Prototyping

When creating wireframes & prototypes:

- ✓ **Use large elements to help users with low vision and/or limited mobility**
- ✓ **Keep other aspects of accessibility (high contrast, large click targets, etc.) in mind when creating mockups**
- ✓ **Simplify content and functionality where possible**

Start early: many accessibility features are **easily implemented** if they are planned from the beginning of website development.

</> Initial Development

Make magnification easy

- ✓ **Set type using relative units (e.g. ems) rather than explicit units (e.g. px)**

Make text accessible

- ✓ **Avoid using scrolling or flashing text**
- ✓ **Avoid putting text in images when possible**

Optimize compatibility with screen readers

- ✓ **Continue to check the site's HTML structure, without CSS applied**
- ✓ **For all `<img>` tags, provide a caption (using `<figure>` and `<figcaption>` tags) *or* a populated `alt` attribute**
  - **Leave the `alt` attribute blank if using a caption—don't use both.**
  - **Focus on content and function rather than just a visual description of the image.**
- ✓ **Ensure the `for` attributes in all `<label>` tags have the same value as the `ID` attribute for the inputs they are associated with**
- ✓ **Keep inputs in the proper order**
  - If inputs must be placed out of order in the HTML, use the `tabindex` attribute to correct the order.
- ✓ **Optimize tables:**
  - Provide a `<caption>` tag to explain the purpose of the table.
  - Summarize tables using the `summary` attribute.
  - Use `<th>` tags for all headers, and use the `scope` attribute to indicate column headers vs row headers.

Optimizing HTML for screen reader compatibility has added benefits of making the HTML more semantic and more usable for *everyone*.

The way the site is visually presented with CSS applied may not be the same order that the screen reader follows, which is why continuing to check the HTML structure is important.

Ensure all inputs on the page are accessible with just the keyboard

- ✓ **Look out for instances where keyboard control isn't available by default**
  - **Examples: drag and drop interfaces, custom HTML elements**
- ✓ **Provide keyboard shortcuts**
- ✓ **Keep `:hover` and `:focus` CSS pseudoclasses identical**

Non-standard HTML elements can be made accessible with ARIA tags.

Make completing tasks easy for the physically impaired

- ✓ **Allow enough time to complete tasks**
- ✓ **Minimize the number of steps needed to complete a task**
- ✓ **Keep click areas large for links and buttons**
  - Add padding around link text

Refining

Check the site for contrast

- ✓ **View the site in Grayscale**

  - Adjust screen settings or import a screenshot into Photoshop

Red/green color blindness affects about **7%** of the male population.

- ✓ **Look at a blurry version of the website**

Make magnification easy

- ✓ **Set type size using relative units (ems) rather than absolute units (px)**

Avoid putting text in images

- ✓ **Create charts and graphs in HTML rather than images where possible.**

Make audio accessible

- ✓ **Provide transcripts for spoken content and subtitles for videos**
- ✓ **Keep volume levels consistent across all audio content**

Use an accessibility checker

- ✓ **tota11y - an accessibility visualization toolkit**
  - http://khan.github.io/tota11y/