### **📅 Day 5 Plan: Testing, Error Handling & Backend Integration Refinement**

**Goal:**

Ensure My application is bug-free, optimized for performance, and ready for deployment.

### **Step:01 Functional Testing**

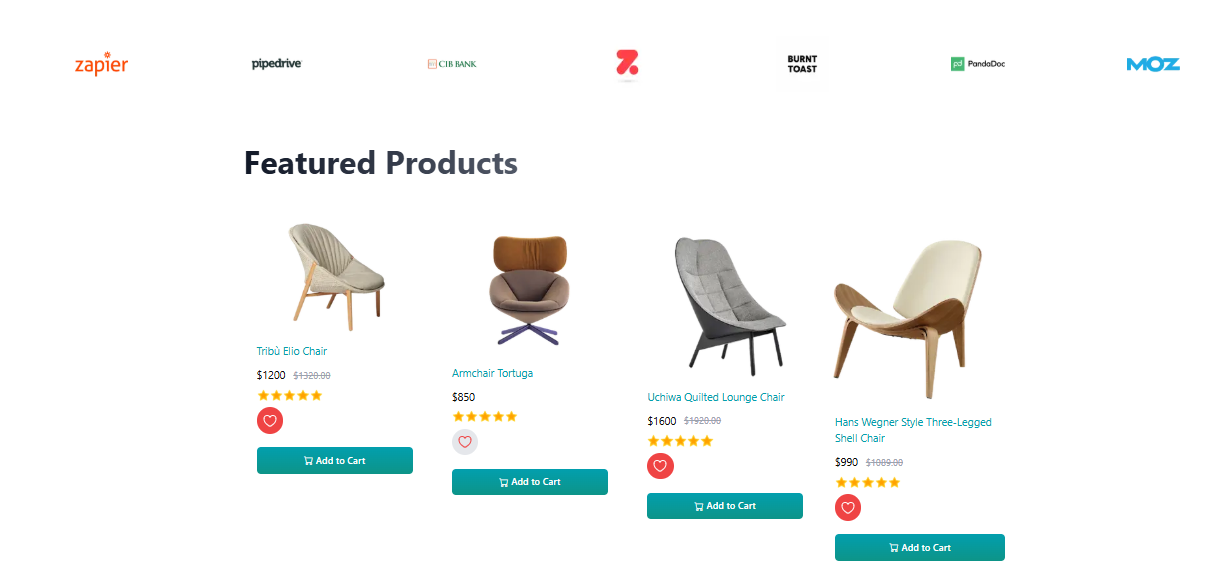
✅ Validate core marketplace features:

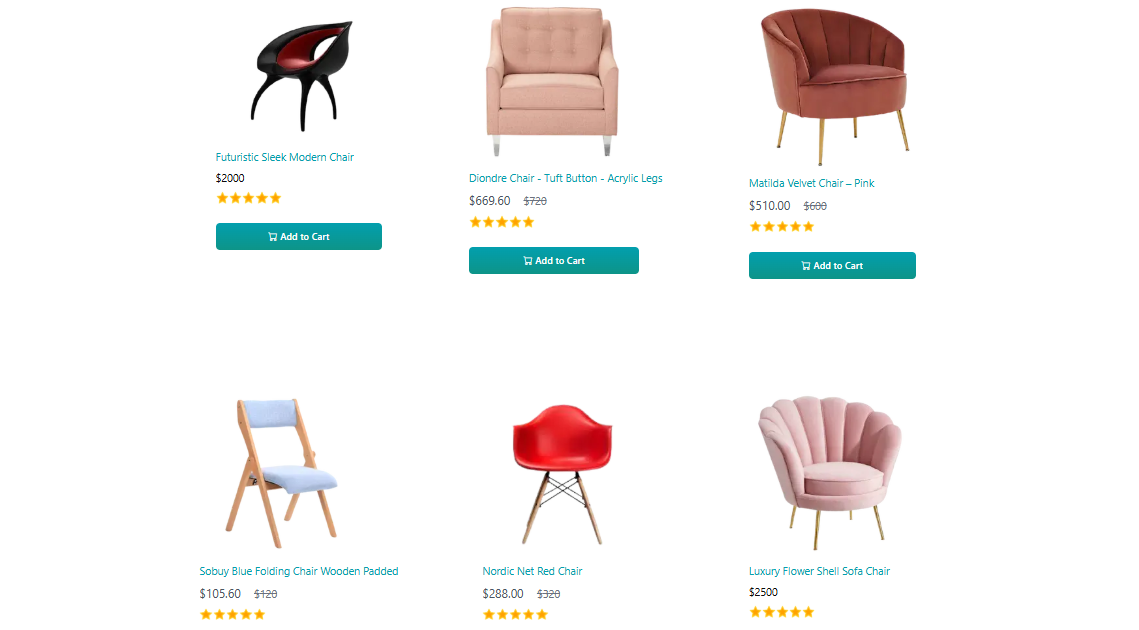
* Product listing, detail pages, search & filters
* Cart operations (Add, update, remove items)
* User authentication and profile management

