

Accessibility

“Five Things You Can Do Right Now” *

1. Fix the usability problems that confuse everyone.

If something confuses most people who use your site, it’s almost certain to confuse users who have accessibility issues. (They don’t suddenly become remarkably smarter because they have a disability.) And it’s very likely that they’re going to have a harder time recovering from their confusion.

For instance, think of the last time you had trouble using a Web site (running into a confusing error message when you submitted a form, for instance). Now imagine trying to solve that problem without being able to see the page.

The single best thing you can do to improve your site’s accessibility is to test it often, and continually smooth out the parts that confuse everyone. In fact, if you don’t do this first, no matter how rigorously you apply accessibility guidelines, people with disabilities still won’t be able to use it. If your site’s not clear to begin with, making it Bobby-compliant is like [insert your favorite putting-lipstick-on-a-pig metaphor here].

2. Read an article.

The best way to learn how to make anything more usable is to watch people actually try to use it. But most of us have no experience at using adaptive technology, let alone watching other people use it.

If you have the time and the motivation, locate one or two blind Web users and spend a few hours with them observing how they actually use their screen reader software.

Fortunately, someone has done the heavy lifting for you. Mary Theofanos and Janice (Ginny) Redish watched 16 blind users using screen readers to do a number of tasks on a variety of sites and reported what they observed in an article titled “Guidelines for Accessible and Usable Web Sites: Observing Users Who Work with Screen Readers.” <http://redish.net/content/papers/interactions.html>

As with any kind of user testing, it produced invaluable insights. Here’s one example of the kinds of things they learned:

Screen-reader users scan with their ears. Most blind users are just as impatient as most sighted users. They want to get the information they need as quickly as possible. They do not listen to every word on the page – just as sighted users do not read every word. They “scan with their ears,” listening to just enough to decide whether to listen further. Many set the voice to speak at an amazingly rapid rate.

They listen to the first few words of a link or line of text. If it does not seem relevant, they move quickly to the next link, next heading, next paragraph. Where a sighted user might find a keyword by scanning over the entire page, a blind user may not hear that keyword if it is not at the beginning of a link or a line of text.

Read this article before you read anything else about accessibility. In 20 minutes, it will give you an appreciation for the problems you're trying to solve that you won't get from any other articles or books.

3. Read a book.

After you've read the above article, you're ready to spend a day (or a weekend) reading a book about Web accessibility. There are several good ones...

Building Accessible Websites by Joe Clark

Constructing Accessible Websites by Jim Thatcher et al.

Maximum Accessibility: Making Your Web Site More Usable for Everyone by John Slatin and Sharron Rush

A CD-Rom called "The WebAIM Guide to Web Accessibility Techniques and Concepts"

These books cover a lot of ground so don't worry about absorbing all of it. For now, you just need to get the big picture.

4. Start using Cascading Style Sheets

Cascading Style Sheets are now so well supported by most browsers that it doesn't make any sense to create a site without them, because the advantages are enormous:

- **Infinitely greater control of formatting**
- **Flexibly.** A single change in a style sheet can change the appearance of an entire site, or automatically generate useful variations like printer-friendly pages.
- **Consistency among browsers.** Workarounds and hacks are still required to ensure that your CSS works across all browsers, but these will fall away as browser makers continue to improve their CSS support.

And implementing CSS will make it easy for you to do two things that will greatly improve your site's accessibility:

- **Serialize your content.** Unlike table-based layout, with CSS you can put your content in sequential order in the source file – which is how a screen reader user will hear it – and still position things where you want them on the page.
- **Allow your text to resize.** CSS makes it easy to make your text resizable, which is enormously helpful for low-vision users (and people old enough to need bifocals).

Probably the fastest way to learn CSS is to get someone who specializes in it to do a “markover” for you – recoding a few of your site's page templates to use CSS - and learn by watching them do it. When you're ready, there are also a number of good books on CSS, especially the ones by Eric Meyer.

5. Go for the low-hanging fruit

Now you're ready to do what most people think of as Web accessibility: implementing specific changes in your HTML code.

As of right now, these are probably the most important things to do:

- **Add appropriate alt text to every image.** Add an alt attribute for images that screen readers should ignore, and add helpful, descriptive text for the rest. All of the Web accessibility books have very good explanations of how to do this.
- **Make your forms work with screen readers.** This largely boils down to using the HTML label element to associate the fields with their prompts, so people know what they're supposed to enter.
- **Create a “Skip to Main Content” link at the beginning of each page.** Imagine having to spend 20 seconds (or a minute, or two) listening to the global navigation at the top of every page before you could look at the content, and you'll understand why this is important.
- **Make all content accessible by keyboard.** Remember, not everyone can use a mouse.
- **Don't use JavaScript without a good reason.** Some adaptive technologies don't support it very well yet.
- **Use client-side (not server-side) image maps.** Alt tags don't work with server-side image maps.

That's it. You'll probably learn how to do a lot more as you go along, but even if you only do what is covered here, you'll be doing a pretty good job.

* Krug, S. (2005). Don't make me think: A common sense approach to web usability, (2nd Ed). pp. 174-179. Berkeley, CA: New Riders.