Day 3 - Self-Validation, API Integration, and Data Migration

Overview

On Day 3 of the Marketplace Builder Hackathon, I focused on validating the work completed in Day 1 (business foundation) and Day 2 (technical foundation). This involved reviewing and refining submissions to ensure alignment with the project goals and requirements. Additionally, I integrated APIs and performed data migration tasks to connect the backend and frontend for a functional marketplace.

Self-Validation for Day 1 and Day 2

Day 1: Business Focus Outcome Validation

- **Goals Reviewed:**
    - Clear definition of business objectives and the problem the marketplace solves.
    - Target audience and unique value proposition validation.
- **Market Research:**
    - Analyzed competitor insights for accuracy.
    - Confirmed alignment of market trends with defined goals.
- **Entity Design:**
    - Reviewed entity relationships (e.g., products, orders, customers).
    - Verified clarity and completeness of paper sketches.

Day 2: Technical Foundation Validation

- **System Architecture:**
    - Verified the architecture diagram for clear interaction between the frontend, Sanity CMS, and APIs.
- **Workflows:**
    - Reviewed steps for user registration, product browsing, and order placement.
- **API Requirements:**
    - Confirmed API endpoints (GET, POST, PUT methods) align with marketplace requirements.
- **Sanity Schema:**

    - Validated field definitions and relationships for products, orders, and other entities.

Day 3 Key Tasks

API Integration

1. **API Documentation:**
    - Integrated APIs for:
        - Fetching product data (`GET /api/products`).
        - Managing orders (`POST /api/order`, `GET /api/order/:id`, `PUT /api/order/:id`).
        - Verifying payment statuses (`GET /api/payment/verify`).
    - Ensured authentication through API keys over HTTPS.
2. **Frontend-Backend Communication:**
    - Connected the Next.js frontend with Sanity CMS via API routes.
    - Created utility functions to fetch and render data dynamically.

Data Migration

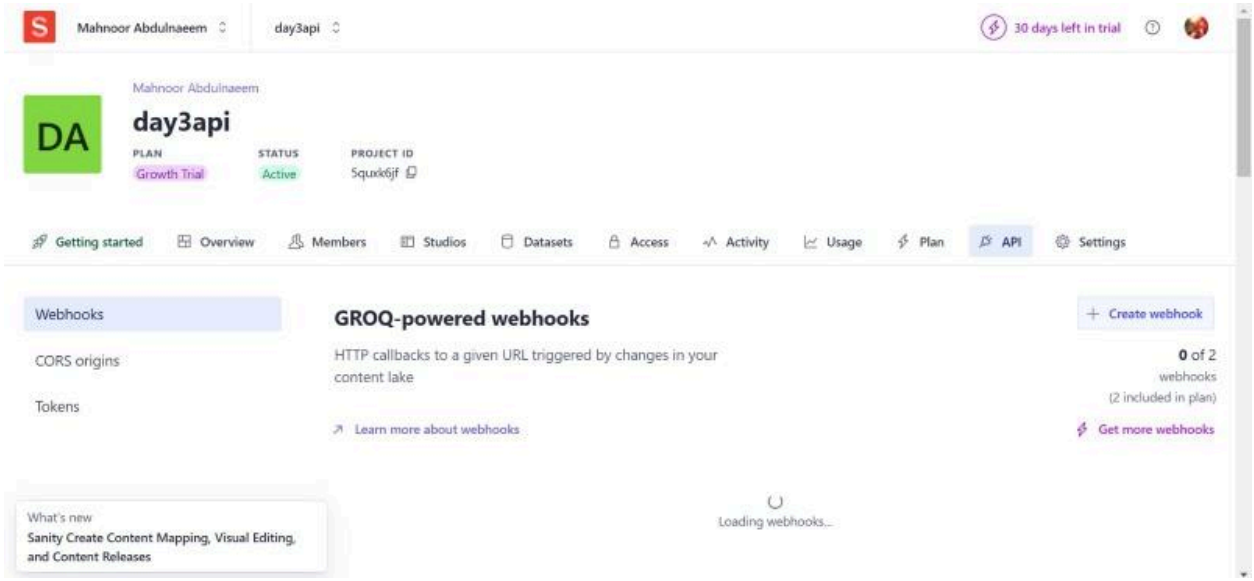
- **Schema Validation:**
    - Adjusted field names in Sanity CMS to align with API response structure.
- **Migration Scripts:**
    - Developed scripts to import product, order, and customer data.
- **Manual Validation:**
    - Reviewed imported data for accuracy and completeness.

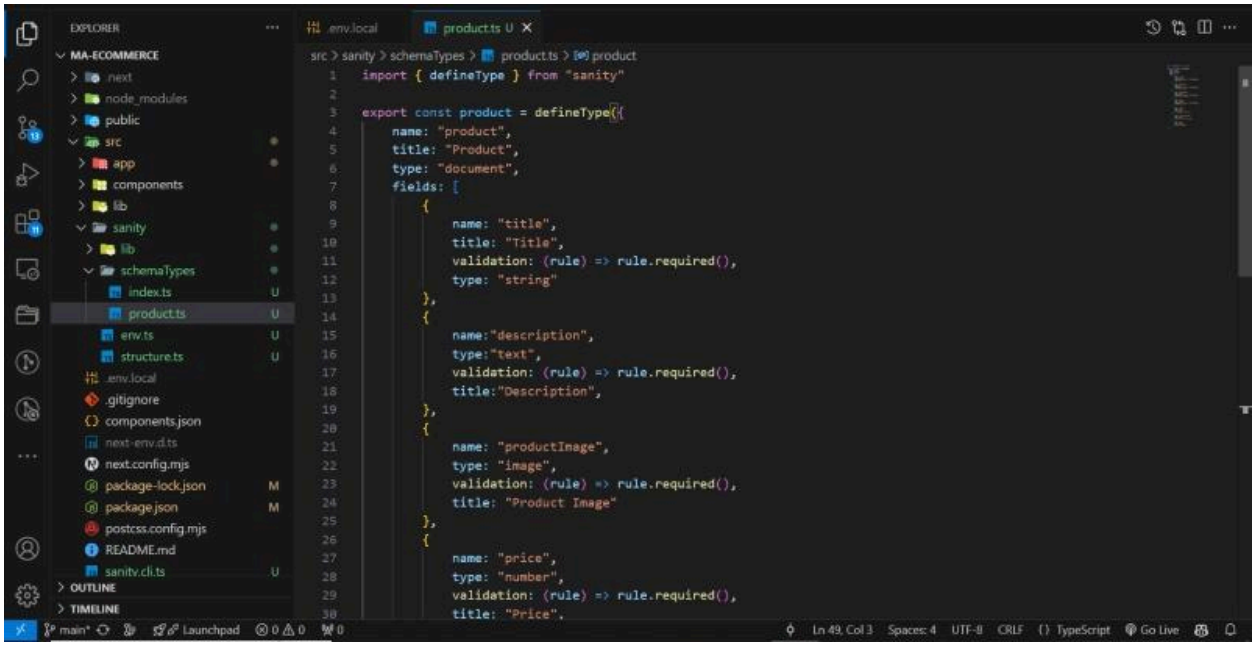
Tech Stack

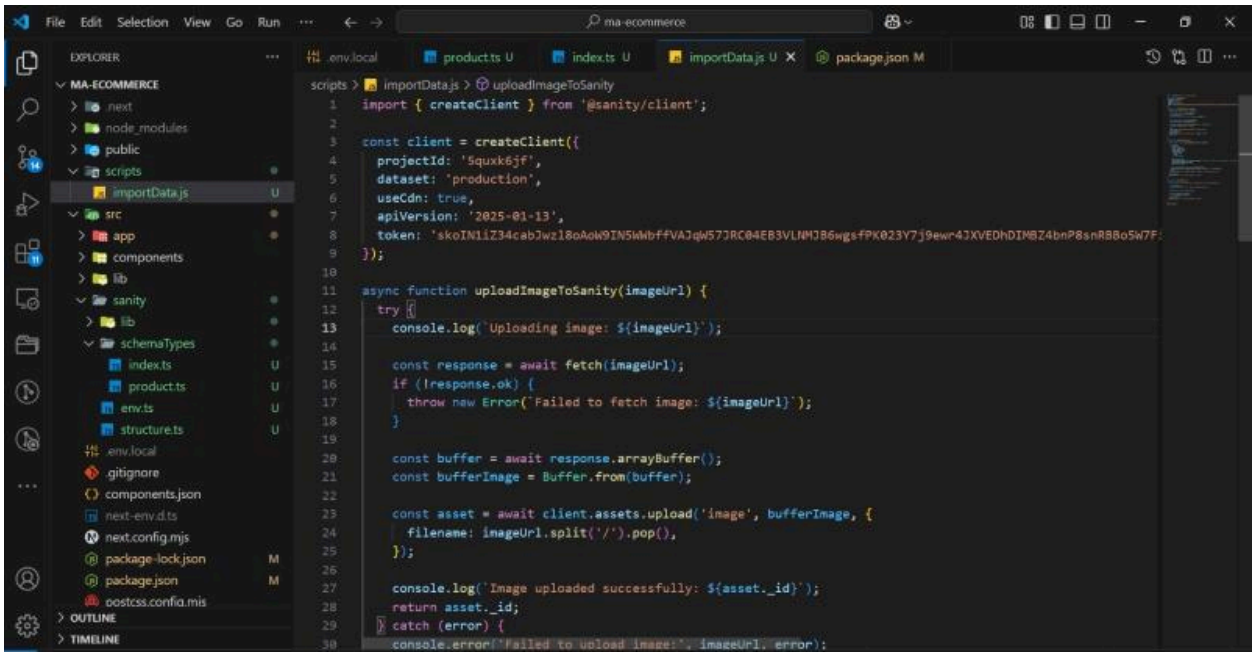
- **Frontend:** Next.js, Tailwind CSS, Zustand
- **Backend:** Sanity CMS
- **API Integration:** Payment and product APIs
- **Authentication:** Clerk for secure login and transactions

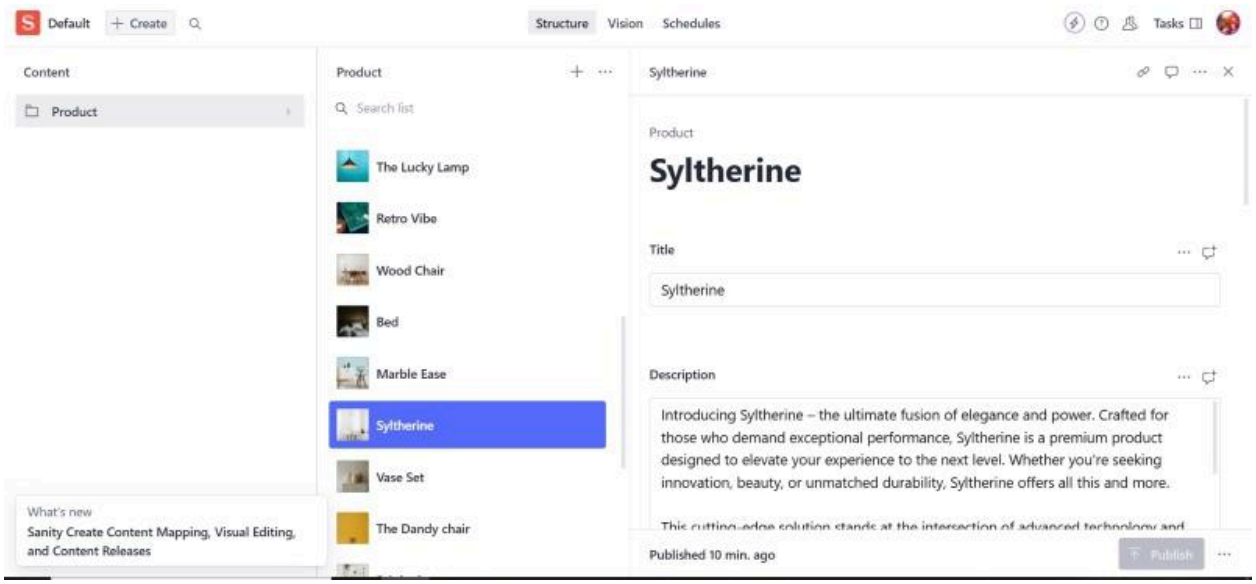
```
Microsoft Windows [Version 10.0.19045.5247]
(c) Microsoft Corporation. All rights reserved.

E:\mahnoor\ma-ecommerce>npm create sanity@latest
Need to install the following packages:
create-sanity@3.70.0
Ok to proceed? (y) y


> ma-ecommerce@0.1.0 npx
> create-sanity

 You are logged in as mahnoorabdulnaeem059@gmail.com using Google
 Fetching existing projects

? Create a new project or select an existing one Create new project
? Your project name: day3api
Your content will be stored in a dataset that can be public or private, depending on
whether you want to query your content with or without authentication.
The default dataset configuration has a public dataset named "production".? Use the default dataset configuration? Yes
 Creating dataset
? Would you like to add configuration files for a Sanity project in this Next.js folder? Yes
? Do you want to use TypeScript? Yes
? Would you like an embedded Sanity Studio? Yes
? What route do you want to use for the Studio? /studio
? Select project template to use Clean project with no predefined schema types
? Would you like to add the project ID and dataset to your .env.local file? Yes
Added http://localhost:3000 to CORS origins
Running 'npm install --legacy-peer-deps --save @sanity/vision@3 sanity@3 @sanity/image-url@1 styled-components@6'
npm warn deprecated @sanity/block-tools@3.70.0: Renamed - use `@portabletext/block-tools` instead. `@sanity/block-tools` will no longer
 receive updates.
```

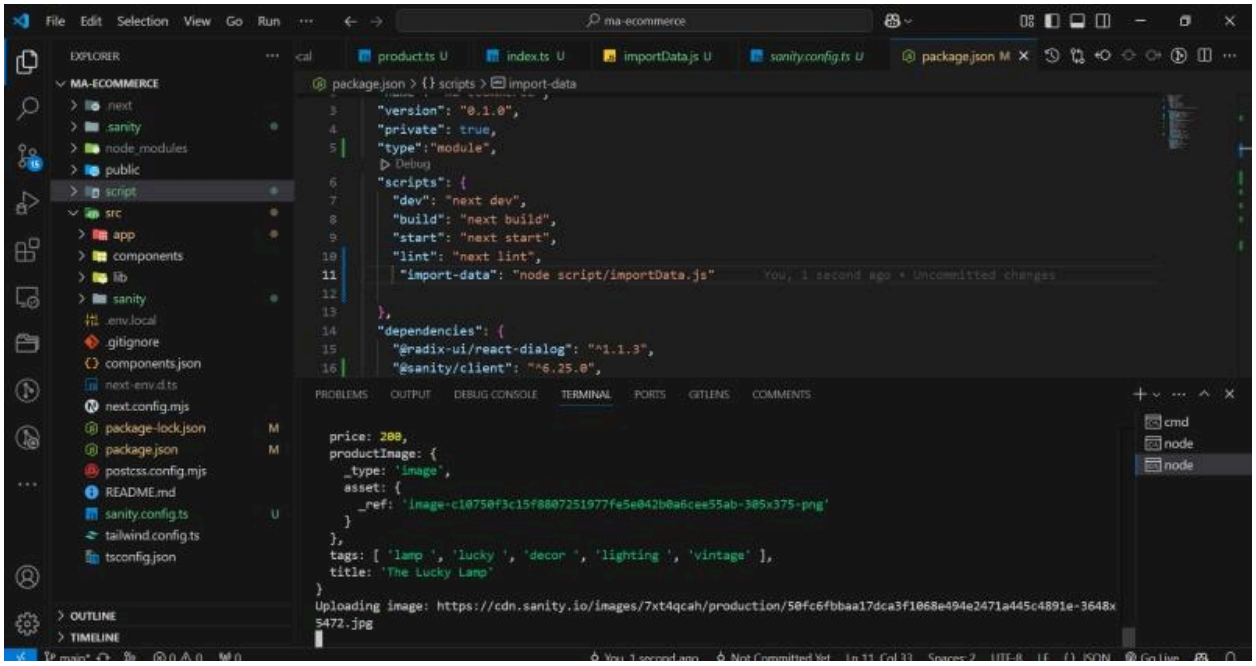
```
 receive updates.

added 927 packages, changed 1 package, and audited 1147 packages in 2m

167 packages are looking for funding
  run `npm fund` for details

1 moderate severity vulnerability

To address all issues, run:
  npm audit fix --force

Run `npm audit` for details.

added 16 packages, and audited 1163 packages in 9s

167 packages are looking for funding
  run `npm fund` for details

1 moderate severity vulnerability

To address all issues, run:
  npm audit fix --force

Run `npm audit` for details.

Success! Your Sanity configuration files has been added to this project

E:\mahnoor\ma-ecommerce>
```

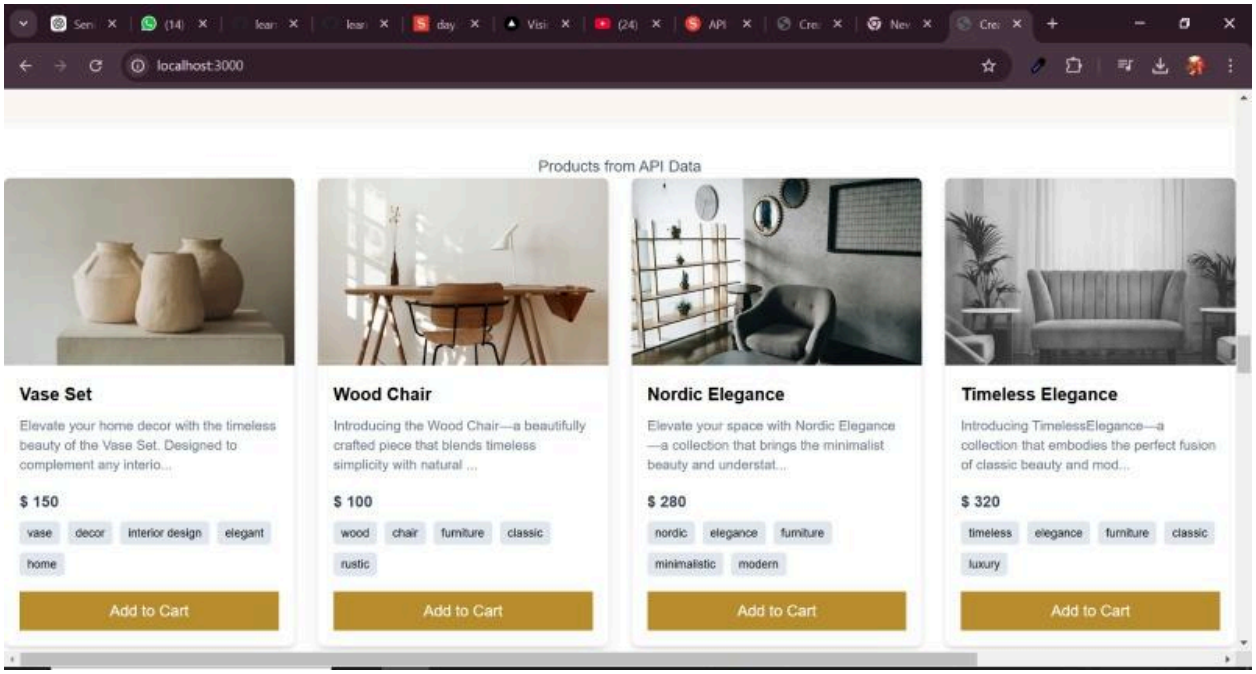


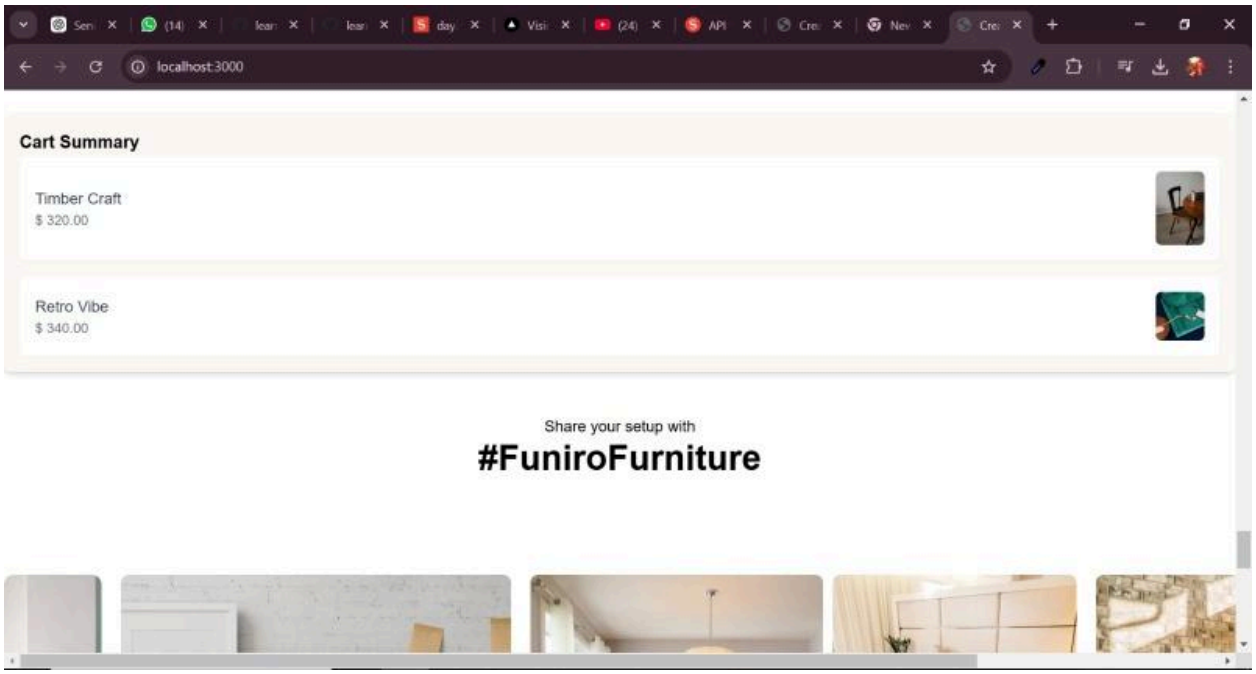

EXPLORER
MA-ECOMMERCE
.next
node_modules
public
src
app
components
lib
sanity
lib
schemaTypes
index.ts
product.ts
env.ts
structure.ts
.env.local
.gitignore
components.json
next-env.d.ts
next.config.mjs
package-lock.json
package.json
postcss.config.mjs
README.md
sanity.cli.ts
OUTLINE
TIMELINE
.env.local
product.ts U
src > sanity > schemaTypes > product.ts > product
import { defineType } from "sanity"
export const product = defineType({
name: "product",
title: "Product",
type: "document",
fields: [
{
name: "title",
title: "Title",
validation: (rule) => rule.required(),
type: "string"
},
{
name:"description",
type:"text",
validation: (rule) => rule.required(),
title:"Description",
},
{
name: "productImage",
type: "image",
validation: (rule) => rule.required(),
title: "Product Image"
},
{
name: "price",
type: "number",
validation: (rule) => rule.required(),
title: "Price",
main*
Launchpad
Ln 49, Col 3
Spaces: 4
UTF-8
CRLF
TypeScript
Go Live


