**  
Dessert  
Soft and rich chocolate muffin topped with chocolate chips  
**\$28.00** ~~\$30.00~~  
Sell, Sweet Available