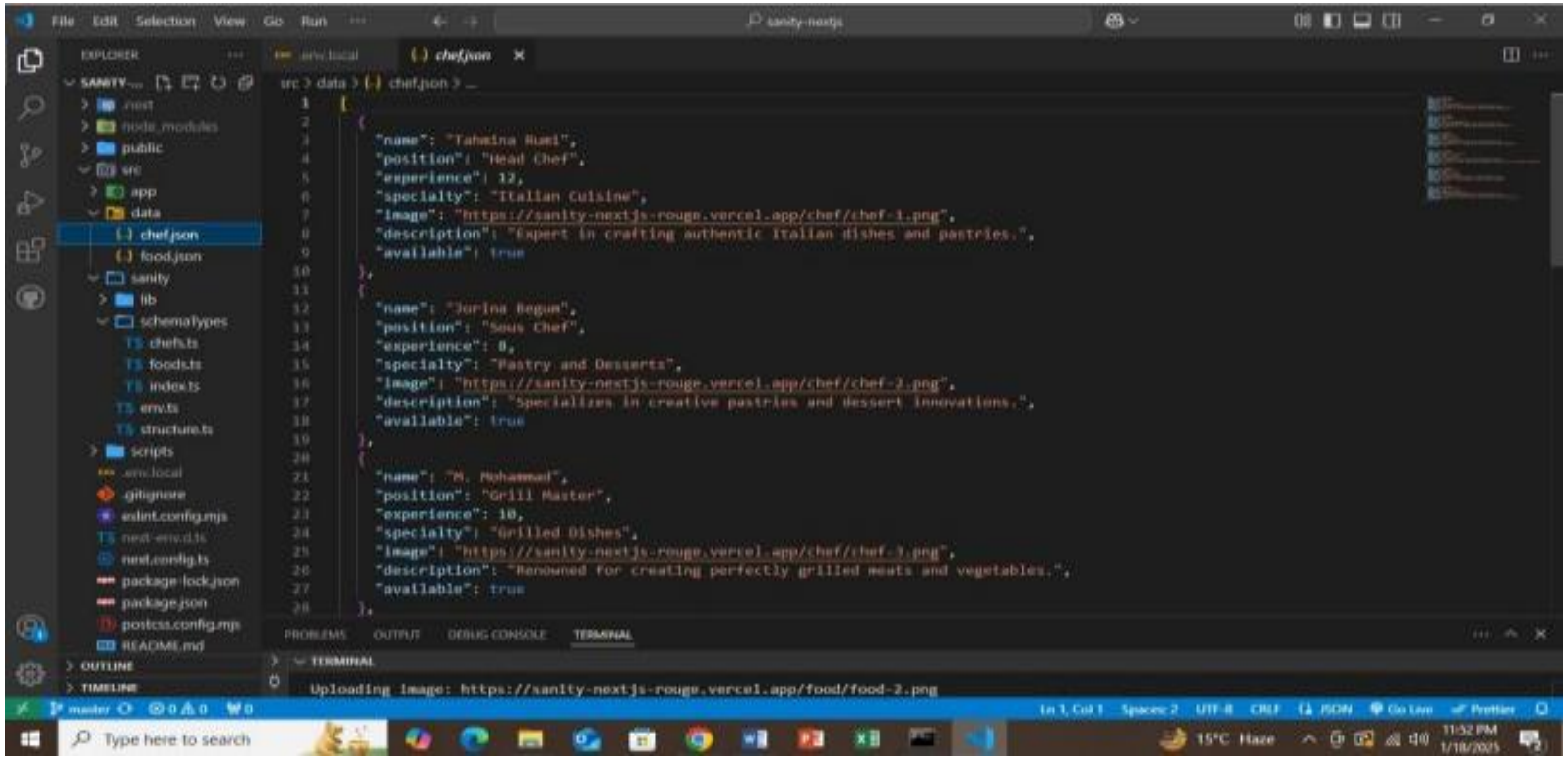
Day 3

API Integration Report - Food Marketplace

- **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:



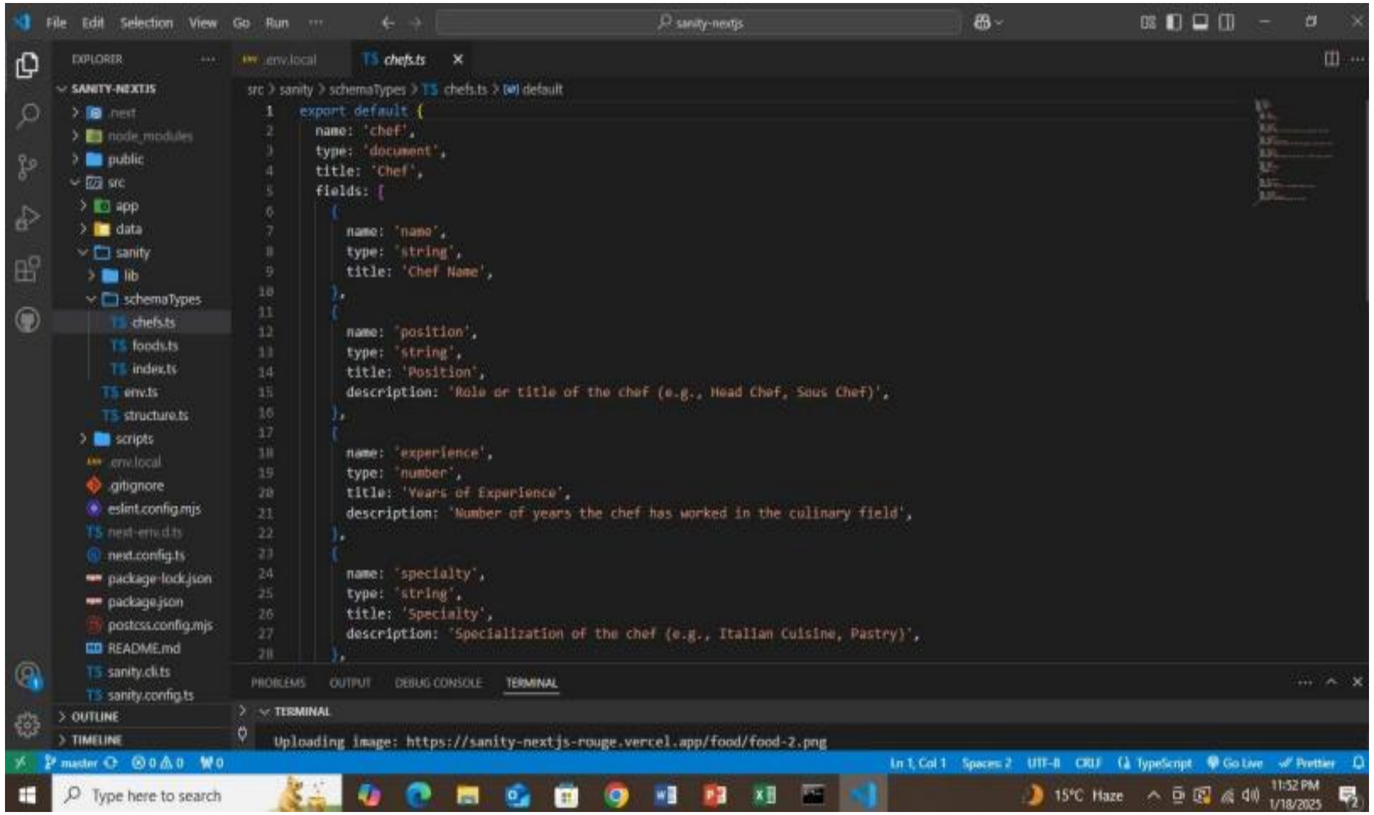
```
[
  {
    "name": "Tahmina Rumi",
    "position": "Head Chef",
    "experience": 12,
    "specialty": "Italian Cuisine",
    "image": "https://sanity-nextjs-rouge.vercel.app/chef/chef-1.png",
    "description": "Expert in crafting authentic Italian dishes and pastries.",
    "available": true
  },
  {
    "name": "Jorina Begum",
    "position": "Sous Chef",
    "experience": 8,
    "specialty": "Pastry and Desserts",
    "image": "https://sanity-nextjs-rouge.vercel.app/chef/chef-2.png",
    "description": "Specializes in creative pastries and dessert innovations.",
    "available": true
  },
  {
    "name": "M. Mohammad",
    "position": "Grill Master",
    "experience": 10,
    "specialty": "Grilled Dishes",
    "image": "https://sanity-nextjs-rouge.vercel.app/chef/chef-3.png",
    "description": "Renowned for creating perfectly grilled meats and vegetables.",
    "available": true
  },
```

```
Uploading image: https://sanity-nextjs-rouge.vercel.app/food/food-2.png
```

