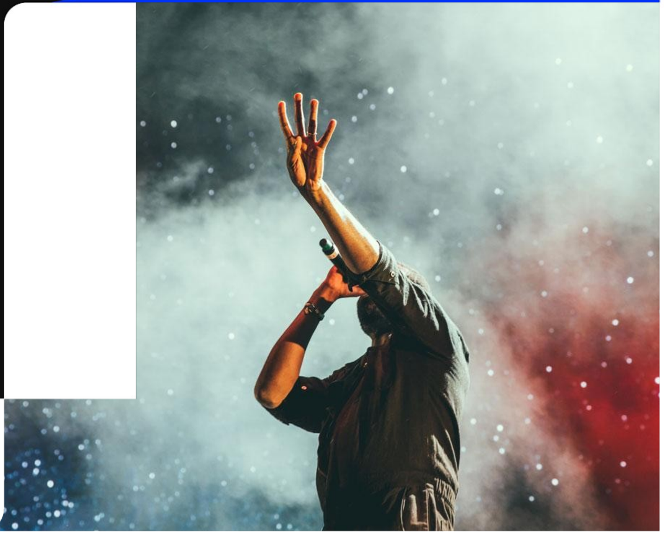


Accessibility & Semantic HTML

weezcrew

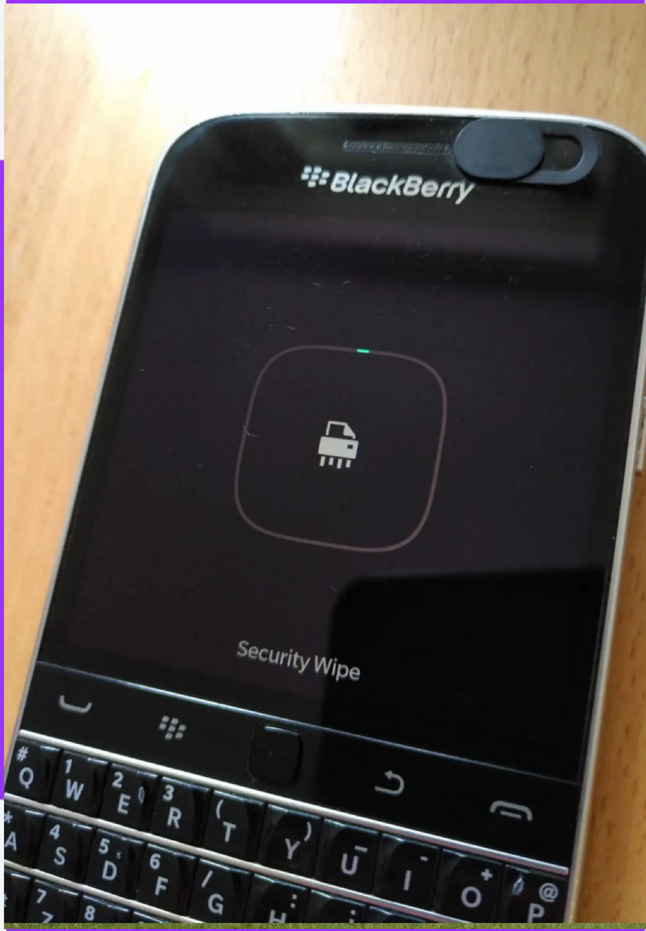
1. Introduction

Qui est ce rosbif ?

- **Ben Goldsworthy 🇬🇧**
- RunEvent/WeezCrew (*née* Recrewteer) in Montpellier since start of 2024
- Dev since ~2011
  - Started in school with VB.NET 🤮
  - Professionally since 2014
  - 2017 BSc (Hons) Computer Science
  - 2018 MSc Cyber Security

Qui est ce rosbif ?

- ➔ **Ben Goldsworthy 🇬🇧**
- ➔ Several years working on events
  - ■ Security
  - ■ *Secourisme*
- ➔ Luddite



weezcrew

2. Accessibility

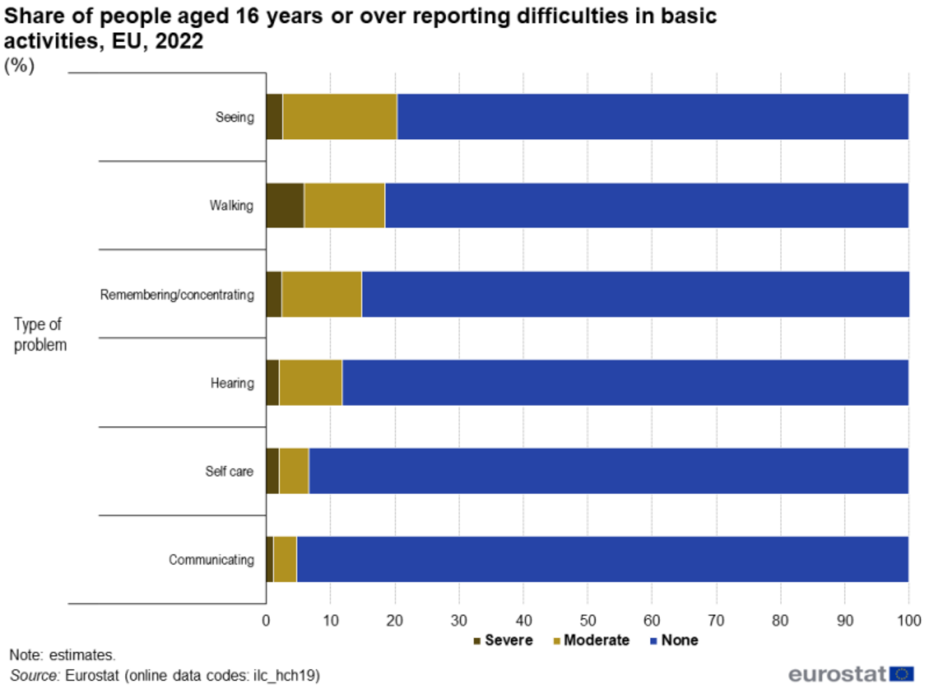
What is Accessibility (a11y)?

- ‘Accessibility is the practice of making your websites **usable by as many people as possible**.’ [1]
- ‘We traditionally think of this as being **about people with disabilities**…’ [*ibid.*]

[1] “What is accessibility?”, *MDN Web Docs* <https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Accessibility/What_is_accessibility>

Disability in the EU

- ➔ ‘In 2023, **27% of the EU population** over the age of 16 had some form of **disability**.’ [1]
  - ~101 million people
  - **1 in 4** adults
- ➔ Disability can be measured as activity *limitation*, or more broadly as *difficulties* in basic activities.
- ➔ ‘In 2022, **more than 1 in 3 (37.0%)** people aged 16 years or over in the EU reported **difficulties** in one or more basic activities.’ [2]



Share of people aged 16 years or over reporting difficulties in basic activities, EU, 2022
(%)

Note: estimates.
Source: Eurostat (online data codes: ilc_hch19)

[1] “Disability in the EU: Facts and Figures”, *European Council* <https://www.consilium.europa.eu/en/infographics/disability-eu-facts-figures>
[2] “Disability statistics - population with difficulties in basic activities”, *Eurostat* < https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Disability_statistics_-_population_with_difficulties_in_basic_activities>

Standards

- ➔ **Web Content/W3C Accessibility Guidelines (WCAG)** [1]
  - Current version: WCAG 2.2
    - WCAG 3.0 in draft stages
  - Defines four principles of accessibility:
    - Perceivable
    - Operable
    - Understandable
    - Robust
  - Defines several guidelines and success criteria for Level A, Level AA and Level AAA compliance
  - Adopted as ISO/IEC 40500:2012
  - Level A & Level AA incorporated as EN 301 549

- ➔ ***Référentiel général d'amélioration de l'accessibilité* (RGAA)** [2]
  - EN 301 549…
  - …with some additions

[1] Web Content Accessibility Guidelines (WCAG) 2.2, *W3C* <https://www.w3.org/TR/WCAG22>
[2] *Référentiel général d'amélioration de l'accessibilité*, *Gouvernement Français* <https://accessibilite.numerique.gouv.fr>

Legislation

- ➔ **France: *Loi № 2005-102*** [1]
  - Businesses earning >250m €/yr MUST comply with RGAA
  - Smaller businesses **SHOULD comply with RGAA**
- ➔ **EU: Web Accessibility Directive (WAD)** [2]
  - Came into effect June 2021
  - Requires public sector bodies **and their vendors** to make websites and mobile apps EN 301 549 compliant
- ➔ **EU: European Accessibility Act (EEA)** [3]
  - Enforcement begins **July 28 2025**
  - Requires EN 301 549 compliance for certain types of products and services
    - Products: Ticketing machines
    - Services: E-commerce

[1] *LOI n° 2005-102 du 11 février 2005 pour l'égalité des droits et des chances, la participation et la citoyenneté des personnes handicapées*, *Légifrance* < https://www.legifrance.gouv.fr/jorf/id/JORFTEXT000000809647>

[2] Directive (EU) 2016/2102 of the European Parliament and of the Council of 26 October 2016 on the accessibility of the websites and mobile applications of public sector bodies, *EUR-Lex* <https://eur-lex.europa.eu/eli/dir/2016/2102/oj>

[3] Directive (EU) 2019/882 of the European Parliament and of the Council of 17 April 2019 on the accessibility requirements for products and services, *EUR-Lex* < https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A32019L0882>

What is Accessibility (a11y)?

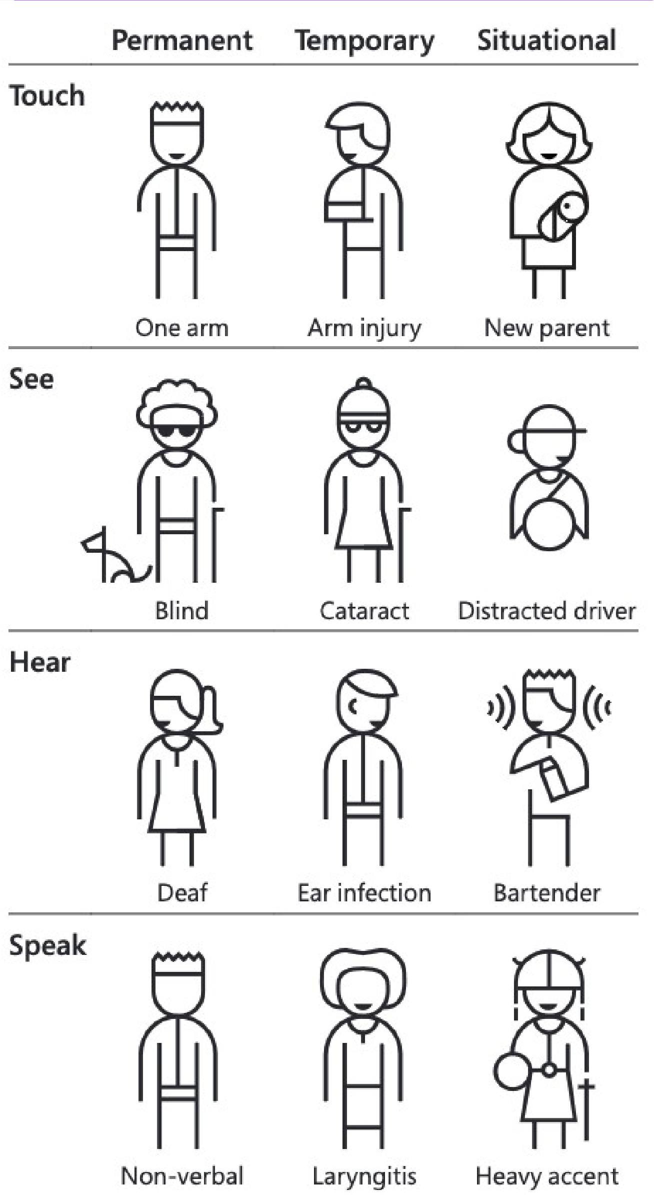
➔ ‘Accessibility is the practice of making your websites **usable by as many people as possible**.’[1]

➔ ‘We traditionally think of this as being about people with disabilities…
…but the practice of making sites accessible **also benefits other groups** such as those using mobile devices, or those with slow network connections.’[1]

[1] “What is accessibility?”, *MDN Web Docs* <https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Accessibility/What_is_accessibility>

Temporalities of Disability

- ➔ Permanent
- ➔ Temporary
- ➔ Situational



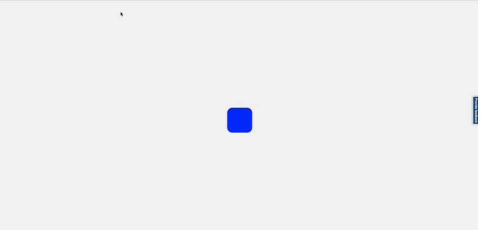
	Permanent	Temporary	Situational
Touch	One arm	Arm injury	New parent
See	Blind	Cataract	Distracted driver
Hear	Deaf	Ear infection	Bartender
Speak	Non-verbal	Laryngitis	Heavy accent

[1] “Inclusive Microsoft Design”, *Microsoft* < https://inclusive.microsoft.design/tools-and-activities/Inclusive101Guidebook.pdf>

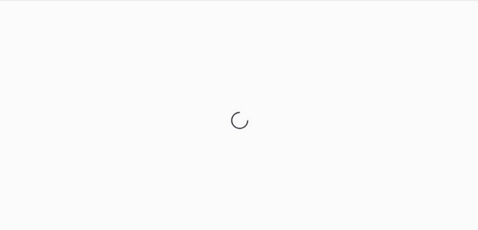
App Load Experience (Ideal)

- ➔ Connection: 5 GHz Wi-Fi (300 Mbps down, 30 Mbps up)
- ➔ Device: MacBook Pro 2024
  - ■ M3 Max CPU
  - ■ 36 GB memory

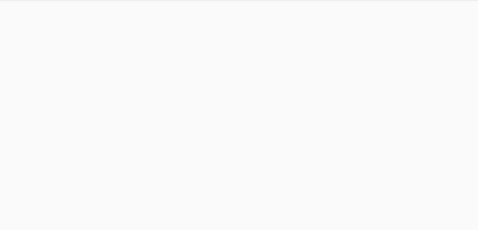
WeezTicket



Recrewteer



WeezAccess

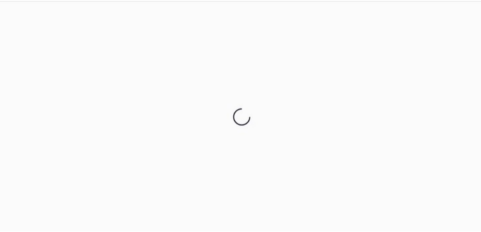


App Load Experience (Realistic)

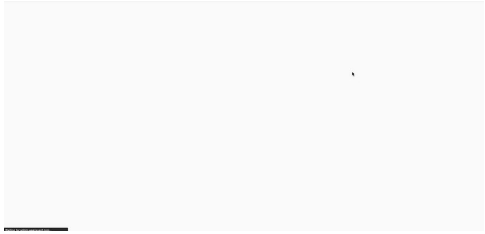
- ➔ Connection: Intermittent, congested festival Wi-Fi OR whatever mobile data speed you get stood in the middle of a field (Firefox ‘Regular 4G’ = 4 Mbps up, 3 Mbps down)
- ➔ Device: Some ancient tablet that the event organisers bought as part of a cheap bulk order

WeezTicket

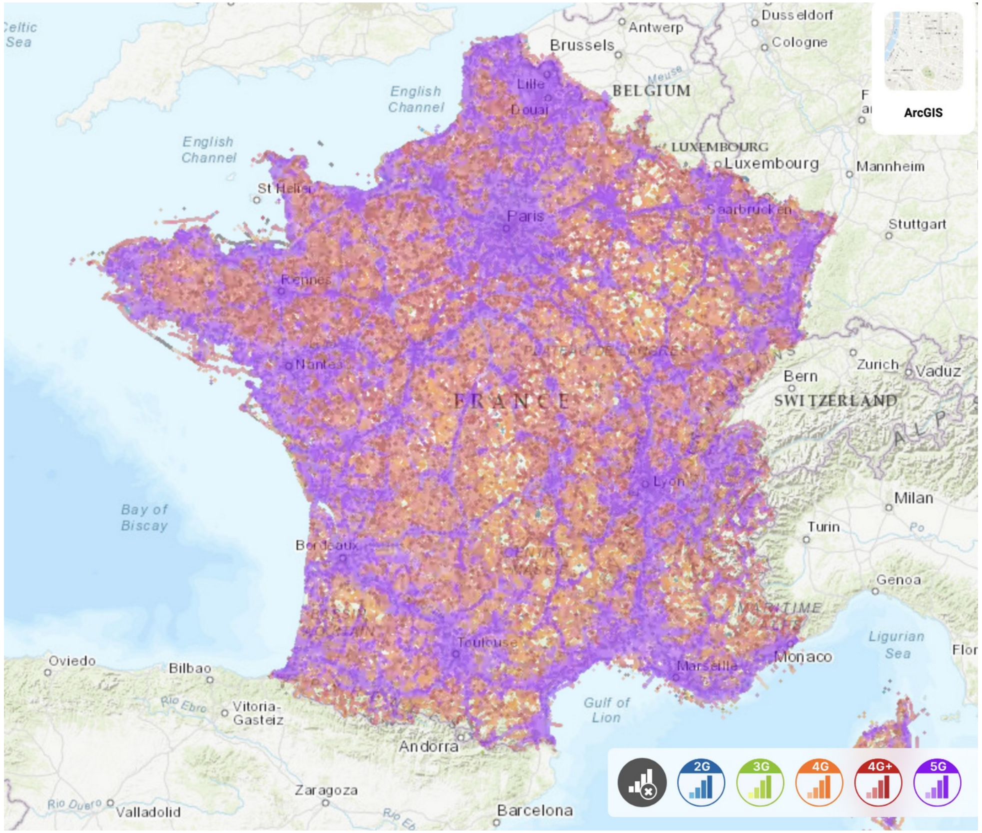
Recrewteer



WeezAccess

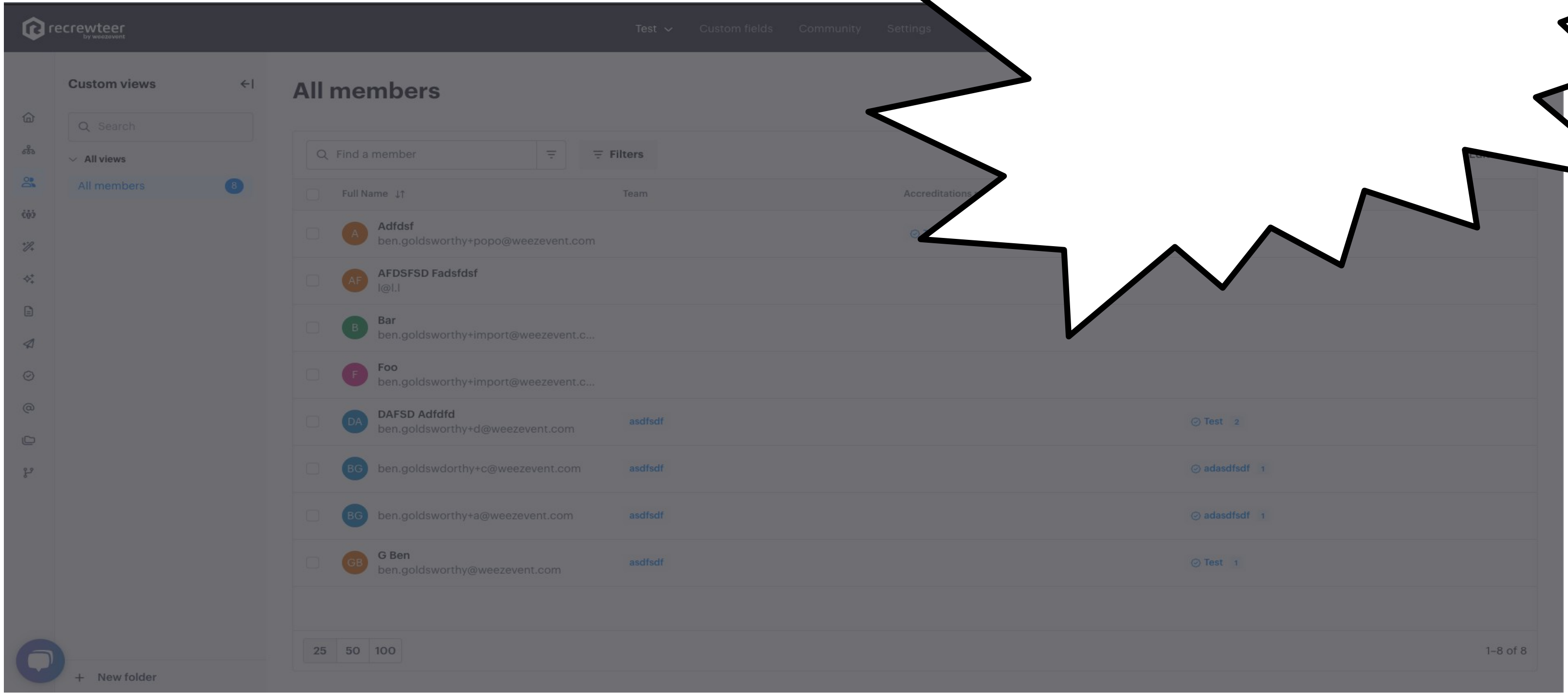


Mobile Data Coverage in France



[1] “3G / 4G / 5G coverage in France”, *nPerf.com* < https://www.nperf.com/en/map/FR/-/21.Orange-Mobile/signal?ll=46.36209301204985&lg=-7.932128906250001&zoom=6>

App Use Context



All members

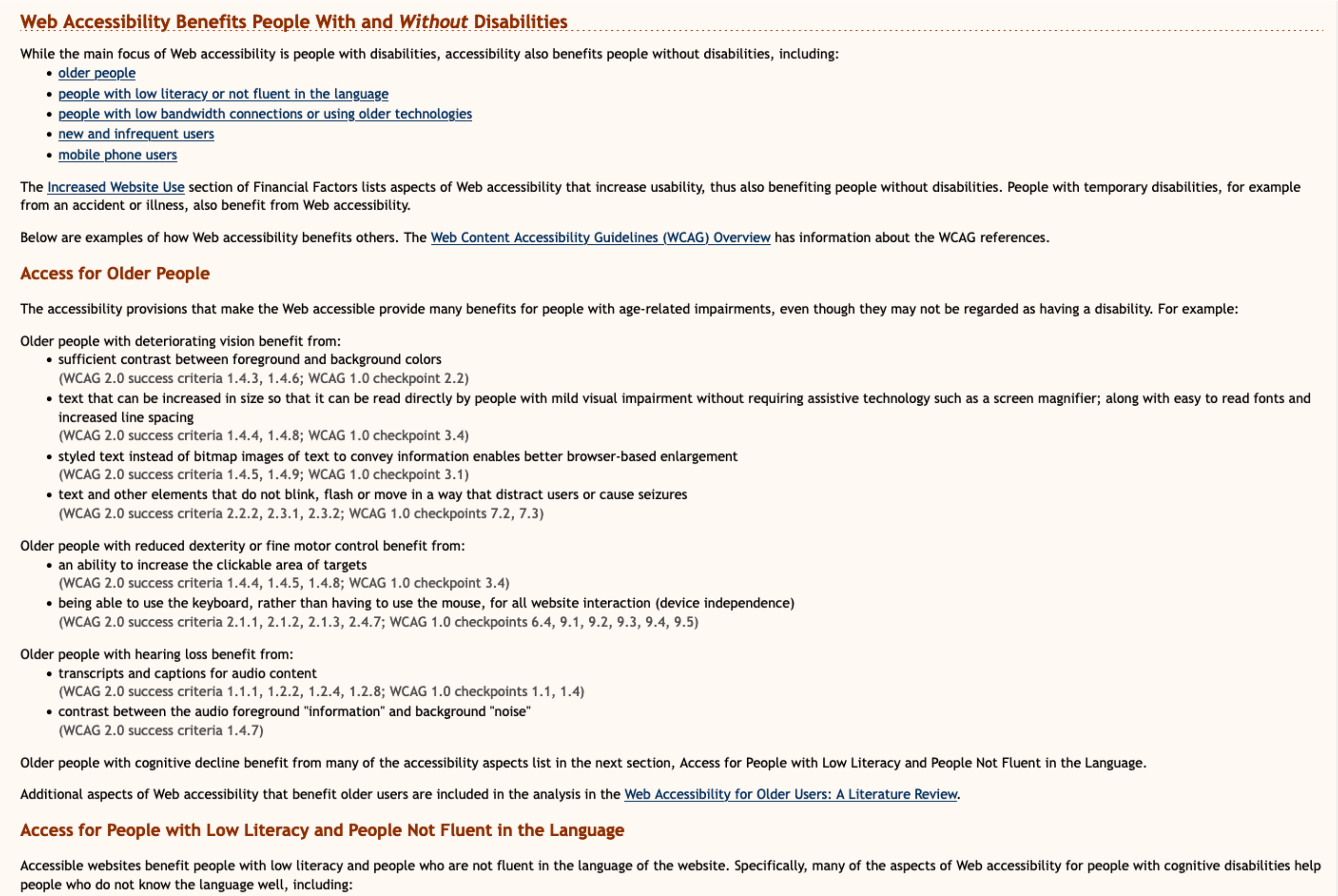
	Full Name	Team	Accreditations
□	A Adfdsf ben.goldsworthy+popo@weezevent.com		
□	AF AFDSFSD Fadsfdsf l@l.l		
□	B Bar ben.goldsworthy+import@weezevent.c...		
□	F Foo ben.goldsworthy+import@weezevent.c...		
□	DA DAFSD Adfdfd ben.goldsworthy+d@weezevent.com	asdfsdf	Test 2
□	BG ben.goldswdorthy+c@weezevent.com	asdfsdf	adasdfsdf 1
□	BG ben.goldsworthy+a@weezevent.com	asdfsdf	adasdfsdf 1
□	GB G Ben ben.goldsworthy@weezevent.com	asdfsdf	Test 1

25 50 100

1–8 of 8

[1] “The Switch”, Angerfist ft. Predator

Accessibility Business Case



Web Accessibility Benefits People With and *Without* Disabilities

While the main focus of Web accessibility is people with disabilities, accessibility also benefits people without disabilities, including:

- older people
- people with low literacy or not fluent in the language
- people with low bandwidth connections or using older technologies
- new and infrequent users
- mobile phone users

The Increased Website Use section of Financial Factors lists aspects of Web accessibility that increase usability, thus also benefiting people without disabilities. People with temporary disabilities, for example from an accident or illness, also benefit from Web accessibility.

Below are examples of how Web accessibility benefits others. The Web Content Accessibility Guidelines (WCAG) Overview has information about the WCAG references.

Access for Older People

The accessibility provisions that make the Web accessible provide many benefits for people with age-related impairments, even though they may not be regarded as having a disability. For example:

Older people with deteriorating vision benefit from:

- sufficient contrast between foreground and background colors
  (WCAG 2.0 success criteria 1.4.3, 1.4.6; WCAG 1.0 checkpoint 2.2)
- text that can be increased in size so that it can be read directly by people with mild visual impairment without requiring assistive technology such as a screen magnifier; along with easy to read fonts and increased line spacing
  (WCAG 2.0 success criteria 1.4.4, 1.4.8; WCAG 1.0 checkpoint 3.4)
- styled text instead of bitmap images of text to convey information enables better browser-based enlargement
  (WCAG 2.0 success criteria 1.4.5, 1.4.9; WCAG 1.0 checkpoint 3.1)
- text and other elements that do not blink, flash or move in a way that distract users or cause seizures
  (WCAG 2.0 success criteria 2.2.2, 2.3.1, 2.3.2; WCAG 1.0 checkpoints 7.2, 7.3)

Older people with reduced dexterity or fine motor control benefit from:

- an ability to increase the clickable area of targets
  (WCAG 2.0 success criteria 1.4.4, 1.4.5, 1.4.8; WCAG 1.0 checkpoint 3.4)
- being able to use the keyboard, rather than having to use the mouse, for all website interaction (device independence)
  (WCAG 2.0 success criteria 2.1.1, 2.1.2, 2.1.3, 2.4.7; WCAG 1.0 checkpoints 6.4, 9.1, 9.2, 9.3, 9.4, 9.5)

Older people with hearing loss benefit from:

- transcripts and captions for audio content
  (WCAG 2.0 success criteria 1.1.1, 1.2.2, 1.2.4, 1.2.8; WCAG 1.0 checkpoints 1.1, 1.4)
- contrast between the audio foreground "information" and background "noise"
  (WCAG 2.0 success criteria 1.4.7)

Older people with cognitive decline benefit from many of the accessibility aspects list in the next section, Access for People with Low Literacy and People Not Fluent in the Language.

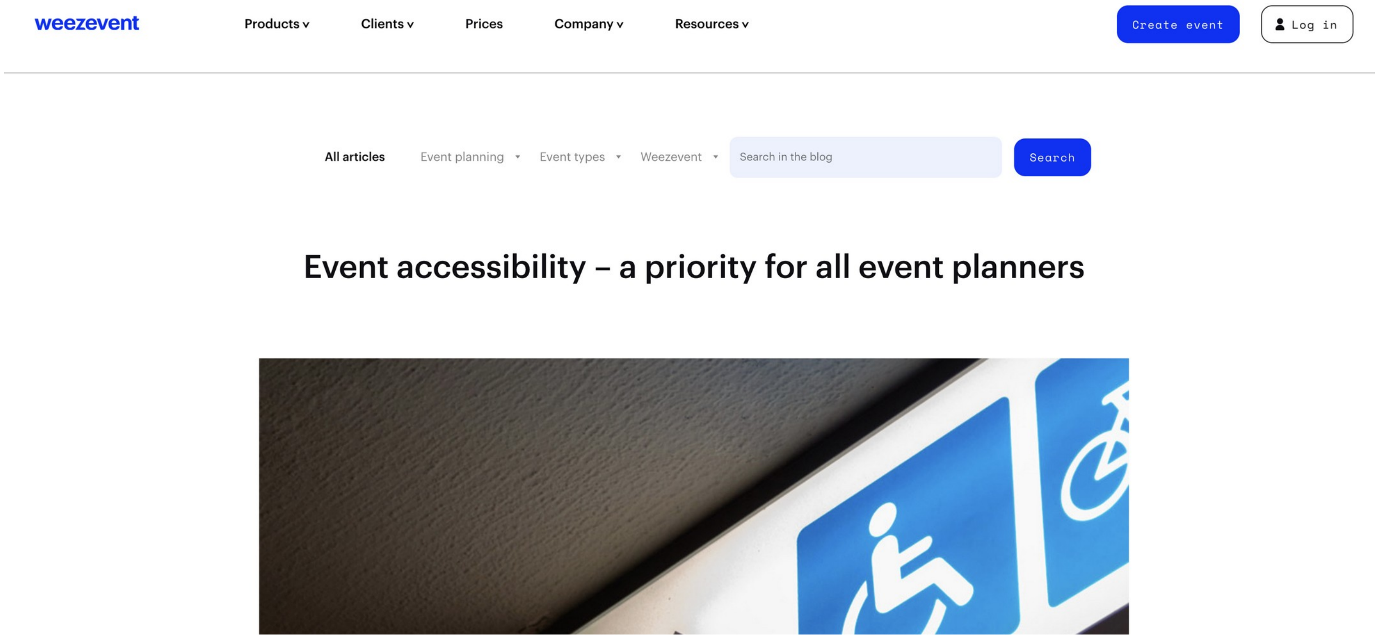
Additional aspects of Web accessibility that benefit older users are included in the analysis in the Web Accessibility for Older Users: A Literature Review.

Access for People with Low Literacy and People Not Fluent in the Language

Accessible websites benefit people with low literacy and people who are not fluent in the language of the website. Specifically, many of the aspects of Web accessibility for people with cognitive disabilities help people who do not know the language well, including:

[1] "Social Factors in Developing a Web Accessibility Business Case for Your Organization", *W3C WAI* <https://www.w3.org/WAI/business-case/archive/soc#groups>

Weez



Event accessibility – a priority for all event planners

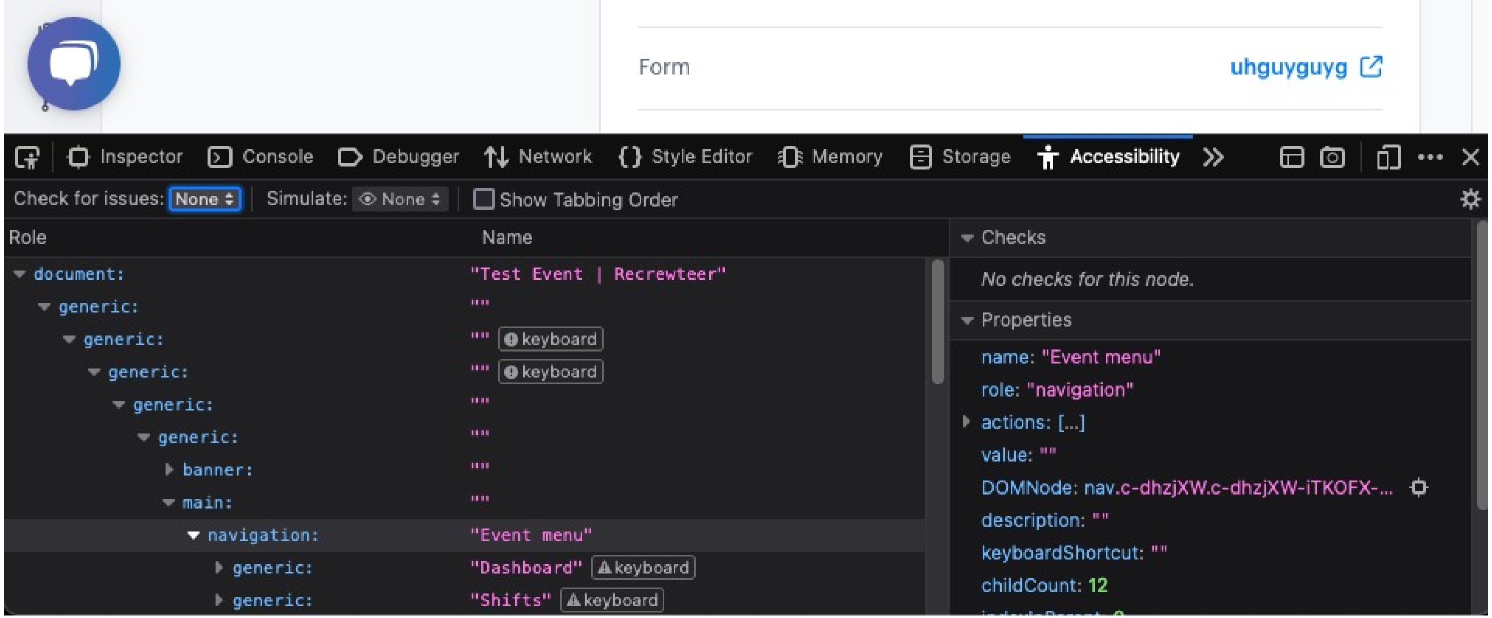
[1] “Event accessibility – a priority for all event planners”, *Weezevent* <https://weezevent.com/en-gb/blog/event-accessibility/>
[2] “*Organiser un événement inclusif et accessible : le guide complet*”, *Weezevent* <https://weezevent.com/fr/blog/organiser-evenement-inclusif-accessible/>

weezcrew

3. Quick Wins

Tools

- ➔ “Web Developer” browser extension [1]
- ➔ Built-in browser Accessibility tab:



Check for issues: None   Simulate: None   □ Show Tabbing Order

```
Role                                   Name
▾ document:                            "Test Event | Recrewteer"
  ▾ generic:                           ""
    ▾ generic:                         "" keyboard
      ▾ generic:                       "" keyboard
        ▾ generic:                     ""
          ▾ generic:                   ""
            ▸ banner:                  ""
            ▾ main:                    ""
              ▾ navigation:            "Event menu"
                ▸ generic:             "Dashboard" keyboard
                ▸ generic:             "Shifts" keyboard
```

▾ Checks

No checks for this node.

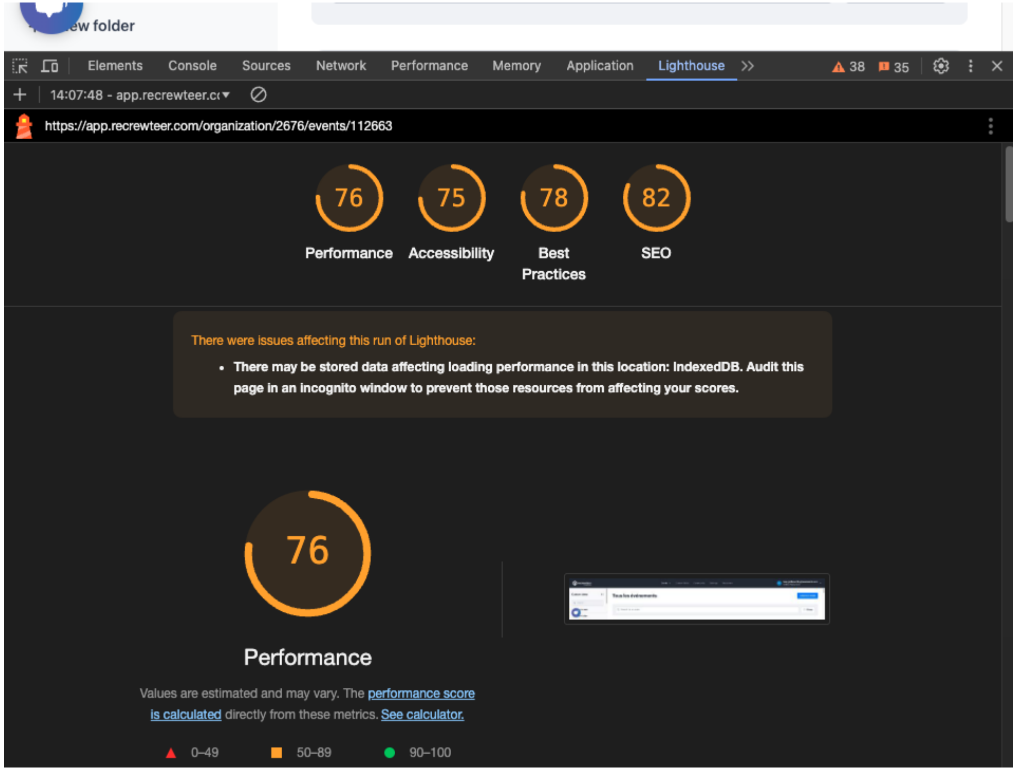
▾ Properties

```
name: "Event menu"
role: "navigation"
▸ actions: [...]
value: ""
DOMNode: nav.c-dhzjXW.c-dhzjXW-iTKOFX-...
description: ""
keyboardShortcut: ""
childCount: 12
```

[1] “Web Developer”, *ChrisPederick.com* <https://chrispederick.com/work/web-developer/>

Tools

- ➔ Lighthouse tab (Chromium-based browsers):



https://app.recrewteer.com/organization/2676/events/112663

76	75	78	82
Performance	Accessibility	Best Practices	SEO

There were issues affecting this run of Lighthouse:

- There may be stored data affecting loading performance in this location: IndexedDB. Audit this page in an incognito window to prevent those resources from affecting your scores.

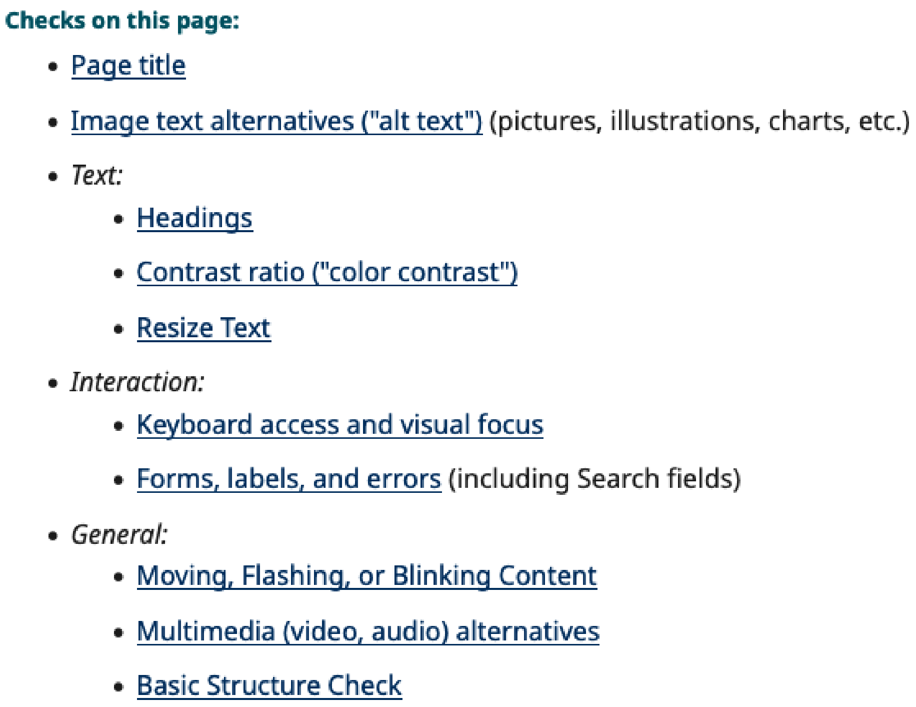
76

Performance

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

▲ 0–49 ■ 50–89 ● 90–100

WAI Easy Checks

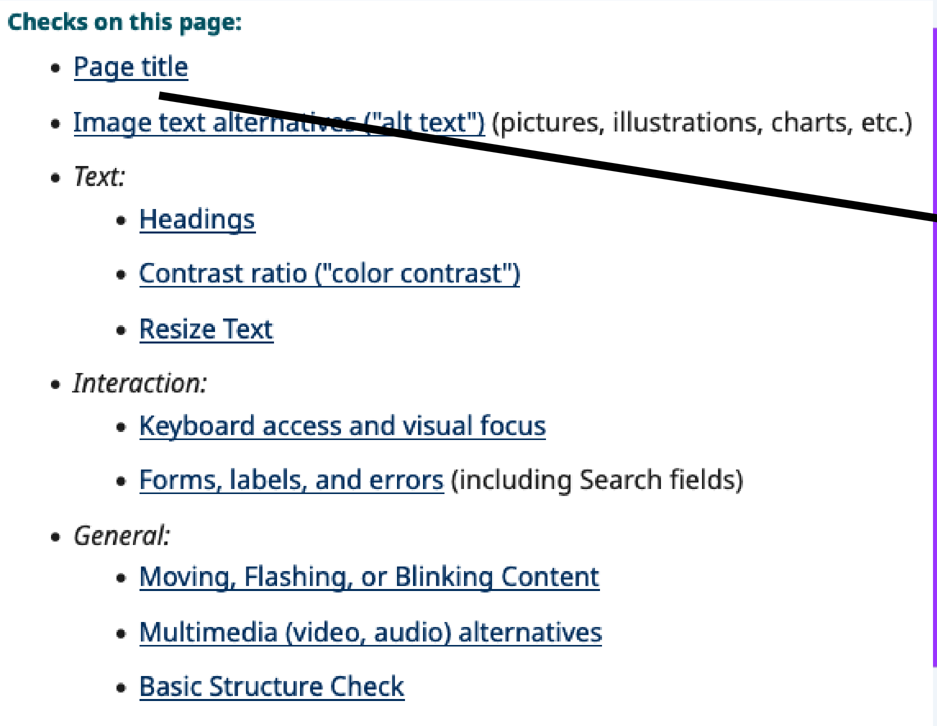


Checks on this page:

- Page title
- Image text alternatives ("alt text") (pictures, illustrations, charts, etc.)
- *Text:*
  - Headings
  - Contrast ratio ("color contrast")
  - Resize Text
- *Interaction:*
  - Keyboard access and visual focus
  - Forms, labels, and errors (including Search fields)
- *General:*
  - Moving, Flashing, or Blinking Content
  - Multimedia (video, audio) alternatives
  - Basic Structure Check

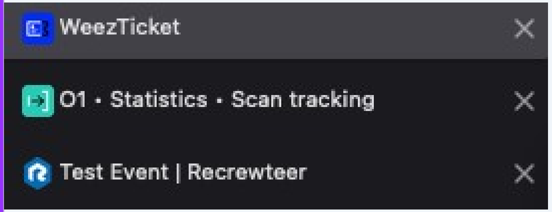
[1] “Easy Checks – A First Review of Web Accessibility”, *W3C WAI* <https://www.w3.org/WAI/test-evaluate/preliminary/>

Easy Checks – Page Title



Checks on this page:

- Page title
- Image text alternatives ("alt text") (pictures, illustrations, charts, etc.)
- *Text:*
  - Headings
  - Contrast ratio ("color contrast")
  - Resize Text
- *Interaction:*
  - Keyboard access and visual focus
  - Forms, labels, and errors (including Search fields)
- *General:*
  - Moving, Flashing, or Blinking Content
  - Multimedia (video, audio) alternatives
  - Basic Structure Check

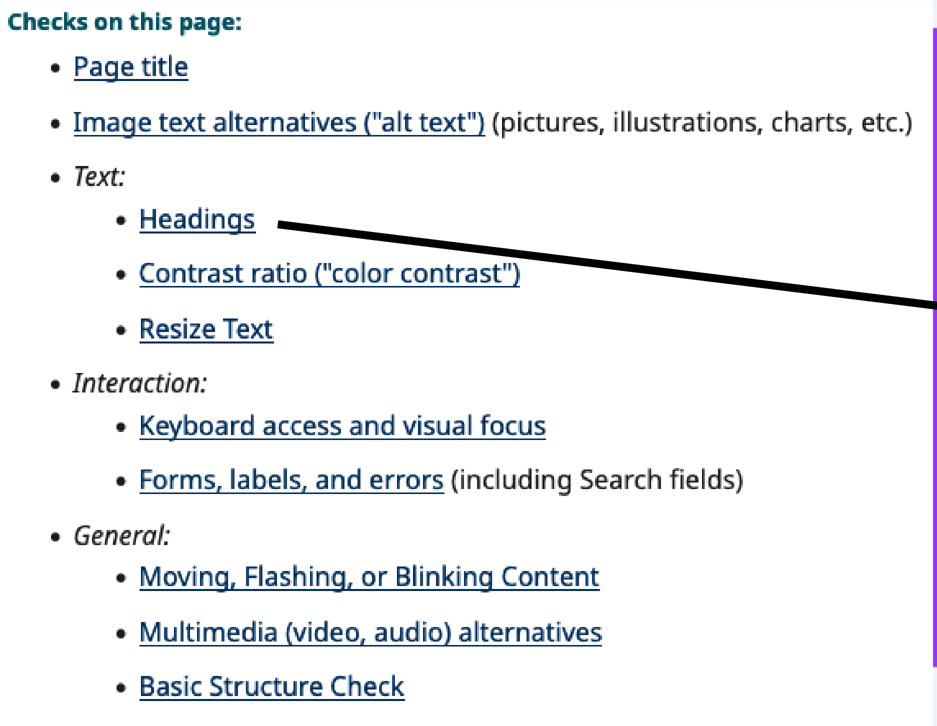


WeezTicket

O1 · Statistics · Scan tracking

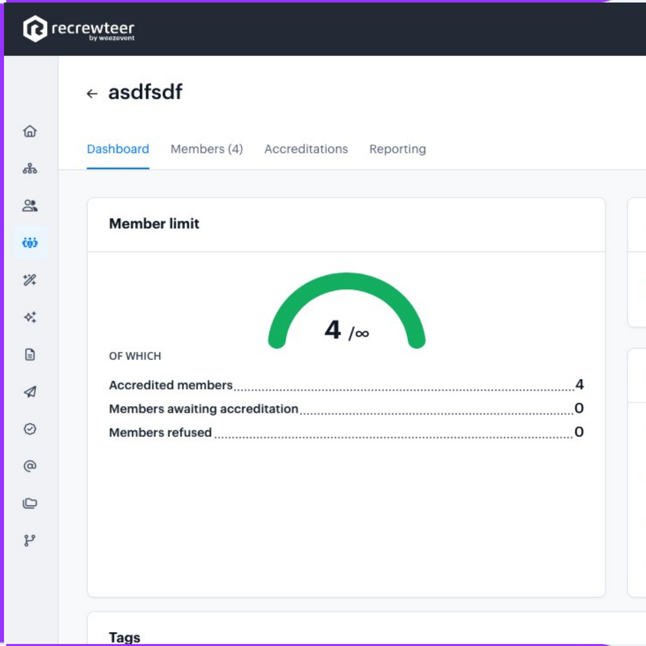
Test Event | Recrewteer

Easy Checks – Headings



Checks on this page:

- Page title
- Image text alternatives ("alt text") (pictures, illustrations, charts, etc.)
- *Text:*
  - Headings
  - Contrast ratio ("color contrast")
  - Resize Text
- *Interaction:*
  - Keyboard access and visual focus
  - Forms, labels, and errors (including Search fields)
- *General:*
  - Moving, Flashing, or Blinking Content
  - Multimedia (video, audio) alternatives
  - Basic Structure Check



← asdfsdf

Member limit

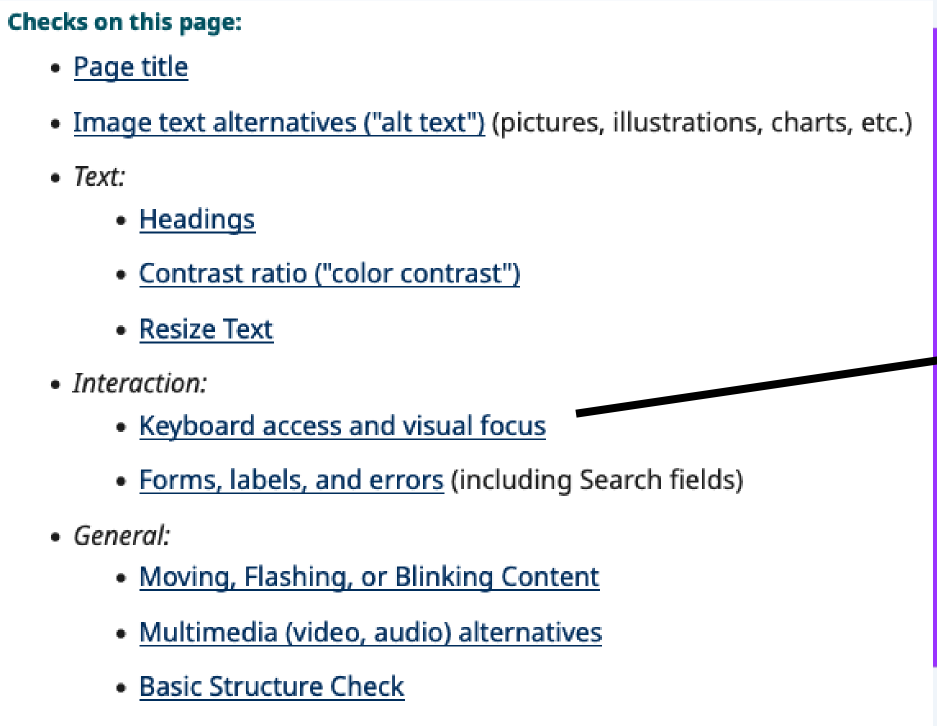
4 /∞

OF WHICH

Accredited members	4
Members awaiting accreditation	0
Members refused	0

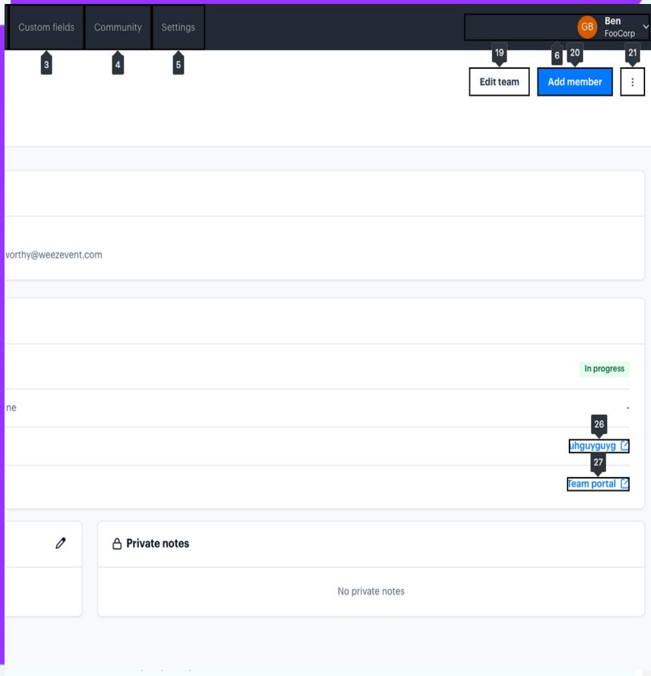
Tags

Easy Checks – Keyboard

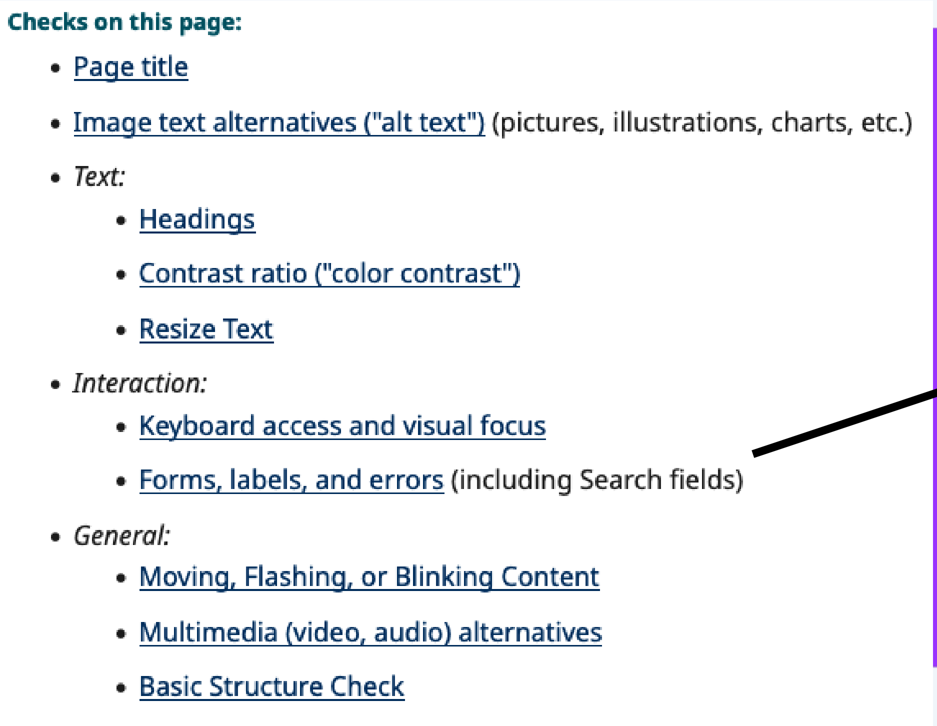


Checks on this page:

- Page title
- Image text alternatives ("alt text") (pictures, illustrations, charts, etc.)
- *Text:*
  - Headings
  - Contrast ratio ("color contrast")
  - Resize Text
- *Interaction:*
  - Keyboard access and visual focus
  - Forms, labels, and errors (including Search fields)
- *General:*
  - Moving, Flashing, or Blinking Content
  - Multimedia (video, audio) alternatives
  - Basic Structure Check

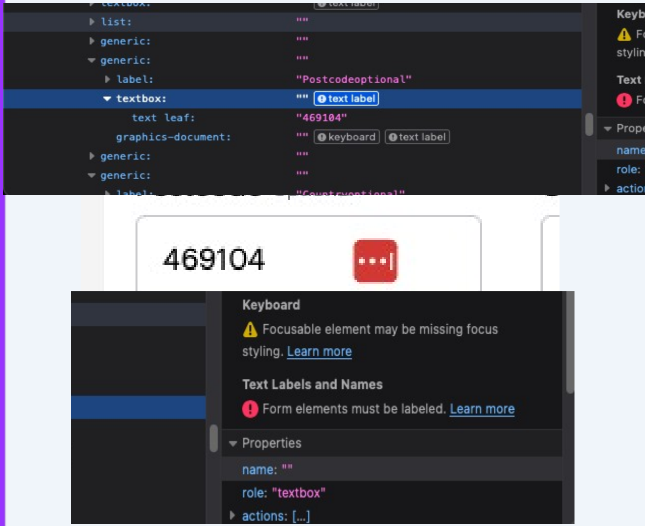


Easy Checks – Forms



Checks on this page:

- Page title
- Image text alternatives ("alt text") (pictures, illustrations, charts, etc.)
- *Text:*
  - Headings
  - Contrast ratio ("color contrast")
  - Resize Text
- *Interaction:*
  - Keyboard access and visual focus
  - Forms, labels, and errors (including Search fields)
- *General:*
  - Moving, Flashing, or Blinking Content
  - Multimedia (video, audio) alternatives
  - Basic Structure Check



```
▸ list:                  ""
▸ generic:               ""
▾ generic:               ""
  ▸ label:               "Postcodeoptional"
  ▾ textbox:             "" text label
      text leaf:         "469104"
    graphics-document:   "" keyboard  text label
▸ generic:               ""
▾ generic:               ""
  ▸ label:               "Countryoptional"
```

469104

Keyboard

⚠ Focusable element may be missing focus styling. Learn more

Text Labels and Names

❗ Form elements must be labeled. Learn more

▾ Properties

name: ""

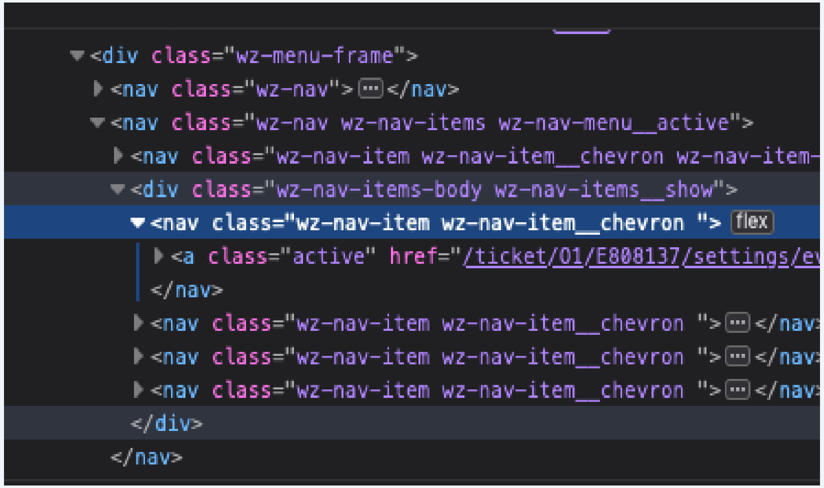
role: "textbox"

▸ actions: [...]

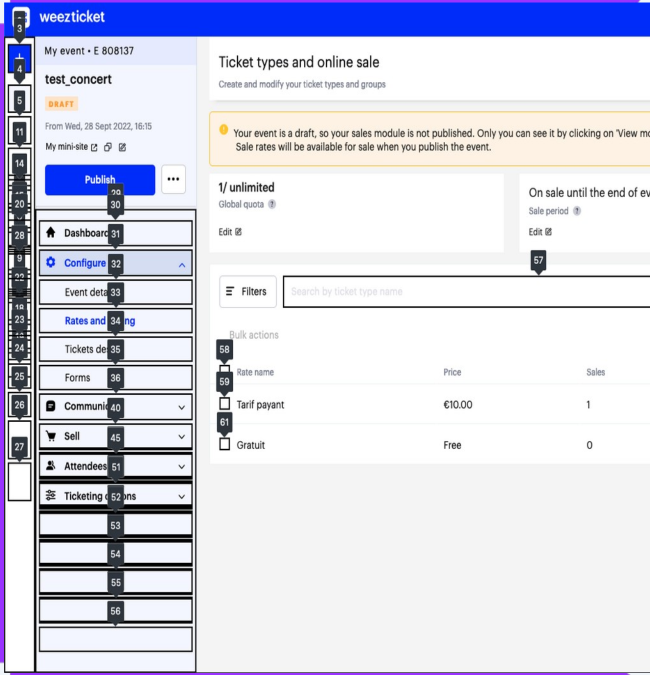
weezcrew

4. Semantic HTML

Did Someone Implement the Keyboard Nav?



```
▼<div class="wz-menu-frame">
  ▶<nav class="wz-nav">⋯</nav>
  ▼<nav class="wz-nav wz-nav-items wz-nav-menu__active">
    ▶<nav class="wz-nav-item wz-nav-item__chevron wz-nav-item-
    ▼<div class="wz-nav-items-body wz-nav-items__show">
      ▼<nav class="wz-nav-item wz-nav-item__chevron "> flex
        ▶<a class="active" href="/ticket/01/E808137/settings/ev
        </nav>
      ▶<nav class="wz-nav-item wz-nav-item__chevron ">⋯</nav
      ▶<nav class="wz-nav-item wz-nav-item__chevron ">⋯</nav
      ▶<nav class="wz-nav-item wz-nav-item__chevron ">⋯</nav
      </div>
    </nav>
```



My event • E 808137

test_concert

DRAFT

From Wed, 28 Sept 2022, 16:15

My mini-site

Ticket types and online sale

Create and modify your ticket types and groups

Your event is a draft, so your sales module is not published. Only you can see it by clicking on 'View mo
Sale rates will be available for sale when you publish the event.

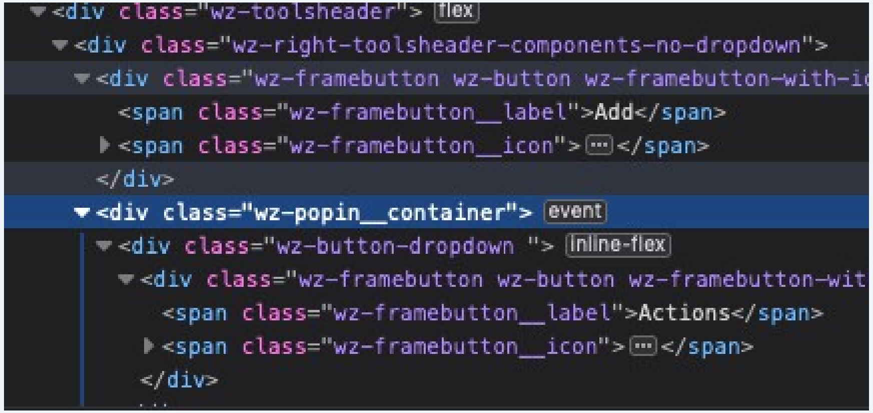
1/ unlimited
Global quota
Edit

On sale until the end of ev
Sale period
Edit

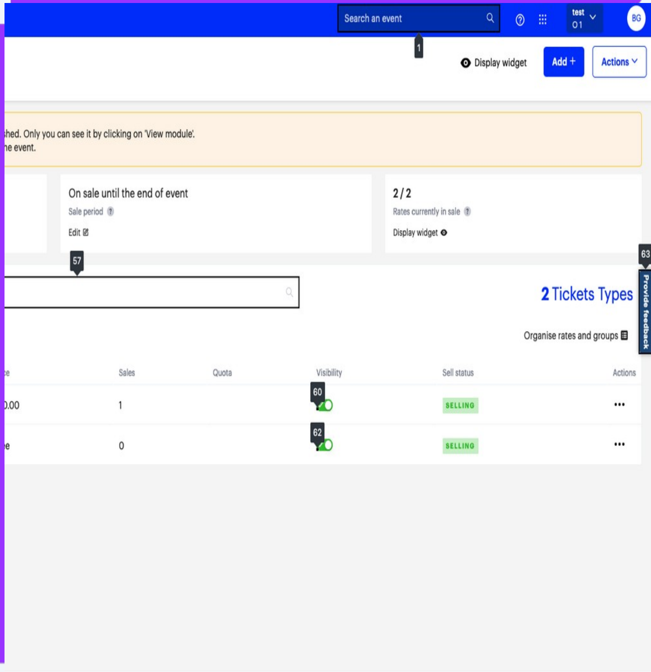
	Rate name	Price	Sales
□	Tarif payant	€10.00	1
□	Gratuit	Free	0

weezcrew

Why Are Things Missing?



```
▼<div class="wz-toolsheader"> flex
  ▼<div class="wz-right-toolsheader-components-no-dropdown">
    ▼<div class="wz-framebutton wz-button wz-framebutton-with-ic
       <span class="wz-framebutton__label">Add</span>
      ▶<span class="wz-framebutton__icon">⋯</span>
     </div>
    ▼<div class="wz-popin__container"> event
      ▼<div class="wz-button-dropdown "> inline-flex
        ▼<div class="wz-framebutton wz-button wz-framebutton-wit
           <span class="wz-framebutton__label">Actions</span>
          ▶<span class="wz-framebutton__icon">⋯</span>
         </div>
         ...
```



shed. Only you can see it by clicking on 'View module'.
he event.

On sale until the end of event
Sale period
Edit

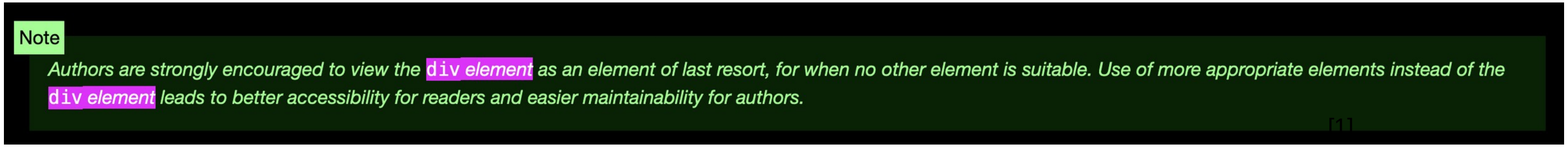
2 / 2
Rates currently in sale
Display widget

2 Tickets Types

Organise rates and groups

e	Sales	Quota	Visibility	Sell status	Actions
0.00	1			SELLING	•••
e	0			SELLING	•••

HTML5 Semantic Elements

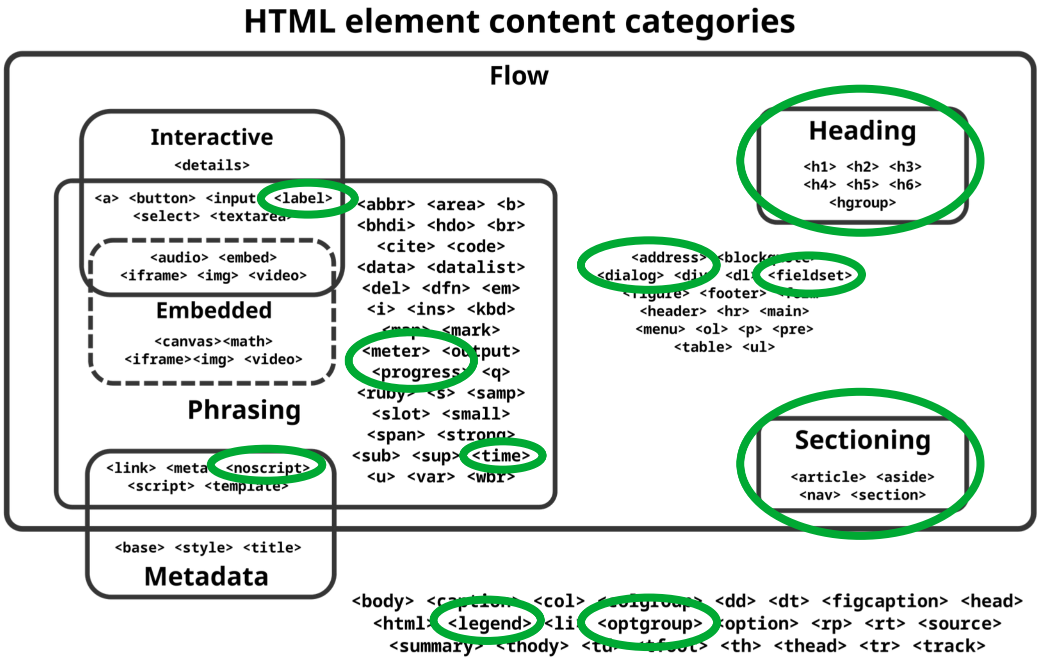


Note

Authors are strongly encouraged to view the `div` *element as an element of last resort, for when no other element is suitable. Use of more appropriate elements instead of the* `div` *element leads to better accessibility for readers and easier maintainability for authors.*

[1]

[1]



HTML element content categories

Flow

Interactive

<details>

<a> <button> <input> <label>
<select> <textarea>

<audio> <embed>
<iframe> <img> <video>

Embedded

<canvas><math>
<iframe><img> <video>

Phrasing

<abbr> <area> <b>
<bhdi> <hdo>
<cite> <code>
<data> <datalist>
<del> <dfn> <em>
<i> <ins> <kbd>
<map> <mark>
<meter> <output>
<progress> <q>
<ruby> <s> <samp>
<slot> <small>
<span> <strong>
<sub> <sup> <time>
<u> <var> <wbr>

<link> <meta> <noscript>
<script> <template>

<base> <style> <title>

Metadata

Heading

<h1> <h2> <h3>
<h4> <h5> <h6>
<hgroup>

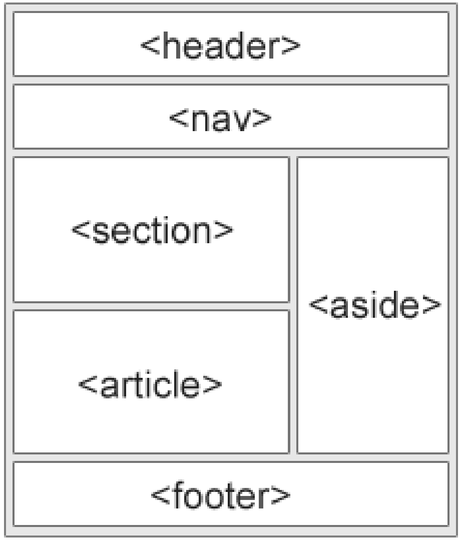
<address> <blockquote>
<dialog> <div> <dl> <fieldset>
<figure> <footer> <form>
<header> <hr> <main>
<menu> <ol> <p> <pre>
<table> <ul>

Sectioning

<article> <aside>
<nav> <section>

<body> <caption> <col> <colgroup> <dd> <dt> <figcaption> <head>
<html> <legend> <li> <optgroup> <option> <rp> <rt> <source>
<summary> <tbody> <td> <tfoot> <th> <thead> <tr> <track>

[2]

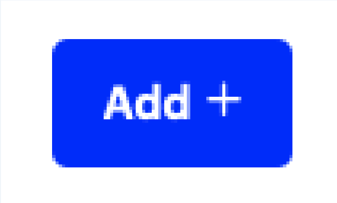


[3]

[1] “HTML Standard”, WHATWG <https://html.spec.whatwg.org/#the-div-element>
[2] “HTML element content categories”, Максим Пе on Wikimedia Commons <https://commons.wikimedia.org/wiki/File:HTML_element_content_categories.svg>
[3] “HTML Semantic Elements”, *W3Schools* <https://www.w3schools.com/html/html5_semantic_elements.asp>

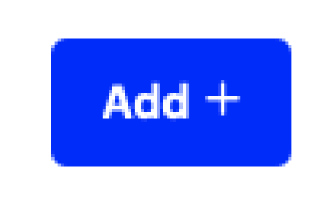
Is This a Button?

- ➔ Visually, yes
- ➔ Functionally, maybe?



Should This Be a `<button>`?

- ➔ Functionally, maybe?



When is a Button Not a `<button>`?

1. When it's a `<div>` with an `onClick()` handler
   - No built-in accessibility properties
   - No built-in keyboard handling

2. When it's an `<input type="submit">` within a `<form>`

3. When it's a `<div>` with `role="button"`

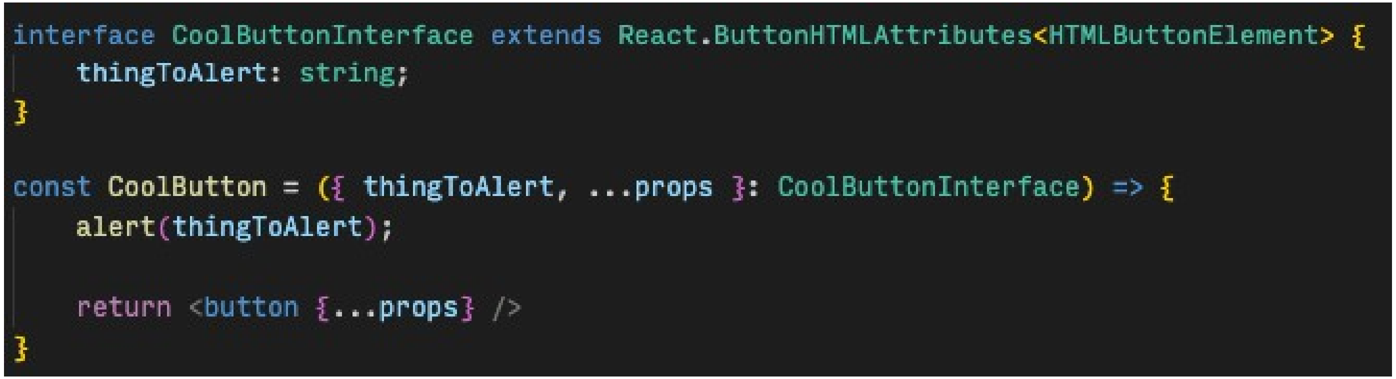
   - First rule of ARIA: 'If you can use a native HTML element or attribute with the semantics and behavior you require **already built in**, instead of re-purposing an element and adding an ARIA role, state or property to make it accessible, **then do so.**' [1]

4. When it's a link to somewhere else
   - Using a `<button>` or `<div onClick={…}>` in place of an `<a href={…}>` is a common anti-pattern [2]
   - No built-in browser history integration
   - No built-in 'open in new tab' etc.

[1] "Using ARIA", *W3C* <https://www.w3.org/TR/using-aria/#rule1>
[2] "Links and Buttons Guide", *CSS-Tricks* <https://css-tricks.com/a-complete-guide-to-links-and-buttons/>

When is a Button BETTER Than a `<button>`?

1. When it’s component that extends the `HTMLButtonElement` interface [1]
   - `React.ButtonHTMLAttributes<HTMLButtonElement>` in React [2]
   - Destructure your custom props, pass everything else to a `<button>` element
   - Or pass a `<button>` as the as prop [3]



```
interface CoolButtonInterface extends React.ButtonHTMLAttributes<HTMLButtonElement> {
    thingToAlert: string;
}

const CoolButton = ({ thingToAlert, ...props }: CoolButtonInterface) => {
    alert(thingToAlert);

    return <button {...props} />
}
```

[1] “HTMLButtonElement”, *MDN Web Docs* <https://developer.mozilla.org/en-US/docs/Web/API/HTMLButtonElement>
[2] “Extend an HTML Element in a Component’s props in React (TS)”, *BobbyHadz.com* <https://bobbyhadz.com/blog/react-typescript-extend-html-element-props>
[3] “The "as" prop in ReactJS and dynamic component names”, *Codem\y’s Blog* <https://www.codemzy.com/blog/reactjs-as-prop>

Be Lazy

1. Web browser features:
   - 300+ W3C standards [1]
   - 1000+ including drafts and candidates [2]
   - 20+ WHATWG standards [3]

2. Decades of engineering
   - Firefox: 21m+ LOC [4]
   - Chrome: 6m+ LOC [5]

3. HTML will always be faster than CSS will always be faster than JS

[1] “W3C Standards and Drafts”, *W3C* <https://www.w3.org/TR/?status%5B0%5D=standard>
[2] *ibid.* <
https://www.w3.org/TR/?status%5B0%5D=draftStandard&status%5B1%5D=candidateStandard&status%5B2%5D=note&status%5B3%5D=dry&status%5B4%5D=cry>
[3] “Standards”, *WHATWG* <https://spec.whatwg.org/>
[4] “Engineering code quality in the Firefox browser: A look at our tools and challenges”, *Mozilla Hacks* <https://hacks.mozilla.org/2020/04/code-quality-tools-at-mozilla/>
[5] “Infographic: How Many Millions of Lines of Code Does it Take?”, *Visual Capitalist* <https://www.visualcapitalist.com/millions-lines-of-code/>

Some Final Cool Things

1. CSS features:
   - `prefers-reduced-motion` media queries [1.1]
   - `prefers-reduced-data` (much less support) [1.2
   - `media="print"` for print styling [2]
   - `::backdrop` (with `<dialog>` element) [3]

2. Microdata markup [4]

3. HTML element attributes:
   - `<a> download` [5]
   - `<input> autocomplete` [6]

4. `.well-known` URIs [7]
   - particularly `change-password` and `security.txt`
   - possibly also `caldav/carddav`

[1.1] “prefers-reduced-motion”, *CSS-Tricks* <https://css-tricks.com/almanac/rules/m/media/prefers-reduced-motion/>
[1.2] “prefers-reduced-data”, *MDN Web Docs* <https://developer.mozilla.org/en-US/docs/Web/CSS/@media/prefers-reduced-data>
[2] “Printing”, *MDN Web Docs* <https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_media_queries/Printing>
[3] “::backdrop”, *MDN Web Docs* <https://developer.mozilla.org/en-US/docs/Web/CSS/::backdrop>
[4] “Getting started with schema.org using Microdata”, *Schema.org* <https://schema.org/docs/gs.html>
[5] “<a>: The Anchor Element”, *MDN Web Docs* <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/a#download>
[6] “HTML attribute: autocomplete”, *MDN Web Docs* <https://developer.mozilla.org/en-US/docs/Web/HTML/Attributes/autocomplete>
[7] “Well-known URI”, *Wikipedia* <https://en.wikipedia.org/wiki/Well-known_URI>

Some Final Cool Things

5. VanillaJS [1]

6. You Might Not Need JavaScript/Lodash [2]

7. Islands architecture [3]
   - Not a new idea! [4]

8. htmx [5]

9. W3C Web Accessibility Initiative (WAI) [6]

10. WebAIM [7]

[1] *VanillaJS* <http://vanilla-js.com/>
[2] *You Might Not Need JavaScript* <http://youmightnotneedjs.com/> and *You Might Not Need Lodash* <https://youmightnotneed.com/lodash/>
[3] “Islands Architecture”, *Patterns.dev* <https://www.patterns.dev/vanilla/islands-architecture>
[4] “JSR 168: Portlet Specification”, *Java Community Process* <https://jcp.org/en/jsr/detail?id=168>. See also Flash, ASP, Shockwave, etc.
[5] *htmx* <https://htmx.org/>
[6] *W3C WAI* <https://www.w3.org/WAI/>
[7] *Web Accessibility In Mind* <https://webaim.org/>

Ben GOLDSWORTHY

Software Developer

–

ben.goldsworthy@weeezevent.com