

Name: Alina Fatima

Roll Number: 347720

E-commerce Marketplace Hackathon Documentation

Overview

The E-commerce Marketplace is a dynamic and user-friendly platform designed to enable businesses to showcase and sell their products online. This project was developed during a hackathon over six intensive days, focusing on leveraging cutting-edge technologies such as Next.js (version 15), Sanity CMS, and Clerk for authentication. The goal was to create an adaptable, scalable e-commerce solution with a seamless user experience.

This documentation provides a detailed account of the development journey, technical decisions, challenges faced, and plans for future enhancements, including live streaming integration and more robust features.

Development Timeline

Day 1: Conceptualization and Marketplace Design

Key Achievements:

- Brainstormed the core idea and identified the business goals:
 - Build an intuitive platform for customers to explore and purchase products.
 - Empower small businesses to sell their offerings with minimal setup.
- Designed a basic marketplace structure.
- **Data Schema Design:**
 - **Entities:** Products, Orders, Customers, and Categories.
 - **Relationships:**
 - Products are categorized into categories.
 - Customers place orders that reference products.

Day 2: Technical Planning

Key Achievements:

- Selected the tech stack:
 - **Frontend:** Next.js 15 with Tailwind CSS for responsive and modern design.
 - **Backend:** Sanity CMS for flexible content management with live editing capabilities.
 - **Authentication:** Clerk for user authentication and session management.
 - **Payment Gateway:** Stripe for secure transactions.
- Identified API requirements:
 - **User Management:** /register, /login, /logout.
 - **Product Management:** /products, /products/:id.
 - **Order Management:** /orders, /orders/:id.
- Deployment Plan:
 - Use Vercel for rapid deployments with server less architecture.

Day 3: Data Schema and Migration

Key Achievements:

- Defined Sanity schemas for products, categories, orders, and sales using TypeScript-generated schemas in Next.js 15 for type safety.
- **Example Product Schema:**
- ```
export const productType = defineType({
 name: 'product',
 title: 'Products',
 type: 'document',
 fields: [
 { name: 'name', title: 'Product Name', type: 'string', validation: Rule => Rule.required() },
 { name: 'slug', title: 'Slug', type: 'slug', options: { source: 'name', maxLength: 96 }, validation: Rule => Rule.required() },
 { name: 'price', title: 'Price', type: 'number', validation: Rule => Rule.required().min(0) },
 { name: 'stock', title: 'Stock', type: 'number', validation: Rule => Rule.required().min(0) },
],
});
```
- Migrated product data from Sanity CMS to the frontend using GROQ queries.
- Example query to fetch all products:
 

```
*[_type == "product"] { title, slug, price, image }
```

## Day 4: Dynamic Frontend Components

### Key Achievements:

- Built reusable components such as Header, Banner, Loading, Add to basket, and Product view for displaying dynamic data.
- Implemented filtering and sorting functionalities for categories and prices.
- Used helper functions to fetch data from Sanity:
 

```
export const getAllCategories = async () => {
 const query = '*[_type=="category"] | order(name asc)';
```

- try {
- const categories = await sanityFetch({ query });
- return categories || [];
- } catch (error) {
- console.error("Error fetching categories", error);
- return [];
- }
- };

## Day 5: Testing and Backend Refinement

### Key Achievements:

- Performed extensive testing:
  - **Functional Testing:** Verified workflows such as product listings, cart operations, and checkout processes.
  - **Performance Testing:** Used Lighthouse to ensure fast load times.
  - **Security Testing:** Validated API keys, HTTPS implementation, and form input handling.
- Configured Stripe CLI to handle webhooks for real-time updates:
- const stripe = require('stripe')(process.env.STRIPE\_API\_KEY);
- 
- app.post('/webhook', express.raw({ type: 'application/json' })), async (req, res) => {
- const sig = req.headers['stripe-signature'];
- let event;
- try {
- event = stripe.webhooks.constructEvent(req.body, sig, process.env.STRIPE\_WEBHOOK\_SECRET);
- } catch (err) {
- res.status(400).send(`Webhook Error: \${err.message}`);
- return;
- }
- // Handle event types
- if (event.type === 'checkout.session.completed') {
- // Update Sanity with order info
- }
- res.json({ received: true });
- };

## Day 6: Deployment and Staging Environment

### Key Achievements:

- Deployed the application to Vercel with CI/CD integration.
- Configured environment variables securely in .env and Vercel.
- Set up a staging environment to validate features and performance before going live.
- Structured GitHub repository for clarity and maintainability:
- EcommerceWebsite/

- |— src/
  - | |— components/
  - | |— pages/
  - |— public/
  - |— .env
  - |— README.md
- 

## Exploration and Learning

- **Next.js 15:**
    - Used its latest features for type-safe schema definitions and optimized rendering.
    - Explored dynamic routing for product and category pages.
  - **Sanity CMS:**
    - Learned to create schemas with live editing capabilities.
    - Mastered GROQ for efficient data querying.
  - **API Integration:**
    - Experimented with Clerk for user management and Stripe for secure payments.
  - **Helper Functions:**
    - Created modular functions to fetch and transform data, ensuring clean code practices.
- 

## Future Enhancements

1. **Live Streaming:**
    - Integrate real-time streaming for product showcases.
  2. **AI-Powered Recommendations:**
    - Implement AI for personalized shopping experiences.
  3. **Localization:**
    - Add multi-language support for a global audience.
  4. **Predictive Analytics:**
    - Use analytics to predict trends and optimize inventory.
- 

## Conclusion

The E-commerce Marketplace project successfully transitioned from concept to deployment in just six days, demonstrating a deep understanding of modern web development practices. With plans to expand features and incorporate advanced technologies, this platform is positioned to deliver exceptional user experiences.

