Audit Report: Vineeri kvaliteediklassid selgeks - estply

Website: https://estply.com/en/2026/07/06/vineeri-kvaliteediklassid-selgeks/
Date: 29.10.2026
Audit Coverage: 100% — all sources returned data
Confidence: medium

Pages Audited (1 of 1):

- https://estply.com/en/2026/07/06/vineeri-kvaliteediklassid-selgeks/

Summary of results

Overall Score: 79 / 100
Status: **Needs Improvement**

Performance is the strongest area — PSI mobile 92 / desktop 93, TBT 0 ms, FCP 1.2 s — but mobile LCP is 3.3 s against 0.8 s on desktop (a 4× divergence), and the Optimized-Web checklist pins the cause: the hero PNG is served with loading="lazy" and no fetchpriority. Accessibility is the weakest dimension: Lighthouse mobile 91, axe flags page-has-heading-one, and the HTML inventory confirms the article page has literally zero headings (h1–h6 all ×0), while Lighthouse's own `button-name` (weight 10) and `target-size` (weight 7) audits both score 0.00. Security basics returns PASS, but the CSP is frame-ancestors-only with no default-src or object-src 'none', HSTS lacks preload, and Referrer-Policy, Permissions-Policy, COOP and CORP are all absent; W3C adds 8 errors (5 from WP Rocket's `text/rocketlazyloadscript` + defer, 2 from srcset candidates missing width descriptors). Image assets are a light load (2 raster images, both with dimensions and srcset), so the image budget barely moves the score. The document is served with `cache-control: max-age=0`, forcing revalidation of the 66.9 KB HTML on every navigation.

Per-page scores

 Needs Improvement · https://estply.com/en/2026/07/06/vineeri-kvaliteediklassid-selgeks/

Score	Performance	Accessibility	Best Practices	SEO	Security
79	92	91	100	100	PASS

PageSpeed Insights — Mobile vs Desktop

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

Strategy	Performance (M / D)	LCP (M / D)	CLS (M / D)
Mobile vs Desktop	**92** / 93	**3.30 s** / 762 ms	0.000 / **0.144**

Optimization Checklist

4 of 5 passing — 4 pass · 0 warn · 1 fail · 3 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	**Pass**	All non-hero raster images use loading="lazy".
Hero image eagerly loaded	**Fail**	Hero image has loading="lazy", which delays LCP (inferred from DOM order/size — Lighthouse LCP element unavailable). Use loading="eager" (or omit loading) and add fetchpriority="high".
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 2 raster images on the page — responsive-image rule does not apply.
Reasonable number of image sizes	N/A	Too few raster images to evaluate srcset width variety.

Item	Status	Detail
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. Eager-load the hero image with high fetch priority `Performance`

- **Impact:** LCP (mobile 3.3 s), FCP
- **Problem:** The Optimized-Web checklist fails "Hero image eagerly loaded": the hero PNG (…6c041c12….png) carries loading="lazy" and no fetchpriority, and mobile LCP is 3.3 s versus 0.8 s on desktop.
- **Solution:**
  - Remove `loading="lazy"` from the hero (or set `loading="eager"`).
  - Add `fetchpriority="high"` and a `preload` hint.

```
<img src="/wp-content/uploads/2026/08/6c041c12….png"
     alt="…" width="251" height="134"
     loading="eager" fetchpriority="high" decoding="async">
<link rel="preload" as="image" href="/wp-
content/uploads/2026/08/6c041c12….png" fetchpriority="high">
```

1B. Add a single H1 and a real heading hierarchy `Accessibility`

- **Impact:** Screen-reader navigation, document outline, SEO
- **Problem:** HTML inventory reports h1 ×0, h2 ×0, h3 ×0, h4 ×0, h5 ×0, h6 ×0, and axe flags `page-has-heading-one` (moderate) on an article page that should have a clear outline.
- **Solution:**
  - Wrap the article title in exactly one `<h1>` .
  - Promote section titles to `<h2>` and sub-sections to `<h3>` ; do not skip levels.
  - Keep the site logo/wordmark out of the heading hierarchy (use a `<span>` or `aria-label` on the home link).

1C. Give every button an accessible name `Accessibility`

- **Impact:** WCAG 4.1.2 Name, Role, Value; Lighthouse accessibility score

- **Problem:** Lighthouse `button-name` scores 0.00 with weight 10 — "Buttons do not have an accessible name" — so icon-only controls are announced as just "button".
- **Solution:**
  - Add visible text, or an `aria-label` / visually hidden span:

```
<button type="button" aria-label="Close cookie banner">
  <svg aria-hidden="true" focusable="false">…</svg>
</button>
```

  - Mark decorative SVGs `aria-hidden="true"` so they don't pollute the name.

Priority 2: Important

Essential for compliance, user reach, and search visibility.

2A. Enlarge touch targets to at least 44×44 px `Accessibility`

- **Impact:** WCAG 2.5.8 Target Size; mobile usability
- **Problem:** Lighthouse `target-size` scores 0.00 (weight 7) and PSI's modeled SEO audit `tapTargets` also fails, meaning interactive controls are too small or too tightly spaced on mobile.
- **Solution:**
  - Ensure every link/button hit area is ≥44×44 CSS px.
  - Add padding rather than enlarging the icon:

```
.icon-link { display:inline-flex; align-items:center; justify-
content:center; min-width:44px; min-height:44px; }
```

  - Add at least 8 px spacing between adjacent targets.

