Audit Report: Perfectly formed web development team - gotoAndPlay

Website: https://play.ee/
Date: 13.10.2026
Audit Coverage: 100% — all sources returned data
Confidence: high

Pages Audited (5 of 5):

- https://play.ee/
- https://play.ee/privacy-policy
- https://play.ee/team
- https://play.ee/et/privaatsustingimused
- https://play.ee/wordpress-support-service

Summary of results

Overall Score: 84 / 100
Status: Needs Improvement

Site overall 84 is the mean of 5 pages. Scores range 74 (https://play.ee/et/privaatsustingimused) → 88 (https://play.ee/team). Weakest page: PSI mobile 74 vs desktop 99 — the gap is driven almost entirely by CLS 0.462 on mobile (desktop 0.004), with the footer as the culprit (two shifts of 0.234 and 0.226). LCP 2.7 s and FCP 2.12 s sit in the warning band, but TBT 0 ms and TTFB 35 ms are excellent, so the performance drag is concentrated in one fixable layout-shift problem. Accessibility 93 is solid but carries 1 serious axe violation (color-contrast on h1 > .heading__main and .button) and 1 moderate (heading-order h1→h4), echoed by 5 W3C errors (4 script type/defer, 1 heading skip). Security basics PASS, but real gaps remain: TLS expires in 13 days, CSP is frame-ancestors-only (no default-src, no object-src 'none'), COOP/CORP missing, SRI absent on 2 cross-origin jsdelivr scripts, and login is served on the default /wp-login.php path. Image budget is effectively N/A (1 base64 placeholder), so the score is not dragged by image assets.

Per-page scores

 Needs Improvement · https://play.ee/

Score	Performance	Accessibility	Best Practices	SEO	Security
88	95	96	100	92	PASS

 Needs Improvement · https://play.ee/privacy-policy

Score	Performance	Accessibility	Best Practices	SEO	Security
86	95	93	100	100	PASS

 Needs Improvement · https://play.ee/team

Score	Performance	Accessibility	Best Practices	SEO	Security
88	94	96	100	100	PASS

 Needs Improvement · https://play.ee/et/privaatsustingimused

Score	Performance	Accessibility	Best Practices	SEO	Security
74	74	93	100	100	PASS

 Needs Improvement · https://play.ee/wordpress-support-service

Score	Performance	Accessibility	Best Practices	SEO	Security
84	92	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://play.ee/	**95** / 100	**2.60 s** / 617 ms	0.001 / **0.007**
https://play.ee/privacy-policy	**95** / 99	**2.57 s** / 705 ms	0.002 / **0.005**
https://play.ee/team	**94** / 99	**2.75 s** / 665 ms	**0.036** / 0.003

URL	Performance (M / D)	LCP (M / D)	CLS (M / D)
https://play.ee/et/privaatsustingimused	**74** / 99	**2.73 s** / 674 ms	**0.462** / 0.004
https://play.ee/wordpress-support-service	**92** / 99	**2.94 s** / 767 ms	**0.017** / 0.001

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 (50 SVGs excluded).
Hero image eagerly loaded	N/A	No raster elements found (50 SVGs 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 (50 SVGs 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. Renew the TLS certificate before it expires `Security`

- **Impact:** Site availability, HTTPS trust
- **Problem:** The TLS certificate expires in 13 days, which will break HTTPS for every visitor if not renewed.
- **Solution:** Renew the certificate now and automate renewal (e.g. certbot / Let's Encrypt with a systemd timer or cron).

  ```
  sudo certbot renew --dry-run
  ```

  Add monitoring/alerting for expiry < 30 days.

1B. Fix color-contrast on the hero heading and primary button `Accessibility`

- **Impact:** WCAG 1.4.3, axe serious violation, PSI accessibility score
- **Problem:** axe reports a serious `color-contrast` violation on `h1 > .heading__main` and `.button`; PSI's `color-contrast` audit scores 0.00 at weight 7.
- **Solution:** Raise contrast to ≥4.5:1 for normal text (≥3:1 for large text):
  - Darken the hero heading color or lighten its background.
  - Adjust the `.button` background/text pair so the ratio passes.
  - Re-run axe after the change; 8 additional nodes were flagged as *incomplete* (alpha/gradient) — verify those manually.

