

# GIAIC Karachi, Pakistan

GIAIC  
Karachi, Pakistan  
+92-333-2236799  
azmataliakbar@gmail.com

Day – 3  
Hackathon – 3

19-01-2025

## General E-Commerce Overview

- **Definition:** General e-commerce refers to online platforms that facilitate buying and selling goods and services.
- **Examples:** Amazon, eBay, Shopify, Flipkart, and Daraz.
- **Importance:** Provides a scalable business model, 24/7 availability, and global reach.

### 1. Understanding the API in Next.js Projects

- **Purpose:** APIs (Application Programming Interfaces) enable communication between different software systems. In e-commerce, they fetch product data from external servers.

- **How APIs Work in Next.js:**
  - **Fetch data using `getServerSideProps` or `getStaticProps` for server-side rendering.**
  - **Use `fetch()` or `Axios` to connect to APIs.**
- **Comparison:**
  - **Storing Large Data Locally: Requires significant storage and slows down performance.**
  - **Fetching Data from APIs: Ensures dynamic updates, scalability, and reduced local storage needs.**

## **2. Next.js Installation for API Integration**

### **1. Install Next.js using**

```
npx create-next-app@latest
```

```
my-ecommerce
```

```
cd my-ecommerce ( you can choose your folder name )
```

```
npm run dev
```

### **API Integration:**

- **Use `pages/api` folder to create custom APIs.**
- **Fetch external APIs in components or server-side methods.**

### 3. Sanity Installation and Schema Creation

- **Importance:** Sanity.io is a headless CMS for structured content, ideal for dynamic e-commerce applications.

- **Steps**

**Install Sanity CLI:**

```
npm install -g @sanity/cli sanity init
```

1. ☐ **Configure project and dataset.**

☐ **Schema:**

- **Defines the structure of data in Sanity.**
- **Example schema for products**

```
export default {
```

```
  name: "product",
```

```
  type: "document",
```

```
  title: "Product",
```

```
  fields: [
```

```
    { name: "name", type: "string", title: "Product Name" },
```

```
    { name: "price", type: "number", title: "Price" },
```

```
    { name: "description", type: "text", title: "Description" },
```

```
    { name: "image", type: "image", title: "Product Image" },
```

```
  ],};
```

#### 4. Data Migration via API

- Use APIs to fetch product data and import it into Sanity.
- Benefits: Real-time synchronization, reduced redundancy, and scalability.

#### 5. Folder and File Structure

my-ecommerce/

├── public/ # Static assets

├── src/

| ├── app/ | | ├── layout.tsx |

| ├── page.tsx | | ├── components/ # UI components

| ├── pages/ # API endpoints and pages

| ├── sanity/ # Sanity configurations and schemas

└── package.json # Dependencies

## 6. Testing API and Data Migration

- **Testing APIs:**
  - Use Postman or Thunder Client to test endpoints.
  - Example test:
    - Request: GET /api/products
    - Expected Response: List of products.

### Code Example

```
async function fetchData( ) {  
  
const res = await fetch('/api/products');  
  
return res.json();  
  
}
```

## 7. Fetching Data into Sanity

- Write a script to fetch data and send it to Sanity:

```
import client from './sanityClient';  
  
async function migrateData() {  
  
const data = await fetchExternalData();  
  
data.forEach(item => {  
  
client.create({ _type: 'product', ...item });  
  
});  
  
}
```

## 8. Displaying Data on Localhost

- Use the map function to render product data dynamically:

```
export default function Products({ products }) {
```

```
  return (
```

```
    <div>
```

```
      {products.map(product => (
```

```
        <div key={product._id}>
```

```
          <h2>{product.name}</h2>
```

```
          <p>{product.price}</p>
```

```
        </div>
```

```
      )))
```

```
    </div>
```

```
  );
```

```
}
```

- Map Method: Iterates over an array, transforming and rendering each element.

## 9. Summary

- Steps:
  1. Install and configure Next.js and Sanity.
  2. Create schemas for structured data.
  3. Fetch and migrate data via APIs.
  4. Display dynamic data using React's map method.

