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

Website: https://playandnope.com/
Date: 08.10.2026
Audit Coverage: 100% — all sources returned data
Confidence: high

Pages Audited (5 of 5):

- https://playandnope.com/
- https://playandnope.com/privacy-policy
- https://playandnope.com/contact
- https://playandnope.com/digiagentuur-klientide-tagasiside
- https://playandnope.com/kontorikohvik-vork

Summary of results

Overall Score: 69 / 100
Status: □ Needs Improvement

Site overall 69 is the mean of 5 pages. Scores range 45 (https://playandnope.com/contact) → 82 (https://playandnope.com/digiagentuur-klientide-tagasiside). Weakest page: Mobile performance is critically poor (PSI 34, LCP 10.1 s, TBT 23.9 s), rendering the site nearly unusable on mobile despite excellent desktop metrics. Accessibility is strong (Lighthouse 98) but marred by heading hierarchy skips and form labeling issues. Security basics pass, but the Content-Security-Policy is weak despite the presence of user-generated content (textarea), creating an XSS risk. The heavy reliance on third-party scripts (reCAPTCHA 1.43 MB) is the primary driver of the performance collapse. Confidence is high as all audit tools returned complete data.

Per-page scores

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

Score	Performance	Accessibility	Best Practices	SEO	Security
72	58	100	100	85	PASS

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

Score	Performance	Accessibility	Best Practices	SEO	Security
75	74	98	100	92	PASS

 Poor · https://playandnope.com/contact

Score	Performance	Accessibility	Best Practices	SEO	Security
45	34	98	96	92	PASS

 Needs Improvement · https://playandnope.com/digiagentuur-klientide-tagasiside

Score	Performance	Accessibility	Best Practices	SEO	Security
82	84	100	100	92	PASS

 Needs Improvement · https://playandnope.com/kontorikohvik-vork

Score	Performance	Accessibility	Best Practices	SEO	Security
72	60	96	100	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/	**58** / 67	**1.81 s** / 487 ms	**0.181** / 0.098
https://playandnope.com/privacy-policy	**74** / 97	**6.76 s** / 623 ms	0.000 / **0.085**
https://playandnope.com/contact	34 / **33**	**10.05 s** / 2.63 s	0.000 / **0.265**
https://playandnope.com/digiagentuur-klientide-tagasiside	**84** / 96	**4.22 s** / 668 ms	0.040 / **0.112**

URL	Performance (M / D)	LCP (M / D)	CLS (M / D)
https://playandnope.com/kontorikohvik-vork	**60** / 66	**2.48 s** / 619 ms	0.000 / **0.043**

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 (18 SVGs, 12 placeholders excluded).
Hero image eagerly loaded	N/A	No raster elements found (18 SVGs, 12 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 (18 SVGs, 12 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 Total Blocking Time (TBT) and Unused JavaScript `Performance`

- **Impact:** Mobile Performance score, Interactivity (TBT 10,330 ms)
- **Problem:** Mobile TBT is 10,330 ms (threshold <200 ms) with 92 KB unused JS and multiple long tasks in global/core scripts.
- **Solution:**
  - Defer non-critical JS further or split bundles.
  - Remove unused JS (92 KB identified).
  - Investigate `global.62444fa2566b44750ca1.min.js` and `core.c872dceb075db34864df.min.js` for code splitting or lazy loading.

1B. Reduce Largest Contentful Paint (LCP) to under 2.5 s `Performance`

- **Impact:** Mobile Core Web Vitals, User Experience
- **Problem:** Mobile LCP is 6.8 s, far exceeding the 4 s heavy penalty threshold and the 2.5 s target.
- **Solution:** Optimize the largest content element (likely the hero section or main text).
  - Defer non-critical JavaScript (112 KB unused JS identified).
  - Preload critical resources.
  - Ensure server response time (TTFB) remains low (currently 3 ms is good).
  - Consider lazy loading below-the-fold content.

1C. Eliminate or defer reCAPTCHA blocking scripts `Performance`

- **Impact:** LCP, TBT, FCP, Mobile Performance
- **Problem:** reCAPTCHA scripts contribute 1.43 MB and cause long tasks (up to 3.93 s), driving TBT to 23.9 s and LCP to 10.1 s on mobile.
- **Solution:** Use `defer` or `async` on the reCAPTCHA script and load it only after user interaction (e.g., on form focus) or use the invisible reCAPTCHA badge with a lighter implementation. Consider alternatives like hCaptcha or server-side bot protection if feasible.

