Day 5- Testing and Backend Refinement – "Furniro"

Introduction

This report outlines the testing, error handling, performance optimization, and backend integration refinement for the responsive e-commerce website built during the Marketplace Builder Hackathon. The project was developed using Next.js and Tailwind CSS and integrated with Sanity API for dynamic content management. The website includes essential features such as a home page, shop page, product details via dynamic routing, a cart with Add-to-Cart functionality, and a checkout process.

Objective Of Day5:

1. Validating all core functionalities, including Add-to-Cart and dynamic routing.
2. Refining error handling for API calls and data fetching from Sanity.
3. Testing the platform's compatibility across devices and browsers.

Testing Scenarios and Results

The following tests were conducted to ensure the platform's functionality and performance:

Functional Testing

Test Case ID	Description	Steps	Expected Result	Actual Result	Status
TC-001	Validate product listing	Load the shop page and verify product displays.	All products appear correctly.	As expected	Passed
TC-002	Add to Cart functionality	Add items to the cart and verify quantity and total price.	Cart updates correctly with total price.	As expected	Passed
TC-003	Dynamic routing for product pages	Navigate to product details via dynamic routing.	Product details load without errors.	As expected	Passed

TC-004	API data fetching from Sanity	Fetch products from Sanity and display them on the shop.	Data loads without delay or errors.	As expected	Passed

Error Handling

Scenario	Implementation	Outcome
Network failures	Used try-catch blocks for API calls. Displayed a fallback UI.	User-friendly error message.
Missing data from Sanity	Added checks for missing fields and default placeholders.	Seamless user experience.
Invalid quantity input	Added validation to prevent negative or non-numeric quantities in the cart.	Error message shown to the user.

Security Testing

- **Measures Implemented:**
  - Secured API communication over HTTPS.
  - Stored sensitive API keys in environment variables.

User Acceptance Testing (UAT)

Real-world scenarios were simulated to test:

- Adding items to the cart and completing the checkout.
- Usability of the website across different devices and screen sizes.

Conclusion

The e-commerce website has been thoroughly tested, optimized, and validated for functionality, performance, and security. I hope to solve any future errors I might encounter in further days of Hackathon.

Prepared by: Fatima Salman(464666)