

**Day 05:**

# Testing, Error Handling and Backend Integration Refinement

By: Kulsoom Imran

## Functional Testing

Test Case ID	Test Case	Steps Taken	Outcome	Status
TC001	Display products listing page	Open the product listing page	All products appears with correct details	Passed
TC002	Image Rendering	Verify product image loads correctly	All images are displayed without any distortion	Passed
TC003	Category Filter	Select a category from the filter panel.	Displays only products from the selected category.	Passed
TC004	Price Filter	To ensure the price filter correctly sorts products from Low to High and High to Low.	For <b>Low to High</b> the cheapest product appears first and for High to Low the most expensive product appears first.	Passed
TC005	Status Filter	Select a status from the filter panel.	Displays only products from the selected status.	Passed
TC006	Search product	Enter keyword in search bar.	Displays products matching the search criteria.	Passed
TC007	Add products to cart	"Add products to cart, update quantity, and calculate totals."	Products are added and removed correctly; totals update accurately.	Passed
TC008	Navigate to Product Details Page	Navigate to the specific product details page from the product listing page.	Correct product details page load on the next page	Passed

TC009	Add products to the wishlist	Add products to wishlist, reloads page and verify ”	Wishlist items persist across sessions.	Passed
TC010	Register a user and log in with credentials.	”Fill out registration form, attempt login and get access to the registered account	Failed to store user information in backend	Failed
TC011	Related Product	Show related products to the opened product	Successfully render related products	Passed

