

# Day-05 TESTING, ERROR HANDLING & BACKEND INTEGRATION:

Made By Mahad Khan:

On Day-05, I focused on testing, error handling, and improving the functionality of my website to ensure a seamless user experience. Below is an overview of the key tasks completed:

## 1. Functional Testing:

- Validated core features such as:
  - Responsiveness:** Ensured the website works smoothly across devices.
  - Product Listing Page:** Verified accurate display of products.
  - Details Page:** Confirmed product details load correctly.
  - Add to Cart Page:** Ensured items could be added, updated, and removed.
  - Checkout Page:** Tested the final purchase workflow

## Testing Report:

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 > Verify products	Products displayed correctly	Products displayed correctly	Passed	Medium	-	No issues found
TC002	Test API error handling	Disconnect API > Refresh page	Show fallback UI with error message	Error message shown	Passed	Medium	-	Handled gracefully
TC003	Check cart functionality	Add product to cart > Verify cart contents	Cart updates with added product	Cart updates as expected	Passed	High	-	Works well
TC004	Ensure responsiveness on mobile	Resize browser window > Check layout	Layout adjusts properly to screen size	Responsive layout working as intended	Passed	High	-	Test successful

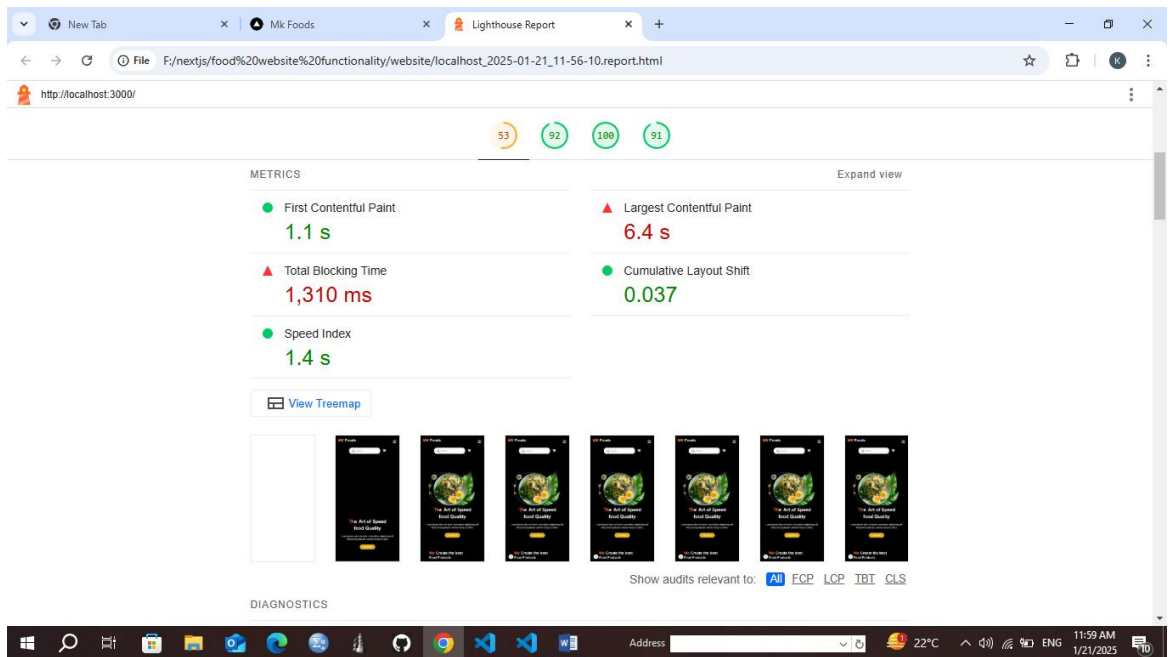
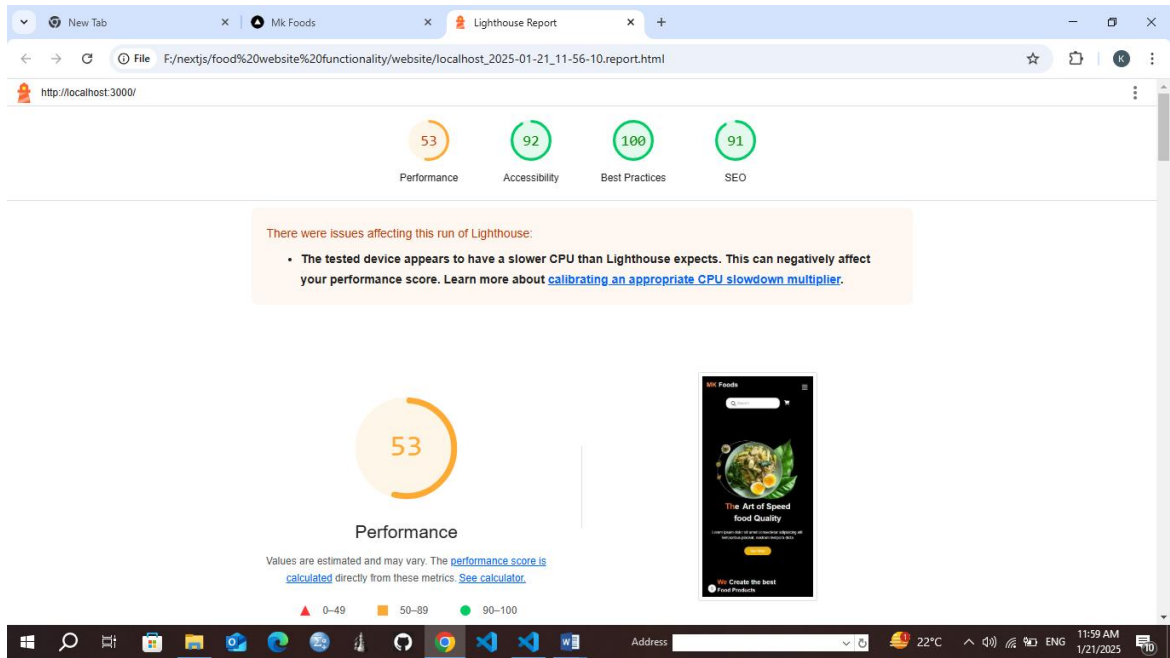
## 2. Error Handling:

- Implemented robust error messages for scenarios such as:
  - **Invalid or Missing Data:** Used fallback UI elements like "No products available."

Providing users with clear feedback and improving the system stability.

## 3. Performance Testing:

- Used **Lighthouse** to analyze and optimize website performance.



New Tab x Mk Foods x Lighthouse Report x +

File F:/nextjs/food%20website%20functionality/website/localhost\_2025-01-21\_11-56-10.report.html

http://localhost:3000/

53 92 100 91

PASSED AUDITS (22) Show

92

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.

CONTRAST

▲ Background and foreground colors do not have a sufficient contrast ratio.

These are opportunities to improve the legibility of your content.

Windows taskbar: 22°C, 11:59 AM 1/21/2025

New Tab x Mk Foods x Lighthouse Report x +

File F:/nextjs/food%20website%20functionality/website/localhost\_2025-01-21\_11-56-10.report.html

http://localhost:3000/

53 92 100 91

NOT APPLICABLE (31) Show

100

Best Practices

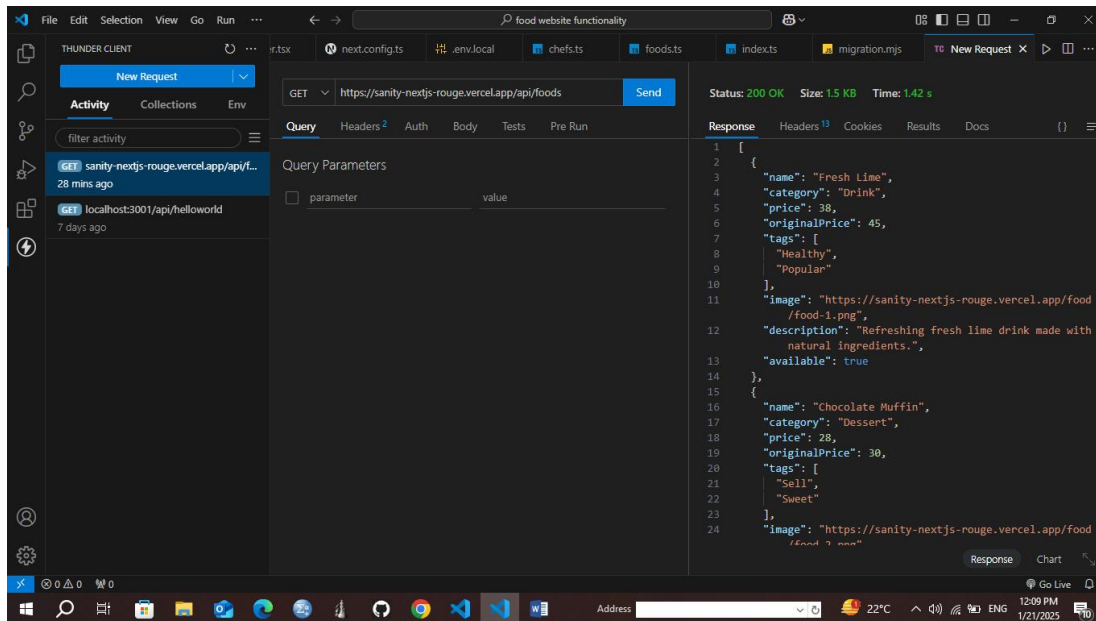
TRUST AND SAFETY

- Ensure CSP is effective against XSS attacks
- Use a strong HSTS policy
- Ensure proper origin isolation with COOP

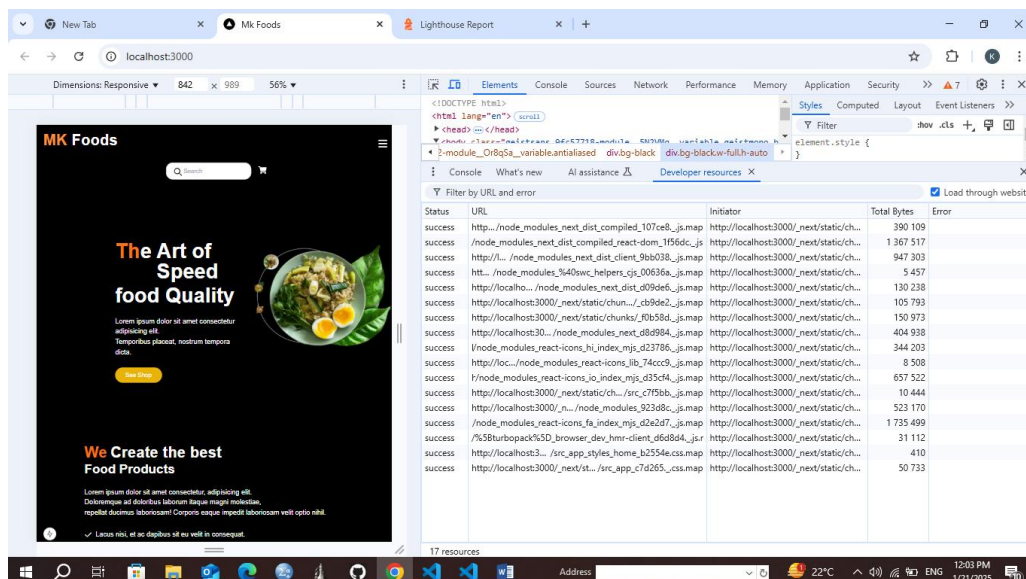
Windows taskbar: 22°C, 11:59 AM 1/21/2025

