MY BROTHER DARRYL

Accessibility Testing

One of the last stages, and most crucial stages of launching a successful website is the final testing process. This is where you simulate using your site as someone who has no idea what they're doing, using different devices, software and in different situations.

Tunnel vision is a huge problem in the industry and many times developers overlook common issues because they're focused on something else. You may want to send your website link to people outside of the project to see how they use the website, and help identify issues and bottlenecks that you may not notice. Gaining feedback from a wide demographic of users can be extremely helpful in the final stages of a website. Using heatmaps and recording mouse movement can also give you insight into how the site is being used. You may find for example an image looks too much like a link, and users are constantly clicking on it trying to gain access to information.

For accessibility testing we take a couple of other things into consideration. Such as double checking colour contrast, descriptive alt tags on images, and checking keyboard navigation. Here are some tools and a handy checklist to help you through this final stage.

Disclaimer:
This is a best practice checklist. It does not guarantee that your site is 100% accessible. Infact, it's highly unlikely that a modern engaging site can be 100% accessible and you should be wary of companies and services that make such promises. However, this list will help you address a wide range of disability conditions and improve the overall experience for everyone who uses your site.

Structure

Your first set of testing items should be the way the site has been crafted. This includes your basic HTML, heading elements, lists, tables, and forms. These items should all be checked for correct usage and consistancy

Structure – Controls

- [ ] **Use the '<a href="...">' element for links**

  Any link that does not have the 'href' attribute inside the 'a' element will not be properly exposed to assistive technology, (for example a link that uses an 'onclick' event in place of an 'href' attribute). If using the a href is not an option, use the role="link" attribute on the 'a' element.

  Documentation

- [ ] **Ensure that links are recognizable**

  Colour alone is not sufficient to indicate the presence of a link. Visual cues like underlines are a universally effective way to communicate the presence of linked content.

  Documentation

- [ ] **Ensure all links have css ':focus' styling**

  Visual focus styles help assistive technology users determine which interactive element is currently focused.

  Documentation

- [ ] **Use the 'button' element for buttons**

  Using html elements other than a button element can be tricky for assistive technology. They must have extra code added to ensure that the element is properly accessible. If you have used something other than a button element, ensure that tests show that this is accessible.

  Documentation

Structure – Controls

- ☐ **Provide a skip link (bypass block) and ensure that it is visible when focused**
  Skip links allow the assistive technology user to skip repeated areas like headers, main navigation, etc, and skip directly to the area they would like to go.
  Documentation

- ☐ **Clearly identify links that open in a new window**
  Assistive technology users can become disoriented when links don't act as they expect them to. Identifying the action of a link helps assistive technology users better understand how to use and navigate the site.
  Documentation (W3)
  Documentation (knowbility)

- ☐ **Make sure that buttons, links and label elements have unique, descriptive content.**
  Buttons and links that use the term "click here" or "read more" are not descriptive enough for assistive technology users. Using more descriptive terms such as "Download PDF file" or "Open the full article" give the user a better understanding of what the link will do.
  Documentation

Structure – Forms

- ☐ **All form inputs should be associated with a corresponding 'label' element with a 'for'/'id'**
  Labels describe the purpose of the field, and associating them using the 'for'/'id' attributes ensures the assistive technology can refer to the correct label when presenting the form field. This pairing guarantees the highest level of browser/ assistive technology support.
  Documentation (W3)
  Documentation (W3-tutorial)

- ☐ **Use 'fieldset' and 'legend' elements**
  'Fieldset' elements group related content to make forms easier to understand. 'Legend' element acts as a heading to identify the group.
  Documentation

- ☐ **Ensure required fields are indicated**
  There are a couple of ways to indicate that a field is required. The '*' alone is not an optimal way as most screen readers will just read this as "star". Changing the field label to "Field Name (required):" or using an image for the '*' and placing an alt="required field" on the image are two of the most common ways to do this.
  Documentation

- ☐ **Check that form error messages are accessible**
  Test your form to ensure that on submit with error, any field errors are read out to users so that visually impared persons know where the issues are.
  - Fill out your form with errors

Structure – Forms

- Any field that has an error should have a descriptive error message on or around the field.
- Focus should be placed either on a list of errors at the top of the form, or on the top-most field in the form that has an error. Ensure that this error is able to be read out by assistive technology.

Documentation (W3)

Documentation (Medium)

☐ **Errors, Warnings, and Success states should not be communicated solely by color.**
Visually impared persons will need more than colour to identify these states. Using color, images, and text ensures that the message will be communicated to all.

Documentation

Structure – Global Code

- ☐ **Validate your HTML**
  Having error free HTML in your website will load faster and help to give users a consistent experience across all browsers and assistive technologies.
  Documentation

- ☐ **Using a 'lang' attribute on the html element**
  Setting the 'lang' (or language) attribute sets the language on screen readers. This setting sets the proper word pronunciation and braille language characters correctly.
  Documentation

- ☐ **Provide a unique title for each page or view**
  Ensure that the "title" element, contained in the document's "head" element is unique and descriptive. The title should clearly announce the basic theme or content of the page. This is also an important place for your SEO. Ensure that it matches and supports the content of the site.
  Documentation

- ☐ **Ensure that viewport zoom is not disabled**
  It is essential that users be allowed to zoom in or out of a site to customize their view to their specific vision needs. You should never disable this feature, even for app-like sites.
  Documentation

Structure – Global Code

- ☐ **Use Landmark elements to indicate important content regions**
  HTML5 landmark elements such as <nav> for navigation areas and <main> for the primary content area communicate the layout and allow the user quick access to these regions. You can also use ARIA landmarks to identify page regions and enhance semantics.
  Documentation (w3)
  Documentation (w3-tutorial)

- ☐ **Avoid the use of "autofocus" on your forms**
  HTML 5 introduced an autofocus attribute for webform controls. The autofocus attribute sets focus on the given field as soon as the page loads. Doing this allows the user to begin typing immediately. But for anyone that uses an assistive device, autofocus can dump them into a form with little to no context.
  Documentation (w3)
  Documentation (Medium)

- ☐ **Do not use session timeouts unless you can notify the user**
  If you must have session timeouts, you need to let the user know the timeout exists ahead of time. They also require significant notice before the timer runs out.
  Documentation

Structure – Global Code

- [ ] **Remove ‘title’ attribute tooltips**

  The title attribute is problematic for:

  - Touch-only devices
  - Navigating by keyboard
  - Using assistive technology
  - Users with motor control impairment
  - Users with cognitive concerns

  Documentation

Structure – Headings

- [ ] **Use heading elements (h1, h2, etc.) to introduce content**

  Organizing a page using heading structure (<h#>) helps users get a sense of the page's organization. Screen readers can allow a users to navigate a page according to it's headings. They can listen to a list of all headings and skip to a desired heading to begin reading at that point. Heading elements should not be used for purely visual design.

  Documentation

- [ ] **Every page should have one <h1> – and no more than one per page**

  The <h1> element should be used to communicate the main title of the page. The <h1> element should not be used for a heading that doesn't change from page to page.

  Documentation

- [ ] **Keep the heading elements in logical sequence**

  Do not skip heading levels. (eg. from <h2> to <h6>). The order of heading elements should descend incrementally. This can cause confusion for assistive technology users.

  Documentation

- [ ] **Don't use bold instead of a heading**

  Use of bold instead of a heading tag will cause screen readers to ignore the text when reading out headings.

  Documentation

Structure – Lists

- ☐ **List content should be contained within ol, ul, and dl elements**

  Proper structure around lists allows assistive technology users to know that the section of content is related.

  Documentation

Structure – Tables

- ☐ **Use the table element for tabular data**

  The structure of a table helps assistive technology users understand the relationship of all the information.

  Documentation

- ☐ **Use the <th> element for table headers**

  Using proper semantic structure for tables helps create a logical relationship in grids for the assistive technology user. For tables that have two headers or irregular headers see the reference link below for more information on markup techniques.

  Documentation

- ☐ **Tables should have a caption element**

  The caption element identifies the overall topic of the table.

  Documentation

Content

How you organize, and deliver your content will either help or hinder disabled users. You should be aware of the following items to help ensure all users can effectively access your content.

Content – Animation

- ☐ **Ensure animation does not cause seizures**
  Strobing or flashing more than 3 times per second can cause seizures or migraines. It can also be distracting and disruptive for persons with cognitive concerns.
  Documentation

- ☐ **Provide a mechanism to pause or hide background video**
  Background video or animation can be distracting, especially if the content is placed over it.
  Documentation

- ☐ **Make sure all animation obeys the 'prefers-reduced-motion' media query**
  Animation can be distracting or disruptive to certain persons. If "reduce motion" setting is activated within the browser, ensure that the site animation is either stopped, removed, or if it is needed to convey meaning, the animation speed is slowed significantly.
  Documentation

Content – Images

- ☐ **All images must have an 'alt' attribute**

  An "alt" or alternative text attribute is text added programmatically that describes the image.
  This attribute has two main functions. Assistive technology uses this text to communicate to the user what is on screen. As well, this is a great attribute for adding SEO keywords. If you are using this as a keyword area, ensure that the text is still descriptive of the image and does not interfere with the website experience if using a screen reader. (for example, if the image is of a house for sale, avoid spammy attempts at keyword stuffing like listing words such as; "sale, house, family". Instead work them into the description "House for sale, perfect for a family"). Decorative images do need an alt tag, but the description can be empty.

