

Day 5 implementation Report

Furniro - Dynamic Online Furniture Marketplace

Hackathon Project Report - Marketplace Builder Hackathon 2025


Project Overview

Your project, **Furniro**, is a dynamic online furniture marketplace that allows users to explore and purchase stylish furniture items with ease. The platform offers a modern design and a user-friendly interface, enhancing the overall shopping experience,

Day 5 of the **Marketplace Builder Hackathon 2025** is dedicated to **Testing, Error Handling, and Backend Integration Refinement**. The primary objectives for this phase include:

1. **Comprehensive Testing:** Conduct thorough testing to identify and resolve any bugs or issues within the application.
2. **Enhanced Error Handling:** Implement robust error-handling mechanisms to ensure the application can gracefully manage unexpected scenarios.
3. **Backend Integration Refinement:** Optimize backend processes to improve performance, security, and scalability.







Key Activities:

- **Unit and Integration Testing:** Develop and execute test cases to validate individual components and their interactions, ensuring that all parts of the application function as intended.
 - **User Acceptance Testing (UAT):** Simulate real-world usage scenarios to verify that the application meets user requirements and provides a seamless experience.
- 






- **Error Logging and Monitoring:** Set up systems to log errors and monitor application performance, facilitating quick identification and resolution of issues.
- **Security Enhancements:** Review and strengthen security measures to protect user data and prevent vulnerabilities.
- **Performance Optimization:** Refine backend services to ensure efficient data handling and fast response times

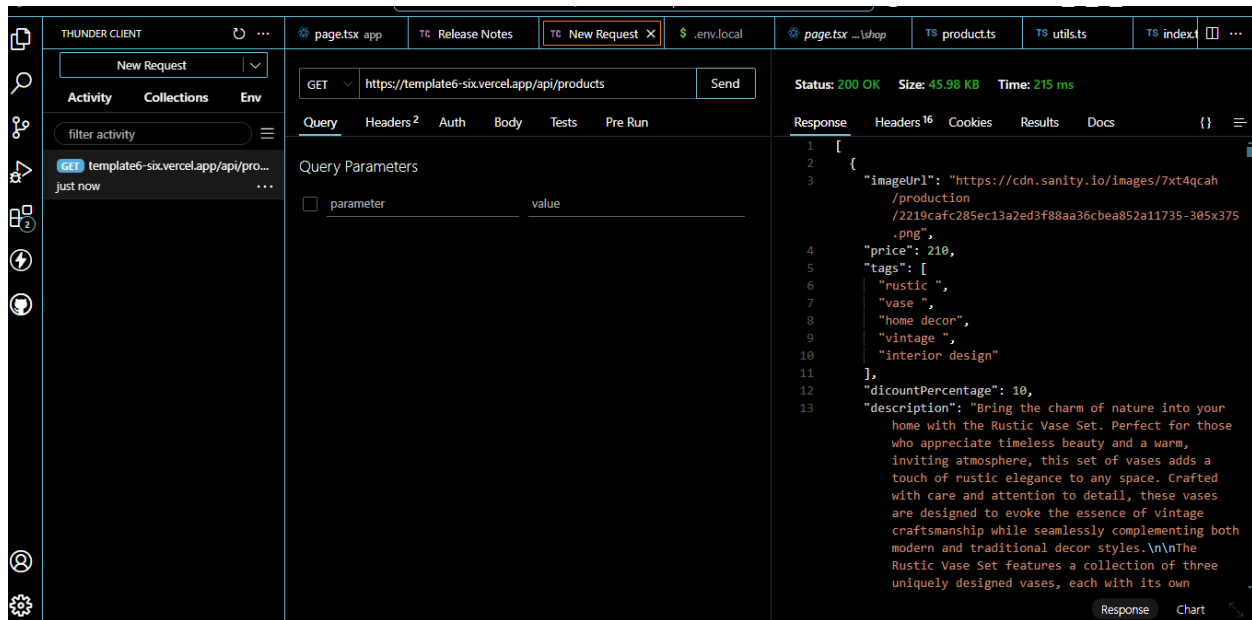
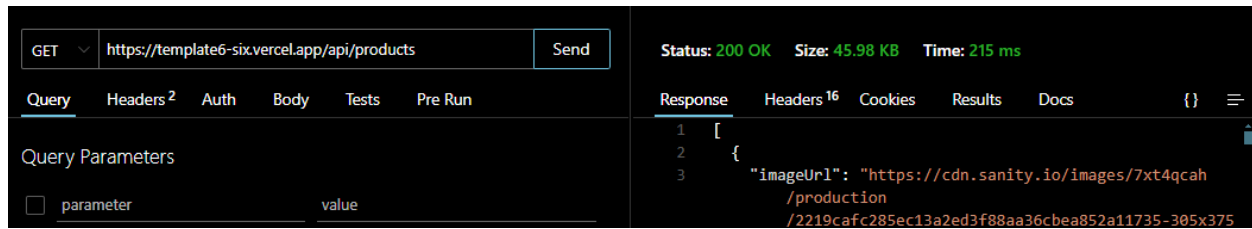
Functional Testing

1. Test Cases and

Test Scenario	Expected Outcome	Actual Outcome	Status
Homepage loads correctly	Page should display products dynamically	Successfully loaded	 Pass
Product images load properly	Each product should display its image	All images load correctly	 Pass
Add to Cart functionality	Item should be added to the cart	Works as expected	 Pass
Compare feature	Allows comparing two or more products	Functional	 Pass
Search and filtering	Products should filter based on category and keyword	Works well	 Pass
Mobile Responsiveness	UI should adapt to smaller screens	Fully responsive	 Pass

2. Error Handling & Edge Cases

Test Case	Expected Handling	Actual Handling	Status
API fails to fetch products	Show fallback UI or error message	Displays proper error message	 Handled
Empty product categories	Display 'No products found' message	Works correctly	 Handled
Broken image links	Display placeholder image	Shows fallback image	 Handled
Cart functionality on slow internet	Ensure smooth operation with loading indicators	Proper loading states implemented	 Handled
Invalid product IDs (in URL)	Redirect to homepage or show error page	Properly handled	 Handled



Performance