Name: Muhammad Bilal Qureshi

Roll Number: 00075879

Day 2 – Technical planning for [Food Tuck Q Commerce Marketplace]

Technical Planning for Quick Commerce Food Truck Website

Tech Stack:

- **Frontend:** Next.js (React framework)
- **Styling:** Tailwind CSS
- **Backend/CMS:** Sanity.io
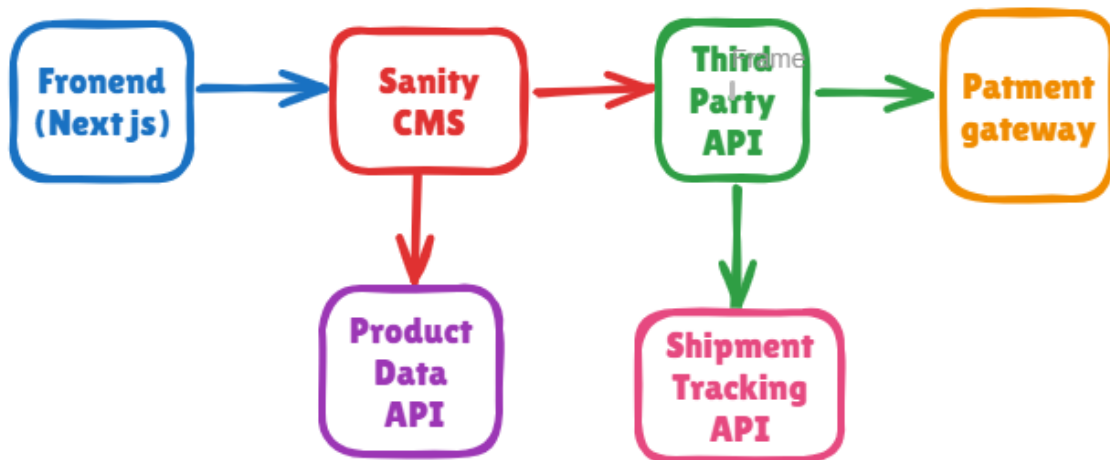- **APIs:** Third-party APIs for shipment, payment, and real-time order tracking

Core Features:

1. **User Authentication** – Secure login/signup using NextAuth.js or Firebase
2. **Product Management** – Dynamic food menu managed via Sanity CMS
3. **Order Processing** – Real-time order updates using WebSockets or Polling
4. **Shipment Integration** – First shipment API for fast delivery tracking
5. **Payment Gateway** – Stripe/Razorpay integration for secure transactions
6. **Cart & Checkout** – Persistent cart with optimized checkout flow
7. **Admin Panel** – Dashboard for order and inventory management

Deployment & Optimization:

- **Hosting:** Vercel for seamless Next.js deployment
- **Database:** Sanity.io for structured content management
- **SEO & Performance:** SSR/ISR for fast page loads, Image Optimization, and caching strategies.

SYSTEM ARCHITECTURE OVERVIEW BY MUHAMMAD BILAL QURESHI

```
Fronend      →    Sanity    →    Third      →    Patment
(Next js)         CMS            Party           gateway
                                 API
                   ↓              ↓
                Product       Shipment
                Data          Tracking
                API           API
```

