

DAY 3: API INTEGRATION REPORT – (SHOP.CO)

API Integration Process:

Overview:

On day 3, we worked on connecting APIs to make the frontend, backend, and Sanity CMS communicate smoothly. This allowed us to fetch and update data easily while keeping the system fast and scalable.

Steps:

1. Selected the necessary API endpoints to handle data, such as fetching products, orders, and user details.
2. Set up API routes in the Next.js app to connect with Sanity CMS.
3. Tested and fixed API calls to ensure smooth data flow between different parts of the application.

SCHEMA ADJUSTMENTS:

Adjustments Made:

1. Modified the “Product” schema to include additional fields like “Tags” and “Category” for better filtering.
2. Updated the “Order” schema to store additional information.
3. Ensured all the fields had appropriate validation rules to maintain data consistency.

MIGRATION STEPS AND TOOLS USED:

Migration Steps:

1. Clone the given repository of migration script.
2. Create Sanity project in my website project.
3. Generate token of Sanity project.
4. Replace Token and Project id with sanity project’s token and project id in migration script.
5. Compile TS file and run.
6. Migration completed.

Tools Used:

- Sanity CLI for schema updates.
- Migration script.

SCREENSHOTS:

1. Sanity Project:

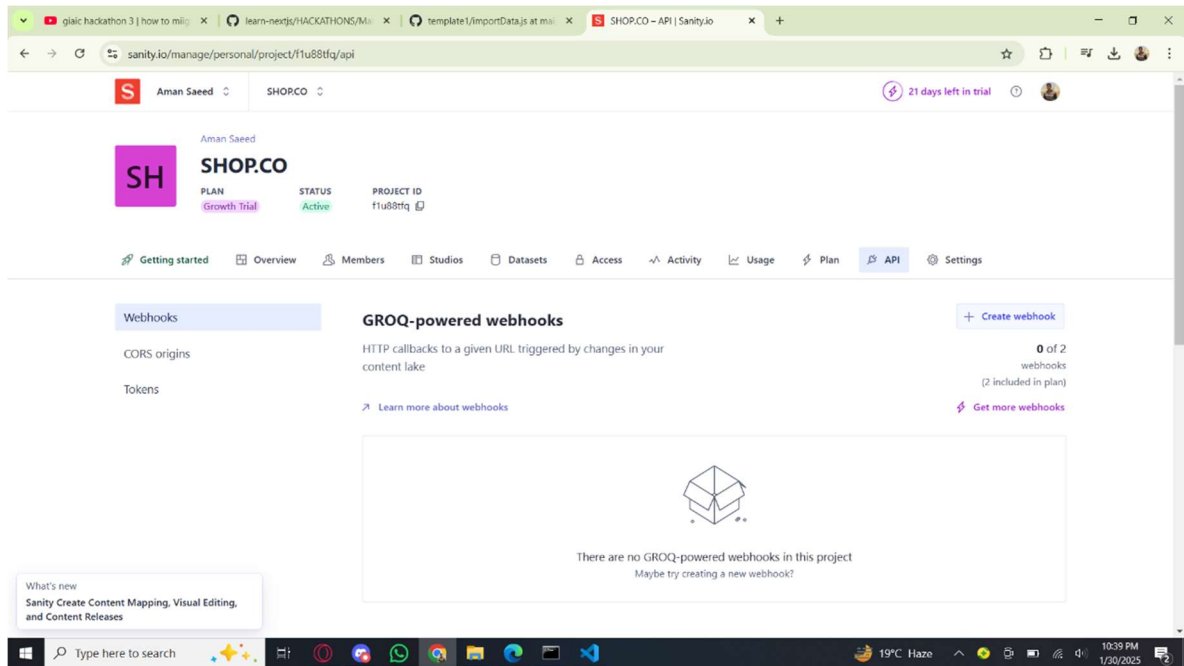


Figure 1: Sanity Project

2. API Token Generation:

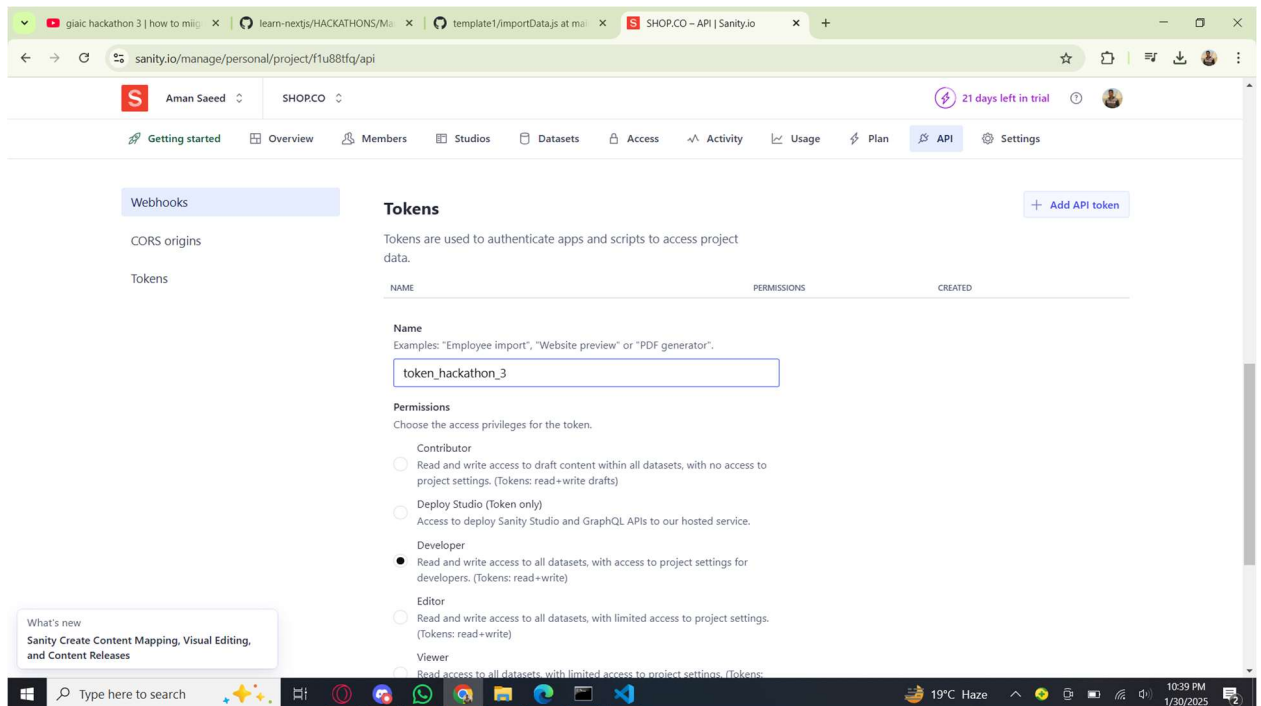


Figure 2: API Token Generation

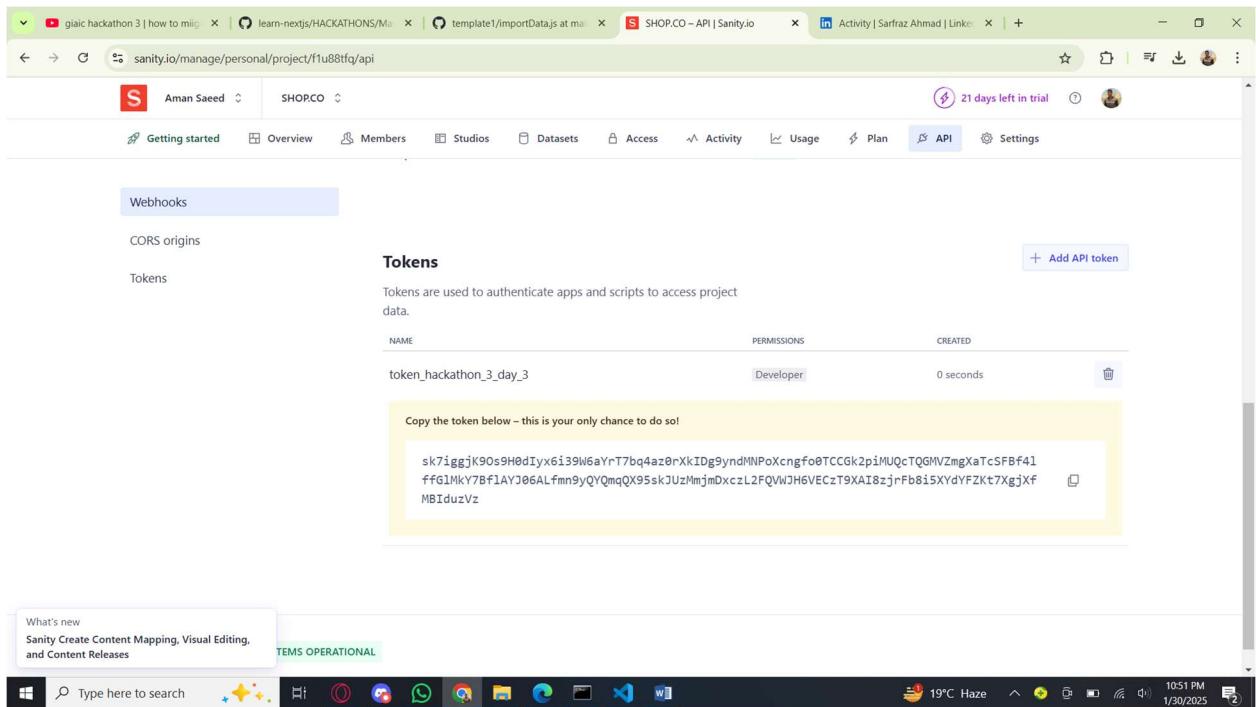


Figure 3: API Token

3. API Calls:

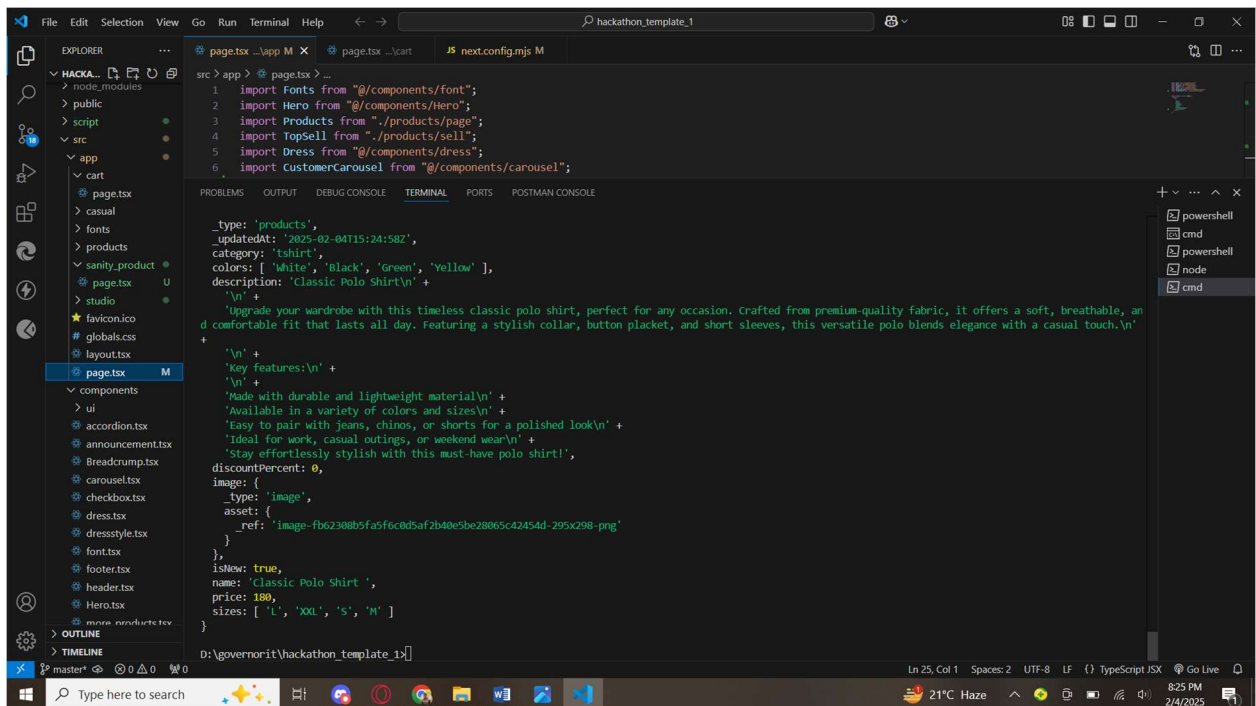
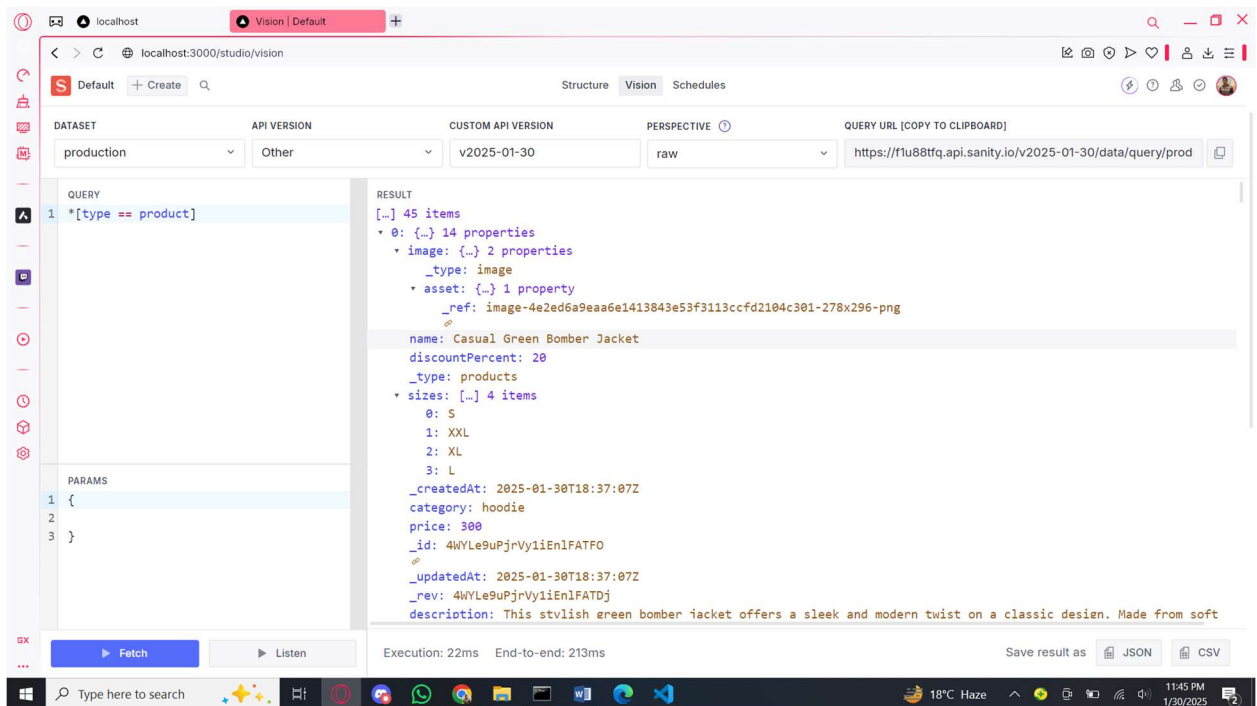
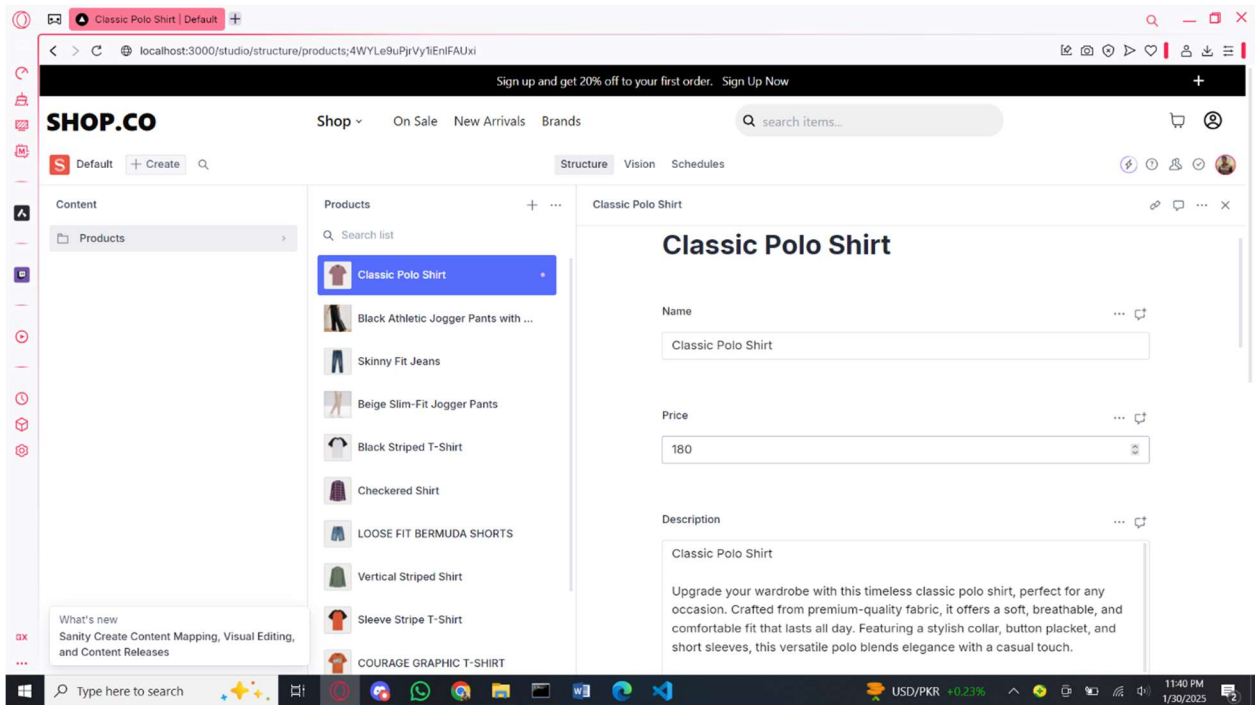
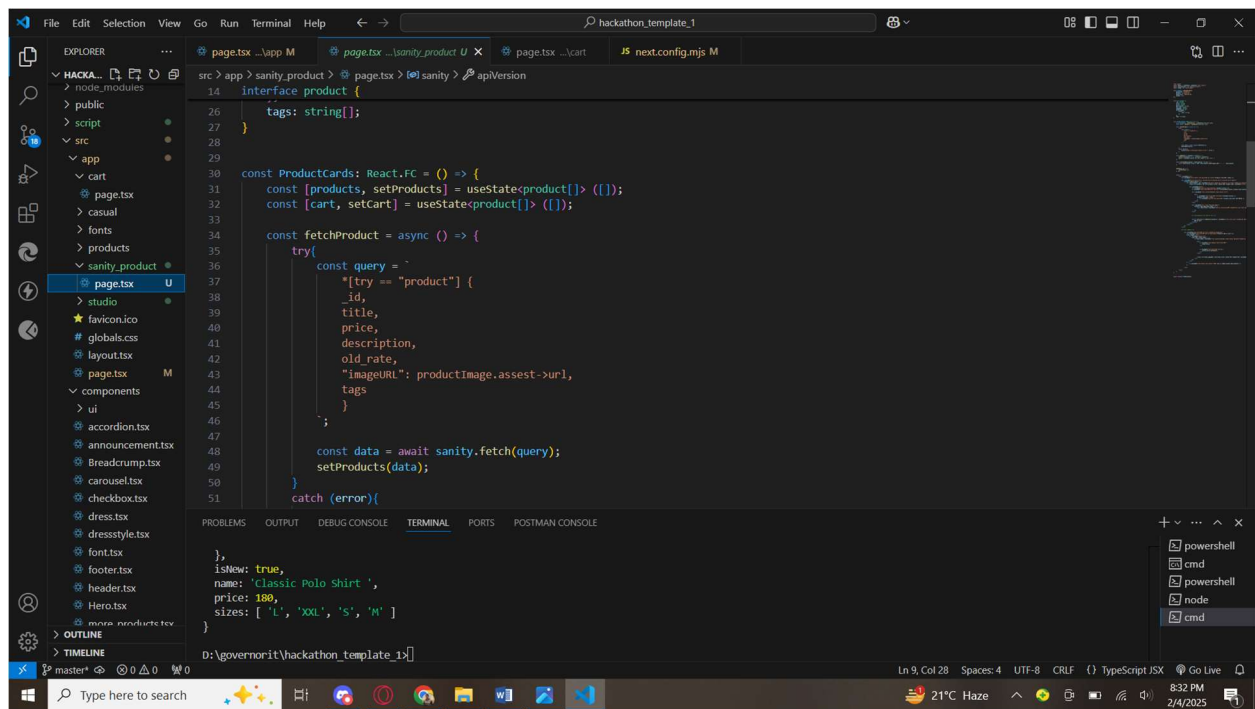


Figure 4: API Calls

4. Populated Fields in Sanity CMS:



5. API Integration:



Conclusion:

Day 3 played a crucial role in achieving smooth API integration and successful data migration. With the foundation now in place, we are well-prepared to enhance functionality and optimize performance in the next stages.