

Day 3

API INTEGRATION REPORT Q-COMMERCE

FOODTUCK

NAME: LALARUKH

ROLL NO: 00448579

PROCESS OF API INTEGRATION

API Integration Process:

The API integration for the Q-Commerce web application was completed in several steps:

1.Understanding API Documentation:

I thoroughly reviewed the API documentation of the third-party services used for food and chef data.

Focused on identifying the key endpoints required for integration, such as food items, chef details, and real-time updates.

2. Setting Up Environment Variables:

I set up environment variables in .env.local to securely store the necessary API keys and tokens for accessing the API:

```
$ .env.local x
$ .env.local
1 NEXT_PUBLIC_SANITY_PROJECT_ID="03x8jbjz"
2 NEXT_PUBLIC_SANITY_DATASET="production"
3 NEXT_PUBLIC_SANITY_API_TOKEN="skCZ6viaJmIX8079B0Zp6Z9fBYdV043XJnH6nEr1b2fFdQ4FHRdD0hXptrviwPPLG1FeedbFiYq1a
4
```

3. Endpoint Integration:

I integrated various endpoints to fetch food item data and chef details.

Added error handling and response validation to ensure the integration is reliable and consistent.

1: Adjustments Made to Schemas

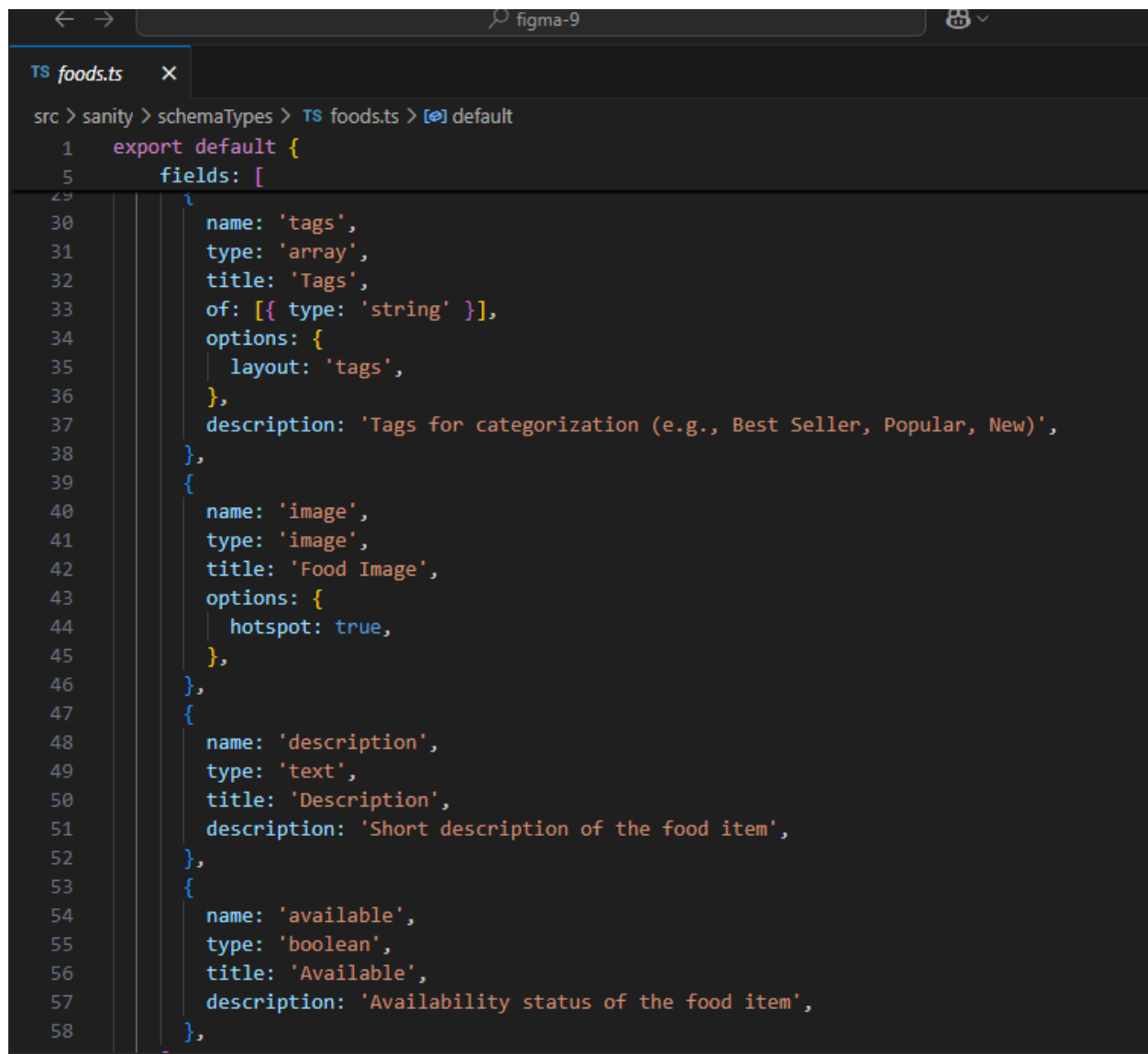
I updated the food schema to accommodate additional fields such as price, ingredients, and availability, ensuring it aligns with the API responses:

FOOD SCHEMA

TS foods.ts X

src > sanity > schemaTypes > TS foods.ts > [🔍] default

```
1  export default {
2    name: 'food',
3    type: 'document',
4    title: 'Food',
5    fields: [
6      {
7        name: 'name',
8        type: 'string',
9        title: 'Food Name',
10     },
11     {
12       name: 'category',
13       type: 'string',
14       title: 'Category',
15       description:
16         'Category of the food item (e.g., Burger, Sandwich, Drink, etc.)',
17     },
18     {
19       name: 'price',
20       type: 'number',
21       title: 'Current Price',
22     },
23     {
24       name: 'originalPrice',
25       type: 'number',
26       title: 'Original Price',
27       description: 'Price before discount (if any)',
28     },
29     {
30       name: 'tags',
31       type: 'array',
32       title: 'Tags',
```

A screenshot of a code editor window titled 'figma-9'. The editor shows a file named 'TS foods.ts' with a breadcrumb path 'src > sanity > schemaTypes > TS foods.ts > default'. The code is a TypeScript schema for food items, starting with 'export default {'. It defines a 'fields' array containing three objects. The first object is for 'tags', with type 'array', title 'Tags', and a description 'Tags for categorization (e.g., Best Seller, Popular, New)'. The second object is for 'image', with type 'image', title 'Food Image', and a 'hotspot' option set to true. The third object is for 'description', with type 'text', title 'Description', and a description 'Short description of the food item'. The fourth object is for 'available', with type 'boolean', title 'Available', and a description 'Availability status of the food item'. The code is syntax-highlighted and includes line numbers from 1 to 58.

```
src > sanity > schemaTypes > TS foods.ts > default
1  export default {
5    fields: [
29      {
30        name: 'tags',
31        type: 'array',
32        title: 'Tags',
33        of: [{ type: 'string' }],
34        options: {
35          layout: 'tags',
36        },
37        description: 'Tags for categorization (e.g., Best Seller, Popular, New)',
38      },
39      {
40        name: 'image',
41        type: 'image',
42        title: 'Food Image',
43        options: {
44          hotspot: true,
45        },
46      },
47      {
48        name: 'description',
49        type: 'text',
50        title: 'Description',
51        description: 'Short description of the food item',
52      },
53      {
54        name: 'available',
55        type: 'boolean',
56        title: 'Available',
57        description: 'Availability status of the food item',
58      },
59    ],
60  },
61 }
```

I also modified the chef schema, adding fields for expertise and experience to enrich the chef data:

TS chefs.ts

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

```
1 export default {
2   name: 'chef',
3   type: 'document',
4   title: 'Chef',
5   fields: [
6     {
7       name: 'name',
8       type: 'string',
9       title: 'Chef Name',
10    },
11    {
12      name: 'position',
13      type: 'string',
14      title: 'Position',
15      description: 'Role or title of the chef (e.g., Head Chef, Sous Chef)',
16    },
17    {
18      name: 'experience',
19      type: 'number',
20      title: 'Years of Experience',
21      description: 'Number of years the chef has worked in the culinary field',
22    },
23    {
24      name: 'specialty',
25      type: 'string',
26      title: 'Specialty',
27      description: 'Specialization of the chef (e.g., Italian Cuisine, Pastry)',
28    },
29    {
30      name: 'image',
31      type: 'image',
32      title: 'Chef Image',
```

```
chefs.ts  X
c > sanity > schemaTypes > TS chefs.ts > [0] default
1  export default {
5    fields: [
28      },
29      {
30        name: 'image',
31        type: 'image',
32        title: 'Chef Image',
33        options: {
34          hotspot: true,
35        },
36      },
37      {
38        name: 'description',
39        type: 'text',
40        title: 'Description',
41        description: 'Short bio or introduction about the chef',
42      },
43      {
44        name: 'available',
45        type: 'boolean',
46        title: 'Currently Active',
47        description: 'Availability status of the chef',
48      },
49    ],
50  };

```

Migration Steps and Tools Used

1. Tools Utilized:

I used `@sanity/client` to migrate the data from the API into the Sanity CMS.

2. Migration Steps:

I fetched food and chef data from the external API.

Transformed the API data to fit the updated Sanity schemas.

Uploaded the transformed data to Sanity using the following migration script.

```
... TS client.ts X
src > sanity > lib > TS client.ts > ...
1 import { createClient } from 'next-sanity'
2
3 import { apiVersion, dataset, projectId } from '../env'
4
5 export const client = createClient({
6   projectId,
7   dataset,
8   apiVersion,
9   useCdn: true, // Set to false if statically generating pages, using ISR or tag-based revalidation
10 })
11
```

```
... TS queries.ts X
src > sanity > lib > TS queries.ts > [⌕] allfoods
1 import { defineQuery } from "next-sanity";
2
3 export const allchefs = defineQuery(`
4   *[_type == "chef"] {
5     _id,
6     name,
7     position,
8     experience,
9     specialty,
10    "imageUrl": image.asset->url
11  } `)
12
13 export const allfoods = defineQuery(`
14   *[_type == "food"] {
15     _id,
16     name,
17     category,
18     price,
19     original price,
20     tags,
21     "imageUrl": image.asset->url
22   } `)
23
24
```

```
src > sanity > lib > TS fetch.ts > sanityFetch
1 import { createClient } from "next-sanity";
2
3 const client = createClient({
4   projectId : "03x8bjbz",
5   dataset : "production",
6   useCdn : true,
7   apiVersion : "2023-10-10"
8 })
9 export async function sanityFetch({query, params = {}}: {query : string , params?: any}){
10   return await client.fetch(query,params)
11 }
```

1. API Calls:

The screenshot shows a Windows 11 desktop with a VS Code editor. The project is named 'figma-9'. The Explorer sidebar on the left lists the following files and folders:

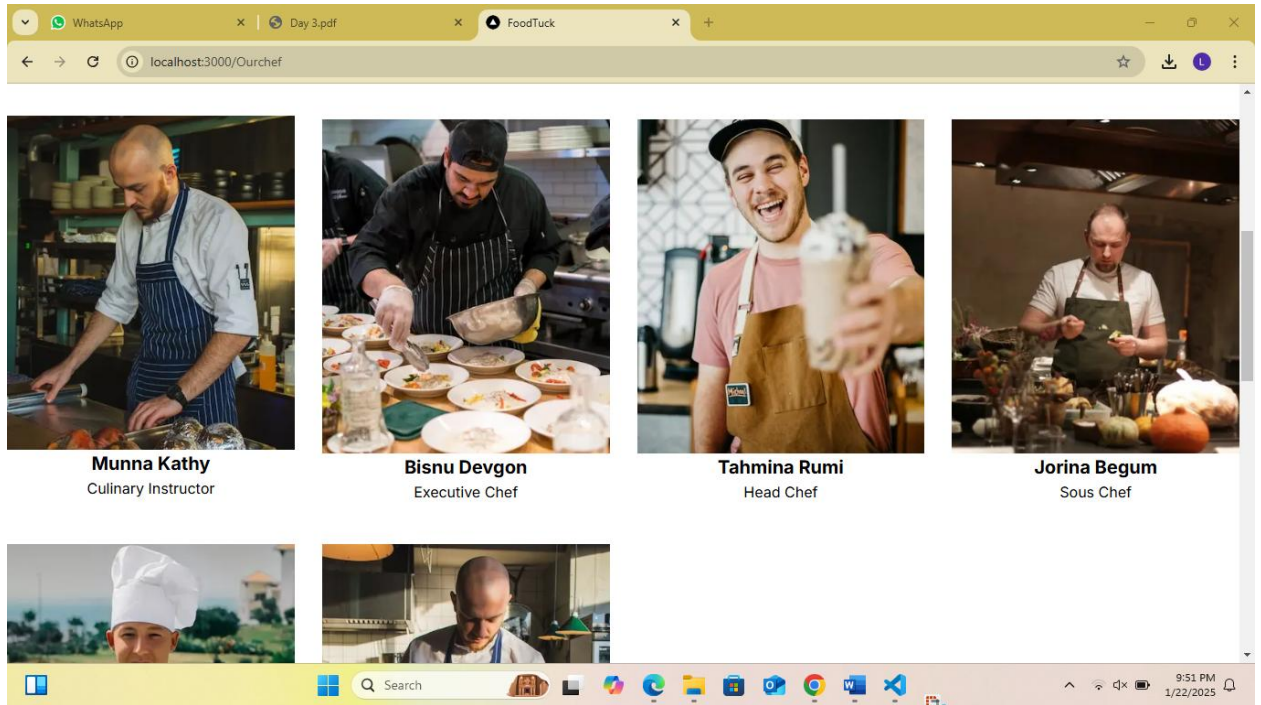
- src
- app
- global.css
- layout.tsx
- page.tsx
- Components
- About.tsx
- Common.tsx
- Food.tsx
- Footer.tsx
- Hero.tsx
- Hero3.tsx
- menu.tsx
- Navbar.tsx
- Service.tsx
- sanity
- .env.local
- .eslintrc.json
- .gitignore
- next-env.d.ts
- next.config.js
- package-lock.json
- package.json
- postcss.config.js
- README.md
- sanity.d.ts
- OUTLINE
- TIMELINE

The main editor shows the 'package.json' file with the following content:

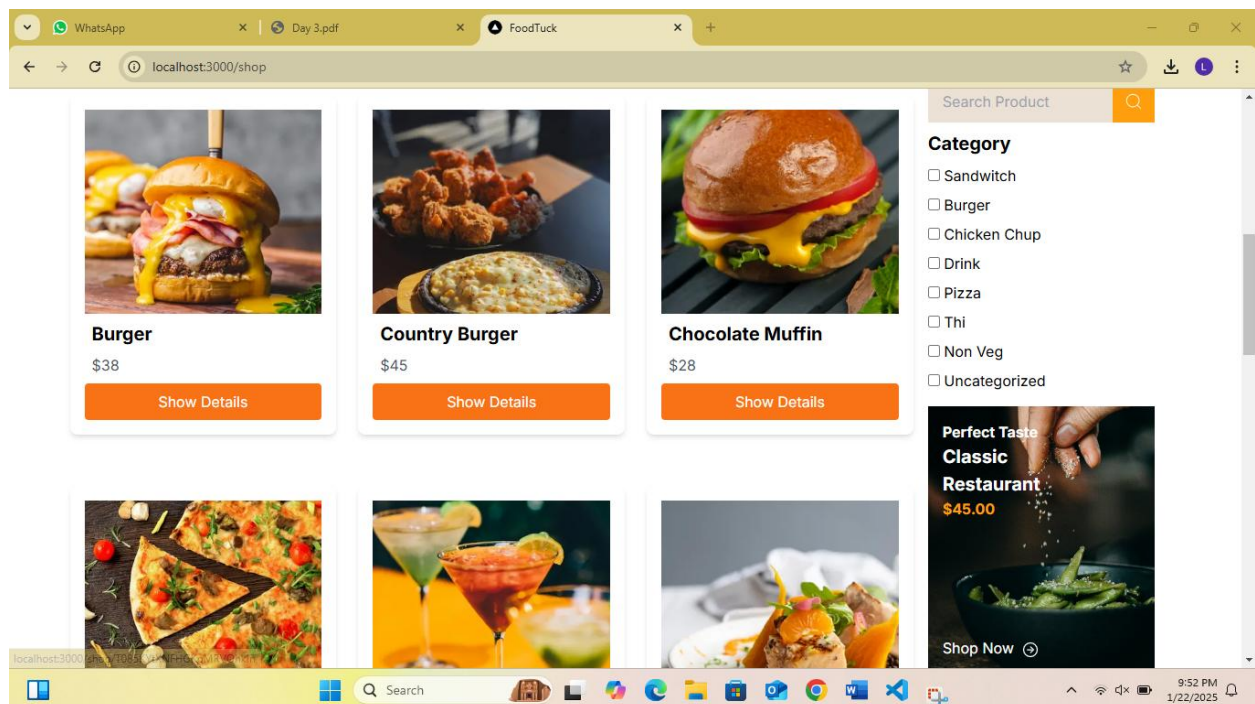
```
{
  "name": "figma-9",
  "version": "0.1.0",
  "private": true,
  "dependencies": {
    "next": "12.1.0",
    "react": "18.0.0",
    "react-dom": "18.0.0",
    "sanity": "2.15.0",
    "sanity-dts": "2.15.0",
    "sanity-plugin-image": "1.0.0",
    "sanity-plugin-media": "1.0.0",
    "sanity-plugin-pdf-viewer": "1.0.0",
    "sanity-plugin-quickstart": "1.0.0",
    "sanity-plugin-view-image": "1.0.0",
    "sanity-plugin-view-pdf": "1.0.0",
    "sanity-plugin-view-video": "1.0.0",
    "sanity-plugin-view-audio": "1.0.0",
    "sanity-plugin-view-geo-point": "1.0.0",
    "sanity-plugin-view-geo-polygon": "1.0.0",
    "sanity-plugin-view-geo-line": "1.0.0",
    "sanity-plugin-view-geo-circle": "1.0.0",
    "sanity-plugin-view-geo-rectangle": "1.0.0",
    "sanity-plugin-view-geo-circle-with-center": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-center": "1.0.0",
    "sanity-plugin-view-geo-circle-with-radius": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-radius": "1.0.0",
    "sanity-plugin-view-geo-circle-with-diameter": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-diameter": "1.0.0",
    "sanity-plugin-view-geo-circle-with-circumference": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-circumference": "1.0.0",
    "sanity-plugin-view-geo-circle-with-area": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-area": "1.0.0",
    "sanity-plugin-view-geo-circle-with-perimeter": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-perimeter": "1.0.0",
    "sanity-plugin-view-geo-circle-with-volume": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-volume": "1.0.0",
    "sanity-plugin-view-geo-circle-with-surface-area": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-surface-area": "1.0.0",
    "sanity-plugin-view-geo-circle-with-mass": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-mass": "1.0.0",
    "sanity-plugin-view-geo-circle-with-moment": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-moment": "1.0.0",
    "sanity-plugin-view-geo-circle-with-inertia": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-inertia": "1.0.0",
    "sanity-plugin-view-geo-circle-with-stress": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-stress": "1.0.0",
    "sanity-plugin-view-geo-circle-with-strain": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-strain": "1.0.0",
    "sanity-plugin-view-geo-circle-with-temperature": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-temperature": "1.0.0",
    "sanity-plugin-view-geo-circle-with-pressure": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-pressure": "1.0.0",
    "sanity-plugin-view-geo-circle-with-current": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-current": "1.0.0",
    "sanity-plugin-view-geo-circle-with-voltage": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-voltage": "1.0.0",
    "sanity-plugin-view-geo-circle-with-power": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-power": "1.0.0",
    "sanity-plugin-view-geo-circle-with-energy": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-energy": "1.0.0",
    "sanity-plugin-view-geo-circle-with-charge": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-charge": "1.0.0",
    "sanity-plugin-view-geo-circle-with-magnetic-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-magnetic-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-electric-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-electric-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-gravitational-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-gravitational-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-weak-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-weak-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-strong-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-strong-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-higgs-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-higgs-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-gluon-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-gluon-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-photon-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-photon-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-neutrino-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-neutrino-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-electron-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-electron-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-proton-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-proton-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-neutron-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-neutron-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-quark-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-quark-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-lepton-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-lepton-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-boson-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-boson-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-fermion-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-fermion-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-scalar-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-scalar-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-vector-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-vector-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-spinor-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-spinor-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-tensor-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-tensor-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-form-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-form-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-gauge-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-gauge-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-gravity-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-gravity-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-electromagnetism-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-electromagnetism-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-strong-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-strong-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-weak-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-weak-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-higgs-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-higgs-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-gluon-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-gluon-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-photon-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-photon-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-neutrino-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-neutrino-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-electron-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-electron-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-proton-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-proton-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-neutron-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-neutron-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-quark-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-quark-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-lepton-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-lepton-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-boson-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-boson-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-fermion-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-fermion-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-scalar-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-scalar-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-vector-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-vector-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-spinor-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-spinor-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-tensor-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-tensor-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-form-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-form-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-gauge-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-gauge-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-gravity-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-gravity-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-electromagnetism-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-electromagnetism-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-strong-interaction-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-strong-interaction-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-weak-interaction-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-weak-interaction-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-higgs-interaction-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-higgs-interaction-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-gluon-interaction-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-gluon-interaction-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-circle-with-photon-interaction-interaction-field": "1.0.0",
    "sanity-plugin-view-geo-rectangle-with-photon-interaction-interaction-field": "1.0.
```

