Architecture Overview

- Frontend: Next.js and React
- Backend: Sanity CMS and API
- Database: Sanity integrated database

Components

1. Frontend
   - Built using Next.js and React
   - Users can order their favorite food items
2. Sanity CMS
   - Used to manage website content
   - Administrators can update content
3. API
   - Handles data transfer between frontend and backend
   - Processes user orders
4. Database
   - Stores website data using Sanity's integrated database

Workflow

1. User Order
   - User orders food items on the website
   - Order details sent to backend via API
2. Order Processing
   - Backend processes order details
   - Updates order status and notifies user
3. Content Management
   - Administrators update website content on CMS
   - Content changes reflected on frontend via API

Technology Stack

- Frontend: Next.js, React, CSS
- Backend: Sanity CMS, API, Node.js
- Database: Sanity integrated database

Benefits

- Scalable and flexible architecture
- Easy content management and updates

- Fast and secure data transfer
- Seamless user experience

Here is the data flow diagram for the food Q-commerce website architecture: