Performance

**Metrics**

<table>
<thead>
<tr>
<th>Metric</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>First Contentful Paint</td>
<td>0.5 s</td>
</tr>
<tr>
<td>Speed Index</td>
<td>0.8 s</td>
</tr>
<tr>
<td>Largest Contentful Paint</td>
<td>0.7 s</td>
</tr>
<tr>
<td>Time to Interactive</td>
<td>0.5 s</td>
</tr>
<tr>
<td>Total Blocking Time</td>
<td>0 ms</td>
</tr>
<tr>
<td>Cumulative Layout Shift</td>
<td>0.008</td>
</tr>
</tbody>
</table>

Values are estimated and may vary. The performance score is calculated directly from these metrics. See calculator.

**Opportunities** — These suggestions can help your page load faster. They don’t directly affect the Performance score.

**Opportunity**

Preload key requests

Consider using `&lt;link rel=preload&gt;` to prioritize fetching resources that are currently requested later in page load. Learn more.

**URL**

...css/fontawesome.min.css?5.13.0_1 (www.unipg.it) 280 ms

Eliminate render-blocking resources

Resources are blocking the first paint of your page. Consider delivering critical JS/CSS inline and deferring all non-critical JS/styles. Learn more.

**URL**

...css/bootstrap.min.css?4.5.3_1 (www.unipg.it) 27.3 KiB 120 ms
### URL Diagnostics

More information about the performance of your application. These numbers don't directly affect the Performance score.

#### Image elements do not have explicit width and height

Set an explicit width and height on image elements to reduce layout shifts and improve CLS. Learn more.

<table>
<thead>
<tr>
<th>URL</th>
<th>Protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="https://www.unipg.it">https://www.unipg.it</a></td>
<td>http/1.1</td>
</tr>
<tr>
<td>...css/bootstrap.min.css?4.5.3_1</td>
<td>http/1.1</td>
</tr>
<tr>
<td>...css/custom.min.css?1.1.0</td>
<td>http/1.1</td>
</tr>
<tr>
<td>...images/logo-260.png</td>
<td>http/1.1</td>
</tr>
<tr>
<td>...2020-11-25-sportello-antiviolenza/sau-2-1350.webp</td>
<td>http/1.1</td>
</tr>
<tr>
<td>...20201118-sharper-2020/sharper-1350.webp</td>
<td>http/1.1</td>
</tr>
<tr>
<td>...20201113-immatricolazioni/immatricolazioni-1350.webp</td>
<td>http/1.1</td>
</tr>
<tr>
<td>...js/jquery.min.js?3.5.1</td>
<td>http/1.1</td>
</tr>
<tr>
<td>...js/bootstrap.bundle.min.js?4.5.3_1</td>
<td>http/1.1</td>
</tr>
<tr>
<td>...js/custom.min.js?1.1.0</td>
<td>http/1.1</td>
</tr>
<tr>
<td>...TitilliumWeb/titillium-web-v7-latin-regular.woff2</td>
<td>http/1.1</td>
</tr>
<tr>
<td>...TitilliumWeb/titillium-web-v7-latin-italic.woff2</td>
<td>http/1.1</td>
</tr>
<tr>
<td>...TitilliumWeb/titillium-web-v7-latin-700.woff2</td>
<td>http/1.1</td>
</tr>
<tr>
<td>...TitilliumWeb/titillium-web-v7-latin-700italic.woff2</td>
<td>http/1.1</td>
</tr>
<tr>
<td>...css/fontawesome.min.css?5.13.0_1</td>
<td>http/1.1</td>
</tr>
<tr>
<td>...webfonts/fa-solid-900.woff2?5.0.13</td>
<td>http/1.1</td>
</tr>
<tr>
<td>...webfonts/fa-brands-400.woff2?5.13.0</td>
<td>http/1.1</td>
</tr>
<tr>
<td>...webfonts/fa-regular-400.woff2?5.13.0</td>
<td>http/1.1</td>
</tr>
<tr>
<td>/manifest.json</td>
<td>http/1.1</td>
</tr>
<tr>
<td>/service-worker.js.php</td>
<td>http/1.1</td>
</tr>
<tr>
<td>...images/logo-130.png</td>
<td>http/1.1</td>
</tr>
<tr>
<td>...images/logo-footer.png</td>
<td>http/1.1</td>
</tr>
<tr>
<td>/images/icon-logo-192.png</td>
<td>http/1.1</td>
</tr>
</tbody>
</table>
Avoid an excessive DOM size — 1,302 elements

A large DOM will increase memory usage, cause longer style calculations, and produce costly layout reflows. Learn more.