1C. Fix color-contrast on the h1 and .how-we-do heading `Accessibility`

- **Impact:** WCAG 1.4.3, axe serious violation
- **Problem:** axe reports 1 serious color-contrast violation on `h1 > .heading__main` and `.how-we-do__intro > .heading--primary.heading > .heading__main`; PSI also flags `color-contrast` (score 0.00, weight 7).
- **Solution:** Raise the foreground/background contrast ratio to ≥4.5:1 for normal text (≥3:1 for large text). Inspect the computed color on `.heading__main` and darken the text or lighten the background until it passes. Verify with DevTools' contrast checker or axe DevTools after the change.

1D. Reduce mobile LCP (2.7 s) and FCP (2.14 s) `Performance`

- **Impact:** LCP, FCP, Speed Index
- **Problem:** Mobile LCP is 2.7 s and FCP 2.14 s — both just above the 'good' thresholds. PSI flags `document-latency-insight` (est. savings 720 ms), `forced-reflow-insight`, and `network-dependency-tree-insight` as high-priority.
- **Solution:**

- Address the document-latency insight: the HTML document request is the slowest request (627 ms) — check server-side rendering / TTFB on the origin and warm the WP Rocket cache for `/team/`.
- Break the network dependency chain: preload the LCP font/hero asset and inline critical CSS.
- Fix the forced reflow: batch DOM reads/writes in the theme JS (avoid layout thrash on load).
- Re-measure with PSI mobile after each change.

1E. Eliminate the footer layout shift driving CLS 0.462 `Performance`

- **Impact:** CLS, mobile Performance score
- **Problem:** Mobile CLS is 0.462 (desktop 0.004), with two shifts on body.privacy-policy > div#page > div.main > footer.footer of 0.234 and 0.226 — the single largest drag on the mobile score.
- **Solution:**
  - Reserve space for the footer before it renders: give `footer.footer` a `min-height` matching its rendered height, or wrap it in a container with `content-visibility: auto; contain-intrinsic-size: <h>`.
  - If the shift comes from late-loading webfonts (fast.fonts.net is loaded), add `font-display: optional` or `size-adjust` / `ascent-override` on the `@font-face` so the fallback metrics match.
  - Avoid injecting content above the footer after load (cookie banner, iframe manager) — render it in a fixed-position layer instead of in flow.
  - Verify with DevTools Performance → Layout Shifts after the change.

1F. Fix color contrast on the H1 and primary button `Accessibility`

- **Impact:** WCAG 1.4.3, axe serious violation, Accessibility score
- **Problem:** axe reports a serious color-contrast violation on `h1 > .heading__main` and `.button`; PSI's `color-contrast` audit scores 0.00 (weight 7).
- **Solution:**
  - Raise the foreground/background contrast ratio to ≥4.5:1 for body text and ≥3:1 for large text (≥24 px or ≥18.66 px bold).
  - Check the actual computed colors in DevTools; the H1 likely uses a light gray or a semi-transparent overlay on a hero image.
  - Re-run axe after the change — 10 additional nodes were flagged as "incomplete" (alpha/gradient backgrounds) and should be verified manually.

1G. Fix color-contrast failures on buttons and capability headings `Accessibility`

- **Impact:** WCAG 1.4.3 (AA), Lighthouse accessibility score
- **Problem:** axe-core reports 1 serious color-contrast violation across 15+ nodes, including `.button--tertiary > .button__inner > .button__text` and the four `.capabilities__heading .heading__small` elements; 14 more nodes are 'incomplete' (axe could not determine).
- **Solution:** Raise foreground/background contrast to ≥4.5:1 for body text and ≥3:1 for large text:
  - Audit the tertiary button palette (likely a light accent on white).
  - Darken `.capabilities__heading .heading__small` text or lighten its background.
  - Manually verify the 14 'incomplete' nodes (alpha/gradient backgrounds axe skips) with a contrast checker.