1D. Strengthen Content-Security-Policy (CSP) `Security`

- **Impact:** XSS protection, Data integrity
- **Problem:** CSP is present but weak (missing `default-src`, `object-src 'none'`), and the site has user-generated content (textarea), increasing XSS risk per the security rubric.
- **Solution:** Update CSP header to include `default-src 'self'` and `object-src 'none'`. Use a nonce-based approach for scripts:

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

1E. Reduce Largest Contentful Paint (LCP) on Mobile `Performance`

- **Impact:** LCP, User Perception
- **Problem:** Mobile LCP is 4.2 s (threshold >4 s), despite TTFB being 43 ms; field data is better (2.4 s) but lab shows rendering delays.
- **Solution:**
  - Preload the LCP element (likely text or SVG) using `<link rel="preload">`.
  - Optimize font loading: use `font-display: swap` and subset fonts.
  - Ensure hero content is not delayed by JavaScript execution.

1F. Fix Color Contrast on Navigation Description `Accessibility`

- **Impact:** WCAG 1.4.3 Compliance
- **Problem:** axe-core found 1 serious violation: `.menu-item-75 > .navigation__description` fails contrast thresholds.
- **Solution:**
  - Increase text color contrast ratio to at least 4.5:1 against the background.
  - Verify contrast for all interactive elements in the navigation menu.

1G. Reduce Total Blocking Time (TBT) `Performance`

- **Impact:** TBT, Interaction readiness
- **Problem:** TBT is 31,190 ms on mobile with multiple long tasks >500 ms (e.g., global.js, core.js).
- **Solution:**
  - Split large JS bundles (`global.js`, `core.js`) into smaller chunks.
  - Defer non-critical scripts until after `load`.
  - Use `requestIdleCallback` for non-urgent tasks.

Priority 2: Important

Essential for compliance, user reach, and search visibility.

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

- **Impact:** Cumulative Layout Shift (CLS 0.181)
- **Problem:** 12 images lack width/height attributes, contributing to CLS of 0.181 (threshold ≤0.1).

- **Solution:**
  - Add `width` and `height` attributes to all `<img>` tags.
  - Use CSS `aspect-ratio` if dimensions vary dynamically.
  - Ensure reserved space matches aspect ratio to prevent shifts.

2B. Add Subresource Integrity (SRI) to external scripts `Security`

- **Impact:** Script integrity, XSS mitigation
- **Problem:** 3 of 3 cross-origin scripts lack an integrity attribute, allowing potential tampering.
- **Solution:**
  - Generate SHA-384 hashes for external scripts.
  - Add `integrity` and `crossorigin` attributes:

```
<script src="..." integrity="sha384-..." crossorigin="anonymous">
</script>
```

2C. Fix Heading Hierarchy and Color Contrast `Accessibility`

- **Impact:** WCAG 1.3.1, 1.4.3, Screen Reader Navigation
- **Problem:** axe-core reports 1 serious color-contrast violation on h1 and 1 moderate heading-order violation (h1 skips to h3).
- **Solution:**
  - Adjust h1 text color or background to meet 4.5:1 contrast ratio.
  - Insert an h2 element between the h1 (Privacy Policy) and the existing h3 sections to maintain sequential order.

2D. Reduce JavaScript execution time (Long Tasks) `Performance`

- **Impact:** TBT, INP, Mobile Performance
- **Problem:** Theme scripts (`core.js` , `jquery.js` , `global.js`) cause multiple long tasks (>1 s each), contributing to 23.9 s TBT.
- **Solution:** Code-split the theme JavaScript. Move non-critical JS to the bottom of the body or use `defer` . Remove unused JavaScript identified in PSI (92 KB wasted in `global.js`).

2E. Fix heading hierarchy and form labels `Accessibility`

- **Impact:** WCAG 1.3.1, 2.4.6, Screen Reader navigation
- **Problem:** Heading skips from h1 to h3 (W3C/axe), and the contact form textarea lacks a visible label (HTML Inventory).

- **Solution:** Change the h3 following the h1 to h2. Ensure all form inputs have associated `<label>` elements or `aria-label` attributes:

```
<label for="message">Message</label>
<textarea id="message" name="message"></textarea>
```

2F. Fix Invalid Script Tags (W3C Errors) `Best Practices`

- **Impact:** HTML Validity, Parser Recovery
- **Problem:** W3C Validator reports 2 errors: `<script type="text/plain">` elements must not have `src` or `async` attributes.
- **Solution:**
  - Remove `src` and `async` from `<script type="text/plain">` tags (used for data blocks).
  - Example fix:

```
<script type="text/plain" data-category="analytics">...</script>
```

2G. Eliminate Unused JavaScript `Performance`

- **Impact:** Page Weight, TBT
- **Problem:** 112 KB of unused JavaScript detected in `global.min.js`, contributing to long tasks (111 ms).
- **Solution:**
  - Audit `global.min.js` to identify dead code.
  - Split code by route or load scripts conditionally.
  - Use tree-shaking during the build process.