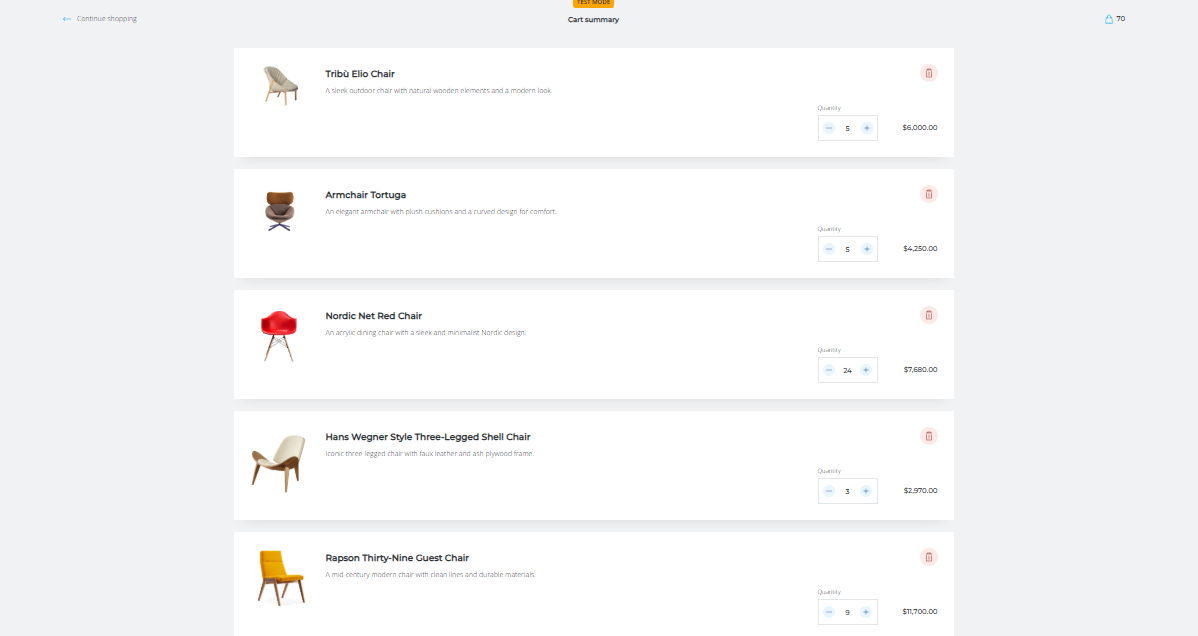
✅ Tools to use:

* **Postman** (API testing)
* **React Testing Library** (Unit testing)
* **Cypress** (End-to-end testing)









### **Step 02 Error Handling**

✅ Implement graceful error handling using **try-catch blocks**

✅ Display meaningful **error messages** for:

* Network failures
* Invalid/missing data
* Server errors

### **Step 03 Performance Optimization**

✅ Use **Lighthouse, GTmetrix, WebPageTest** to identify performance bottlenecks

✅ Apply optimization strategies:

* **Compress images** (TinyPNG, WebP)
* **Lazy loading** for large assets
* **Minimize Typescript & CSS**
* **Enable caching**

### **Step 04 Cross-Browser & Device Testing**

✅ Test application on different browsers:

* **Chrome, Firefox, Safari, Edge**

✅ Use **BrowserStack or LambdaTest** for device testing

✅ Ensure **responsive design** across all screen sizes

### **Break & Review Progress**

🚀 Stay hydrated, eat something nutritious, and mentally reset for the next phase.

