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

Website: https://playandnope.com/
Date: 12.10.2026
Audit Coverage: 93% — PageSpeed Insights (desktop): PSI HTTP 500; PageSpeed Insights: mobile: PSI request timed out after 240000ms; desktop: PSI HTTP 500; W3C Validator: W3C HTTP 429
Confidence: low

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: 60 / 100
Status: □ □ **Needs Improvement**

Site overall 60 is the mean of 4 pages. Scores range 45 (https://playandnope.com/contact) → 74 (https://playandnope.com/privacy-policy). Weakest page: Performance is the dominant drag: PSI mobile 34 / desktop 43, with mobile LCP 12.4 s (desktop 2.1 s), TBT 27.81 s, Speed Index 9.38 s, and TTI 43.5 s — the site is effectively unusable on mobile hardware. The TBT is driven by long tasks from core.js (1.72 s), jquery (1.08 s), and repeated global.js tasks (300–400 ms each), plus 1.42 MB of reCAPTCHA JS from gstatic loaded three times. Accessibility is excellent (PSI 99/100, axe 1 minor violation), and image assets are clean (5 SVGs, all with alt/dimensions/lazy). SEO scores 100 but tapTargets and structuredData audits fail, and the Security basics verdict is PASS with weak CSP (only frame-ancestors, no default-src/object-src) and missing COOP/CORP. The mobile/desktop performance gap (34 vs 43, LCP 12.4 s vs 2.1 s) confirms the mobile experience is the outlier and the primary reason the overall lands in the Poor band.

Per-page scores

 Poor · https://playandnope.com/

Score	Performance	Accessibility	Best Practices	SEO	Security
57	34	100	100	92	PASS

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

Score	Performance	Accessibility	Best Practices	SEO	Security
74	75	100	100	100	PASS

 Poor · https://playandnope.com/contact

Score	Performance	Accessibility	Best Practices	SEO	Security
45	34	99	96	100	PASS

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

Score	Performance	Accessibility	Best Practices	SEO	Security
64	60	98	100	100	PASS

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

Score	Performance	Accessibility	Best Practices	SEO	Security
—	—	—	—	—	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 / —	5.85 s / —	0.000 / —
https://playandnope.com/privacy-policy	**75** / 98	**6.48 s** / 605 ms	0.000 / **0.084**

URL	Performance (M / D)	LCP (M / D)	CLS (M / D)
https://playandnope.com/contact	**34** / 43	**12.36 s** / 3.00 s	0.055 / **0.083**
https://playandnope.com/full-stack-digital-agency	60 / —	2.20 s / —	0.013 / —

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. Cut Total Blocking Time by splitting and trimming JavaScript `Performance`

- **Impact:** TBT, TTI, INP, Max Potential FID
- **Problem:** TBT is 27.52 s and TTI 41.9 s; long tasks come from core.c872dceb075db34864df.min.js (2.29 s), jquery.f5f8c1b60d5ba64e00fb.min.js (1.25 s) and eight ~350–380 ms tasks from global.62444fa2566b44750ca1.min.js, which also wastes 92 KB of unused JS.
- **Solution:**
  - Drop jQuery / jquery-migrate if the theme does not truly need them; the theme loads both `jquery.min.js` and `wp-includes/js/jquery/jquery-migrate.min.js`.
  - Code-split `global.min.js` so only the modules used on the landing page are parsed; the audit shows 92 KB of unused code in that single file.
  - Break the 2.29 s core.js task into smaller chunks (dynamic `import()` or deferred init on `requestIdleCallback`).
  - Audit third-party scripts (jsdelivr cookieconsent + iframemanager) for main-thread cost.

1B. Fix forced reflow / layout thrash `Performance`

- **Impact:** TBT, INP, rendering
- **Problem:** `forced-reflow-insight` scores 0.00 and is listed as a high-priority fix; forced synchronous layouts block the main thread and compound the 27.52 s TBT.
- **Solution:**
  - Batch DOM reads before writes; never interleave `offsetHeight` / `getBoundingClientRect()` with style mutations in a loop.
  - Cache layout values instead of re-reading them per element.
  - Use `requestAnimationFrame` for visual updates and `ResizeObserver` instead of resize-driven read/write cycles.
  - Profile with DevTools Performance panel → "Forced reflow" markers to find the offending loop.

