abudullah saleem

day 5 of haaktone

Documentation for Functional Testing Implementation

1. **Optimize Images:**
   - **Compression:** Ensure images are compressed without significant quality loss to reduce file sizes.
   - **Responsive Formats:** Use modern formats like WebP for better compression and quality.
   - **Proper Dimensions:** Serve images with dimensions that match their display size to avoid unnecessary scaling.
2. **Minimize CSS and JavaScript:**
   - **Minification:** Remove unnecessary characters (like whitespaces and comments) from your CSS and JS files to reduce their size.
   - **Bundling:** Combine multiple CSS and JS files into single files to reduce the number of HTTP requests.
   - **Defer Non-Critical JS:** Use the defer attribute to delay the loading of non-essential JavaScript until after the main content has loaded.
3. **Leverage Browser Caching:**
   - **Set Expiry Headers:** Configure your server to specify how long browsers should cache certain files, reducing the need to re-download them on subsequent visits.
4. **Reduce Server Response Time:**
   - **Efficient Hosting:** Choose a reliable hosting provider known for fast response times.
   - **Database Optimization:** Regularly clean and optimize your database to ensure quick data retrieval.
   - **Content Delivery Network (CDN):** Utilize a CDN to distribute your content globally, ensuring faster delivery to users based on their geographic location.
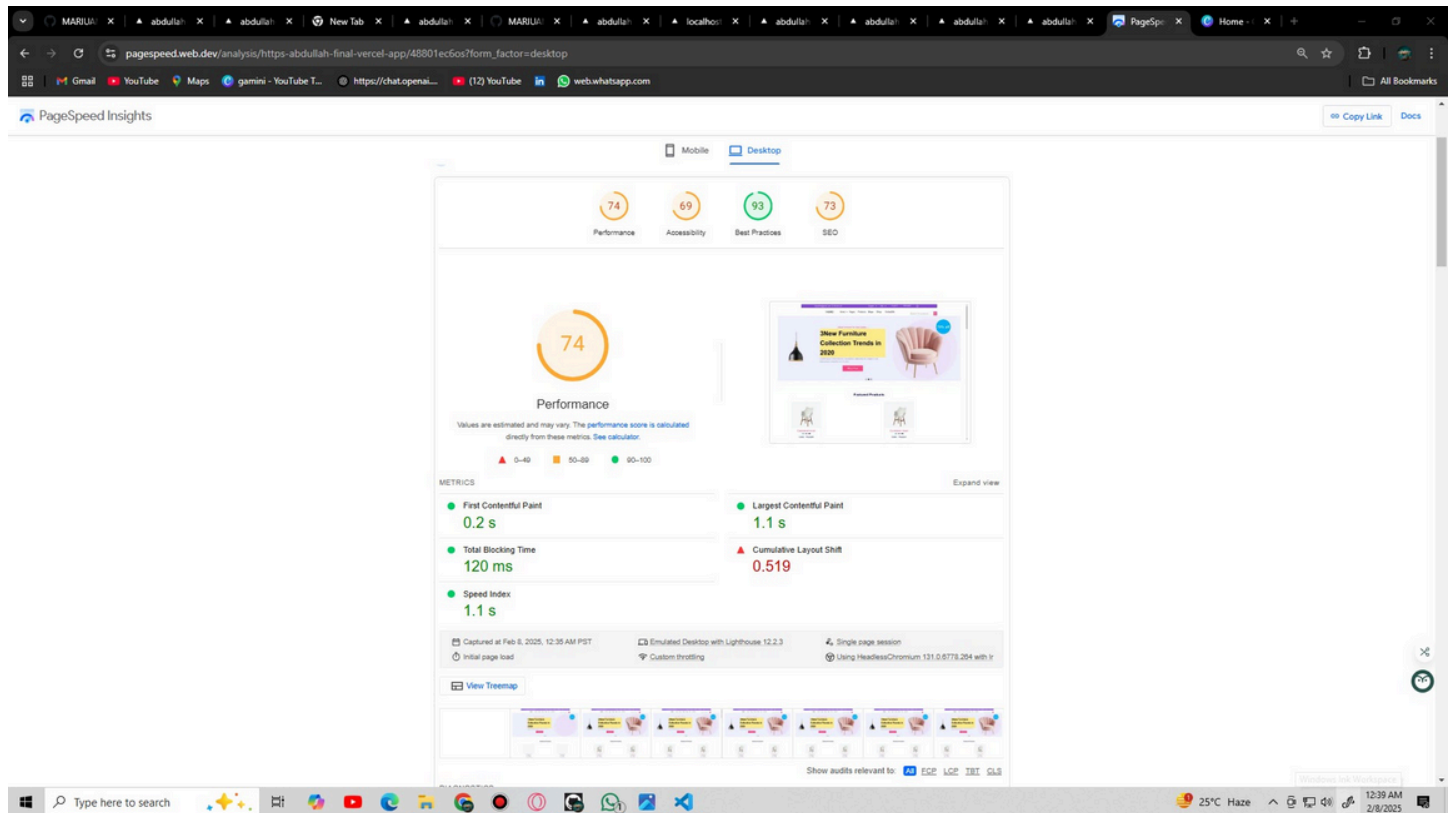5. **Eliminate Render-Blocking Resources:**
   - **Critical CSS:** Inline essential CSS directly into the HTML to allow the browser to render content faster.
   - **Async and Defer JS:** Implement async or defer attributes for JavaScript files to prevent them from blocking the rendering of the page.
6. **Enable Text Compression:**
   - **Gzip or Brotli:** Configure your server to compress text-based resources, such as HTML, CSS, and JavaScript files, before sending them to the browser.
7. **Ensure Proper Resource Loading:**

- ○ **Lazy Loading:** Implement lazy loading for images and videos so that they load only when they come into the viewport, reducing initial load times.
- ○ **Preload Key Resources:** Use <link rel="preload"> for critical resources to instruct the browser to load them early.

PageSpeed Insights

Copy Link   Docs

Mobile   Desktop

74

Performance

Values are estimated and may vary. The performance score is calculated directly from these metrics. See calculator.

▲ 0–49   ■ 50–89   ● 90–100

METRICS   Expand view

● First Contentful Paint   ● Largest Contentful Paint
0.2 s   **1.1 s**

● Total Blocking Time   ▲ Cumulative Layout Shift
120 ms   **0.519**

● Speed Index
1.1 s

Captured at Feb 8, 2025, 12:38 AM PST   Emulated Desktop with Lighthouse 12.2.3   Single page session
Initial page load   Custom throttling   Using HeadlessChromium 121.0.6779.264 with lr

View Treemap

Show audits relevant to: All FCP LCP TBT CLS

DIAGNOSTICS

▲ Avoid large layout shifts — 2 layout shifts found

▲ Serve images in next-gen formats — Potential savings of 514 KiB

▲ Avoid an excessive DOM size — 838 elements

▲ Properly size images — Potential savings of 326 KiB

▲ Reduce unused JavaScript — Potential savings of 44 KiB