



DAY 3-API INTEGRATION REPORT-COMFORTY MARKETPLACE

Prepared by: Laiba Naz

Date: [18-01-25]

1. INTRODUCTION

Ye report hackathon ke Day 3 ke activities explain karti hai jo Team Comforty ne complete ki. Focus tha product data ko Comforty Marketplace mein migrate karna using Sanity CMS. Integration process, schema adjustments aur migration steps neeche detail mein diye gaye hain.

2. API INTEGRATION PROCESS

API integration ka purpose seamless data migration aur interaction ensure karna tha Sanity CMS backend aur marketplace frontend ke darmiyan. Neeche steps diye gaye hain:

SANITY API CONFIGURATION

Migration process start karne ke liye ye steps follow kiye gaye hain:

- Product ID aur API Token ko use kiya gaya Sanity CMS API ke saath connect karne ke liye.
- Sensitive data ki safety ensure karne ke liye credentials environment variables ke taur par store kiye gaye.

SCHEMA ADJUSTMENTS

Sanity CMS schema folder mein do naye files add kiye gaye (sanity/schemas ke andar):

1. ****product.ts**** - Product data store karne ke liye, jaise title, description, price aur image.
2. ****categories.ts**** - Products ko categorize karne aur CMS mein link karne ke liye.

Ye files ensure karte hain ke data structured aur easily queryable ho.

3. DATA MIGRATION PROCESS

Migration process mein ek script banayi gayi jo product data Sanity CMS se local database mein transfer kare. Process kaise execute kiya gaya, neeche diya gaya hai:

Migration Script Banana

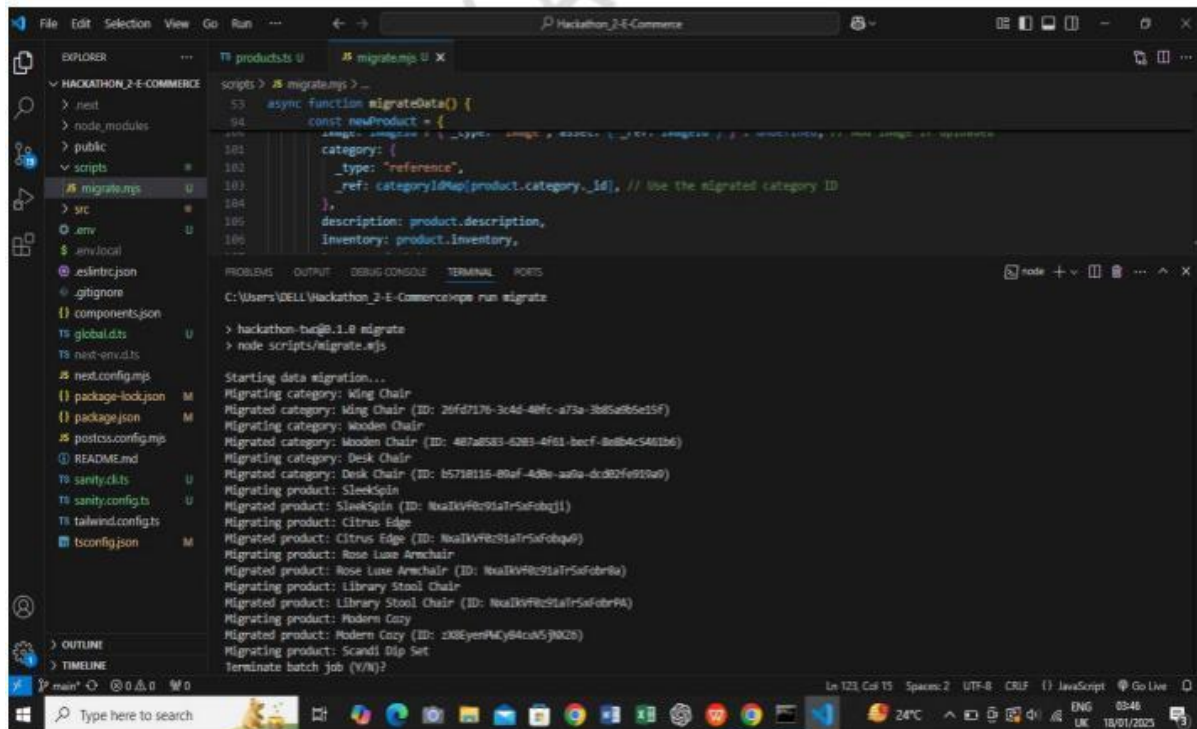
- Root directory mein ek nayi folder `scripts` ke naam se banayi gayi.
- Ek custom script `migrate.mjs` likha gaya jo Sanity CMS se data fetch karke local database mein migrate kare.

Migration Execute Karna

Migration initiate karne ke liye terminal mein ye command run ki gayi:

``npm run migrate``

Is command ne migration process start ki aur data Sanity CMS se local database mein successfully transfer hua.



The screenshot shows a VS Code editor with a file explorer on the left, a code editor in the center, and a terminal at the bottom. The file explorer shows the project structure for 'HACKATHON_2-E-COMMERCE', with the 'scripts' folder expanded. The code editor shows the content of 'migrate.mjs', which is a JavaScript script for migrating data from Sanity CMS to a local database. The terminal shows the command 'npm run migrate' being executed, followed by the output of the script, which lists the migrated categories and products.

```

53  async function migrateData() {
94    const newProduct = {
101      category: {
102        _type: "reference",
103        _ref: categoryIdMap[product.category_id], // Use the migrated category ID
104      },
105      description: product.description,
106      inventory: product.inventory,
  
```

```

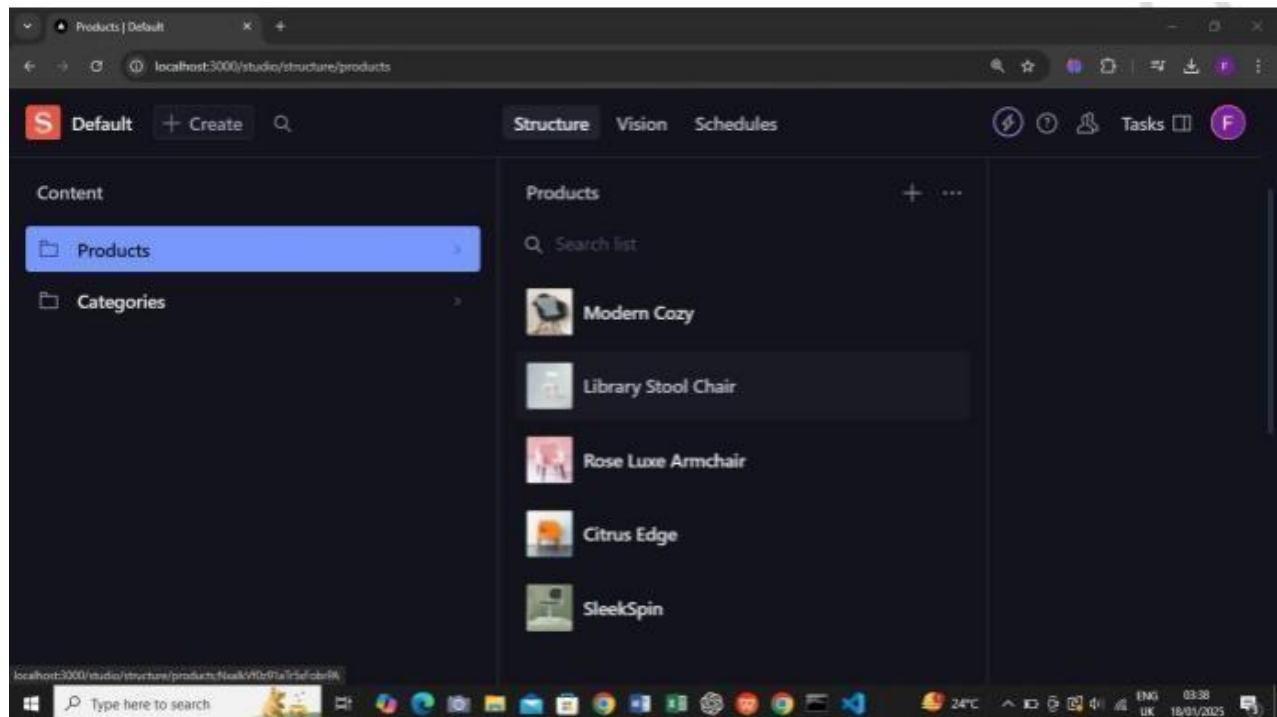
> hackathon-bui@1.0 migrate
> node scripts/migrate.mjs

Starting data migration...
Migrating category: King Chair
Migrated category: King Chair (ID: 26fd7176-3c4d-48fc-a73a-3b85a95e15f)
Migrating category: Wooden Chair
Migrated category: Wooden Chair (ID: 487a8583-6283-4f61-becf-b0b4c5401b6)
Migrating category: Desk Chair
Migrated category: Desk Chair (ID: b5718115-80af-4d8e-a20a-dc3d2fe910a0)
Migrating product: SleekSpin
Migrated product: SleekSpin (ID: NxaIKvF8c91aTr5aFobqj1)
Migrating product: Citrus Edge
Migrated product: Citrus Edge (ID: NxaIKvF8c91aTr5aFobq40)
Migrating product: Rose Line Armchair
Migrated product: Rose Line Armchair (ID: NxaIKvF8c91aTr5aFobq9a)
Migrating product: Library Stool Chair
Migrated product: Library Stool Chair (ID: NxaIKvF8c91aTr5aFobq9A)
Migrating product: Modern Cozy
Migrated product: Modern Cozy (ID: zX8EyenP4Cy84cuX5j902s)
Migrating product: Scandi Dip Set
Migrated product: Scandi Dip Set (ID: NxaIKvF8c91aTr5aFobq9A)
Terminate batch job (Y/N)?
  
```

VERIFICATION

Migration verify karne ke liye ye steps perform kiye gaye:

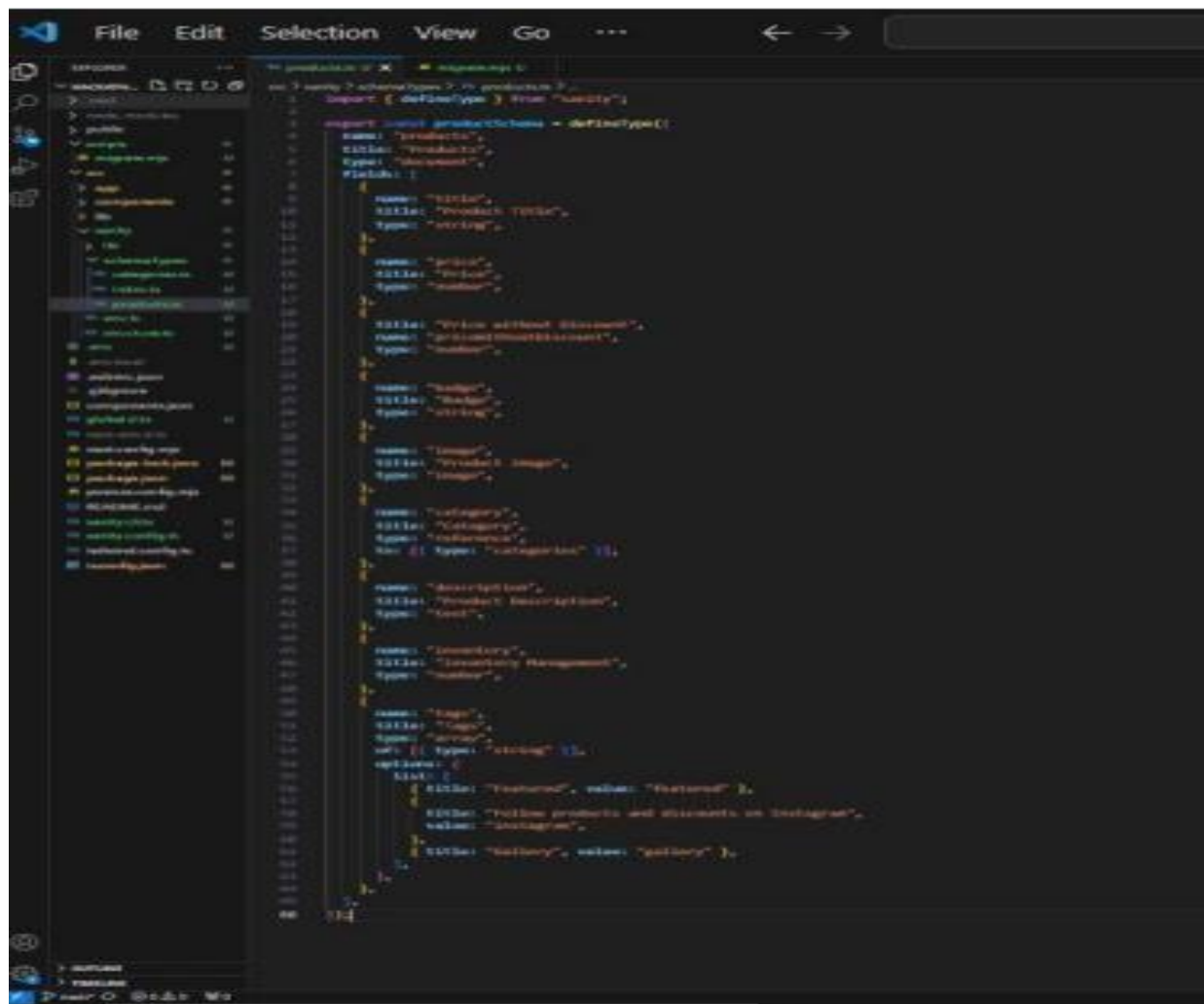
- Project ko locally run kiya gaya.
- `/studio/structure` endpoint localhost par access kiya gaya taake check ho ke data migrate ho gaya hai.
- CMS ke liye use hone wala Sanity account login kiya gaya consistency aur data correctness ensure karne ke liye.



4. SCHEMA ADJUSTMENTS

Data structure do schemas ke zariye organize kiya gaya:

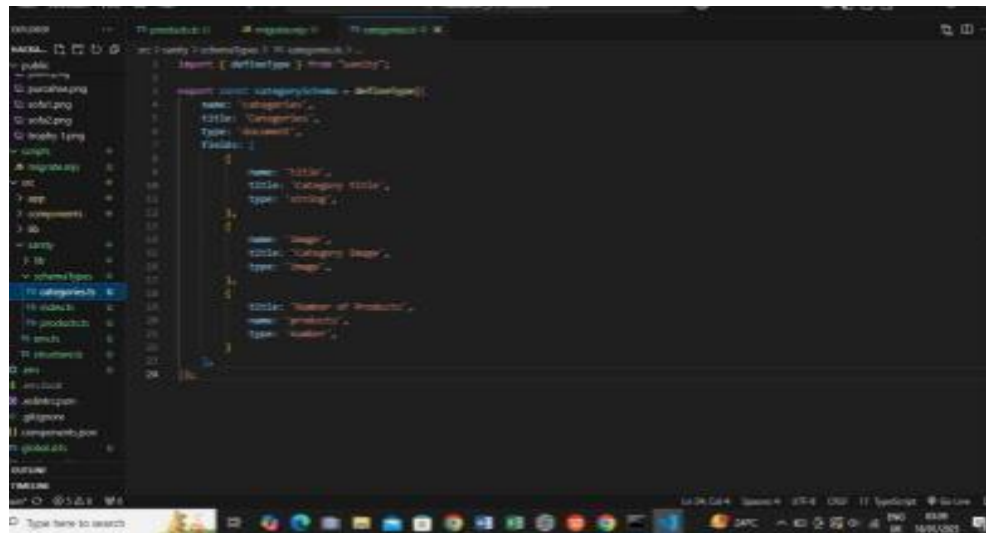
****Product Schema (product.ts)****



****Fields:****

- Title: Product ka name.
- Description: Product ka detailed description.
- Price: Product ki cost.
- Image: Product ka image(s).

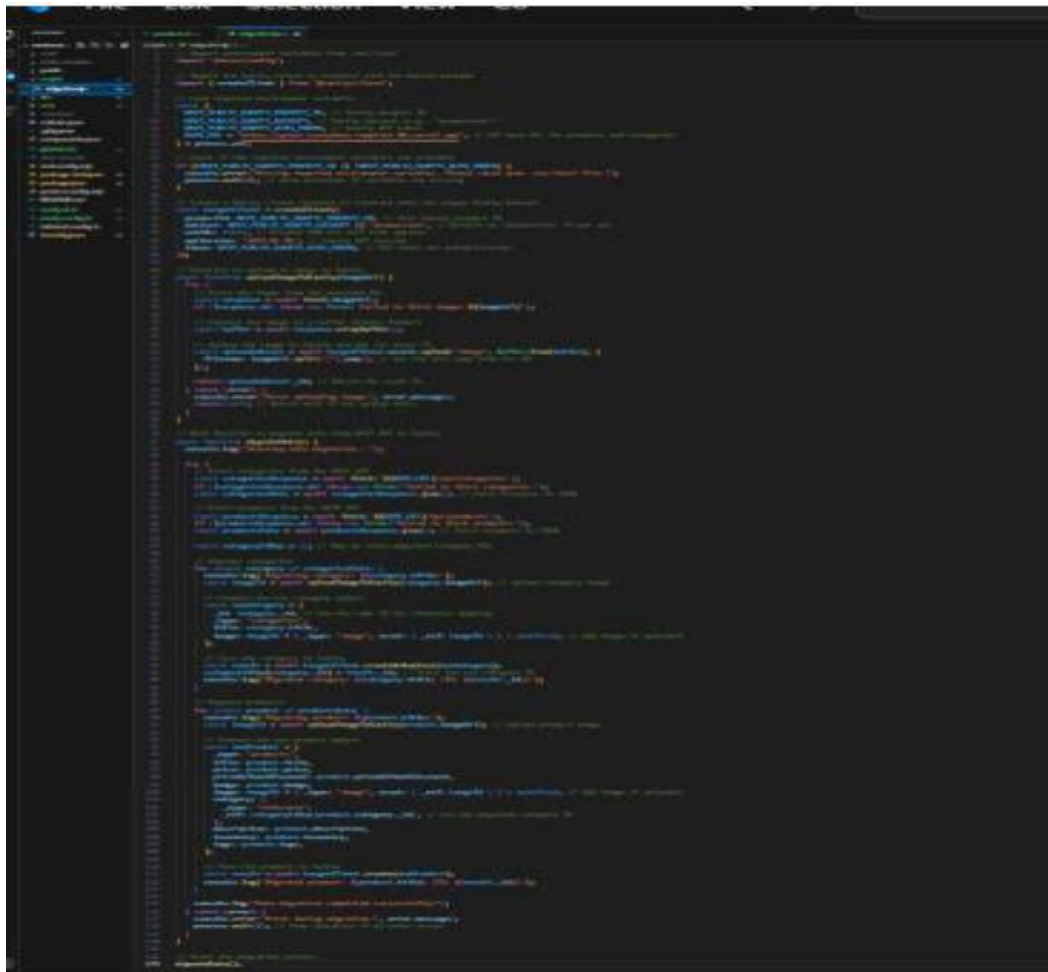
Categories Schema (categories.ts)



- ****Fields:****
- Name: Category ka name.
- Products: Is category se belong karne wale products ka reference.

5. MIGRATION SCRIPT

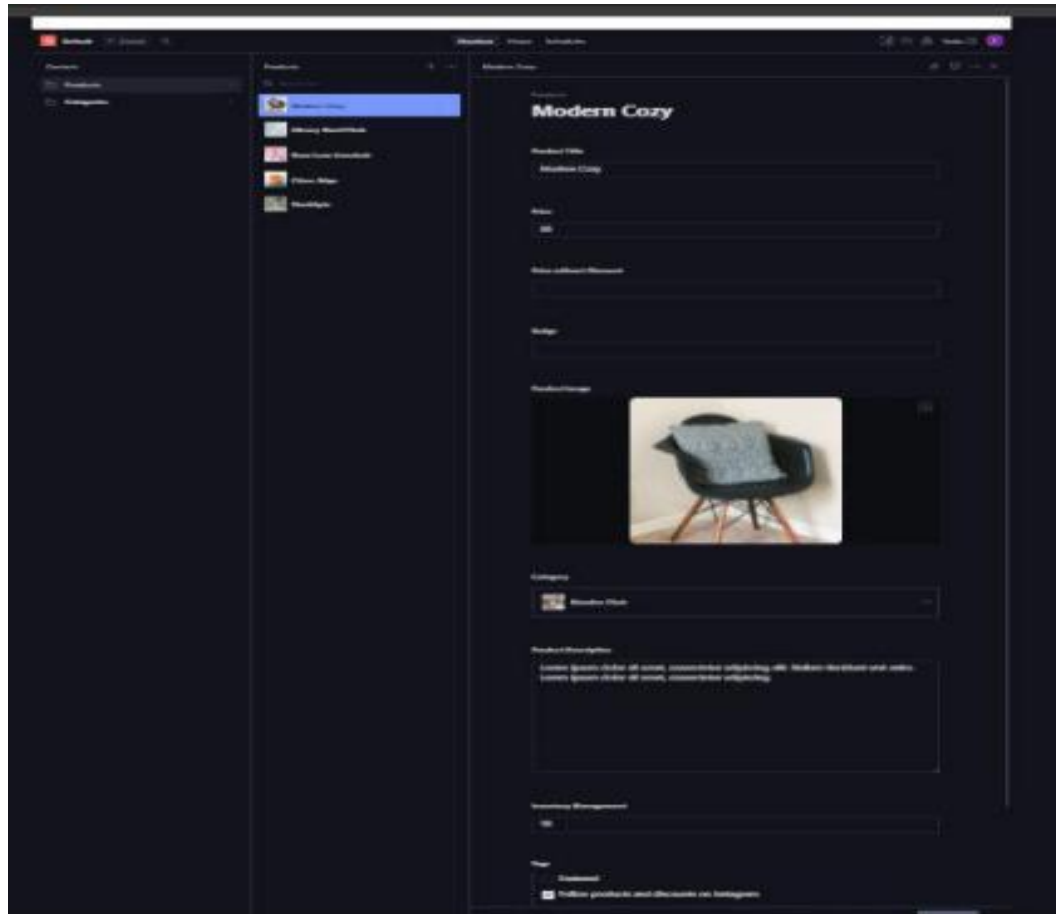
Migration script (`migrate.mjs`) banayi gayi taake data ko automate tarike se Sanity



CMS se local database mein transfer kiya ja sake. Is script ne data fetch karke
`client.create()` method ka use kiya data insert karne ke liye.

6. API INTEGRATION SCREENSHOTS

****Sanity Product Schema:****



- ****Title Field:****

- Type: String
- Purpose: Product ka name store karna (e.g., "Comfortable Chair").
- Validation: Ensure karta hai ke ye field empty na ho kyunki ye required field hai.

- ****Image Field:****

- Type: Image
- Purpose: Product ka ek single image store karna.
- Options:
 - High-resolution images ko support karta hai.
 - Accessibility ke liye alternative text allow karta hai.

- ****Description Field:****

- Type: Text
- Purpose: Product ka detailed description provide karta hai, jaise features, dimensions, aur other details.

