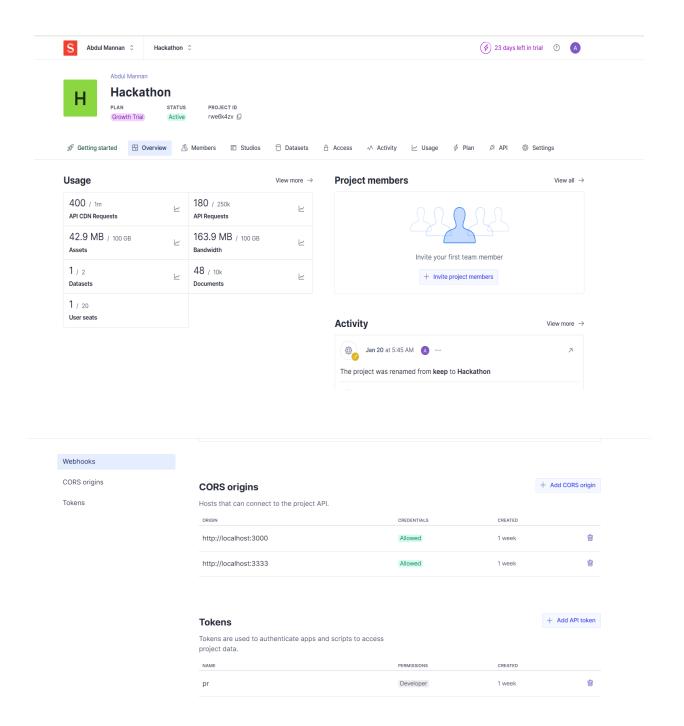
Day 3 Api Integration and Migration

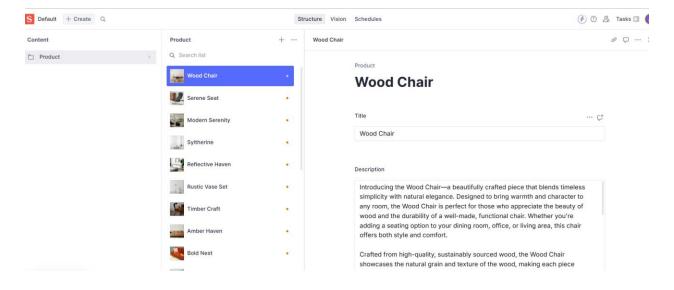


```
ф
                                               다 다 건 영 script > 15 importDatajs > 영 uploadImageToSanity > 영 asset
1 import { createClient } from '@sanity/client';
     ∨ HACKATHON
                                                         00
024
     JS importData.js
                                                         Pieces: Comment | Pieces: Explain
async function uploadImageToSanity(imageUrl) {
    try {
        servels log("Unloading image: $fimageUrl) }
}
                                                            console.log(`Uploading image: ${imageUrl}`);
                                                           const response = await fetch(imageUrl);
if (!response.ok) (
    throw new Error(`Failed to fetch image: $(imageUrl)`);
}
         > About
0
                                                             const buffer = await response.arrayBuffer();
const bufferImage = Buffer.from(buffer);
                                                             console.log(`Image uploaded successfully: ${asset._id}`);
return asset. id;
} catch (error) {
console.error('Failed to upload image:', imageUrl, error);
         > Team
```

```
HACKATHON
                            中の甘む
                                            src > sanity > schemaTypes > TS product.ts > [1] product > [1] fields
                                                   import { defineType } from "sanity"
                                              Click to add a breakpoint
                                                    export const product = defineType({
 > Contact
                                                        name: "product",
 > ListPage
                                                        title: "Product",
 ✓ Login
                                                         type: "document",
  🛱 page.tsx
                                                         fields: [
 > Pricing
                                                                 name: "title",

✓ products \ [id]

  🛱 page.tsx
 > Products1
                                      ٠
                                                                 validation: (rule) => rule.required(),
 ✓ ProductsFromApi
                                                                 type: "string"
  🛱 page.tsx
 > studio
                                                                 name: "description",
 > Team
                                                                 type:"text",
 tavicon.ico
 # globals.css
                                                                 validation: (rule) => rule.required(),
 🗱 layout.tsx
                                      М
                                                                 title: "Description",
 page.tsx
> components
                                                                 name: "productImage",
type: "image",
 > lib
                                                                 validation: (rule) => rule.required(),
                                                                 title: "Product Image"
 TS product.ts
                                                                 name: "price",
 TS env.ts
                                                                 type: "number",
TS middleware.ts
                                                                 validation: (rule) => rule.required(),
                                                                 title: "Price",
 .eslintrc.json
DUTLINE
                                                                 name: "tags",
                                                                 type: "array
IMELINE
```



```
EXPLORER

☆ page.tsx ...\ProductsFromApi U

                                                                                                     {} package.json M X

⇔ page.tsx ...\[id] U

                              中になり
/ HACKATHON
                                              {} package.json > ...
                                                        "name": "hackathon",

∨ ProductsFromApi

                                                        "private": true,
   > studio
                                                        "type": "module",
   > Team
                                                        Debug
  ★ favicon.ico
                                                        "scripts": {
                                                          "dev": "next dev",
  # globals.css
                                                          "build": "next build",
  layout.tsx
                                       М
                                                          "start": "next start",
  🔅 page.tsx
                                                          "lint": "next lint",
  > components
                                                          "import-data": "node script/importData.js"
  > lib
                                                        "dependencies": {
   > lib
                                                          "@clerk/nextjs": "^6.9.15",
```

