Areeba Awan
Roll No: 00262538

Hackathon-3 [2025]

Comforty Furniture Marketplace Project Complete Documentation from Day1 to Day-6

Project Overview

Comforty is a modern e-commerce platform designed to simplify the buying and selling of furniture online. The goal of this platform is to empower small businesses and entrepreneurs by providing them with an easy-to-use, secure, and efficient marketplace. Over the course of six days, the project progressed through several stages, from conceptualization and technical planning to deployment and staging. This documentation outlines the entire development process and highlights the key milestones and tasks completed.

Day 1: Conceptualization and Marketplace Design

Key Achievements:

- **Marketplace Type:** A general-purpose e-commerce platform for furniture sales.
- **Business Goals:**
  - Empower small businesses by offering an accessible online marketplace.
  - Provide customers with an easy and secure platform to buy and sell furniture.
- **Data Schema Design:**
  - **Entities:**
    - **Products:** Details of furniture available for sale.
    - **Orders:** Customer orders for products.

        - **Customers:** User accounts and order history.
        - **Delivery Zones:** Geographic zones for delivery logistics.
    - **Relationships:**
        - Customers can place orders, each referencing products.
        - Delivery zones are linked to specific drivers for fulfillment.

Day 2: Technical Planning

Key Achievements:

- **Tech Stack:**
    - **Frontend:** Next.js for the React framework with Tailwind CSS for styling.
    - **Backend:** Sanity CMS for content management.
    - **Database:** Sanity Headless CMS to store sensitive user data and manage authentication.
    - **APIs:**
        - **ShipEngine** for order tracking.
        - **Stripe** for secure payment processing.
- **API Requirements:**
    - **User Management:**
        - **/register –** For user registration.
        - **/login –** For user login.
        - **/verify-route –** For verifying user sessions.
    - **Product Management:**
        - **/products –** To fetch all products.
        - **/product/:id –** To fetch specific product details.
    - **Orders:**
        - **/orders** (POST) – To create a new order.
        - **/shipment/:id** (GET) – To track shipments.
- **Deployment Plan:**
    - **Frontend:** Deployed on Vercel for seamless CI/CD.
    - **Backend:** Hosted on AWS Lambda for serverless architecture.