1C. Improve LCP, FCP and Speed Index `Performance`

- **Impact:** LCP, FCP, Speed Index

- **Problem:** LCP is 6.8 s, FCP 4.49 s and Speed Index 8.89 s; `lcp-breakdown-insight` and `network-dependency-tree-insight` both score 0.00, indicating a long critical path.
- **Solution:**
  - Identify the LCP element and preload it (`<link rel="preload" as="image" fetchpriority="high">` or font preload).
  - Flatten the critical request chain — the network dependency tree insight flags a deep chain.
  - Inline critical CSS and defer the 3 external stylesheets.
  - Reduce the 105.3 KB of font payload (3 font files) with `font-display: swap` and subsetting.

1D. Cut mobile LCP from 6.5 s `Performance`

- **Impact:** LCP, FCP, Speed Index, TTI
- **Problem:** Mobile LCP is 6.5 s (desktop 0.6 s); `document-latency-insight` estimates 610 ms savings and `network-dependency-tree-insight` is failing.
- **Solution:**
  - Identify the actual LCP element in DevTools Performance and preload it (`<link rel="preload" as="image" fetchpriority="high">` for images, `rel="preload" as="font" crossorigin` for the LCP font).
  - Flatten the critical request chain: inline critical CSS, defer non-critical CSS, and avoid chaining the LCP resource behind JS.
  - The page loads 3 fonts totalling 105.3 KB — subset them and use `font-display: swap` .

1E. Remove unused JavaScript and the 120 ms long task `Performance`

- **Impact:** TBT, LCP, TTI
- **Problem:** `global.62444fa2566b44750ca1.min.js` wastes 112 KB and produces a 120 ms long task on mobile.
- **Solution:**
  - Code-split the theme bundle so only the modules used on this page ship.
  - Tree-shake jQuery-dependent code; the page already defers scripts, so the cost is parse/execute, not blocking.
  - Audit `global.min.js` with coverage in DevTools and drop dead branches.

1F. Fix color-contrast on navigation descriptions `Accessibility`

- **Impact:** WCAG 1.4.3 (AA)

- **Problem:** axe reports a serious `color-contrast` violation on `.menu-item-630 > .navigation__description`, `.menu-item-116 > .navigation__description`, and `.menu-item-74 > .navigation__description`.
- **Solution:**
  - Raise the text/background contrast to at least 4.5:1 (3:1 for large text).
  - Example: change the muted nav description color from a light grey to `#595959` on white, or darken the background.
  - Re-run axe after the change; 10 additional nodes were flagged as "incomplete" and should be verified manually.

1G. Eliminate 27.8 s of Total Blocking Time from theme JS `Performance`

- **Impact:** TBT, TTI, INP, Max Potential FID
- **Problem:** TBT is 27,810 ms on mobile (23,860 ms desktop) with long tasks from core.js (1.72 s), jquery.js (1.08 s), and eight global.js tasks of 300–400 ms each; TTI is 43.5 s and Max Potential FID is 1,720 ms.
- **Solution:** Break up the main-thread work:
  - Audit `core.js` / `global.js` for synchronous DOM work; split into route-level chunks and load non-critical modules on interaction.
  - Drop jQuery if the theme allows, or replace with vanilla equivalents for the remaining uses.
  - Move heavy init (sliders, animations, analytics) behind `requestIdleCallback` or `IntersectionObserver`.
  - Re-measure with Lighthouse after each change; target TBT < 200 ms.

1H. Fix 12.4 s mobile LCP `Performance`

- **Impact:** LCP, Speed Index, FCP
- **Problem:** Mobile LCP is 12.4 s (desktop 2.1 s) and Speed Index is 9.38 s; the LCP breakdown and network-dependency-tree insights both score 0.00.
- **Solution:**
  - Identify the LCP element in DevTools Performance panel (likely the hero heading/image).
  - Preload the LCP resource: `<link rel="preload" as="image" href="/hero.webp" fetchpriority="high">`.
  - Remove render-blocking CSS/JS from the critical path; inline critical CSS.
  - Reduce the network dependency chain — the dependency-tree insight flags a deep chain.