- **Suggested Platforms: Amazon, eBay, Shopify, Flipkart, Daraz.**

## **Visual Block Diagram**

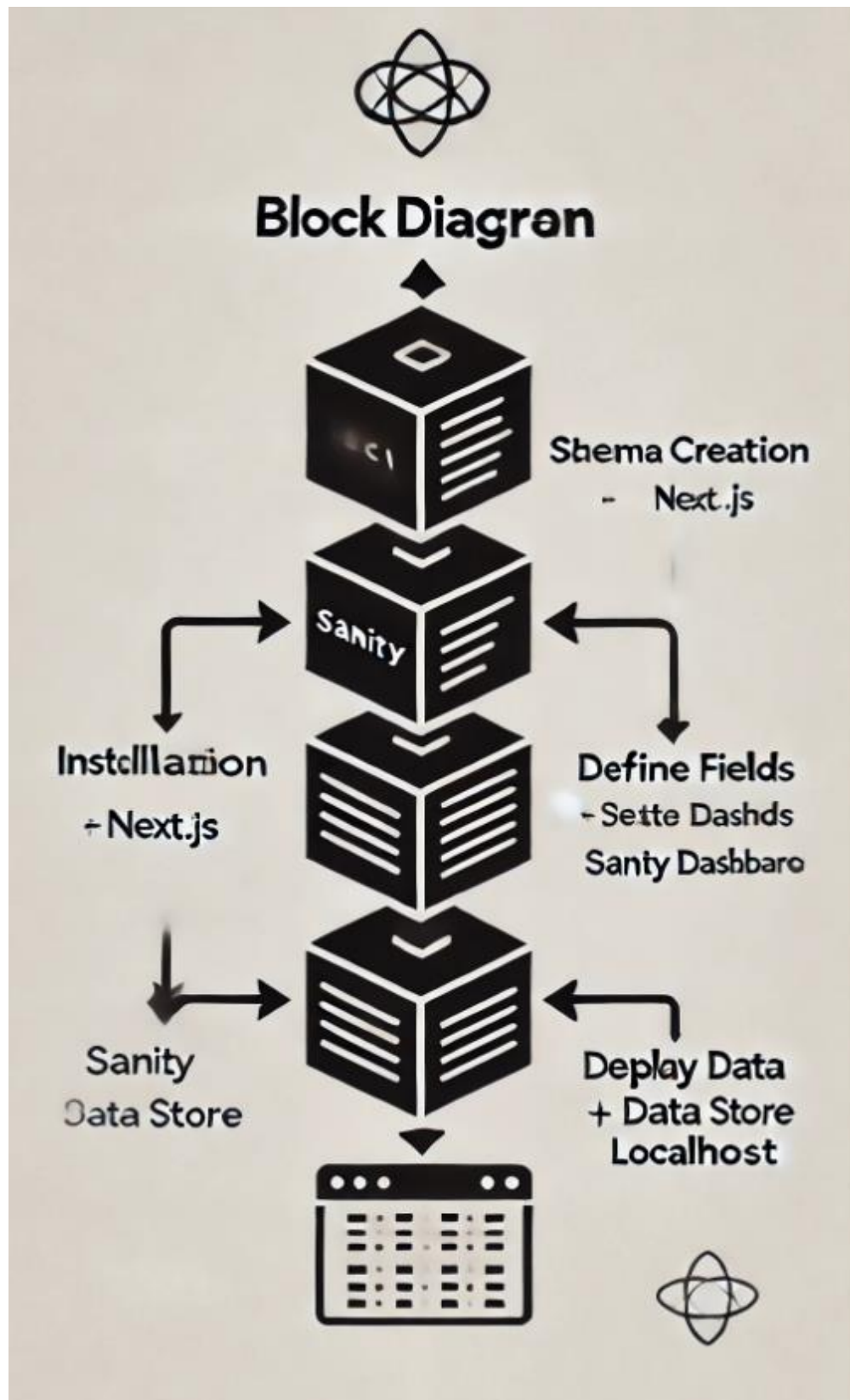
- 1. Installation**  
[Next.js] → [Sanity]  
↓
- 2. Schema Creation**  
[Define Fields] → [Sanity Dashboard]  
↓
- 3. Data Migration**  
[Fetch API] → [Sanity Data Store]  
↓
- 4. Display Data**  
[React Components] → [Localhost]

**Note: Check diagram at next page.**

# Next.JS







Prepared By: Azmat Ali