Day 2:

Planning The Technical Foundation :

1. *Frontend Requirements:*

User Interface:

1. Create a user-friendly design for browsing products effortlessly.
2. Implement features for smooth navigation and intuitive user interactions.

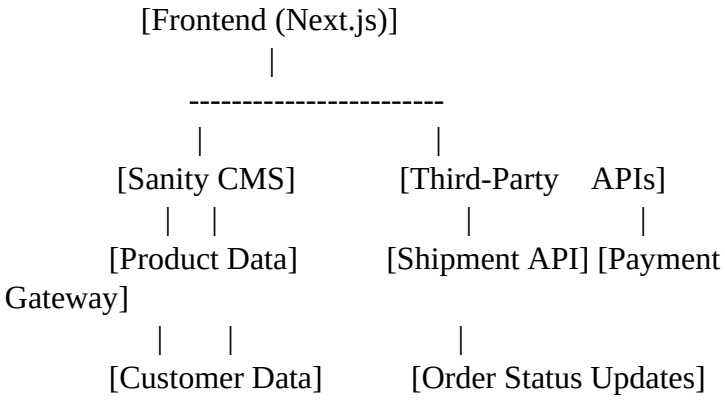
Responsive Design:

1.Ensure the marketplace works seamlessly on both mobile and desktop devices.

Implement Third-Party APIs:

- Shipment Tracking API: Allow customers to track the status of their orders.
- Payment Gateway API (e.g., Stripe or Easypaisa): Ensure secure and smooth payment processing.
- Ensure all APIs integrate seamlessly with both the frontend and backend, providing the necessary data for functionality.

2. *Design System Architecture:*



Key Workflow to include:

1. **User Registration Workflow:**
   - **Flow:**

User signs up → Form data is sent to Sanity CMS → User record is created → Confirmation email/message sent to the user.

- **Purpose:**

Ensures seamless registration and account management for users.

2. **Product Browsing Workflow:**
   - **Flow:**

User navigates the marketplace → Frontend makes an API request to Sanity CMS → Product data (name, description, price, stock, images) is fetched → Data is dynamically displayed on the site.

   - **Purpose:**

Enables users to view product details and make purchasing decisions.

3. **Order Placement Workflow:**
   - **Steps:**

1. User adds products to the cart on the frontend.
2. At checkout, the order details (products, total amount, and user information) are sent to Sanity CMS via an API request.
3. The order is recorded in Sanity CMS with a unique orderId.
4. The user is redirected to the payment gateway for processing the payment.
5. Once payment is successful, a confirmation is sent to the user and updated in Sanity CMS.

   - **Purpose:**

Ensures orders are seamlessly recorded and payment is securely processed.

4. **Shipment Tracking Workflow:**

- **Steps:**

1. The user checks the order status on the frontend.
2. The frontend fetches real-time order tracking details via a Third-Party Shipment API.
3. The shipment details (current location, expected delivery time) are displayed to the user.
4. Updates are logged back in Sanity CMS for record-keeping.

- **Purpose:**

Keeps users informed about their order delivery status.

API Endpoints:

1.Fetch All Products

- **Endpoint Name:** /products
- **Method:** GET
- **Description:** Fetch all available products from Sanity CMS, including perishable food items.
- **Response Example:**

JSON DATA :


```
[

{

"id": 1,
"name": "Chicken Pizza",
"price": 250,
"stock": 200,
"category": "Fast Food",
"image": "pizza.jpg"

},

]
```


2.Create a New Order

- **Endpoint Name:** /orders
- **Method:** POST
- **Description:** Create a new food order in Sanity CMS.
- **Payload Example:**

JSON DATA


```
 {

"customerId": "91314141",
 "customerName": "Anaya",
"address": "Karachi, Pakistan",
 "products": [

{
"productId": 1,
"quantity": 2,
 "price": 250
},

 {
"productId": 2,
 "quantity": 1,
 "price": 180
 }

 ],

 "totalAmount": 680,
"paymentStatus": "Paid"
 }
```


3.Fetch Real-Time Delivery Updates

- **Endpoint Name:** /express-delivery-status
- **Method:** GET
- **Description:** Fetch real-time delivery updates for food items.
- **Response**

JSON DATA


```
{
 "orderId": 12345,
 "status": "Out for Delivery",
 "ETA": "20 mins"
 }
```