Day 3 - API Integration Report – AS foods

API Integration Process

Overview:

The integration involves connecting an external API that provides data for foods and chefs to a **Sanity CMS** project, allowing seamless data migration and management.

---

2. Steps Taken:

- **Environment Setup:**
  - Used **dotenv** to securely load environment variables from `.env.local`.
  - Key variables included:
    - `NEXT_PUBLIC_SANITY_PROJECT_ID`
    - `NEXT_PUBLIC_SANITY_DATASET`
    - `SANITY_TOKEN`
- **Sanity Client Creation:**
  - Utilized **@sanity/client** to establish a connection to the Sanity project.
  - Configured the client with the following credentials:
    - **Project ID**
    - **Dataset**
    - **API Version**
    - **Authentication Token**
- **Data Fetching:**
  - Made concurrent API calls using **axios** to fetch food and chef data from the endpoints:
    - **Foods API**: `https://sanity-nextjs-rouge.vercel.app/api/foods`
    - **Chefs API**: `https://sanity-nextjs-rouge.vercel.app/api/chefs`
- **Data Processing:**
  - Iterated through the fetched data to process it.
  - Uploaded images to **Sanity's asset library** using the `client.assets.upload()` method.
- **Sanity Document Creation:**
  - Transformed the fetched API data into Sanity-compatible document structures.
  - Uploaded each document to Sanity using `client.create()`.
- **Error Handling:**
  - Implemented **try-catch** blocks for handling errors during API calls and Sanity operations.
  - Logged errors for efficient debugging and troubleshooting.

Adjustments Made to Schemas:

- **API Endpoints Used**:
  - **Foods**: https://sanity-nextjs-rouge.vercel.app/api/foods
  - **Chefs**: https://sanity-nextjs-rouge.vercel.app/api/chefs

By Taha Saif

Day 3 - API Integration Report – AS foods

1. Chef Schema Enhancements:

- **By Ghaniya Khan**
- **Image Field**: Enabled the hotspot option, allowing for flexible cropping and adaptable displays across different screen sizes.

Key Fields Added:

- **Category**: A string field to categorize items (e.g., Burgers, Drinks).
- **Price**: A number field to specify the current price of each item.
- **Original Price**: An optional field for showing discounted prices.
- **Tags**: An array field for categorizing and tagging the food items.
- **Available**: A boolean field indicating stock availability.

These changes help provide better control over product data, ensuring a seamless and adaptable content management experience within **Sanity CMS**.

Migration Steps and Tools Used

1. Migration Steps:

- **Preparation**:
  - Analyzed the API data structure and aligned it with the existing Sanity CMS schemas.
  - Created new Sanity schemas for food and chef, incorporating all necessary fields to support the migration process.
- **Data Import Script**:
  - Developed the `import-data.mjs` script to automate the process of data fetching, processing, and uploading to the CMS.
  - Used **axios** for making API calls to fetch data and **@sanity/client** for interacting with the Sanity CMS to upload and manage content.
- **Image Handling**:
  - Downloaded images from the API using **axios** with `responseType: 'arraybuffer'` to fetch raw image data.
  - Uploaded the images to **Sanity CMS** and stored the image references in the appropriate document fields.
- **Document Creation**:
  - Mapped the fields from the API data to the corresponding Sanity schema fields, ensuring proper alignment.
  - Implemented fallback values for optional or missing fields, ensuring data consistency.

2. Tools Used:

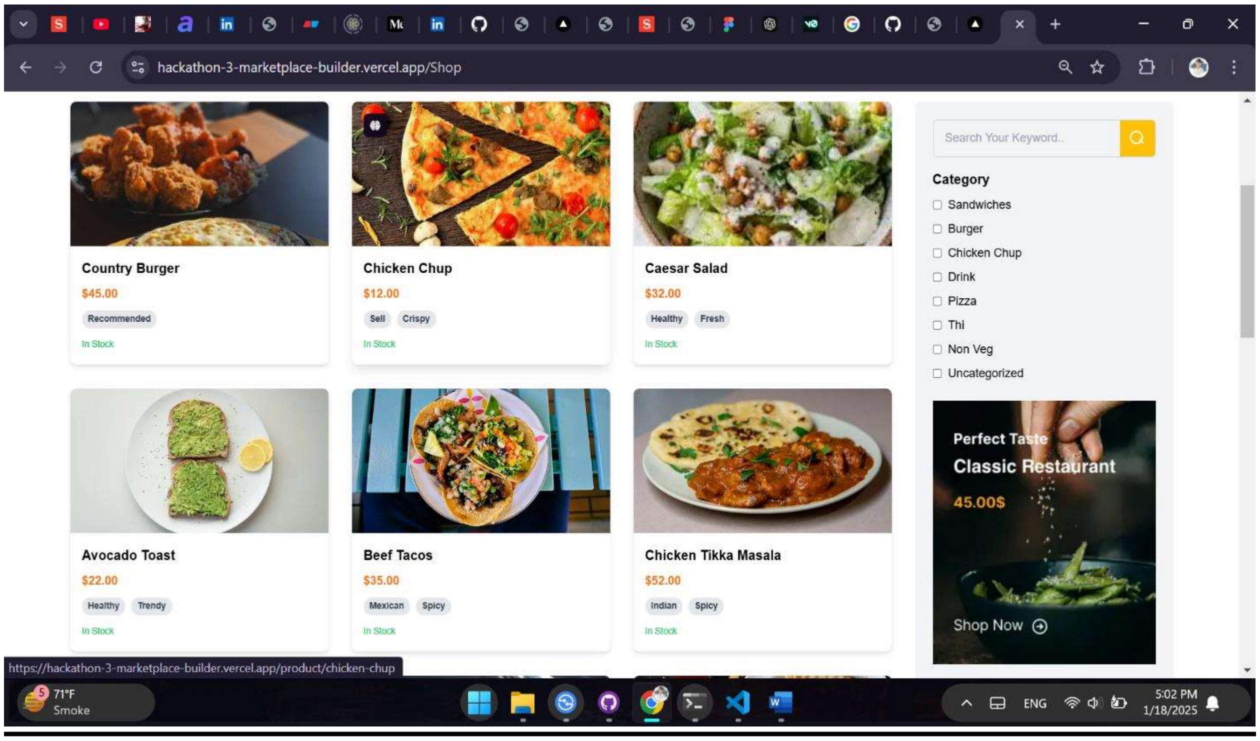
- **Node.js Modules**:
  - **axios**: Used for making HTTP requests to interact with the API.