<table>
<thead>
<tr>
<th>Statistic</th>
<th>Element</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total DOM Elements</td>
<td></td>
<td>NaN</td>
</tr>
<tr>
<td>Maximum DOM Depth</td>
<td>&lt;em class=&quot;far fa-compass d-block fa-4x text-info py-2&quot;&gt;</td>
<td>17</td>
</tr>
<tr>
<td>Maximum Child Elements</td>
<td>&lt;ul class=&quot;list-level-1 collapse dropdown-menu&quot;&gt;</td>
<td>19</td>
</tr>
</tbody>
</table>

Avoid chaining critical requests — 11 chains found

The Critical Request Chains below show you what resources are loaded with a high priority. Consider reducing the length of chains, reducing the download size of resources, or deferring the download of unnecessary resources to improve page load. Learn more.

Maximum critical path latency: 1,330 ms

Initial Navigation

https://www.unipg.it

- …css/bootstrap.min.css?4.5.3_1 (www.unipg.it) - 70 ms, 27.34 KiB
- …css/custom.min.css?1.1.0 (www.unipg.it) - 180 ms, 15.08 KiB
- …js/jquery.min.js?3.5.1 (www.unipg.it)
  - …css/fontawesome.min.css?5.13.0_1 (www.unipg.it) - 110 ms, 92.37 KiB
  - …webfonts/fa-solid-900.woff2?5.0.13 (www.unipg.it) - 110 ms, 92.37 KiB
  - …webfonts/fa-brands-400.woff2?5.13.0 (www.unipg.it) - 130 ms, 95.02 KiB
  - …webfonts/fa-regular-400.woff2?5.13.0 (www.unipg.it) - 110 ms, 19.21 KiB
- …js/bootstrap.bundle.min.js?4.5.3_1 (www.unipg.it) - 150 ms, 21.83 KiB
- …js/custom.min.js?1.1.0 (www.unipg.it) - 150 ms, 5.89 KiB
- …TitilliumWeb/titillium-web-v7-latin-regular.woff2 (www.unipg.it) - 110 ms, 24.73 KiB
- …TitilliumWeb/titillium-web-v7-latin-italic.woff2 (www.unipg.it) - 110 ms, 14.03 KiB
- …TitilliumWeb/titillium-web-v7-latin-700.woff2 (www.unipg.it) - 140 ms, 12.02 KiB
- …TitilliumWeb/titillium-web-v7-latin-700italic.woff2 (www.unipg.it) - 150 ms, 13.75 KiB

Keep request counts low and transfer sizes small — 34 requests • 1,057 KiB

To set budgets for the quantity and size of page resources, add a budget.json file. Learn more.

<table>
<thead>
<tr>
<th>Resource Type</th>
<th>Requests</th>
<th>Transfer Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total</td>
<td>34</td>
<td>1,056.6 KiB</td>
</tr>
<tr>
<td>Image</td>
<td>5</td>
<td>407.3 KiB</td>
</tr>
<tr>
<td>Font</td>
<td>7</td>
<td>271.1 KiB</td>
</tr>
<tr>
<td>Script</td>
<td>5</td>
<td>161.7 KiB</td>
</tr>
<tr>
<td>Other</td>
<td>13</td>
<td>142.7 KiB</td>
</tr>
<tr>
<td>Resource Type</td>
<td>Requests</td>
<td>Transfer Size</td>
</tr>
<tr>
<td>---------------</td>
<td>----------</td>
<td>---------------</td>
</tr>
<tr>
<td>Stylesheet</td>
<td>3</td>
<td>55.2 KiB</td>
</tr>
<tr>
<td>Document</td>
<td>1</td>
<td>18.6 KiB</td>
</tr>
<tr>
<td>Media</td>
<td>0</td>
<td>0 KiB</td>
</tr>
<tr>
<td>Third-party</td>
<td>3</td>
<td>97.2 KiB</td>
</tr>
</tbody>
</table>

Largest Contentful Paint element — 1 element found

This is the largest contentful element painted within the viewport. Learn More

Element

img.w-100

Avoid large layout shifts — 5 elements found

These DOM elements contribute most to the CLS of the page.

