

Day 3 - API Integration Report - Food Marketplace

Prepared by: Aisha Junaid

Date: [18-jan-2025]

1. Introduction

This report details the accomplishments of Day 3 for the Food Marketplace project developed by Team 9. The focus of this stage was to integrate the Sanity CMS API, define schemas for food-related data, and execute the migration process to import structured information into the local database.

2. API Integration

The Sanity CMS API was integrated to establish seamless communication between the backend and the content management system. The key configurations included:

Project ID and API Token were utilized to connect to Sanity CMS.

Environment variables were employed to securely store sensitive credentials.

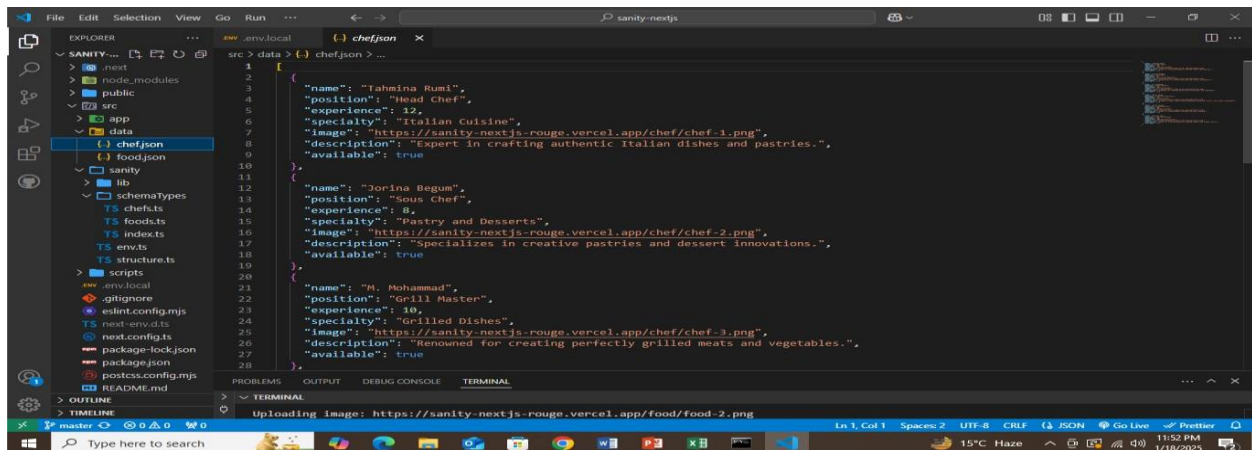
The integration laid the foundation for structured data storage and retrieval.

3. Schema Creation

Two schemas were developed to organize the data effectively:

Food Schema (food.ts)

This schema was designed to store food item detail:



```
1 {
2
3
4   "name": "Tahmina Rumi",
5   "position": "Head Chef",
6   "experience": 12,
7   "specialty": "Italian Cuisine",
8   "image": "https://sanity-nextjs-rouge.vercel.app/chef/chef-1.png",
9   "description": "Expert in crafting authentic Italian dishes and pastries.",
10  "available": true
11 }
12
13 {
14   "name": "Jorina Begum",
15   "position": "Sous Chef",
16   "experience": 8,
17   "specialty": "Pastry and Desserts",
18   "image": "https://sanity-nextjs-rouge.vercel.app/chef/chef-2.png",
19   "description": "Specializes in creative pastries and dessert innovations.",
20   "available": true
21 }
22
23 {
24   "name": "M. Mohammad",
25   "position": "Grill Master",
26   "experience": 10,
27   "specialty": "Grilled Dishes",
28   "image": "https://sanity-nextjs-rouge.vercel.app/chef/chef-3.png",
29   "description": "Renowned for creating perfectly grilled meats and vegetables.",
30   "available": true
31 }
```

Fields:

Name: Name of the food item (String).