By Taha Saif

  - **dotenv**: Managed environment variables securely for the integration.
  - **@sanity/client**: Interfaced with the Sanity CMS for managing documents and assets.
- **Sanity Features**:
  - **Asset Management**: Utilized Sanity's asset management system to handle image uploads efficiently.
  - **API Versioning**: Ensured consistent schema support across different environments by using API versioning.
- **Utilities**:
  - **fileURLToPath** and **path**: Used these utilities for resolving file paths during the image import process, ensuring compatibility with the file system.

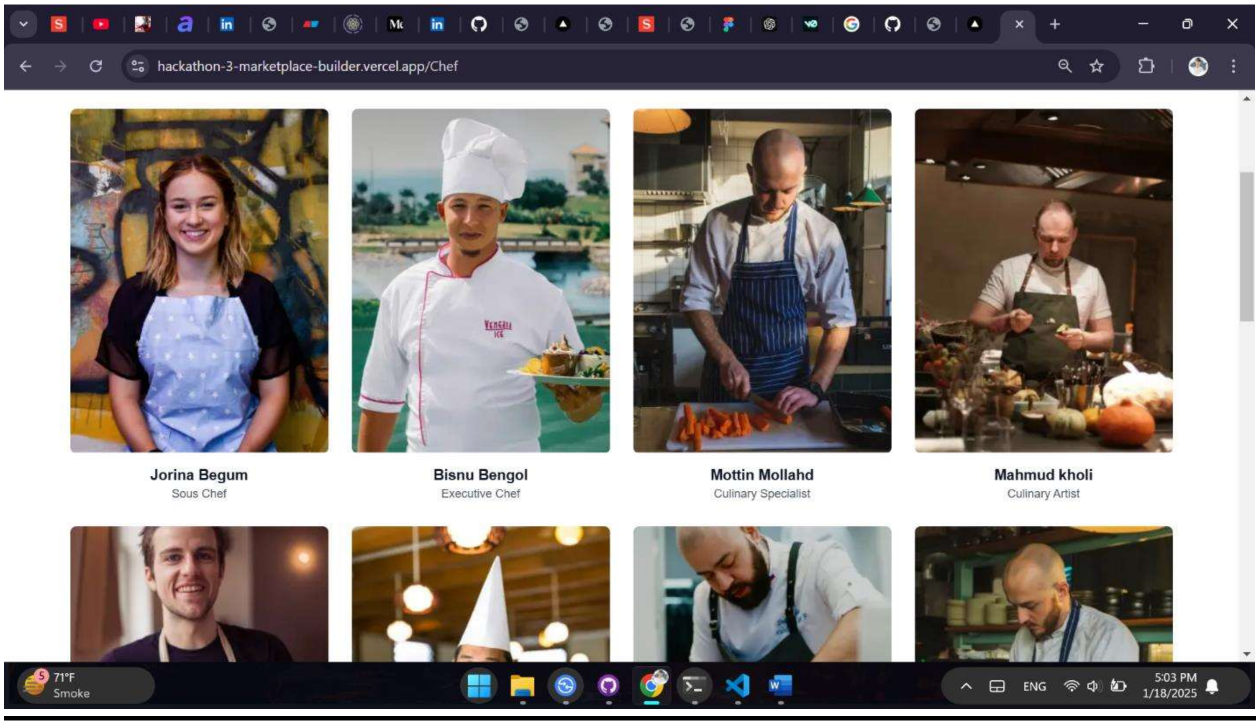
Data successfully displayed in the frontend:

Foods Data:



Chefs Data:

Day 3 - API Integration Report – AS foods



Data Population in Sanity CMS:

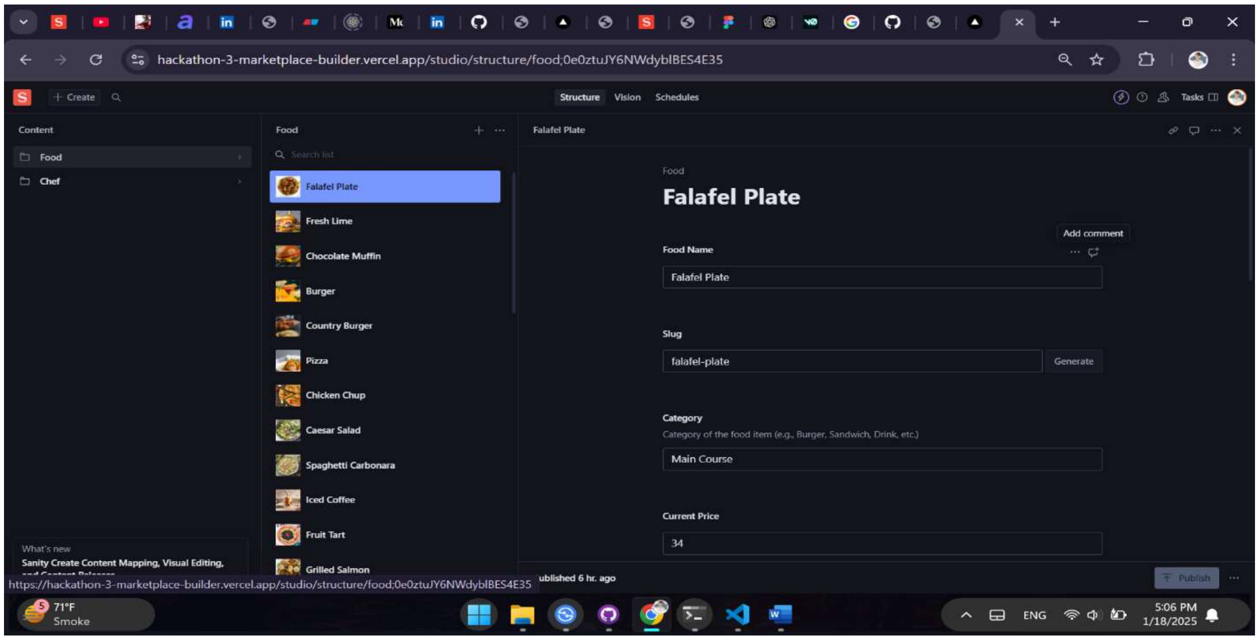
- Chef Schema:
    - Position: Added a string field to specify the chef's role.
    - Experience: Added a number field to capture the number of years of experience.
    - Image: Enabled the hotspot option to allow flexible cropping of images.
    - Availability: Added a boolean field to track whether the chef is available.
- Food Schema:
    - Category: Created a string field to categorize foods (e.g., Burgers, Drinks).
    - Price: Added a number field to store the current price of items.
    - Original Price: Included an optional number field for discounted prices.
    - Tags: Enabled an array field for food item tags (e.g., spicy, vegetarian).
    - Available: Added a boolean field to indicate food availability.

Foods:

Foods:

By Taha Saif

Day 3 - API Integration Report – AS foods



hackathon-3-marketplace-builder.vercel.app/studio/structure/food;0e0ztuJY6NWdybIBES4E35

Content
- Food
- Chef

Food
- Falafel Plate
- Fresh Lime
- Chocolate Muffin
- Burger
- Country Burger
- Pizza
- Chicken Chup
- Caesar Salad
- Spaghetti Carbonara
- Iced Coffee
- Fruit Tart
- Grilled Salmon

Food
Falafel Plate

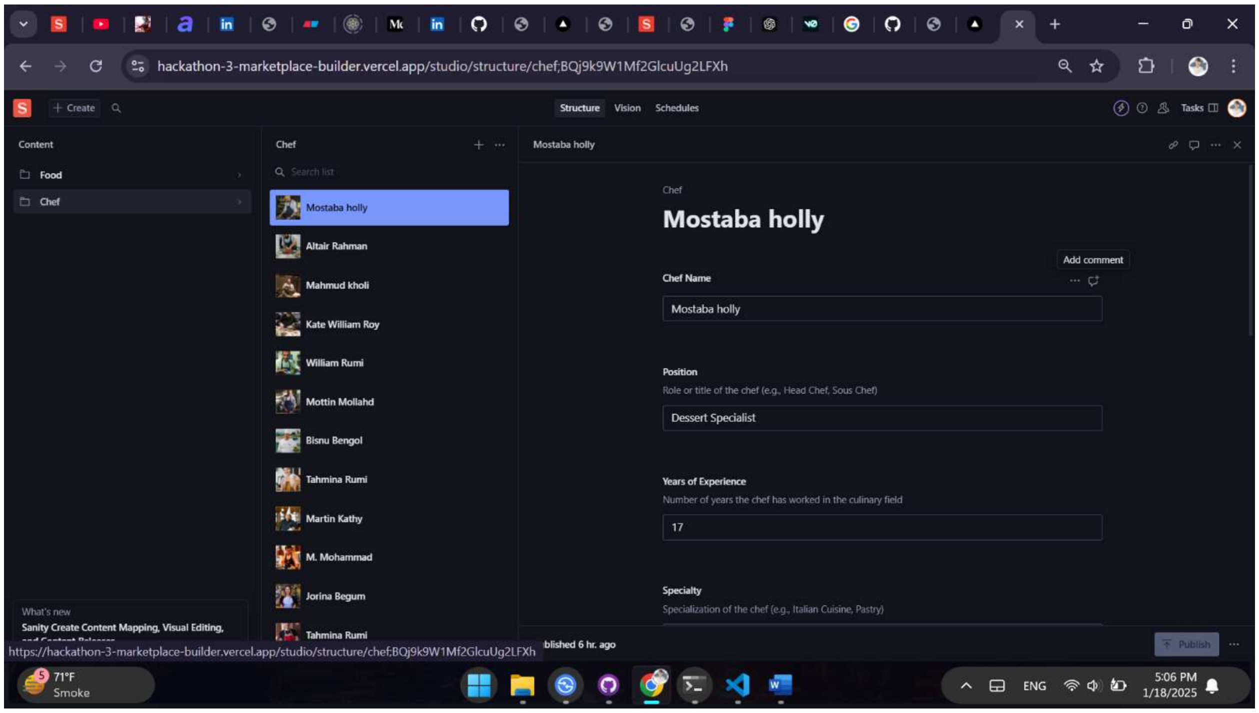
Food Name: Falafel Plate

Slug: falafel-plate

Category
Category of the food item (e.g., Burger, Sandwich, Drink, etc.)
Main Course

