

Arishah

Friday 9-12 (Morning)

Day 3 - API Integration Report - [Q-Commerce]

Project Overview:

Project Name: Foodtuck

FoodTuck is an online food ordering platform that allows customers to order food from a variety of options available on the website. Customers can browse through the menu, add items to their cart, and place orders easily. The platform uses API integration for real-time data fetching and payment processing. We have integrated third-party payment APIs to provide secure transactions and order tracking APIs to keep customers updated on their order status.

API Integration Process for FoodTuck Marketplace

Reviewed API Documentation:

- I thoroughly examined the provided API documentation to understand the available endpoints (e.g., /foods, /chefs).
- I analyzed the structure of the data returned by the API, focusing on field names (such as name, price, image) and their respective data types (e.g., string, number, array).

Set Up API Calls:

I used Postman to test the API endpoints and verify that the data was being returned as expected.

Instead of using fetch, I opted for Axios to make GET requests to the API endpoints. This ensured better error handling and more control over the requests.

I created utility functions within my Next.js project to interact with the APIs, where I fetched data and logged the responses in the console to confirm the accuracy of the data.

Adjustments Made to Schemas:

- **Identified Schema Requirements:** To effectively import data from the external API (for foods and chefs), I first reviewed the existing schemas in my CMS (Sanity) to ensure they could accommodate the data structure required for both food items and chefs.
- **Modified the Food Schema:** The food schema was adjusted to hold data fetched from the API, such as food name, category, price, image, and tags. The following changes were made:
 - ○ **Added Fields:**
 - **name:** Stores the name of the food item.
 - **category:** The category of food (e.g., appetizer, main course, etc.).
 - **price:** The price of the food item.
 - **tags:** An array field to store any tags related to the food (e.g., vegetarian, spicy).
 - **description:** A field to store a brief description of the food item.
 - **image:** A reference to the image asset of the food.
 - **Field Adjustments:**
 - **Updated** the price field to support numeric values (for price calculation).
 - **Implemented** the image field as a reference, linking to assets uploaded through the API.
- **Modified the Chef Schema:** Similarly, I adjusted the chef schema to store data fetched from the chefs API. Changes made included:
 - **Added Fields:**
 - **name:** Stores the name of the chef.
 - **position:** The position or role of the chef (e.g., head chef, sous chef).

- **experience:** The experience level of the chef (e.g., years of experience).
- **specialty:** The chef's specialty or area of expertise (e.g., Italian cuisine, pastry).
- **description:** A brief description of the chef's background.
- **image:** A reference to the image asset of the chef.
- **Field Adjustments:**
 - Updated the experience field to allow numeric values, providing a better representation of the chef's expertise.

Migration Steps and Tools Used:

- **Preparation for Migration:** Before starting the migration process, I first reviewed the structure of the data that needed to be imported. This involved understanding the relationship between the different types of data (i.e., foods and chefs) and ensuring the target CMS ([Sanity](#)) had the necessary schemas to accommodate the data.
- **Selecting the Migration Tool:** For this migration, I used custom scripts written in JavaScript to fetch data from the external API and upload it to Sanity CMS. The following tools were used:
 - **Axios:** Used for making HTTP GET requests to fetch foods and chefs data from the provided API endpoints.
 - **Sanity Client:** Used for interacting with the Sanity CMS to upload data. The `@sanity/client` package was configured with the necessary project ID, dataset, and API token.
 - **Node.js Environment:** The migration script was written using Node.js, allowing easy integration with Sanity's API and efficient handling of data imports.
- **Steps Taken for Data Migration:**
 - **Fetching Data from API:**
 - I used Axios to make GET requests to the API endpoints for both foods and chefs. The data returned by the API was logged to verify its correctness.
 - Both API responses were stored in variables for further processing.

- **Image Uploading to Sanity:**
 - For each food and chef, I uploaded images to Sanity using the Sanity client. The images were fetched using Axios as an array buffer, then uploaded as assets to Sanity.
 - I ensured the images were successfully uploaded and linked to the appropriate food or chef entries as references.
- **Creating and Uploading Documents to Sanity:**
 - For each food item and chef, I created new documents using the modified food and chef schemas.
 - The food and chef documents were populated with data from the API, including image references, price, category, and other relevant fields.
 - These documents were then uploaded to the CMS using the `client.create()` method from the Sanity client.
- **Error Handling and Logging:**
 - Throughout the migration process, I implemented error handling to capture any issues that arose, such as failed API requests or image upload errors.
 - Detailed logs were maintained to track the success of each API call, image upload, and document creation, helping in troubleshooting if necessary.
- **Tools Used for Migration:**
 - **Axios:** To fetch the data from the API.
 - **Sanity Client:** For creating documents and managing assets.
 - **Node.js:** For writing the migration script and executing it on the server.
- The migration was completed successfully with all food items and chefs uploaded to the Sanity CMS.

Task	Status
API Documentation Review	✓
Schema Structure Validation	✓
Data Migration Process	✓
API Integration in Next.js	✓
Submission Preparation	✓

```
[  
  {  
    "name": "Fresh Lime",  
    "category": "Drink",  
    "price": 38,  
    "originalPrice": 45,  
    "tags": [  
      "Healthy",  
      "Popular"  
    ],  
    "image": "https://sanity-nextjs-rouge.vercel.app/food/food-1.png",  
    "description": "Refreshing fresh lime drink made with natural ingredients.",  
    "available": true  
  },  
  {  
    "name": "Chocolate Muffin",  
    "category": "Dessert",  
    "price": 28,  
    "originalPrice": 30,  
    "tags": [  
      "Sell",  
      "Sweet"  
    ],  
    "image": "https://sanity-nextjs-rouge.vercel.app/food/food-2.png",  
    "description": "Soft and rich chocolate muffin topped with chocolate chips.",  
    "available": true  
  },  
  {  
    "name": "Burger",  
    "category": "Sandwich",  
    "price": 21,  
    "originalPrice": 45,  
    "tags": [  
      "Popular"  
    ],  
    "image": "https://sanity-nextjs-rouge.vercel.app/food/food-3.png",  
    "description": "Juicy beef burger with fresh lettuce, tomatoes, and cheese.",  
    "available": true  
  },  
  {  
    "name": "Country Burger",  
    "category": "Sandwich",  
    "price": 45,  
    "originalPrice": 50,  
    "tags": [  
      "Recommended"  
    ],  
    "image": "https://sanity-nextjs-rouge.vercel.app/food/food-4.png",  
    "description": "Classic country-style burger served with fries.",  
    "available": true  
  },  
  {  
    "name": "Pizza",  
    "category": "Main Course",  
    "price": 43,  
    "originalPrice": 50,  
    "tags": [  
      "Cheesy",  
      "Vegetarian"  
    ],  
    "image": "https://sanity-nextjs-rouge.vercel.app/food/food-5.png",  
    "description": "Delicious vegetarian pizza topped with fresh vegetables and cheese.",  
    "available": true  
  },  
  {  
    "name": "Chicken Chup",  
    "category": "Appetizer",  
    "price": 12,  
    "originalPrice": 15,  
    "tags": [  
      "Sell",  
      "Crispy"  
    ],  
    "image": "https://sanity-nextjs-rouge.vercel.app/food/food-6.png",  
    "description": "Crispy fried chicken bites served with dipping sauce.",  
    "available": true  
  }  
]
```

```
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' });
    const buffer = Buffer.from(response.data);
    const asset = await client.assets.upload('image', buffer, {
      filename: imageUrl.split('/')[1].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 importData() {
  try {
    console.log('Fetching food, chef data from API...');

    // API endpoint containing data
    const $promise = [];
    $promise.push(
      axios.get('https://sanity-nextjs-rouge.vercel.app/api/foods')
    );
    $promise.push(
      axios.get('https://sanity-nextjs-rouge.vercel.app/api/chefs')
    );

    const [foodsResponse, chefsResponse] = await Promise.all($promise);
    const foods = foodsResponse.data;
    const chefs = chefsResponse.data;

    for (const food of foods) {
      console.log(`Processing food: ${food.name}`);

      let imageRef = null;
      if (food.image) {
        imageRef = await uploadImageToSanity(food.image);
      }

      const sanityFood = {
        _type: 'food',
        name: food.name,
        category: food.category || null,
        price: food.price,
        originalPrice: food.originalPrice || null,
        tags: food.tags || [],
        description: food.description || '',
        available: food.available !== undefined ? food.available : true,
        image: imageRef
      ? {
          _type: 'image',
          asset: {
            _type: 'reference',
            _ref: imageRef,
          }
        }
      : undefined,
      };
      console.log(`Uploading food to Sanity: ${sanityFood.name}`);
      const result = await client.create(sanityFood);
      console.log(`Food uploaded successfully: ${result._id}`);
    }

    for (const chef of chefs) {
      console.log(`Processing chef: ${chef.name}`);

      let imageRef = null;
      if (chef.image) {
        imageRef = await uploadImageToSanity(chef.image);
      }

      const sanityChef = {
        _type: 'chef',
        name: chef.name,
        position: chef.position || null,
        experience: chef.experience || 0,
        speciality: chef.speciality || '',
        description: chef.description || '',
        available: chef.available !== undefined ? chef.available : true,
        image: imageRef
      ? {
          _type: 'image',
          asset: {
            _type: 'reference',
            _ref: imageRef,
          }
        }
      : undefined,
      };
      console.log(`Uploading chef to Sanity: ${sanityChef.name}`);
      const result = await client.create(sanityChef);
      console.log(`Chef uploaded successfully: ${result._id}`);
    }

    console.log('Data import completed successfully!');
  } catch (error) {
    console.error(`Error importing data: ${error}`);
  }
}
```

```
const food = {
  name: 'food',
  type: 'document',
  title: 'Food',
  fields: [
    {
      name: 'name',
      type: 'string',
      title: 'Food Name',
    },
    {
      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,
      },
      description: 'Small Images',
      of: [{ type: 'image' }],
      description: 'Array of small images for the food item',
    },
    {
      name: 'description',
      type: 'text',
      title: 'Description',
      description: 'Short description of the food item',
    },
    {
      name: 'detailDescription',
      type: 'text',
      title: 'Detailed Description',
      description: 'Detailed description of the food item',
    },
    {
      name: 'available',
      type: 'boolean',
      title: 'Available',
      description: 'Availability status of the food item',
    },
    {
      name: 'id',
      type: 'string',
      title: 'Food ID',
      description: 'Unique identifier for the food item',
    },
    {
      name: 'rating',
      type: 'number',
      title: 'Rating',
      description: 'Overall rating of the food item (e.g., out of 5)',
    },
    {
      name: 'reviews',
      type: 'number',
      title: 'Reviews',
      description: 'Numerical ratings for individual reviews',
    },
    {
      name: 'benefits',
      type: 'array',
      title: 'Benefits',
      of: [{ type: 'string' }],
      description: 'Array of benefits associated with the food item',
    },
  ],
};

export default food
```

```
1 const chefSchema = {
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',
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  };
51
52 export default chefSchema;
```

```
import { type SchemaTypeDefinition } from 'sanity';
import chef from './chefs';
import food from './foods';

export const schema: { types: SchemaTypeDefinition[] } =
{ types: [food, chef],
};
```


Food

Chicken Shawarma

Food Name
Chicken Shawarma

Category
Category of the food item (e.g., Burger, Sandwich, Drink, etc.)
Non Veg

Current Price
700

Original Price
Price before discount (if any)
760

Published 4 hr. ago

Chicken Shawarma

Chef

Mike Johnson

Chef Name
Mike Johnson

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

Years of Experience
Number of years the chef has worked in the culinary field
20

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

Published 2 hr. ago

Mike Johnson

Foodtuck

- [Home](#)
- [Menu](#)
- [Blog](#)
- [About](#)
- [Shop](#)
- [Contact](#)

Our Shop

Home > [Shop](#)



Lava Pizza
\$500



Classic BLT
\$35 \$40



Pulled Pork
\$19 \$23



Margherita Pizza
\$20 \$27



Fruit Punch
\$16 \$19



Pepperoni Pizza
\$17 \$20



Baklava
\$499



Prawn Curry
\$500 \$600



Cold Brew Coffee
\$9 \$11



Fruit Tart
\$370 \$399



Four Cheese Pizza
\$14 \$19



Club Sandwich
\$7 \$10



Grilled Lamb Chops
\$399 \$460



Turkey and Avocado
\$23 \$27

« 1 2 3 »

Still You Need Our Support?

Don't wait, make a smart & logical quote here. It's pretty easy.

[Subscribe Now](#)

About Us

We are a team dedicated to providing the best services. Learn more about our mission and vision.

 **Opening Hours**
Mon - Sat(8.00 - 6.00)
Sunday - Closed

Help?

[Home](#)
[Menu](#)
[Blog](#)
[Pages](#)
[About](#)
[Shop](#)
[Contact](#)

Help?

