Hackathon 3

TESTING, ERROR HANDLING, AND BACKEND INTEGRATION REFINEMENT

E-Commerce

Introduction

"Thoroughly test all components of "My Marketplace" optimize performance, and refine backend integration to ensure a smooth, error-free user experience that can efficiently handle customer-facing traffic during deployment"
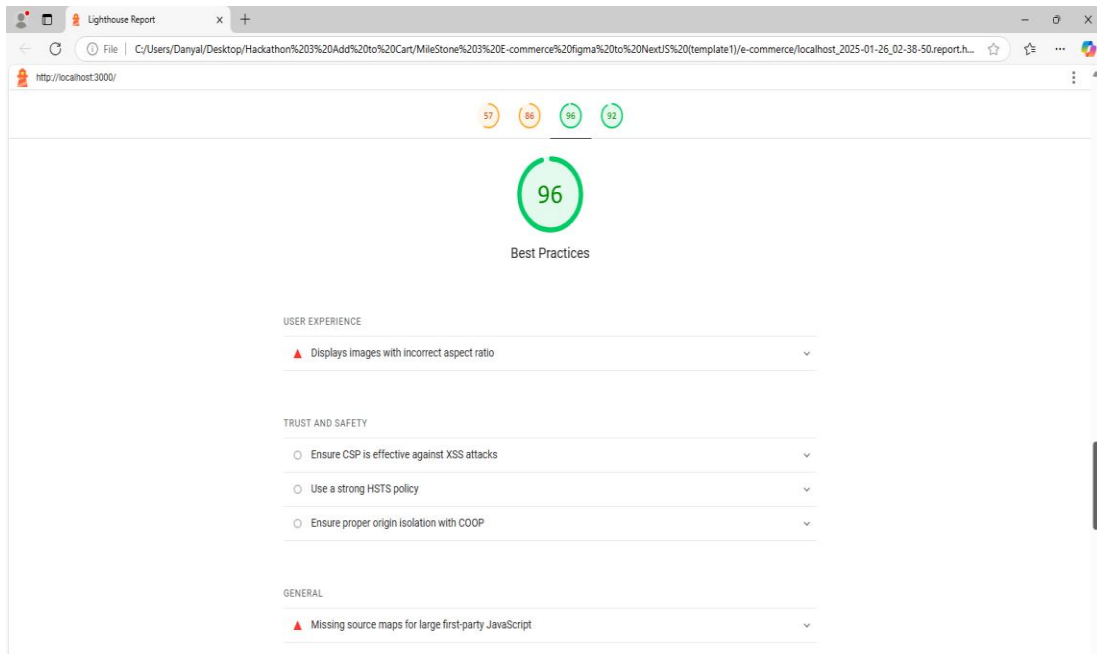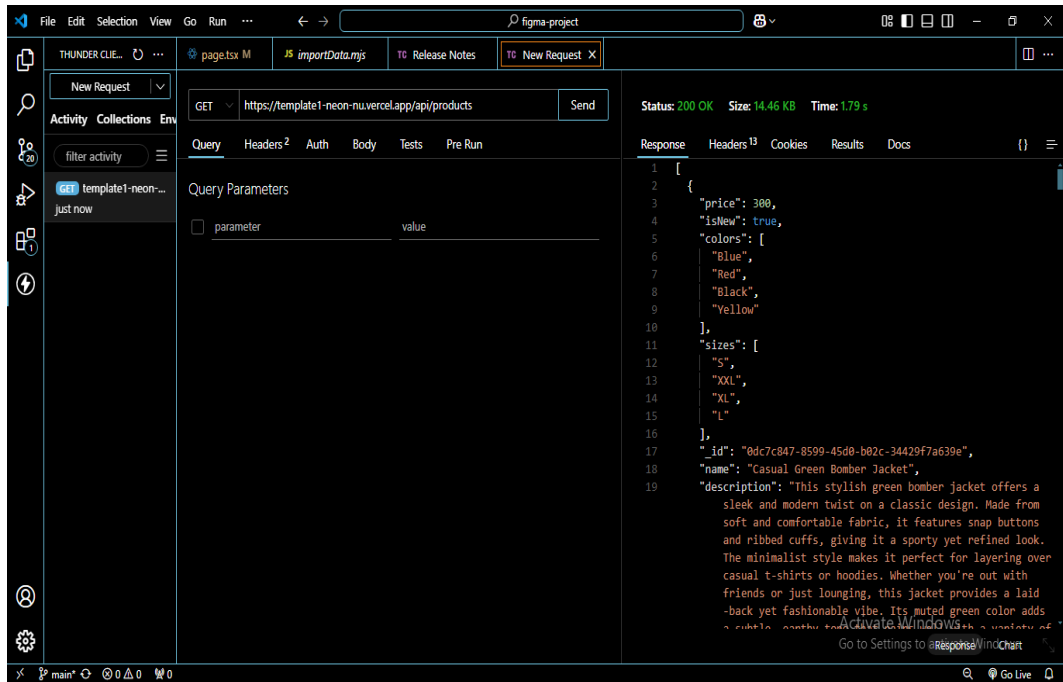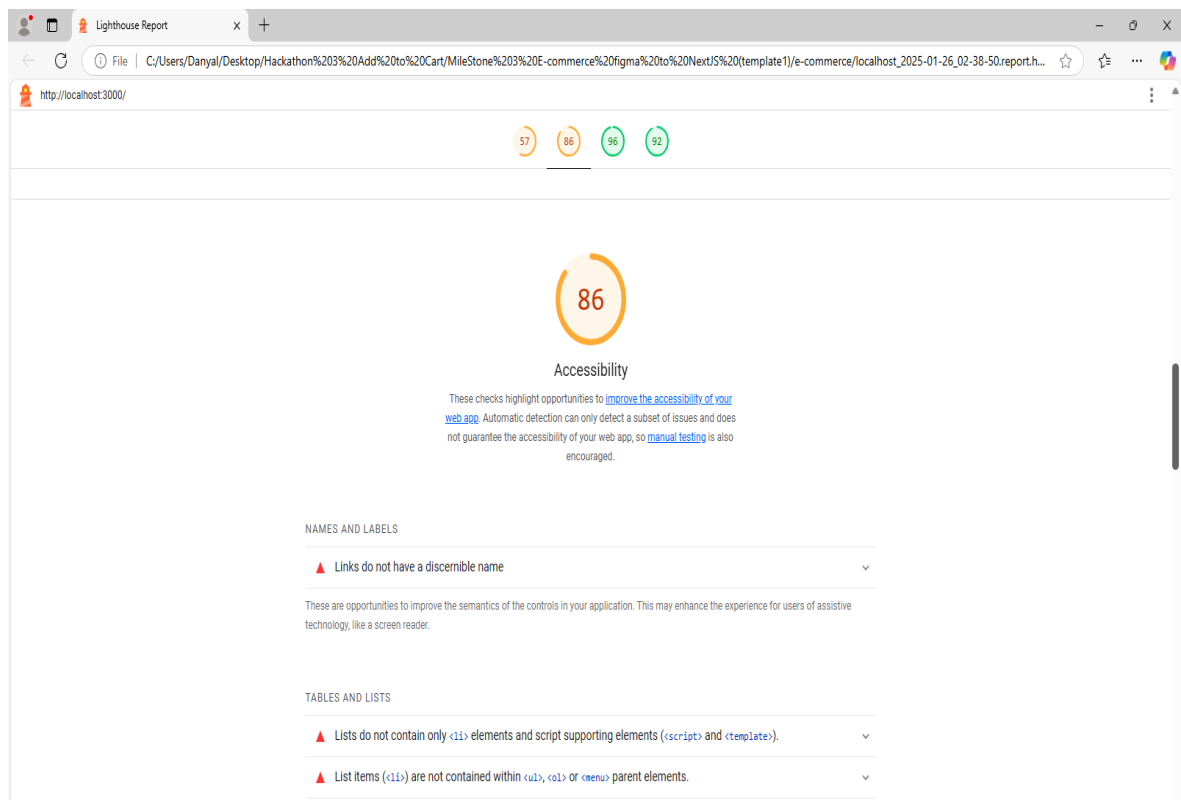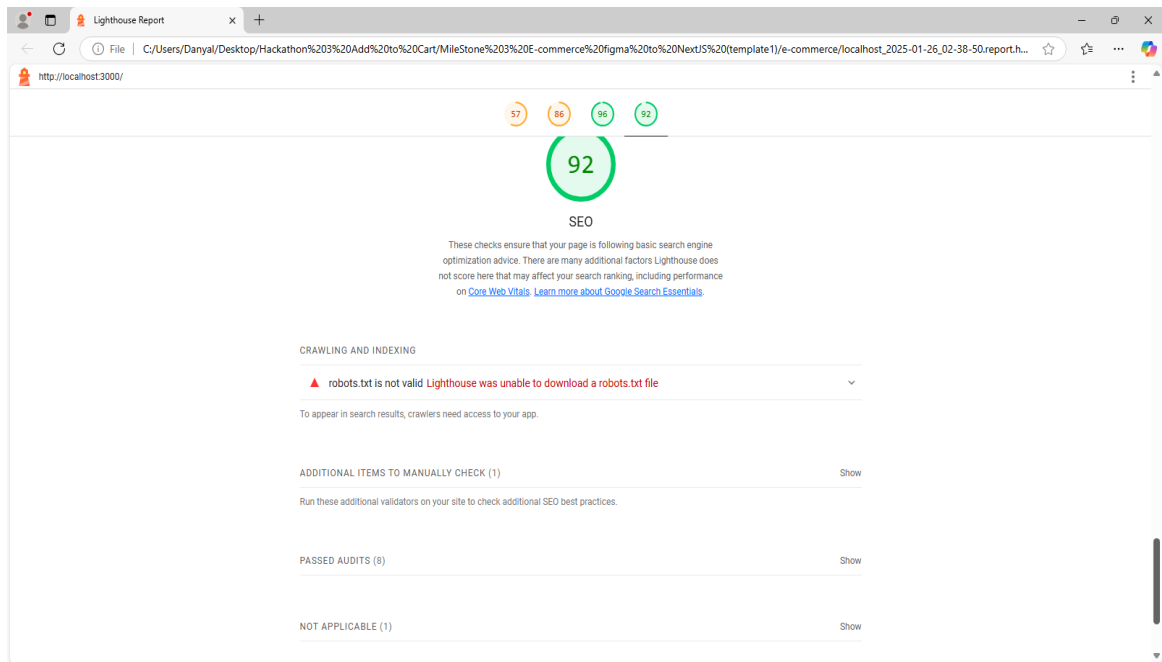
1. Functional Testing