Current Price: 34

Chefs:



hackathon-3-marketplace-builder.vercel.app/studio/structure/chef;BQj9k9W1Mf2GlcuUg2LFXh

Content
- Food
- Chef

Chef
- Mostaba holly
- Altair Rahman
- Mahmud kholi
- Kate William Roy
- William Rumi
- Mottin Mollahd
- Bisnu Bengol
- Tahmina Rumi
- Martin Kathy
- M. Mohammad
- Jorina Begum
- Tahmina Rumi

Chef
Mostaba holly

Chef Name: Mostaba holly

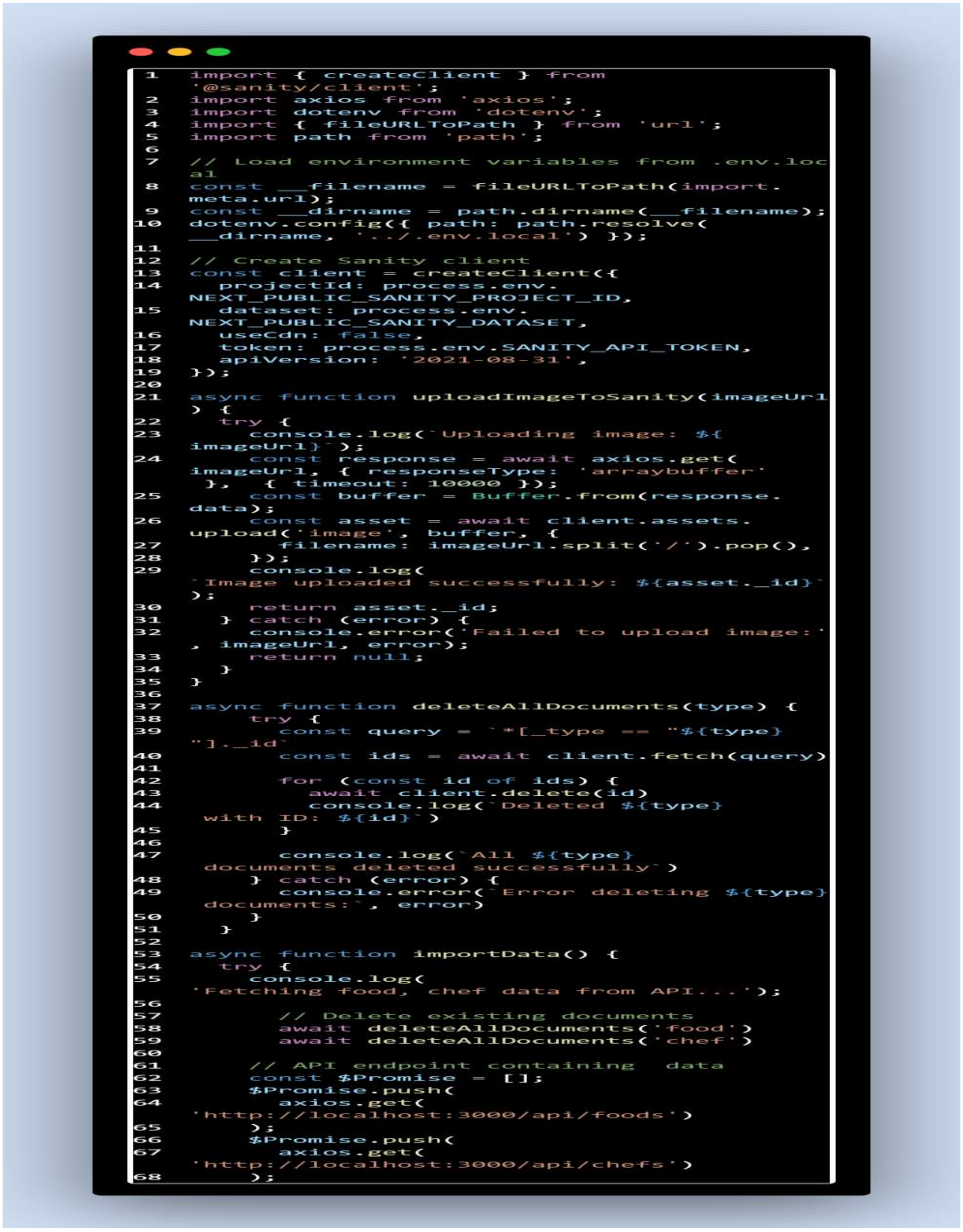
Position
Role or title of the chef (e.g., Head Chef, Sous Chef)
Dessert Specialist

Years of Experience
Number of years the chef has worked in the culinary field
17

Specialty
Specialization of the chef (e.g., Italian Cuisine, Pastry)

