https://slayd.in/

American English ⌄

77	70	100	100
Performance	Accessibility	Best Practices	SEO

77

Performance

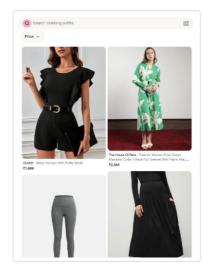Values are estimated and may vary. The [performance score is calculated](#) directly from these metrics. [See calculator.](#)

▲ 0–49   ■ 50–89   ● 90–100

METRICS   Expand view
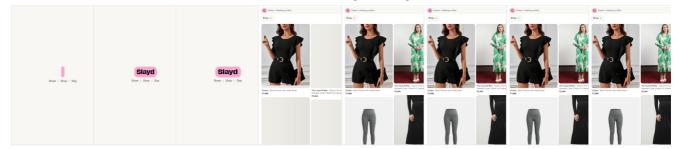
■ First Contentful Paint	▲ Largest Contentful Paint
1.4 s	**3.2 s**
● Total Blocking Time	● Cumulative Layout Shift
0 ms	**0.007**
■ Speed Index	
1.6 s	

📢 Later this year, insights will replace performance audits. Learn more and provide feedback here.

Go back to audits

Show audits relevant to: **All** FCP LCP CLS

INSIGHTS

🔺 **Use efficient cache lifetimes** — Est savings of 5,132 KiB   ⌄

A long cache lifetime can speed up repeat visits to your page. Learn more. FCP LCP

☑ Show 3rd-party resources (25)

Request	Cache TTL	Transfer Size
cloudfront.net		**3,029 KiB**
/b_6OtjTJe (d19dlu1w9mnmln.cloudfront.net)	None	493 KiB
/media/b_h5Wu6Ry (d19dlu1w9mnmln.cloudfront.net)	None	422 KiB
/b_PMXLLCS (d19dlu1w9mnmln.cloudfront.net)	None	300 KiB
/b_42KRWYG (d19dlu1w9mnmln.cloudfront.net)	None	230 KiB
/b_w26xziO (d19dlu1w9mnmln.cloudfront.net)	None	220 KiB
/b_tDxzbVq (d19dlu1w9mnmln.cloudfront.net)	None	201 KiB
/b_BJtU4k6 (d19dlu1w9mnmln.cloudfront.net)	None	188 KiB
/b_ElIH7zm (d19dlu1w9mnmln.cloudfront.net)	None	147 KiB
/media/b_qssHYDs (d19dlu1w9mnmln.cloudfront.net)	None	140 KiB
/b_4Vsskp2 (d19dlu1w9mnmln.cloudfront.net)	None	127 KiB
/b_EZgJfmP (d19dlu1w9mnmln.cloudfront.net)	None	100 KiB
/b_W5d8SCm (d19dlu1w9mnmln.cloudfront.net)	None	98 KiB
/media/b_OnCQuqn (d19dlu1w9mnmln.cloudfront.net)	None	94 KiB
/media/b_kakFTDC (d19dlu1w9mnmln.cloudfront.net)	None	73 KiB
/b_VhRlA6CW (d19dlu1w9mnmln.cloudfront.net)	None	63 KiB
/b_2xHQUqi (d19dlu1w9mnmln.cloudfront.net)	None	60 KiB
/b_5Cvs6JNt (d19dlu1w9mnmln.cloudfront.net)	None	30 KiB
/media/b_CiRdR5Y (d19dlu1w9mnmln.cloudfront.net)	None	29 KiB
/b_LPN09cv (d19dlu1w9mnmln.cloudfront.net)	None	10 KiB
/b_HVpGIEN (d19dlu1w9mnmln.cloudfront.net)	None	5 KiB
slayd.in 1st Party		**2,046 KiB**

Request	Cache TTL	Transfer Size
/assets/index-kSBiKzVV.js (slayd.in)	None	1,107 KiB
/fonts/HelveticaNeue-Roman.otf (slayd.in)	None	616 KiB
/fonts/HelveticaNeue-Medium.otf (slayd.in)	None	255 KiB
/assets/index-D_qX400Y.css (slayd.in)	None	30 KiB
/assets/vendor-BQK9_sJn.js (slayd.in)	None	12 KiB
...brands/Nyka.png (slayd.in)	None	7 KiB
...brands/Littlebox.png (slayd.in)	None	4 KiB
...brands/Ajio.png (slayd.in)	None	4 KiB
...brands/Newme.png (slayd.in)	None	3 KiB
...brands/Myntra.png (slayd.in)	None	3 KiB
...brands/Savana.png (slayd.in)	None	3 KiB
...brands/Zara.png (slayd.in)	None	2 KiB
posthog.com		**46 KiB**
/static/recorder.js?v=1.248.0 (us-assets.i.posthog.com)	5m	38 KiB
/static/web-vitals.js?v=1.248.0 (us-assets.i.posthog.com)	5m	4 KiB
/static/dead-clicks-autocapture.js?v=1.248.0 (us-assets.i.posthog.com)	4h	3 KiB
...phc_ODvWT.../config.js (us-assets.i.posthog.com)	5m	1 KiB
Clarity Utility		**12 KiB**
...0.8.15/clarity.js (www.clarity.ms)	1d	12 KiB

🔺 Modern HTTP

HTTP/2 and HTTP/3 offer many benefits over HTTP/1.1, such as multiplexing. [Learn more about using modern HTTP](). FCP LCP

URL	Protocol
slayd.in 1st Party	
https://slayd.in	http/1.1
/assets/index-kSBiKzVV.js (slayd.in)	http/1.1
/assets/vendor-BQK9_sJn.js (slayd.in)	http/1.1
/assets/index-D_qX400Y.css (slayd.in)	http/1.1
/fonts/HelveticaNeue-Roman.otf (slayd.in)	http/1.1
/icons/logo.svg (slayd.in)	http/1.1
...brands/Savana.png (slayd.in)	http/1.1
...brands/Myntra.png (slayd.in)	http/1.1
...brands/Newme.png (slayd.in)	http/1.1
...brands/Zara.png (slayd.in)	http/1.1
...brands/Ajio.png (slayd.in)	http/1.1
...brands/Nyka.png (slayd.in)	http/1.1

URL	Protocol
…brands/Littlebox.png (slayd.in)	http/1.1
/fonts/HelveticaNeue-Medium.otf (slayd.in)	http/1.1

🔺 Render blocking requests   ⌃

Requests are blocking the page's initial render, which may delay LCP. [Deferring or inlining](#) can move these network requests out of the critical path. `FCP` `LCP`

