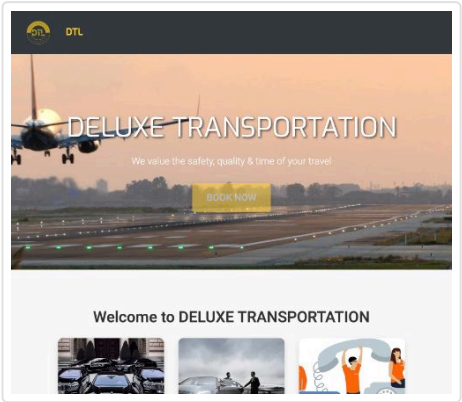


Performance

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



METRICS

Expand view

First Contentful Paint

0.9 s

Largest Contentful Paint

1.5 s

Total Blocking Time

0 ms

Cumulative Layout Shift

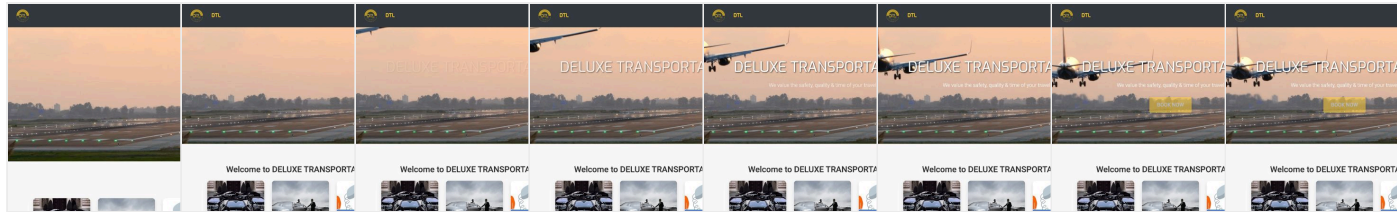
0.017

Speed Index

0.9 s



[View Treemap](#)



Show audits relevant to: [All](#) [FCP](#) [LCP](#) [TBT](#) [CLS](#)

DIAGNOSTICS

- ▲ Properly size images — Potential savings of 817 KiB ▼
- ▲ Eliminate render-blocking resources — Potential savings of 460 ms ▼
- ▲ Serve images in next-gen formats — Potential savings of 384 KiB ▼
- ▲ Preconnect to required origins — Potential savings of 110 ms ▼
- ▲ Largest Contentful Paint element — 1,530 ms ▼
- ▲ Page prevented back/forward cache restoration — 1 failure reason ▼
- Image elements do not have explicit [width](#) and [height](#) ▼
- Serve static assets with an efficient cache policy — 5 resources found ▼
- Ensure text remains visible during webfont load ▼
- Avoid serving legacy JavaScript to modern browsers — Potential savings of 8 KiB ▼

Reduce unused CSS — Potential savings of 26 KiB	▼
Reduce unused JavaScript — Potential savings of 342 KiB	▼
○ Avoid large layout shifts — 1 layout shift found	▼
○ Initial server response time was short — Root document took 20 ms	▼
○ Avoids enormous network payloads — Total size was 1,123 KiB	▼
○ Avoids an excessive DOM size — 77 elements	▼
○ Avoid chaining critical requests — 7 chains found	▼
○ JavaScript execution time — 0.1 s	▼
○ Minimizes main-thread work — 0.3 s	▼
○ Minimize third-party usage — Third-party code blocked the main thread for 0 ms	▼
○ Avoid long main-thread tasks — 1 long task found	▼

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

PASSED AUDITS (17)

Show



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

▲ Links do not have a discernible name



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.

NAVIGATION

▲ Heading elements are not in a sequentially-descending order



These are opportunities to improve keyboard navigation in your application.

AUDIO AND VIDEO

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



These are opportunities to provide alternative content for audio and video. This may improve the experience for users with hearing or vision impairments.

ADDITIONAL ITEMS TO MANUALLY CHECK (10)

Show

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

PASSED AUDITS (10)

Show

NOT APPLICABLE (44)

Show



Best Practices

GENERAL

▲ Uses deprecated APIs — 1 warning found



▲ Browser errors were logged to the console



TRUST AND SAFETY

○ Ensure CSP is effective against XSS attacks

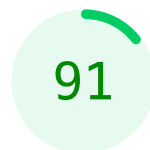


PASSED AUDITS (12)

Show

NOT APPLICABLE (3)

Show



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](#).

CONTENT BEST PRACTICES

▲ Document does not have a meta description



Format your HTML in a way that enables crawlers to better understand your app's content.

ADDITIONAL ITEMS TO MANUALLY CHECK (1)

Show

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

PASSED AUDITS (7)

Show

NOT APPLICABLE (2)

Show

■ Captured at Sep 4, 2024,
10:27 PM GMT+1

■ Initial page load

■ Emulated Desktop with
Lighthouse 12.1.0

■ Custom throttling

■ Single page session

■ Using Chromium 128.0.0.0
with devtools