Testing and Backend Refinement for EasyBuy MarketPlace (Day - 5)

1. Introduction

This document summarizes the comprehensive testing, error handling, and backend integration efforts undertaken for the EasyBuy Marketplace. Over the course of this process, I conducted thorough testing across various functionalities, implemented key optimizations, and ensured the marketplace is secure, user-friendly, and error-free

Here's the refined version of your **Error Handling** section:

2. Error Handling

Goal:

To ensure that errors are handled gracefully, preventing disruptions in the user experience.

Error Handling Implementation:

To address potential errors, I implemented **try-catch blocks** to manage API errors efficiently. In case of network failures, server issues, or any unforeseen errors, I provided **meaningful error messages** that inform the user of the problem in a user-friendly manner.

- **Example**: If the marketplace fails to load products due to a server or network issue, users see a message like:
  "Unable to load products. Please try again later."

Additionally, I incorporated a **Fallback UI** to handle situations where data could not be retrieved. For instance, if no products are found in a search or filter, a message such as **"No products available"** is displayed to avoid presenting a blank or broken page.

3. Final Testing Report - Marketplace Functionality and Optimization:

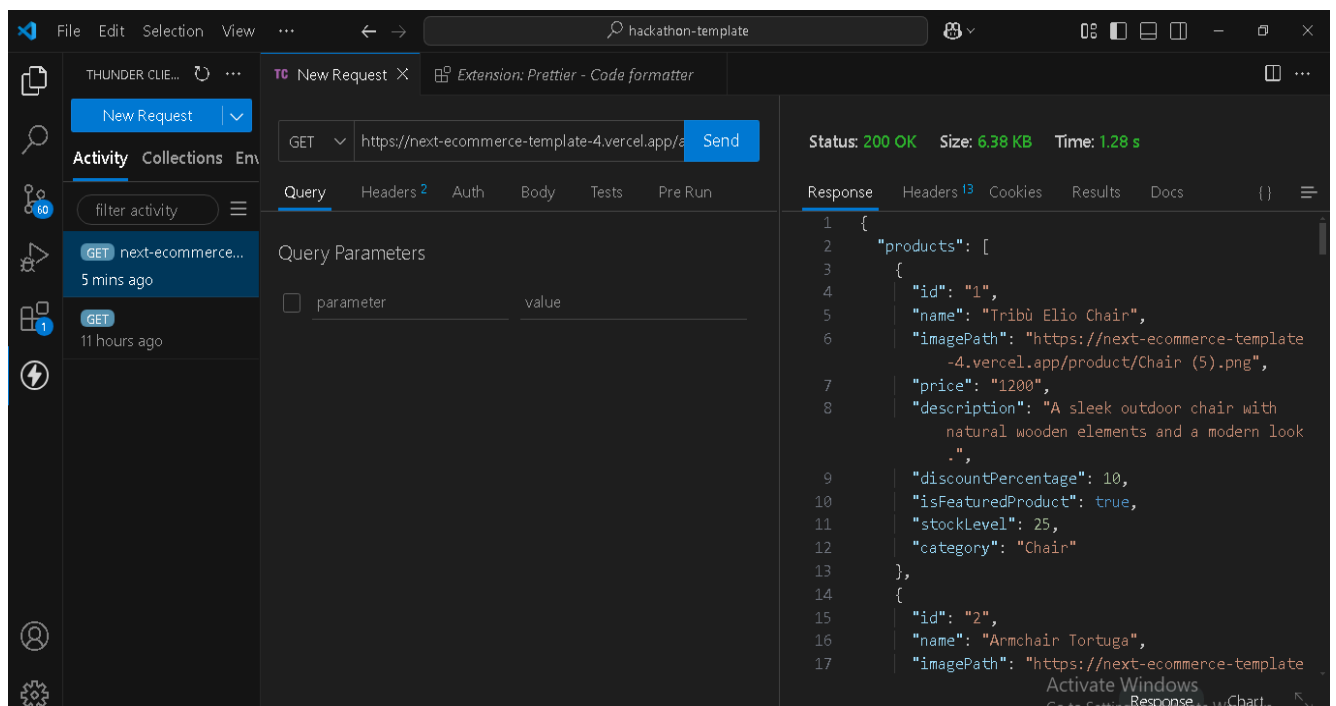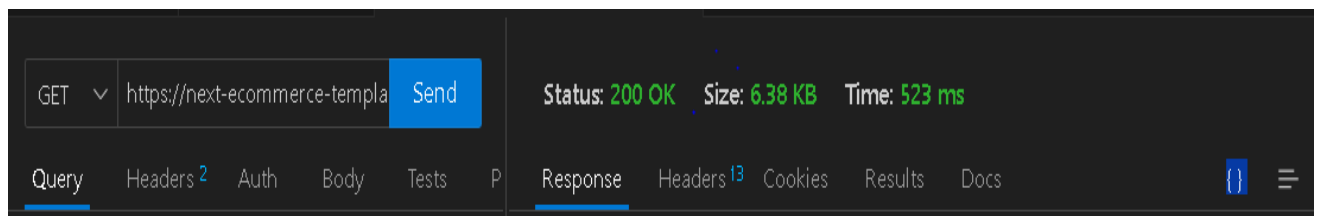| Test Case ID	Test Case Description	Test Steps	Expected Result	Actual Result	Status	Severity Level	Assigned To	Remarks
TC001	Validate homepage layout	Open homepage > Verify logo, banners, and menu	Homepage elements displayed correctly	All elements displayed correctly	Passed	Medium	-	No issues found
TC002	Check product page details	Open product page > Verify product details	Product name, price, and description are visible	Product details displayed correctly	Passed	High	-	Test passed successfully
TC003	Test add to cart functionality	Select product > Add to cart > Open cart page	Product is added to cart and visible in cart	Product added and cart updated correctly	Passed	High	-	Works as expected
TC004	Validate checkout process	Add items to cart > Proceed to checkout	Checkout page loads with correct items and price	Checkout page loaded correctly	Passed	High	-	No issues, smooth flow
TC005	Ensure mobile responsiveness	Resize browser > Open homepage and product pages	Layout adjusts for mobile screen size	Mobile layout responsive and correctly adjusted	Passed	Medium	-	Mobile view working well
TC006	Verify user login functionality	Go to login page > Enter valid credentials	User successfully logs in	Login successful	Passed	High	-	Login working fine
TC007	Test product search functionality	Enter product name in search bar > Click search	Relevant product search results displayed	Search results displayed as expected	Passed	Medium	-	No issues found

