# MARKETPLACE TECHNICAL FOUNDATION
# FURNIAURA

## * TECHNICAL PLAN FOR MY MARKETPLACE :-

I am building a general e-commerce marketplace using the following technologies:

- Front-End : Next.JS
- Back-End / Data Management : Sanity CMS
- Cart Functionality : Using Local storage
- Shipping Management : Ship Engine API.

## 1- SYSTEM ARCHITECTURE :-

The marketplace is structured as follows :

### A) FRONT-END :-

- Framework :- Next.JS for building a responsive, user friendly interface.
- Routing :-
  - / :- homepage
  - /shop :- displays all products from Sanity.
  - /shop/[slug] :- display product details dynamically by ID.
  - /cart :- Show items added to cart.
  - /checkout :- checkout page to finalize the order.

### B) BACK-END :-

- Sanity CMS :- for storing product details and order data.

### C) CART MANAGEMENT :-

- Local Storage :- to temporarily store user cart data.

### D) SHIPPING :-

- Ship Engine API :- for calculating shipping rates and generating ID's.

## 2- WORKFLOWS :-

- PRODUCTS WORK FLOW :-
  - a) Products are stored in Sanity with following details :-
    - Name , Price , Description , Quantity , Image , Slug .
  - b) The Next.JS app fetches products from sanity using GROQ queries.
  - c) Users can view product details dynamically via /shop/[slug] .

- CART WORK FLOW :-
  - a) Users add products to their cart from the product page.
  - b) The cart is managed using local storage.
  - c) The cart page (/cart) retrieves data from local storage to display items .

- CHECKOUT WORKFLOW :-
  - a) Users proceeds to checkout from cart page.
  - b) Order details are sent to sanity via an API route.
  - c) Shipping costs are calculated using Ship Engine.

- SHIPPING WORKFLOW :-
  - a) Shipping detail (weight , destination e.t.c) are sent to ship engine to calculate rates.
  - b) The API returns a shipping cost and tracking ID.
  - c) The user recieves their tracking ID on the order confirmation page.

## 3- API's AND ENDPOINTS :-

### a) FETCH PRODUCTS :-

- Description :- Get products from Sanity CMS.
- End point :- /api/shop
- Method :- GET
- Response Example :-

```
[{ "id" : "1",
   "name" : "Product A" ,
   "price" : 100 ,
   " stock" : 20 ,
   "image" : "url-to-image" }]
```

b) ADD TO CART (LOCAL STORAGE) :-

- Description :- Manage cart functionality on client side.
- Methods :-

  For adding :-

  ```
  localStorage.setItem ('cart', JSON.stringify (cartItem));
  ```

  Get Items :-

  ```
  const cartItems = JSON.parse (localStorage.getItem('cart'));
  ```

c) ORDER CREATION :-

- Description :- Save the order to sanity.
- End point :- /api/orders.
- Method :- POST.
- Payload Example :-

```
{
  "orderId" : "12345",
  "customerName" : "Ali",
  "products" : [
    { "id" : "1", "name" : "Product A" , "price" : 100 , ..
  ],
  "total" : 200 ,
  "shippingCost" : 20
}
```

d) SHIPPING API :-

- Provider :- Ship Engine
- End Point :- /rates
- Method :- POST
- Payload Example :-

```
{
  "carrier-code" : "usps",
  "from-country-code" : "US",
  "to-country-code" : "US",
  "to-postal-code" : "90001",
  "weight" : { "value" : 2 , "unit" : "pound" }
}
```

- Response Example :- { "rate" : 15.0 ,
  "tracking-id" : "SE12345" }

## 4- SANITY SCHEMA :-

- Product Schema :-

```
import { defineType } from 'sanity';

export default defineType ({
 name: 'product',
 type: 'document',
 title: 'Product',
 fields: [
   { name : 'name', type: 'string', title : 'Name' },
     ............ ] });
```

- Order Schema :-

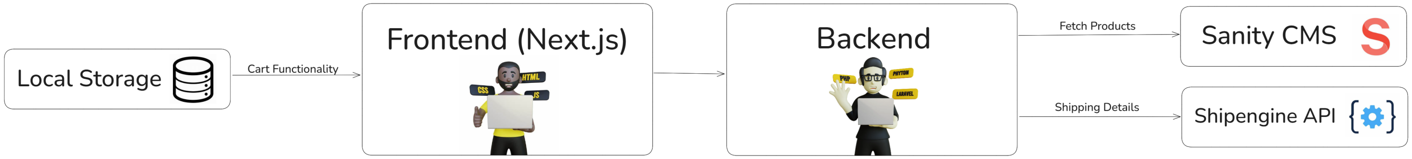
```
import { defineType } from 'sanity';

export default definType ({
 name : 'order',
 type : 'document',
 title : 'Order',
 fields : [
  { name: 'orderID', type: 'string', title : 'Order ID' },
    ......... ] });
```

## 5) CATEGORY SPECIFIC INSTRUCTIONS :-

- Marketplace Type :- General E-Commerce.
- Business Goals :-
  - Build a user friendly platform for online shopping.
  - Simplify cart management and ensure smooth checkout.
  - Provide accurate shipping information via integration with Ship Engine.
  - Manage product data efficiently using sanity CMS.

# "Marketplace Technical Foundation - FurniAura"



Muhammad Nofil Shoaib