User Registeration

```
User
 ↓
Sign-Up    →    Google Auth
                    ↓
                Sanity CMS
```
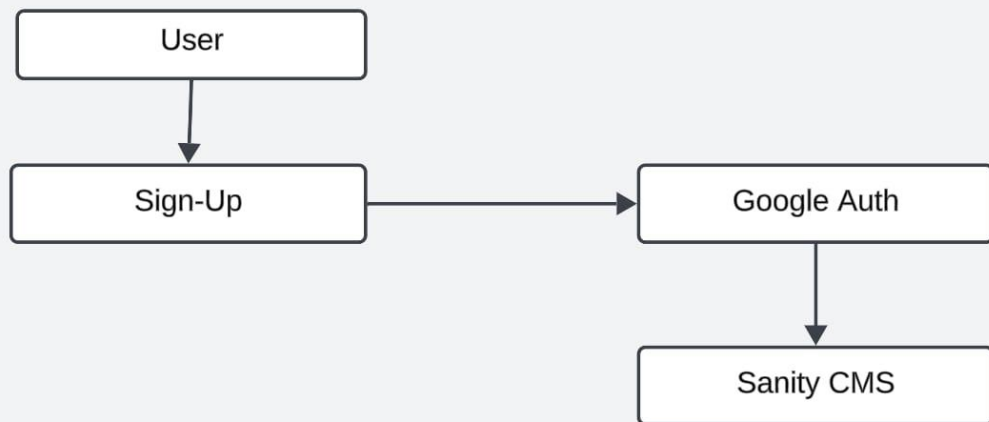
Product Browsing and Order Placement

```
                    ┌──────────────────┐
                    │       User       │
                    └──────────────────┘
                             │
                             ▼
                    ┌──────────────────┐          ┌────────────────────────┐
                    │Products Marketplace│───────▶│       Sanity CMS       │
                    └──────────────────┘          │ (Fetching Product(s) Data)│
                             │                     └────────────────────────┘
                             ▼
                    ┌──────────────────┐
                    │  Product Details │
                    └──────────────────┘
                             │
                             ▼
                    ┌──────────────────┐          ┌────────────────────────┐
                    │       Cart       │─────────▶│       Sanity CMS       │
                    └──────────────────┘          │(verify if user is register or not)│
                             │                     └────────────────────────┘
                             ▼                                │
┌────────────────────┐  ┌──────────────────┐                 ▼
│     Sanity CMS     │◀─│  Order placement │          ┌────────────────────────┐
│(Storing Order Details)│ └──────────────────┘──────▶ │    Data Verification    │
└────────────────────┘           │                    │  (Address,ContactInfo etc)│
                                 │                     └────────────────────────┘
                                 ▼                                │
                    ┌──────────────────┐          ┌────────────────────────┐
                    │ Order Confirmation│◀────────│  Customer Infomation   │
                    └──────────────────┘          │         Form           │
                                                  └────────────────────────┘
```

Shipment Tracking

```
┌──────────┐
│   User   │
└────┬─────┘
     │
     ▼
┌──────────────┐        ┌─────────────────┐
│ User Profile │───────▶│   Sanity CMS    │
└──────┬───────┘        │   (user data)   │
       │                └────────┬────────┘
       ▼                         │
┌───────────────────┐           │
│ Order History List │◀─────────┘
└─────────┬─────────┘
          │
          ▼
┌────────────────────┐      ┌──────────────────┐
│ View Order Status  │─────▶│ Shipment Tracking │
└─────────┬──────────┘      │    API Call       │
          │                 └─────────┬─────────┘
          ▼                           │
┌────────────────────┐◀──────────────┘
│  Shipment Status   │
└────────────────────┘
```
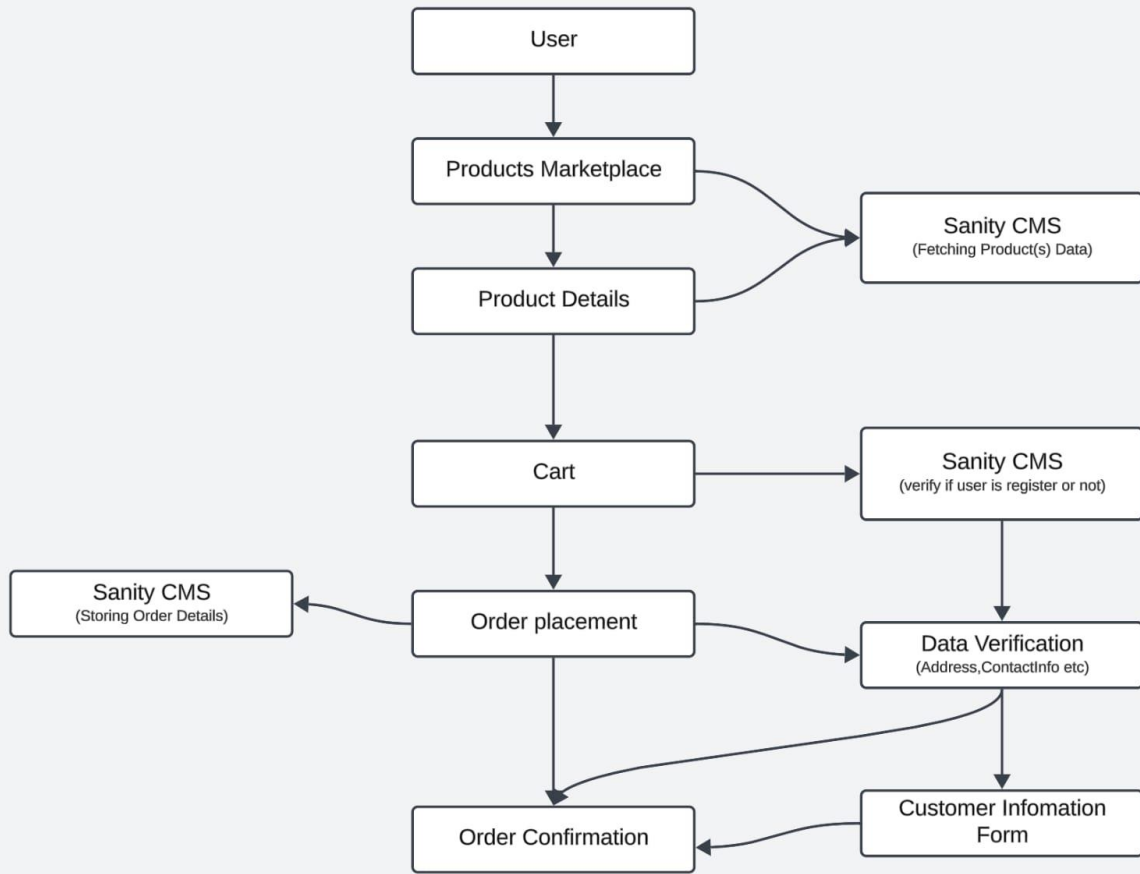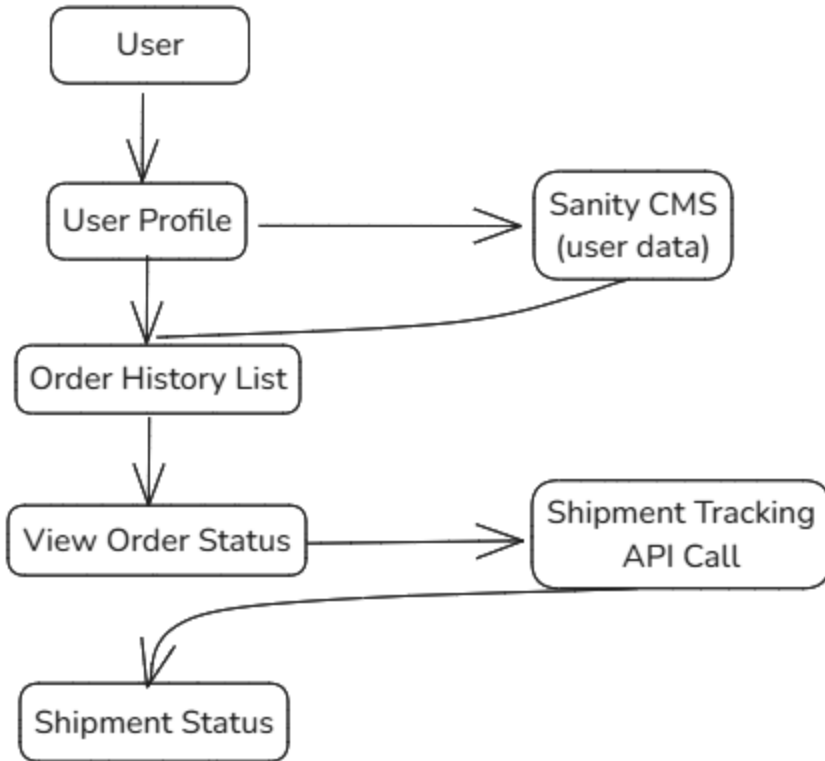
```javascript
export default {
  name: 'product',
  type: 'document',
  title: 'Product',
  fields: [
    {
      name: 'name',
      type: 'string',
      title: 'Product Name',
      description: 'Name of the product.',
      validation: (Rule) => Rule.required().max(100).error('Product name is required
and must be under 100 characters.'),
    },
    {
      name: 'slug',
      type: 'slug',
      title: 'Slug',
      description: 'URL-friendly identifier for the product.',
      options: {
        source: 'name',
        maxLength: 200,
      },
      validation: (Rule) => Rule.required().error('Slug is required for product
identification.'),
    },
    {
      name: 'description',
      type: 'text',
      title: 'Description',
      description: 'Detailed description of the product.',
      validation: (Rule) =>
        Rule.required()
          .min(20)
          .max(500)
          .error('Description must be between 20 and 500 characters.'),
    },
  ],
};
```

```
{
  name: 'price',
  type: 'number',
  title: 'Price',
  description: 'Price of the product.',
  validation: (Rule) =>
    Rule.required()
      .min(0)
      .error('Product price must be a positive value.'),
},
{
  name: 'discountPercentage',
  type: 'number',
  title: 'Discount Percentage',
  description: 'Percentage discount on the product.',
  validation: (Rule) =>
    Rule.min(0)
      .max(100)
      .error('Discount percentage must be between 0 and 100.'),
},
{
  name: 'priceWithoutDiscount',
  type: 'number',
  title: 'Price Without Discount',
  description: 'Original price of the product before discount.',
  readOnly: true,
  initialValue: (doc) => doc.price / (1 - doc.discountPercentage / 100),
},
```