



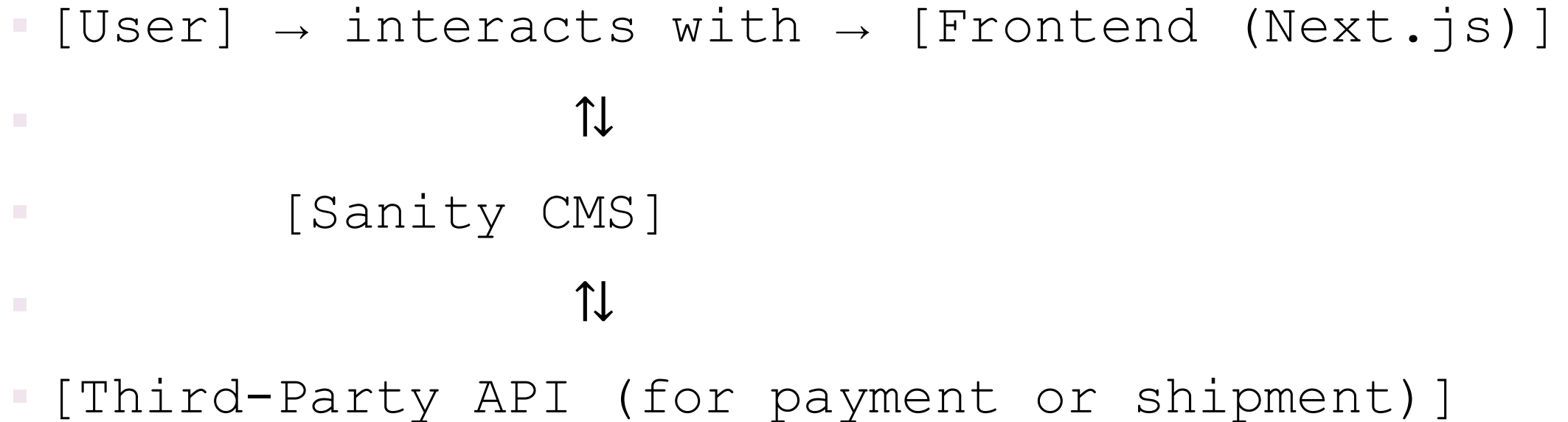
DAY 2 HACKATHONE

**Marketplace Technical Foundation - Furniture
Marketplace**

1. System Architecture

Diagram

Below is the high-level system architecture diagram:



Description:

- Frontend (Next.js):
 - Manages user interactions and displays the furniture marketplace interface.
 - Handles requests to fetch products and submit orders.
- Sanity CMS:
 - Acts as the backend for managing products, orders, and customers.
 - Provides APIs to fetch data (products) and save data (orders).
- Data Flow:
 - **GET /products:** Frontend requests product details from Sanity CMS.
 - **POST /orders:** Frontend sends customer orders to Sanity CMS.
- Third-Party API :
 - Used for payment processing or shipment tracking.

2. API Documentation

GET /products

- **Purpose:** Fetch all available products (Furniture).
 - [
 - { "id": 1, "name": "Sofa",
"price": 500 },
 - { "id": 2, "name": "Chair",
"price": 1500 }
 -]

Example Response:

```
{"message": "Order created", "order": {"id": 12345, "name": "Jane Doe", "product": "Sofa"}}
```

POST /orders

- **Purpose:** Save customer order details.

```
{  
  "name": "Jane Doe",  
  "email":  
  "jane@example.com",  
  "product": "Sofa"  
}
```

3. Sanity Schemas

■ **Product Schema**

```
■ export default {  
■   name: 'product',  
■   type: 'document',  
■   fields: [  
■     { name: 'name', type: 'string', title: 'Product Name' },  
■     { name: 'price', type: 'number', title: 'Price' },  
■     { name: 'image', type: 'image', title: 'Product Image' },  
■     { name: 'description', type: 'text', title: 'Description' }  
■   ]  
■ };
```

■ **Order Schema:**

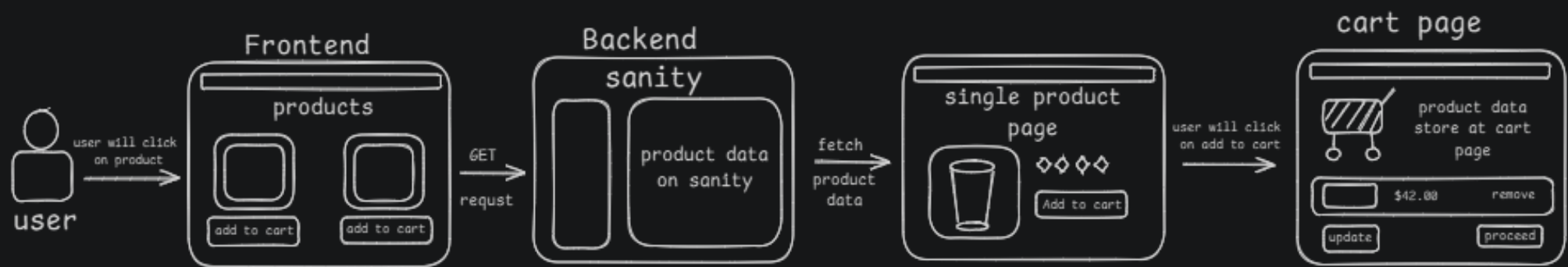
```
■ export default {  
■   name: 'order',  
■   type: 'document',  
■   fields: [  
■     { name: 'name', type: 'string', title: 'Customer Name' },  
■     { name: 'email', type: 'string', title: 'Customer Email' },  
■     { name: 'product', type: 'reference', to: [{ type: 'product'  
■       }], title: 'Product Ordered' },  
■     { name: 'orderDate', type: 'datetime', title: 'Order Date' }  
■   ]  
■ };
```

4. Workflow Description

- **User Browsing Products**
 - 1. User visits the homepage.
 - 2. Frontend sends a **GET** request to /products.
 - 3. Sanity CMS returns a list of products.
 - 4. Frontend displays the products dynamically.

Placing an Order:

- User selects a product and proceeds to checkout.
- Frontend sends a **POST** request to `/orders` with customer and product details.
- Sanity CMS saves the order and returns a confirmation.



Thank You!