DAY 3 - API INTEGRATION

AND DATA MIGRATION

Document Title: "Day 3 - API Integration Report - CommerceCore"
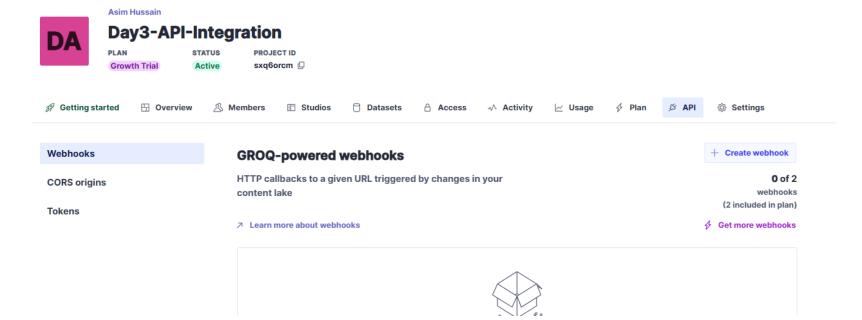
What to Submit:

o **A report documenting:**
- API integration process.
- Adjustments made to schemas.
- Migration steps and tools used.

o **Screenshots of:**
- API calls
- Data successfully displayed in the frontend.
- Populated Sanity CMS fields.

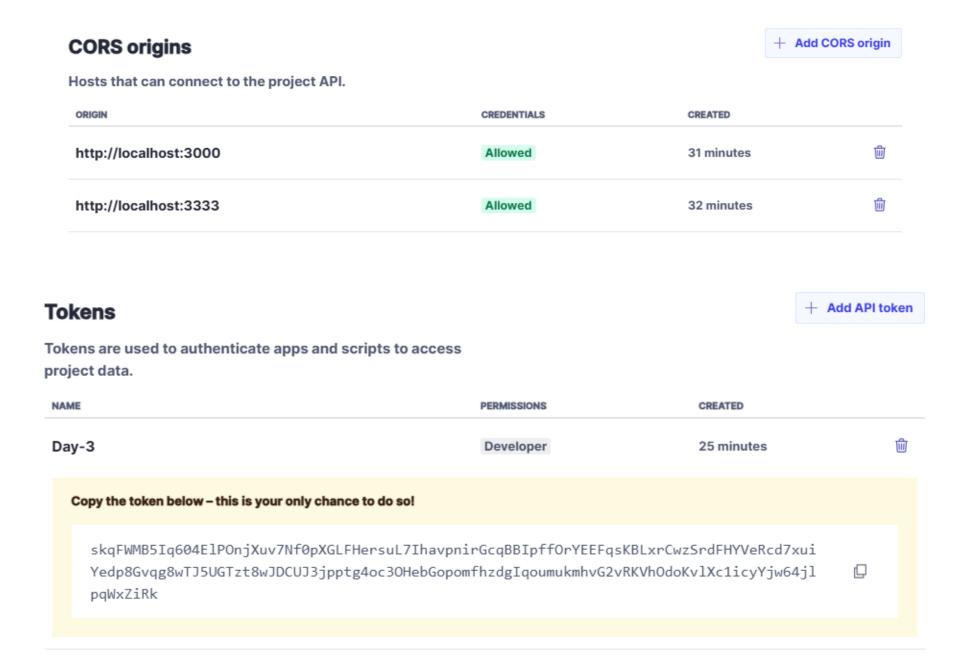o **Code snippets for API integration and migration scripts.**

Day 3 Checklist:

Self-Validation Checklist:

- *API Understanding*   ✓
- *Schema Validation*   ✓
- *Data Migration*   ✓
- *API Integration in Next.js*   ✓
- *Submission Preparation*   ✓

CORS origins

Hosts that can connect to the project API.

+ Add CORS origin

ORIGIN	CREDENTIALS	CREATED	
http://localhost:3000	Allowed	31 minutes	🗑
http://localhost:3333	Allowed	32 minutes	🗑

Tokens

+ Add API token

Tokens are used to authenticate apps and scripts to access project data.

NAME	PERMISSIONS	CREATED	
Day-3	Developer	25 minutes	🗑

Copy the token below – this is your only chance to do so!

skqFWMB5Iq604ElPOnjXuv7Nf0pXGLFHersuL7IhavpnirGcqBBIpffOrYEEFqsKBLxrCwzSrdFHYVeRcd7xui
Yedp8Gvqg8wTJ5UGTzt8wJDCUJ3jpptg4oc3OHebGopomfhzdgIqoumukmhvG2vRKVhOdoKvlXc1icyYjw64jl
pqWxZiRk

EXPLORER   ···

$ .env.local   TS product.ts U ✕   TS index.ts U

∨ E-COMMERCE-HAC...

> .next
> node_modules
> public
∨ src ●
  > app ●
  > components
  > lib
  ∨ sanity ●
    > lib ●
    ∨ schemaTypes ●
      TS index.ts   U
      TS product.ts   U
    TS env.ts   U
    TS structure.ts   U
  $ .env.local
  ◎ .eslintrc.json
  ◆ .gitignore
  {} components.json
  TS next-env.d.ts
  JS next.config.mjs
  {} package-lock.json   M
  {} package.json   M
  JS postcss.config.mjs
  ⓘ README.md
  TS sanity.cli.ts   U

> OUTLINE

src > sanity > schemaTypes > TS product.ts > [∅] productSchema

```ts
1   export const productSchema = {
2       name: 'product',
3       title: 'Product',
4       type: 'document',
5       fields: [
6         {
7           name: 'productName',
8           title: 'Product Name',
9           type: 'string',
10        },
11        {
12          name: 'category',
13          title: 'Category',
14          type: 'string',
15        },
16        {
17          name: 'price',
18          title: 'Price',
19          type: 'number',
20        },
21        {
22          name: 'inventory',
23          title: 'Inventory',
24          type: 'number',
25        },
26        {
27          name: 'colors',
28          title: 'Colors',
29          type: 'array',
30          of: [{ type: 'string' }],
```

```
$ .env.local      TS product.ts U      TS index.ts U      JS data-migration.mjs U      {} package.json M

scripts > JS data-migration.mjs > [∅] client > token
1   import { createClient } from '@sanity/client';
2   import axios from 'axios';
3   import dotenv from 'dotenv';
4   import { fileURLToPath } from 'url';
5   import path from 'path';
6
7   // Load environment variables from .env.local
8   const __filename = fileURLToPath(import.meta.url);
9   const __dirname = path.dirname(__filename);
10  dotenv.config({ path: path.resolve(__dirname, '../.env.local') });
11
12  // Create Sanity client
13  const client = createClient({
14    projectId: "sxq6orcm",
15    dataset: process.env.NEXT_PUBLIC_SANITY_DATASET,
16    useCdn: false,
17    token: "skqFWMB5Iq604ElPOnjXuv7Nf0pXGLFHersuL7IhavpnirGcqBBIpffOrYEEFqsKBLxrCwzSrdFHYVeRcd7
18    apiVersion: '2021-08-31'
19  });
20
21
22  async function uploadImageToSanity(imageUrl) {
23    try {
24      console.log(`Uploading image: ${imageUrl}`);
25      const response = await axios.get(imageUrl, { responseType: 'arraybuffer' });
26      const buffer = Buffer.from(response.data);
27      const asset = await client.assets.upload('image', buffer, {
28        filename: imageUrl.split('/').pop()
29      });
30      console.log(`Image uploaded successfully: ${asset._id}`);
31      return asset._id;
32    } catch (error) {
```

Ln 17, Col 192   Spaces: 2   UTF-8   CRLF

```
{} package.json > {} dependencies
1   {
2     "name": "my-hackathon",
3     "version": "0.1.0",
4     "private": true,
5     "type": "module",
      ▷ Debug
6     "scripts": {
7       "dev": "next dev",
8       "build": "next build",
9       "start": "next start",
10      "lint": "next lint",
11      "import-data": "node script/importSanityData.mjs"
12    },
13    "dependencies": {
14      "@next/font": "^14.2.15",
15      "@radix-ui/react-slot": "^1.1.0",
```

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    PORTS

> node scripts/data-migration.mjs

migrating data please wait...
products ==>>  [
  {
    productName: "Nike Air Force 1 Mid '07",
    category: "Men's Shoes",
    price: 10795,
    inventory: 20,
    colors: [ 'White' ],
    status: 'Just In',
    image: 'https://template-03-api.vercel.app/products/1.png',
```

Ln 11, Col 53   Spa