Code Snippets for API Integeration and Migration Scripts

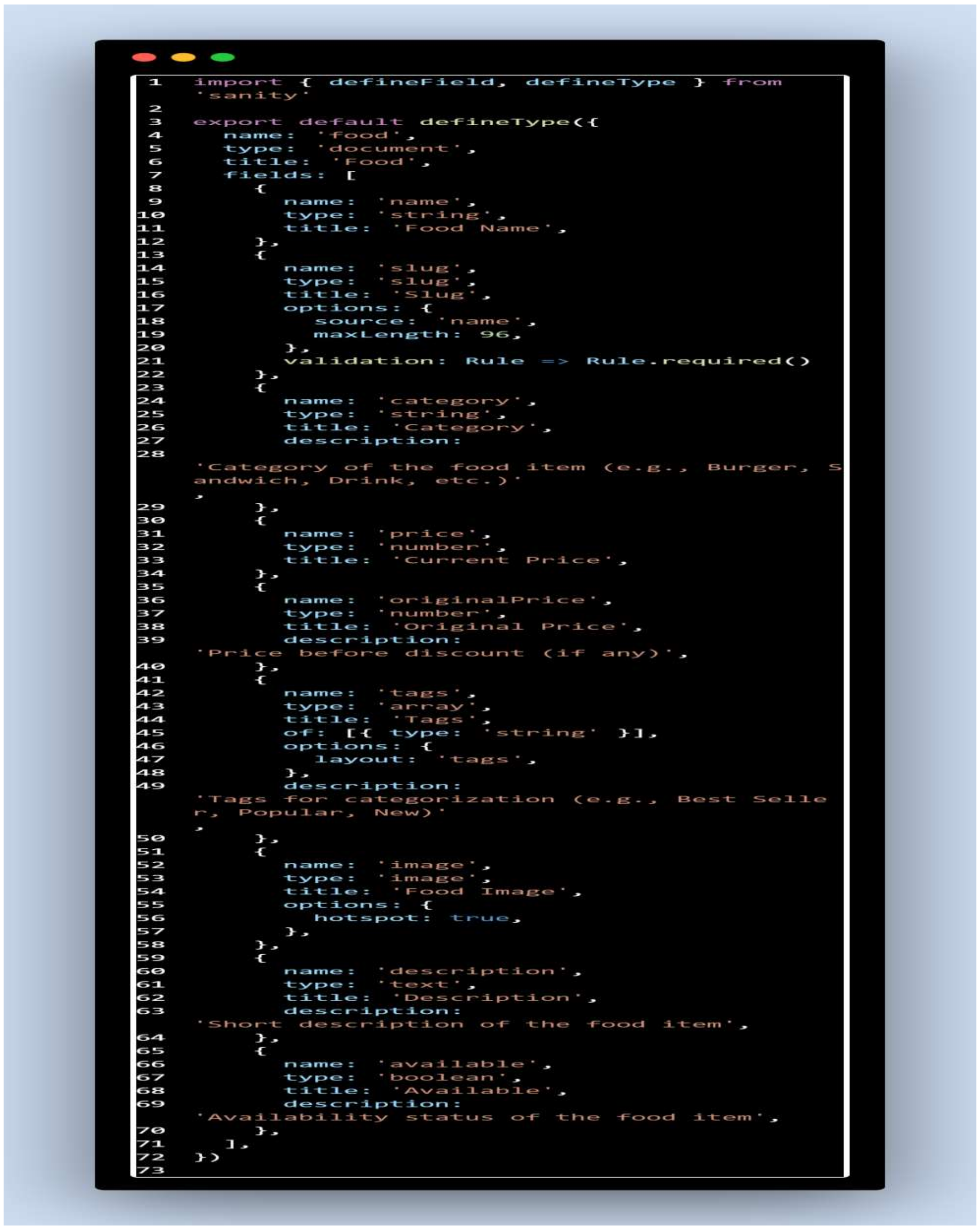
By Taha Saif



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

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

// Create Sanity client
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' }, { timeout: 10000 });
    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 deleteAllDocuments(type) {
    try {
      const query = `*[_type == "${type}"]._id`
      const ids = await client.fetch(query)

      for (const id of ids) {
        await client.delete(id)
        console.log(`Deleted ${type} with ID: ${id}`)
      }

      console.log(`All ${type} documents deleted successfully`)
    } catch (error) {
      console.error(`Error deleting ${type} documents:`, error)
    }
  }

async function importData() {
  try {
    console.log('Fetching food, chef data from API...');

      // Delete existing documents
      await deleteAllDocuments('food')
      await deleteAllDocuments('chef')