Prepared by: **Aqsa Mansoor Qurayshi**

Fields:

- Name: Name of the food item (String).
- Description: Details about the food item (Text).
- Price: Price of the food item (Number). Im
- age: 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.



src > sanity > schemaTypes > chefs.ts > default

```
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 Chef, 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)',
    },
```

```
Uploading image: https://sanity-nextjs-rouge.vercel.app/food/food-2.png
```

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 query able 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.

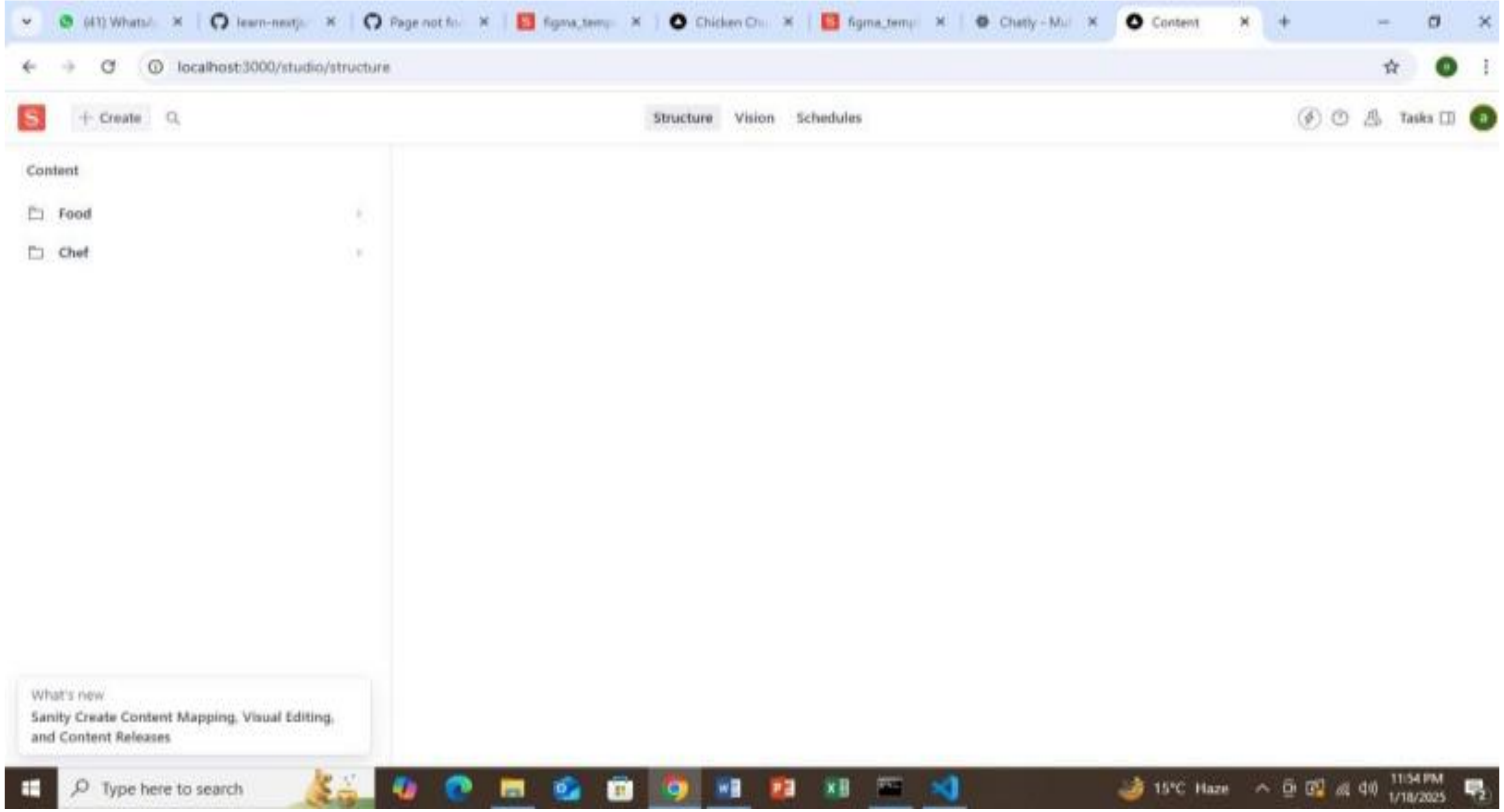
The migration process was triggered using the following command in the terminal: **npm run importData**

