

Day 5 – Testing, Error Handling, And Backend Integration Refinement

Testing and Error Resolution in Comforty

1. Frontend Testing

To ensure a smooth user experience, I rigorously test the UI and interactions of Comforty. This includes:

- **Responsiveness Testing:** Checking layouts on different screen sizes using Chrome DevTools.
- **Component Validation:** Ensuring all UI components render correctly in Next.js with Tailwind CSS.
- **Authentication Testing:** Verifying Clerk authentication flow, ensuring only authenticated users can access the website.

2. Backend and API Testing

The backend functionalities, including API integrations, undergo extensive testing:

- **Sanity CMS API:** Ensuring real-time inventory updates and content retrieval without errors.
- **Stripe API:** Checking order creation, successful payments, and error handling during checkout.

3. Debugging & Error Handling

Common issues and solutions encountered during development:

- **CORS Errors:** Resolved by correctly configuring API headers and Next.js API routes.
- **Authentication Issues:** Fixed redirect loops and session persistence with Clerk's configuration.
- **API Failures:** Implemented proper error handling and console debugging for failed API responses.

4. Tools Used for Debugging

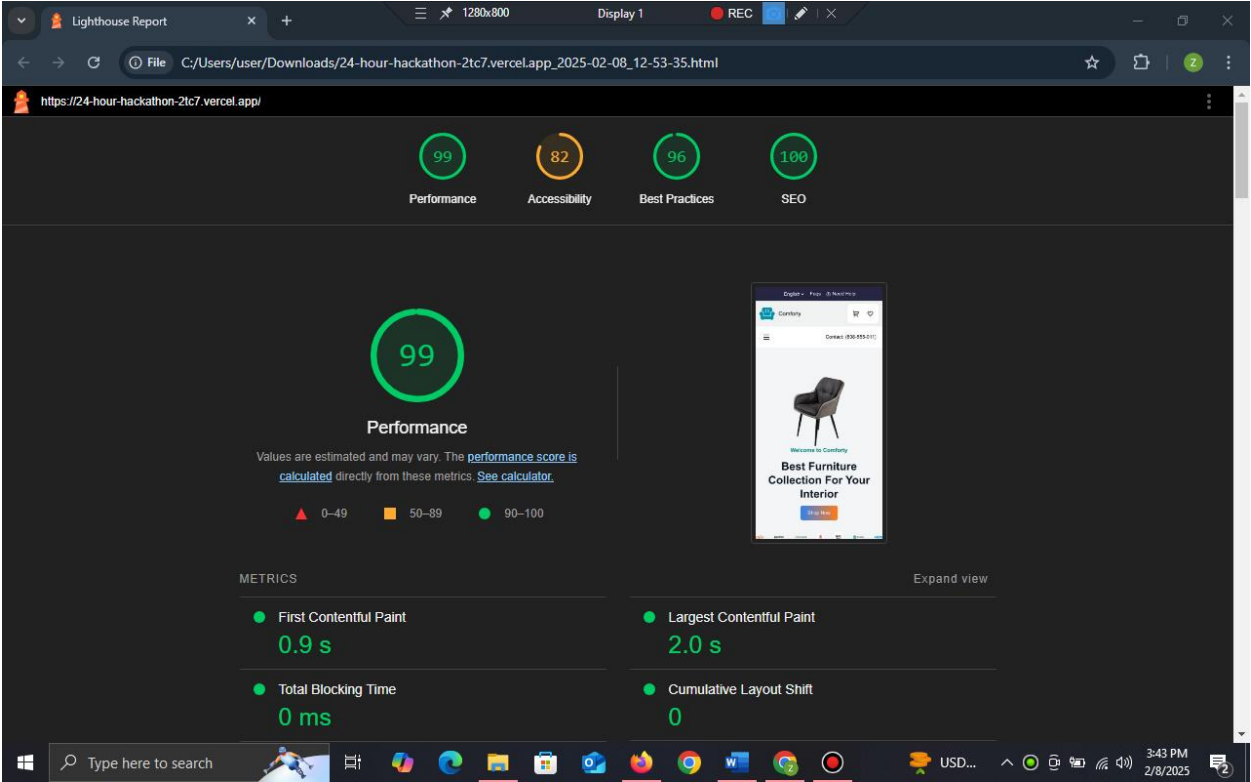
- **Next.js DevTools** – Debugging React components and state changes.
- **Postman** – Testing API endpoints before integration.
- **Console Logs & Error Tracking** – Using `console.log()`, `try/catch` blocks, and Next.js error handling.