2. Display of products (images, description, prices & reviews).
3. Cart operations: Add, update, and remove items from the cart.
4. Dynamic routing: Verify individual product detail pages' load correctly.
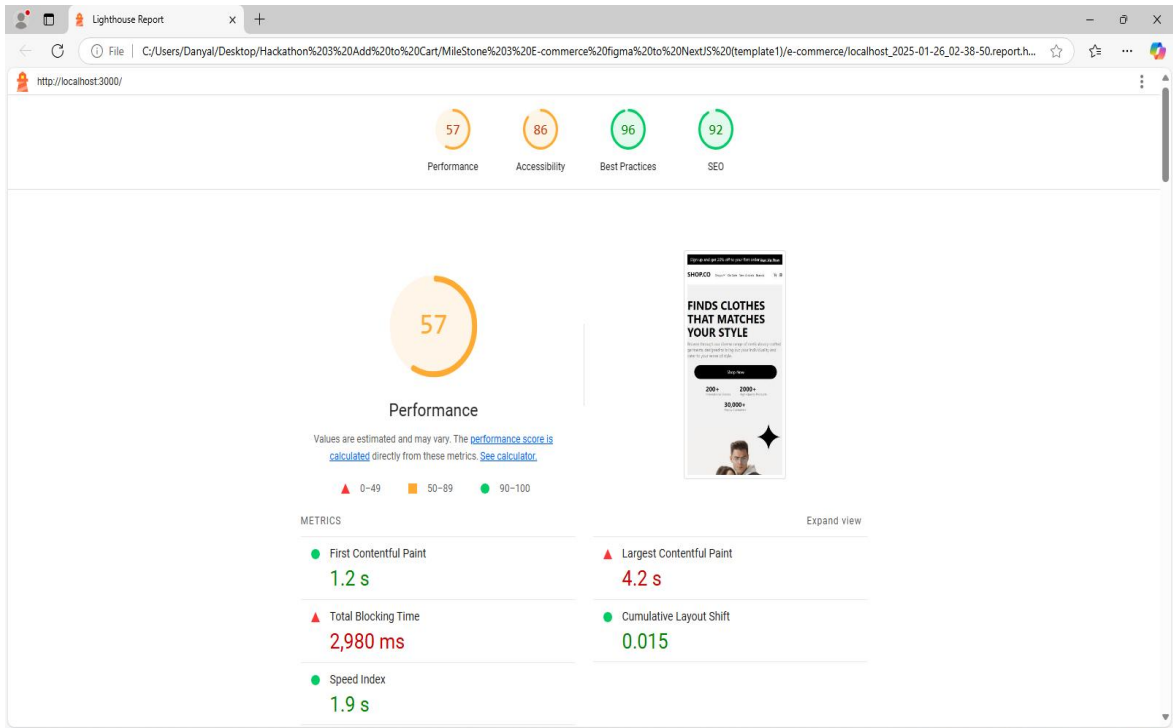5. Navigation Link.