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

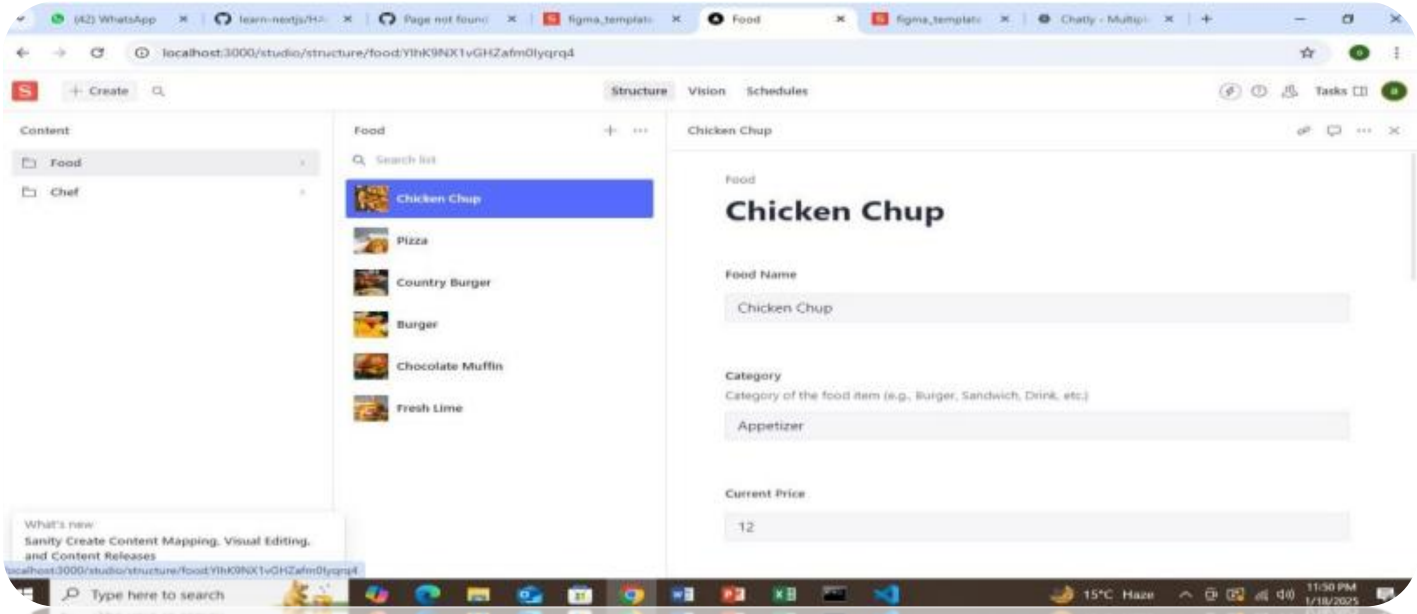
Verification

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.



