Audit Report: Perfectly formed web development team - gotoAndPlay

Website: https://play.ee/
Date: 02.08.2026
Audit Coverage: 100% — all sources returned data
Confidence: high

Pages Audited (5 of 5):

- https://play.ee/
- https://play.ee/web-development-case-studies/
- https://play.ee/software-development-work-index/
- https://play.ee/wordpress-support-service/
- https://play.ee/web-development-in-estonia/

Summary of results

Overall Score: 76 / 100
Status: Needs Improvement

Site overall 76 is the mean of 5 pages. Scores range 72 (https://play.ee/wordpress-support-service/) → 80 (https://play.ee/web-development-in-estonia/). Weakest page: Mobile performance is excellent (94/100) with strong Core Web Vitals, though LCP sits at 2.6 s just above the 2.5 s threshold. Security configuration is the weakest area, with a 20/100 header grade and HTTP traffic failing to redirect to HTTPS. Accessibility has two serious violations (contrast, link names) and missing landmarks, while HTML validation shows 7 errors including parser recovery failure. The overall score reflects high technical performance offset by significant security and compliance debt.

Per-page scores

 Needs Improvement · https://play.ee/

Score	Performance	Accessibility	Best Practices	SEO	Security
78	97	100	100	92	20

 Needs Improvement · https://play.ee/web-development-case-studies/

Score	Performance	Accessibility	Best Practices	SEO	Security
74	94	90	96	100	20

 Needs Improvement · https://play.ee/software-development-work-index/

Score	Performance	Accessibility	Best Practices	SEO	Security
74	95	90	96	100	20

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

Score	Performance	Accessibility	Best Practices	SEO	Security
72	94	91	100	100	20

 Needs Improvement · https://play.ee/web-development-in-estonia/

Score	Performance	Accessibility	Best Practices	SEO	Security
80	97	94	100	100	20

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/	**97** / 100	**2.18 s** / 533 ms	0.002 / **0.008**
https://play.ee/web-development-case-studies/	**94** / 100	**2.58 s** / 546 ms	0.000 / **0.003**
https://play.ee/software-development-work-index/	**95** / 100	**2.51 s** / 597 ms	**0.041** / 0.003
https://play.ee/wordpress-support-service/	**94** / 100	**2.58 s** / 545 ms	0.000 / **0.003**
https://play.ee/web-development-in-estonia/	**97** / 100	**2.25 s** / 574 ms	0.002 / **0.005**

Optimization Checklist

2 of 2 passing — 2 pass · 0 warn · 0 fail · 5 n/a

Item	Status	Detail
Page caching plugin / CDN active	**Pass**	Caching plugin detected (WP Rocket)
Images lazy-loaded	N/A	No raster elements found (39 SVGs, 13 placeholders excluded).
Hero image eagerly loaded	N/A	No raster elements found (39 SVGs, 13 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 (39 SVGs, 13 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. Force HTTPS Redirect `Security`

- **Impact:** Transport security, downgrade attacks
- **Problem:** Security Headers report states 'http://play.ee/ does not redirect to HTTPS', leaving users vulnerable to downgrade attacks.
- **Solution:** Configure the web server (Apache/Nginx) to redirect all HTTP traffic to HTTPS:

```
RewriteEngine On
RewriteCond %{HTTPS} off
```

```
RewriteRule ^(.*)$ https://%{HTTP_HOST}%{REQUEST_URI} [L,R=301]
```

1B. Force HTTPS redirect on HTTP requests `Security`

- **Impact:** Transport security, MITM protection
- **Problem:** Security Headers audit reports 'HTTP redirect: □' — http://play.ee/web-development-case-studies/ does not redirect to HTTPS.
- **Solution:** Configure server (Apache/Nginx) to return 301/302 redirect for all HTTP traffic to HTTPS.

  Apache example:

```
RewriteEngine On
RewriteCond %{HTTPS} off
RewriteRule ^(.*)$ https://%{HTTP_HOST}%{REQUEST_URI} [L,R=301]
```

1C. Add HSTS and X-Content-Type-Options headers `Security`

- **Impact:** Protocol downgrade protection, MIME sniffing
- **Problem:** Security Headers grade 20/100; HSTS and X-Content-Type-Options are missing regardless of site signals.
- **Solution:** Add headers to server configuration.