1H. Renew TLS certificate before it expires in 13 days `Security`

- **Impact:** Site availability, HTTPS trust
- **Problem:** Security basics reports 'TLS certificate expires in 13 days' — an expired cert will hard-fail every visitor's browser.
- **Solution:** Renew the certificate now and automate renewal (e.g. certbot / Let's Encrypt with a systemd timer or cron). Add monitoring that alerts at 30/14/7 days remaining.

Priority 2: Important

Essential for compliance, user reach, and search visibility.

2A. Eliminate render-blocking resources `Performance`

- **Impact:** FCP, LCP, Speed Index
- **Problem:** PSI flags render-blocking-insight with an estimated 860 ms savings and a network-dependency-tree issue; mobile FCP is 1.96 s and LCP 2.6 s.
- **Solution:**
  - Inline critical CSS for above-the-fold content and defer the rest.
  - Add `defer` or `async` to non-critical scripts (only 3 of 23 scripts currently use defer).
  - Preconnect to `fast.fonts.net` and `cdn.jsdelivr.net` (already 1 preconnect present).

- Consider `media="print" onload="this.media='all'"` for non-critical stylesheets.

2B. Fix color contrast on flagged elements `Accessibility`

- **Impact:** WCAG 1.4.3, PSI accessibility score
- **Problem:** PSI's color-contrast audit scored 0.00 (weight 7) and axe flagged 37 nodes as 'incomplete' because it could not determine contrast (likely alpha/gradients).
- **Solution:**
  - Manually inspect the 37 flagged nodes in DevTools' contrast checker.
  - Raise text/background contrast to at least 4.5:1 (3:1 for large text).
  - Avoid semi-transparent text over images; add a solid backdrop or darken the text color.

2C. Replace vague 'read more' link text `SEO`

- **Impact:** SEO link-text audit, screen-reader link purpose (WCAG 2.4.4)
- **Problem:** PSI's link-text audit failed with 9 links lacking descriptive text; the HTML inventory shows 'read more' repeated 9 times.
- **Solution:** Give each link a unique, descriptive label:

  ```
  <a href="/case-study/foo" aria-label="Read the Foo case study">Read more</a>
  ```

  Or replace the visible text with the destination name.

2D. Fix failing tap targets `SEO`

- **Impact:** Mobile usability, SEO
- **Problem:** PSI's SEO category fails the `tapTargets` audit on mobile.
- **Solution:**
  - Ensure every interactive element is at least 44×44 px (WCAG 2.5.8).
  - Add padding to small icon links and language switchers ('eng'/'est' anchors).
  - Verify spacing between adjacent targets is ≥8 px.

2E. Set a real Cache-Control policy on the document `Performance`

- **Impact:** Repeat-visit TTFB, LCP
- **Problem:** The document response ships `cache-control: max-age=0`, forcing revalidation on every navigation despite WP Rocket being active.

- **Solution:** Configure WP Rocket (or the origin) to send a positive max-age for the HTML document, e.g.:

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

  Keep `max-age=0` only for logged-in or personalized responses.

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

- **Impact:** Supply-chain integrity of CDN scripts
- **Problem:** 2 of 2 cross-origin scripts (jsdelivr cookieconsent and iframemanager) lack an `integrity` attribute.
- **Solution:** Pin versions and add SRI hashes:

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

  Better still, self-host these small libraries.

2G. Move the WordPress login off /wp-login.php `Security`

- **Impact:** Brute-force / credential-stuffing exposure
- **Problem:** The security basics table flags 'Login not on default path' as a warn — the login form is served on the default /wp-login.php path.
- **Solution:**
  - Rename the login URL (e.g. WPS Hide Login) or restrict /wp-login.php by IP.
  - Enable rate limiting and 2FA on the login form.
  - Ensure xmlrpc.php stays blocked (already passing).

2H. Cache the document instead of `max-age=0` `Performance`

- **Impact:** Repeat-visit TTFB, FCP, LCP
- **Problem:** The document response sets `cache-control: max-age=0`, forcing revalidation on every navigation even though the page is a static privacy policy.
- **Solution:** Serve a positive max-age with revalidation:

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

  WP Rocket is already active — confirm the page-cache TTL is not being overridden by the origin.

2I. Fix heading-order skip (h1 → h4) `Accessibility`

- **Impact:** WCAG 1.3.1, screen-reader outline, W3C validity
- **Problem:** The document jumps from `h1` to `h4` ("About the cookies used on our website"), flagged by both axe (`heading-order`, moderate) and the W3C validator.
- **Solution:** Use sequential levels:

  ```
  <h1>About our privacy policy</h1>
  <h2>About the cookies used on our website</h2>
  <h2>Manage cookie preferences</h2>
  ```

  Adjust CSS selectors so visual sizing is preserved.

2J. Renew the TLS certificate (expires in 13 days) `Security`

- **Impact:** Availability, trust
- **Problem:** Security basics reports the TLS certificate expires in 13 days — an outage risk if renewal fails.
- **Solution:**
  - Confirm automated renewal (certbot / ACME) is scheduled and working.
  - Add expiry monitoring with alerting at 30/14/7 days.

2K. Fix document cache-control (max-age=0) `Performance`

- **Impact:** Repeat-visit TTFB, LCP
- **Problem:** The document response ships `cache-control: max-age=0` even though WP Rocket is detected — repeat visitors revalidate the HTML on every navigation.
- **Solution:** Configure WP Rocket (or the origin) to send a positive `max-age` for the HTML document, e.g.:

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

  Keep `max-age=0` only for logged-in / dynamic pages.

2L. Fix 4 W3C errors from Rocket Lazyload script tags `Best Practices`

- **Impact:** HTML validity, parser correctness
- **Problem:** W3C reports 4 errors: `A "script" element with a "type" attribute whose value is neither a JavaScript MIME type, "module", "importmap", nor "speculationrules" (i.e., a data block) must not have a "defer" attribute` — first at line 1738 on `type="text/rocketlazyloadscript"`.

- **Solution:** Rocket Lazyload rewrites `<script>` tags to `type="text/rocketlazyloadscript"` and keeps `defer`, which is invalid HTML. Either:
  - Drop the `defer` attribute on the rewritten tags (Lazyload will re-add it on restore), or
  - Switch to a valid pattern such as `type="application/json"` data blocks plus a small loader, or
  - Update WP Rocket to a version that emits valid markup.

2M. Renew TLS certificate (expires in 13 days) `Security`

- **Impact:** Availability, trust
- **Problem:** Security basics reports `TLS certificate expires in 13 days` (warn tier).
- **Solution:** Renew the certificate now and enable automated renewal (Let's Encrypt / ACME or your CA's auto-renew). Add a monitoring alert at 30 days before expiry.

2N. Fix heading order skip (h1 → h4) `Accessibility`

- **Impact:** WCAG 1.3.1, screen-reader navigation, W3C validity
- **Problem:** The heading outline jumps from h1 straight to h4 ("Meie veebilehel kasutatavatest küpsistest"), skipping two levels — flagged by axe (moderate), PSI (`heading-order` score 0.00), and W3C.
- **Solution:**
  - Change the two `<h4>` elements to `<h2>` (or insert `<h2>` / `<h3>` levels) so the outline descends sequentially: h1 → h2 → h3.
  - Keep visual sizing via CSS classes rather than heading level.
  - Re-check the outline in the HTML Inventory after the change.

