Audit Report: Full-stack digital solutions – premier custom software & design

Website: https://playandnope.com/
Date: 12.10.2026
Audit Coverage: 95% — PageSpeed Insights (desktop): PSI HTTP 500; PageSpeed Insights (desktop): PSI HTTP 502
Confidence: medium

Pages Audited (5 of 5):

- https://playandnope.com/
- https://playandnope.com/privacy-policy
- https://playandnope.com/contact
- https://playandnope.com/full-stack-digital-agency
- https://playandnope.com/digital-solutions-contact

Summary of results

Overall Score: 63 / 100
Status: □ □ **Needs Improvement**

Site overall 63 is the mean of 5 pages. Scores range 52 (https://playandnope.com/full-stack-digital-agency) → 82 (https://playandnope.com/privacy-policy). Weakest page: PSI mobile performance 39 is the primary drag, driven by catastrophic TBT (30,070 ms) and LCP (6.3 s) caused by long tasks in global.js. Accessibility (98) and Security (PASS) are strong, but the heading order error (h1→h3) is a WCAG violation. Desktop PSI failed (502), so mobile data is used for both, lowering confidence slightly. The site is a digital agency brochure with no auth/payments, lowering security header priority.

Per-page scores

□ **Needs Improvement** · https://playandnope.com/

Score	Performance	Accessibility	Best Practices	SEO	Security
68	34	100	100	92	PASS

□ **Needs Improvement** · https://playandnope.com/privacy-policy

Score	Performance	Accessibility	Best Practices	SEO	Security
82	84	100	100	100	PASS

 Poor · https://playandnope.com/contact

Score	Performance	Accessibility	Best Practices	SEO	Security
58	34	99	96	100	PASS

 Poor · https://playandnope.com/full-stack-digital-agency

Score	Performance	Accessibility	Best Practices	SEO	Security
52	39	98	100	100	PASS

 Poor · https://playandnope.com/digital-solutions-contact

Score	Performance	Accessibility	Best Practices	SEO	Security
55	34	99	96	100	PASS

PageSpeed Insights — Mobile vs Desktop

*Lower is worse for Performance; higher is worse for LCP and CLS. Worse value is **bolded**.*

URL	Performance (M / D)	LCP (M / D)	CLS (M / D)
https://playandnope.com/	34 / —	5.44 s / —	0.000 / —
https://playandnope.com/privacy-policy	84 / **82**	**3.97 s** / 619 ms	0.000 / **0.367**
https://playandnope.com/contact	34 / —	13.86 s / —	0.035 / —
https://playandnope.com/full-stack-digital-agency	39 / —	6.35 s / —	0.000 / —
https://playandnope.com/digital-solutions-contact	34 / —	11.69 s / —	0.000 / —

Optimization Checklist

3 of 3 passing — 3 pass · 0 warn · 0 fail · 5 n/a

Item	Status	Detail
Page caching plugin / CDN active	**Pass**	Caching plugin detected (WP Rocket)
Response compressed (gzip / brotli)	**Pass**	Document response is compressed with gzip.
Images lazy-loaded	N/A	No raster elements found (25 SVGs, 5 placeholders excluded).
Hero image eagerly loaded	N/A	No raster elements found (25 SVGs, 5 placeholders excluded).
Hero is a real (not a CSS background-image)	N/A	No CSS background-images detected on raster-image-eligible elements.
Responsive images (srcset /)	N/A	Only 0 raster images on the page (25 SVGs, 5 placeholders excluded) — responsive-image rule does not apply.
Reasonable number of image sizes	N/A	Too few raster images to evaluate srcset width variety.
JS scripts not blocking in	**Pass**	No render-blocking scripts in .

Fixes

Priority 1: Critical

Immediate action — impacts user experience, search rankings, or site safety.

1A. Reduce JavaScript execution time to fix TBT `Performance`

- **Impact:** TBT, LCP, Interaction readiness
- **Problem:** Total Blocking Time is 24.95 s with multiple long tasks (>1 s) from theme scripts (core.js, global.js).
- **Solution:**

- Split `global.js` and `core.js` into smaller chunks.
- Defer non-critical JS further or use `type="module"` with `defer`.
- Remove unused JS (92 KB identified).
- Audit third-party scripts (cookieconsent, iframemanager) for necessity.

