CMS Workshop: Accessibility

Hello and good morning! My name is Charlotte Koh, and I am the University Webmaster for USF. I'm really excited to speak with you today about accessibility, since it is incredibly important to the work we all do on the web. My goal for this workshop is to provide a high level overview of what exactly accessibility is, and then transition into demonstrating some basic things we can do in the CMS to improve our content.

First, I would like to emphasize that no one's content is perfect. The reality is that we are all going to make mistakes, and what's important is that we learn from them and commit to doing better. I definitely don't get accessibility right 100% of the time, but I try to do my best and be open to feedback, and that's all I'm going to ask of you.

What is Web Accessibility?

So, what exactly is web accessibility? According to the World Wide Web Consortium's Web Accessibility Initiative, it means that "websites, tools, and technologies are designed and developed so that people with disabilities can use them."

People can experience varying degrees of disabilities that can impact their ability to access the Web. This can include:

- Auditory disabilities, such as mild hearing loss and deafness
- Cognitive, learning, and neurological disabilities, which impact how people process information. For example, ADHD, autism, dyslexia, etc.
- Physical disabilities, sometimes referred to as motor disabilities, which can include tremors, arthritis, reduced dexterity, and amputation
- Speech disabilities, such as stuttering and muteness
- And finally visual disabilities, which can include low vision, blindness, and color blindness

What's great about accessibility is that when you ensure your content is accessible, you improve the experience on your website for everyone, not just users with disabilities. People visiting the USF website may be using their phone, or perhaps they have a slow Internet connection. Alternatively, someone may have what's called a "temporary disability" such as a broken arm, or perhaps a "situational disability" where they want to watch a video but can't listen to audio at the moment.

Laws and Guidelines

Aside from accessibility just being the right thing to do, it's also the law, specifically two laws - the Americans with Disabilities Act and Section 508 of the Rehabilitation Act. Legal requirements for web accessibility have generally been vague in the past, but Section 508 underwent a refresh in 2018 that requires us to meet or exceed the Level AA standards of the Web Content Accessibility Guidelines 2.0 (WCAG 2.0).

The Web Content Accessibility Guidelines have four principles that we follow, referred to as POUR:

- Perceivable: Users must be able to perceive information and user interface components
- Operable: Users must be able to operate user interface components and navigation
- Understandable: Users must be able to understand information and the operation of user interface components
- Robust: Users must be able to access content by different means (including assistive technology) as technology advances

Now that we have a foundation for accessibility and why it's important, let's talk about what we can do to make our CMS content more accessible and user-friendly.

Titles

First up is titles. Pages in the CMS should have titles that describe their topic or purpose. Descriptive titles can help users identify a page without the need to read through the page's content.

By default, CMS pages have the title of "Title" on page creation. The field is a bit tucked away, and we recognize that sometimes folks forget to update it, so we're working on a way to have it be more relevant in case it slips through the cracks and gets published. We can find the "Title" field by navigating to the page's Properties and then Parameters, and the field will be the first on the page. Let's change our page title from "Title" to "CMS Accessibility Workshop".

The CMS User Guide's entry for Search Engine Optimization has further examples of page titles.

Headings

Next, let's talk about headings. Headings help convey how information on a page is structured. They do not need to be lengthy, but they should be descriptive.

Headings have different levels, or ranks, and can be nested to create subsections of content. They go from H1 (the most important) to H6 (the least important). You should avoid skipping heading levels unless you are closing a subsection, as it can be confusing to users (for example, going from H3 to H5 vs going from H5 to H3). Also, please do not use headings as a way to style or emphasize text. They should only be used if the text in question is a heading.

Some heading levels are used within the CMS templates and are not meant to be utilized in the what-you-see-is-what-you-get, or WYSIWYG, editing regions. For example, on this text page, "CMS Workshop" is an H1 and "Accessibility" is an H2. That means you would want to use H3-H6 when organizing your content.

Some other page templates might function a little differently, such as news articles allowing you to use H2-H6. If you ever have any questions about which headings to use, please don't hesitate to reach out.

Since this page is about accessibility, let’s add headings for the different sections I’ll be discussing. We’ll use the dropdown menu that says “Paragraph” to create H3s for “Titles”, “Headings”, “Links”, “Images”, and “Documents”. The “Images” section is going to cover a bit more than the others, so we’ll add H4s to create the subsections “Alt Text”, “Images of Text and Contrast”, and “Infographics”.

As you add headings to a page, keep in mind that you should use proper capitalization regardless of how the text is styled. Write the text like you normally would, and let the templates handle if the heading become styled as all caps, as is the case with our H3s on text pages.

Links

Let’s continue on to links. The language we use for links should be meaningful and descriptive. We want users to understand the purpose of a link so they can decide if they want to follow it.

