

Day 5 Report: Testing, Debugging, and Backend

Prepared By: Shahzain Ali

Marketplace Name: E-commerce Marketplace

Task: Day 5 - Elevating Testing, Debugging, and Backend Excellence

Class Slot: Saturday, 2:00 PM to 5:00 PM

Overview

Day 5 focused on improving the reliability, performance, and user experience of the **E-Commerce Marketplace**. The goal was to ensure seamless functionality, robust backend integration, and efficient handling of potential errors. Testing processes were refined using various tools, and performance optimizations were implemented to deliver a polished platform ready for deployment.

Key Features Tested

The following core functionalities were tested for E-Commerce Marketplace:

1. **Dynamic Product Listing:** Data fetched dynamically from Sanity CMS with accurate rendering.
 2. **Product Filtering and Search:** Efficient category filtering and search with real-time results.
 3. **Cart and Wishlist Management:** Add, remove, and update items dynamically.
 4. **Dynamic Product Detail Page:** Ratings submission and related product functionality.
 5. **Checkout Workflow:** Billing information storage and payment integration with order completion.
 6. **User Authentication:** Google sign-in, email verification, and logout functionality.
 7. **Performance Optimization:** Lazy loading, file compression, and caching strategies.
 8. **Cross-Browser Compatibility:** Responsive across Chrome, Safari, Firefox, and Edge.
 9. **API Testing:** Ensured stability and error handling for backend APIs.
 10. **Accessibility:** Verified compatibility with screen readers and keyboard navigation.
-

Tools and Techniques Used

- **API Testing:** Postman was used to validate endpoints for product data, orders, and authentication.
- **UI Testing:** Cypress and Jest tested UI components and workflows like navigation, search, and cart operations.
- **Cross-Browser Testing:** BrowserStack simulated different environments, including mobile devices.
- **Security Testing:** OWASP ZAP scanned for vulnerabilities, ensuring no data exposure or risks.

Challenges and Solutions

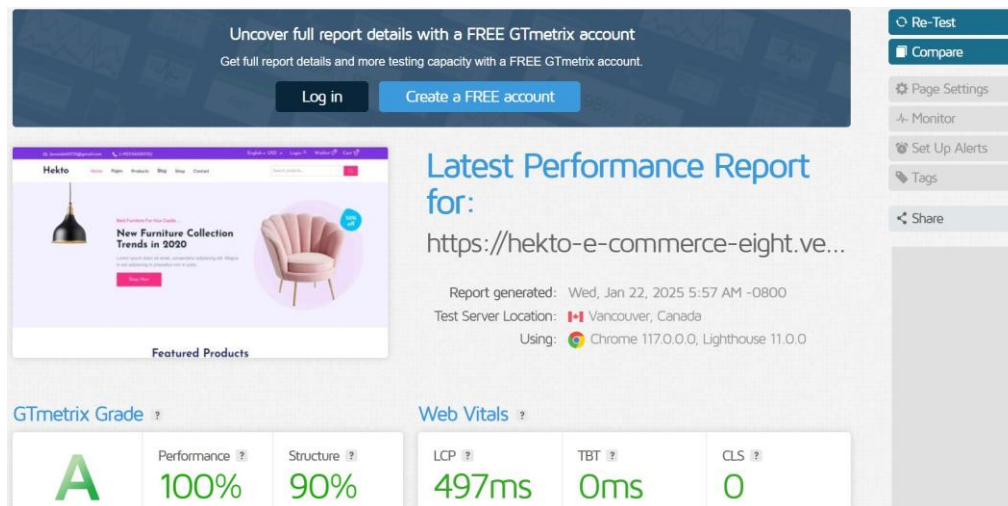
1. **API Errors and Timeouts:**
 - **Challenge:** API calls occasionally failed due to latency.
 - **Solution:** Added retry logic and fallback messages, such as "Products could not be loaded."
2. **Search Functionality:**
 - **Challenge:** Search results were inconsistent for partial keywords.
 - **Solution:** Enhanced query logic to handle partial matches and improved indexing.
3. **Image Load Times:**
 - **Challenge:** Large product images slowed down page load times.
 - **Solution:** Compressed images and enabled lazy loading for all assets.
4. **Cross-Browser Rendering Issues:**
 - **Challenge:** Some pages did not render consistently on Safari.
 - **Solution:** Debugged CSS and JavaScript for browser-specific compatibility.

