



Quick Web Accessibility

A guide for busy
voluntary sector
managers and designers

Produced by the Sensory Trust

Contents

What's in this guide?

How much will it all cost?

About the accessibility guidelines

How do I know if my web site is accessible?

Design guide

Quick dos and don'ts

Techniques for priority 1 accessibility

1. Use stylesheets
2. Use HTML markup properly
3. Use relative sizes for text and tables
4. Make sure all images and other non-text elements have a text alternative
5. Identify the natural language of the text
6. Use the simplest and clearest language possible
7. Make sure pages in the site can be easily navigated by everyone.
8. Ensure that your site can be navigated without a mouse
9. Ensure data tables identify row and/or column headers
10. Provide an alternative

Credits

About the author

About the Sensory Trust

What's in this guide?

This guide shows you

- How to **check your site** for accessibility
- How to **make your site accessible** to the basic Priority 1 level as set out in the Web Access Initiative guidelines.

This guide is designed to be **easy to use** in order to make the process of adding accessibility to your site as painless as possible.

For managers

The guide outlines the techniques you can use to **assess your existing site for accessibility**. After you have followed the instructions in this guide, you should know the questions to ask your designers, and be able to check that the job has been done properly.

For designers

This guide details the techniques used to **design for basic accessibility** requirements. It only scratches the surface of what you can do to make your site accessible, but it's designed to make it easy for you to quickly make your web site compliant and will point you in the right direction should you want to do more.

What do I need to know already?

For those involved in **building or maintaining web sites**, this guide assumes that you have a basic knowledge of HTML and web design.

For **managers**, all you need is the desire to make your web site accessible.

It's worth saying here that you might have to sacrifice some of the glitz of your old site in order to make your new site accessible and easier to use.

Good web design revolves around valuable content.

Why be accessible?

You have a moral obligation to provide the same level of service to **all your customers** regardless of their age or level of ability. You also have a **legal obligation** to make your services, including your web site, accessible to as many people as is reasonable practicable. These obligations are set out in the Disability Discrimination Act.

The good news is that by designing your web site for accessibility you will be making it **easier for everyone** to use. Your message will be more readily understood, and you will achieve more through your site.

More visitors, more user satisfaction, better communication.

"The power of the Web is in its universality. Access by everyone regardless of disability is an essential aspect."

Tim Berners-Lee
W3C Director and inventor of the World Wide Web

How much will it all cost?

More good news here. The Sensory Trust has put together this free guide so that managers and designers can do a lot of the work **easily, quickly and economically**.

Bolting accessibility on to an existing site is much more time-consuming than developing a new site with accessibility built in. This guide deals with **Priority 1 accessibility** (see page 8) in order to make the bolting on process as quick and painless – and economical – as possible.

If you plan to redevelop your web site in the near future, you should consider **going beyond Priority 1**, as many of the access techniques can be easily integrated into any web site redevelopment plan.

About the accessibility guidelines

The **Web Accessibility Initiative** (WAI) is the organisation responsible for developing the accessibility guidelines that are now in use throughout the World Wide Web. The guidelines are divided into **3 priorities**. The effect of failing to meet these guidelines is summarized below.

Fail Priority 1 – Some users will be **denied** access to page content.

Fail Priority 2 – Some users will find access **extremely difficult**, and may be denied page content.

Fail Priority 3 – Some users will find access difficult to and may be denied some elements of the page content.

In order to provide a quick and economical route to making your web site accessible, this pack only deals with **Priority 1** accessibility.

There are other **additional guidelines** produced by the different governments but for simplicity this pack, which is aimed at UK organisations, is based on the WAI guidelines.

How do I know if my web site is accessible?

There are **two methods** of checking a web site for accessibility. You will need to **use both** to gain a full and accurate picture of your web site's accessibility status.

Access testing software

bobby.watchfire.com/bobby/html/en/index.jsp

This software allows you to check the accessibility of your site one page at a time. You can also buy the full software to run from your own computer. This enables you to check all the pages in your site for accessibility. Bobby is just one of a number of tools out there that do the same thing.

See also Lift for Dreamweaver (www.usablenet.com)

Manual checklists

You will also find a set of accessibility **checklists** at the Web Accessibility Initiative web site (www.w3.org/WAI/) which you can either download or print out from the site.

The accessibility checking software will pick up a lot of the accessibility issues, but the computer cannot check things like colour contrast, adequate text alternatives for images, plain English and so on.

Human checking must be done!!!

Design guide

Tools

The easiest way to make sure your HTML code conforms to a recognized standard is to **use a good authoring tool**. For the professional, there is one clear market leader: Macromedia Dreamweaver (www.macromedia.com). This allows many optional extras to be added (such as a Bobby style accessibility checker) which make building accessible web pages easy.

If you are going to do the work in-house and you are a non-profit organisation, you may be able to get discounted prices on this software. Check with your supplier.