## Error Handling

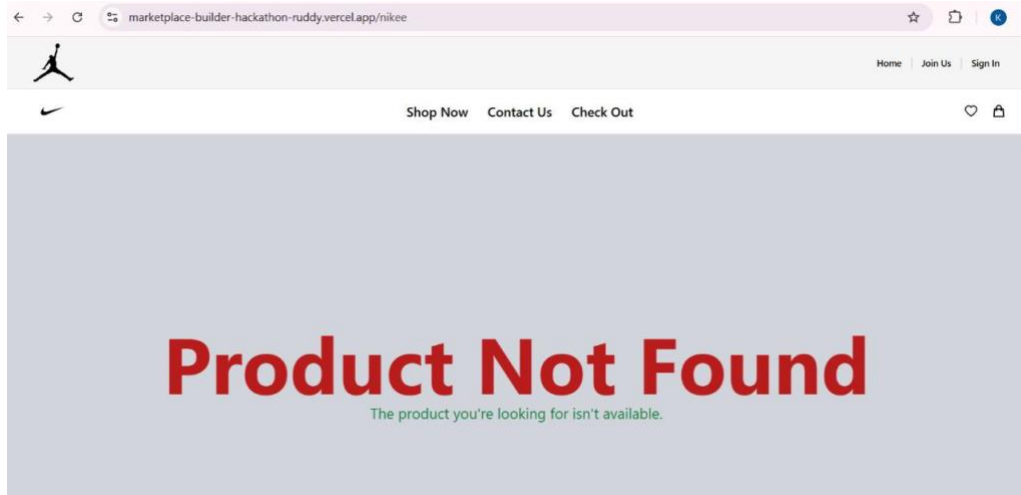
```
export default function Notfound(){
  return(
    <div className="h-screen flex flex-col items-center justify-center bg-gray-300">
      <h1 className="text-8xl text-left font-bold text-red-700"> Product Not Found</h1>
      <p className="text-lg text-green-700 text-center">The product you're looking for isn't available.</p>
    </div>
  )
}
```

```
useEffect(() => {
  const getProducts = async () => {
    try {
      const fetchedProducts = await fetchProducts();
      setProducts(fetchedProducts);
    } catch (err) {
      console.error('Error fetching products:', err);
    }
  };
  getProducts();
}, []);
```

```
const toggleWishlist = () => {
  try {
    let wishlist = JSON.parse(localStorage.getItem('wishlist') || '[]');
    if (!Array.isArray(wishlist)) {
      wishlist = [];
    }

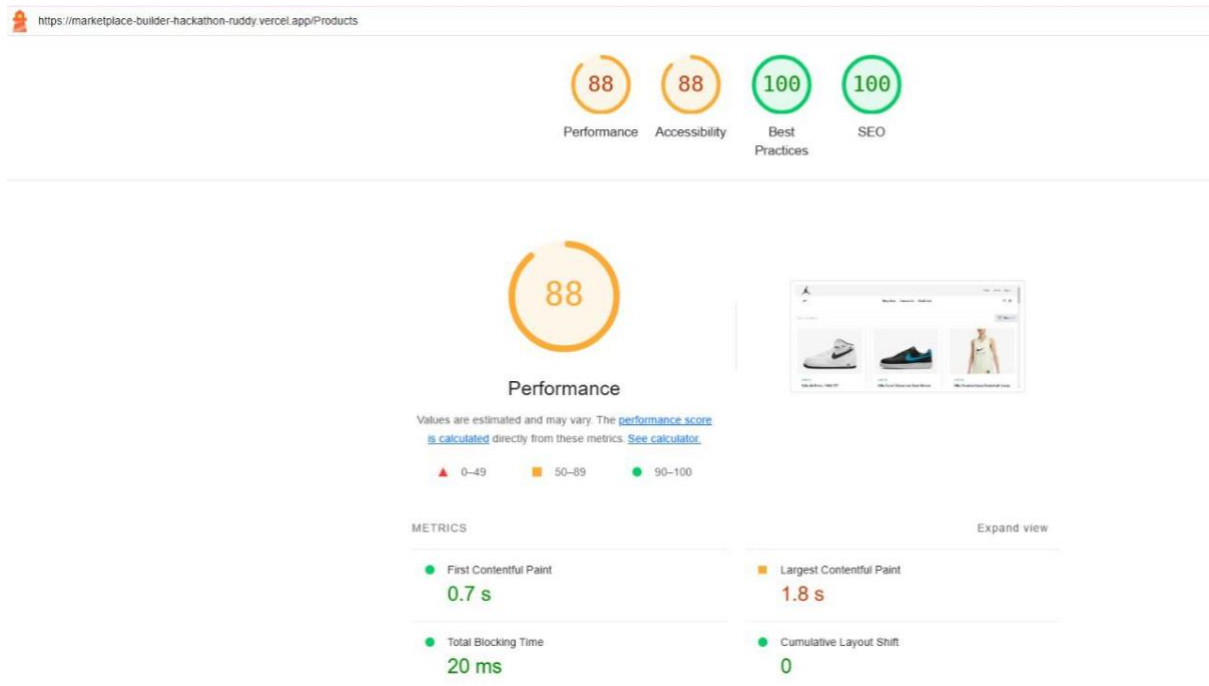
    if (isInWishlist) {
      wishlist = wishlist.filter((item: { id: string }) => item.id !== product.id);
    } else {
      wishlist.push(product);
    }

    localStorage.setItem('wishlist', JSON.stringify(wishlist));
    setIsInWishlist(!isInWishlist);
  } catch (error) {
    console.error('Error updating wishlist:', error);
  }
};
```



## Performance Optimization

I perform the Lighthouse performance test to know about my website performance optimization. Below is the screenshot of Lighthouse report



## Cross Browser and Device Testing

My Website is Responsive for all devices as it pass the below tests

TC012	View on Google Chrome and Microsoft Edge	Open the chrome and Microsoft Edge to view the website	It shows responsiveness	Passed
TC013	Inspect on Browser	Inspect the website on browser for different screens	It shows responsiveness	Passed
TC014	Check on Mobile phone	View the website on Mobile phone ensuring responsiveness	It shows responsiveness	Passed

## User Acceptance Testing (U A T)

## **Perform Tasks:**

- Browsing products
- Display specific product details when clicked on the product
- Show related products on product display page
- Add to Cart multiple products
- Remove products from the Cart
- Checkout option showcasing total prices of the products
- Add product to the wishlist/favorite
- Remove product from wishlist/ favorite
- Filter the products by category and status
- Filter products by price, like Low to High , and High to Low
- Search products by name