- ****Price Field:****

- Type: Number
- Purpose: Product ki cost ko represent karta hai. -
Validation: Ensure karta hai ke price ek positive value ho (e.g., greaterThan(0)).

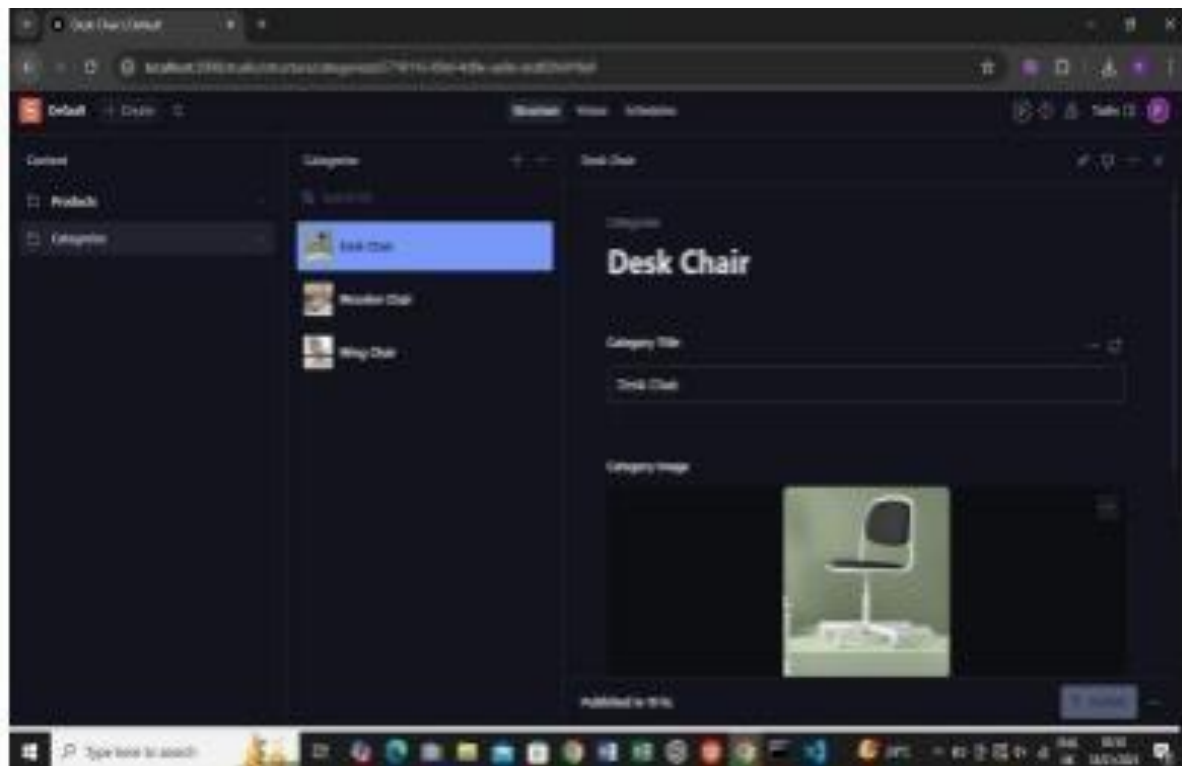
- ****Category Field:****

- Type: Reference ya String
- Purpose: Product ko ek category ke saath associate karta hai (e.g., "Sofas", "Sectionals").
- Benefits: Products ko unki categories ke base par filter aur organize karne mein help karta hai.

****Sanity Categories Schema:****

****Fields:****

- Name: Category ka name (e.g., "Sofas").
- Products: Is category se belong karne wale products ka reference.



7. CONCLUSION

Hackathon ke Day 3 par Comforty Marketplace ke backend mein significant progress hui:

- Data migration process successfully execute hua, jo product aur category data ko Sanity CMS se local database mein transfer karta hai.
- Product aur category ke liye schemas create aur implement kiye gaye.

- Frontend ke saath API integration successful raha, jo marketplace ko data fetch aur display karne ka enable karta ha