

Technical foundation

Comforty

(General E-commerce Website)

- **Overview:**

Comforty is an innovative online furniture marketplace designed to offer customers a seamless shopping experience with high-quality furniture and customizable options. Leveraging modern technologies such as Next.js for the frontend, Sanity CMS for backend data management, and third-party APIs for payment and shipping, Comforty ensures a robust, scalable, and user-friendly platform.

- **Front-End Requirements:**

1. A user friendly interface for product browsing
2. Responsive design for mobile and desktop users
3. Essential pages:
 - Home Page
 - Products
 - Product Page
 - About
 - Questions
 - Cart
 - Contact Us
 - Privacy Policy

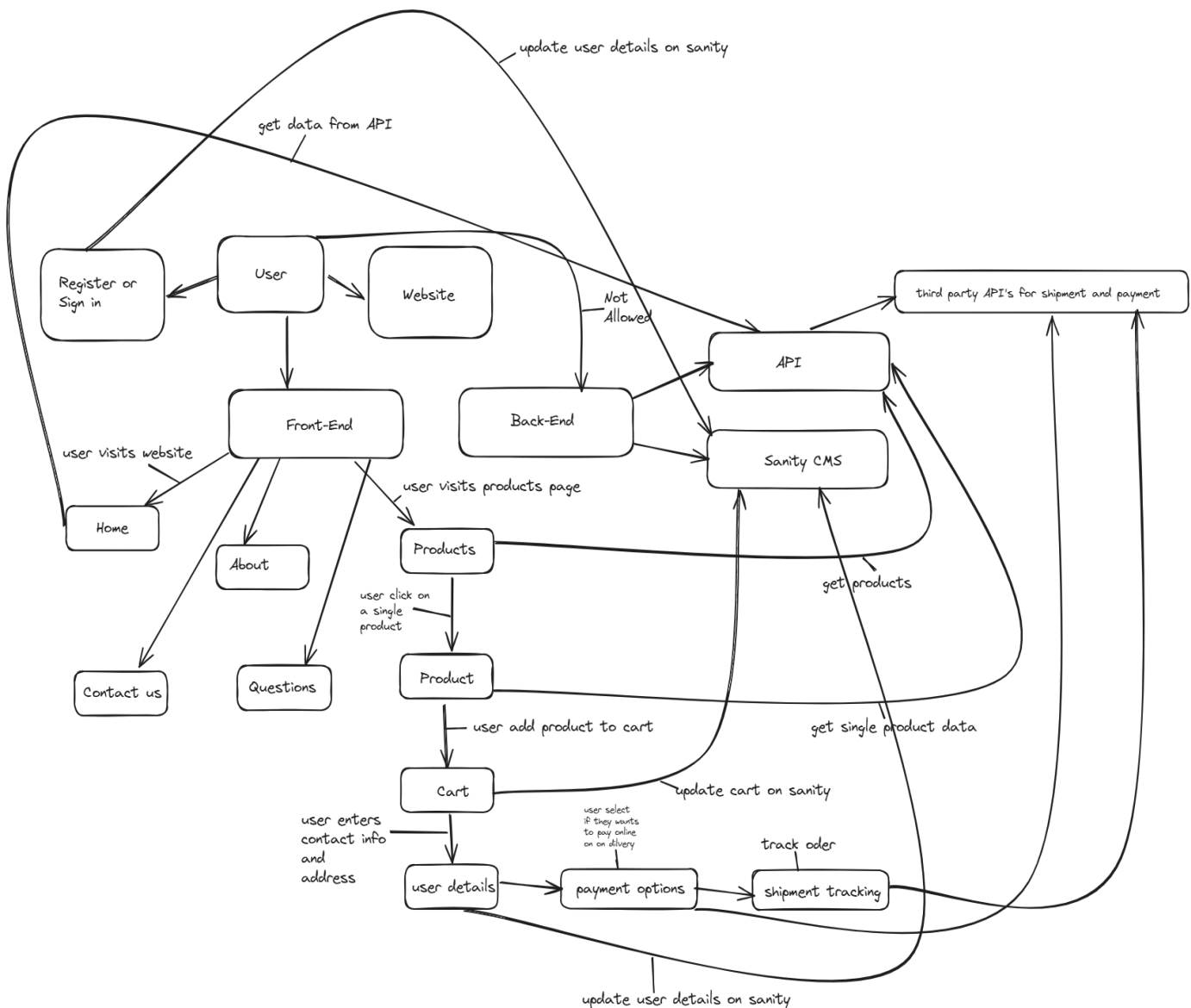
- **Back-End Requirements:**

1. We will use Sanity CMS to manage our product data, Customer details and Order records, shipment details.

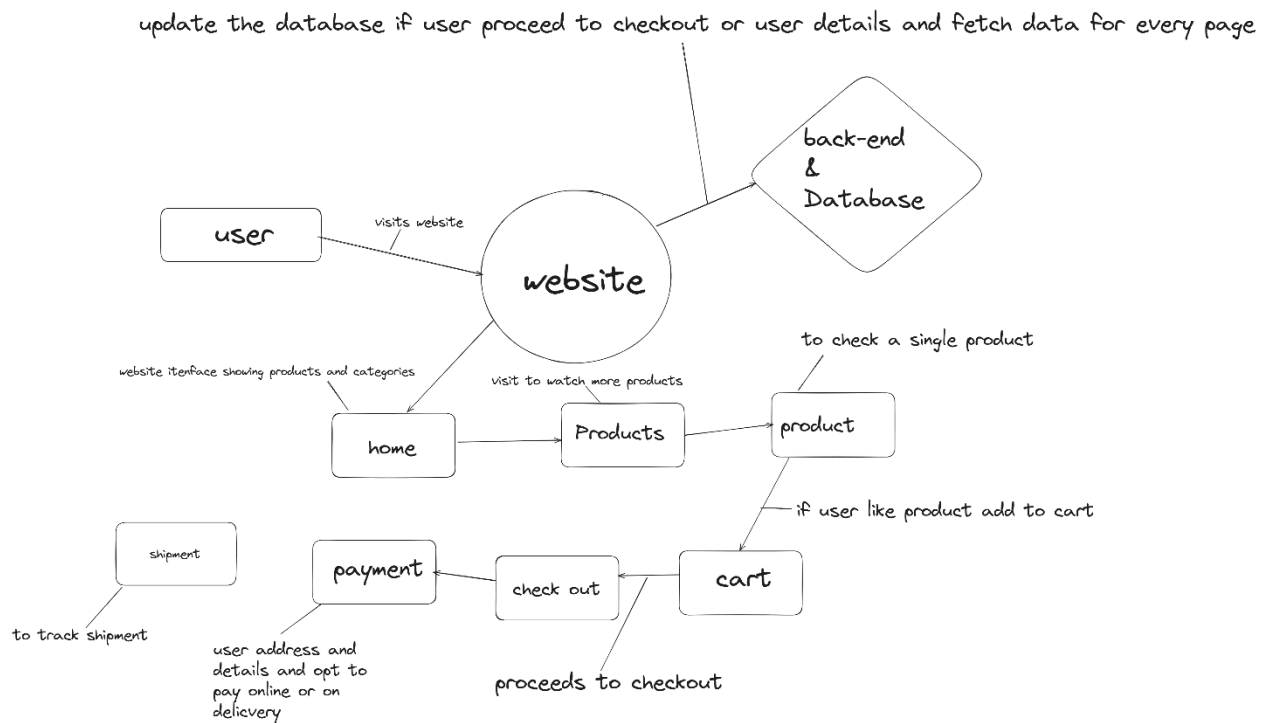
• Third-Party API's:

1. We will use mock API's for our website data.
2. Integrate third party API's for shipment tracking, payment gateway, And other required services.

• System Architecture:



• Workflows:



- **Product Browsing:** the user visits website the using the /products endpoint the data is being displayed on the page.
- **For specific Product:** user likes a specific product. The product id is return to show that specific product price details on the page.