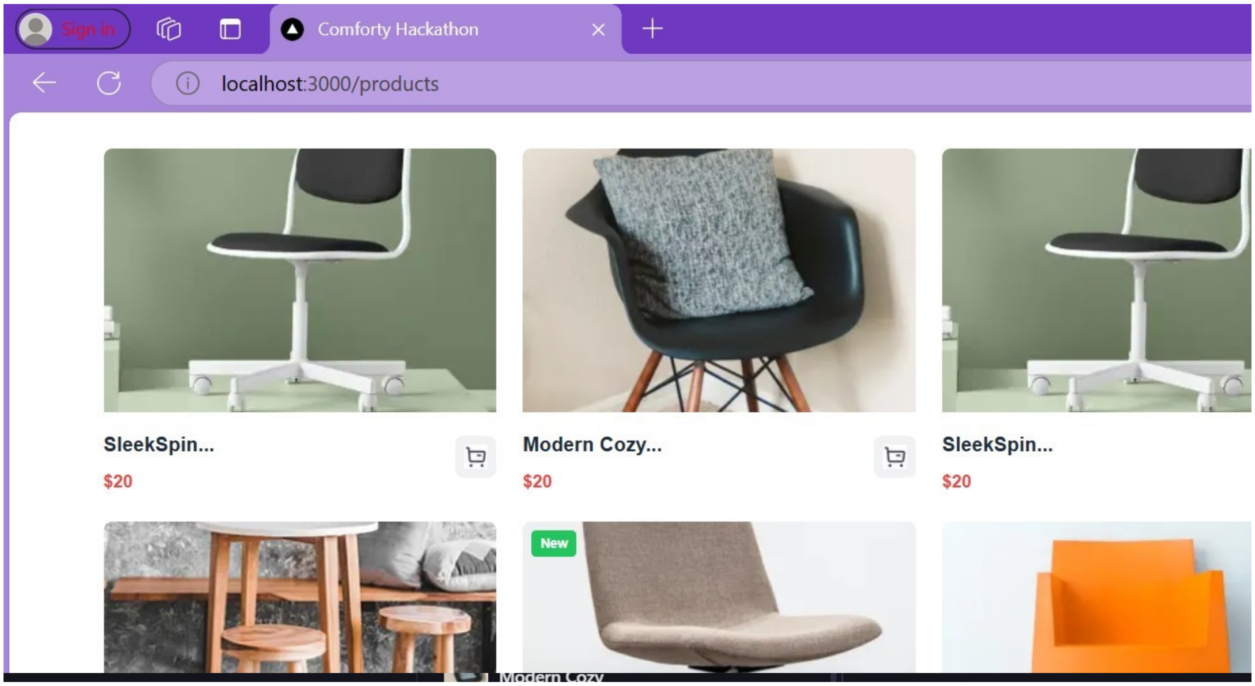
Day 3: Data Migration

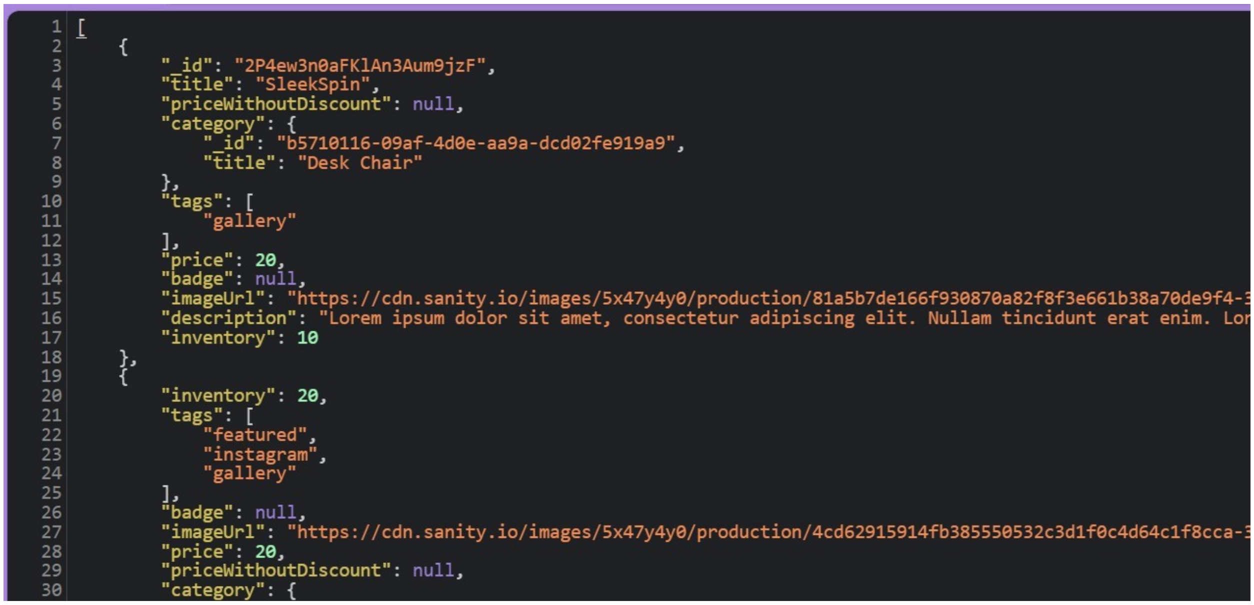
Key Achievements:

- **Custom Migration Code:**
  - Migrated product data from **Sanity CMS** to **Next.js** using **GROQ** queries.
  - Example **GROQ Query:**

