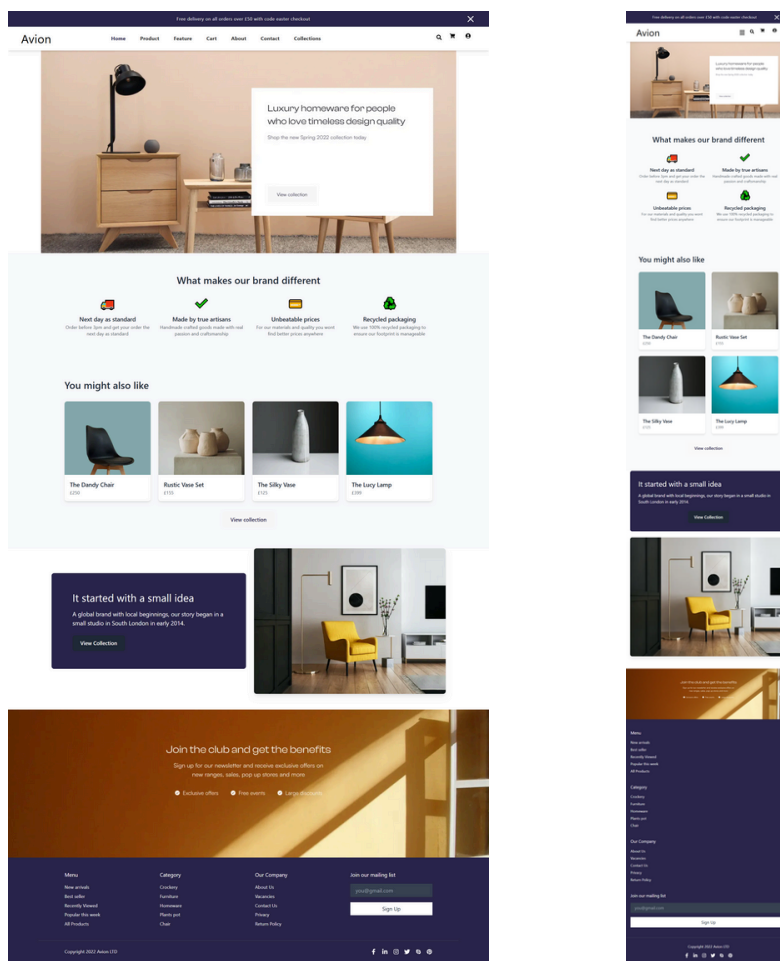


Testing and Backend Refinement

Responsiveness Testing

I ensured responsiveness of Website, it is completely responsive on every devices



Error handling:

To handle errors effectively, I would use try-catch blocks to manage network failures and unexpected server errors. I would validate API responses to check for missing or invalid data and ensure fallback UI is shown to users with meaningful messages, such as 'No products available' when the data is empty. Here's an example implementation.

1. Network Failures:

I've used a try-catch block to handle any potential network errors, such as no internet connectivity or a timeout. If the fetch request fails, the catch block sets an appropriate error message ex: `setError("Failed to fetch product details.");`

2. Invalid or Missing Data:

After receiving the response, I check its status using `response.ok`. If the response indicates an error (e.g., 404 or 400), I extract the error message from the API response or default to a meaningful fallback: ex: `setError(data.error || "Product not found");`