Ideally, the link’s purpose can be distinguished from the link text alone, without needing additional context. Sometimes assistive technology can present a list of all the links on a page to a user, and a bunch of “click here” links is not very helpful.

Let’s add a link within our “Link” section to the Student Accessibility Services Accessibility Guide. We want to avoid doing something like “Click here to view the guide.” Instead, let’s write “Student Accessibility Services has an excellent Accessibility Guide,” linking the words “Accessibility Guide”.

If you want extra credit and a gold star from me, consider including relevant information about the link target as part of the linked text. For example, “Accessibility Guide (PDF, 1.1MB)”.

Images

Alt Text

And now we’ve reached images. One of the building blocks of accessibility is alternative text for images, or alt text. Assistive technology can use alt text to describe an image to a user, and it also can appear in case an image fails to load.

When coming up with alt text for your graphics, think of what you would want a screen reader to say when it comes across the image, or how you would describe the image to another person. We shouldn’t include the phrases “picture of” or “image of”, as screen readers automatically announce that an image is an image.

Let’s add a “Medium Media – Right” snippet to the page, and replace the placeholder with a doodle. When adding an image to a page via the WYSIWYG editor, the “Alternative description” field functions as alt text, and you should try to limit the text to 100 characters or less. If your alt text is beginning to look pretty lengthy, consider if all that information would be better presented as part of the page's text. For now, we'll update the "Alternative description" field to read "Illustration of a velociraptor, looking to the right over its shoulder."

Images of Text and Contrast

Next, we're going to talk about images of text, as well as contrast. It is preferable to have information in text form vs images of text. When text is presented within an image, it cannot be adjusted by users if necessary (such as increasing font size). The exception is if you find yourself in a situation where an image is absolutely necessary, such as it cannot be formatted in the CMS to the same visual effect or it's a logo.

In these situations you want to make sure the image has appropriate alt text and contrast between the text and its background. The guidelines we follow require a contrast ratio of 4.5:1 for normal text and 3:1 for large text (which is 14pt/18.5px and bold or larger, or 18pt/24px or larger).

Luckily, an organization called WebAIM has created a Contrast Checker to calculate the contrast of your foreground and background colors. Using the tool, I can determine that our USF Green (#006747) has a contrast ratio of 6.92:1 when paired with white, which passes for both normal and large text. This meets our requirement of conforming to WCAG's Level AA standards. As an aside, if we wanted to reach for Level AAA conformance, we would need to meet a contrast ratio of 7:1 for normal text, which is why the tool says "Fail" for WCAG AAA.

Infographics

And the last part we will touch on with images is about infographics. Infographics are a little bit more complicated than your standard image. If you are unable to convey all of the infographic's content within the confines of alt text, you need to create a text version of the infographic. You can either include the text version on the same page as the infographic along with a meaningful heading, or you can link to a separate page that houses the text version. If you go the link route, the link should be immediately before or after the infographic and the linked text should also be meaningful.

A great example of an infographic alongside a text description is on the WebAIM site. The infographic is presented first, followed by the text version. The infographic's alt text reads "Web Accessibility for Designers infographic with text description below".

Documents

Last, but not least, are documents. Documents you create, including Word documents, PDFs, and PowerPoints, should also be accessible. Guidance in this area is a bit outside of my purview, but Student Accessibility Services has a very thorough guide to ensuring your documents are accessible, along with other valuable information.

Tools

So, there are a wide variety of tools and resources to help us improve our content, several of which are linked on the CMS User Guide Accessibility page. There's the WebAIM Contrast Checker I mentioned earlier, as well as links to screen readers and suggestions like color blindness tools.

One tool I specifically want to bring to your attention is the WebAIM WAVE evaluation tool. WAVE allows you to submit a URL and it returns a really cool overlay that indicates errors and potential issues. If you decide to try out WAVE, it will check everything on the page, including things managed by the templates that are outside of your control. So don't be alarmed if it highlights things you can't edit, instead focus on your content and such. Aside from the Summary and Details panes, the Order and Structure panes also provide insight into how assistive technology interprets your page.

Questions

That brings us to the end of our accessibility overview. Does anyone have any questions?

Conclusion

If you're interested in learning more about web accessibility beyond what we've covered today, I recommend going to the source itself and checking out the Web Content Accessibility Guidelines. They can be pretty intimidating at first, but the website offers overviews, quick references, and pages that break down each guideline.

We covered a lot today, so thank you for sticking with me. Your time is valuable, and I appreciate that you joined us. I'll be sharing a transcript I wrote up beforehand, as well as a list of the links I mentioned. Please feel free to reach out if you ever want to chat about accessibility, and I hope you all have a great day!