2O. Renew the TLS certificate before it expires `Security`

- **Impact:** Availability, transport security
- **Problem:** The TLS certificate expires in 13 days — an expired cert will hard-fail all HTTPS connections and break the site for every visitor.
- **Solution:**
  - Renew the certificate now (Let's Encrypt / your CA) and confirm auto-renewal is configured.
  - Add a monitoring alert at 30/14/7 days before expiry.
  - After renewal, verify the chain and protocol with `openssl s_client -connect play.ee:443 -servername play.ee`.

2P. Fix document caching (Cache-Control: max-age=0) `Performance`

- **Impact:** Repeat-visit TTFB, FCP, LCP
- **Problem:** The document response sets `cache-control: max-age=0` with an `expires` in the past, forcing revalidation on every visit even though WP Rocket is detected and the page is a static privacy policy.
- **Solution:**
  - In WP Rocket (or via `.htaccess`), set a positive `max-age` for HTML, e.g. `Cache-Control: public, max-age=3600, stale-while-revalidate=86400`.
  - Keep `Vary: Accept-Encoding` and ensure the cache is purged on content updates.
  - Confirm the canonical URL is warmed after deploy so the first visitor doesn't pay the cold-start cost.

2Q. Fix W3C script type/defer errors `Best Practices`

- **Impact:** HTML validity, spec compliance
- **Problem:** W3C reports 4 errors: `<script type="text/rocketlazyloadscript" ... defer>` — a data-block script type must not carry `defer`.
- **Solution:**
  - WP Rocket's lazyload rewrites `type` to `text/rocketlazyloadscript`; the `defer` attribute is invalid on a non-JS MIME type.
  - Either drop `defer` from those tags or configure WP Rocket to use a valid JS MIME type (`text/javascript`) with `defer`.
  - Re-validate at validator.w3.org after the change.

2R. Reduce mobile LCP / FCP via document latency and dependency chain `Performance`

- **Impact:** LCP (2.9 s mobile), FCP (2.19 s mobile), Speed Index
- **Problem:** PSI flags `document-latency-insight` (est. 670 ms savings), `forced-reflow-insight`, and `network-dependency-tree-insight` as high-priority; mobile LCP is 2.9 s vs 0.8 s on desktop.
- **Solution:**
  - Serve the HTML document from cache/CDN edge (WP Rocket is active but `cache-control: max-age=0` on the document forces revalidation — raise max-age for anonymous visitors).
  - Flatten the critical request chain: preconnect to `fast.fonts.net` and `cdn.jsdelivr.net` (only 2 preconnects currently, 0 preloads).

- Preload the LCP image/font and eliminate forced reflow by batching DOM reads/writes in theme JS.

2S. Add explicit width/height to 16 images `Performance`

- **Impact:** CLS, layout stability
- **Problem:** HTML inventory shows 16 of 31 images lack explicit width/height attributes; PSI already records a 0.017 layout shift from `div.canvas__inner > section.section > div.section__content`.
- **Solution:** Add intrinsic dimensions (or `aspect-ratio` in CSS) to every `<img>`:

```
<img src="…" alt="…" width="942" height="629">
```

For SVGs, set `width` / `height` attributes or a CSS `aspect-ratio` on the wrapper.

2T. Use a single

per page `SEO`

- **Impact: Document outline, screen-reader navigation**
- **Problem: Heading outline shows h1 ×2 — 'Your worry-free WordPress website' and 'additional services to upgrade your online presence'.**
- **Solution: Demote the second h1 to an h2 (or restructure the section so it uses an h2**

heading). Keep exactly one h1 per page.

2U. Add HSTS preload directive

`Security`

- **Impact: Transport security, preload-list eligibility**
- **Problem: HSTS is `max-age=31536000;` `includeSubDomains` but lacks `preload`; Security basics marks 'HSTS preload' as fail.**
- **Solution: Append `preload` and submit to hstspreload.org:**

```
Strict-Transport-Security: max-age=31536000; includeSubDomains;
preload
```

Confirm all subdomains are HTTPS-only before submitting.

Priority 3: Best Practice

Recommended for long-term maintainability.