1I. Defer or lazy-load reCAPTCHA `Performance`

- **Impact:** TBT, page weight, LCP
- **Problem:** reCAPTCHA accounts for 1.42 MB from gstatic.com across 7 requests, with three separate loads of recaptcha__en.js wasting 176 KB + 172 KB + 168 KB, and the slowest request taking 3.27 s.
- **Solution:**
  - Load reCAPTCHA only when the form enters the viewport or on first user interaction:

```
const io = new IntersectionObserver(([e]) => {
  if (e.isIntersecting) { loadRecaptcha(); io.disconnect(); }
});
io.observe(document.querySelector('form'));
```

  - Ensure the script is only injected once (the three duplicate loads suggest a double-enqueue bug in the Gravity Forms integration).
  - Consider a lighter alternative (hCaptcha, Turnstile) if the form is low-risk.

1J. Harden the Content-Security-Policy `Security`

- **Impact:** XSS defense-in-depth
- **Problem:** CSP is only `frame-ancestors 'self';` — no `default-src`, no `object-src 'none'`, and no script/resource restrictions. The site accepts user input via a textarea form, so XSS would persist into stored content.
- **Solution:** Deploy a nonce-based CSP with `'strict-dynamic'` rather than a flat allowlist:

```
Content-Security-Policy:
  default-src 'self';
  script-src 'nonce-{random}' 'strict-dynamic'
https://www.gstatic.com https://www.google.com;
  object-src 'none';
  base-uri 'none';
  frame-ancestors 'self';
```

  Emit the nonce server-side per request and add it to every `<script>` tag.

1K. Break up long tasks to cut 30.9 s of Total Blocking Time `Performance`

- **Impact:** TBT, TTI, INP, Max Potential FID
- **Problem:** TBT is 30,870 ms and TTI 44.2 s, with long tasks of 918 ms (global.min.js), 859 ms (core.min.js) and eight more 450–500 ms tasks on the same theme bundle.

- **Solution:**
  - Split `global.min.js` / `core.min.js` per-route and load non-critical modules on interaction (`import()` on click/scroll).
  - Audit the theme bundle for synchronous loops and heavy DOM work; move work into `requestIdleCallback` or a Web Worker.
  - Drop `jquery-migrate.min.js` if nothing depends on it.
  - Verify with Lighthouse after each change — target TBT < 200 ms.

1L. Eliminate the 0.615 layout shift from the cookie-consent modal `Performance`

- **Impact:** CLS
- **Problem:** The layout-shift culprit is `body.wp-singular > div#cc-main > div.cm-wrapper > div.cm` with a shift of 0.615 — the cookie-consent banner is injected after paint and pushes content.
- **Solution:**
  - Reserve space for the consent banner before it renders (fixed-position overlay, or a placeholder with the same height).
  - Load the `cookieconsent` UMD script with `defer` and initialise it before first paint, or render the banner server-side.
  - Re-run PSI and confirm CLS < 0.1.

1M. Trim unused JavaScript and reduce Speed Index `Performance`

- **Impact:** Speed Index, TBT, page weight
- **Problem:** 92 KB of `global.62444fa2566b44750ca1.min.js` is unused, and Speed Index is 6.59 s (score 0.38).
- **Solution:**
  - Run `npx source-map-explorer` or webpack-bundle-analyzer against the theme bundle and remove dead modules.
  - Code-split below-the-fold widgets (playlist, podcast, jeopardy, snake game embeds) so they load on interaction.
  - Defer or lazy-init the `iframemanager` and `cookieconsent` scripts from jsdelivr.

Priority 2: Important

Essential for compliance, user reach, and search visibility.

2A. Enable browser caching on the HTML document `Performance`

- **Impact:** Repeat-visit TTFB, LCP

- **Problem:** The document response sets `cache-control: max-age=0` with an `expires` equal to the request time, so every navigation re-fetches the HTML despite WP Rocket being active.
- **Solution:**
  - Configure WP Rocket (or the origin) to send a positive `Cache-Control: max-age` for anonymous HTML, e.g. `max-age=600, s-maxage=3600`.
  - Keep `no-cache` only for logged-in or cart/checkout responses.