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

- **Impact:** Repeat-visit TTFB, navigation latency
- **Problem:** The document response is `cache-control: max-age=0` with `expires` equal to the request timestamp, so the 66.9 KB HTML is revalidated on every navigation despite WP Rocket being active.
- **Solution:**
  - In WP Rocket → Cache, enable browser caching and set an HTML TTL (e.g. 10 minutes) with `s-maxage` for the CDN.
  - Target header:

```
Cache-Control: public, max-age=600, s-maxage=3600
```

- Keep `max-age=0` only for logged-in / cart / preview contexts.

2C. Cut render-blocking CSS and unused rules `Performance`

- **Impact:** FCP, LCP, Speed Index
- **Problem:** PSI flags `render-blocking-insight` (est. 560 ms savings) and `unused-css-rules` (est. 33 KiB) across 2 external stylesheets plus 3 inline blocks (9.6 KB).
- **Solution:**
  - Inline the above-the-fold critical CSS and load the rest asynchronously:

```
<link rel="preload" as="style" href="/theme.css"
onload="this.rel='stylesheet'">
```

  - Enable WP Rocket "Remove Unused CSS" and verify the generated used-CSS file covers the article template.
  - Defer non-critical inline `<style>` blocks behind the fold.

2D. Fix srcset candidates missing width descriptors `Best Practices`

- **Impact:** W3C validity, responsive image selection
- **Problem:** W3C reports 2 errors (lines 399 and 419): `srcset` values have no width specified while a `sizes` attribute is present, so browsers cannot pick a candidate correctly.
- **Solution:**
  - Add a width descriptor to every candidate:

```
<img src="…6c041c12….png"
     srcset="…6c041c12-251w.png 251w, …6c041c12-502w.png 502w"
     sizes="(max-width: 600px) 100vw, 251px"
     width="251" height="134" alt="…">
```

  - Or drop `sizes` if only a single candidate is offered.

2E. Harden the WordPress install path and login `Security`

- **Impact:** Brute-force exposure, takeover path
- **Problem:** Security basics warns that `/wp-admin/install.php` is reachable and the login form is served on the default `/wp-login.php` path.
- **Solution:**

```
<Files "install.php">
  Require all denied
```

```
</Files>
```

- Move the login to a custom slug (WP Rocket/WPS Hide Login) or put `/wp-login.php` behind a WAF rate limit + 2FA.
- Keep xmlrpc.php blocked (already passing).

Priority 3: Best Practice

Recommended for long-term maintainability.

3A. 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'`, so no script or resource restrictions exist. Site signals show no auth, payments, user content or ecommerce, so XSS impact is limited and this is P3 rather than P1.
- **Solution:**
  - Add 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'; upgrade-insecure-requests
```

  - Roll out in `Content-Security-Policy-Report-Only` first and watch the reports before enforcing.

3B. Complete the missing low-cost security headers `Security`

- **Impact:** Referrer leakage, feature abuse, origin isolation, info disclosure
- **Problem:** HSTS lacks `preload`; Referrer-Policy, Permissions-Policy, Cross-Origin-Opener-Policy and Cross-Origin-Resource-Policy are all missing, and the `Server: Apache / ZoneOS` header discloses the stack.
- **Solution:**

```
Header always set Strict-Transport-Security "max-age=63072000;
includeSubDomains; preload"
Header always set Referrer-Policy "strict-origin-when-cross-origin"
Header always set Permissions-Policy "geolocation=(), microphone=
(), camera=()"
Header always set Cross-Origin-Opener-Policy "same-origin"
Header always set Cross-Origin-Resource-Policy "same-origin"
Header always set X-Permitted-Cross-Domain-Policies "none"
```

```
ServerSignature Off
ServerTokens Prod
```

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

3C. Add Subresource Integrity to the jsDelivr script `Security`

- **Impact:** Supply-chain tampering risk
- **Problem:** Security basics reports 1 of 1 cross-origin scripts lacks an `integrity` attribute — the iframemanager.js loaded from cdn.jsdelivr.net.
- **Solution:**

```
<script
src="https://cdn.jsdelivr.net/gh/orestbida/iframemanager@1.3.0/dist
/iframemanager.js"
        integrity="sha384-…" crossorigin="anonymous" defer>
</script>
```

  - Generate the hash with `openssl dgst -sha384 -binary file.js | openssl base64 -A`.
  - Better: self-host the 2.7 KB file and drop the third-party origin entirely.

3D. Fix the invalid script type / defer combinations `Best Practices`

- **Impact:** W3C validity (5 of 8 errors)
- **Problem:** W3C reports 5 errors for `type="text/rocketlazyloadscript"` combined with `defer` and 1 for the same type with `async` — a data-block script type must not carry those attributes.
- **Solution:**

  - WP Rocket rewrites `src` to `data-rocket-src`; drop `defer` / `async` from those tags and let the lazyloader handle ordering:

```
<script type="text/rocketlazyloadscript" data-rocket-src="…">
</script>
```

  - Alternatively exclude the affected scripts from delay/lazyload in WP Rocket's file optimization settings.

3E. Add Article structured data and disambiguate repeated link text `SEO`

- **Impact:** Rich results eligibility, anchor-text clarity
- **Problem:** PSI's modeled `structuredData` audit fails and the single JSON-LD block is typed "Unknown"; 7 link texts repeat verbatim ("contact" ×2, "about us" ×2, "products" ×2, "resources" ×2, "code of conduct" ×3).

- **Solution:**
  - Emit a valid `Article` / `BlogPosting` graph with `headline`, `datePublished`, `author`, `image` and `inLanguage`.
  - Validate at search.google.com/test/rich-results.
  - Disambiguate repeated anchors with `aria-label` or visually hidden context, e.g. `<a href="/en/contact/">Contact<span class="sr-only"> our sales team</span></a>`.