2. Error Handling

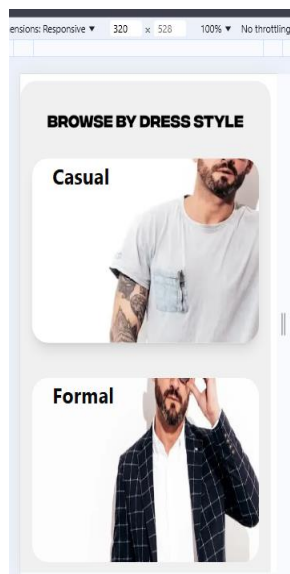Use try-catch blocks to handle APIs Error. Fallback UI: Display all items perfectly
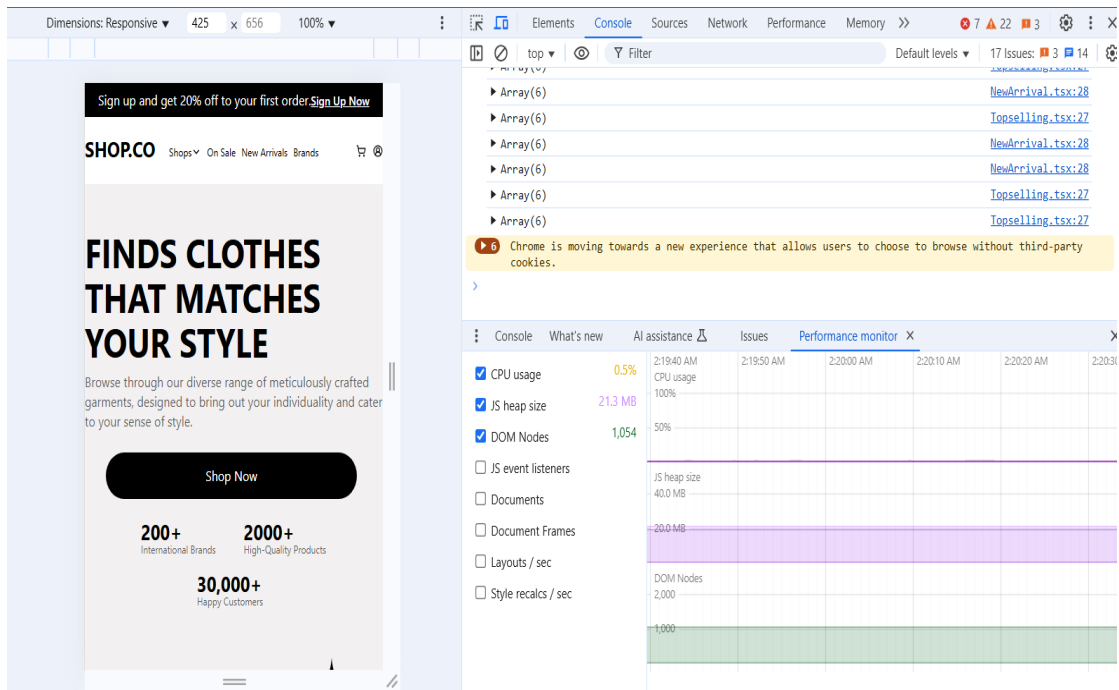
3. Performance Testing