  Documentation (w3)

  Documentation (w3 – Tutorial)

  Documentation (Moz)

- ☐ **Provide text alternatives for complex images**

  Complex images are images such as charts, graphs, infographics that contain a lot of text in image form and cannot be read out by a screen reader. Complex images can also be difficult to understand by people with learning disabilities or low vision.
  Most screen readers cut off alt text around 125

Content – Images

characters. This means for complex images we need to link a page or document that fully explains the image textually. Use the longdesc="" attribute to specify a hyperlink to a detailed description of an image.

Documentation (w3)

Documentation (Moz)

- ☐ **Avoid using text images**

  Text as an image cannot be read out to users. If you must use an image of text - for example an image of a logo - ensure that the alt text mirrors the text within the image.

Content – Media

- ☐ **Audio and video should not autoplay**

  Unexpected video or audio can be distracting, disruptive, especially for certain kinds of cognitive disabilities. Certain kinds of autoplaying video and animation can also trigger seizures and migraines.

  Documentation

- ☐ **Ensure that media controls have appropriate markup**

  Proper labeling of audio and video controls allow assistive devices to properly convey functionality to the user.

  Documentation

- ☐ **Check media controls**

  If you have provided a global pause function, ensure that the space bar does not cause media playback. Also ensure that this global function does not interfere with the space key's ability to scroll the page when not focusing on a form control.
  Check that you are able to control play, stop, pause, etc. all by use of your keyboard. Ensure that the user cannot get 'trapped' within the interactive item.

- ☐ **Videos with any audio must have closed captioning**

  There are actually quite a few good reasons to always have closed captioning on videos.

  - Many users will watch videos in office, with the

Content – Media

sound off.

- Deaf users are given the ability to follow and understand the video.
- Captions add another dimension to SEO. The text helps search engines categorize and add your videos to search results - and videos with closed captioning rank higher in search results.

Documentation (W3)

Documentation (3PlayMedia)

☐ **Check for, and remove seizure triggers**

Avoid flashing items 3 times per second or more. Strobing and flashing animations may trigger seizures.

Documentation

☐ **Ensure audio transcripts are available**

Just like video, there is more than one reason to add transcript links to your audio clips.

- The user may be in the office or may just prefer to read rather than listen to the information.
- Allows deaf users access to the information contained within the audio clip.
- Increases your SEO and makes the topic and information searchable.

Documentation (W3)

Documentation (Moz)

Content – SVG

- [ ] **SVG elements should have role="img"**

  VoiceOver on macOS and iOS will not correctly convey the element as an image if role="img" has not been added to the SVG. This code also stops browsers from traversing to the SVG.

  If your SVG is not interactive but is placed inside an interactive item or other focusable element, use the focusable="false" attribute to ensure that the SVG does not inherit focusability.

  Documentation

- [ ] **SVG must have accessible text**

  SVG images don't have alt text, but they do have accessible alternatives.

  The title tag provides an accessible title for the element that contains it. The <desc> or description tag provides a longer more complete description of the element.

  For decorative SVG's adding aria-hidden="true" is the equivalent to an empty image 'alt' tag

  It's important to note that SVG titles and descriptions will need to have unique id's added and then referenced by aria-labelledby="" on the SVG.

  Documentation (W3)

  Documentation (css tricks)

Content – Text

- ☐ **Use plain language and avoid figures of speech, idioms, and complicated metaphors**
  When your content is for a general audience, it's important to keep your content at an 8th Grade reading level. Any advanced terms or language should be linked to an explanation or have a quick explanation in parenthesis.
  Documentation (W3)
  Documentation (Datayze)

- ☐ **Use left-aligned text for left-to-right languages and right-aligned for right-to-left languages and centered text should be used sparingly.**
  While this doesn't affect anyone using a screen reader, some users will find that center aligned text and justified text can be difficult to read.
  Documentation

Visual

These rules add visual enhancements to allow for better accessibility.