What's new
Sanity Create Content Mapping, Visual Editing, and Content Releases



localhost:3000/studio/structure/food;YIhK9NX1vGHZafm0lyqrq4
Create
Structure
Vision
Schedules
Tasks
Content
Food
Chef

Food

- Chicken Chup
- Pizza
- Country Burger
- Burger
- Chocolate Muffin
- Fresh Lime

Food

Chicken Chup

Food Name

Chicken Chup

Category

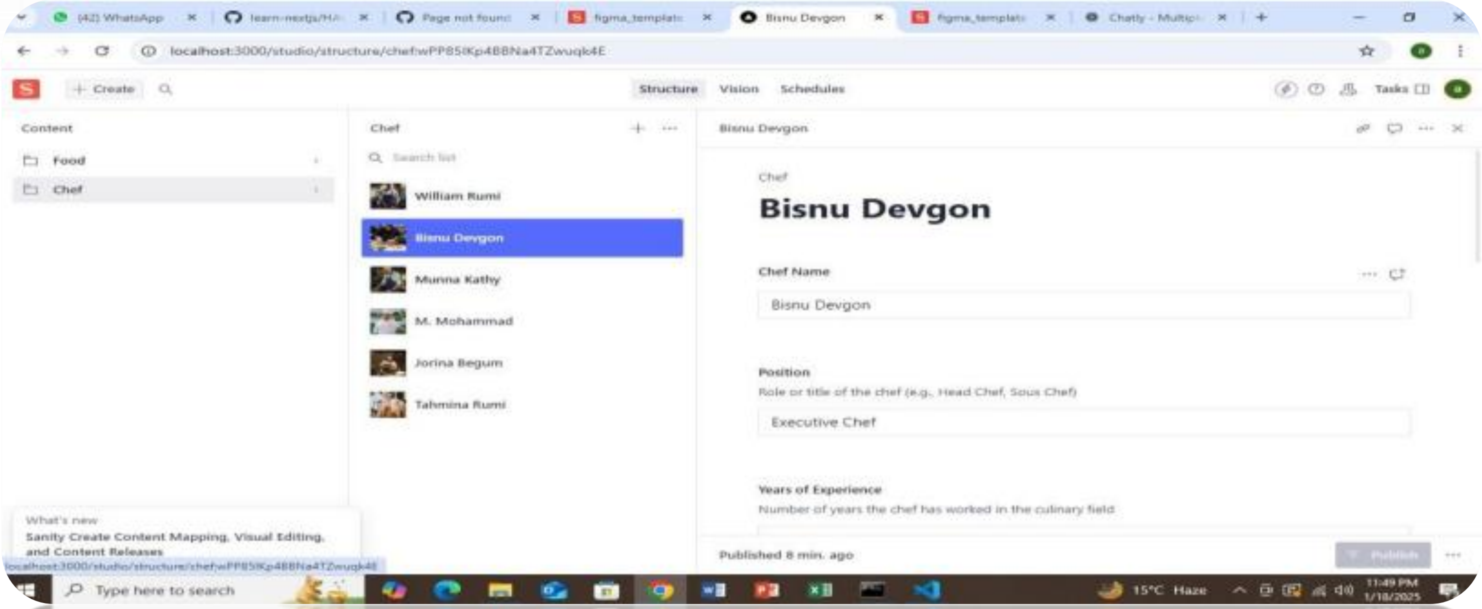
Category of the food item (e.g., Burger, Sandwich, Drink, etc.)

