**Day 3 - API Integration Report**

## **1. Introduction**

**This report covers the API integration and data migration process for the [Your Marketplace Name] project. The main objective was to integrate external APIs to populate the marketplace with dynamic product data, configure CMS schemas, and migrate the data into Sanity CMS.**

## **2. API Integration Process**

### **API Endpoints Used:**

* **GET /products: Fetches a list of products with relevant details such as name, description, price, etc.**
* **GET /categories: Fetches categories of products for display on the site.**

### **Steps Taken:**

1. **API Review and Planning: I reviewed the documentation for each API endpoint and understood the structure of the data, including authentication methods and response formats.**
2. **Making API Calls: I used Axios to make API calls from the Next.js app. The primary API call was for fetching the product data.**
3. **Data Parsing and State Management: The fetched data was stored in React state using useState and displayed dynamically in the product listing page.**
4. **Error Handling: Error handling was implemented for failed API calls to improve user experience.**

## **3. Schema Adjustments**

### **Original Schema vs Adjusted Schema**

**The schema for products in Sanity CMS needed to be adjusted to match the data fetched from the external API.**

### **Key Adjustments:**

* **The product\_name field from the API response was mapped to title in the Sanity schema.**
* **The product\_description was mapped to description.**
* **Additional fields like price, image\_url, and category were added.**

## **4. Migration Steps and Tools Used**

### **Steps Taken for Data Migration:**

1. **Fetch Data from API: Data was fetched from the API using Axios.**
2. **Data Mapping: The response data was mapped to match the fields in the Sanity CMS schema.**
3. **Data Insertion into Sanity CMS: Data was programmatically inserted into Sanity CMS using their client API, with fields correctly mapped to their respective schema.**

## **7. Conclusion**

**The API integration process was completed, allowing dynamic data to fetch and populate Sanity CMS with products. This setup ensures a scalable backend that can dynamically update product listings. The migration process ensured seamless integration with the front end, enabling an interactive marketplace experience.**