☑ Show 3rd-party resources (1)

URL	Transfer Size	Est Savings
Google Fonts `Cdn`	**0.6 KiB**	**300 ms**
/css2?family=Reddit+Sans+Condensed:wght@200..900&display=swap (fonts.googleapis.com)	0.6 KiB	300 ms
slayd.in `1st Party`	**30.0 KiB**	**140 ms**
/assets/index-D_qX400Y.css (slayd.in)	30.0 KiB	140 ms

🔺 Improve image delivery   — Est savings of 2,429 KiB   ⌃

Reducing the download time of images can improve the perceived load time of the page and LCP. [Learn more about optimizing image size](#) `FCP` `LCP`
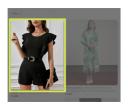
☑ Show 3rd-party resources (16)

URL	Resource Size	Est Savings
cloudfront.net	**2,945.4 KiB**	**2,422.4 KiB**
/b_6OtjTJe (d19dlu1w9mnmln.cloudfront.net)	492.4 KiB	437.4 KiB
This image file is larger than it needs to be (1784x2379) for its displayed dimensions (619x767). Use responsive images to reduce the image download size.		437.4 KiB
/media/b_h5Wu6Ry (d19dlu1w9mnmln.cloudfront.net)	421.2 KiB	374.1 KiB
This image file is larger than it needs to be (1784x2379) for its displayed dimensions (619x767). Use responsive images to reduce the image download size.		374.1 KiB
/b_PMXLLCS (d19dlu1w9mnmln.cloudfront.net)	298.9 KiB	273.2 KiB
This image file is larger than it needs to be (2026x2700) for its displayed dimensions (613x767). Use responsive images to reduce the image download size.		273.2 KiB
/b_w26xziO (d19dlu1w9mnmln.cloudfront.net)	219.7 KiB	202.4 KiB
This image file is larger than it needs to be (2048x3072) for its displayed dimensions (575x863). Use responsive images to reduce the image download size.		202.4 KiB
/b_42KRWYG (d19dlu1w9mnmln.cloudfront.net)	229.2 KiB	165.4 KiB

URL	Resource Size	Est Savings
This image file is larger than it needs to be (1126x1500) for its displayed dimensions (613x767). Use responsive images to reduce the image download size.		165.4 KiB
/b_tDxzbVq (d19dlu1w9mnmln.cloudfront.net)	200.3 KiB	163.4 KiB
This image file is larger than it needs to be (1340x1785) for its displayed dimensions (576x767). Use responsive images to reduce the image download size.		163.4 KiB
/b_BJtU4k6 (d19dlu1w9mnmln.cloudfront.net)	187.3 KiB	153.3 KiB
This image file is larger than it needs to be (1350x1800) for its displayed dimensions (575x767). Use responsive images to reduce the image download size.		153.3 KiB
/b_ElIH7zm (d19dlu1w9mnmln.cloudfront.net)	146.3 KiB	112.7 KiB
This image file is larger than it needs to be (1200x1800) for its displayed dimensions (575x863). Use responsive images to reduce the image download size.		112.7 KiB
/b_4Vsskp2 (d19dlu1w9mnmln.cloudfront.net)	126.4 KiB	107.8 KiB
This image file is larger than it needs to be (1500x2000) for its displayed dimensions (575x767). Use responsive images to reduce the image download size.		107.8 KiB
/media/b_qssHYDs (d19dlu1w9mnmln.cloudfront.net)	139.2 KiB	99.7 KiB
This image file is larger than it needs to be (1080x1440) for its displayed dimensions (575x767). Use responsive images to reduce the image download size.		99.7 KiB
/b_EZgJfmP (d19dlu1w9mnmln.cloudfront.net)	99.1 KiB	76.4 KiB
This image file is larger than it needs to be (1200x1800) for its displayed dimensions (575x863). Use responsive images to reduce the image download size.		76.4 KiB
/media/b_OnCQuqn (d19dlu1w9mnmln.cloudfront.net)	93.8 KiB	67.2 KiB
This image file is larger than it needs to be (1080x1440) for its displayed dimensions (575x767). Use responsive images to reduce the image download size.		67.2 KiB
/b_W5d8SCm (d19dlu1w9mnmln.cloudfront.net)	97.1 KiB	63.7 KiB
This image file is larger than it needs to be (1013x1350) for its displayed dimensions (613x767). Use responsive images to reduce the image download size.		63.7 KiB
/b_VhRlA6CW (d19dlu1w9mnmln.cloudfront.net)	62.7 KiB	53.5 KiB
This image file is larger than it needs to be (1500x2250) for its displayed dimensions (575x863). Use responsive images to reduce the image download size.		53.5 KiB
/b_2xHQUqi (d19dlu1w9mnmln.cloudfront.net)	59.1 KiB	42.4 KiB
This image file is larger than it needs to be (1080x1440) for its displayed dimensions (575x767). Use responsive images to reduce the image download size.		42.4 KiB
/media/b_kakFTDC (d19dlu1w9mnmln.cloudfront.net)	72.5 KiB	29.9 KiB

URL	Resource Size	Est Savings
Using a modern image format (WebP, AVIF) or increasing the image compression could improve this image's download size.		29.9 KiB
slayd.in [1st Party]	**7.1 KiB**	**6.8 KiB**
...brands/Nyka.png (slayd.in)	7.1 KiB	6.8 KiB
Using a modern image format (WebP, AVIF) or increasing the image compression could improve this image's download size.		6.6 KiB

▲ LCP request discovery   ⌃

Optimize LCP by making the LCP image discoverable from the HTML immediately, and avoiding lazy-loading LCP

🚫 fetchpriority=high should be applied

🚫 Request is discoverable in initial document

🚫 lazy load not applied

Black Romper With Ruffle Detail

```html
<img alt="Black Romper With Ruffle Detail" class="_imageStyle_1gm8f_23"
loading="lazy" src="https://d19dlu1w9mnmln.cloudfront.net/b_BJtU4k6">
```

▲ Network dependency tree   ⌃

Avoid chaining critical requests by reducing the length of chains, reducing the download size of resources, or deferring the download of unnecessary resources to improve page load. LCP