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

- **Impact:** XSS defense-in-depth

- **Problem:** CSP is present but weak: only `frame-ancestors 'self'` is set — no `default-src`, no `object-src 'none'`. Site signals show no auth, payments, or user content, so real XSS impact is low.
- **Solution:** Add baseline directives now and move to nonce-based script-src when you add any dynamic surface:

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

  For scripts, prefer `script-src 'nonce-{random}' 'strict-dynamic'` over a flat allowlist.

3B. Add HSTS preload and COOP/CORP headers `Security`

- **Impact:** Transport hardening, origin isolation
- **Problem:** HSTS lacks the `preload` directive, and Cross-Origin-Opener-Policy and Cross-Origin-Resource-Policy are both missing.
- **Solution:**

  ```
  Strict-Transport-Security: max-age=31536000; includeSubDomains;
  preload
  Cross-Origin-Opener-Policy: same-origin
  Cross-Origin-Resource-Policy: same-origin
  ```

  Submit the domain at hstspreload.org once the preload directive is live.

3C. Remove unused CSS and JavaScript `Performance`

- **Impact:** FCP, LCP, render-blocking
- **Problem:** PSI reports 30 KiB of unused CSS and 23 KB of unused jQuery (`jquery.bfe1bb19d13b3c17b682.min.js`).
- **Solution:**
  - Audit the theme CSS with Coverage and split per-template stylesheets.
  - Check whether jQuery is still required; if not, drop it or replace usages with vanilla JS.
  - WP Rocket's 'Remove Unused CSS' feature can automate part of this.

3D. Fix W3C script type/defer errors `Best Practices`

- **Impact:** HTML validity
- **Problem:** W3C reports 4 errors: a `<script>` with a non-JS `type` (e.g. `text/rocketlazyloadscript`) must not carry `defer`.

- **Solution:** WP Rocket's lazyload script type is not a JavaScript MIME type, so `defer` is invalid. Either:
  - Remove `defer` from those `text/rocketlazyloadscript` tags, or
  - Configure WP Rocket to use a valid JS MIME type with `defer`.

3E. Add structured data for the agency `SEO`

- **Impact:** Rich results eligibility
- **Problem:** PSI's `structuredData` audit fails; the single JSON-LD block has an unrecognized type.
- **Solution:** Add a valid `Organization` / `ProfessionalService` JSON-LD block with name, url, logo, and sameAs social profiles:

```
{"@context":"https://schema.org","@type":"Organization","name":"got
oAndPlay","url":"https://play.ee/"}
```

3F. Suppress the Server header `Security`

- **Impact:** Technology disclosure
- **Problem:** The response discloses `Server: Apache`, which the security basics table flags as a warn.
- **Solution:**

```
ServerTokens Prod
ServerSignature Off
```

  Or strip the header at the reverse proxy / CDN edge.

3G. Add explicit dimensions to the remaining image `Performance`

- **Impact:** CLS
- **Problem:** 1 image lacks explicit width/height attributes (the rest of the 50 SVGs are dimensioned).
- **Solution:** Add `width` and `height` (or an `aspect-ratio` CSS rule) to the remaining image so the browser can reserve space before paint.

3H. Strengthen the Content-Security-Policy `Security`

- **Impact:** XSS defense-in-depth
- **Problem:** CSP is present but only sets `frame-ancestors 'self'` — no `default-src`, no `object-src 'none'`, so it provides no script/resource restriction. Site signals show no auth, payments, or user content, so XSS impact is low.

- **Solution:** Add a nonce-based 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';
  ```

  Roll out in report-only mode first to catch breakage.

3I. Add HSTS preload, COOP, and CORP headers `Security`

- **Impact:** Transport hardening, origin isolation
- **Problem:** HSTS lacks the `preload` directive; `Cross-Origin-Opener-Policy` and `Cross-Origin-Resource-Policy` are missing.
- **Solution:**

  ```
  Strict-Transport-Security: max-age=31536000; includeSubDomains;
  preload
  Cross-Origin-Opener-Policy: same-origin
  Cross-Origin-Resource-Policy: same-origin
  ```

  Submit to hstspreload.org once the preload directive is live.

3J. Add SRI to cross-origin scripts `Best Practices`

- **Impact:** Supply-chain integrity
- **Problem:** 2 of 2 cross-origin scripts (jsDelivr cookieconsent / iframemanager) lack an `integrity` attribute.
- **Solution:** Pin versions and add hashes:

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

3K. Fix W3C `script type` / `defer` errors `Best Practices`

- **Impact:** HTML validity
- **Problem:** W3C reports 4 errors: a `script` with a non-JS `type` (WP Rocket's `text/rocketlazyloadscript`) must not carry `defer`.
- **Solution:** Remove `defer` from the Rocket-lazyloaded script tags, or let WP Rocket emit the correct attributes — the loader already handles deferred execution.

3L. Disambiguate repeated link text `SEO`

- **Impact:** Link-purpose clarity, WCAG 2.4.4

- **Problem:** Repeated vague anchors: "view our instagram feed" ×3, "visit our facebook page" ×3, "visit our linkedin page" ×3, "eng"/"est" ×3.
- **Solution:** Add `aria-label` or visually-hidden context:

```
<a href="…" aria-label="View our Instagram feed">view our instagram feed</a>
```

3M. Move the WordPress login off the default path `Security`

- **Impact:** Brute-force surface
- **Problem:** Security basics flags the login form on the default `/wp-login.php` path.
- **Solution:**
  - Rename the login URL (e.g. WPS Hide Login) or restrict `/wp-login.php` by IP.
  - Enforce 2FA and rate limiting on the login endpoint.

3N. Harden CSP (default-src, object-src 'none') `Security`

- **Impact:** XSS defense-in-depth
- **Problem:** CSP is present but weak: `frame-ancestors 'self';` only — no `default-src`, no `object-src 'none'`. Site signals show no auth, payments, or user content, so XSS attack surface is minimal and this is P3 rather than P1.
- **Solution:** If/when you add login or user content, deploy a nonce-based CSP 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';
```

  For a brochure site as-is, prioritize the P1/P2 items above.

3O. Add COOP/CORP and HSTS preload `Security`

- **Impact:** Origin isolation, HSTS preload list
- **Problem:** Security basics flags `Cross-Origin-Opener-Policy missing`, `Cross-Origin-Resource-Policy missing`, and `HSTS lacks preload`.
- **Solution:**

```
Cross-Origin-Opener-Policy: same-origin
Cross-Origin-Resource-Policy: same-origin
Strict-Transport-Security: max-age=31536000; includeSubDomains; preload
```

  Submit the domain to hstspreload.org after adding `preload`.

3P. Remove unused jQuery (23 KB) `Performance`

- **Impact:** JS bytes, TBT
- **Problem:** PSI flags `jquery.bfe1bb19d13b3c17b682.min.js — 23 KB wasted`; the theme loads jQuery but only a fraction is used.
- **Solution:** Audit theme JS for jQuery usage and replace remaining calls with vanilla JS, or drop the dependency entirely. If jQuery is still needed, load it only on pages that require it.

3Q. Add captions to `Accessibility`

- **Impact:** WCAG 1.2.2
- **Problem:** axe reports 7 incomplete `video-caption` nodes — it could not determine whether captions exist.
- **Solution:** Add `<track kind="captions">` to each `<video>` element, or mark decorative videos with `aria-hidden="true"` and no audio.

3R. Fix tap targets, structured data, and vague link text `SEO`

- **Impact:** Mobile usability, rich results
- **Problem:** PSI flags `tapTargets` and `structuredData` as failing modeled SEO audits; HTML inventory shows repeated vague link text (`view our instagram feed` ×3, `visit our facebook page` ×3, `visit our linkedin page` ×3, `eng` ×3, `est` ×3).
- **Solution:**
  - Ensure interactive elements are ≥44×44 px with adequate spacing.
  - Add valid JSON-LD (the current block is typed `Unknown`) — use `Organization` / `Person` schema for the team page.
  - Disambiguate repeated link text with `aria-label` or visible context, e.g. `<a href="…" aria-label="Play.ee on Instagram">view our instagram feed</a>`.

3S. Remove unused jQuery and reduce script weight `Performance`

- **Impact:** JS bytes, main-thread work
- **Problem:** PSI flags 23 KB of wasted jQuery (`jquery.bfe1bb19d13b3c17b682.min.js`); 26 scripts total with 1 render-blocking.

- **Solution:**
  - Audit whether jQuery is still required by the theme; if not, drop it.
  - Defer or async the remaining non-critical scripts.
  - Consider replacing jQuery-dependent snippets with vanilla JS.

3T. Disambiguate repeated vague link text `Accessibility`

- **Impact:** WCAG 2.4.4, screen-reader link lists
- **Problem:** Repeated link text appears 3× each: "vaata meie instagrami", "külasta meie facebooki lehte", "külasta meie linkedin lehte", plus "eng"/"est" and "privaatsustingimused" ×2.
- **Solution:**
  - Add `aria-label` to each icon/social link to distinguish them, e.g. `<a href="..." aria-label="Instagram – gotoAndPlay">`.
  - For language switchers, use `aria-label="Switch to English"` / `"Lülitu eesti keelele"`.

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

- **Impact:** XSS defense-in-depth
- **Problem:** CSP is only `frame-ancestors 'self';` — missing `default-src` and `object-src 'none'`. Site signals show no auth, payments, or user content, so XSS impact is limited and this is P3 per the rubric.
- **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';
  ```

  Roll out in Report-Only first to catch breakage from WP Rocket's inline scripts.

3V. Add COOP and CORP headers `Security`

- **Impact:** Cross-origin isolation, side-channel mitigation
- **Problem:** Cross-Origin-Opener-Policy and Cross-Origin-Resource-Policy are both missing; Security basics marks both as fail.
- **Solution:**

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

  Low priority for a brochure site with no auth or payments.

3W. Fix W3C errors from rocketlazyloadscript attributes `Best Practices`

- **Impact:** HTML validity, standards compliance
- **Problem:** W3C reports 5 errors — 4× `defer` and 1× `async` on `<script type="text/rocketlazyloadscript">`, which is a data block, not a JavaScript MIME type.
- **Solution:** WP Rocket rewrites script tags to `type="text/rocketlazyloadscript"` and moves `src` to `data-rocket-src`. Either:
  - Exclude these scripts from WP Rocket's JS delay/optimization so they keep `type="text/javascript"`, or
  - Configure WP Rocket to strip the `async` / `defer` attributes on rewritten tags (they are meaningless on a data block).

3X. Move login off /wp-login.php and hide the server header `Security`

- **Impact:** Attack-surface reduction, information disclosure
- **Problem:** Security basics warns 'Login form is served on the default /wp-login.php path' and 'Response headers disclose server technology' (`Server: Apache`).
- **Solution:**
  - Rename the login URL (e.g. WPS Hide Login plugin) or front it with a WAF rule limiting /wp-login.php.
  - Suppress the Server header:

```
ServerTokens Prod
ServerSignature Off
```

3Y. Trim unused jQuery and resolve preload warning `Performance`

- **Impact:** JS bytes, console cleanliness
- **Problem:** PSI reports 23 KB of unused JavaScript in `jquery.bfe1bb19d13b3c17b682.min.js`; the runtime capture logs 'Couldn't load preload assets'.
- **Solution:**
  - Audit theme JS for jQuery usage and drop the dependency where possible (or replace with vanilla equivalents).
  - Fix the preload warning by ensuring every `<link rel="preload">` points to a resource that is actually fetched (check the font preloads for `fast.fonts.net`).