By continuously testing and resolving errors, I ensure Comforty runs smoothly, providing a seamless shopping experience for users.

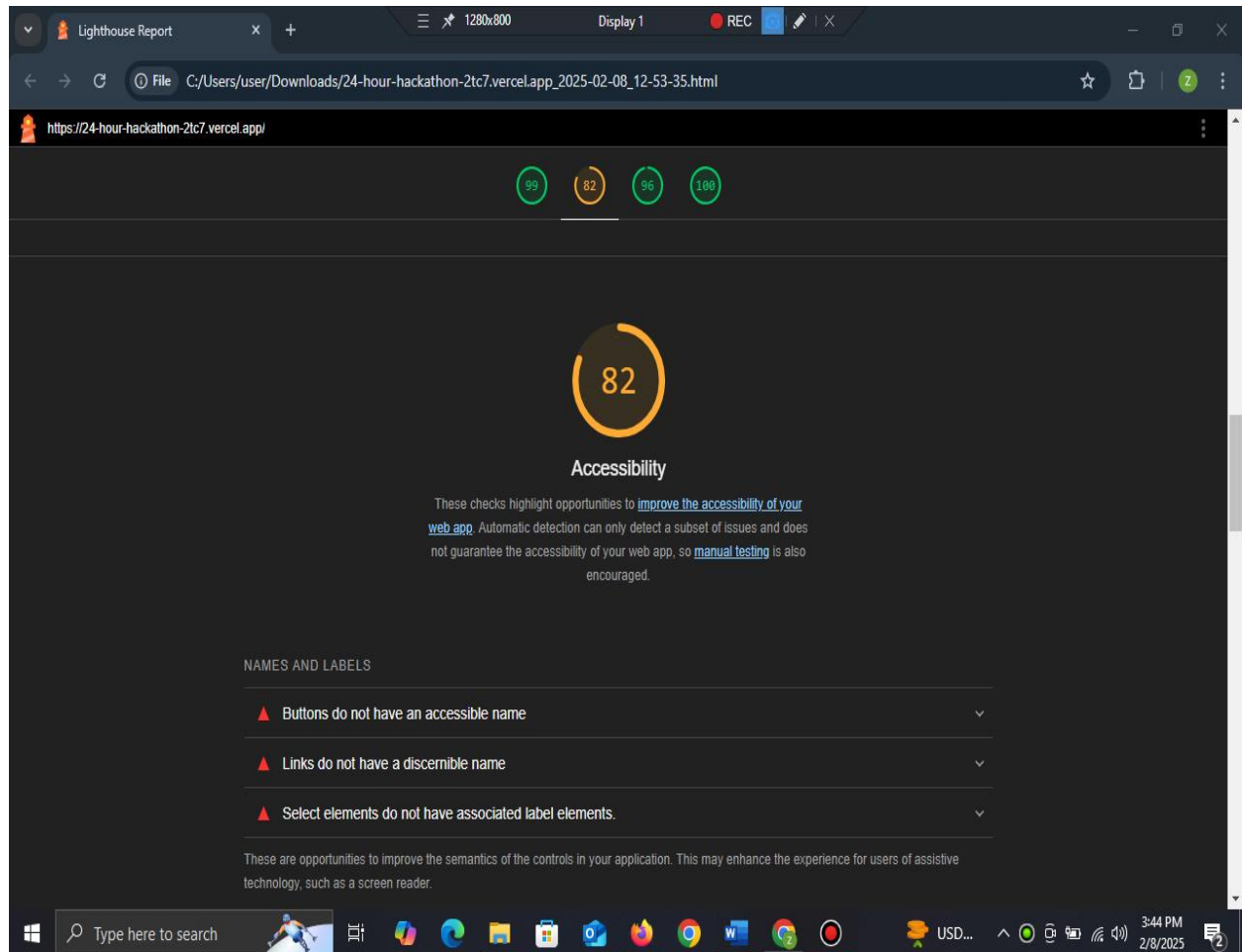
Test Case ID	Test Case Description	Test Steps	Expected Result
TC001	Validate product listing page	Open product page > Verify products	Products displayed correctly And smartly
TC002	Test API error handling	Disconnect API > Refresh page	Show fallback UI with error message
TC003	Check cart functionality	Add product to cart > Verify cart contents	Cart updates with added product
TC004	Ensure responsiveness on mobile	Resize browser window > Check layout	Layout adjusts properly to all screen size