Maximum critical path latency: **1,458 ms**

Initial Navigation

- https://slayd.in **- 24 ms,** 1.35 KiB
  - /css2?family=Reddit+Sans+Condensed:wght@200..900&display=swap (fonts.googleapis.com) **- 133 ms,** 0.58 KiB
  - /assets/index-kSBiKzVV.js (slayd.in) **- 263 ms,** 1,107.25 KiB
    - /api/homepage/ (api.slayd.in) **- 690 ms,** 9.96 KiB
    - /decide/?v=4&ip=1&_=175...&ver=1.248.0&compression=base64 (us.i.posthog.com) **- 527 ms,** 0.94 KiB
    - /e/?ip=1&_=175...&ver=1.248.0&compression=gzip-js (us.i.posthog.com) **- 549 ms,** 0.28 KiB
  - /assets/index-D_qX400Y.css (slayd.in) **- 75 ms,** 30.02 KiB
  - /fonts/HelveticaNeue-Roman.otf (slayd.in) **- 430 ms,** 615.89 KiB
  - /fonts/HelveticaNeue-Medium.otf (slayd.in) **- 1,458 ms,** 254.64 KiB

■ Legacy JavaScript — Est savings of 12 KiB   ⌃

Polyfills and transforms enable legacy browsers to use new JavaScript features. However, many aren't necessary for modern browsers. Consider modifying your JavaScript build process to not transpile

[Baseline](#) features, unless you know you must support legacy browsers. [Learn why most sites can deploy ES6+ code without transpiling](#) `FCP` `LCP`

URL	Wasted bytes
posthog.com	**12.0 KiB**
/static/recorder.js?v=1.248.0 (us-assets.i.posthog.com)	12.0 KiB
[../../../node_modules/.pnpm/@rrweb+record@2.0.0-alpha.17_patch_hash=46frarym7rupyfspsu67b3afhe/node_modules/@rrweb/record/dist/record.js:4680:10](#) `Array.from`	

○ Layout shift culprits   ⌃

Layout shifts occur when elements move absent any user interaction. [Investigate the causes of layout shifts](#), such as elements being added, removed, or their fonts changing as the page loads. `CLS`

Element	Layout shift score
Total	0.007
Black High Rise Ruffle Solid Pocket Maxi Skirt `<img alt="Black High Rise Ruffle Solid Pocket Maxi Skirt" class="_imageStyle_1gm8f_23" loading="lazy" src="https://d19dlu1w9mnmln.cloudfront.net/b_4Vsskp2">`	0.007

○ LCP by phase   ⌃

Each [phase has specific improvement strategies](#). Ideally, most of the LCP time should be spent on loading the resources, not within delays. `LCP`

Phase	Duration
Time to first byte	20 ms
Resource load delay	1,330 ms
Resource load duration	190 ms
Element render delay	80 ms

 Black Romper With Ruffle Detail
`<img alt="Black Romper With Ruffle Detail" class="_imageStyle_1gm8f_23" loading="lazy" src="https://d19dlu1w9mnmln.cloudfront.net/b_BJtU4k6">`

○ 3rd parties   ⌃

3rd party code can significantly impact load performance. [Reduce and defer loading of 3rd party code](#) to prioritize your page's content.

3rd party	Transfer size	Main thread time
Clarity `Utility`	**32 KiB**	**44 ms**
…0.8.15/clarity.js (www.clarity.ms)	30 KiB	44 ms
/tag/qrmpd2xepk (www.clarity.ms)	1 KiB	1 ms
/collect (b.clarity.ms)	1 KiB	0 ms
/c.gif?ctsa=… (c.clarity.ms)	0 KiB	0 ms
posthog.com	**50 KiB**	**37 ms**
/static/recorder.js?v=1.248.0 (us-assets.i.posthog.com)	40 KiB	35 ms
/static/dead-clicks-autocapture.js?v=1.248.0 (us-assets.i.posthog.com)	5 KiB	1 ms
/static/web-vitals.js?v=1.248.0 (us-assets.i.posthog.com)	4 KiB	0 ms
…phc_ODvWT…/config.js (us-assets.i.posthog.com)	1 KiB	0 ms
/decide/?v=4&ip=1&_=175…&ver=1.248.0&compression=base64 (us.i.posthog.com)	1 KiB	0 ms
/e/?ip=1&_=175…&ver=1.248.0&compression=gzip-js (us.i.posthog.com)	0 KiB	0 ms
Google Fonts `Cdn`	**1 KiB**	**0 ms**
/css2?family=Reddit+Sans+Condensed:wght@200..900&display=swap (fonts.googleapis.com)	1 KiB	0 ms
cloudfront.net	**3,029 KiB**	**0 ms**
/b_6OtjTJe (d19dlu1w9mnmln.cloudfront.net)	493 KiB	0 ms
/media/b_h5Wu6Ry (d19dlu1w9mnmln.cloudfront.net)	422 KiB	0 ms
/b_PMXLLCS (d19dlu1w9mnmln.cloudfront.net)	300 KiB	0 ms
/b_42KRWYG (d19dlu1w9mnmln.cloudfront.net)	230 KiB	0 ms
/b_w26xziO (d19dlu1w9mnmln.cloudfront.net)	220 KiB	0 ms
/b_tDxzbVq (d19dlu1w9mnmln.cloudfront.net)	201 KiB	0 ms
/b_BJtU4k6 (d19dlu1w9mnmln.cloudfront.net)	188 KiB	0 ms
/b_ELIH7zm (d19dlu1w9mnmln.cloudfront.net)	147 KiB	0 ms
/media/b_qssHYDs (d19dlu1w9mnmln.cloudfront.net)	140 KiB	0 ms
/b_4Vsskp2 (d19dlu1w9mnmln.cloudfront.net)	127 KiB	0 ms
/b_EZgJfmP (d19dlu1w9mnmln.cloudfront.net)	100 KiB	0 ms
/b_W5d8SCm (d19dlu1w9mnmln.cloudfront.net)	98 KiB	0 ms
/media/b_OnCQuqn (d19dlu1w9mnmln.cloudfront.net)	94 KiB	0 ms
/media/b_kakFTDC (d19dlu1w9mnmln.cloudfront.net)	73 KiB	0 ms
/b_VhRIA6CW (d19dlu1w9mnmln.cloudfront.net)	63 KiB	0 ms
/b_2xHQUqi (d19dlu1w9mnmln.cloudfront.net)	60 KiB	0 ms
/b_5Cvs6JNt (d19dlu1w9mnmln.cloudfront.net)	30 KiB	0 ms
/media/b_CiRdR5Y (d19dlu1w9mnmln.cloudfront.net)	29 KiB	0 ms
/b_LPN09cv (d19dlu1w9mnmln.cloudfront.net)	10 KiB	0 ms
/b_HVpGIEN (d19dlu1w9mnmln.cloudfront.net)	5 KiB	0 ms

These insights are also available in the Chrome DevTools Performance Panel – record a trace to view more detailed information.