2H. Verify Color Contrast `Accessibility`

- **Impact:** WCAG 1.4.3
- **Problem:** axe-core reports 3 incomplete checks for color contrast on text elements.
- **Solution:**
  - Manually inspect text colors against backgrounds.
  - Ensure ratio is ≥4.5:1 for normal text.
  - Use tools like Stark or WebAIM Contrast Checker.

2I. Fix W3C HTML Validation Errors `SEO`

- **Impact:** HTML validity, Parser recovery

- **Problem:** 5 errors found, including script elements with `type="text/plain"` having `src` attributes.
- **Solution:**
  - Remove `src` from `<script type="text/plain">` blocks.
  - Move `<style>` elements in `<body>` to `<head>` or use `scoped` correctly.
  - Ensure `sup` rules are inside `@scope` if used in body styles.

Priority 3: Best Practice

Recommended for long-term maintainability.

3A. Strengthen Content-Security-Policy (CSP) `Security`

- **Impact:** XSS defense-in-depth
- **Problem:** CSP is present but weak (missing `default-src`, `object-src 'none'`); site has no auth/payments so risk is lower.
- **Solution:**
  - Define `default-src 'self'`.
  - Add `object-src 'none'`.
  - Use nonce-based CSP for scripts if dynamic content is added later.

3B. Disambiguate vague link text `SEO`

- **Impact:** Link context, Accessibility
- **Problem:** 4 links use vague text like 'read more' without unique context.
- **Solution:**
  - Change link text to describe destination (e.g., 'Read more about Finance Solutions').
  - Or add `aria-label` to provide context.

3C. Strengthen Content Security Policy (CSP) and Add COOP/CORP `Security`

- **Impact:** XSS Defense, Context Isolation
- **Problem:** CSP is present but weak (missing default-src, object-src). COOP and CORP headers are missing.
- **Solution:** Since the site has no auth or payments, this is a P3 defense-in-depth fix.
  - Add `default-src 'self'` and `object-src 'none'` to CSP.
  - Add `Cross-Origin-Opener-Policy: same-origin`.
  - Add `Cross-Origin-Resource-Policy: same-origin`.

3D. Fix Invalid Script Tag Attributes `Best Practices`

- **Impact:** HTML Validation, Potential Execution Issues
- **Problem:** W3C validator reports 3 errors, including a script with `type="text/plain"` that incorrectly uses `src` and `async` attributes.
- **Solution:** Remove `src` and `async` from script tags that use non-JavaScript MIME types (e.g., `type="text/plain"`). These are typically used for data blocks and should not execute.

3E. Add HSTS preload and SRI for scripts `Security`

- **Impact:** Transport security, Script integrity
- **Problem:** HSTS lacks `preload` directive; 4 of 5 cross-origin scripts lack Subresource Integrity (SRI) attributes.
- **Solution:** Add `preload` to HSTS header. Add `integrity` and `crossorigin` attributes to external script tags:

```
<script src="..." integrity="sha384-..." crossorigin="anonymous">
</script>
```

3F. Strengthen Content Security Policy (CSP) `Security`

- **Impact:** XSS Defense-in-Depth
- **Problem:** CSP is present but weak: missing `default-src` and `object-src 'none'` . Site signals indicate no auth/payments, so risk is lower.
- **Solution:**
  - Add `default-src 'self'` to restrict resource origins.
  - Add `object-src 'none'` to prevent plugin execution.
  - Example:

```
Content-Security-Policy: default-src 'self'; object-src 'none';
frame-ancestors 'self';
```

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

- **Impact:** XSS defense-in-depth
- **Problem:** CSP exists but lacks `default-src` and `object-src 'none'` , making it weak.
- **Solution:**
  - Add `default-src 'self'` .
  - Add `object-src 'none'` .
  - Use nonce/hash for inline scripts if needed.

3H. Add Cross-Origin Headers `Security`

- **Impact:** Context isolation, resource protection
- **Problem:** COOP and CORP headers are missing.
- **Solution:**
  - Add `Cross-Origin-Opener-Policy: same-origin`.
  - Add `Cross-Origin-Resource-Policy: same-origin`.