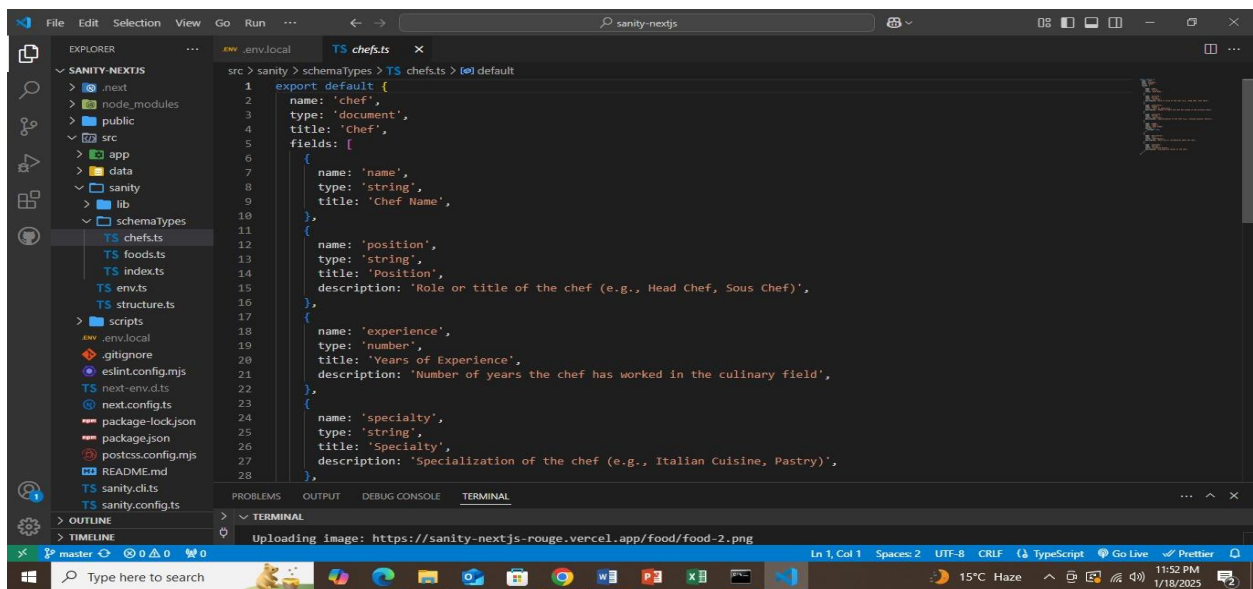
Description: Details about the food item (Text).

Price: Price of the food item (Number).

Image: An image representing the food item (Image).

Chef Schema (chef.ts)

This schema was created to store details about chefs associated with the food items.



Fields:

Name: Name of the chef (String).

Bio: Short biography or description of the chef (Text).

Photo: Chef's photograph (Image).

These schemas ensured structured and queryable data to power the Food Marketplace application.

4. Data Migration

The data migration process involved transferring food and chef data from Sanity CMS to the local application. Below are the steps:

Script Setup

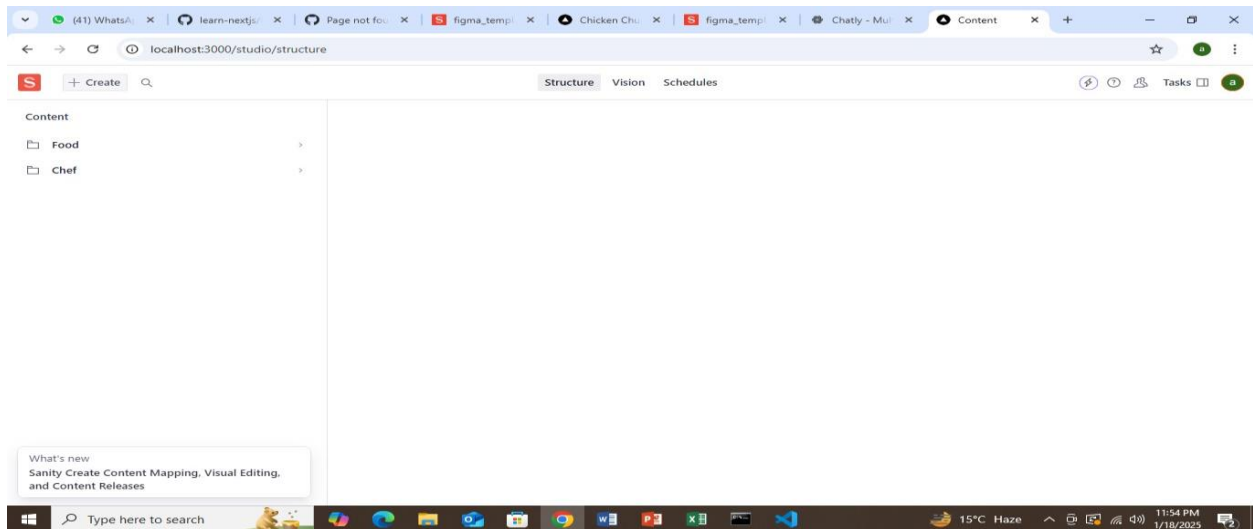
A scripts folder was created at the root directory.

A migration script named `importData.mjs` was written to handle the transfer of data.

Execution

The migration process was triggered using the following command in the terminal:

`npm run importData`



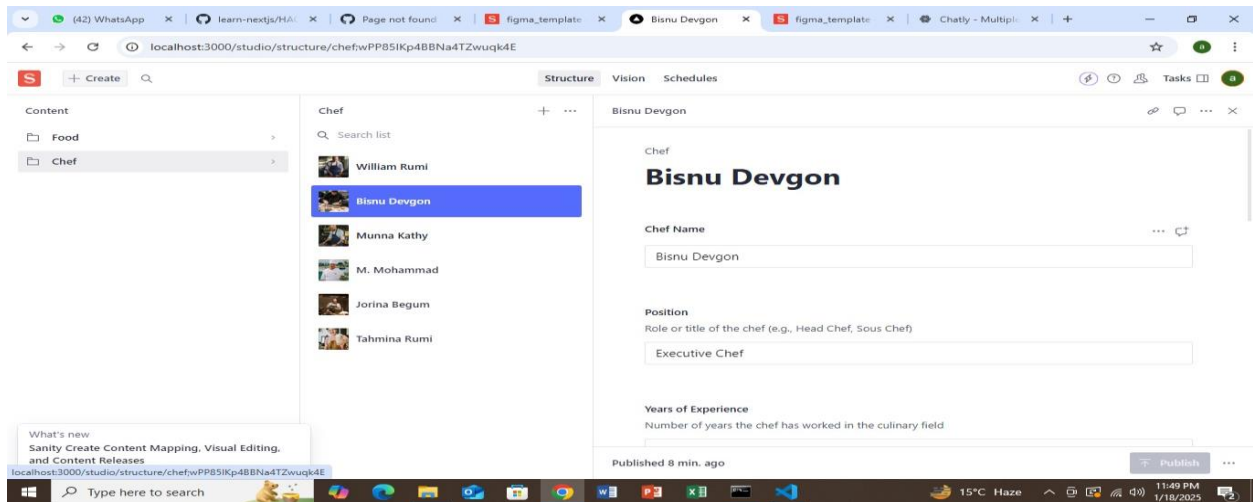
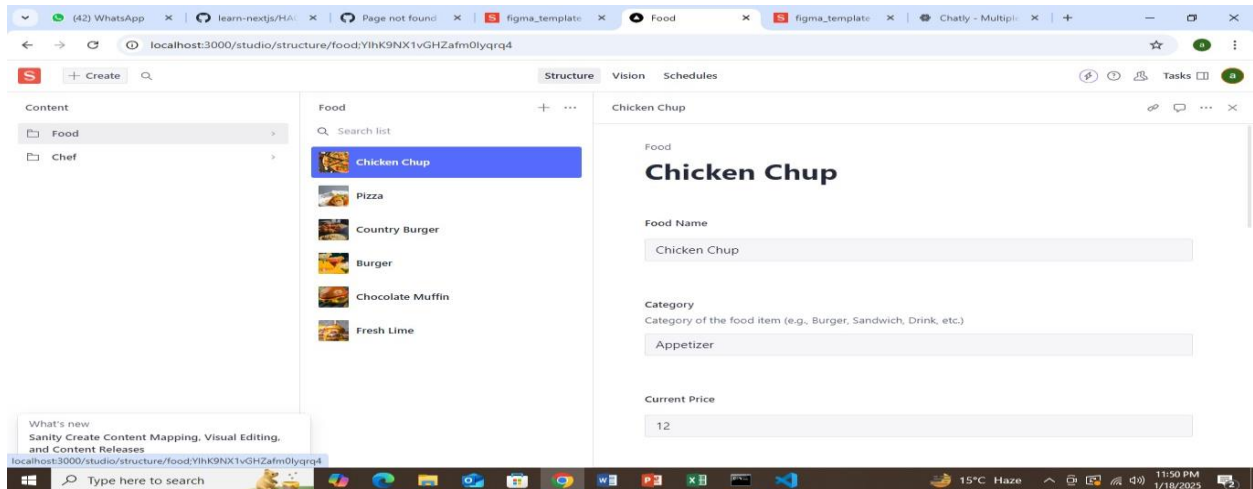
This command executed the script, which imported food and chef data into the local database.