2B. Add Subresource Integrity to cross-origin scripts `Security`

- **Impact:** Supply-chain integrity
- **Problem:** 2 of 2 cross-origin scripts (jsdelivr cookieconsent + iframemanager) lack an `integrity` attribute, so a CDN compromise would execute arbitrary code.
- **Solution:**

```
<script
src="https://cdn.jsdelivr.net/gh/orestbida/cookieconsent@3.0.1/dist
/cookieconsent.umd.js"
        integrity="sha384-…" crossorigin="anonymous" defer>
</script>
```

  Generate the hash with `openssl dgst -sha384 -binary file.js | openssl base64 -A`, or self-host the two libraries.

2C. Add HSTS preload `Security`

- **Impact:** Transport security
- **Problem:** HSTS is set (`max-age=31536000; includeSubDomains`) but lacks the `preload` directive, so the domain cannot be submitted to the preload list.
- **Solution:**

```
Strict-Transport-Security: max-age=63072000; includeSubDomains;
preload
```

  Then submit at hstspreload.org once all subdomains are HTTPS-only.

2D. Replace vague and non-descriptive link text `SEO`

- **Impact:** Link purpose, SEO
- **Problem:** The `link-text` audit fails with 4 links lacking descriptive text, and the inventory shows "read more" repeated 4 times plus repeated "design & innovation", "development" and "technical maintenance" anchors.
- **Solution:**

- Rewrite "read more" links to name the destination, e.g. "Read the Full-stack digital solutions case study".
- Where the visible text must stay short, add `aria-label` with the full purpose.
- Ensure each repeated service link points to a distinct, clearly named target.

2E. Fix structured data `SEO`

- **Impact:** Rich results eligibility
- **Problem:** The `structuredData` audit fails and the single JSON-LD block is typed "Unknown", so search engines cannot interpret the organization/service markup.
- **Solution:**
  - Validate the existing JSON-LD with the Rich Results Test and Schema Markup Validator.
  - Add an explicit `@type` (e.g. `Organization`, `WebSite`, `ProfessionalService`) with `name`, `url`, `logo` and `sameAs`.
  - Keep `@context` as `https://schema.org`.

2F. Enlarge tap targets `SEO`

- **Impact:** Mobile usability
- **Problem:** The `tapTargets` audit fails, meaning interactive elements are smaller than the 48×48 px recommended minimum on mobile.
- **Solution:**
  - Give nav links, footer links and icon buttons at least 48×48 px of tappable area (padding counts).
  - Add spacing between adjacent targets so mis-taps are less likely.

2G. Eliminate the /privacy-policy → /privacy-policy/ redirect `Performance`

- **Impact:** TTFB, LCP, FCP
- **Problem:** Browser runtime shows two document requests: `/privacy-policy` (483 ms, 0 B) followed by `/privacy-policy/` (372 ms, 21.1 KB).
- **Solution:**
  - Update internal links and the canonical to the trailing-slash URL so the redirect is skipped.
  - If the redirect is intentional, ensure it is a 301 (not 302) so browsers cache it.

2H. Fix tap targets and structured data `SEO`

- **Impact:** Mobile usability, rich results

- **Problem:** PSI modeled audits `tapTargets` and `structuredData` are failing; the single JSON-LD block is typed "Unknown".
- **Solution:**
  - Ensure interactive elements are at least 44×44 px with adequate spacing.
  - Replace the generic JSON-LD with a valid `WebPage` / `Organization` schema (`@context` , `@type` , `name` , `url`).

2I. Ship modern JavaScript to drop legacy polyfills `Performance`

- **Impact:** Bundle size, TBT
- **Problem:** `legacy-javascript-insight` estimates 10 KiB of savings from transpiled/polyfilled code.
- **Solution:**
  - Update `browserslist` to modern targets and rebuild the theme bundle.
  - Serve ES2017+ output with a `nomodule` fallback only if you still support legacy browsers.

2J. Set a real Cache-Control on the HTML document `Performance`

- **Impact:** Repeat-visit TTFB, LCP
- **Problem:** The document response is served with `cache-control: max-age=0` , forcing revalidation on every navigation despite WP Rocket being active.
- **Solution:** For a mostly-static marketing page, allow short browser caching with stale-while-revalidate:

  ```
  Cache-Control: public, max-age=600, stale-while-revalidate=86400
  ```

  Keep `max-age=0` only on pages that must reflect logged-in state.

2K. Add HSTS preload, COOP, and CORP `Security`

- **Impact:** Transport security, origin isolation
- **Problem:** HSTS is `max-age=31536000; includeSubDomains` but lacks `preload` ; Cross-Origin-Opener-Policy and Cross-Origin-Resource-Policy are both missing.
- **Solution:**

  ```
  Strict-Transport-Security: max-age=63072000; includeSubDomains;
  preload
  Cross-Origin-Opener-Policy: same-origin
  Cross-Origin-Resource-Policy: same-origin
  ```

  Then submit the domain at hstspreload.org.

2L. Label the unlabeled textarea in the contact form `Accessibility`

- **Impact:** WCAG 1.3.1, 3.3.2, screen-reader form completion
- **Problem:** Form 1 contains a `textarea` with no label (and 12 hidden inputs without labels, which are lower risk). axe also flagged 23 color-contrast nodes as 'incomplete'.
- **Solution:**
  - Add a visible `<label for="...">` or `aria-label` to the textarea.
  - Manually verify the 23 color-contrast nodes axe could not resolve (gradients/alpha) against WCAG 4.5:1.

2M. Fix the reCAPTCHA console error `Best Practices`

- **Impact:** Console errors, Best Practices score
- **Problem:** Browser runtime logged `requestStorageAccess: Permission denied` from the reCAPTCHA anchor iframe, and PSI's `errors-in-console` audit fails.
- **Solution:**
  - Update the reCAPTCHA integration to the current API version and ensure the iframe is loaded in a first-party context.
  - If the error persists, wrap the call in a try/catch and fall back to a server-side challenge.

2N. Fix failing tapTargets and structuredData audits `SEO`

- **Impact:** Mobile SEO, rich results eligibility
- **Problem:** PSI reports `tapTargets` and `structuredData` as failing modeled audits; the single JSON-LD block is typed 'Unknown'.
- **Solution:**
  - Ensure all interactive elements meet the 48×48 px tap-target minimum on mobile.
  - Replace the generic JSON-LD with a valid `Organization` / `LocalBusiness` schema including `name`, `url`, `logo`, and `contactPoint`.

2O. Strengthen the Content-Security-Policy `Security`

- **Impact:** XSS defense-in-depth
- **Problem:** CSP is `frame-ancestors 'self';` only — no `default-src`, no `object-src 'none'`, no script/resource restrictions. The site serves a WordPress admin login at `/wp-login.php`, so an XSS would be session-relevant.
- **Solution:** Deploy a nonce-based policy rather than a flat allowlist:

```
Content-Security-Policy: default-src 'self'; object-src 'none';
base-uri 'none';
  script-src 'nonce-{random}' 'strict-dynamic' https:;
  frame-ancestors 'self';
```

Generate the nonce per request in the theme and add it to every `<script>` tag.

2P. Add HSTS preload and COOP/CORP headers `Security`

- **Impact:** Transport security, origin isolation
- **Problem:** HSTS is `max-age=31536000; includeSubDomains` but lacks `preload`; Cross-Origin-Opener-Policy and Cross-Origin-Resource-Policy are both missing.
- **Solution:**

```
Strict-Transport-Security: max-age=63072000; includeSubDomains;
preload
Cross-Origin-Opener-Policy: same-origin
Cross-Origin-Resource-Policy: same-origin
```

Submit the domain at hstspreload.org once the header is live.

2Q. Fix the h1 → h3 heading skip `Accessibility`

- **Impact:** WCAG 1.3.1, screen-reader navigation
- **Problem:** axe reports a moderate `heading-order` violation on `.hero__description > h3`, and the W3C validator flags the same skip (h1 "Get to know us" → h3 "Full-stack digital agency powered by curiosity").
- **Solution:**
  - Change the hero `<h3>` to `<h2>` (or promote the h1 to a section heading and demote the hero to h2).
  - Audit the remaining 12 h3s for the same pattern — the outline shows h2 ×7 and h3 ×12 with no h4s.
  - Re-run axe and confirm the `heading-order` rule passes.

