Modern website design for sighted and visually impaired users

John R Hudson

12th November 2024

1 Introduction

Between 2011 and 2018 a number of additions were made to HTML and CSS to increase website accessibility for visually impaired users; these rely on

1. the five semantic elements (figure 1) introduced in 2011 and

2. the website designer laying out the HTML elements in a way which makes it easy for a visually impaired user to 'read' the page while making the content visually attractive to a sighted user through the use of CSS properties.

One aspect of the thinking behind the five semantic elements was that computer screens had evolved from being almost square in the 20th century to elongated rectangles in the 21st century, resulting in very large line lengths for the main content; putting the menu on the left and creating a space on the right for links and matter which is not central to the content of the webpage results in shorter line lengths for the main content.

However this did not make any difference when visually impaired users were presented, in a traditional website, with the content in the order `<header>`, `<nav>`, `<article>`, `<aside>`, `<footer>`. By declaring the `<main>` element as a flex container it is possible to define the order and properties of the `<nav>`, `<article>` and `<aside>` elements within the `<main>` element in the CSS file so that, even when the content of the `<article>` element comes before the content of the `<nav>` element in HTML for the benefit of a screenreader user, the content of the HTML elements will still display on a screen in way which suits a sighted user:

```
main {display: flex;}
main>article {order: 2;  width: 58%; padding-left: 3em; padding-right: 3em;}
main>nav {order: 1; width: 19%; padding: 1em;}
main>aside {order: 3; width: 19%; padding: 1em;}
```

(The widths and padding were chosen to suit the particular website from whose CSS file this example was copied but can be different from those quoted in the example.)

Moving all the menus from the `<header>` element to the `<nav>` element makes the `<header>` element just a brief 'header' to the page.

To a screenreader the start of the Home page of the Heath Old Boys Association website looks like figure 2. 'Heath Old Boys Association' and 'Welcome' are the only content of the
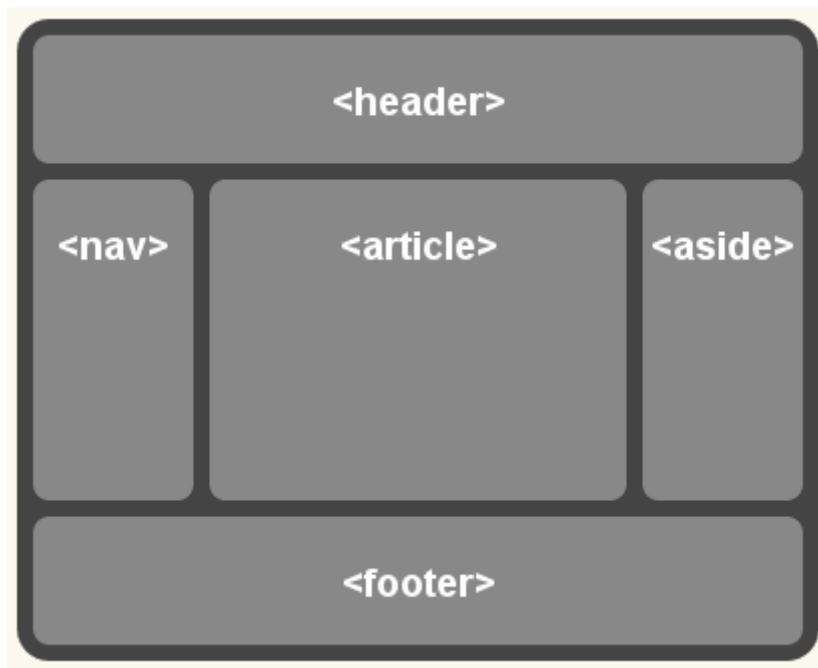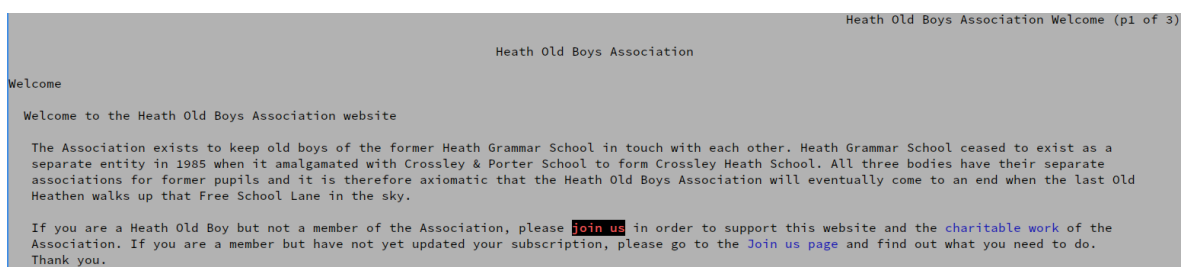
Figure 1: The five semantic elements

Figure 2: Screenreader view of Heath Old Boys Association website Home page

`<header>` element. The next line 'Welcome to the Heath Old Boys Association website' is the title of the `<article>` element on the Home page and the content of the element follows. For the sighted user there is an elongated montage of images of the school and the apple and pear window (which was a noted element of the original building) as well as the words right-aligned in the `<header>` element; this and the colours used for the border and the text are all defined in the CSS file:

```
header {border-top: medium solid #C71585; border-right:
medium solid #C71585; border-left: medium solid #C71585;
background: transparent url("graphics/Topback_new.png")
no-repeat; padding-right: 1em;}
h1 {font: bold 180% sans-serif; text-align: right; color: gold;}
h2 {font: bold 160% sans-serif; text-align: right; color: gold;}
```

The only thing which changes in the headers to the other pages on the website is the text which is in the HTML element.

2 The <header> element

With the menus moved to the `<nav>` element, the only things a `<header>` element holds in an accessible website is a heading to the page and possibly options to log in and to change language. Each of these can be handled by links in HTML; to add any explanatory text for a visually impaired user, add it in HTML but set the font size to 1pt so that it appears in the screenreader but is invisible on a sighted user's screen. To enhance the log in or language buttons for a sighted user assign `mask` related CSS properties to them so that, for example, a national flag represents the button link to the pages in that language.

3 The <main> element

The `<main>` element follows the `<header>` element and contains the `<article>`, `<nav>` and `<aside>` elements.

4 The <article> element

For a visually impaired user the `<article>` element should come first as they will normally have clicked on the page in order to read its content. A number of HTML elements and attributes are particularly useful for visually impaired users:

- the `<i>` element is used to indicate a foreign language which is normally printed in italics — to italicise English use the `<em>` element. The `<i>` element takes the `lang` attribute as in `<i lang="fr">` to indicate that the words in the element are French.[1] The screenreader will either change language if it has the capacity or offer to spell out the words;

---

[1]For the full list of language tags, see http://www.iana.org/assignments/language-subtag-registry

- the `<time>` element, for example, `<time datetime="2024-08-31T18:00+01">August 31, 2024 at 7pm</time>` where `+01` indicates BST, allows a machine readable date (and time) to be specified along with a human readable date (and time) ensuring that any ambiguity in the way the date (and time) is expressed in human readable form is clarified;

- the `alt=""` attribute has long been available but is widely misunderstood; it should contain a description of the image to which it relates, such as, 'Three rows of boys with the front row seated and the others standing and the sports master in the middle of the back row.' Where the image is in a `<figure>` element, the `<figcaption>` element will ideally contain the names of all those in the image;

- the `voice-*=" "` attributes allow you to designate the type of voice to be used to by a screenreader; for example, `voice-family=" "` designates whether the screenreader is to use a male or female voice, that of a child, a young adult or an older adult; `old` designates an older adult voice; `young` designates a young adult voice; `preserve` ensures that child elements inherit the voice:

      blockquote.oldmale {voice-family: old male; preserve;}
      blockquote.oldfemale {voice-family: old female; preserve;}

- the WAI-ARIA attributes have long been available, though they were only formally adopted by W3C in 2014 and have more recently been reduced in number and given more specific meanings. In many cases, there is only one WAI-ARIA attribute appropriate to a particular element but, in other cases, the choice of attribute is important for clarifying the purpose of the element; for example, an image intended to illustrate the article should normally have `role="img"` and a completed `alt=""` attribute; an image added purely for the benefit of a sighted reader should have `role="presentation"`;

- where an element containing a visible label is present, an `id=""` attribute can be added to the element and the child elements linked to the `id=""` attribute with `aria-labelledby=""`; otherwise use `aria-label=" "` to label the child element where there is no other form of label in the element or its parent element.

5 The <nav> element

The `<nav>` element will normally be the next element which a visually impaired user encounters but sometimes the `<aside>` element may come first as, for example, where there is a list of recently added pages. The same principles apply to creating the menus in the `<nav>` element as apply to any links the `<header>` element: construct the menus in HTML so that they suit the needs of a visually impaired user and then use CSS to highlight or 'prettify' particular elements in the menus; for example:

      li.nav {font: bold 120% cursive; list-style: none; line-height: 150%}
      li.aside {font: bold 100% sans-serif; list-style: none; line-height: 150%}

The increased line height ensures that menu items are spaced for the benefit of sighted smartphone users.

6 The <aside> element

The `<aside>` element, if present, will normally come after the `<nav>` element. Putting this before the `<nav>` element can tell the visually impaired user something significant, such as, what is new, and give them the chance to go straight to those pages rather than scrolling through the links in the `<nav>` element to see if there is something new. However, in most cases, putting the `<nav>` element second will be most helpful to a visually impaired user and, even where an `<aside>` element comes before a `<nav>` element, that will normally only be on one page in the whole website.

In this example of an `<aside>` element coming after the `<article>` element, there is a link to a page about searching the website, a list of dates for your diary and then a list of a dozen or so recent additions, all of which may be more useful for a screenreader user to be aware of than the menu in the `<nav>` element:

```
<aside role="complementary">
<h3><a href="content/Searching.html" type="text/html" role="link">
Searching the website</a></h3>
<hr>
<h3 id="recent">Dates for your diary</h3>
<ul role="list">

    <li class="aside"><time datetime="2024-12-14">14 December 2024
    </time><br><a href="news/2024_Christmas_Day_Trip.html"
    type="text/html" role="link">Christmas Day Trip</a></li>

</ul>
<hr>
<h3 id="recent">Recent additions</h3>
<ul role="list">

    <li class="aside"><a href="memories/1952_Cricket.html"
    type="text/html" role="link">1952 Cricket Team</a>
    <span> -- NEW</span></li>
    ...
```

7 The <footer> element

The `<footer>` element follows the `<main>` element and normally holds information about the website unrelated to the content of the website, such as contact details and links to social media. For both a sighted and a visually impaired user, this information normally comes at the bottom/end of the webpage and, in many cases, a visually impaired user will have left the webpage on reaching the `<nav>` element. Consideration may need to given as to whether any important links in the `<footer>` element should be duplicated for a visually impaired user in the `<nav>` element so that they do not leave the webpage before reaching them; however, given the limited number of accessible websites there are at present most visually impaired users will be aware that such information is normally at the bottom/end of the webpage.