Actual Result	Status	Severity Level	Assigned To	Remarks
Products displayed correctly	Passed	Low	-	No issues found
Error message shown	Passed	Medium	-	Handled perfectly
Cart updates as expected	Passed	High	-	Works as expected
Responsive layout working as intended	Passed	Medium	-	Test successful

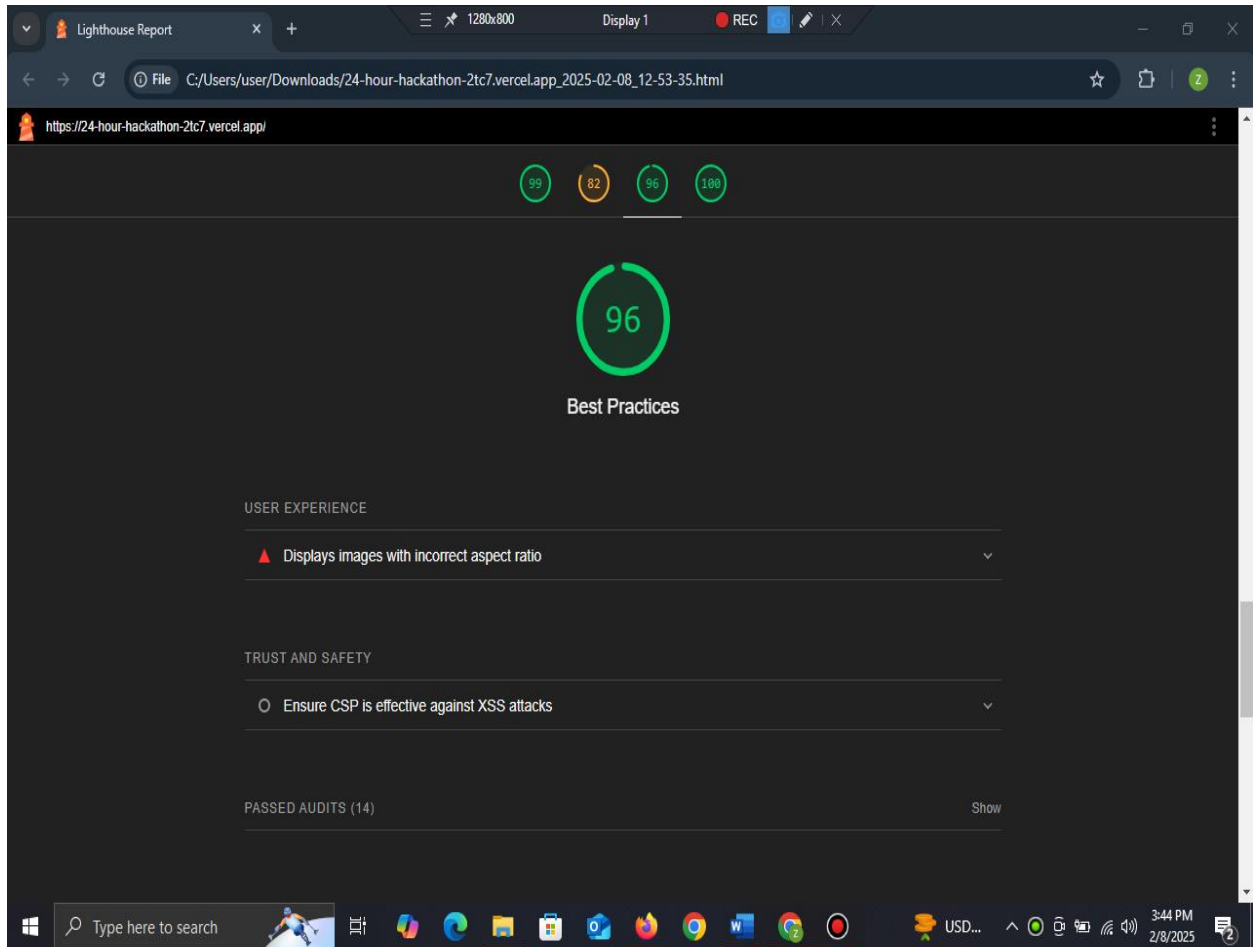
So, here’s some testing:



Accessibility



Best Practices



SEO