Verification

After the migration, the following steps verified its success:

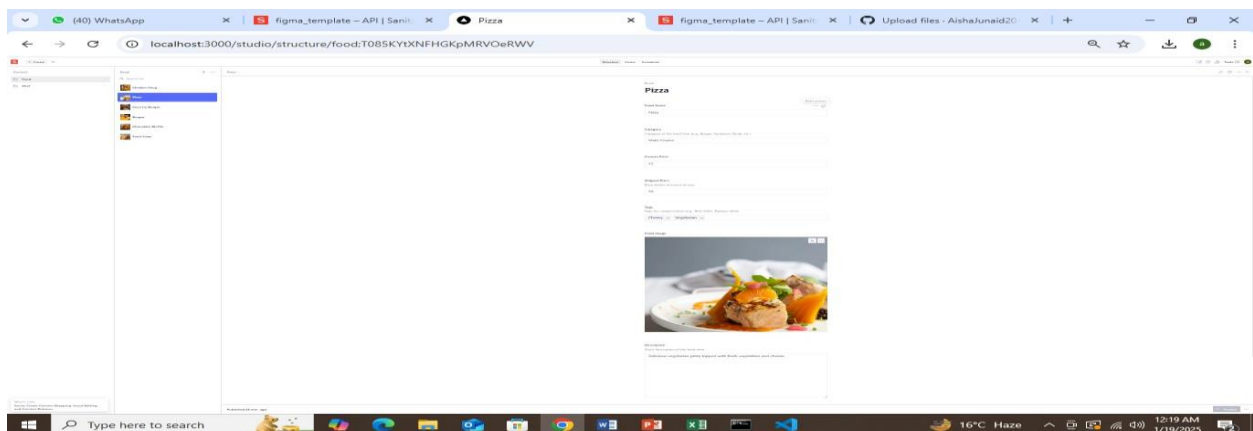
The project was run locally, and the `/studio/structure` endpoint was accessed.

The same Sanity CMS account was logged into to confirm the visibility of migrated data.



5. Migration Script

Below is the code snippet used in the importData.mjs script to migrate the data:



6. Screenshots

Add the following screenshots when preparing the report in MS Word:

Screenshot 1: Schema folder displaying food.ts and chef.ts files.

Screenshot 2: Output of the npm run importData command.

Screenshot 3: Sanity Studio showing the migrated food and chef data.

7. Conclusion

On Day 3, successfully integrated the Sanity CMS API and completed data migration for the Food Marketplace project. The highlights include:

Developing the food and chef schemas to organize data efficiently.

Automating the migration process through the importData.mjs script.

Verifying the successful migration by inspecting the Sanity Studio structure.

These achievements provide a robust backend setup for the Food Marketplace, paving the way for seamless frontend integration and user interaction in the upcoming stages of development.