3. Fallback UI:

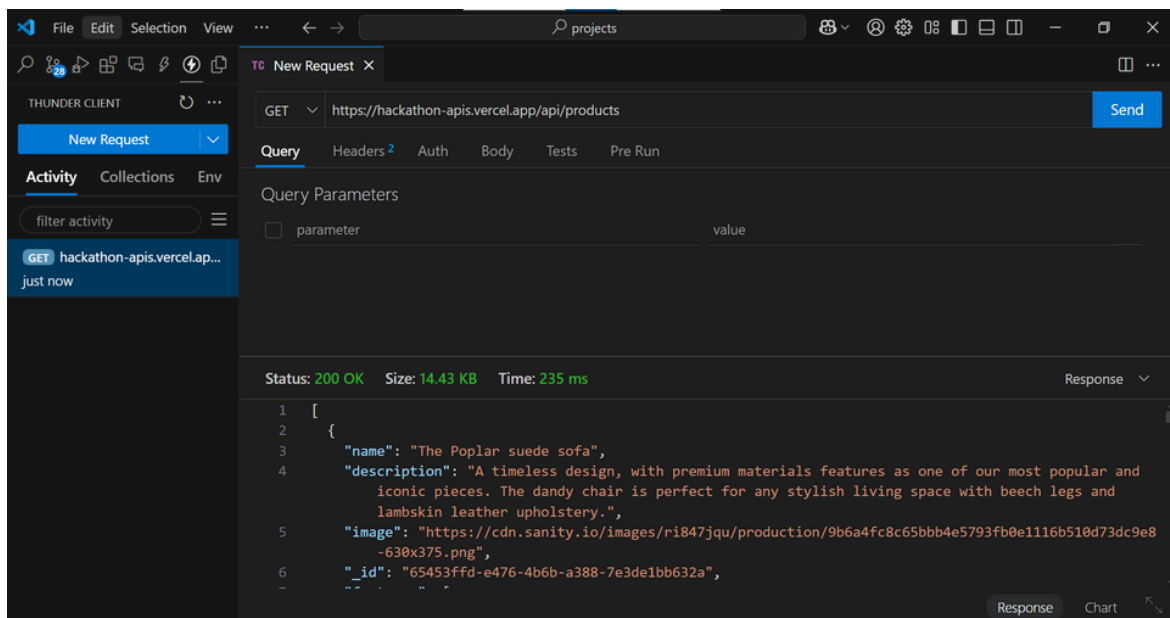
While the data is being fetched, a loading state (`setLoading`) is managed to display a spinner or loading indicator. If an error occurs, a fallback UI is shown with a message to inform the user.



```
1  useEffect(() => {  
2    async function fetchProducts() {  
3      try {  
4        const response = await fetch('/api/products');  
5        if (!response.ok) {  
6          throw new Error(`Error: ${response.statusText}`);  
7        }  
8        const data = await response.json();  
9        setProducts(data);  
10     } catch (err: any) {  
11       setError(err);  
12     } finally {  
13       setLoading(false);  
14     }  
15   }  
16  
17   fetchProducts();  
18 }, []);  
19
```

Api response test:

All API responses were tested thoroughly using Thunder Client. The tests covered various scenarios, and all APIs returned a successful response with a status of 200 OK. Below are some examples of the tested



Test Case Report

	A	B	C	D	E	F	G	H	I	J
1	Test Cases ID	Test Case Description	Test Steps	Expected Result	Actual Result	Status	Severty Level	Assigned to	Remarks	
2	TC001	Validate Product listing page	Open Product page> verify	Product displayed correctly	Product displayed correctly	Passed	High	-	No issues found	
3	TC002	Test Api Error handling	Disconect api> refresh page	Show fallback UI with error message	Show fallback UI with error message	Paased	High	-	Handle successfully	
4	TC003	Check Cart functionality	Add product to cart > verify	Card updates with products	same as expected	Passed	High	-	works as expected	
5	TC004	Ensure responsiveness on mobile	Resize browser window > Verify	100% responsive and adoptable	100% same as expected	Passed	High	-	Successful	
6	TC005	Authentication	signup > create account > login	100% user can login signup, verifyemail	same as expected	Passed	High	-	Successful	
7	TC006	Search products	search > result	show products on keywords and tags	products showing accordingly	Passed	Medium	-	works perfectly	
8										
9										

Product Listing Page (TC001): All products were displayed correctly with no visible issues.

API Error Handling (TC002): Successfully handled API disconnection by displaying a fallback UI and error message.

Cart Functionality (TC003): Cart updated accurately with selected products, confirming functionality is intact.

Responsiveness (TC004): The application is 100% responsive and adaptable to different screen sizes.

Authentication (TC005): User authentication, including signup, email verification, and login, worked as expected.

Search Products (TC006): The search functionality correctly displayed relevant products based on keywords and tags

Performance Optimization:

Analyze Performance Tool Used:

Lighthouse analysis was conducted to identify performance metrics and potential areas for improvement.