  Apache example:

```
Header always set Strict-Transport-Security "max-age=63072000;
includeSubDomains; preload"
Header always set X-Content-Type-Options "nosniff"
```

1D. Enable HSTS `Security`

- **Impact:** Protocol downgrade protection
- **Problem:** strict-transport-security header is missing; security score is 20/100.
- **Solution:** Add HSTS header with max-age >= 1 year and includeSubDomains.

```
Header always set Strict-Transport-Security "max-age=63072000;
includeSubDomains"
```

1E. Force HTTPS redirect on HTTP traffic `Security`

- **Impact:** Transport security, data integrity
- **Problem:** HTTP requests to http://play.ee/wordpress-support-service/ do not redirect to HTTPS, exposing users to potential MITM attacks.

- **Solution:** Configure the web server (Apache/Nginx) to return a 301 redirect for all HTTP traffic to the HTTPS equivalent:

```
RewriteEngine On
RewriteCond %{HTTPS} off
RewriteRule ^(.*)$ https://%{HTTP_HOST}%{REQUEST_URI} [L,R=301]
```

1F. Enforce HTTPS redirect for all HTTP requests `Security`

- **Impact:** Transport security, data integrity
- **Problem:** Security report indicates http://play.ee/... does not redirect to HTTPS, leaving traffic vulnerable to interception.
- **Solution:** Configure the web server (Apache/Nginx) to return a 301 redirect from HTTP to HTTPS for all requests:

```
RewriteEngine On
RewriteCond %{HTTPS} off
RewriteRule ^(.*)$ https://%{HTTP_HOST}%{REQUEST_URI} [L,R=301]
```

Priority 2: Important

Essential for compliance, user reach, and search visibility.

2A. Add HSTS and X-Content-Type-Options `Security`

- **Impact:** Header security grade, MIME sniffing protection
- **Problem:** Security Headers grade is 20/100; HSTS and X-Content-Type-Options are missing despite being cheap baseline defenses.
- **Solution:** Add the following headers to the server response:

```
Header always set Strict-Transport-Security "max-age=63072000;
includeSubDomains"
Header always set X-Content-Type-Options "nosniff"
```

2B. Fix W3C HTML Validation Errors `Best Practices`

- **Impact:** Parser recovery, rendering consistency
- **Problem:** W3C Validator reports 7 errors including 'Bad start tag in iframe in noscript in head' and parser recovery failure at line 100.
- **Solution:** Move the GTM iframe snippet out of the `<head>` or ensure it is properly closed within `<body>`. Remove invalid `name` attributes on `<meta>` tags inside the head context.

2C. Add Main Landmark and Skip Link `Accessibility`

- **Impact:** Screen reader navigation, WCAG 2.4.1
- **Problem:** HTML Inventory shows 'main: missing' and 'Skip-to-content link: missing'; axe-core flags 'region' and 'landmark-unique' violations.
- **Solution:** Wrap the primary content in `<main>` and add a skip link at the top of the `<body>`:

```
<a href="#main-content" class="skip-link">Skip to content</a>
...
<main id="main-content">
  <!-- content -->
</main>
```

2D. Fix color contrast and add main landmark `Accessibility`

- **Impact:** WCAG 1.4.3, 1.3.1
- **Problem:** axe-core reports 1 serious violation (color-contrast on .heading__main etc.) and missing `main` landmark.
- **Solution:**
  - Increase contrast ratio to ≥4.5:1 for text elements.
  - Wrap primary content in `<main>` tag.
  - Add skip link: `<a href="#main-content" class="skip-link">Skip to content</a>`.

2E. Add main landmark and fix contrast `Accessibility`

- **Impact:** Screen reader navigation, WCAG 1.4.3
- **Problem:** PSI `landmark-one-main` failed; axe reports 1 serious color-contrast violation on h1/span.
- **Solution:**
  - Wrap primary content in `<main>` tag.
  - Increase contrast ratio for `.heading__main` and `p > span` to ≥4.5:1.
  - Add skip-to-content link at top of page.

2F. Fix HTML validation errors `Best Practices`

- **Impact:** Rendering stability, SEO
- **Problem:** W3C Validator reports 7 errors including malformed `<noscript>` / `<iframe>` in `<head>` and parser recovery failure.
- **Solution:** Move `<noscript>` containing GTM iframe outside `<head>` or ensure valid nesting. Remove invalid `name` attribute on `<meta>` tags. Validate full document after fixes.

2G. Fix color contrast and link names `Accessibility`

- **Impact:** WCAG 1.4.3, 2.4.4 compliance
- **Problem:** axe-core reports 2 serious violations: color-contrast on multiple nodes and link-name on logo grid links.
- **Solution:**
  - Increase contrast ratio on `.button--tertiary` and `.heading__small` elements to ≥4.5:1.
  - Add `aria-label` or visible text to logo grid links (e.g., `<a href="" aria-label="Partner Logo">`).

2H. Resolve W3C validation errors and H1 structure `SEO`

- **Impact:** Document outline, SEO indexing
- **Problem:** W3C validator found 7 errors including parser recovery failure; HTML inventory shows 2 `<h1>` elements.
- **Solution:**
  - Ensure only one `<h1>` exists per page.
  - Fix `<noscript><iframe>` nesting in `<head>` (move to `<body>`).
  - Remove invalid `name` attribute on `<meta>` tags.

2I. Add HSTS and X-Content-Type-Options headers `Security`

- **Impact:** Clickjacking, MIME sniffing, downgrade attacks
- **Problem:** HSTS and X-Content-Type-Options are missing; grade is 20/100. These are baseline headers required regardless of site signals.
- **Solution:** Add the following headers to the server configuration:

```
Header always set Strict-Transport-Security "max-age=63072000;
includeSubDomains; preload"
Header always set X-Content-Type-Options "nosniff"
```

2J. Resolve W3C HTML validation errors `Best Practices`

- **Impact:** Parser reliability, SEO crawling
- **Problem:** 7 errors found, including parser recovery failure at line 101 and invalid `<iframe>` inside `<noscript>` within `<head>`.
- **Solution:**
  - Move `<noscript><iframe>...</iframe></noscript>` out of the `<head>` (Google Tag Manager snippet should be in `<body>` or properly placed).

- Ensure `<meta>` tags are valid and do not use forbidden attributes like `name` in contexts requiring `property`.

Priority 3: Best Practice

Recommended for long-term maintainability.

3A. Add Explicit Width and Height to Images `Performance`

- **Impact:** CLS, layout stability
- **Problem:** HTML Inventory shows 52 images without width/height attributes, which risks layout shifts despite current low CLS.
- **Solution:** Ensure all `<img>` tags include `width` and `height` attributes matching the intrinsic dimensions:

```
<img src="image.svg" alt="..." width="360" height="420">
```

3B. Add explicit width/height to images `Performance`

- **Impact:** CLS, Layout stability
- **Problem:** HTML Inventory shows 56 images without width/height attributes, risking layout shifts despite current CLS 0.000.
- **Solution:** Add `width` and `height` attributes to all `<img>` tags matching the intrinsic aspect ratio.

```
<img src="hero.jpg" alt="..." width="1200" height="600">
```

3C. Implement Content Security Policy `Security`

- **Impact:** XSS defense-in-depth
- **Problem:** CSP is missing (only frame-ancestors set). Site signals show no auth/payments, so risk is lower but defense is recommended.
- **Solution:** Deploy a strict CSP with nonce/hash approach when ready.

```
Header always set Content-Security-Policy "default-src 'self';
script-src 'nonce-{random}' 'strict-dynamic';"
```

3D. Implement Content-Security-Policy (CSP) `Security`

- **Impact:** XSS defense-in-depth
- **Problem:** CSP is missing or weak (only `frame-ancestors`); site signals indicate no auth/payments, lowering immediate risk.
- **Solution:** Deploy a strict CSP with nonce/hash for scripts:

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

3E. Remove unused CSS and JavaScript `Performance`

- **Impact:** FCP, TBT, Page weight
- **Problem:** PSI findings indicate 31 KiB unused CSS and 23 KiB unused JS, contributing to FCP warning (1.96 s on mobile).
- **Solution:**
  - Use PurgeCSS or similar tooling to remove unused CSS rules.
  - Code-split JavaScript bundles and defer non-critical scripts.
  - Verify WP Rocket configuration to ensure unused CSS is purged.