Quick dos and don'ts

Do

1. Use cascading stylesheets (css) and make sure that pages can be read without them.
2. Use HTML markup properly.
3. Use relative sizes for text and tables.
4. Make sure all images and other non-text elements have a text alternative.
5. Identify the natural language of the text.
6. Use the simplest and clearest language possible.
7. Make sure pages in the site can be easily navigated by everyone.
8. Ensure that your site can be navigated without a mouse.
9. Ensure data tables identify row and/or column headers.
10. Provide an alternative page if all else fails.

Don't

1. Don't use frames – they are unnecessarily complicated, make bookmarking difficult. and can confuse some screen readers if not implemented properly.
2. Don't use deprecated HTML tags like (bold) or <I> (italic)
3. Don't use HTML to control the look of a page.
4. Don't use patterned or textured backgrounds/watermarks.
5. Don't use fixed sized fonts.
6. Don't use fixed width layout tables.
7. Don't use blinking text, animated gifs or anything that causes flickering on the page.
8. Don't rely on colour to convey information (e.g. red text for important words)
9. Don't rely on applets and scripts to provide content – provide alternatives.
10. Don't use multi-column layouts for text unless you make sure the layout still makes sense when read with screen reading software.

Techniques for priority 1 accessibility

By using the techniques outlined in the following pages, and using access checking software, or the WAI checklists, you should be able to quickly bring your web site into line with basic Priority 1 accessibility guidelines.

This booklet is not meant to be a complete guide to web accessibility techniques. With the philosophy that a little accessibility is better than none, it's written with the **real world** in mind. The techniques here are presented in an easy to follow format that is designed to get you through the basics as quickly and painlessly as possible.

You may want more detail on some of the techniques, or you might want to take your access work further. In this section you'll also find links to web sites that go into more detail, and give more examples.

1. Use stylesheets

Use external stylesheets to control how your web pages appear. Link them using the <LINK> tag in the <HEAD> of the HTML document.

```
<LINK rel="stylesheet" href="/css/style1.css" type="text/css">
```

Using external stylesheets means that you effectively separate the style of the page from the content structure and means that, providing you have used the HTML markup correctly, screen reading software should be able to interpret the page correctly.

Using stylesheets also allows some users such as people with visual impairment or Dyslexia to view your site with their own stylesheets which they have customized to their own preferences.

When testing your pages, make sure that they can still be read with stylesheets turned off in your browser preferences.

2. Use HTML markup properly

Use HTML tags for their **intended purpose only**.

HTML is designed to define the structure of a document, **not control what it looks like**. For instance, the <H1> tag should be used only for the main heading on a page, not for making text bigger. Sub headings should be <H2>. Sub-sub headings <H3> and so on.

If you think the text looks too big, or you want to change the font or colour, use stylesheets, not HTML.

It is important to get the HTML structure of the page right, as there are many different screen readers and other assistive technologies that use the markup to interpret how to deliver the text. Using headings correctly allows some screen readers to jump from heading to heading to quickly find content of interest. With this in mind, make your headings descriptive: 'Castles in Andalucia' is better than 'Section 3', for instance.

More information on structure and presentation can be found at

www.w3.org/TR/WCAG10-CORE-TECHS/#structure

3. Use relative sizes for text and tables

Many users can read text on the screen if they set their browser to display the largest size text. Using relative sizes for fonts and tables in your stylesheets and HTML allows the user to change these sizes and allows the layout to fill the existing screen.

Filling the screen means that the layout will also shrink to fit small screens like old monitor displays of PDAs, which means it should eliminate, or reduce, the need for horizontal scrolling.

Relative sizes are in %, ems and exs. It is safest to use % as the other measures are not fully supported in older browsers.

Be prepared to sacrifice some control of your page layout in favour of accessibility.

4. Make sure all images and other non-text elements have a text alternative

Use the ALT attribute to describe pictures on your page. Screen reading software will read this attribute out to blind users, so make the ALT text useful. For instance, if you have a photograph of a windmill, don't put 'photo 1'. Make it descriptive. 'Photo of a windmill in Castille' is much better.

If the picture requires a lot of explanation (for instance a graph of results) then it is better to provide a description on a separate page. There is a LONGDESC attribute which you can use, but this isn't fully supported across browsers. A common workaround is to place a 'D' next to the image which links to a page with a description of the image but this isn't recommended. It's better to use longdesc as support for this is increasing and this is valid markup.

Use the same techniques for video, Shockwave/Flash or other multimedia content that is unreadable.

Many web pages are held together with 'invisible' spacer images. If you lay your pages out using CSS, you will eliminate the need for these, but if you can't, then don't forget to give each one a blank ALT attribute so that blind people won't be left wondering whether the image was important.

```
<IMG src="spacer.gif" alt=" ">
```

5. Identify the natural language of the text

Use markup to identify the natural language of the page. This makes audio output from the page more accurate and comprehensible. The code for a page written in UK English appears in the <HTML> tag like this:

```
<HTML lang="en-gb">
```

If you want to quote other languages within an English document, add the attribute to the tag surrounding the text:

```
He has what the French call<EM lang="fr">joi de vivre</EM>.
```

More information on language codes at:

www.w3.org/TR/REC-html40/struct/dirlang.html#h-8.1.1

6. Use the simplest and clearest language possible

The best HTML in the world won't help your site if the text is confusing, badly organized and unnecessarily complex.

Keep your pages short, and organize them so that the message of the page is summed up in a sentence or two first, and then expanded after that. This makes it easy for people to judge quickly what the page is about, and whether or not to read it all.

Break up pages into logical sections and use headings to describe each section.

Use HTML markup properly to break up large areas of text. Use bullet points if appropriate.

You can find more information about Plain English and how to write for the web at:

www.plainenglish.co.uk/plainenglishguide.html

www.useit.com/papers/webwriting/

7. Make sure that all pages can be navigated by everyone.

Image maps are images with ‘hotspots’ that the user can click to go to other pages. Server-side image maps can cause problems for some people. Most commercial web authoring tools allow you to create client-side image maps instead.

When you create image map hotspots, don’t forget to provide each one with an ALT attribute.

It’s a good idea to provide a text version of the links in your image map somewhere on the page too.

Provide a link at the start of every page that allows users to skip over repetitive elements like navigation bars and go straight to the content.

Provide a link at the end of the content that goes back to the top of the page.

Some browsers list the links separately from the content. For that reason, use link text that makes sense when read out of context. For instance, ‘click here’ will mean nothing to many users, but ‘More about windmills in Castille’ will.

8. Ensure that your site can be navigated without a mouse

Some people have difficulty using a mouse. Make sure your web pages can be navigated using only the keyboard.

Most browsers allow the user to tab between links on a page. You can specify a logical order for the links by giving each one a number using the 'tabindex' attribute.

```
<a href="home.html" tabindex="20">
```

It is a good idea to number your links in tens: 10, 20, 30 and so on. That way, if you want to add more links at a later date, you don't have to renumber every one, just add numbers (11, 12 etc.) in-between.

Keyboard shortcuts can also be provided for the most common links on a page using the 'accesskey' attribute.

```
<a href="home.html" tabindex="20" accesskey="H">
```

In the example above, the user hits a key or keys (depending on their system) plus the letter key to go to that page.

9. Ensure data tables identify row and/or column headers

Data tables, such as those used in a spreadsheet or timetable, need to have their headers specified, and related to each column or row. This means that tabular data will make sense when read out by a screen reader.

Identify column or row headers with the <TH> tag, and table cells with the <TD> tag.

The simplest way to associate the cells in a row or column with its header is to use the SCOPE attribute:

```
<TH scope="col">Organisation</TH>
```

This links all cells in the column below with the header.

Other scope values are “row”, “rowgroup” and “colgroup”.

10. Provide an alternative

If, after all your best efforts, you still can't create a page that conforms to basic accessibility requirements, then provide a link to an equivalent page that is accessible, provides the same information and is **updated as often** as the original page.

Congratulations!

You should now have a web site that conforms to basic accessibility requirements. Many more people will be able to read your pages than before, and everyone should find it easier to get around your site.

So shout about it! Add an accessibility statement to your site. This should explain what your accessibility policy is, what work you have done, what areas of the site still may have problems, what you plan to do in the future.

Offer your users alternative ways to contact you, and alternative formats for information that still may be inaccessible.

Disclaimer

While every effort has been made to ensure that the information contained in this guide is accurate, the authors accept no liability for damage or loss resulting in the use of this guide. This guide is provided for information only and does not constitute advice or form a contract of any kind.

Credits

Special thanks to David Sloan of the Dundee University Media Access Group for his valuable input and suggestions.

About the author

Lars Stenberg has been working in web design since 1995. For years he has advocated accessibility and usability in web site design and has overseen the redesign of many web sites including the Scottish Parliament.

He now works for the Sensory Trust as Information Designer, working on the provision of information for a wide range of audiences through a broad range of media.

lstenberg@sensorytrust.org.uk

About the Sensory Trust

The Sensory Trust works with disabled and socially excluded people to inspire, support and promote best practice in inclusive environmental design and management. We do this through a programme of networking, advice delivery, design, research and dissemination of findings.

Our inclusive design team is involved in a range of projects that explore new ways of designing and managing public outdoor space. Inclusive design covers all aspects of landscape access and inclusion, from removal of physical barriers to interpretation and information provision. These projects are promoted as examples of best practice.

We work in collaboration with key organizations and projects throughout the UK, for example the Eden Project in Cornwall, Sheffield City Council, English Nature's Heathland Restoration Programme and Landlife's Urban Wildflower Centre in Liverpool.

We are not a web design company. This guide is impartial and designed to be as helpful as possible. It is published as part of our programme of supporting good practice in the provision of information and interpretation.

www.sensorytrust.org.uk