2R. Serve WebP/AVIF for the JPEG uploads `Performance`

- **Impact:** Image bytes, LCP, Speed Index
- **Problem:** The runtime loads 8 JPEGs totalling 287.5 KB, each taking ~1.03 s (e.g. `img-03-388x530.jpg` at 58.4 KB); no WebP or AVIF is served.
- **Solution:**
  - Convert uploads to WebP/AVIF and add `<picture>` fallbacks.
  - Keep the existing `srcset` / `sizes` and `width` / `height` attributes.

- Re-encode at the actual rendered size — several files are 530 px tall but may render smaller.

2S. Enable browser caching for the HTML document `Performance`

- **Impact:** Repeat-visit TTFB, LCP
- **Problem:** The document response sets `cache-control: max-age=0` with an `expires` timestamp equal to the request time, so every navigation re-fetches the full 210 KB HTML.
- **Solution:**
  - In WP Rocket, enable page caching with a sensible TTL and let the CDN serve the cached HTML.
  - Set `Cache-Control: public, max-age=600, stale-while-revalidate=86400` on the document.
  - Keep `Vary: Accept-Encoding` (already present).

Priority 3: Best Practice

Recommended for long-term maintainability.

3A. Add explicit width/height to 5 images `Accessibility`

- **Impact:** CLS
- **Problem:** 5 of 30 images lack explicit `width` / `height`; CLS is currently 0.000 but the risk is latent on slower devices and during font/image swaps.
- **Solution:**

```
<img src="…" alt="…" width="440" height="138" loading="lazy"
decoding="async">
```

  Also add `aspect-ratio` in CSS for any responsive container.

3B. Harden the Content-Security-Policy `Security`

- **Impact:** XSS defense-in-depth
- **Problem:** CSP is present but weak — `frame-ancestors 'self';` only, with no `default-src` and `object-src` not set to `'none'`. Site signals show no auth, payments, user content or e-commerce, so real XSS impact is limited and this is not a P1.
- **Solution:**
  - Add `default-src 'self'` and `object-src 'none'` as a baseline.
  - Move to a nonce + `'strict-dynamic'` policy rather than a flat allowlist:

```
Content-Security-Policy: default-src 'self'; script-src 'nonce-
{random}' 'strict-dynamic'; object-src 'none'; base-uri 'none';
frame-ancestors 'self';
```

3C. Add COOP and CORP headers `Security`

- **Impact:** Origin isolation
- **Problem:** `Cross-Origin-Opener-Policy` and `Cross-Origin-Resource-Policy` are both missing; with no auth or payment surface the exploit value is low, so this is a low-priority hardening item.
- **Solution:**

```
Cross-Origin-Opener-Policy: same-origin
Cross-Origin-Resource-Policy: same-origin
```

3D. Reduce WordPress attack surface and technology disclosure `Security`

- **Impact:** Brute-force exposure, fingerprinting
- **Problem:** The login form is served on the default `/wp-login.php` path and the `Server: Apache` header discloses the web server.
- **Solution:**
  - Move or rename the login URL (e.g. a security plugin's custom login slug) and add rate limiting / 2FA.
  - Suppress the server banner: `ServerTokens Prod` and `ServerSignature Off` in Apache.
  - Set `X-Permitted-Cross-Domain-Policies: none`.

3E. Ship source maps for first-party JavaScript `Best Practices`

- **Impact:** Debuggability
- **Problem:** The `valid-source-maps` audit fails — large first-party bundles such as core.min.js and global.min.js ship without source maps.
- **Solution:**
  - Emit `.map` files in the production build and reference them with `//# sourceMappingURL=`.
  - Restrict public access to maps if you do not want them indexed.

3F. Verify color contrast and link distinguishability flagged as incomplete `Accessibility`

- **Impact:** WCAG 1.4.3, 1.4.1

- **Problem:** axe could not determine contrast for 19 nodes and link distinguishability for 1 node (gradients/alpha backgrounds are skipped automatically).
- **Solution:**
  - Manually check the 19 flagged nodes against 4.5:1 (normal text) / 3:1 (large text) using DevTools' contrast picker.
  - Ensure in-text links are underlined or otherwise distinguishable without relying on color alone.

3G. Tighten the Content-Security-Policy `Security`

- **Impact:** XSS defense-in-depth
- **Problem:** CSP is `frame-ancestors 'self';` only — no `default-src` and no `object-src 'none'`. Site signals show no auth, payments, or user content, so XSS impact is low today.
- **Solution:**
  - Add `default-src 'self'; object-src 'none'; base-uri 'none';`.
  - If you later add login or user content, move to a nonce + `'strict-dynamic'` policy rather than a flat allowlist.

