



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.

- Print Summary
- Print Expanded
- Copy JSON
- Save as HTML
- Save as JSON
- Open in Viewer
- Toggle Dark Theme

Contrast — These are opportunities to improve the legibility of your content.

▲ **Background and foreground colors do not have a sufficient contrast ratio.** ^

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

Failing Elements

span.tweet-date

span.tweet-date

Names and labels — 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.

▲ **Links do not 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 more](#).

Failing Elements

a#toggle-icon.toggle.responsive-menu-toggle-icon

a

a

a

a

a

a

a

a

Additional items to manually check (11) — These items address areas which an automated testing tool cannot cover. ^

Learn more in our guide on [conducting an accessibility review](#).

● **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.](#)
- Headings don't skip levels** ^

Headings are used to create an outline for the page and heading levels are not skipped. [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 (17) ^

- `[aria-*`] attributes match their roles** ^

Each ARIA `role` supports a specific subset of `aria-*` attributes. Mismatching these invalidates the `aria-*` attributes. [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.](#)

● **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.](#)

● **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 the page are unique** ^

The value of an id attribute must be unique to prevent other instances from being overlooked by 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.](#)

● **Lists contain only 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.](#)

● **List items () are contained within or parent elements** ^

Screen readers require list items () to be contained within a parent or to be announced properly. [Learn more.](#)

● **[user-scalable="no"] is not used in the <meta name="viewport"> element and the [maximum-scale] attribute is not 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.](#)

● **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.](#)

Not applicable (16)

● [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.](#)

● <audio> elements contain a <track> element with [kind="captions"]

Captions make audio elements usable for deaf or hearing-impaired users, providing critical information such as who is talking, what they're saying, and other non-speech information. [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.](#)

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

When definition lists are not properly marked up, screen readers may produce confusing or inaccurate output. [Learn more.](#)

● Definition list items are wrapped in <dl> elements

Definition list items (`<dt>` and `Learn more.

● <frame> or <iframe> elements have a title

Screen reader users rely on frame titles to describe the contents of frames. [Learn more.](#)

● <input type="image"> elements have [alt] text

When an image is being used as an `` button, providing alternative text can help screen reader users understand the purpose of the button. [Learn more.](#)

● Form elements have associated labels

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 `` elements helps screen readers convey meaning to users. [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 ` ` 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.](#)

Runtime Settings

URL	https://www.ddfire.gov.uk/
Fetch time	Jul 15, 2020, 11:03 AM GMT+1
Device	Emulated Nexus 5X
Network throttling	150 ms TCP RTT, 1,638.4 Kbps throughput (Simulated)
CPU throttling	4x slowdown (Simulated)
User agent (host)	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_4) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/83.0.4103.116 Safari/537.36
User agent (network)	Mozilla/5.0 (Linux; Android 6.0.1; Nexus 5 Build/MRA58N) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/74.0.3694.0 Mobile Safari/537.36 Chrome-Lighthouse
CPU/Memory Power	1182

Generated by **Lighthouse** 5.7.1 | [File an issue](#)