- **Cart:** user likes the product and wants to buy it so user add it to the cart using the endpoint /cart/add.

- **API Endpoints:**

Endpoint	method	purpose	response
/products	GET	Fetch all products	{“id” : 1,“name”:"product a",“price”:160 ,.....}
/products/:id	GET	Fetch details of a product by id	{“id” : 1,“name”:"product a",“price”:160}
/categories	GET	Fetch all categories	[{ "id": 1, "name": "Electronics" }, { "id": 2, "name": "Fashion" }]
/categories/:id	GET	Fetch category of a product by id	{ "id": 1, "name": "Electronics" },
/products	POST	Add new item	Added successfully

- **Sanity Schema:**

```
export const product = {
  name: "product",
  title: "Product",
  type: "document",
  fields: [
    {
      name: "id",
      title: "ID",
      type: "string",
    },
    {
      name: "name",
      title: "Name",
      type: "string",
    },
    {
      name: "description",
      title: "Description",
      type: "string",
    },
    {
      name: "image",
      title: "Image",
      type: "image",
    },
    {
      name: "price",
      title: "Price",
      type: "number",
    },
    {
      name: "stock",
```

```
        title: "Stock",
        type: "number",
    },
    {
        name: "category",
        title: "Category",
        type: "string",
    },
],
};
```