<table>
<thead>
<tr>
<th>Element</th>
<th>CLS Contribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>ul.navbar-nav.mr-auto</td>
<td>0.003</td>
</tr>
<tr>
<td>ul.navbar-nav</td>
<td>0.001</td>
</tr>
<tr>
<td>ul.headerlist.list-inline</td>
<td>0.001</td>
</tr>
<tr>
<td>div.navbar-nav</td>
<td>0.001</td>
</tr>
<tr>
<td>div#accreditamento-miur.up-fontsize-80.pb-sm-2.font-italic.m-auto.m-sm-0.text-dark</td>
<td>0</td>
</tr>
</tbody>
</table>

Passed audits (26)

Properly size images

Serve images that are appropriately-sized to save cellular data and improve load time. Learn more.

Defer offscreen images

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

Minify CSS

Minifying CSS files can reduce network payload sizes. Learn more.

Minify JavaScript

Minifying JavaScript files can reduce payload sizes and script parse time. Learn more.

Remove unused CSS — Potential savings of 32 KiB

Remove dead rules from stylesheets and defer the loading of CSS not used for above-the-fold content to reduce unnecessary bytes consumed by network activity. Learn more.
URL | Transfer Size | Potential Savings
--- | --- | ---
...css/fontawesome.min.css?5.13.0_1 | 12.8 KiB | 12.3 KiB

Remove unused JavaScript — Potential savings of 54 KiB

Remove unused JavaScript to reduce bytes consumed by network activity. Learn more.

URL | Transfer Size | Potential Savings
--- | --- | ---
/gtag/js?id=UA-54353041-2 | 60 KiB | 32.5 KiB
...js/jquery.min.js?3.5.1 | 37.2 KiB | 21.7 KiB

Efficiently encode images

Optimized images load faster and consume less cellular data. Learn more.

Serve images in next-gen formats

Image formats like JPEG 2000, JPEG XR, and WebP often provide better compression than PNG or JPEG, which means faster downloads and less data consumption. Learn more.

Enable text compression

Text-based resources should be served with compression (gzip, deflate or brotli) to minimize total network bytes. Learn more.

Preconnect to required origins — Potential savings of 80 ms

Consider adding ‘preconnect’ or ‘dns-prefetch’ resource hints to establish early connections to important third-party origins. Learn more.

URL | Potential Savings
--- | ---
https://www.google-analytics.com | 80 ms

Initial server response time was short — Root document took 480 ms

Keep the server response time for the main document short because all other requests depend on it. Learn more.

Avoid multiple page redirects

Redirects introduce additional delays before the page can be loaded. Learn more.

Use video formats for animated content

Large GIFs are inefficient for delivering animated content. Consider using MPEG4/WebM videos for animations and PNG/WebP for static images instead of GIF to save network bytes. Learn more

Remove duplicate modules in JavaScript bundles

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

Avoid serving legacy JavaScript to modern browsers — Potential savings of 0 KiB

Polyfills and transforms enable legacy browsers to use new JavaScript features. However, many aren’t necessary for modern browsers. For your bundled JavaScript, adopt a modern script deployment strategy using module/nomodule feature detection to reduce the amount of code shipped to modern browsers, while retaining support for legacy browsers. Learn More
### Avoids enormous network payloads — Total size was 1,057 KiB

Large network payloads cost users real money and are highly correlated with long load times. [Learn more.]

### Uses efficient cache policy on static assets — 1 resource found

A long cache lifetime can speed up repeat visits to your page. [Learn more.]

### 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.]

### JavaScript execution time — 0.1 s

Consider reducing the time spent parsing, compiling, and executing JS. You may find delivering smaller JS payloads helps with this. [Learn more.]