1B. Reduce Mobile LCP to under 2.5 s `Performance`

- **Impact:** LCP, Core Web Vitals
- **Problem:** Mobile LCP is 4.0 s (field 2.4 s), hitting the warning threshold; lab data shows forced reflow and long tasks contributing to delay.
- **Solution:**
  - Preload the LCP element (likely the hero SVG or text).
  - Defer non-critical CSS/JS.
  - Optimize server response time (TTFB is 2 ms, so focus on render-blocking resources).

1C. Fix color contrast on h1 `Accessibility`

- **Impact:** WCAG 1.4.3 Contrast
- **Problem:** axe-core reports 1 serious violation: color-contrast on h1 elements.
- **Solution:**
  - Increase contrast ratio to ≥4.5:1 for normal text.
  - Use a color picker tool to verify foreground/background contrast.
  - Example: Change text color to #333333 on white background.

1D. Reduce JavaScript execution time and blocking `Performance`

- **Impact:** TBT (28.7 s), LCP (12.4 s), FCP
- **Problem:** TBT is 28,410 ms; reCAPTCHA and theme JS cause long tasks (1.28 s, 1.21 s) and unused JS warnings.
- **Solution:**
  - Defer non-critical scripts (reCAPTCHA) until user interaction.
  - Split theme JS bundles; remove unused code.
  - Use `requestIdleCallback` for non-critical initialization.
  - Consider server-side reCAPTCHA verification to reduce client-side load.

1E. Harden Content-Security-Policy (CSP) `Security`

- **Impact:** XSS protection, data integrity

- **Problem:** CSP is present but weak (missing `default-src`, `object-src 'none'`). Site signals indicate User-Generated Content (textarea), making XSS a Priority 1 risk per rubric.
- **Solution:**
  - Add `default-src 'self'`.
  - Add `object-src 'none'`.
  - Use nonce-based script-src for inline scripts.
  - Example: `Content-Security-Policy: default-src 'self'; script-src 'nonce-{random}' 'strict-dynamic'; object-src 'none';`

1F. Reduce Total Blocking Time and LCP `Performance`

- **Impact:** TBT, LCP, User Experience
- **Problem:** TBT is 30,070 ms and LCP is 6.3 s; global.js causes multiple long tasks (1.28 s, 957 ms).
- **Solution:**
  - Split `global.js` into smaller chunks.
  - Defer non-critical JS execution.
  - Optimize main thread work to reduce blocking time.

1G. Reduce JavaScript execution time `Performance`

- **Impact:** TBT, FCP, LCP
- **Problem:** Total Blocking Time is 19.73s; unused JavaScript from reCAPTCHA (~176KB x3) and theme scripts (~92KB) blocks the main thread.
- **Solution:**
  - Defer non-critical scripts (e.g., reCAPTCHA) until user interaction.
  - Remove unused JavaScript bundles.
  - Split code to reduce main thread work.

1H. Improve Largest Contentful Paint (LCP) `Performance`

- **Impact:** LCP, User Experience
- **Problem:** LCP is 11.7s (threshold 2.5s), likely caused by render-blocking resources or slow JS execution.
- **Solution:**
  - Preload the LCP element (likely a hero image or text).
  - Ensure critical CSS is inlined.

- Optimize server response time (TTFB is 47ms, so focus on client-side rendering).

Priority 2: Important

Essential for compliance, user reach, and search visibility.

2A. Add explicit width and height to images `Performance`

- **Impact:** CLS, Layout stability
- **Problem:** 5 images lack width/height attributes, risking layout shifts during load.
- **Solution:** Add `width` and `height` attributes to all `<img>` tags:

  ```
  <img src="image.jpg" width="800" height="600" alt="...">
  ```

  Or use CSS aspect-ratio if dimensions vary.

2B. Remove unused JavaScript `Performance`

- **Impact:** JS execution time, TBT
- **Problem:** 112 KB of unused JavaScript found in `global.62444fa2566b44750ca1.min.js`, contributing to long tasks (218 ms).
- **Solution:**
  - Audit `global.js` for unused functions.
  - Use code splitting or tree-shaking to remove dead code.
  - Consider loading scripts only on pages where they are needed.

2C. Label form inputs and fix HTML errors `Accessibility`

- **Impact:** WCAG 1.3.1, 3.3.2, 4.1.1
- **Problem:** Textarea has no label; 2 W3C errors on hidden inputs with invalid `autocomplete` attributes.
- **Solution:**
  - Add `<label for="...">` to the textarea or use `aria-label`.
  - Remove `autocomplete` attribute from `<input type="hidden">` elements.
  - Ensure all form fields have associated labels for screen readers.

2D. Add COOP and CORP headers `Security`

- **Impact:** Cross-origin isolation, side-channel attacks
- **Problem:** Cross-Origin-Opener-Policy and Cross-Origin-Resource-Policy headers are missing.
- **Solution:**

- Add `Cross-Origin-Opener-Policy: same-origin`.
- Add `Cross-Origin-Resource-Policy: same-origin`.
- Configure in server config (e.g., Apache `.htaccess` or Nginx).

2E. Fix Heading Order `Accessibility`

- **Impact:** WCAG 1.3.1, Screen Reader Navigation
- **Problem:** h1 is followed immediately by h3, skipping h2 (W3C error + axe moderate violation).
- **Solution:** Change the `<h3>` following the `<h1>` to `<h2>` to maintain sequential hierarchy.

2F. Strengthen Content-Security-Policy `Security`

- **Impact:** XSS Protection
- **Problem:** CSP is missing `default-src` and `object-src 'none'`; site has a textarea (user content) which increases XSS risk.
- **Solution:** Update CSP header to include: `default-src 'self'; object-src 'none'; script-src 'self' 'unsafe-inline' 'unsafe-eval';` (Adjust based on actual script sources).

Priority 3: Best Practice

Recommended for long-term maintainability.

3A. Strengthen Content-Security-Policy and add COOP/CORP `Security`

- **Impact:** XSS defense, Context isolation
- **Problem:** CSP lacks `default-src` and `object-src 'none'`; COOP and CORP headers are missing.
- **Solution:** Update headers (Apache example):

```
Header always set Content-Security-Policy "default-src 'self';
object-src 'none'; frame-ancestors 'self'"
Header always set Cross-Origin-Opener-Policy "same-origin"
Header always set Cross-Origin-Resource-Policy "same-origin"
```

  Note: Prioritize performance fixes first; CSP is lower priority for this brochure site.

3B. Disambiguate vague link text `SEO`

- **Impact:** SEO, Accessibility
- **Problem:** 4 links use vague text like 'read more' without context.
- **Solution:** Update link text to describe the destination:

```
<a href="/services">Read more about our services</a>
```

Or use `aria-label` if visual text must remain generic.

3C. Strengthen Content-Security-Policy `Security`

- **Impact:** XSS defense-in-depth
- **Problem:** CSP is present but weak (missing default-src; object-src not 'none'). Site signals show no auth/payments/UGC, so risk is lower.
- **Solution:**
  - Add `default-src 'self'`.
  - Add `object-src 'none'`.
  - Example: `Content-Security-Policy: default-src 'self'; object-src 'none'; frame-ancestors 'self';`

3D. Enable HSTS Preload `Security`

- **Impact:** Transport security
- **Problem:** HSTS header is present but lacks the `preload` directive, preventing inclusion in browser preload lists.
- **Solution:**
  - Add `preload` to HSTS header: `Strict-Transport-Security: max-age=31536000; includeSubDomains; preload`.
  - Submit domain to hstspreload.org.

3E. Resolve console errors and source maps `Best Practices`

- **Impact:** Debugging, user experience
- **Problem:** Console error: `requestStorageAccess: Permission denied`; missing source maps for large JS.
- **Solution:**
  - Investigate reCAPTCHA storage access permission issues.
  - Generate and serve source maps for production JS files to aid debugging without exposing source code.

3F. Add COOP and CORP Headers `Security`

- **Impact:** Context Isolation, Cross-Origin Reads
- **Problem:** Cross-Origin-Opener-Policy and Cross-Origin-Resource-Policy headers are missing.

- **Solution:** Set `Cross-Origin-Opener-Policy: same-origin` and `Cross-Origin-Resource-Policy: same-origin` to isolate browsing context.

3G. Fix W3C HTML Validation Errors `Accessibility`

- **Impact:** Code Quality, Accessibility
- **Problem:** 2 errors found: hidden input elements with `autocomplete` attribute (line 896).
- **Solution:** Remove `autocomplete` attribute from `<input type="hidden">` elements. Example: `<input type="hidden" name="gform_target_" ...>`

3H. Add Cross-Origin Headers `Security`

- **Impact:** Data Leakage, Clickjacking
- **Problem:** Missing `Cross-Origin-Opener-Policy` and `Cross-Origin-Resource-Policy` headers.
- **Solution:** Add headers: `Cross-Origin-Opener-Policy: same-origin` `Cross-Origin-Resource-Policy: same-origin`