```
Image uploaded successfully: image-257e162393cc8f3486965fd649496e3ef9e23be-348x348-png
Uploading image: https://template-03-api.vercel.app/products/4.png
Image uploaded su Follow link (ctrl + click)  l1df72a5b65db2f78d3119889a742c134480265-348x348-png
Uploading image: https://template-03-api.vercel.app/products/5.png
Image uploaded successfully: image-46998b7aa7bbf7d305755b2964da5bf167434d40-348x348-png
Uploading image: https://template-03-api.vercel.app/products/6.png
Image uploaded successfully: image-0618b5440becd53aced91c6de2bd3c84757be8d1-348x348-png
Uploading image: https://template-03-api.vercel.app/products/7.png
Image uploaded successfully: image-b039b2d23e832f1d29375236a80f613bc011dbc3-348x348-png
Data migrated successfully!

E:\New folder\E-Commerce-Hackathon>
```

EXPLORER

E-COMMERCE-HACKATHON
- .next
- node_modules
- pTypes
  - productType.ts
- public
- scripts
  - data-migration.mjs
- src
  - app
    - (auth)
    - APIdata
      - page.tsx
    - FindAStore
    - fonts
    - Help
      - page.tsx
    - JoinUs
    - Men
    - NewFeatured
    - studio
    - Women
  - favicon.ico
  - globals.css
  - layout.tsx
  - page.tsx

Tabs: Header.tsx M | page.tsx U | productType.ts U | product.ts U | queries.ts U

src > app > APIdata > page.tsx > [∞] page > ⊙ useEffect() callback > ⊙ acquireProducts

```tsx
1   "use client";
2   import React , { useState , useEffect } from 'react';
3   import Image from "next/image";
4   import { ProductType } from '../../../pTypes/productType';
5   import { client } from '@/sanity/lib/client';
6   import Header from '@/components/Header';
7   import Footer from '@/components/Footer';
8   import { urlFor } from '@/sanity/lib/image';
9   import { AllData } from '@/sanity/lib/queries';
10
11  const page = () => {
12
13
14    const [product, setProduct] = useState<ProductType[]>([]);
15
16    useEffect(() => {
17      async function acquireProducts() {
18        const fetchedProducts:ProductType[] = await client.fetch(AllData)
19        setProduct(fetchedProducts)
20      }
21      acquireProducts();
22
23    },[])
24
25
26
27    return (
28      <div>
29        <Header />
30          <div>
31
32            <div className='max-w-6xl mx-auto px-4 py-8'>
```

Ln 20, Col 6   Spaces: 2   UTF-8   CRLF   {} TypeScript JSX   Go Live   Prettier

---

EXPLORER

E-COMMERCE-HACKATHON
- src
  - components
    - Hero.tsx
    - SpecificFooter.tsx
    - SpecificHeader.tsx
  - lib
  - sanity
    - lib
      - client.ts

Tabs: Header.tsx M | page.tsx U | productType.ts U | product.ts U | queries.ts U

src > sanity > lib > queries.ts > ...

```ts
1   import { groq } from "next-sanity";
2
3   export const AllData = groq`*[_type == "product"]`;
4   export const Four = groq`*[_type == "product"][0..3]`;
```

Default   + Create   Structure Vision Schedules   Tasks

Content
- Product

Product
Search list
- Nike Waffle One SE
- Nike Metcon 8
- Nike Blazer Mid '77 Vintage
- Nike Renew Run 3
- Nike Zoom Fly 5
- Nike Pegasus 40
- Nike Air Max 270
- Nike Court Vision Low
- Nike Dri-FIT UV Hyverse
- Nike Dunk Low Retro SE
- Nike Standard Issue Basketball Jersey
- Air Jordan 1 Elevate Low
- Nike Air Force 1 React
- Nike Air Force 1 PLT.AF.ORM
- Nike Court Vision Low Next Nature

Find a Store   Help   Join Us   Sign In

New & Featured   Men   Women   Kids   Sale   SNKRS   Search

Data From API Sanity

Nike Air Force 1 Mid '07 $10795	Nike Court Vision Low Next Nature $4995	Air Jordan 1 Elevate Low $11895	Nike Dri-FIT UV Hyverse $2495
Nike Air Max 270 $13295	Nike Zoom Fly 5 $11295	Nike Air Force 1 PLT.AF.ORM $8695	Nike Standard Issue Basketball Jersey $2895