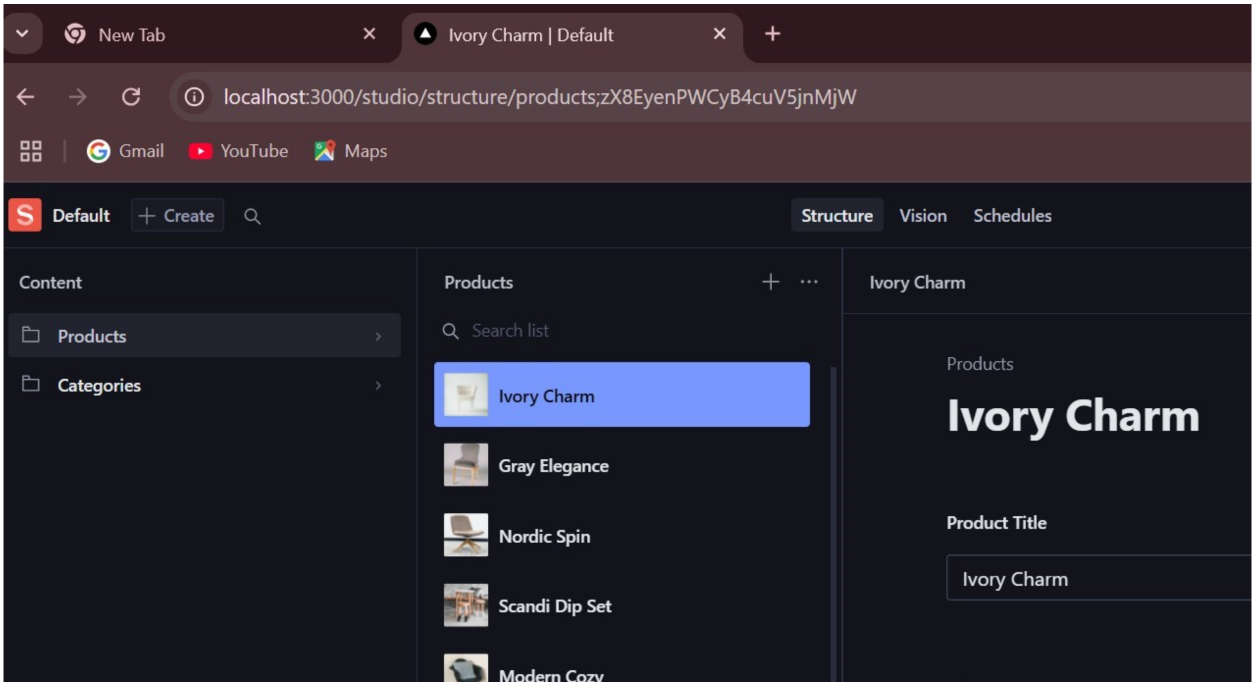
    groq
    ***[_type == "product"] {title, description, price, image}**

- **Schema Definition:**
  - **Products Schema:**
    - Title
    - Slug (for dynamic routing)
    - Description
    - Price
    - Image (for displaying product visuals)
- **Client Integration:**
  - Dynamically fetched and displayed data on the homepage using the Sanity CMS API.





```
[
    {
        "_id": "2P4ew3n0aFKlAn3Aum9jzF",
        "title": "SleekSpin",
        "priceWithoutDiscount": null,
        "category": {
            "_id": "b5710116-09af-4d0e-aa9a-dcd02fe919a9",
            "title": "Desk Chair"
        },
        "tags": [
            "gallery"
        ],
        "price": 20,
        "badge": null,
        "imageUrl": "https://cdn.sanity.io/images/5x47y4y0/production/81a5b7de166f930870a82f8f3e661b38a70de9f4-3
        "description": "Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam tincidunt erat enim. Lor
        "inventory": 10
    },
    {
        "inventory": 20,
        "tags": [
            "featured",
            "instagram",
            "gallery"
        ],
        "badge": null,
        "imageUrl": "https://cdn.sanity.io/images/5x47y4y0/production/4cd62915914fb385550532c3d1f0c4d64c1f8cca-3
        "price": 20,
        "priceWithoutDiscount": null,
        "category": {
```



Product Title: Ivory Charm

Day 4: Building Dynamic Frontend Components

Key Achievements:

- **Dynamic Product Listings:**
  - Created the **ProductList** component to display products dynamically fetched from Sanity CMS.
- **Filters and Sorting:**
  - Implemented filtering options based on categories and price range.
  - Added sorting options like price (low-high), price (high-low), and popularity.
- **Reusable Components:**
  - **ProductCard:** Displays product images, titles, and prices.
  - **FilterSidebar:** A sidebar for easy filtering and sorting of products.

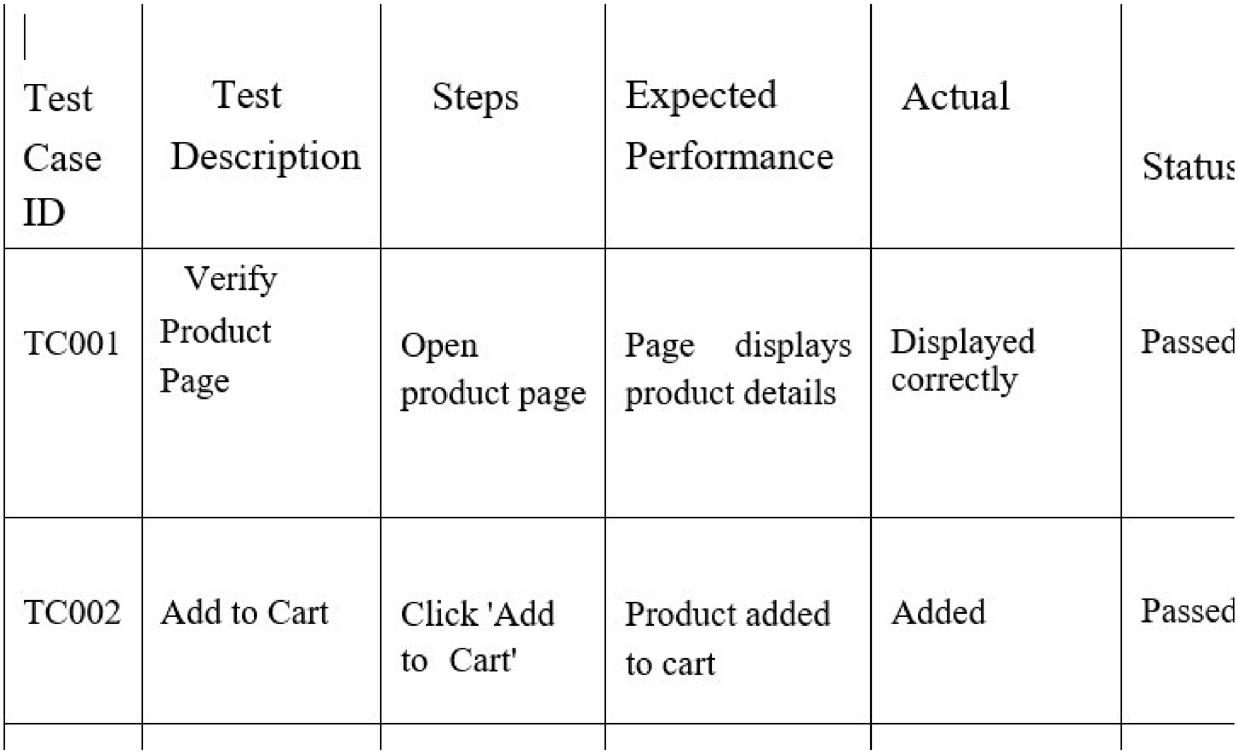
    - **PaginationControls:** Allows users to navigate through multiple pages of product listings.

Day 5: Testing and Backend Refinement

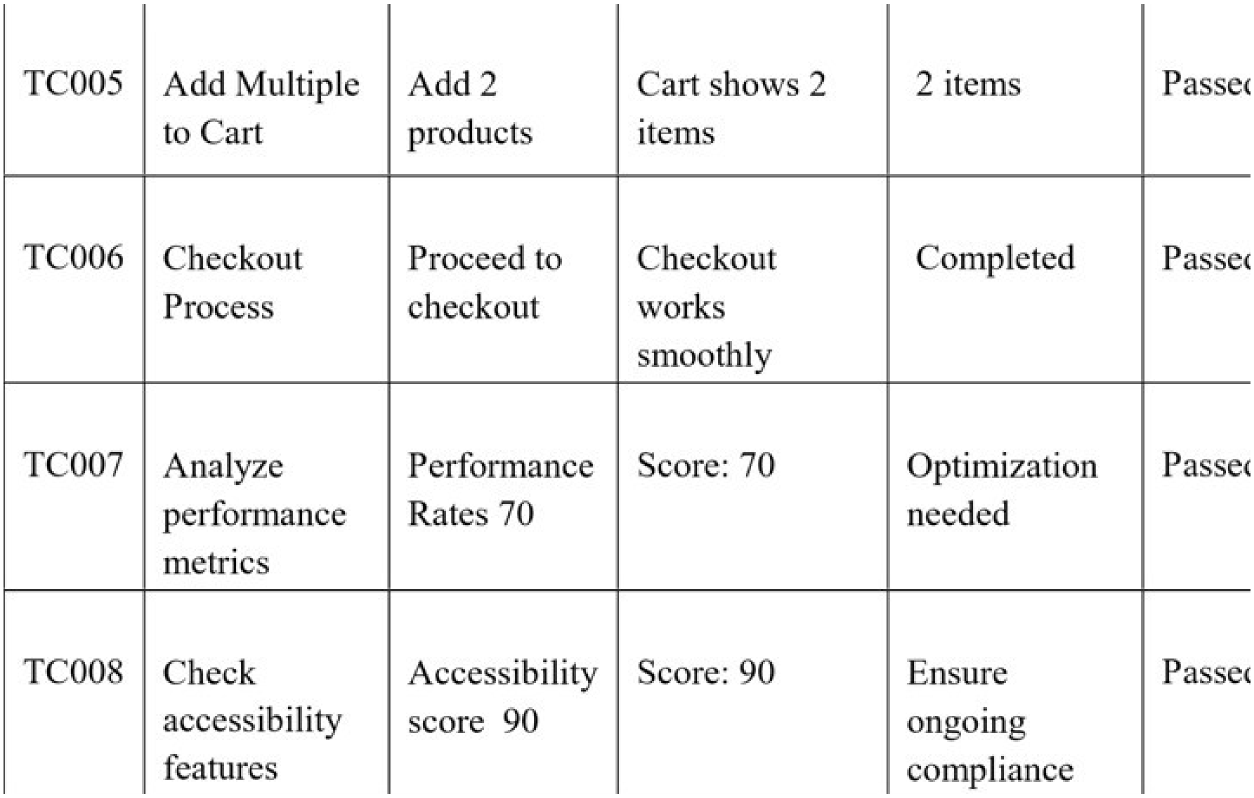
Key Achievements:

- **Testing Types:**
    - **Functional Testing:**
        - Verified key workflows such as product listings, cart operations, and API interactions.
    - **Performance Testing:**
        - Analyzed performance using **Lighthouse** to check load times and responsiveness.
    - **Security Testing:**
        - Ensured secure handling of input fields, API keys, and HTTPS communication.