## 4. API Testing:

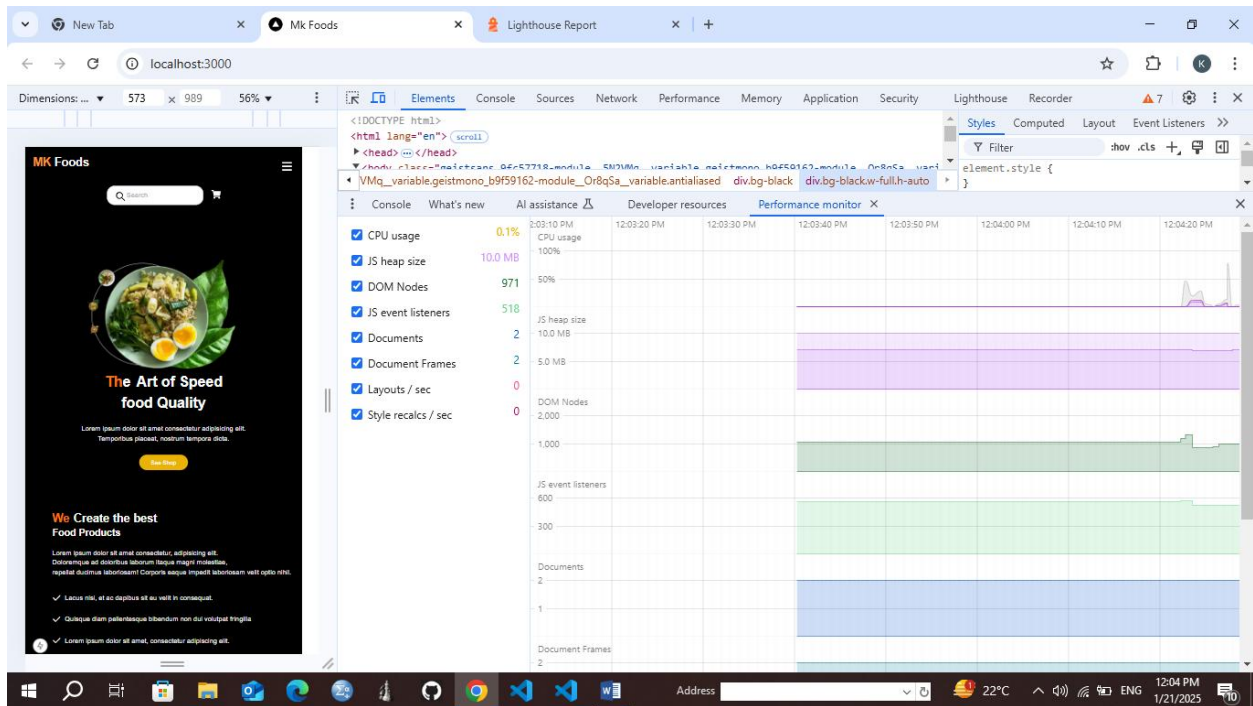
- Verified API endpoints using tools like **Postman** and **Thunderclient**.
- Handled edge cases, such as network failures or invalid data, with appropriate error messages.



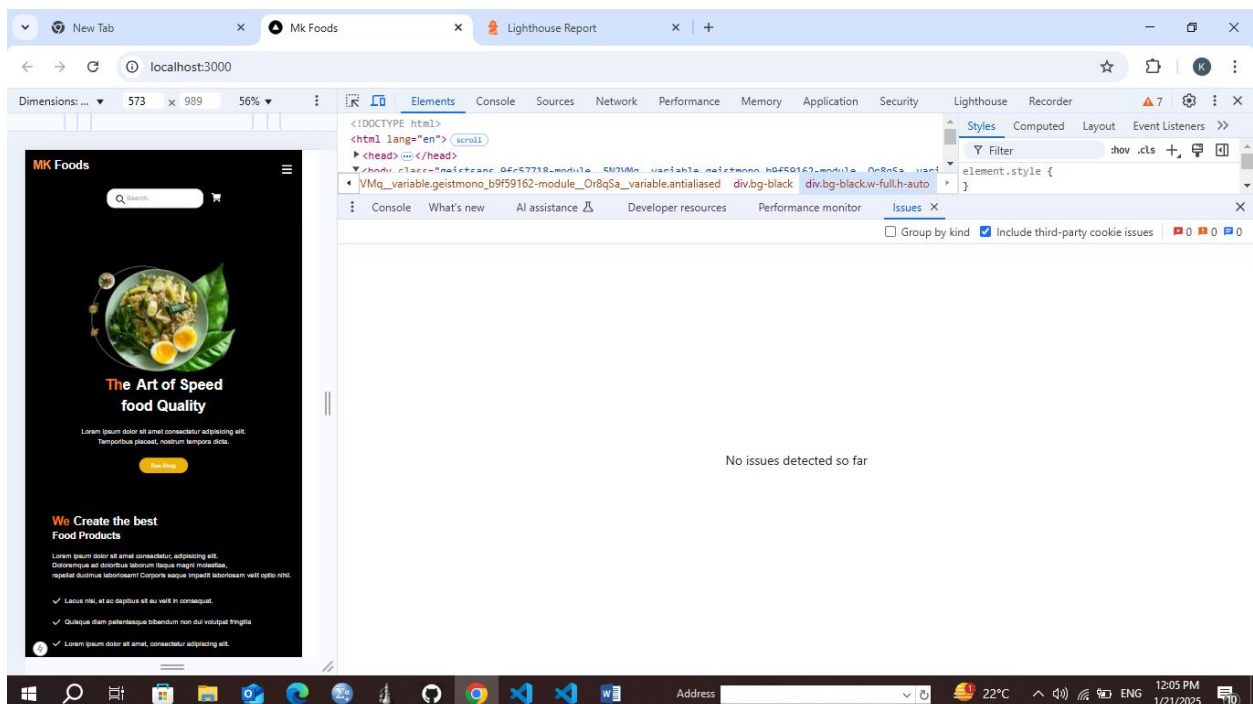
## Developer Resources:



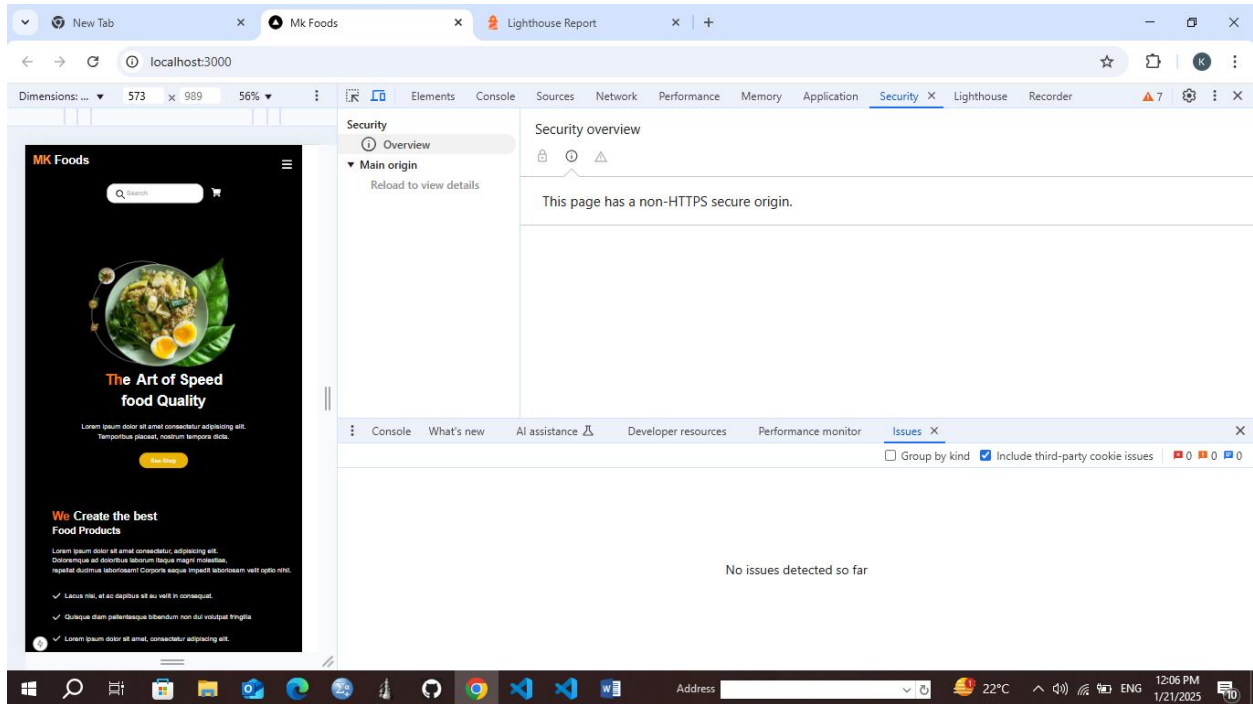
# Performance Monitor:



# Issues:



# Security:



On completing Day-05, the website is now equipped with robust functionality, clear error handling mechanisms, and optimized performance. All major components have been thoroughly tested, ensuring a seamless user experience. Moving forward, the focus will be on deploying the website and gathering user feedback to identify further enhancements.

## END OF THE DAY – 05.

## Made By Mahad Khan.