

Testing Report for FoodTuck Website

Prepared By: Rabia (Testing Team)

Date: January 22, 2025

Overview

This report documents the testing, error handling, and backend refinement conducted for the FoodTuck Q-Commerce marketplace. The goal was to ensure functionality, responsiveness, and reliability of the platform, following professional testing standards.

Test Cases Executed

1. Product Listing Page Validation

- **Test Case ID:** TC001
 - ✓ **Steps:** Open the products page and verify product display.
 - ✓ **Expected Result:** Products should display successfully.
 - ✓ **Actual Result:** Products displayed successfully.
 - ✓ **Severity Level:** Low
 - ✓ **Remarks:** No issues found.

2. API Error Handling

- **Test Case ID:** TC002
 - ✓ **Steps:** Disconnect API > Refresh the page.
 - ✓ **Expected Result:** Display an error message.
 - ✓ **Actual Result:** Error message displayed.
 - ✓ **Severity Level:** Low
 - ✓ **Remarks:** Handled gracefully.

3. Cart Functionality

- **Test Case ID:** TC003
 - ✓ **Steps:** Add items to the cart and navigate to the cart.
 - ✓ **Expected Result:** Products should appear in the cart.
 - ✓ **Actual Result:** Products displayed correctly.
 - ✓ **Severity Level:** Medium

- ✓ **Remarks:** Working as intended.

4. Responsiveness on Mobile

- **Test Case ID:** TC004
- ✓ **Steps:** Inspect website using mobile device simulation.
- ✓ **Expected Result:** Content adjusts to the screen size.
- ✓ **Actual Result:** Content adjusts correctly.
- ✓ **Severity Level:** Medium
- ✓ **Remarks:** Responsive design confirmed.

5. Search Bar Validation

- **Test Case ID:** TC005
- ✓ **Steps:** Search for non-existent items.
- ✓ **Expected Result:** "No products found" dropdown should appear.
- ✓ **Actual Result:** Dropdown did not appear.
- ✓ **Severity Level:** High
- ✓ **Remarks:** Needs fixing to handle irrelevant search queries.

6. Dynamic Product Detail Pages

- **Test Case ID:** TC006
- ✓ **Steps:** Open a product card and validate its dynamic page.
- ✓ **Expected Result:** The correct product details page opens.
- ✓ **Actual Result:** Functions as expected.
- ✓ **Severity Level:** Low
- ✓ **Remarks:** No issues found.

7. Cross-Browser Compatibility

- **Test Case ID:** TC007
- ✓ **Steps:** Open the website on Chrome, Firefox, and Microsoft Edge.
- ✓ **Expected Result:** Consistent functionality and layout.
- ✓ **Actual Result:** Worked seamlessly across all tested browsers.
- ✓ **Severity Level:** Low
- ✓ **Remarks:** No issues found.

Performance Optimization

- **Tools Used:** Lighthouse, GTmetrix
 - **Steps Taken:**
 1. Compressed images using TinyPNG.
 2. Implemented lazy loading for assets.
 3. Reduced unused CSS and JavaScript.
 4. Enabled browser caching for faster repeat visits.
 - **Results:**
 - Initial load time reduced to under 2 seconds.
 - Improved Lighthouse performance score: **95/100**.
-

Error Handling

- **API Failures:** Used try-catch blocks to manage errors.
 - **Fallback UI:** Displayed user-friendly messages (e.g., "No products available").
 - **Examples:**

```
try {  
  const data = await fetchProducts();  
  setProducts(data);  
} catch (error) {  
  console.error("Failed to fetch products:", error);  
  setError("Unable to load products. Please try again later.");  
}
```
-

Cross-Browser and Device Testing

- **Browsers Tested:** Chrome, Firefox, Safari, Edge.
 - **Devices Tested:** Desktop, tablet, iPhone 12.
 - **Results:**
 - ✓ Fully responsive across all tested devices and browsers.
 - ✓ No layout issues observed.
-

Security Measures

Input fields validated to prevent injection attacks.

HTTPS enabled for secure communication.

API keys stored in environment variables.

Tools Used: OWASP ZAP, Burp Suite.

Challenges Faced

1. Search bar functionality failed to display "No products found" dropdown for irrelevant searches.
 - **Resolution:** To be fixed in the next sprint.
2. Initial performance optimization was hindered by large image sizes.
 - **Resolution:** Images compressed effectively.

Conclusion

The FoodTuck website passed most functional and non-functional testing parameters, with the exception of the search bar issue. It is now optimized for performance and responsive design, ready for real-world deployment after resolving the identified issues.

Next Steps

1. Fix the search bar dropdown issue.
2. Conduct additional user acceptance testing (UAT) with real users.
3. Re-test after implementing fixes.

Attachments

- CSV-based testing report.
- Lighthouse and GTmetrix performance audit logs.
- Security testing summary report.