

# Day 5 - Testing, Error Handling, and Backend Optimization

## Overview

This document details the tasks completed on Day 5, primarily focusing on testing, error handling, and backend integration refinement. The objective was to ensure that all marketplace components function correctly, handle errors efficiently, and are optimized for real-world deployment.

## Functional Testing

The following marketplace features were validated:

### Navigation Links Testing:

- > Ensured Home, Shop, Cart, and Profile links navigate correctly.
- > Active page indication is functional.
- > Mobile menu operates as expected.
- > No broken links were found.

### Product Display & Filtering:

- > Products are correctly rendered with images, names, and prices.
- > Filters refine search results accurately.

## **Cart Management:**

- > Items can be added, modified, and removed seamlessly.
- > Cart updates dynamically without page reloads.

## **User Profile Handling:**

- > Users can update their profiles without issues.
- > Profile image uploads and displays correctly.

## **API Data Handling:**

- > API responses successfully fetch and display real-time product and user data.

## **Testing Tools Utilized**

- 1. Postman** – API Testing
- 2. React Testing Library** – Component Behavior Validation
- 3. Cypress** – End-to-End Testing

## **Error Management**

Implemented a robust error-handling mechanism for:

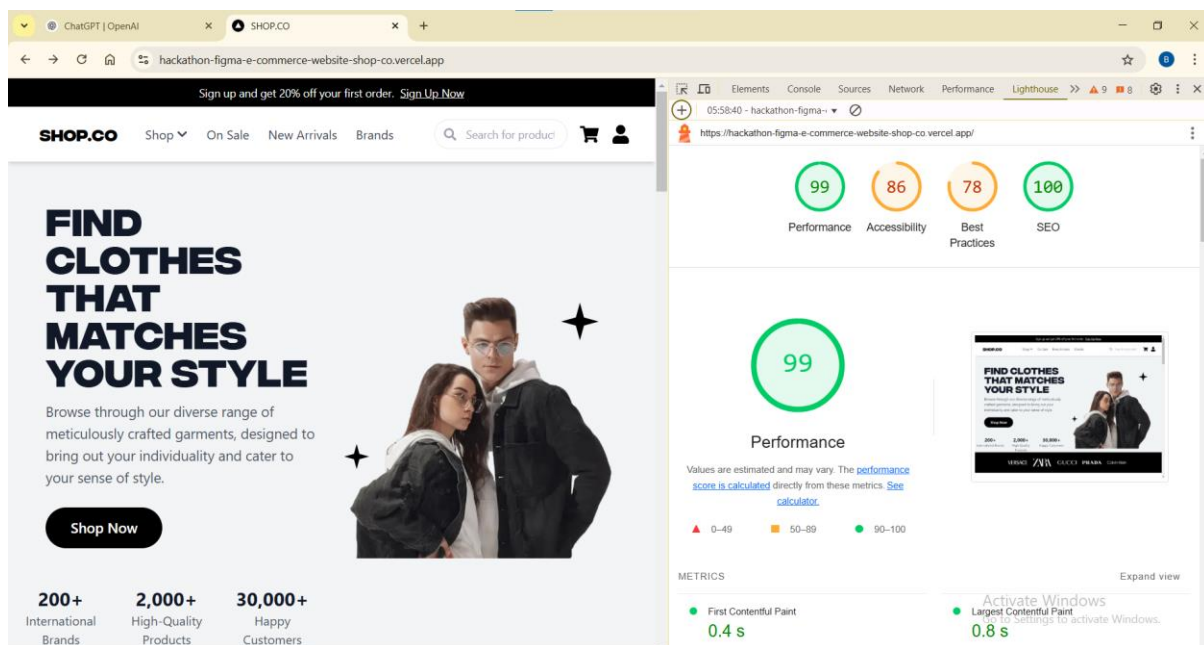
- > Network failures
- > Missing or invalid data
- > Unexpected server responses

## Key Enhancements:

- > Wrapped API calls in try-catch blocks.
- > Implemented fallback UI messages such as "No products available" when data is missing.