DIAGNOSTICS

🔺 Reduce unused JavaScript — Est savings of 585 KiB   ⌃

Reduce unused JavaScript and defer loading scripts until they are required to decrease bytes consumed by network activity. Learn how to reduce unused JavaScript. FCP LCP

URL	Transfer Size	Est Savings
slayd.in 1st Party	1,106.9 KiB	585.3 KiB
/assets/index-kSBiKzVV.js (slayd.in)	1,106.9 KiB	585.3 KiB

🔺 Use HTTP/2 — 14 requests not served via HTTP/2   ⌃

HTTP/2 offers many benefits over HTTP/1.1, including binary headers and multiplexing. Learn more about HTTP/2. LCP FCP

URL	Protocol
slayd.in 1st Party	
https://slayd.in	http/1.1
/assets/index-kSBiKzVV.js (slayd.in)	http/1.1
/assets/vendor-BQK9_sJn.js (slayd.in)	http/1.1
/assets/index-D_qX400Y.css (slayd.in)	http/1.1
/fonts/HelveticaNeue-Roman.otf (slayd.in)	http/1.1
/icons/logo.svg (slayd.in)	http/1.1
...brands/Savana.png (slayd.in)	http/1.1
...brands/Myntra.png (slayd.in)	http/1.1
...brands/Newme.png (slayd.in)	http/1.1
...brands/Zara.png (slayd.in)	http/1.1
...brands/Ajio.png (slayd.in)	http/1.1
...brands/Nyka.png (slayd.in)	http/1.1
...brands/Littlebox.png (slayd.in)	http/1.1
/fonts/HelveticaNeue-Medium.otf (slayd.in)	http/1.1

▲ **Reduce unused CSS** — Est savings of 26 KiB ⌃

Reduce unused rules from stylesheets and defer CSS not used for above-the-fold content to decrease bytes consumed by network activity. [Learn how to reduce unused CSS](#). FCP LCP

URL	Transfer Size	Est Savings
slayd.in 1st Party	**29.8 KiB**	**26.3 KiB**
/assets/index-D_qX400Y.css (slayd.in)	29.8 KiB	26.3 KiB

▲ **Page prevented back/forward cache restoration** — 1 failure reason ⌃

Many navigations are performed by going back to a previous page, or forwards again. The back/forward cache (bfcache) can speed up these return navigations. [Learn more about the bfcache](#)

Failure reason	Failure type
The page was opened using '`window.open()`' and another tab has a reference to it, or the page opened a window.	Not actionable
https://slayd.in	

■ **Avoid serving legacy JavaScript to modern browsers** — Est savings of 38 KiB ⌃

Polyfills and transforms enable legacy browsers to use new JavaScript features. However, many aren't necessary for modern browsers. Consider modifying your JavaScript build process to not transpile [Baseline](#) features, unless you know you must support legacy browsers. [Learn why most sites can deploy ES6+ code without transpiling](#) FCP LCP

☐ Show 3rd-party resources (1)

URL	Est Savings
slayd.in 1st Party	**25.6 KiB**
/assets/index-kSBiKzVV.js (slayd.in)	25.6 KiB
/assets/index-kSBiKzVV.js:10:187012 (slayd.in) Math.trunc	

■ **Avoid enormous network payloads** — Total size was 5,171 KiB ⌃

Large network payloads cost users real money and are highly correlated with long load times. [Learn how to reduce payload sizes](#).

☑ Show 3rd-party resources (7)

URL	Transfer Size
cloudfront.net	**2,054.2 KiB**

URL	Transfer Size
/b_6OtjTJe (d19dlu1w9mnmln.cloudfront.net)	493.3 KiB
/media/b_h5Wu6Ry (d19dlu1w9mnmln.cloudfront.net)	422.2 KiB
/b_PMXLLCS (d19dlu1w9mnmln.cloudfront.net)	299.6 KiB
/b_42KRWYG (d19dlu1w9mnmln.cloudfront.net)	229.9 KiB
/b_w26xziO (d19dlu1w9mnmln.cloudfront.net)	220.4 KiB
/b_tDxzbVq (d19dlu1w9mnmln.cloudfront.net)	201.0 KiB
/b_BJtU4k6 (d19dlu1w9mnmln.cloudfront.net)	187.8 KiB
slayd.in `1st Party`	**1,977.8 KiB**
/assets/index-kSBiKzVV.js (slayd.in)	1,107.3 KiB
/fonts/HelveticaNeue-Roman.otf (slayd.in)	615.9 KiB
/fonts/HelveticaNeue-Medium.otf (slayd.in)	254.6 KiB

More information about the performance of your application. These numbers don't [directly affect](#) the Performance score.

PASSED AUDITS (18) Hide

🟢 Document request latency ⌃

Your first network request is the most important. Reduce its latency by avoiding redirects, ensuring a fast server response, and enabling text compression. `FCP` `LCP`

✅ Avoids redirects

✅ Server responds quickly

✅ Applies text compression

🟢 Optimize DOM size ⌃

A large DOM can increase the duration of style calculations and layout reflows, impacting page responsiveness. A large DOM will also increase memory usage. [Learn how to avoid an excessive DOM size](#).

Statistic	Element	Value
Total elements		398
Most children	Outzidr - Black Romper With Ruffle Detail ₹1,699 Savana - Gym Legging ₹517 ... `<div class="_column_l36qx_13">`	10

Statistic	Element	Value
DOM depth	₹4,499 `<span>`	18

○ Duplicated JavaScript   ⌃

Remove large, duplicate JavaScript modules from bundles to reduce unnecessary bytes consumed by network activity.

○ Font display   ⌃

Consider setting `font-display` to `swap` or `optional` to ensure text is consistently visible. `swap` can be further optimized to mitigate layout shifts with font metric overrides.

● Forced reflow   ⌃

Many APIs, typically reading layout geometry, force the rendering engine to pause script execution in order to calculate the style and layout. Learn more about forced reflow and its mitigations.

○ INP by phase   ⌃

Start investigating with the longest phase. Delays can be minimized. To reduce processing duration, optimize the main-thread costs, often JS.

● Optimize viewport for mobile   ⌃

Tap interactions may be delayed by up to 300 ms if the viewport is not optimized for mobile.

```
head > meta
<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1, user-
scalable=no, vi…">
```

● Defer offscreen images   ⌃

Consider lazy-loading offscreen and hidden images after all critical resources have finished loading to lower time to interactive. Learn how to defer offscreen images. FCP LCP

● Minify CSS   ⌃

Minifying CSS files can reduce network payload sizes. Learn how to minify CSS. FCP LCP

● Minify JavaScript   ⌃

Minifying JavaScript files can reduce payload sizes and script parse time. Learn how to minify JavaScript. FCP LCP

● Preconnect to required origins ⌃

Warnings:

- A `<link rel=preconnect>` was found for "https://fonts.gstatic.com" but was not used by the browser. Only use `preconnect` for important origins that the page will certainly request.
- More than 2 `<link rel=preconnect>` connections were found. These should be used sparingly and only to the most important origins.

Consider adding `preconnect` or `dns-prefetch` resource hints to establish early connections to important third-party origins. [Learn how to preconnect to required origins](). `LCP` `FCP`

○ User Timing marks and measures ⌃

Consider instrumenting your app with the User Timing API to measure your app's real-world performance during key user experiences. [Learn more about User Timing marks]().

● JavaScript execution time — 0.2 s ⌃

Consider reducing the time spent parsing, compiling, and executing JS. You may find delivering smaller JS payloads helps with this. [Learn how to reduce Javascript execution time](). `TBT`

URL	Total CPU Time	Script Evaluation	Script Parse
slayd.in `1st Party`	**316 ms**	**222 ms**	**2 ms**
/assets/index-kSBiKzVV.js (slayd.in)	253 ms	220 ms	1 ms
https://slayd.in	64 ms	2 ms	0 ms
Unattributable	**70 ms**	**3 ms**	**0 ms**
Unattributable	70 ms	3 ms	0 ms

● Minimizes main-thread work — 0.4 s ⌃

Consider reducing the time spent parsing, compiling and executing JS. You may find delivering smaller JS payloads helps with this. [Learn how to minimize main-thread work]() `TBT`

Category	Time Spent
Script Evaluation	278 ms
Other	118 ms
Rendering	23 ms
Style & Layout	19 ms

Category	Time Spent
Script Parsing & Compilation	5 ms
Garbage Collection	1 ms
Parse HTML & CSS	1 ms

○ Lazy load third-party resources with facades   ⌃

Some third-party embeds can be lazy loaded. Consider replacing them with a facade until they are required. [Learn how to defer third-parties with a facade](). `TBT`

● Uses passive listeners to improve scrolling performance   ⌃

Consider marking your touch and wheel event listeners as `passive` to improve your page's scroll performance. [Learn more about adopting passive event listeners]().

● Avoids `document.write()`   ⌃

For users on slow connections, external scripts dynamically injected via `document.write()` can delay page load by tens of seconds. [Learn how to avoid document.write()]().

○ Avoid long main-thread tasks   ⌃

Lists the longest tasks on the main thread, useful for identifying worst contributors to input delay. [Learn how to avoid long main-thread tasks]() `TBT`
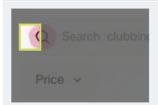
70

Accessibility

These checks highlight opportunities to [improve the accessibility of your web app](). Automatic detection can only detect a subset of issues and does not guarantee the accessibility of your web app, so [manual testing]() is also encouraged.

NAMES AND LABELS

🔺 **Buttons do not have an accessible name**   ⌃

When a button doesn't have an accessible name, screen readers announce it as "button", making it unusable for users who rely on screen readers. [Learn how to make buttons more accessible](#).

Failing Elements
div._container_12v3g_3 > div._row_12v3g_10 > div._backButtonWrapper_12v3g_15 > button._backButton_12v3g_15 `<button class="_backButton_12v3g_15" style="opacity: 0;">`

These are opportunities to improve the semantics of the controls in your application. This may enhance the experience for users of assistive technology, like a screen reader.

BEST PRACTICES

🔺 `[user-scalable="no"]` is used in the `<meta name="viewport">` element or the `[maximum-scale]` attribute is less than 5.   ⌃

Disabling zooming is problematic for users with low vision who rely on screen magnification to properly see the contents of a web page. [Learn more about the viewport meta tag](#).

Failing Elements
head > meta `<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1, user-scalable=no, vi…">`

These items highlight common accessibility best practices.

CONTRAST

🔺 Background and foreground colors do not have a sufficient contrast ratio.   ⌃

Low-contrast text is difficult or impossible for many users to read. [Learn how to provide sufficient color contrast](#).

Failing Elements
div._container_12v3g_3 > div._row_12v3g_10 > div._searchWrapper_12v3g_30 > input._searchInput_12v3g_40 `<input class="_searchInput_12v3g_40" placeholder="Search 'clubbing outfits'" type="text" value="" style="padding-left: 43px;">`

These are opportunities to improve the legibility of your content.

ADDITIONAL ITEMS TO MANUALLY CHECK (10)   Hide

○   Interactive controls are keyboard focusable   ∧

Custom interactive controls are keyboard focusable and display a focus indicator. Learn how to make custom controls focusable.

○   Interactive elements indicate their purpose and state   ∧

Interactive elements, such as links and buttons, should indicate their state and be distinguishable from non-interactive elements. Learn how to decorate interactive elements with affordance hints.

○   The page has a logical tab order   ∧

Tabbing through the page follows the visual layout. Users cannot focus elements that are offscreen. Learn more about logical tab ordering.

○   Visual order on the page follows DOM order   ∧

DOM order matches the visual order, improving navigation for assistive technology. Learn more about DOM and visual ordering.

○   User focus is not accidentally trapped in a region   ∧

A user can tab into and out of any control or region without accidentally trapping their focus. Learn how to avoid focus traps.

○   The user's focus is directed to new content added to the page   ∧

If new content, such as a dialog, is added to the page, the user's focus is directed to it. Learn how to direct focus to new content.

○   HTML5 landmark elements are used to improve navigation   ∧

Landmark elements (`<main>`, `<nav>`, etc.) are used to improve the keyboard navigation of the page for assistive technology. Learn more about landmark elements.

○   Offscreen content is hidden from assistive technology   ∧

Offscreen content is hidden with display: none or aria-hidden=true. Learn how to properly hide offscreen content.

○   Custom controls have associated labels   ∧

Custom interactive controls have associated labels, provided by aria-label or aria-labelledby. Learn more about custom controls and labels.

○   Custom controls have ARIA roles   ∧

Custom interactive controls have appropriate ARIA roles. [Learn how to add roles to custom controls](#).

These items address areas which an automated testing tool cannot cover. Learn more in our guide on [conducting an accessibility review](#).

PASSED AUDITS (9)   Hide

🟢 `[aria-hidden="true"]` is not present on the document `<body>`   ⌃

Assistive technologies, like screen readers, work inconsistently when `aria-hidden="true"` is set on the document <body>. [Learn how `aria-hidden` affects the document body](#).

🟢 Image elements have `[alt]` attributes   ⌃

Informative elements should aim for short, descriptive alternate text. Decorative elements can be ignored with an empty alt attribute. [Learn more about the `alt` attribute](#).

🟢 Document has a `<title>` element   ⌃

The title gives screen reader users an overview of the page, and search engine users rely on it heavily to determine if a page is relevant to their search. [Learn more about document titles](#).

🟢 `<html>` element has a `[lang]` attribute   ⌃

If a page doesn't specify a `lang` attribute, a screen reader assumes that the page is in the default language that the user chose when setting up the screen reader. If the page isn't actually in the default language, then the screen reader might not announce the page's text correctly. [Learn more about the lang attribute](#).

🟢 `<html>` element has a valid value for its `[lang]` attribute   ⌃

Specifying a valid [BCP 47 language](#) helps screen readers announce text properly. [Learn how to use the lang attribute](#).

🟢 Form elements have associated labels   ⌃

Labels ensure that form controls are announced properly by assistive technologies, like screen readers. [Learn more about form element labels](#).

🟢 Links have a discernible name   ⌃

Link text (and alternate text for images, when used as links) that is discernible, unique, and focusable improves the navigation experience for screen reader users. Learn how to make links accessible.

● Touch targets have sufficient size and spacing.   ⌃

Touch targets with sufficient size and spacing help users who may have difficulty targeting small controls to activate the targets. Learn more about touch targets.

● Image elements do not have `[alt]` attributes that are redundant text.   ⌃

Informative elements should aim for short, descriptive alternative text. Alternative text that is exactly the same as the text adjacent to the link or image is potentially confusing for screen reader users, because the text will be read twice. Learn more about the `alt` attribute.

NOT APPLICABLE (45)   Hide

○ `[accesskey]` values are unique   ⌃

Access keys let users quickly focus a part of the page. For proper navigation, each access key must be unique. Learn more about access keys.

○ `[aria-*]` attributes match their roles   ⌃

Each ARIA `role` supports a specific subset of `aria-*` attributes. Mismatching these invalidates the `aria-*` attributes. Learn how to match ARIA attributes to their roles.

○ Uses ARIA roles only on compatible elements   ⌃

Many HTML elements can only be assigned certain ARIA roles. Using ARIA roles where they are not allowed can interfere with the accessibility of the web page. Learn more about ARIA roles.

○ `button`, `link`, and `menuitem` elements have accessible names   ⌃

When an element doesn't have an accessible name, screen readers announce it with a generic name, making it unusable for users who rely on screen readers. Learn how to make command elements more accessible.

○ ARIA attributes are used as specified for the element's role   ⌃

Some ARIA attributes are only allowed on an element under certain conditions. Learn more about conditional ARIA attributes.

○ Deprecated ARIA roles were not used ∧

Deprecated ARIA roles may not be processed correctly by assistive technology. [Learn more about deprecated ARIA roles](#).

○ Elements with `role="dialog"` or `role="alertdialog"` have accessible names. ∧

ARIA dialog elements without accessible names may prevent screen readers users from discerning the purpose of these elements. [Learn how to make ARIA dialog elements more accessible](#).

○ `[aria-hidden="true"]` elements do not contain focusable descendents ∧

Focusable descendents within an `[aria-hidden="true"]` element prevent those interactive elements from being available to users of assistive technologies like screen readers. [Learn how `aria-hidden` affects focusable elements](#).

○ ARIA input fields have accessible names ∧

When an input field doesn't have an accessible name, screen readers announce it with a generic name, making it unusable for users who rely on screen readers. [Learn more about input field labels](#).

○ ARIA `meter` elements have accessible names ∧

When a meter element doesn't have an accessible name, screen readers announce it with a generic name, making it unusable for users who rely on screen readers. [Learn how to name `meter` elements](#).

○ ARIA `progressbar` elements have accessible names ∧

When a `progressbar` element doesn't have an accessible name, screen readers announce it with a generic name, making it unusable for users who rely on screen readers. [Learn how to label `progressbar` elements](#).

○ Elements use only permitted ARIA attributes ∧

Using ARIA attributes in roles where they are prohibited can mean that important information is not communicated to users of assistive technologies. [Learn more about prohibited ARIA roles](#).

○ `[role]`s have all required `[aria-*]` attributes ∧

Some ARIA roles have required attributes that describe the state of the element to screen readers. [Learn more about roles and required attributes](#).

○ Elements with an ARIA `[role]` that require children to contain a specific `[role]` have all required children. ∧

Some ARIA parent roles must contain specific child roles to perform their intended accessibility functions. [Learn more about roles and required children elements](#).

○ `[role]`s are contained by their required parent element ⌃

Some ARIA child roles must be contained by specific parent roles to properly perform their intended accessibility functions. [Learn more about ARIA roles and required parent element](#).

○ `[role]` values are valid ⌃

ARIA roles must have valid values in order to perform their intended accessibility functions. [Learn more about valid ARIA roles](#).

○ Elements with the `role=text` attribute do not have focusable descendents. ⌃

Adding `role=text` around a text node split by markup enables VoiceOver to treat it as one phrase, but the element's focusable descendents will not be announced. [Learn more about the `role=text` attribute](#).

○ ARIA toggle fields have accessible names ⌃

When a toggle field doesn't have an accessible name, screen readers announce it with a generic name, making it unusable for users who rely on screen readers. [Learn more about toggle fields](#).

○ ARIA `tooltip` elements have accessible names ⌃

When a tooltip element doesn't have an accessible name, screen readers announce it with a generic name, making it unusable for users who rely on screen readers. [Learn how to name `tooltip` elements](#).

○ ARIA `treeitem` elements have accessible names ⌃

When a `treeitem` element doesn't have an accessible name, screen readers announce it with a generic name, making it unusable for users who rely on screen readers. [Learn more about labeling `treeitem` elements](#).

○ `[aria-*]` attributes have valid values ⌃

Assistive technologies, like screen readers, can't interpret ARIA attributes with invalid values. [Learn more about valid values for ARIA attributes](#).

○ `[aria-*]` attributes are valid and not misspelled ⌃

Assistive technologies, like screen readers, can't interpret ARIA attributes with invalid names. [Learn more about valid ARIA attributes](#).