3H. Add HSTS preload, COOP and CORP `Security`

- **Impact:** Transport hardening, origin isolation
- **Problem:** HSTS is `max-age=31536000; includeSubDomains` but lacks `preload`; COOP and CORP headers are missing.
- **Solution:**

```
Strict-Transport-Security: max-age=63072000; includeSubDomains;
preload
Cross-Origin-Opener-Policy: same-origin
Cross-Origin-Resource-Policy: same-origin
```

  Submit the domain to hstspreload.org after the preload directive is live.

3I. Reduce technology disclosure and move the login path `Security`

- **Impact:** Attack-surface reduction
- **Problem:** `Server: Apache` is disclosed and the WordPress login form is served on the default `/wp-login.php` path.
- **Solution:**
  - Set `ServerTokens Prod` and `ServerSignature Off` in Apache.
  - Move the login form behind a custom path or a WAF rule, and rate-limit `/wp-login.php`.

3J. Disambiguate repeated link text `Accessibility`

- **Impact:** WCAG 2.4.4 link purpose
- **Problem:** The inventory lists repeated vague link text ("home", "work", "about", "contact", etc. ×2) and axe flags one `link-in-text-block` node as incomplete.
- **Solution:**
  - Add `aria-label` or visually hidden text to distinguish repeated nav links.
  - Ensure in-text links are distinguishable without relying on color alone (underline or bold).

3K. Set a sensible Cache-Control for the document `Performance`

- **Impact:** Repeat-visit TTFB
- **Problem:** The document response sets `cache-control: max-age=0`, forcing revalidation on every visit despite WP Rocket being active.
- **Solution:**
  - Allow short-lived browser caching with revalidation, e.g. `Cache-Control: public, max-age=600, stale-while-revalidate=86400`.
  - Keep WP Rocket's page cache for the server side.

3L. Fix W3C validation errors on hidden inputs `Best Practices`

- **Impact:** HTML validity
- **Problem:** W3C reports 2 errors: hidden inputs carry `autocomplete='off'` (first at line 896, `gform_hidden`).
- **Solution:** Remove `autocomplete` from `type="hidden"` inputs — it is invalid per the HTML spec and has no effect on hidden fields.

3M. Reduce technology disclosure and add cross-domain policy header `Security`

- **Impact:** Fingerprinting surface
- **Problem:** The `Server: Apache` header is exposed and `X-Permitted-Cross-Domain-Policies` is missing.
- **Solution:**

```
ServerTokens Prod
ServerSignature Off
Header always set X-Permitted-Cross-Domain-Policies "none"
```

3N. Ship source maps for first-party JS `Best Practices`

- **Impact:** Debuggability, error tracking

- **Problem:** PSI's `valid-source-maps` audit fails — large first-party bundles (core.js, global.js) ship without source maps.
- **Solution:** Emit `.map` files in the production build and serve them from the same origin (or a private error-tracking endpoint).

3O. Add SRI to cross-origin scripts and hide the server banner `Security`

- **Impact:** Supply-chain integrity, information disclosure
- **Problem:** 2 of 2 cross-origin scripts (jsdelivr `iframemanager` and `cookieconsent`) lack `integrity` attributes, and the `Server: Apache` header discloses the stack.
- **Solution:**
  - Add `integrity="sha384-…" crossorigin="anonymous"` to both jsdelivr `<script>` tags, or self-host the libraries.
  - Set `ServerTokens Prod` and `ServerSignature Off` in Apache to suppress the version banner.
  - Consider moving the WordPress login off `/wp-login.php` (currently on the default path).

3P. Fix structured data and tap targets `SEO`

- **Impact:** Rich results eligibility, mobile usability
- **Problem:** PSI's modeled audits flag `structuredData` and `tapTargets` as failing, and the single JSON-LD block is reported with type "Unknown".
- **Solution:**
  - Validate the JSON-LD with Google's Rich Results Test and declare an explicit `@type` (e.g. `Organization`, `WebSite`).
  - Ensure interactive elements meet the 44×44 px minimum touch-target size, especially in the nav and footer.