Marketplace Technical Foundation

Bandage

(e-commerce website)

- ## System Architecture Frontend:

  (UI) <--> Sanity CMS <--> Third-Party APIs

Description of Components:

- **Frontend (UI)**

  Using NextJs For designing My frontend

  This is the user friendly application where customers interact with the marketplace. The frontend is responsible for fetching product listing , handling user input (add to cart , signUp , LogIn) and display order details and checkout.

- **Sanity CMS**

  The sanity CMS used to manage and deliver product data , inventory , and other dynamic content to the frontend .

  It stores information about products, categories, order details with the help of schema .

- **Third party API:**

  These APIs handle external interaction, such as real time inventory updates, shipping and delivery status .

Data Flow:

- **User Interaction**:
    - The user interacts with the frontend, which fetches data from Sanity CMS and third-party APIs.
- **Data Fetching**:
    - The frontend requests product data and other dynamic content from the Sanity CMS.
    - If applicable (for Q-commerce), it may fetch real-time stock data from third-party APIs.
- **Data Updates**:
    - The frontend may send updates back to the Sanity CMS or third-party APIs, such as adding a product to the cart, updating order status, or completing a purchase.

• System Architecture Frontend:

General E-Commerce Workflows

1. **User Browses Products**:
    - **Frontend**: The user navigates through product categories and views product listings. User also see single product detail dynamically.
    - **Sanity CMS**: The frontend requests product listings from the Sanity.

2. **User Adds Products to Cart**:
    - **Frontend**: User clicks on "Add to Cart" for a specific product.
    - **Sanity CMS**: Cart data is stored temporarily in data base. After include payment method data send to third-party api for shipment. Sanity update the inventory data.

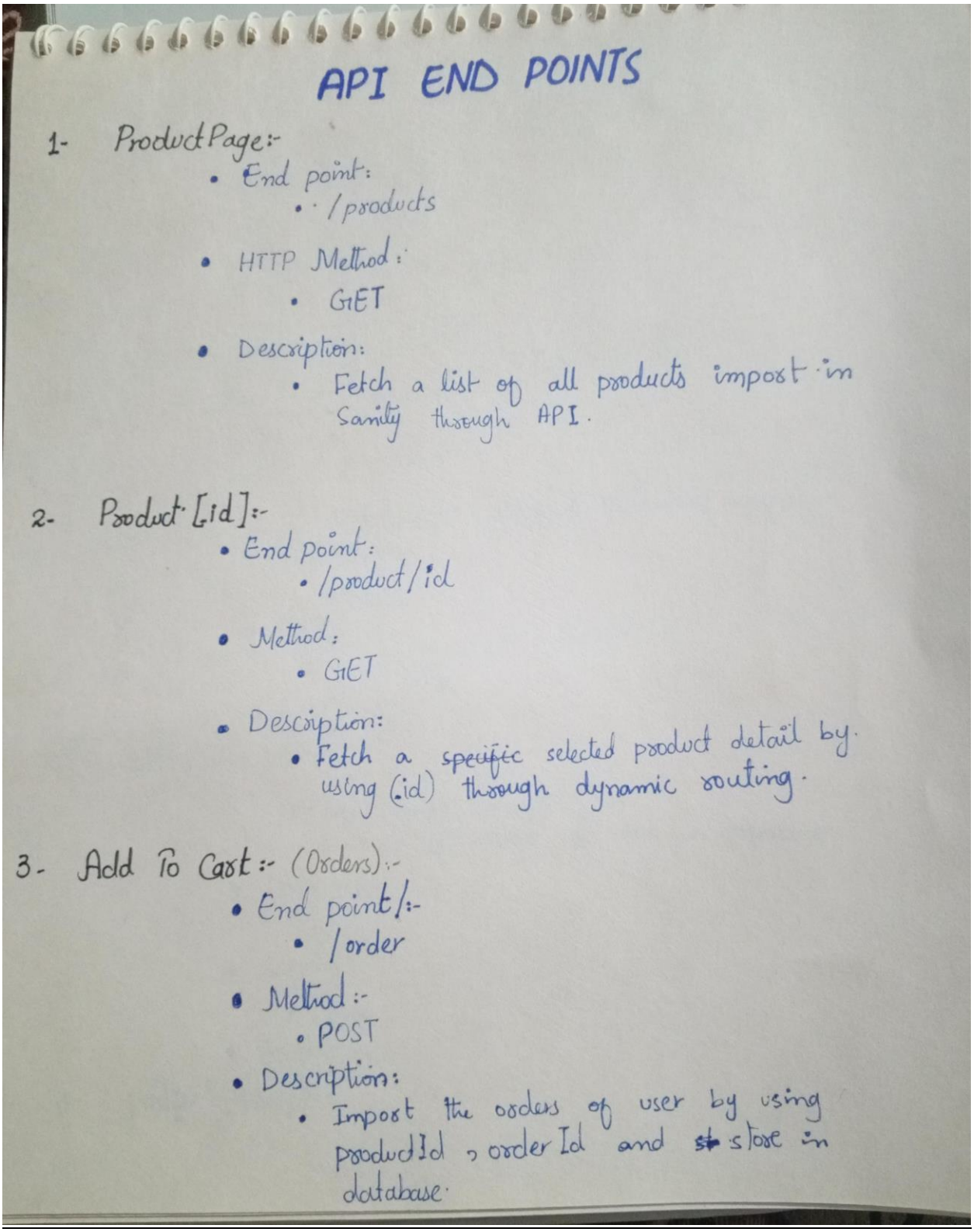
3. **User Completes Checkout**:
    - **Frontend**: User inputs shipping details and payment information.
    - **Sanity CMS**: The frontend may fetch available shipping options from CMS.
    - **Third-Party APIs**: Payment gateway is invoked to process payment (e.g., Stripe, PayPal).