○ The page contains a heading, skip link, or landmark region ⌃

Adding ways to bypass repetitive content lets keyboard users navigate the page more efficiently. [Learn more about bypass blocks](#).

○ `<dl>`'s contain only properly-ordered `<dt>` and `<dd>` groups, `<script>`, `<template>` or `<div>` elements.   ⌃

When definition lists are not properly marked up, screen readers may produce confusing or inaccurate output. Learn how to structure definition lists correctly.

○ Definition list items are wrapped in `<dl>` elements   ⌃

Definition list items (`<dt>` and `<dd>`) must be wrapped in a parent `<dl>` element to ensure that screen readers can properly announce them. Learn how to structure definition lists correctly.

○ ARIA IDs are unique   ⌃

The value of an ARIA ID must be unique to prevent other instances from being overlooked by assistive technologies. Learn how to fix duplicate ARIA IDs.

○ No form fields have multiple labels   ⌃

Form fields with multiple labels can be confusingly announced by assistive technologies like screen readers which use either the first, the last, or all of the labels. Learn how to use form labels.

○ `<frame>` or `<iframe>` elements have a title   ⌃

Screen reader users rely on frame titles to describe the contents of frames. Learn more about frame titles.

○ Heading elements appear in a sequentially-descending order   ⌃

Properly ordered headings that do not skip levels convey the semantic structure of the page, making it easier to navigate and understand when using assistive technologies. Learn more about heading order.

○ `<html>` element has an `[xml:lang]` attribute with the same base language as the `[lang]` attribute.   ⌃

If the webpage does not specify a consistent language, then the screen reader might not announce the page's text correctly. Learn more about the `lang` attribute.

○ Input buttons have discernible text.   ⌃

Adding discernable and accessible text to input buttons may help screen reader users understand the purpose of the input button. Learn more about input buttons.

○ `<input type="image">` elements have `[alt]` text   ⌃

When an image is being used as an `<input>` button, providing alternative text can help screen reader users understand the purpose of the button. Learn about input image alt text.

○ Links are distinguishable without relying on color.   ⌃

Low-contrast text is difficult or impossible for many users to read. Link text that is discernible improves the experience for users with low vision. [Learn how to make links distinguishable](#).

○ Lists contain only `<li>` elements and script supporting elements (`<script>` and `<template>`).   ⌄

Screen readers have a specific way of announcing lists. Ensuring proper list structure aids screen reader output. [Learn more about proper list structure](#).

○ List items (`<li>`) are contained within `<ul>`, `<ol>` or `<menu>` parent elements   ⌄

Screen readers require list items (`<li>`) to be contained within a parent `<ul>`, `<ol>` or `<menu>` to be announced properly. [Learn more about proper list structure](#).

○ The document does not use `<meta http-equiv="refresh">`   ⌄

Users do not expect a page to refresh automatically, and doing so will move focus back to the top of the page. This may create a frustrating or confusing experience. [Learn more about the refresh meta tag](#).

○ `<object>` elements have alternate text   ⌄

Screen readers cannot translate non-text content. Adding alternate text to `<object>` elements helps screen readers convey meaning to users. [Learn more about alt text for `object` elements](#).

○ Select elements have associated label elements.   ⌄

Form elements without effective labels can create frustrating experiences for screen reader users. [Learn more about the `select` element](#).

○ Skip links are focusable.   ⌄

Including a skip link can help users skip to the main content to save time. [Learn more about skip links](#).

○ No element has a `[tabindex]` value greater than 0   ⌄

A value greater than 0 implies an explicit navigation ordering. Although technically valid, this often creates frustrating experiences for users who rely on assistive technologies. [Learn more about the `tabindex` attribute](#).

○ Tables have different content in the summary attribute and `<caption>`.   ⌄

The summary attribute should describe the table structure, while `<caption>` should have the onscreen title. Accurate table mark-up helps users of screen readers. [Learn more about summary and caption](#).

○ Cells in a `<table>` element that use the `[headers]` attribute refer to table cells within the same table.   ⌄

Screen readers have features to make navigating tables easier. Ensuring `<td>` cells using the `[headers]` attribute only refer to other cells in the same table may improve the experience for screen reader users. Learn more about the `headers` attribute.

○ `<th>` elements and elements with `[role="columnheader"/"rowheader"]` have data cells they describe.   ⌃

Screen readers have features to make navigating tables easier. Ensuring table headers always refer to some set of cells may improve the experience for screen reader users. Learn more about table headers.

○ `[lang]` attributes have a valid value   ⌃

Specifying a valid BCP 47 language on elements helps ensure that text is pronounced correctly by a screen reader. Learn how to use the `lang` attribute.

○ `<video>` elements contain a `<track>` element with `[kind="captions"]`   ⌃

When a video provides a caption it is easier for deaf and hearing impaired users to access its information. Learn more about video captions.

100

Best Practices

TRUST AND SAFETY

○ Ensure CSP is effective against XSS attacks   ⌃

A strong Content Security Policy (CSP) significantly reduces the risk of cross-site scripting (XSS) attacks. Learn how to use a CSP to prevent XSS

Description	Directive	Severity
No CSP found in enforcement mode		High

○ Use a strong HSTS policy   ⌃

Deployment of the HSTS header significantly reduces the risk of downgrading HTTP connections and eavesdropping attacks. A rollout in stages, starting with a low max-age is recommended. [Learn more about using a strong HSTS policy.](#)

Description	Directive	☑ Severity
No HSTS header found		High

○ Ensure proper origin isolation with COOP   ⌃

The Cross-Origin-Opener-Policy (COOP) can be used to isolate the top-level window from other documents such as pop-ups. [Learn more about deploying the COOP header.](#)

Description	Directive	Severity
No COOP header found		High

○ Mitigate clickjacking with XFO or CSP   ⌃

The `X-Frame-Options` (XFO) header or the `frame-ancestors` directive in the `Content-Security-Policy` (CSP) header control where a page can be embedded. These can mitigate clickjacking attacks by blocking some or all sites from embedding the page. [Learn more about mitigating clickjacking](#).

Description	Severity
No frame control policy found	High

GENERAL

○ Detected JavaScript libraries   ⌃

All front-end JavaScript libraries detected on the page. [Learn more about this JavaScript library detection diagnostic audit](#).

Name	Version
Create React App	

🔺 Missing source maps for large first-party JavaScript   ⌃

Source maps translate minified code to the original source code. This helps developers debug in production. In addition, Lighthouse is able to provide further insights. Consider deploying source maps to take advantage of these benefits. [Learn more about source maps](#).