2. Data Displayed on Frontend:

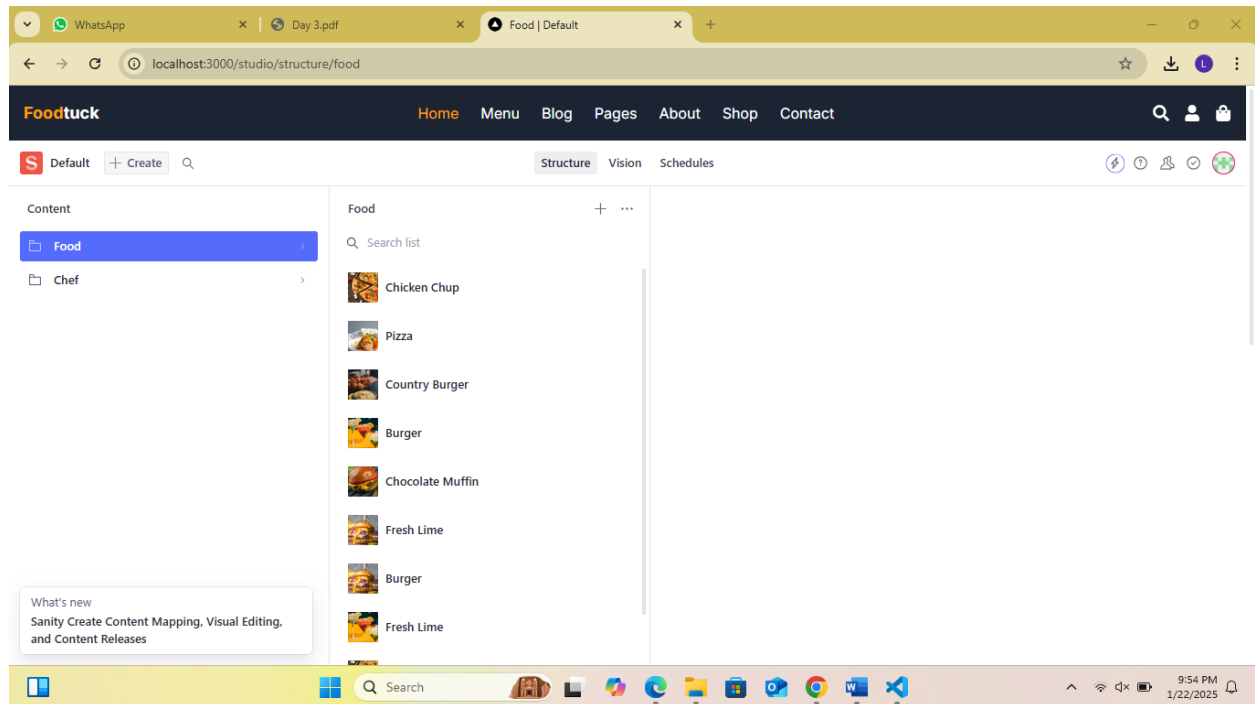
CHEF



FOOD



3. Populated Sanity CMS Fields:



This report outlines the steps I took to integrate APIs into the Q-commerce website, update schemas, and migrate the data into Sanity CMS. Let me know if you need any additional details!