Day 3 - API Integration Report

Project : Hekto - E-commerce Marketplace

Submitted By: Muhammad Anas
Slot: Friday 9-12
Roll Number: 55344

Objective

The goal of Day 3 was to integrate APIs and migrate product data into Sanity CMS to build a scalable e-commerce backend for Hekto. The process involved fetching data from the provided API, validating schema compatibility, and displaying data dynamically in the Next.js frontend.

Key Implementations

1. API Integration

- Fetched product listings and categories using **fetch()**
- Successfully integrated the provided API to fetch product listings, categories, and other relevant marketplace data.
- Implemented **SSR** and **SSG** for performance optimization.

2. Data Migration to Sanity CMS

- Mapped API fields to Sanity schema:
  - product_title → name
  - product_price → price
  - product_category → category
- Ensured correct data types and schema validation.

3. Frontend Integration & Display

- Dynamically rendered API data on product listing pages.
- Added error handling and fallback UI for seamless user experience.
- Used Next.js features for better performance.

4. Debugging & Error Handling

- Tested API responses with **Postman** and browser DevTools.
- Implemented try/catch blocks and logging mechanisms.

Best Practices

- **.env Usage:** Secured API keys.
- **Modular Code:** Utility functions for API requests.
- **Data Validation:** Checked and mapped fields before storage.
- **Version Control:** Tracked commits on GitHub.
- **Testing:** Validated API responses before rendering.

Challenges & Solutions

Challenge	Solution
Schema mismatch	Adjusted schema and mapped fields
Slow API response	Implemented caching and lazy loading
API failures	Added error messages and fallback UI

Submission Checklist:

✔️ API integration completed and tested. ✔️ Data migration to Sanity CMS completed. ✔️ Schema adjustments and validation completed. ✔️ Frontend successfully displaying API data. ✔️ Screenshots and documentation prepared for submission.

Conclusion

Successfully integrated API and migrated data into Sanity CMS, ensuring a scalable, dynamic backend for Hekto. The approach enhances compatibility with headless CMS and future expansions.