Content – Color Contrast

- ☐ **14-18pt text should have a minimum contrast of 4.5:1**

  Smaller text needs more contrast in order for users with minor visual impairments or color blindness to be able to read. Thinner fonts may need even more contrast. Consider using different applications that simulate visual impairments to test your site.

  Documentation (W3)

  Documentation (Contrast Analyser)

  Documentation (Color Safe)

- ☐ **18pt or larger text should have a minimum contrast of 3:1**

  While larger text can have less contrast then smaller text because of its size, if you are using a thin version of the font, you may still want to stick with 4.5:1 contrast. Thick or bold heading fonts with the 3:1 contrast will be highly readable.

  Documentation (W3)

  Documentation (Contrast Analyser)

  Documentation (Color Safe)

- ☐ **Icons should have a minimum contrast of 3.0:1**

  Even though icons are thought of being used mostly for decoration, they still add information to the visual interpretation of the site. A contrast of 3.0:1 will be sufficient for most icons, but if you are using an icon or graphic with thin lines, opt for an even deeper contrast to enhance readability.

  Documentation (W3)

  Documentation (Contrast Analyser)

Content – Color Contrast

- ☐ **Input field borders should have a minimum contrast of 3.0:1**
  It can be very frustrating to users if they cannot see the edges of the input form fields. Users with low vision using a mouse or touchscreen are unsure where to click/touch. Increasing the border contrast of these areas helps avoid unwanted selections and confusion.
  Documentation (W3)
  Documentation (Contrast Analyser)

- ☐ **Check the contrast of any text that overlaps an image or video**
  If text is on top of an image or video, is the text still legible? You may need to darken the area behind the text with a semi-transparent color, or darken / thicken the text. If this area is responsive, test it on multiple devices sizes to ensure that wherever the text lands over the image it is still legible.
  Documentation

- ☐ **Check selection colour contrast**
  Some people may highlight sections of your site. If your text is highlighted or selected, is the contrast still sufficient to read?
  Documentation

Content – Focus Indicator

- [ ] **Focus states are not hidden**

  A focus state allows for a visual cue so that a person navigating with a keyboard, switch or screen reader can see where they currently are on a page. All browsers have a focus state coded into them. Turning these off might make a website look cleaner, but users who use keyboard navigation or assistive devices then have no idea where they are on the site.

  Documentation

- [ ] **Check keyboard focus order & ensure all interactive elements receive focus**

  The focus order should match the visual layout of the page so that it's easy to navigate using a keyboard or assistive technology. Can a person navigating with a keyboard or screen reader move around the page in a predictable way? If for example - using the tab key - to navigate causes the user to skip from top to bottom and then back up again, it will be confusing.

  Documentation (W3 – Structure Separation)

  Documentation (W3 – Focus Order)

- [ ] **Remove focus on invisible elements**

  At times an element may receive focus that is invisible - which is confusing to assistive technology users. You should remove the ability to focus on an element that is not meant to be discoverable (ie: inactive dropdown menus, off screen navigations or modals). Use tabindex="-1" to disable focus.

  Documentation

Device Testing

Testing on multiple devices is EXTREMELY important to ensure that your users receive a consistant experience across all mediums. This is just as important in accessibility. You should test on different devices using programs that simulate different ability levels and disabilities.

Device Testing – Browser Testing

- [ ] **Test all major browsers**

  Ensure that you test in all major browers, Chrome, Internet Explorer, Edge, Safari, and Firefox. Check that the layout and interactive elements (buttons, forms etc) look and function consistantly across these browsers.

- [ ] **Navigate by keyboard only**

  This testing will allow you to check to make sure a person using keyboard-only or other assistive technology can effectively navigate your website.

  Documentation

- [ ] **Inverted color test**

  Inverting the colours allow you to make sure that your site is still readable with a serious visual impairment.

  Chrome Extension

  FireFox Extension

  Apple Devices

  IE and Edge Browsers

- [ ] **Greyscale test**

  This allows you to ensure no interactive items use color only to mark them. Carefully check that you are able to recognize buttons and other items without color.

  Chrome Extensions

  FireFox Extension

  Apple Devices

Device Testing – Browser Testing

- ☐ **Turn images off**

  Turn images off and make sure that the Alternative text (Alt) on images will communicate what the images are to blind users.

  Disable Images in Chrome

  Disable Images in IE 11

  Disable Images in Firefox

  Disable Images in Safari

- ☐ **High contrast**

  Check the site in high contrast mode to ensure that someone with low vision is still able to read all sections of your site. Pay particular attention to any text on top of images.

  Chrome Extensions

  Firefox

  Apple Devices

- ☐ **Zoom to 200%**

  Expect that users WILL zoom into your site. The site should be readable and functional when zoomed to 200%. Test to see if content overlaps or if any content becomes unreadable.

  Documentation

- ☐ **Spacing between elements is maintained on different screens.**

  Check your site on different device sizes, browsers and at different zoom levels to ensure that adequate spacing is still maintained between elements.

  Documentation

Device Testing – Mobile/Touch

- [ ] **Can the site be rotated to any orientation?**
  You cannot control what orientation or device your site will be used on. You need to ensure that your site design stays consistent and readable on any size, on any device.
  Documentation

- [ ] **Remove horizontal scrolling**
  Horizontal scrolling can be difficult for some users. Unless this is a specific and explained function of the site, do not use horizontal scrolling.
  Documentation

- [ ] **Check that buttons and links can be activated easily**
  Buttons that don't trigger easily can be frustrating and confusing to some users. Ensure that items like hamburger menus, social icons, and any other interactive item can be easily triggered by a wide range of hand sizes or stylus devices.
  Documentation

- [ ] **Ensure sufficient space between clickable items**
  Anyone with large fingers or hand tremors could have a difficult time navigating or scrolling past clickable items that don't have adequate spacing.
  Documentation

Call to Action

STRUCTURE

CONTROLS

- ☐ Use ‘a href’ elements for links
- ☐ Links are recognizable
- ☐ Links have visible focus states
- ☐ ‘Button’ element is used for buttons
- ☐ Skip link has been added
- ☐ New window Links are marked
- ☐ Buttons & links have descriptive content

FORMS

- ☐ All inputs have ‘label’
- ☐ All inputs/Labels have ‘for’/’id’ values
- ☐ Fieldsets & legends used
- ☐ Autocomplete is allowed
- ☐ Required fields are indicated
- ☐ Input errors are accessible to screen readers
- ☐ Error states not communicated solely by color

GLOBAL CODE

- ☐ Validate HTML
- ☐ ‘lang’ attribute is on html element
- ☐ Each page has a unique title
- ☐ Viewport zoom is NOT disabled
- ☐ Landmarks used to indicate important content regions
- ☐ Autofocus is NOT used
- ☐ Session timeouts are NOT used or user is notified ahead of time.
- ☐ No ‘title’ attribute tooltips

HEADINGS

- ☐ Heading Elements are used to introduce content
- ☐ Only one ‘h1’ per page
- ☐ Heading levels are not skipped
- ☐ Bold text has not used as a heading

LISTS

- ☐ List content is within ‘ol’, ‘ul’, or ‘dl’

TABLES

- ☐ Tables have been used for all tabular data
- ☐ Headings are marked with <th>
- ☐ Table Caption is used

CONTENT

ANIMATION

- ☐ No seizure triggers found
- ☐ Can be paused or hidden
- ☐ Honors reduced-motion setting

IMAGES

- ☐ All images have ‘alt’ tags
- ☐ Complex images have text alternatives
- ☐ Text images have the text mirrored in ‘alt’ tag

MEDIA
☐ No autoplay
☐ Media controls have assistive markup
☐ Media controls have been tested
☐ Videos have closed captioning
☐ No seizure triggers found
☐ New window Links are marked
☐ Audio tracks have transcripts

SVG's
☐ Have title or description
☐ SVG's images have role="img"

TEXT
☐ Use of plain language
☐ Centered text & right aligned text is used sparingly

VISUAL

COLOR CONTRAST
☐ 14-18pt text has 4.5:1 minimum
☐ 18pt + text has 3:1 minimum
☐ Icons have3.0:1 minimum
☐ Borders have 3.0:1 minimum
☐ Check contrast of text over images / videos
☐ Check highlighted text contrast

FOCUS INDICATOR
☐ Focus states are not hidden
☐ Keyboard focus order is logical
☐ Invisible elements don't get focus

DEVICE TESTING

BROWSER TESTS
☐ Test all major browsers
☐ Navigate by keyboard only
☐ Inverted color test
☐ Greyscale test
☐ Turn images off
☐ High contrast
☐ Zoom to 200%
☐ Spacing between elements is maintained on different screens.

MOBILE/TOUCH
☐ Check rotation
☐ No horizontal scrolling
☐ Links are easily activated
☐ Sufficient space between links