    // API endpoint containing  data
    const $Promise = [];
    $Promise.push(
      axios.get('http://localhost:3000/api/foods')
    );
    $Promise.push(
      axios.get('http://localhost:3000/api/chefs')
    );
```

Foods:

Day 3 - API Integration Report – AS foods

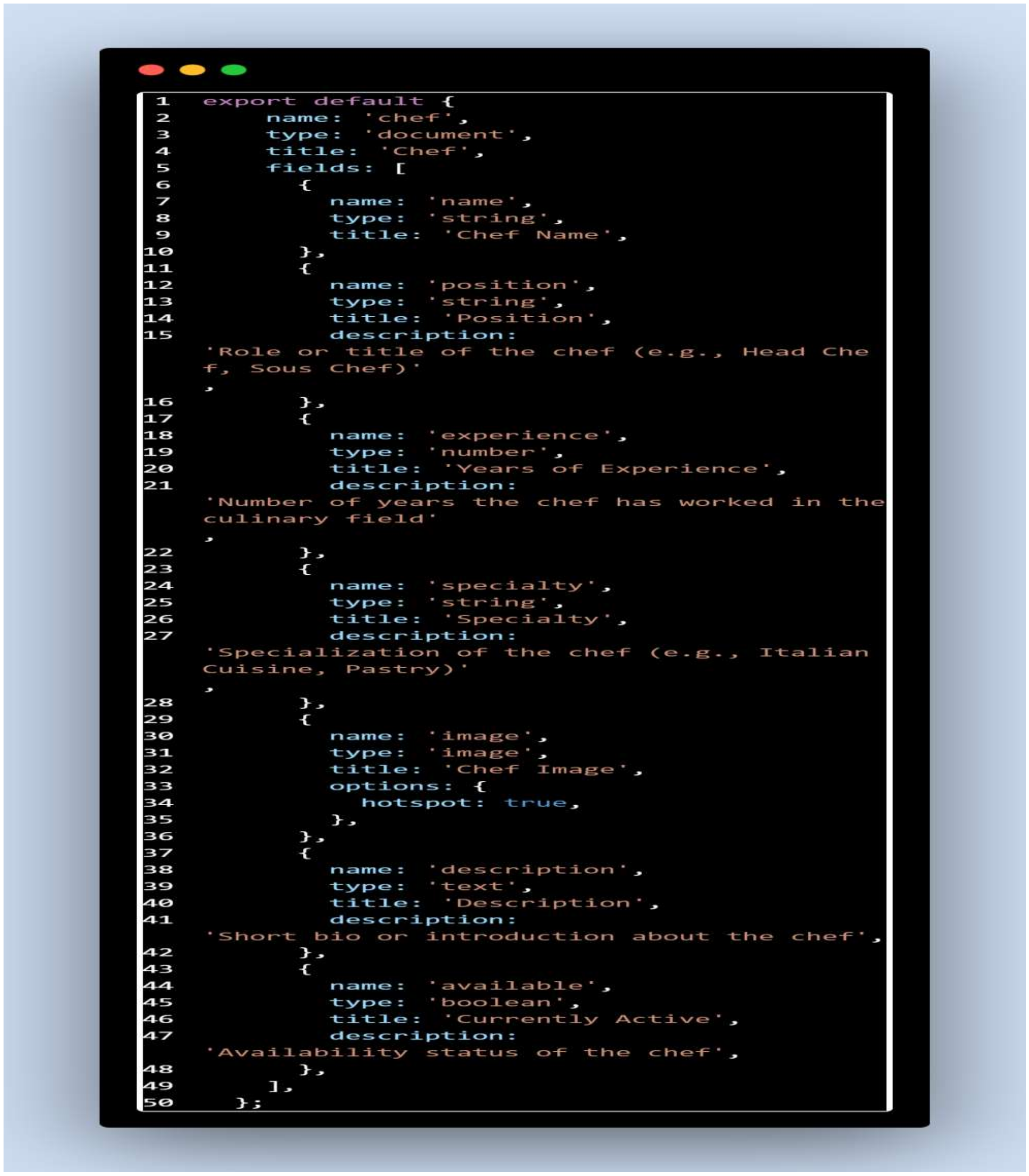


```
import { defineField, defineType } from 'sanity'