### Minimizes main-thread work — 0.5 s

---

<table>
<thead>
<tr>
<th>URL</th>
<th>Potential Savings</th>
<th>Transfer Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>../js/bootstrap.bundle.min.js?4.5.3_1 (<a href="http://www.unipg.it">www.unipg.it</a>)</td>
<td></td>
<td>0 KiB</td>
</tr>
<tr>
<td>bootstrap.bundle.min.js:6</td>
<td></td>
<td>@babel/plugin-transform-classes</td>
</tr>
<tr>
<td>URL</td>
<td>Transfer Size</td>
<td></td>
</tr>
<tr>
<td>../20211118-sharper-2020/sharper-1350.webp (<a href="http://www.unipg.it">www.unipg.it</a>)</td>
<td>160.6 KiB</td>
<td></td>
</tr>
<tr>
<td>../2020-11-25-sportello-antiviolenza/sau-2-1350.webp (<a href="http://www.unipg.it">www.unipg.it</a>)</td>
<td>105.2 KiB</td>
<td></td>
</tr>
<tr>
<td>../webfonts/fa-brands-400.woff2?5.13.0 (<a href="http://www.unipg.it">www.unipg.it</a>)</td>
<td>95 KiB</td>
<td></td>
</tr>
<tr>
<td>../webfonts/fa-solid-900.woff2?5.0.13 (<a href="http://www.unipg.it">www.unipg.it</a>)</td>
<td>92.4 KiB</td>
<td></td>
</tr>
<tr>
<td>../20201113-immatricolazioni/immatricolazioni-1350.webp (<a href="http://www.unipg.it">www.unipg.it</a>)</td>
<td>91.1 KiB</td>
<td></td>
</tr>
<tr>
<td>/gtag/js?id=UA-54353041-2 (<a href="http://www.googletagmanager.com">www.googletagmanager.com</a>)</td>
<td>60 KiB</td>
<td></td>
</tr>
<tr>
<td>../js/jquery.min.js?3.5.1 (<a href="http://www.unipg.it">www.unipg.it</a>)</td>
<td>37.2 KiB</td>
<td></td>
</tr>
<tr>
<td>/analytics.js (<a href="http://www.google-analytics.com">www.google-analytics.com</a>)</td>
<td>36.8 KiB</td>
<td></td>
</tr>
<tr>
<td><a href="https://www.unipg.it">https://www.unipg.it</a></td>
<td>36.2 KiB</td>
<td></td>
</tr>
<tr>
<td>../js/jquery.min.js?3.5.1 (<a href="http://www.unipg.it">www.unipg.it</a>)</td>
<td>30.7 KiB</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>URL</th>
<th>Cache TTL</th>
<th>Transfer Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>/analytics.js (<a href="http://www.google-analytics.com">www.google-analytics.com</a>)</td>
<td>2 h</td>
<td>37 KiB</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>URL</th>
<th>Total CPU Time</th>
<th>Script Evaluation</th>
<th>Script Parse</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="https://www.unipg.it">https://www.unipg.it</a></td>
<td>187 ms</td>
<td>10 ms</td>
<td>1 ms</td>
</tr>
<tr>
<td>Unattributable</td>
<td>119 ms</td>
<td>3 ms</td>
<td>0 ms</td>
</tr>
<tr>
<td>../js/jquery.min.js?3.5.1 (<a href="http://www.unipg.it">www.unipg.it</a>)</td>
<td>57 ms</td>
<td>44 ms</td>
<td>3 ms</td>
</tr>
</tbody>
</table>

- URL: File path to the JavaScript file.
- Potential Savings: Size reduction opportunities.
- Transfer Size: Size of the file after compression.
- Cache TTL: Lifetime of the cache for static assets.
- Total CPU Time: Time spent in script evaluation.
- Script Evaluation: Time spent parsing and compiling.
- Script Parse: Time spent executing.

---
Consider reducing the time spent parsing, compiling and executing JS. You may find delivering smaller JS payloads helps with this. Learn more

<table>
<thead>
<tr>
<th>Category</th>
<th>Time Spent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Other</td>
<td>143 ms</td>
</tr>
<tr>
<td>Script Evaluation</td>
<td>137 ms</td>
</tr>
<tr>
<td>Style &amp; Layout</td>
<td>108 ms</td>
</tr>
<tr>
<td>Rendering</td>
<td>41 ms</td>
</tr>
<tr>
<td>Parse HTML &amp; CSS</td>
<td>31 ms</td>
</tr>
<tr>
<td>Script Parsing &amp; Compilation</td>
<td>25 ms</td>
</tr>
</tbody>
</table>

All text remains visible during webfont loads

Leverage the font-display CSS feature to ensure text is user-visible while webfonts are loading. Learn more.

Minimize third-party usage — Third-party code blocked the main thread for 0 ms

Third-party code can significantly impact load performance. Limit the number of redundant third-party providers and try to load third-party code after your page has primarily finished loading. Learn more.

<table>
<thead>
<tr>
<th>Third-Party</th>
<th>Transfer Size</th>
<th>Main-Thread Blocking Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Google Tag Manager</td>
<td>60 KiB</td>
<td>0 ms</td>
</tr>
<tr>
<td>Google Analytics</td>
<td>37 KiB</td>
<td>0 ms</td>
</tr>
</tbody>
</table>

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.