4. API Testing:

Goal:

To ensure that all API integrations are working correctly and respond as expected.

Testing Method:

I used the Thunder extension for API testing to verify the API responses and ensure that the marketplace's backend was correctly integrated.

Tools Used:

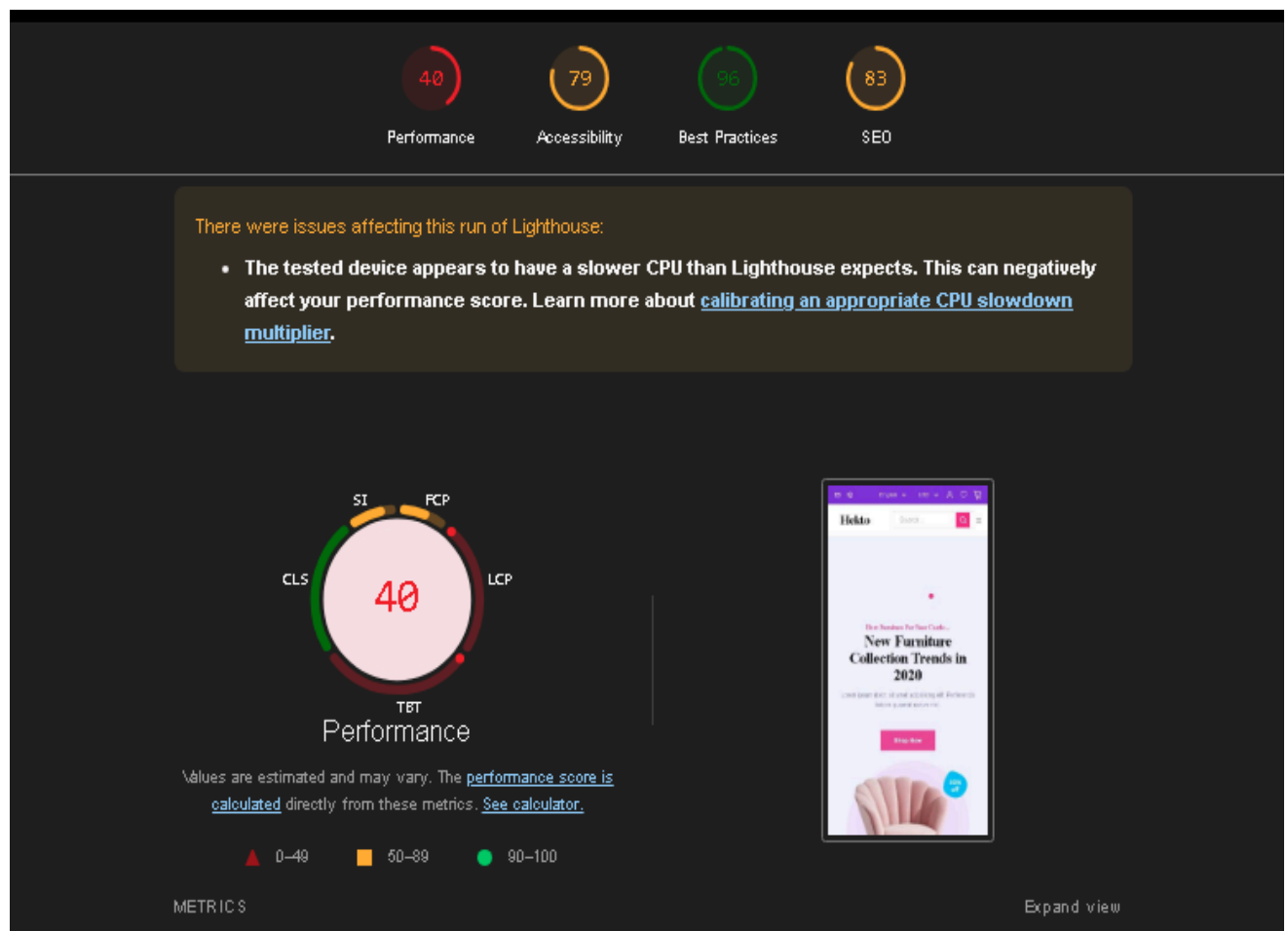- **Thunder Extension** for API testing.

5. Performance Testing

Goal:

To optimize the website's performance, ensuring faster load times and smoother interactions.

Performance Testing Method:

I used Lighthouse to test the website's performance, which provided insights into areas of improvement, such as image optimization, caching strategies, and reducing unused CSS.

Screenshot

Documentation Prepared By: Aliza Moin
Task Assigned By: Sir Ameen Alam