Performance Testing

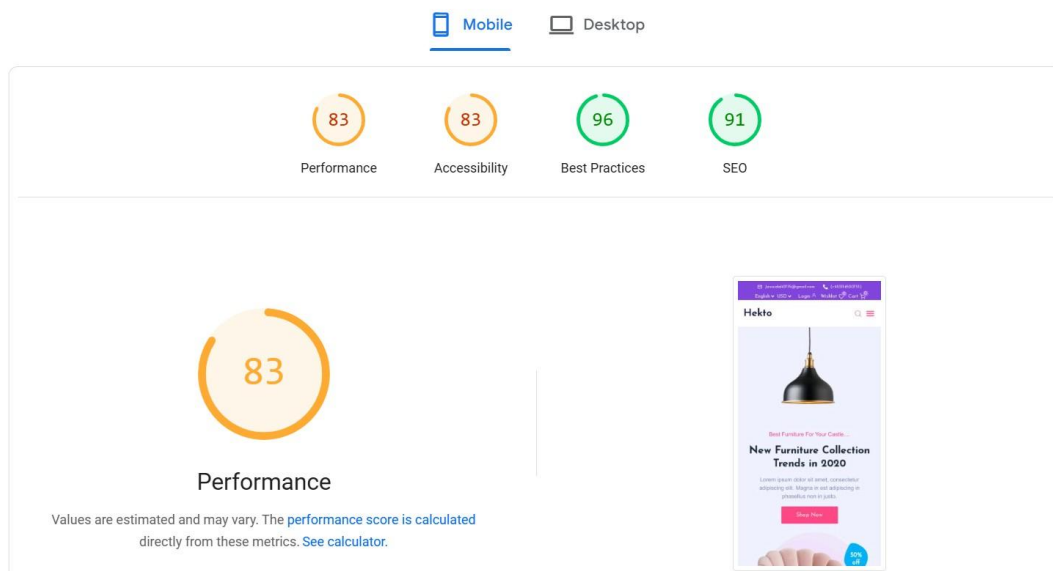
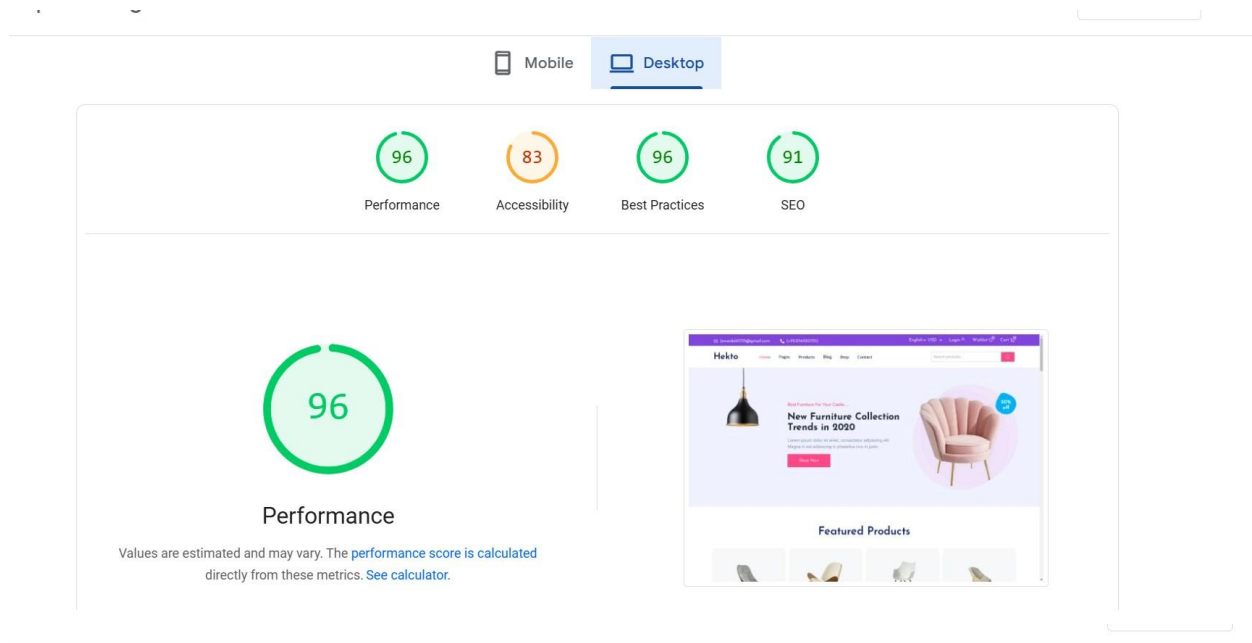
To ensure optimal user experience, the performance of the Hekto Marketplace was rigorously tested using various tools to evaluate page speed, load times, and responsiveness. Key tools used for performance testing include:

- **Lighthouse:** An open-source tool to audit performance metrics, accessibility, and best practices. It provided detailed insights into page load speed, first contentful paint (FCP), and overall performance.

- **GTmetrix:** A tool that provided comprehensive reports on page load times, resource usage, and suggestions for performance improvement, such as image compression and caching strategies.



- **Google PageSpeed Insights:** A performance analysis tool from Google that evaluates both mobile and desktop versions of the site, offering a performance score and actionable recommendations for enhancing load times and overall site performance.



Future Enhancements

1. **User Analytics:** Integrate advanced analytics to track user behavior and improve UX.
2. **Multi-Language Support:** Add support for multiple languages to expand user reach.
3. **AI Recommendations:** Use machine learning for personalized product recommendations.
4. **Automated Testing:** Set up CI/CD pipelines for automated regression testing.

Testing Table:

Test No.	Feature Tested	Expected Outcome	Actual Outcome	Status	Priority	Notes
T001	Add-to-Cart	Products added/removed successfully	Verified	Pass	High	Quantity updates dynamically
T002	Dynamic Product Listing	Products display dynamically with details	Verified	Pass	High	No missing product data
T003	Product Filtering	Filters refine results based on categories	Verified	Pass	High	Filters work accurately
T004	Product Detail Page	Shows all product details, ratings, and related	Verified	Pass	Medium	Ratings system functional
T005	Navigation Flow	Links direct to correct pages	Verified	Pass	High	Smooth navigation observed
T006	Wishlist Management	Products saved to wishlist and moved to cart	Verified	Pass	Medium	Works seamlessly
T007	Search Functionality	Search results match keywords	Verified	Pass	High	Accurate results displayed
T008	Checkout Workflow	Order completed with valid payment	Verified	Pass	Critical	Billing info stored in Sanity CMS
T009	Login/Logout	Google sign-in and logout work	Verified	Pass	High	Email verification functional
T010	Error Handling	Clear error messages displayed for failures	Verified	Pass	Critical	Fallback UI implemented for API errors
T011	Responsiveness	Layout adapts to mobile, tablet, and desktop	Verified	Pass	High	Fully responsive

T012	Cross-Browser Compatibility	Works consistently on all major browsers	Verified	Pass	Medium	Tested on Chrome, Firefox, Safari, and Edge
------	--------------------------------	---	----------	------	--------	---