4. **Order Confirmation**:
    - **Frontend**: Displays order confirmation page with order summary.
    - **Sanity CMS**: Updates product stock level based on the order.
    - **Third-Party APIs**: If applicable, shipping API is invoked to track delivery.

• Category-Specific Instruction:

- **Product Browsing**:
    - Use endpoints like `/products` to fetch product listings from the CMS.
    - Display products in categories and allow filtering based on price, brand, etc.
- **Cart Management**:
    - Track user cart via endpoints like `/cart`. The frontend should handle cart operations (add, remove, update quantity).

4. API End Points:



API END POINTS

1- ProductPage:-
- End point:
  - /products
- HTTP Method:
  - GET
- Description:
  - Fetch a list of all products import in Sanity through API.

2- Product [id]:-
- End point:
  - /product/id
- Method:
  - GET
- Description:
  - Fetch a specific selected product detail by using (id) through dynamic routing.

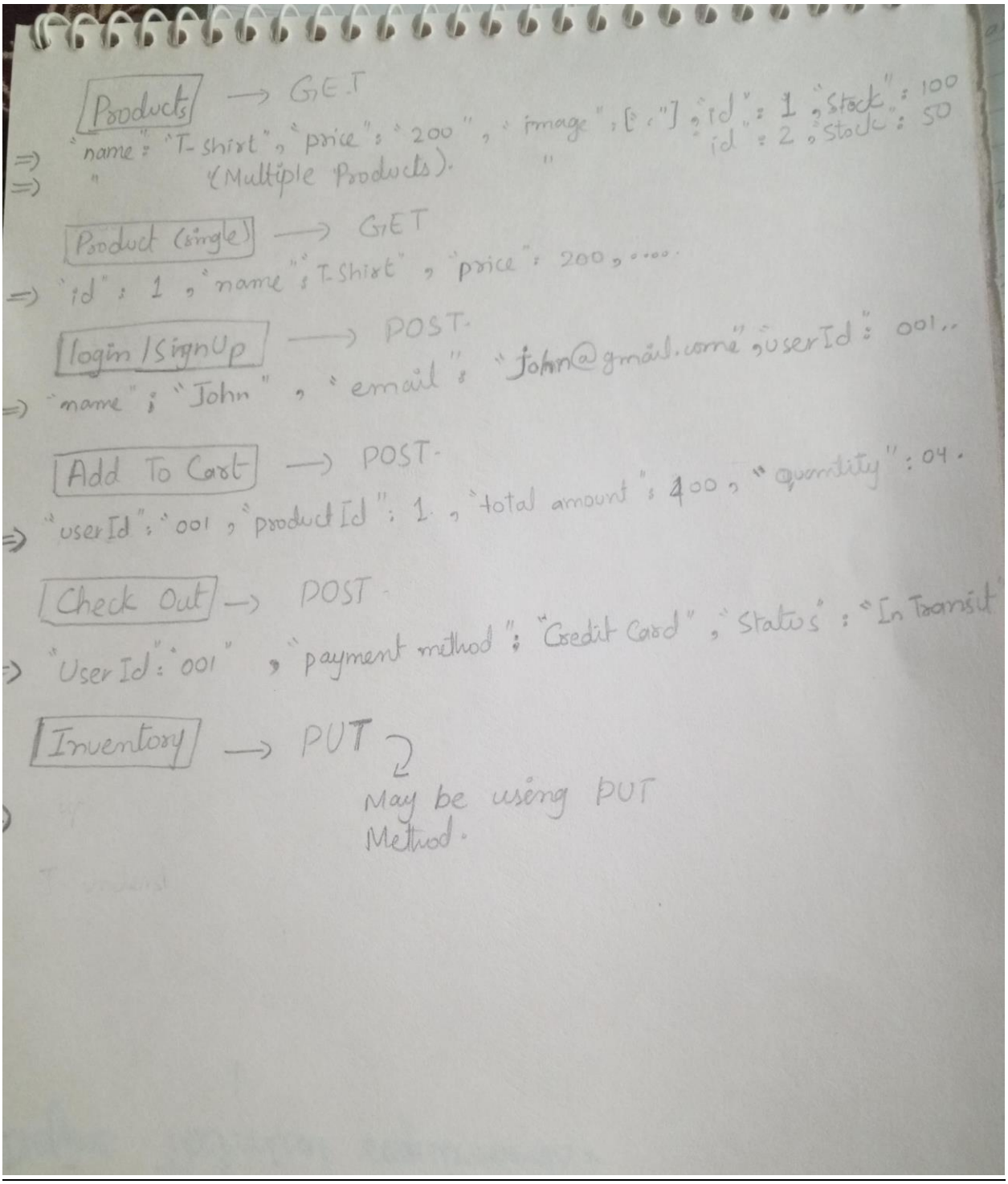
3- Add To Cart:- (Orders):-
- End point:-
  - /order
- Method:-
  - POST
- Description:
  - Import the orders of user by using productId, orderId and store in database.

4- LogIn / Sign Up:

- Endpoint:-
  - /user.
- method:
  - POST
- Description:
  - Impost the data of user in database like (name, email).

5- Check Out Page:

- Endpoint
  - /checkout
- Method:
  - POST
- Description:
  Impost the shipping, payment method..



4.Sanity Schema

- **Shows In image section**