Appetizer

Current Price

12

What's new
Sanity Create Content Mapping, Visual Editing, and Content Releases
Type here to search
15°C Haze
11:50 PM
1/18/2025



localhost:3000/studio/structure/chef;wPP85lKp4BBNa4TZwuqk4E
Create
Structure
Vision
Schedules
Tasks
Content
Food
Chef

Chef

- William Rumi
- Bisnu Devgon
- Munna Kathy
- M. Mohammad
- Jorina Begum
- Tahmina Rumi

Chef

Bisnu Devgon

Chef Name

Bisnu Devgon

Position

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

Executive Chef

Years of Experience

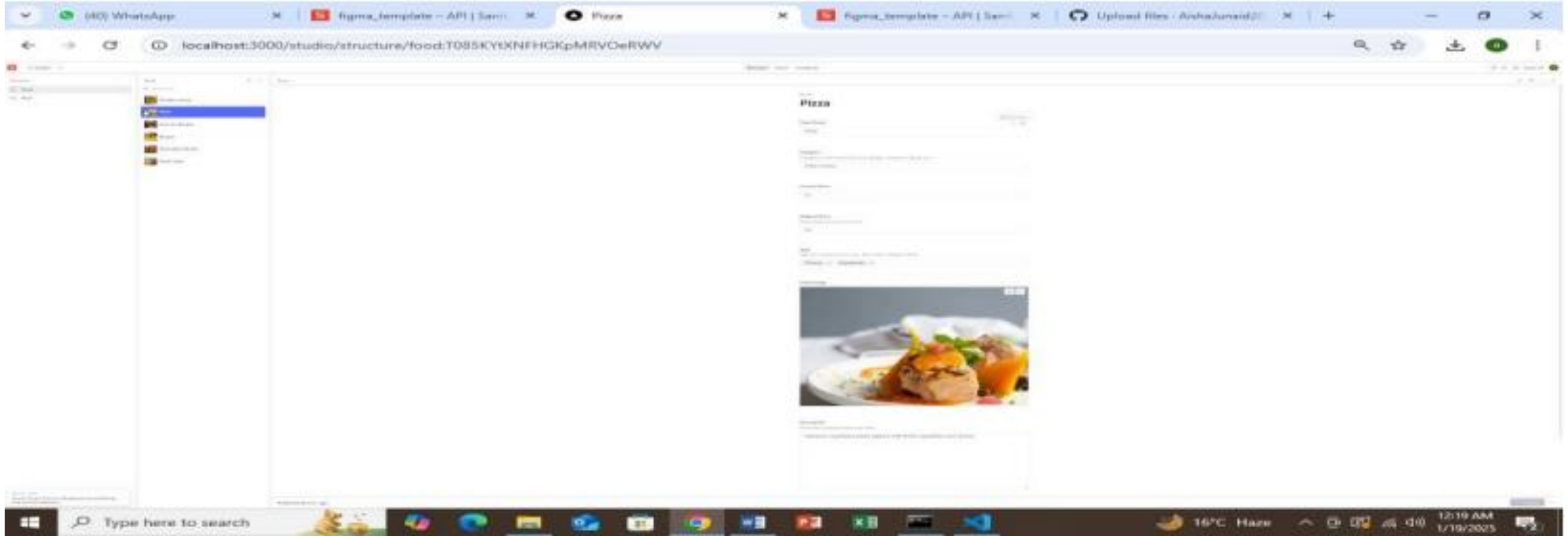
Number of years the chef has worked in the culinary field

Published 8 min. ago

Publish
What's new
Sanity Create Content Mapping, Visual Editing, and Content Releases
Type here to search
15°C Haze
11:49 PM
1/18/2025

5. Migration Script

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


(40) WhatsApp
figma_template - API | Sani
Pizza
figma_template - API | Sani
Upload files · AishaJunaid
localhost:3000/studio/structure/food;T085KYtXNFHGKpMRVOeRWV
Pizza
Type here to search
16°C Haze
12:19 AM
1/19/2025

6. Screenshots

Add the following screenshots when preparing the report in PowerPoint :
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.