File
Edit
Selection
View
Go
Run
ma-ecommerce
EXPLORER
MA-ECOMMERCE
.next
node_modules
public
scripts
importData.js
src
app
components
lib
sanity
lib
schemaTypes
index.ts
product.ts
env.ts
structure.ts
.env.local
.gitignore
components.json
next-env.d.ts
next.config.mjs
package-lock.json
package.json
postcss.config.mjs
OUTLINE
TIMELINE
.env.local
product.ts U
index.ts U
importData.js U
package.json M
scripts > importData.js > uploadImageToSanity
import { createClient } from '@sanity/client';
const client = createClient({
projectId: '5quxk6jf',
dataset: 'production',
useCdn: true,
apiVersion: '2025-01-13',
token: 'skoIN1iZ34cabJwzl8oAoW9IN5WWbffVAJqW57JRC04EB3VLNMJB6wgsfPK023Y7j9ewr4JXVEDhDIMBZ4bnP8snRBBo5W7F
});
async function uploadImageToSanity(imageUrl) {
try {
console.log(`Uploading image: ${imageUrl}`);
const response = await fetch(imageUrl);
if (!response.ok) {
throw new Error(`Failed to fetch image: ${imageUrl}`);
}
const buffer = await response.arrayBuffer();
const bufferImage = Buffer.from(buffer);
const asset = await client.assets.upload('image', bufferImage, {
filename: imageUrl.split('/').pop(),
});
console.log(`Image uploaded successfully: ${asset._id}`);
return asset._id;
} catch (error) {
console.error('Failed to upload image:', imageUrl, error);


Default
+ Create
Structure
Vision
Schedules
Tasks
Content
Product
Product
Search list
The Lucky Lamp
Retro Vibe
Wood Chair
Bed
Marble Ease
Syltherine
Vase Set
The Dandy chair
What's new
Sanity Create Content Mapping, Visual Editing, and Content Releases
Syltherine
Product
Syltherine
Title
Syltherine
Description
Introducing Syltherine – the ultimate fusion of elegance and power. Crafted for those who demand exceptional performance, Syltherine is a premium product designed to elevate your experience to the next level. Whether you're seeking innovation, beauty, or unmatched durability, Syltherine offers all this and more.
Published 10 min. ago
Publish


File Edit Selection View Go Run
ma-ecommerce
EXPLORER
MA-ECOMMERCE
next
sanity
node_modules
public
script
src
app
components
lib
sanity
.env.local
.gitignore
components.json
next-env.d.ts
next.config.mjs
package-lock.json
package.json
postcss.config.mjs
README.md
sanity.config.ts
tailwind.config.ts
tsconfig.json
OUTLINE
TIMELINE
product.ts U
index.ts U
importData.js U
sanity.config.ts U
package.json M
package.json > {} scripts > import-data
"version": "0.1.0",
"private": true,
"type":"module",
Debug
"scripts": {
"dev": "next dev",
"build": "next build",
"start": "next start",
"lint": "next lint",
"import-data": "node script/importData.js"
You, 1 second ago • Uncommitted changes
},
"dependencies": {
"@radix-ui/react-dialog": "^1.1.3",
"@sanity/client": "^6.25.0",
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
GITLENS
COMMENTS
cmd
node
node
price: 200,
productImage: {
_type: 'image',
asset: {
_ref: 'image-c10750f3c15f8807251977fe5e042b0a6cee55ab-305x375-png'
}
},
tags: ['lamp ', 'lucky ', 'decor ', 'lighting ', 'vintage'],
title: 'The Lucky Lamp'
}
Uploading image: https://cdn.sanity.io/images/7xt4qcah/production/50fc6fbbaa17dca3f1068e494e2471a445c4891e-3648x5472.jpg





Day 3 Summary

On Day 3, I successfully:

1. Validated the work completed in Days 1 and 2 using the checklist.

2. Integrated third-party APIs to handle product data, order management, and payment verification.
3. Migrated data into Sanity CMS and validated schema compatibility.
4. Enhanced frontend-backend communication through API routes.