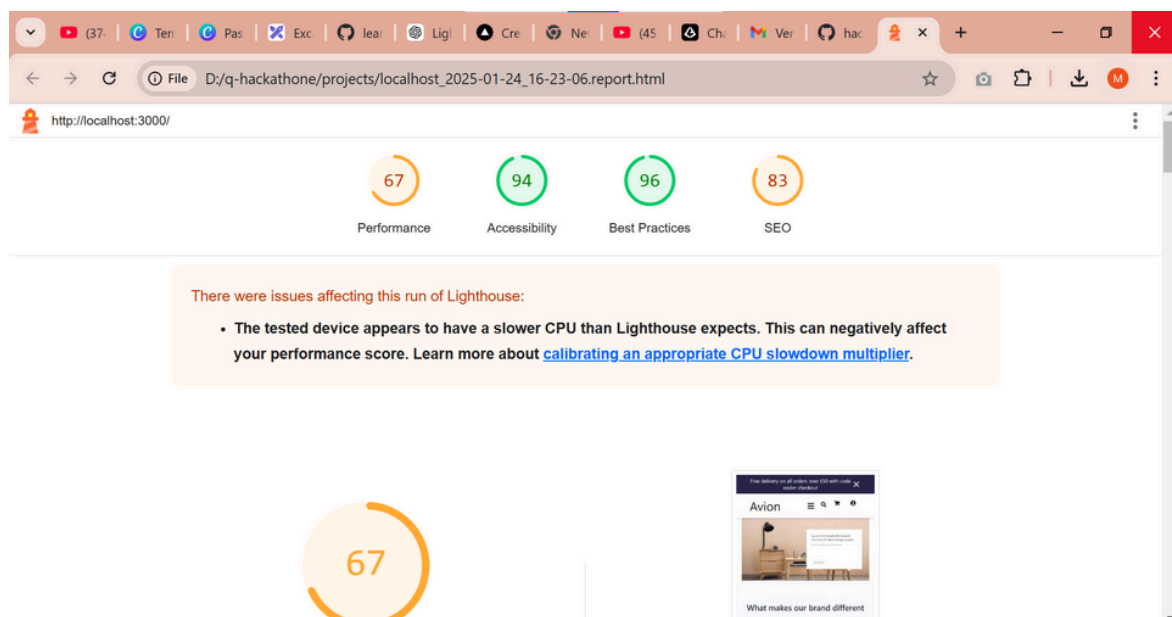
Results:

Performance: 59 - Indicates room for significant improvement, particularly in areas like reducing unused CSS and optimizing JavaScript bundles.

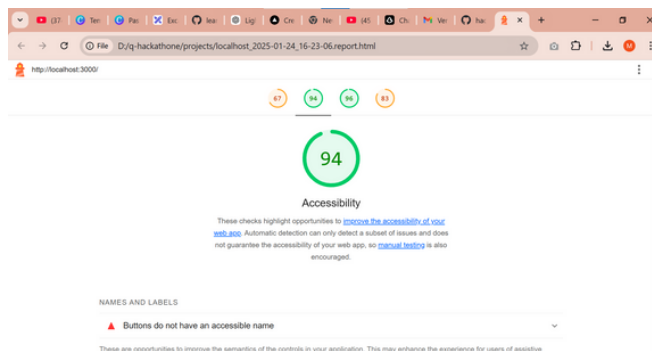
Accessibility: 82 - The website is reasonably accessible, but minor adjustments can enhance the user experience further.

Best Practices: 96 - Strong adherence to web development best practices with minimal issues.

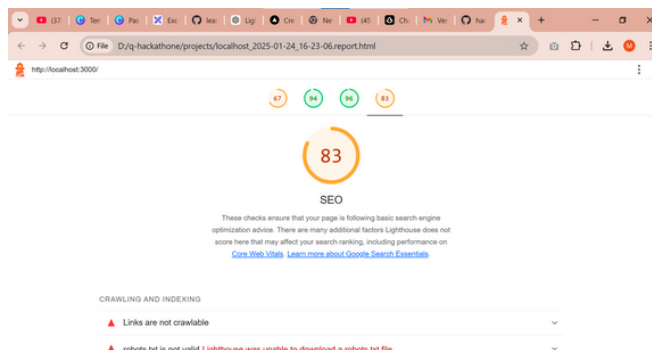
SEO: 92 - Excellent SEO practices, ensuring good visibility and searchability.



Accessibility:



SEO:



Best Practices:

