



HOW-TO GUIDE

Web Accessibility: What It Is, Why You Need It, How to Get It

MIGHTY  CITIZEN

FORMERLY TRADEMARK MEDIA

Table of Contents

| | |
|--|-----------|
| What is Web Accessibility? | 3 |
| Accessibility Laws and Rules | 4 |
| 4 Principles of Web Accessibility | 5 |
| Basic Accessibility Techniques | 6 |
| Accessibility Resources | 12 |

What is Web Accessibility?

Welcome to the world of web accessibility—where the decisions you make about your organization’s digital identity can impact more than 50 million Americans.

Web accessibility refers to the design of websites, apps, and other digital content so that it’s accessible to every user, *regardless of their abilities or disabilities.*

For example: Accessible websites are developed, designed, and managed to ensure that users with a disability—blindness, motor impairment, deafness, etc.—can use them just as effectively as users without a disability.

The Scope of Accessibility

Accessibility cannot be an afterthought when constructing your website. Instead, it should be considered just as carefully as other website decisions—e.g., which pages to build, what interactivity to add, etc. To let your website’s accessibility fall into disrepair and irrelevance is to purposefully ignore the needs of countless users.

56.7 million Americans have some form of disability (U.S. Census)

This includes:

- 19.2 million with difficulty lifting or grasping—impacting their use of a mouse and keyboard
- 15.2 million with cognitive, mental, or emotional impairments
- 8.1 million with vision impairment—who often use a “screen reader” to use the web
- 7.6 million with hearing impairment

Each of these Americans may need adaptive technology to consume Internet content—including screen readers, screen magnifiers, adapted keyboards, etc. And the way you design and build your website can have serious consequences for each of these groups.

Accessibility Laws and Rules

Section 508

When anyone discusses accessible digital design, they inevitably mention Section 508. This is the section of the Rehabilitation Act of 1973 that applies to all Federal information technology—requiring, in part, that everything the government puts online must be accessible to people with disabilities.

Over time, Section 508 guidelines have been adopted by other industries and sectors, including many state governments.

2018 Updates to Section 508

In January 2018, a new set of accessibility rules were enacted as part of an update to Section 508—and governments across the country are having to adapt to them. Instead of drafting its own accessibility policies and guidelines, the federal government simply pointed to the guidelines defined by WCAG AA.

WCAG

WCAG—which stands for Web Content Accessibility Guidelines—defines (in extreme detail) how to make a website more accessible to people with disabilities. It’s the universally accepted standard for web accessible design.

WCAG offers three levels of accessibility—each stricter than the last:

- Level A - the least accessible level, not often used by organizations
- Level AA - the middle-of-the-road option, highly accessible and widely adopted as the standard
- Level AAA - fully accessible, but owing to how it impacts site design and features, often considered too strict by most organizations

Breaking the Rules

In 2017, more than 800 lawsuits were filed claiming that certain websites weren’t accessible to people with disabilities ([Source](#)). Maintaining an inaccessible website can cost your organization dearly.

4 Principles of Accessibility

WCAG has identified four key metrics that help evaluate a website’s accessibility. They are:

Perceivable

Refers to whether the website content can be perceived by a user with disability (including users who employ assistive technology, e.g., screen readers). This guideline asks whether the content can be seen and heard.

Operable

Websites are interactive. Accessible websites are interactive to *everyone*. This metric asks whether users with disabilities can *use* the website as effectively—even if not precisely the same *way*—as users who don’t have disabilities. If you can do it online, *anyone* should be able to do it online.

Understandable

Is the content and site structure understandable to everyone? Does the site’s structure, architecture, interactivity, etc. support the understanding of the content? If, instead, the website’s design makes content less “understandable” to people using assistive technology, it fails on this measure.

Robust

This principle has to do with how many different ways a website makes content accessible. Highly accessible websites, for example, are constructed to be interpreted by a number of assistive technologies (e.g., screen readers). Additionally, accessible sites are built in a flexible manner that ensures they’ll be usable as technology evolves.

Basic Accessibility Techniques

You know web accessibility is important for your organization. But how do you achieve it? Here are some tactics you and/or your website developer should use to achieve a high level of accessibility.

Language Attribute

Screen readers “speak” the content of a website aloud, and they’re often used by people with a visual impairment.

Because a screen reader pronounces the words aloud, it must know which language it’s speaking. Your website should include a **language attribute in the HTML**. It looks something like this:

```
<html lang="en">
```

```
<span lang="fr">“C’est la vie,”</span> the  
woman said as she walked out of the house  
forever.
```

In this example, you’ve defined your language as English (“en”) but, when a section in French appears, the HTML indicates the change in language using the “fr” code.

Make sure a user can add these tags if needed in a WYSIWYG or other text block in your site. Don’t restrict the use of span tags in your WYSIWYG.

Hyperlink Text

Screen reader users will often have the screen reader just read aloud all of the hyperlinks on a webpage when they’re trying to navigate quickly through a site.

Help enhance their ability to do this by ensuring that **your hyperlinks use language that tells the user where the link is sending them**.

Bad Hyperlink Example

Click here to read more about our donation options.

In this bad example, the link is simply the words “Click here,” which wouldn’t tell a user employing a screen reader where the link goes. Where is “here,” exactly?

Good Hyperlink Example

[Learn more about our donation options.](#)

In this good example, the screen reader would tell the user that this link is going to “more about our donation options.”

Skip Links

Offer users a way to skip navigation or other elements that appear on every page—e.g., top menu, footer, etc.

For example, you could add a link at the top of each page. [Skip to Main Content](#) or [Skip Navigation](#). These links can be “hidden” from the normal site interface but will appear when assistive technology is used to read the page.