SEO

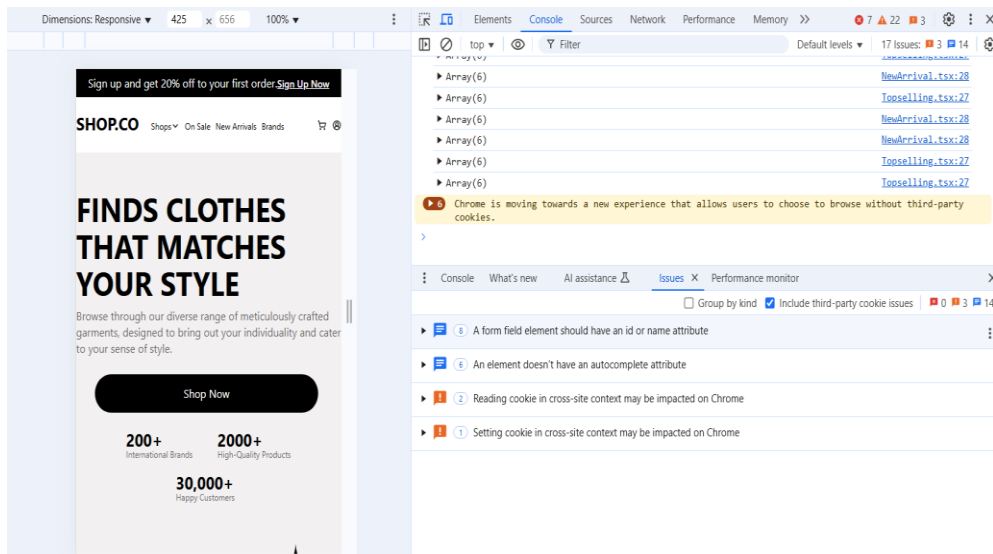These checks ensure that your page is following basic search engine optimization advice. There are many additional factors Lighthouse does not score here that may affect your search ranking, including performance on Core Web Vitals. Learn more about Google Search Essentials.

CRAWLING AND INDEXING

⚠ robots.txt is not valid Lighthouse was unable to download a robots.txt file

To appear in search results, crawlers need access to your app.

ADDITIONAL ITEMS TO MANUALLY CHECK (1)   Show

Run these additional validators on your site to check additional SEO best practices.

PASSED AUDITS (8)   Show

NOT APPLICABLE (1)   Show

Accessibility

These checks highlight opportunities to improve the accessibility of your web app. Automatic detection can only detect a subset of issues and does not guarantee the accessibility of your web app, so manual testing is also encouraged.

NAMES AND LABELS

⚠ Links do not have a discernible name

These are opportunities to improve the semantics of the controls in your application. This may enhance the experience for users of assistive technology, like a screen reader.

TABLES AND LISTS

⚠ Lists do not contain only `<li>` elements and script supporting elements (`<script>` and `<template>`).

⚠ List items (`<li>`) are not contained within `<ul>`, `<ol>` or `<menu>` parent elements.

[3]

Developer Resource

Performance Monitor

Issue

Testing Report (CSV Format)

Test Case ID	Test Case Description	Test Steps	Expected Result	Actual Result	Status	Severity Level	Assigned To	Remarks
TC001	Validate Product listing page	Open Product Page/check product grid	Products displayed correctly	Products Displayed correctly	passed	Low	-	All functional
TC002	Test API handling	API Successfully Working	Fallback UI shows "All products"	Fallback UI displayed	Passed	Medium	-	Error Handling found
TC003	Add To Cart	Add product to cart/ check out	Cart updates with added product	Cart updates as expected	passed	High	-	Works as expected
TC004	Search Functionality	Search for Products	Relevant Products are displayed	Correct result displayed	passed	Medium	-	Test Successfully
TC005	Mobile Responsiveness	Resize browser to mobile view	Layout adjusts for mobile view	Layout adapts perfectly	passed	Medium	-	Responsive On all tested devices

Conclusion

The e-commerce website is now thoroughly tested & optimized for real-world usage. Key functionalities have been validating, error handling, mechanisms implemented and performance optimized. Final deployment is recommended following minor fixes highlighted during tested.