[FAQ](#)
[Terms & Conditions](#)
[Reporting](#)
[Documentation](#)
[Support Policy](#)
[Privacy Help](#)

Recent Posts


20 Feb 2022
Keep Your Business


20 Feb 2022
Keep Your Business


20 Feb 2022
Keep Your Business

Foodtuck

Home Menu Blog About Shop Contact

Shop Details

Home > Shop Details

Chicken Shawarma



Thinly sliced marinated chicken, grilled and wrapped in pita bread with garlic sauce and pickles.

\$700 \$760

4 rating | 25 reviews

1

Add to Cart

Add to Wishlist

Category: Non Veg

Share: [Facebook](#) [Twitter](#) [Instagram](#) [Pinterest](#) [Email](#)

Description

Chicken Shawarma is a beloved Middle Eastern street food made by slow-cooking marinated chicken on a vertical rotisserie. The chicken is seasoned with a rich blend of spices such as cumin, coriander, turmeric, garlic, and paprika, giving it a bold, smoky flavor. Once cooked, the chicken is thinly sliced and often served wrapped in pita or flatbread, along with a variety of toppings such as lettuce, tomatoes, cucumbers, onions, and tahini sauce or garlic sauce. This dish is perfect for those who enjoy a flavorful, savory, and satisfying meal.

Key Benefits

- "High in protein, which supports muscle repair and growth."
- "Rich in vitamins such as B6, which is important for metabolism and brain health."
- "Contains antioxidants from spices like turmeric and garlic, which help fight inflammation."
- "Provides a quick energy boost, making it a great meal for busy days."



Veggie Burger

\$8 \$15



Fish and Chips

\$800 \$750



Meat Lovers Pizza

\$18 \$20



Buffalo Chicken Pizza

\$22

Still You Need Our Support?

Don't wait, make a smart & logical quote here. It's pretty easy.

Enter Your Email [Subscribe Now](#)

About Us

We are a team dedicated to providing the best services. Learn more about our mission and vision.



Opening Hours
Mon - Sat(8.00 - 6.00)
Sunday - Closed

Help?

Home

Menu

Blog

Pages

About

Shop

Contact

Help?

FAQ

Terms & Conditions

Reporting

Documentation

Support Policy

Privacy Help

Recent Posts

20 Feb 2022

Keep Your Business

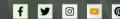
20 Feb 2022

Keep Your Business

20 Feb 2022

Keep Your Business

Copyright © 2022 by Ayeman. All Rights Reserved.



Foodtuck

Home Menu Blog About Shop Contact

Search icon

Profile icon

Cart icon

Notification icon (1)

Our Chef

Home > Chef

John Doe
Head Chef
Italian Cuisine

John is a highly skilled chef with over 15 years of experience specializing in Italian dishes.

Currently Inactive

Jane Smith
Sous Chef
Pastries and Desserts

Jane is known for her innovative pastries and has been honing her skills for over 10 years.

Currently Inactive

Bisnu Devgon
Executive Chef
Global Cuisine

Expert in international cuisines and menu planning.

Currently Active

Mike Johnson
Executive Chef
Asian Fusion

Mike brings 20 years of experience, focusing on blending traditional Asian flavors with modern techniques.

Currently Inactive

M. Mohammad
Grill Master
Grilled Dishes

Renowned for creating perfectly grilled meats and vegetables.

Currently Active

William Rumi
Chef de Cuisine
Seafood Specialties

Master of crafting exquisite seafood dishes with unique flavors.

Currently Active

Tahmina Rumi
Head Chef
Italian Cuisine

Expert in crafting authentic Italian dishes and pastries.

Currently Active

Jorina Begum
Sous Chef
Pastry and Desserts

Specializes in creative pastries and dessert innovations.

Currently Active

Munna Kathy
Culinary Instructor
Asian Fusion

Pioneer in Asian fusion dishes blending traditional flavors with modern techniques.

Currently Active

Still You Need Our Support?

Don't wait, make a smart & logical quote here. It's pretty easy.

Enter Your Email

Subscribe Now

About Us

We are a team dedicated to providing the best services. Learn more about our mission and vision.

Opening Hours

Mon - Sat(8.00 - 6.00)
Sunday - Closed

Help?

Home

Menu

Blog

Pages

About

Shop

FAQ

Terms & Conditions

Reporting

Documentation

Support Policy

Recent Posts

20 Feb 2022 **Keep Your Business**

20 Feb 2022 **Keep Your Business**

20 Feb 2022 **Keep Your Business**