export default defineType({
  name: 'food',
  type: 'document',
  title: 'Food',
  fields: [
    {
      name: 'name',
      type: 'string',
      title: 'Food Name',
    },
    {
      name: 'slug',
      type: 'slug',
      title: 'Slug',
      options: {
        source: 'name',
        maxLength: 96,
      },
      validation: Rule => Rule.required()
    },
    {
      name: 'category',
      type: 'string',
      title: 'Category',
      description:
'Category of the food item (e.g., Burger, Sandwich, Drink, etc.)'
,
    },
    {
      name: 'price',
      type: 'number',
      title: 'Current Price',
    },
    {
      name: 'originalPrice',
      type: 'number',
      title: 'Original Price',
      description:
'Price before discount (if any)',
    },
    {
      name: 'tags',
      type: 'array',
      title: 'Tags',
      of: [{ type: 'string' }],
      options: {
        layout: 'tags',
      },
      description:
'Tags for categorization (e.g., Best Seller, Popular, New)'
,
    },
    {
      name: 'image',
      type: 'image',
      title: 'Food Image',
      options: {
        hotspot: true,
      },
    },
    {
      name: 'description',
      type: 'text',
      title: 'Description',
      description:
'Short description of the food item',
    },
    {
      name: 'available',
      type: 'boolean',
      title: 'Available',
      description:
'Availability status of the food item',
    },
  ],
})
```

Chefs:

By Taha Saif

Day 3 - API Integration Report – AS foods



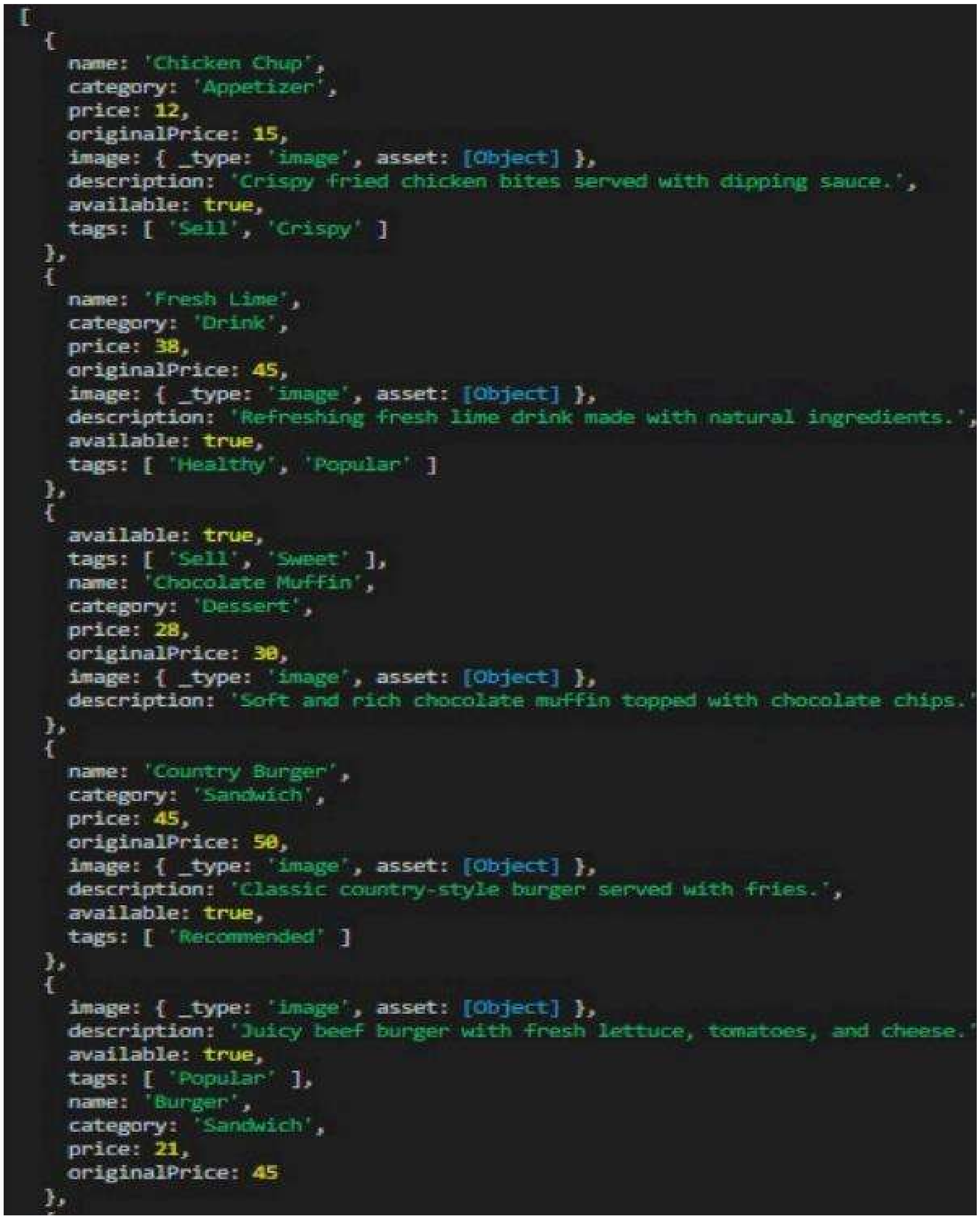
```
export default {
    name: 'chef',
    type: 'document',
    title: 'Chef',
    fields: [
      {
        name: 'name',
        type: 'string',
        title: 'Chef Name',
      },
      {
        name: 'position',
        type: 'string',
        title: 'Position',
        description:
'Role or title of the chef (e.g., Head Che
f, Sous Chef)'
,
      },
      {
        name: 'experience',
        type: 'number',
        title: 'Years of Experience',
        description:
'Number of years the chef has worked in the
culinary field'
,
      },
      {
        name: 'specialty',
        type: 'string',
        title: 'Specialty',
        description:
'Specialization of the chef (e.g., Italian
Cuisine, Pastry)'
,
      },
      {
        name: 'image',
        type: 'image',
        title: 'Chef Image',
        options: {
          hotspot: true,
        },
      },
      {
        name: 'description',
        type: 'text',
        title: 'Description',
        description:
'Short bio or introduction about the chef',
      },
      {
        name: 'available',
        type: 'boolean',
        title: 'Currently Active',
        description:
'Availability status of the chef',
      },
    ],
  };
```

API Calls

Food:

By Taha Saif

Day 3 - API Integration Report – AS foods

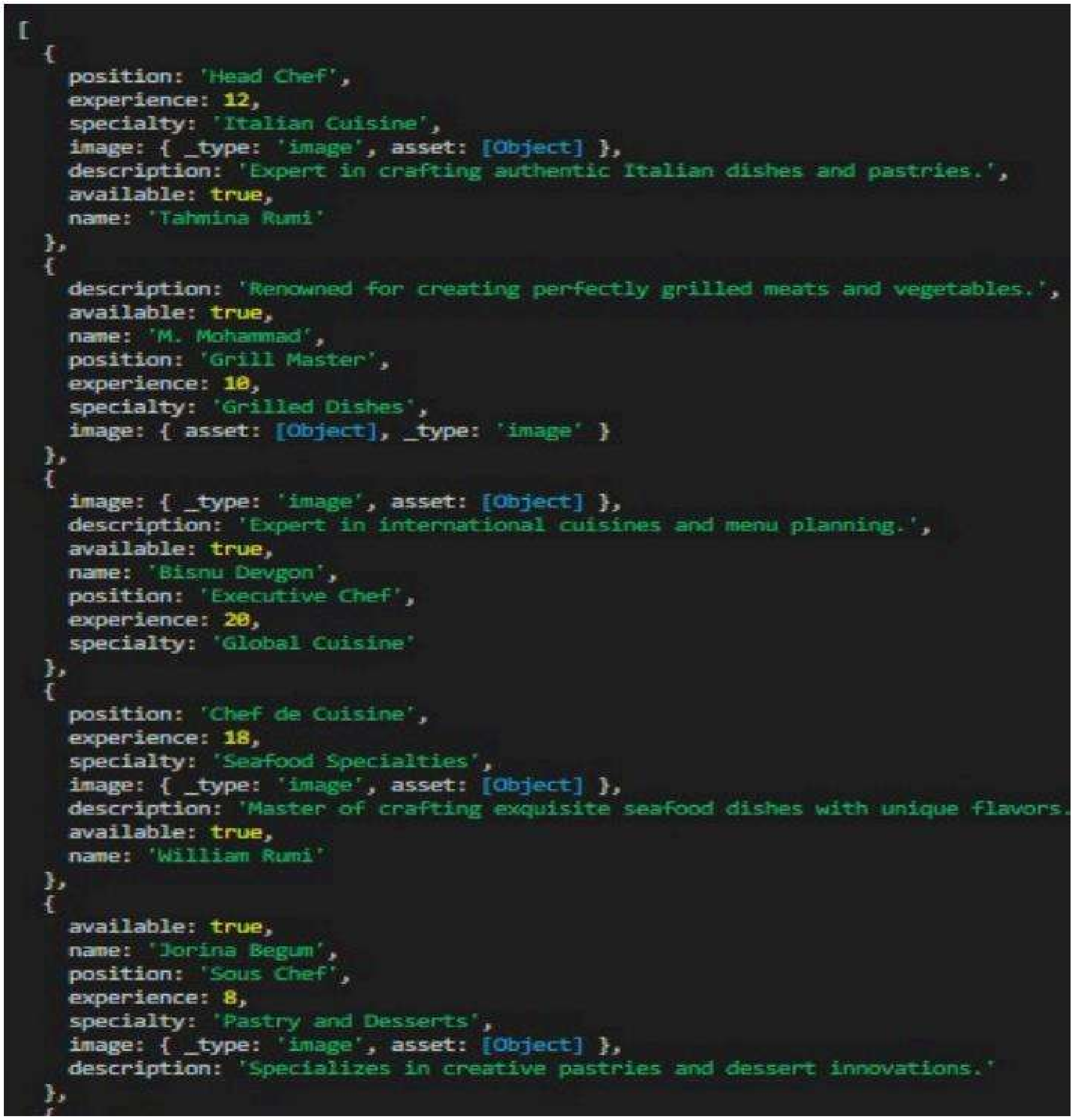


```
[
  {
    name: 'Chicken Chup',
    category: 'Appetizer',
    price: 12,
    originalPrice: 15,
    image: { _type: 'image', asset: [Object] },
    description: 'Crispy fried chicken bites served with dipping sauce.',
    available: true,
    tags: [ 'Sell', 'Crispy' ]
  },
  {
    name: 'Fresh Lime',
    category: 'Drink',
    price: 38,
    originalPrice: 45,
    image: { _type: 'image', asset: [Object] },
    description: 'Refreshing fresh lime drink made with natural ingredients.',
    available: true,
    tags: [ 'Healthy', 'Popular' ]
  },
  {
    available: true,
    tags: [ 'Sell', 'Sweet' ],
    name: 'Chocolate Muffin',
    category: 'Dessert',
    price: 28,
    originalPrice: 30,
    image: { _type: 'image', asset: [Object] },
    description: 'Soft and rich chocolate muffin topped with chocolate chips.'
  },
  {
    name: 'Country Burger',
    category: 'Sandwich',
    price: 45,
    originalPrice: 50,
    image: { _type: 'image', asset: [Object] },
    description: 'Classic country-style burger served with fries.',
    available: true,
    tags: [ 'Recommended' ]
  },
  {
    image: { _type: 'image', asset: [Object] },
    description: 'Juicy beef burger with fresh lettuce, tomatoes, and cheese.'
    available: true,
    tags: [ 'Popular' ],
    name: 'Burger',
    category: 'Sandwich',
    price: 21,
    originalPrice: 45
  },
```

Chefs:

By Taha Saif

Day 3 - API Integration Report – AS foods



```
[
  {
    position: 'Head Chef',
    experience: 12,
    specialty: 'Italian Cuisine',
    image: { _type: 'image', asset: [Object] },
    description: 'Expert in crafting authentic Italian dishes and pastries.',
    available: true,
    name: 'Tahmina Rumi'
  },
  {
    description: 'Renowned for creating perfectly grilled meats and vegetables.',
    available: true,
    name: 'M. Mohammad',
    position: 'Grill Master',
    experience: 10,
    specialty: 'Grilled Dishes',
    image: { asset: [Object], _type: 'image' }
  },
  {
    image: { _type: 'image', asset: [Object] },
    description: 'Expert in international cuisines and menu planning.',
    available: true,
    name: 'Bisnu Devgon',
    position: 'Executive Chef',
    experience: 20,
    specialty: 'Global Cuisine'
  },
  {
    position: 'Chef de Cuisine',
    experience: 18,
    specialty: 'Seafood Specialties',
    image: { _type: 'image', asset: [Object] },
    description: 'Master of crafting exquisite seafood dishes with unique flavors.
    available: true,
    name: 'William Rumi'
  },
  {
    available: true,
    name: 'Jorina Begum',
    position: 'Sous Chef',
    experience: 8,
    specialty: 'Pastry and Desserts',
    image: { _type: 'image', asset: [Object] },
    description: 'Specializes in creative pastries and dessert innovations.'
  },
```

Final check:

API Understanding	Schema Validation	Data Migration	API Integration in Next.js	Submission Preparation
✓	✓	✓	✓	✓

By Taha Saif