Page Structure

Perhaps nothing is more helpful to users with a disability than content structure on a webpage. This means building your pages using the tools at your WYSIWYG disposal—including:

- Headlines
- Subheads
- Bullet lists
- Number lists

For headlines, you should build in hierarchical order. In other words, an H1 first, then an H2, then an H3, and so on. If you nest an H1 inside of an H2, it can confuse screen readers.

ALT Tags for Images

ALT tags are “alternative” tags added to images in your CMS that describe what the image is. Again, screen readers will pronounce each image’s ALT tag (if it has one), helping the user understand what’s on the page beyond mere text. ALT tags do not appear on your site, but instead in the code itself, allowing screen readers to find them without displaying them directly on the front end of your site.

Most CMS platforms will give you the option, when uploading/adding an image to a webpage, to enter an ALT tag.

You should always add an ALT tag *unless the image doesn't convey meaning*. For example:



UT Seeks Private Partner to Develop New Basketball Arena

UT Austin is making the first step toward the construction of a new on-campus basketball arena by soliciting interest from potential qualified developers to build and operate a new arena and basketball training facility. [READ MORE >](#)

Currently, this news item has an ALT tag of “UT basketball arena” but, because it doesn’t convey meaning (or further explain the news text below) and the heading below makes it clear that we’re already talking about the UT basketball arena, it should probably be blank.

Tips for Effective ALT Tags

- Writing effective ALT Tags takes some practice. [Check out WebAIM’s tips.](#)
- Present both the content *and* function of the image. Remember, ALT tags depend heavily on the context of the image. Example: “Woman feeding her baby to show the proper method.”
- Don’t make them redundant to surrounding text. If there’s text around an image that describes the image, you don’t need an ALT tag.
- Don’t use “Image of…” or “Graphic of…” in the ALT tag. The screen reader will make this clear to the user.

Tables

When adding a table to your site (in your WYSIWYG), you should connect the column and row headings to specific data. This will help users identify which headers go with which pieces of data.

For example:

```
<th scope="col">Job Title</th> or  
<th scope="row">Name</th>
```

Additionally, you'll want to identify any table captions. Use this as the first tag inside your table markup:

```
<table>□  
  <caption>Class Schedule</caption>
```

Color

Color should *not* be the only indicator used to convey information. There are two key color-related issues for accessibility.

First, hyperlinks should include an underline; they shouldn't simply be a different color from your body text. Without an underline, users with color blindness won't be able to distinguish hyperlink text from regular text.

[You can check your site's color contrast here.](#)

Second, you need to ensure the color of your text contrast sharply with your background color. Otherwise, users with certain vision challenges won't be able to see the text at all.

Note: Color blindness affects as much as 8% of all men (and just 0.25% of women).

Forms

You want to ensure that all forms on your website can be completed by all users. This means that anyone could submit a form using *only* a keyboard and screen reader.

Every element of your forms—text fields, checkboxes, drop-down lists, etc.—has a label in your code. And each label must be associated with the correct form element.

For example:

```
<label for="eyecolor">Eye Color</label>
<input type="text" name="eyecolor"
id="eyecolor">
```

You'll also want to identify *required* form fields with the correct HTML attribute:

```
<input type="text" name="username" required
aria-required="true">
```

Radio Buttons & Checkboxes

These two types of form fields require some special HTML as well. Wrap radio buttons and checkbox groups with “fieldset tags.”

```
<fieldset>
  <legend>Do you love disco?</legend>
  <div>
    <input type="radio" name="discolove"
id="absolutely" value="Absolutely">
    <label for="absolutely">Absolutely! Disco is
amazing.</label>
  </div>
  <div>
    <input type="radio" name="format" id="growup"
value="Grow Up">
    <label for="growup">Grow up. Disco is
lame.</label>
  </div>
</fieldset>
```

Captions and Transcripts

If you have video and/or live audio on your website, you must have captions and a transcript to be fully accessible. The video player must be accessible to keyboard-only users and users of screen reader tools.

Recently, YouTube has made great strides in video accessibility—allowing anyone to add captions to videos using YouTube’s native tools. You can also add transcript files to videos on YouTube, Vimeo, etc.

In your transcript file, you’ll also want to include a separate “description track” to describe what’s happening in the video.

Focus Indicators

If there are elements on a webpage with which a user will need to interact, you should add a “focus indicator” to help direct the (keyboard-only) user’s attention.

The code for a focus indicator looks like this:

```

a:hover,
a:focus {
    outline:1px solid red;
    background:yellow;
}

```

Note: Every a:hover element should have an a:focus element.

Drop-Down Menus

Use a drop-down menu that can be operated with a keyboard alone (without requiring 1,000 keystrokes!). Also, use drop-down menus that can be operated by people who have a difficult time keeping the mouse in one place for a long time.

In short, try to use the menu with a keyboard and see how well it works for you.

Accessibility Resources

WebAim Web Accessibility Check: <http://wave.webaim.org>

W3 Quick Reference: <https://www.w3.org/WAI/WCAG20/quickref>

WebAIM WCAG Checklist: <https://webaim.org/standards/wcag/checklist>

Sortsite - Testing and Mapping Tools: <https://www.powermapper.com/products/>

Pa11y - Free Accessibility Tools – <http://pa11y.org>

Asqatasun - Test Your Website for Accessibility: <http://asqatasun.org>



aXe from Deque - Accessibility Testing: <https://www.deque.com/products/axe/>

WorldSpace Comply - Advanced Accessibility Compliance:
<https://www.deque.com/products/worldspace-comply/>



Let's Build Something Useful. Together.

hello@mightycitizen.com | (512) 459-7000