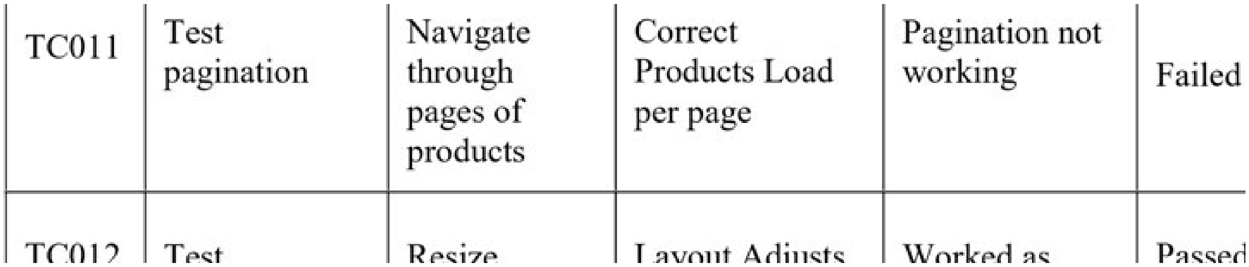
- ***CSV-Based Testing Report:***



Test Case ID	Test Description	Steps	Expected Performance	Actual	Status
TC001	Verify Product Page	Open product page	Page displays product details	Displayed correctly	Passed
TC002	Add to Cart	Click 'Add to Cart'	Product added to cart	Added	Passed



TC005	Add Multiple to Cart	Add 2 products	Cart shows 2 items	2 items	Passed
TC006	Checkout Process	Proceed to checkout	Checkout works smoothly	Completed	Passed
TC007	Analyze performance metrics	Performance Rates 70	Score: 70	Optimization needed	Passed
TC008	Check accessibility features	Accessibility score 90	Score: 90	Ensure ongoing compliance	Passed



TC011	Test pagination	Navigate through pages of products	Correct Products Load per page	Pagination not working	Failed
TC012	Test	Resize	Layout Adjusts	Worked as	Passed

Day 6: Deployment Preparation and Staging Environment Setup

Key Achievements:

- **Deployment Strategy:**
  - **Frontend Deployment:** Deployed on **Vercel** to enable fast iteration and CI/CD integration.
  - **Backend Deployment:** Set up **AWS Lambda** for a serverless backend.
- **Environment Variables:**
  - Sensitive data (e.g., API keys, project IDs) were securely stored in .env files, then uploaded to Vercel for safe management.
  - Example .env file:

    ```
    NEXT_PUBLIC_SANITY_PROJECT_ID=your_project_id
    NEXT_PUBLIC_SANITY_DATASET=production
    API_KEY=your_api_key
    ```

- **Staging Environment:**
  - Deployed a staging version of the platform to ensure functionality is correct in a production-like environment.
- **Staging Testing:**
  - **Functional Testing:** Verified core features like product listings and checkout flow.
  - **Performance Testing:** Used **GTmetrix** for evaluating website speed and responsiveness.
  - **Security Testing:** Ensured API calls and user data handling were secure with HTTPS.
- **Documentation:**
  - Created a **README.md** to document the project setup, structure, and deployment instructions.

        - Organized the GitHub repository with the following structure:
            - src/
            - public/
            - documents/

GitHub Repository Structure:

```
Comforty/
├── src/
│   ├── components/
│   │   ├── ProductCard.js
│   │   ├── FilterSidebar.js
│   │   ├── PaginationControls.js
│   └── pages/
│       ├── index.js
│       ├── product/
│       └── [slug].js
├── public/
│   ├── images/
│   └── assets/
├── documents/
│   ├── Day_1_Conceptualization.pdf
│   ├── Day_2_Technical_Planning.pdf
│   ├── Day_3_Data_Migration.pdf
│   ├── Day_4_Dynamic_Components.pdf
│   ├── Day_5_Testing_Report.csv
│   └── Day_6_Deployment.pdf
├── .env
├── README.md
```

Conclusion

The **Comforty** Furniture Marketplace project successfully transitioned from a conceptual idea to a fully developed, deployable platform. The project features a dynamic, user-friendly interface, secure backend services, and has been thoroughly tested for functionality, performance, and security.

Next Steps:

1. Address any minor issues or improvements noted during the staging testing phase.
2. Monitor the live environment for user feedback and performance metrics.
3. Plan for scaling the platform with features such as multi-language support, advanced search filters, and predictive analytics.

This marks the **successful completion** of the **Comforty** Furniture Marketplace project!

Prepared with Passion By Areeba Awan