☑ Show 3rd-party resources (3)

URL	Map URL
slayd.in `1st Party`	
/assets/index-kSBiKzVV.js (slayd.in)	/assets/index-kSBiKzVV.js.map (slayd.in)
Large JavaScript file is missing a source map Error: Timed out fetching resource	
/assets/vendor-BQK9_sJn.js (slayd.in)	/assets/vendor-BQK9_sJn.js.map (slayd.in)
posthog.com	
/static/web-vitals.js?v=1.248.0 (us-assets.i.posthog.com)	/static/web-vitals.js.map (us-assets.i.posthog.com)
/static/recorder.js?v=1.248.0 (us-assets.i.posthog.com)	/static/recorder.js.map (us-assets.i.posthog.com)
/static/dead-clicks-autocapture.js?v=1.248.0 (us-assets.i.posthog.com)	/static/dead-clicks-autocapture.js.map (us-assets.i.posthog.com)

PASSED AUDITS (13)   Hide

🟢 Uses HTTPS   ⌃

All sites should be protected with HTTPS, even ones that don't handle sensitive data. This includes avoiding [mixed content](), where some resources are loaded over HTTP despite the initial request being served over HTTPS. HTTPS prevents intruders from tampering with or passively listening in on the communications between your app and your users, and is a prerequisite for HTTP/2 and many new web platform APIs. [Learn more about HTTPS]().

🟢 Avoids deprecated APIs   ⌃

Deprecated APIs will eventually be removed from the browser. [Learn more about deprecated APIs]().

🟢 Avoids third-party cookies   ⌃

Third-party cookies may be blocked in some contexts. [Learn more about preparing for third-party cookie restrictions]().

🟢 Allows users to paste into input fields   ⌃

Preventing input pasting is a bad practice for the UX, and weakens security by blocking password managers.[Learn more about user-friendly input fields]().

🟢 Avoids requesting the geolocation permission on page load   ⌃

Users are mistrustful of or confused by sites that request their location without context. Consider tying the request to a user action instead. [Learn more about the geolocation permission](#).

● Avoids requesting the notification permission on page load ⌃

Users are mistrustful of or confused by sites that request to send notifications without context. Consider tying the request to user gestures instead. [Learn more about responsibly getting permission for notifications](#).

● Displays images with correct aspect ratio ⌃

Image display dimensions should match natural aspect ratio. [Learn more about image aspect ratio](#).

● Serves images with appropriate resolution ⌃

Image natural dimensions should be proportional to the display size and the pixel ratio to maximize image clarity. [Learn how to provide responsive images](#).

● Has a `<meta name="viewport">` tag with `width` or `initial-scale` ⌃

A `<meta name="viewport">` not only optimizes your app for mobile screen sizes, but also prevents [a 300 millisecond delay to user input](#). [Learn more about using the viewport meta tag](#).

● Page has the HTML doctype ⌃

Specifying a doctype prevents the browser from switching to quirks-mode. [Learn more about the doctype declaration](#).

● Properly defines charset ⌃

A character encoding declaration is required. It can be done with a `<meta>` tag in the first 1024 bytes of the HTML or in the Content-Type HTTP response header. [Learn more about declaring the character encoding](#).

● No browser errors logged to the console ⌃

Errors logged to the console indicate unresolved problems. They can come from network request failures and other browser concerns. [Learn more about this errors in console diagnostic audit](#)

● No issues in the `Issues` panel in Chrome Devtools ⌃

Issues logged to the `Issues` panel in Chrome Devtools indicate unresolved problems. They can come from network request failures, insufficient security controls, and other browser concerns. Open up the

Issues panel in Chrome DevTools for more details on each issue.

NOT APPLICABLE (2)   Hide

○   Redirects HTTP traffic to HTTPS   ⌃

Make sure that you redirect all HTTP traffic to HTTPS in order to enable secure web features for all
your users. Learn more.

○   Document uses legible font sizes   ⌃

Font sizes less than 12px are too small to be legible and require mobile visitors to "pinch to zoom" in
order to read. Strive to have >60% of page text ≥12px. Learn more about legible font sizes.

100

SEO

These checks ensure that your page is following basic search engine
optimization advice. There are many additional factors Lighthouse
does not score here that may affect your search ranking, including
performance on Core Web Vitals. Learn more about Google Search
Essentials.

ADDITIONAL ITEMS TO MANUALLY CHECK (1)   Hide

○   Structured data is valid   ⌃

Run the Structured Data Testing Tool and the Structured Data Linter to validate structured data. Learn
more about Structured Data.

Run these additional validators on your site to check additional SEO best practices.

PASSED AUDITS (9)   Hide

●   Page isn't blocked from indexing   ⌃

Search engines are unable to include your pages in search results if they don't have permission to crawl them. Learn more about crawler directives.

● Document has a `<title>` element ⌃

The title gives screen reader users an overview of the page, and search engine users rely on it heavily to determine if a page is relevant to their search. Learn more about document titles.

● Document has a meta description ⌃

Meta descriptions may be included in search results to concisely summarize page content. Learn more about the meta description.

● Page has successful HTTP status code ⌃

Pages with unsuccessful HTTP status codes may not be indexed properly. Learn more about HTTP status codes.

● Links have descriptive text ⌃

Descriptive link text helps search engines understand your content. Learn how to make links more accessible.

● Links are crawlable ⌃

Search engines may use `href` attributes on links to crawl websites. Ensure that the `href` attribute of anchor elements links to an appropriate destination, so more pages of the site can be discovered. Learn how to make links crawlable

● robots.txt is valid ⌃

If your robots.txt file is malformed, crawlers may not be able to understand how you want your website to be crawled or indexed. Learn more about robots.txt.

● Image elements have `[alt]` attributes ⌃

Informative elements should aim for short, descriptive alternate text. Decorative elements can be ignored with an empty alt attribute. Learn more about the `alt` attribute.

● Document has a valid `hreflang` ⌃

hreflang links tell search engines what version of a page they should list in search results for a given language or region. Learn more about `hreflang`.

NOT APPLICABLE (1)   Hide

○   Document has a valid `rel=canonical`   ⌃

Canonical links suggest which URL to show in search results. Learn more about canonical links.

📅 Captured at Jul 1, 2025, 10:56 PM   🖥 Emulated Desktop with Lighthouse   👥 Single page session
    GMT+5:30   12.6.0
⏱ Initial page load   📡 Custom throttling   ⊚ Using Chromium 138.0.0.0 with
                                                                                          devtools

Generated by **Lighthouse** 12.6.0 | File an issue