Avoids `document.write()`

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

Avoid long main-thread tasks

Lists the longest tasks on the main thread, useful for identifying worst contributors to input delay. Learn more

Avoid non-composited animations

Animations which are not composited can be janky and contribute to CLS. Learn more

100

Accessibility

These checks highlight opportunities to improve the accessibility of your web app. Only a subset of accessibility issues can be automatically detected so manual testing is also encouraged.

Additional items to manually check (10) — These items address areas which an automated testing tool cannot cover. Learn
The page has a logical tab order
Tabbing through the page follows the visual layout. Users cannot focus elements that are offscreen. Learn more.

Interactive controls are keyboard focusable
Custom interactive controls are keyboard focusable and display a focus indicator. Learn more.

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 more.

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 more.

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 more.

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

Custom controls have ARIA roles
Custom interactive controls have appropriate ARIA roles. Learn more.

Visual order on the page follows DOM order
DOM order matches the visual order, improving navigation for assistive technology. Learn more.

Offscreen content is hidden from assistive technology
Offscreen content is hidden with display: none or aria-hidden=true. Learn more.

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.

Passed audits (23)

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

[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 more.

[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 more.

[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.
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.

[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.

[role] values are valid

ARIA roles must have valid values in order to perform their intended accessibility functions. Learn more.

[aria-*] attributes have valid values

Assistive technologies, like screen readers, can't interpret ARIA attributes with invalid values. Learn more.

[aria-*] attributes are valid and not misspelled

Assistive technologies, like screen readers, can't interpret ARIA attributes with invalid names. Learn more.

Buttons 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 more.

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.

Background and foreground colors have a sufficient contrast ratio

Low-contrast text is difficult or impossible for many users to read. Learn more.

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.

[id] attributes on active, focusable elements are unique

All focusable elements must have a unique 'id' to ensure that they're visible to assistive technologies. Learn more.

ARIA IDs are unique

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

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.

<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.

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

Specifying a valid BCP 47 language helps screen readers announce text properly. Learn more.

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](#).

<table>
<thead>
<tr>
<th>Links have a discernible name</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>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. <a href="#">Learn more</a>.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Lists contain only <code>&lt;li&gt;</code> elements and script supporting elements (<code>&lt;script&gt;</code> and <code>&lt;template&gt;</code>).</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Screen readers have a specific way of announcing lists. Ensuring proper list structure aids screen reader output. <a href="#">Learn more</a>.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>List items (<code>&lt;li&gt;</code>) are contained within <code>&lt;ul&gt;</code> or <code>&lt;ol&gt;</code> parent elements</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Screen readers require list items (<code>&lt;li&gt;``) to be contained within a parent </code>&lt;ul&gt;<code>or</code>&lt;ol&gt;` to be announced properly. <a href="#">Learn more</a>.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><code>[user-scalable=&quot;no&quot;]</code> is not used in the <code>&lt;meta name=&quot;viewport&quot;&gt;</code> element and the <code>[maximum-scale]</code> attribute is not less than 5.</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Disabling zooming is problematic for users with low vision who rely on screen magnification to properly see the contents of a web page. <a href="#">Learn more</a>.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Not applicable (18)</th>
<th></th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th><code>[accesskey]</code> values are unique</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Access keys let users quickly focus a part of the page. For proper navigation, each access key must be unique. <a href="#">Learn more</a>.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ARIA input fields have accessible names</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>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. <a href="#">Learn more</a>.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ARIA toggle fields have accessible names</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>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. <a href="#">Learn more</a>.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><code>&lt;dl&gt;</code>'s contain only properly-ordered <code>&lt;dt&gt;</code> and <code>&lt;dd&gt;</code> groups, <code>&lt;script&gt;</code>, <code>&lt;template&gt;</code>, or <code>&lt;div&gt;</code> elements.</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>When definition lists are not properly marked up, screen readers may produce confusing or inaccurate output. <a href="#">Learn more</a>.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Definition list items are wrapped in <code>&lt;dl&gt;</code> elements</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Definition list items (<code>&lt;dt&gt;</code> and <code>&lt;dd&gt;</code>) must be wrapped in a parent <code>&lt;dl&gt;</code> element to ensure that screen readers can properly announce them. <a href="#">Learn more</a>.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>No form fields have multiple labels</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>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. <a href="#">Learn more</a>.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><code>&lt;frame&gt;</code> or <code>&lt;iframe&gt;</code> elements have a title</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Screen reader users rely on frame titles to describe the contents of frames. <a href="#">Learn more</a>.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><code>&lt;input type=&quot;image&quot;&gt;</code> elements have <code>[alt]</code> text</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>When an image is being used as an <code>&lt;input&gt;</code> button, providing alternative text can help screen reader users understand the purpose of the button. <a href="#">Learn more</a>.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Form elements have associated labels</th>
<th></th>
</tr>
</thead>
</table>
Labels ensure that form controls are announced properly by assistive technologies, like screen readers. [Learn more.]

Presentational `<table>` elements avoid using `<th>, <caption>` or the `[summary]` attribute.

A table being used for layout purposes should not include data elements, such as the th or caption elements or the summary attribute, because this can create a confusing experience for screen reader users. [Learn more.]

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.]

`<object>` elements have `[alt]` text

Screen readers cannot translate non-text content. Adding alt text to `object` elements helps screen readers convey meaning to users. [Learn more.]

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.]

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.]

`<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.]

`[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 more.]

`<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.]

`<video>` elements contain a `<track>` element with `[kind="description"]`

Audio descriptions provide relevant information for videos that dialogue cannot, such as facial expressions and scenes. [Learn more.]

---

Best Practices

Passed audits (15)

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.

Links to cross-origin destinations are safe

Add `rel="noopener"` or `rel="noreferrer"` to any external links to improve performance and prevent security vulnerabilities. Learn more.

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.

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.

Avoids front-end JavaScript libraries with known security vulnerabilities

Some third-party scripts may contain known security vulnerabilities that are easily identified and exploited by attackers. Learn more.

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.

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.

Avoids front-end JavaScript libraries with known security vulnerabilities

Some third-party scripts may contain known security vulnerabilities that are easily identified and exploited by attackers. Learn more.

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.

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.

Avoids front-end JavaScript libraries with known security vulnerabilities

Some third-party scripts may contain known security vulnerabilities that are easily identified and exploited by attackers. Learn more.

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.

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.

Avoids front-end JavaScript libraries with known security vulnerabilities

Some third-party scripts may contain known security vulnerabilities that are easily identified and exploited by attackers. Learn more.

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.

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.

Avoids front-end JavaScript libraries with known security vulnerabilities

Some third-party scripts may contain known security vulnerabilities that are easily identified and exploited by attackers. Learn more.

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.

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.

Avoids front-end JavaScript libraries with known security vulnerabilities

Some third-party scripts may contain known security vulnerabilities that are easily identified and exploited by attackers. Learn more.

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.

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.

Avoids front-end JavaScript libraries with known security vulnerabilities

Some third-party scripts may contain known security vulnerabilities that are easily identified and exploited by attackers. Learn more.

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.

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.

Avoids front-end JavaScript libraries with known security vulnerabilities

Some third-party scripts may contain known security vulnerabilities that are easily identified and exploited by attackers. Learn more.

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.

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.

Avoids front-end JavaScript libraries with known security vulnerabilities

Some third-party scripts may contain known security vulnerabilities that are easily identified and exploited by attackers. Learn more.

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.

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.

Avoids front-end JavaScript libraries with known security vulnerabilities

Some third-party scripts may contain known security vulnerabilities that are easily identified and exploited by attackers. Learn more.

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.

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.

Avoids front-end JavaScript libraries with known security vulnerabilities

Some third-party scripts may contain known security vulnerabilities that are easily identified and exploited by attackers. Learn more.

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.

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.

Avoids front-end JavaScript libraries with known security vulnerabilities

Some third-party scripts may contain known security vulnerabilities that are easily identified and exploited by attackers. Learn more.

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.

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.

Avoids front-end JavaScript libraries with known security vulnerabilities

Some third-party scripts may contain known security vulnerabilities that are easily identified and exploited by attackers. Learn more.

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.

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.

Avoids front-end JavaScript libraries with known security vulnerabilities

Some third-party scripts may contain known security vulnerabilities that are easily identified and exploited by attackers. Learn more.

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.

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.
These checks ensure that your page is optimized for search engine results ranking. There are additional factors Lighthouse does not check that may affect your search ranking. Learn more.

Additional items to manually check (1) — Run these additional validators on your site to check additional SEO best practices.

Structured data is valid

Run the [Structured Data Testing Tool](https://developer.mozilla.org/en-US/docs/Web/Performance/Structured_data_testing_tool) and the [Structured Data Linter](https://developers.google.com/search/docs/structured-data/lint) to validate structured data. Learn more.

Passed audits (12)

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

Add a `<meta name="viewport">` tag to optimize your app for mobile screens. Learn more.

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.

Document has a meta description

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

Page has successful HTTP status code

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

Links have descriptive text

Descriptive link text helps search engines understand your content. Learn more.

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 More

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.

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.

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.

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.

Document has a valid rel=canonical

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

Document avoids plugins

Search engines can't index plugin content, and many devices restrict plugins or don't support them. 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.

Tap targets are sized appropriately

Interactive elements like buttons and links should be large enough (48x48px), and have enough space around them, to be easy enough to tap without overlapping onto other elements. Learn more.

Progressive Web App

These checks validate the aspects of a Progressive Web App. Learn more.

Fast and reliable

Page load is fast enough on mobile networks

A fast page load over a cellular network ensures a good mobile user experience. Learn more.

Current page responds with a 200 when offline

If you're building a Progressive Web App, consider using a service worker so that your app can work offline. Learn more.

A service worker enables your web app to be reliable in unpredictable network conditions. Learn more.

Installable

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.

Registers a service worker that controls page and start_url

The service worker is the technology that enables your app to use many Progressive Web App features, such as offline, add to homescreen, and push notifications. Learn more.

Web app manifest meets the installability requirements

Browsers can proactively prompt users to add your app to their homescreen, which can lead to higher engagement. Learn more.

PWA Optimized

Redirects HTTP traffic to HTTPS

If you've already set up HTTPS, make sure that you redirect all HTTP traffic to HTTPS in order to enable secure web features for all your users. Learn more.

Configured for a custom splash screen

A themed splash screen ensures a high-quality experience when users launch your app from their homescreens. Learn more.

Sets a theme color for the address bar.

The browser address bar can be themed to match your site. Learn more.

Content is sized correctly for the viewport

If the width of your app's content doesn't match the width of the viewport, your app might not be optimized for mobile screens. Learn more.

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

Add a `</meta name="viewport">` tag to optimize your app for mobile screens. Learn more.

Contains some content when JavaScript is not available

Your app should display some content when JavaScript is disabled, even if it's just a warning to the user that JavaScript is required to use the app. Learn more.

Provides a valid apple-touch-icon

For ideal appearance on iOS when users add a progressive web app to the home screen, define an `apple-touch-icon`. It must point to a non-transparent 192px (or 180px) square PNG. Learn More.

Manifest has a maskable icon

A maskable icon ensures that the image fills the entire shape without being letterboxed when installing the app on a device. Learn more.

Additional items to manually check (3) — These checks are required by the baseline PWA Checklist but are not automatically checked by Lighthouse. They do not affect your score but it's important that you verify them manually.

Site works cross-browser

To reach the most number of users, sites should work across every major browser. Learn more.
Page transitions don't feel like they block on the network

Transitions should feel snappy as you tap around, even on a slow network. This experience is key to a user's perception of performance. [Learn more.](#)

Each page has a URL

Ensure individual pages are deep linkable via URL and that URLs are unique for the purpose of shareability on social media. [Learn more.](#)

<table>
<thead>
<tr>
<th>Runtime Settings</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>URL</td>
<td><a href="https://www.unipg.it/">https://www.unipg.it/</a></td>
</tr>
<tr>
<td>Fetch Time</td>
<td>Nov 24, 2020, 6:05 PM GMT+1</td>
</tr>
<tr>
<td>Device</td>
<td>Emulated Desktop</td>
</tr>
<tr>
<td>Network throttling</td>
<td>40 ms TCP RTT, 10,240 Kbps throughput (Simulated)</td>
</tr>
<tr>
<td>CPU throttling</td>
<td>1x slowdown (Simulated)</td>
</tr>
<tr>
<td>Channel</td>
<td>devtools</td>
</tr>
<tr>
<td>User agent (host)</td>
<td>Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36</td>
</tr>
<tr>
<td>User agent (network)</td>
<td>Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/84.0.4143.7 Safari/537.36 Chrome-Lighthouse</td>
</tr>
<tr>
<td>CPU/Memory Power</td>
<td>323</td>
</tr>
</tbody>
</table>

Generated by [Lighthouse 6.2.0](#) | [File an issue](#)