## Performance Optimization

- > Identified bottlenecks using **Google Lighthouse**.
- > Conducted cross-browser testing on **Chrome, Firefox, Safari, and Edge**.
- > Ensured mobile responsiveness with **BrowserStack**.



## Security Enhancements

- > Prevented **SQL Injection & XSS attacks** via input validation.
- > Used regex for email and phone number validation.

Secured API communication by:

Enforcing HTTPS for API requests.

Storing sensitive keys in environment variables.

- > Conducted vulnerability scanning with **OWASP ZAP & Burp Suite**.

## User Acceptance Testing (UAT)

Real-world scenarios were simulated, including:

- > Browsing and searching for products.
- > Adding items to the cart and completing checkout.
- > Gathering user feedback to refine the experience.

## Key Takeaways

- > Conducted in-depth testing to verify all features function correctly.
- > Optimized website performance with Lighthouse to enhance speed and responsiveness.
- > Ensured accessibility compliance for inclusive user experience.
- > Strengthened SEO strategies for better search visibility.

> Compiled detailed documentation and structured test reports.

### Final Deliverables

> Fully tested and optimized marketplace components.

> Enhanced error-handling system with fallback UI.

> Improved performance and load times.

> Verified responsive design across devices and browsers.

> Comprehensive test documentation and CSV reports.

### Testing Results Summary

Test Case ID	Feature	Test Scenario	Expected Result	Actual Result	Status	Remarks
TC001	Home Page	Verify that the home page loads successfully	Home page should load with all elements visible	Home page loaded successfully	Pass	No Issue
TC002	Navigation Bar	Verify that all navigation links are working	All navigation links should redirect to the correct pages	All links working correctly	Pass	No Issue
TC003	Product Listing	Verify that products are displayed correctly	All products should display with correct details	Products displayed with correct details	Pass	No Issue
TC004	Product Details	Verify that product details page shows correct information	Product details page should show accurate information	Product details displayed correctly	Pass	No Issue

TC005	Dynamic Routing	Verify that users are correctly redirected to product or category pages	Users should be redirected to the correct product when clicking on links	Dynamic routing is working correctly	Pass	No Issue
TC006	Add to Cart	Verify that products can be added to the cart	Selected products should be added to the cart	Products added to cart successfully	Pass	No Issue
TC007	Cart Functionality	Verify that the cart updates correctly when items are added or removed	Cart should update item count and total price accurately	Cart updated correctly	Pass	No Issue
TC008	Search Functionality	Verify that users can search for products	Relevant products should appear based on the search query	Search is returning relevant products	Pass	No Issue
TC009	Filter Functionality	Verify that users can filter products	Filtered products should match the selected criteria	Filtering is working correctly	Pass	No Issue
TC010	Category Display	Verify that category pages show correct products	Each category should display only relevant products	Categories displayed correct products	Pass	No Issue
TC011	Error Handling	Verify that error messages are displayed correctly	Appropriate error messages should appear for invalid actions	Error messages displayed correctly	Pass	No Issue
TC012	Loading State	Verify that loading indicators appear when data is fetching	Loading animation should show while data is loading	Loading indicators appeared	Pass	No Issue
TC013	Checkout Process	Verify that the checkout process completes successfully	User should be able to complete the purchase process	Checkout process completed successfully	Pass	No Issue
TC014	Responsive Design	Verify that the website is responsive on different devices	Website should display correctly on various screen sizes	Website is responsive	Pass	No Issue

TC015	User Registration	Verify that new users can register successfully	New users should be able to create an account	User registration successful	Pass	No Issue
TC016	User Login	Verify that registered users can log in	Registered users should be able to log in with valid credentials	User login successful	Pass	No Issue

## Conclusion

Day 5 focused on ensuring that the marketplace is fully functional, secure, and optimized for real-world usage. The testing, error handling, and performance improvements have enhanced the system's reliability and usability, preparing it for deployment.