### **Step 05 Security Testing**

✅ Prevent security vulnerabilities:

* **Sanitize inputs** to avoid **SQL Injection & XSS attacks**
* **Store API keys in .env files**
* **Use HTTPS for secure communication**

✅ Security testing tools:

* **OWASP ZAP** (Automated vulnerability scanning)
* **Burp Suite** (Advanced penetration testing)

### **Step 06 User Acceptance Testing (UAT)**

✅ Simulate real-world user behavior:

* Browsing, adding products, checking out
* Validating workflows for smooth navigation

✅ Collect feedback from **peers & mentors**

### **Step 07 Documentation Updates**

✅ Prepare a **CSV-based testing report** including:

* Test case ID
* Description
* Steps
* Expected vs. actual results
* Status (Pass/Fail)
* Severity level

✅ Write a **technical report (PDF)** summarizing:

* **Findings from testing**
* **Performance optimizations**
* **Security enhancement**

### **🎯 End-of-Day Deliverables**

✔ Fully tested marketplace with all features validated

✔ Well-documented error handling and fallback UI

✔ Optimized performance with fast load times

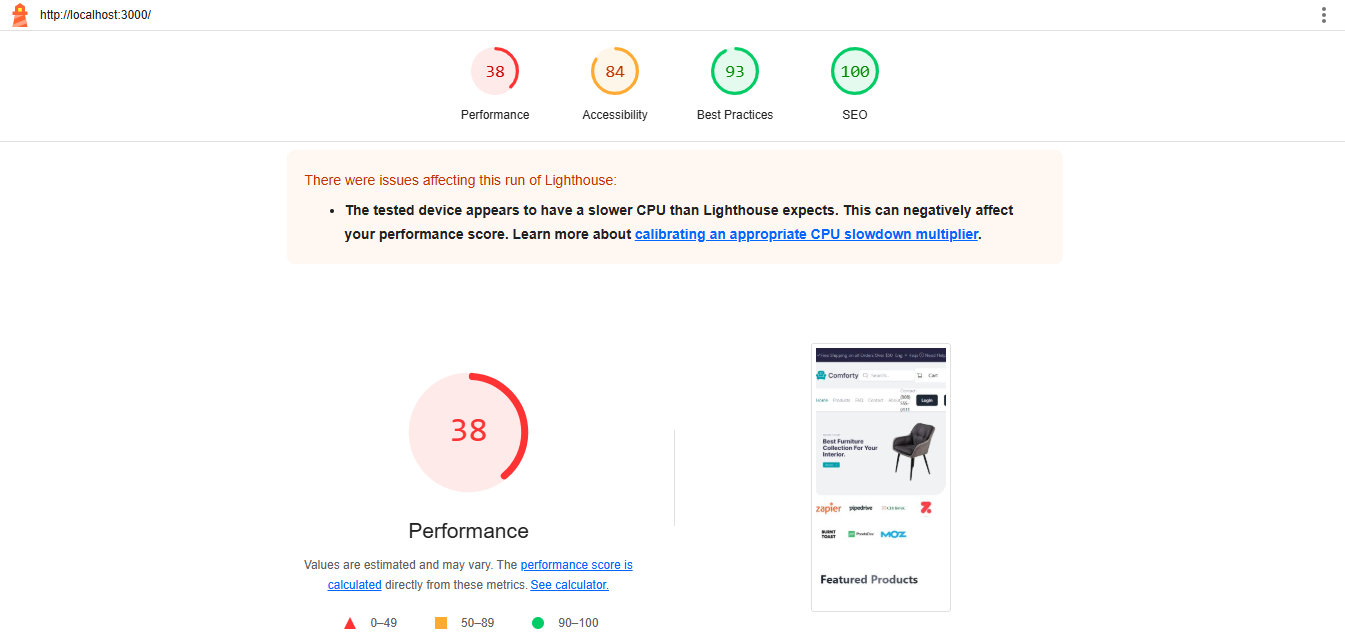
✔ Responsive design tested across multiple devices

✔ Security vulnerabilities addressed

✔ Complete documentation & testing report

# CSV test report

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Test Case Description** | **Test Steps** | **Expected Result** | **Actual Result** | **Status** | **Severity Level** | **Assigned To** | **Remarks** |  |
| 2 | TC001 | Validate product listing page | 1. Open product page 2. Verify product data against Sanity CMS 3. Check for accurate rendering of name, price, and images | Products displayed correctly with accurate details | Products displayed correctly with accurate details | Passed | High | - | No issues found |
| 3 | TC002 | Test API error handling | 1. Disconnect API or simulate network error 2. Refresh the page 3. Verify fallback UI and error message display | Fallback UI with appropriate error message is shown | Error message displayed as expected | Passed | Medium | - | Error handled gracefully with clear UI feedback |
| 4 | TC003 | Check cart functionality | 1. Add a product to the cart 2. Navigate to the cart page 3. Verify product details and quantity are displayed correctly | Cart updates with added product and accurate details | Cart updated as expected with correct details | Passed | High | - | Functionality works |



# Conclusion:

Day 5 was all about **testing, error handling, and refining backend integrations** to ensure my marketplace is ready for real-world deployment. I focused on **comprehensive testing** to validate the functionality, performance, and security of my application, ensuring a seamless user experience.