

## Grocery Store Web Application Testing Report

### 1. Test Cases Executed and Their Results

Test Case ID	Test Case Description	Test Steps	Expected Result	Actual Result	Status	Severity Level	Assigned To
TC-001	Measure initial page load speed	Load the page and measure time using dev tools	Page loads within 2 seconds	Page loads in 1.6 seconds	Passed	Low	N/A
TC-002	Check layout on different screen sizes	Resize browser and inspect UI elements	Layout adjusts without breaking	Layout adjusts correctly	Passed	Low	N/A
TC-003	Test search input with various queries	Enter search terms and observe results	Relevant results are displayed	Correct results are shown	Passed	Medium	N/A
TC-004	Verify navigation links work correctly	Click each navigation link and check redirections	Correct pages are loaded	All links navigate correctly	Passed	Low	N/A
TC-005	Ensure proper sliding and navigation	Interact with carousel buttons	Slides transition smoothly	Smooth transitions observed	Passed	Low	N/A
TC-006	Evaluate contrast, labels, and ARIA roles	Run accessibility audit via Lighthouse	Passes accessibility score above 90	Accessibility score: 90	Passed	Medium	N/A
TC-007	Validate proper use of meta tags	Inspect meta tags in HTML	Proper tags for SEO present	All required tags are present	Passed	Low	N/A
TC-008	Check validation for required fields	Submit forms with and without required fields	Error messages displayed for missing fields	Validation messages appear	Passed	High	N/A

2. Performance Optimization Steps Taken

- **Image Optimization:** Compressed images to reduce load time without compromising quality.
- **Lazy Loading:** Implemented lazy loading for images to improve page rendering speed.
- **Minification:** Minified CSS and JavaScript files to reduce file sizes.
- **Caching Strategies:** Enabled browser caching to store static assets locally.
- **Code Splitting:** Divided large JavaScript bundles into smaller, manageable chunks.
- **CDN Usage:** Deployed assets through a Content Delivery Network (CDN) for faster content delivery.

3. Security Measures Implemented

- **HTTPS Enforcement:** Ensured the website is served over HTTPS for secure communication.
- **Input Sanitization:** Applied input validation to prevent SQL injection and XSS attacks.
- **CORS Policy:** Configured Cross-Origin Resource Sharing (CORS) to restrict unauthorized access.
- **Authentication:** Implemented JWT-based authentication for user sessions.
- **Content Security Policy (CSP):** Enforced CSP headers to mitigate XSS risks.
- **Regular Security Scans:** Conducted periodic vulnerability assessments.

4. Challenges Faced and Resolutions Applied

Challenge	Resolution Applied
Slow Initial Page Load	Implemented server-side rendering (SSR)
Mobile Layout Overflows	Fixed CSS flexbox issues and tested on various devices
Third-Party Script Conflicts	Deferred non-critical scripts to prevent delays
High Largest Contentful Paint (LCP)	Optimized critical CSS and reduced render-blocking resources

# SCREENSHOTS:

