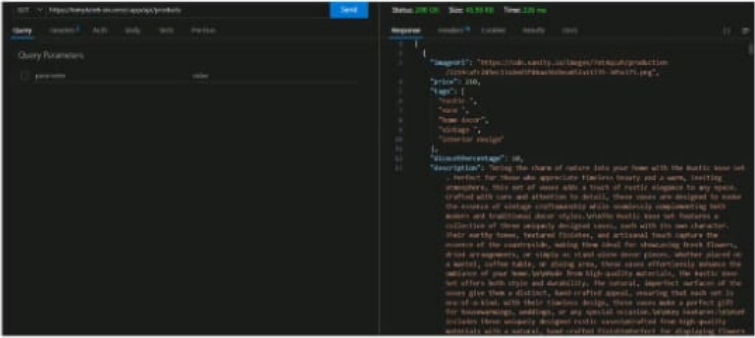


## Double Check from Postmar

There is no error handling required as the API data is being received correctly. I have also verified it using Postman and Thunder Client.



### CSV Testing Testing Reports

Test Case ID	Test Description	Steps	Expected	Actual	Status	Severity	Assigned	Remarks
TC001	Verify Product Page	Open product page	Page displays product details	Displayed correctly	Passed	Low	Ameen	Success
TC002	Add to Cart	Click 'Add to Cart'	Product added to cart	Added	Passed	Low	Ameen	Success
TC003	Price Display	Open product page	Price visible correctly	Visible	Passed	Low	Ameen	Success
TC004	Empty Cart	Go to cart	Cart shows '0 items'	Shows '0 items'	Passed	Low	Ameen	Success
TC005	Add Multiple to Cart	Add 2 products	Cart shows 2 items	2 items	Passed	Low	Ameen	Success
TC006	Checkout Process	Proceed to checkout	Checkout works smoothly	Completed	Passed	High	Ameen	Success

### Conclusion

- 1.Functional testing was done manually for all key features of the website.
- 2.Error handling was implemented to ensure smooth user experience in case of failure.
- 3.I optimized performance for faster load times and improved SEO.
- 4.Cross-browser and device testing was performed to ensure compatibility across multiple platforms.
- 5.Security measures were put in place to protect user data and prevent common vulnerabilities.

## Testing and Backend Refinement

### Overview

I have completed the final rounds of testing, error handling, and backend integration refinements for the Comforty Marketplace. This platform specializes in offering a wide variety of stools, chairs, and sofas. The objective was to ensure that all website components functioned smoothly, were optimized for speed, and were ready for real-world deployment.

### Overview of Tasks Completed

#### Functional Testing

I thoroughly tested the core features of the website, including product pages for stools, chairs, and sofas, category listings, cart operations, and user profile management. I manually ensured that each page is loading correctly, that products are displayed with their details, and that users can navigate between pages without issues.

#### Error Handling

I added error messages for situations where something goes wrong, like network failures or missing data. For example, if a product is not found or a category has no products, I made sure the website shows a clear message to the user: "No products available" or similar. I implemented fallback UI elements to provide a better experience if the site encounters problems, ensuring users know something went wrong.

#### Backend Integration

I worked on integrating the backend so the product data is fetched properly from the server. If the server fails, the website shows a fallback UI that handles the error smoothly without crashing. I also ensured that the API correctly serves product data, so users see the latest updates to the product catalog.

## Trust and Safety

Ensure CSP is effective against XSS attacks

Use a strong HSTS policy

Ensure proper origin isolation with COOP

General

100

Best Practices

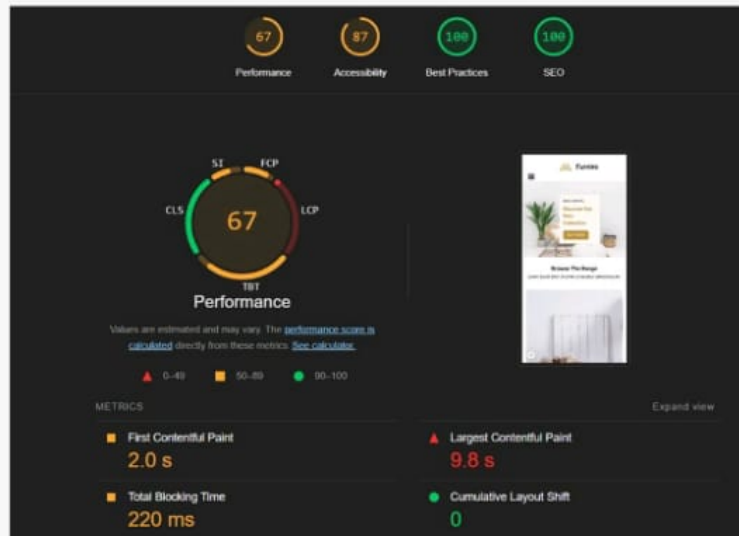
### TRUST AND SAFETY

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

### GENERAL

- ☐ Missing source maps for large first-party JavaScript

## Performance



## 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.

