

# Automated Accessibility Inspection

[www.wegmans.com](http://www.wegmans.com)

Josh Hochman

# 1 Evaluation Results

## 1.1 Automated Tool Details

For my Automated Accessibility Inspection, I have chosen to analyze the website for the Wegmans supermarket chain using the ANDI Accessibility Testing Tool.

My primary focus is on the homepage found at [www.wegmans.com](http://www.wegmans.com). Although this is the same page examined for my Manual Accessibility Inspection, some elements of the web page have been modified by the developers since my first review. Notably, the supermarket products being advertised have shifted in focus from the Thanksgiving holiday to Hanukkah. Therefore, some accessibility problems specifically related to webpage elements for newer products may not have been present during the initial review.

I opted to test the website using version 119.0.6045.200 (64-bit) of the Google Chrome web browser.

## 1.2 Automated Accessibility Evaluation Results

### Violation #1

|  |  |
|--|--|
| <p>WCAG<br/>Guideline:<br/>[2.4.3 (A)]</p> | <p>§ <b>Success Criterion 2.4.3 Focus Order</b></p> <p>(Level A)</p> <p>If a <a href="#">Web page</a> can be <a href="#">navigated sequentially</a> and the navigation sequences affect meaning or operation, focusable components receive focus in an order that preserves meaning and operability.</p> <p>(Campbell et al., 2023)</p> <div data-bbox="1127 1199 1377 1268" style="border: 1px solid black; padding: 2px;"> <p><a href="#">Understanding Focus Order</a><br/><a href="#">How to Meet Focus Order</a></p> </div> |
| <p>Type of<br/>Violation</p>               | <p><b>False Keyboard Access Alert</b></p>  |

[Page of Violation] > [Location on Page]

Pick Your Perfect Ham & Beef Roasts SHOP ALL

Wegmans Ready to Cook Garlic Studded Beef Rib Roast (Boneless)  
\$78.24 /ea  
\$18.99/lb  
Meat Department  
★★★★★ 2

Wegmans Choice Angus Beef Bone-In Rib Roast, 3-4 Ribs, Raised without Antibiotics  
\$218.88 /ea  
\$27.99/lb  
Meat Department  
★★★★★

Wegmans Choice Angus Beef Tenderloin Roast, Trimmed & Tied  
\$184.43 /ea  
\$42.99/lb  
Meat Department  
★★★★★

Wegmans Bone-In Celebration Ham  
\$66.14 /ea  
\$4.49/lb  
Meat Department  
★★★★★ 29

Wegmans Boneless Celebration Ham  
\$54.06 /ea  
\$6.49/lb  
Meat Department  
★★★★★ 14

```

<button aria-hidden="true" tabindex="-1" class="css-1gczm2n ANDI598-element ANDI598-highlight ANDI598-element-active" data-andi598-index="43">
  <div class="css-114w6pd">
    
  </div>
</button>
  
```

**ANDI** focusable elements tab order title attributes label tags

Element: <button>  
Accessibility Components: 4  
group: Product  
aria-hidden: true  
tabindex: -1  
max-clip: alt: Wegmans Choice Angus Beef Bone-In Rib Roast, 3-4 Ribs, Raised without Antibiotics  
ANDI Output:  
Product Wegmans Choice Angus Beef Bone-In Rib Roast, 3-4 Ribs, Raised without Antibiotics  
▲ Element is focusable but has or is contained by [aria-hidden=true] ▲ Focusable element is not in keyboard tab order; should it be tabbable?

Focusable Elements Found: 174  
Accessibility Alerts: 75  
Keyboard Access Alerts: (21)  
1. ▲ Focusable element is not in keyboard tab order; should it be tabbable?  
2. ▲ Focusable element is not in keyboard tab order; should it be tabbable?

Description of Violation:

According to WCAG Success Criterion 2.4.3, web pages must be designed in such a way that a user can navigate between elements using the TAB key (Campbell et al., 2023). This method of navigation must serve as an effective substitute for users that cannot use a mouse. To meet this criterion, elements must be coded in such a way that pressing the TAB key results in the focus moving between elements in a logical order.

Upon inspection of [wegmans.com](http://wegmans.com), the fANDI tool generated multiple instances of the following warning:

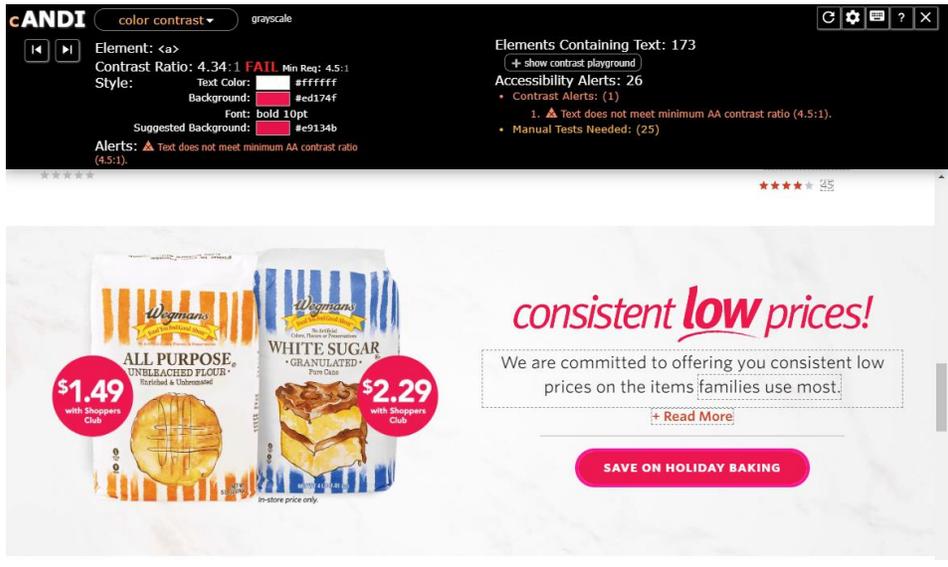
**“Focusable element is not in keyboard tab order; should it be tabbable?”**

According to the help contents for ANDI (Social Security Administration Accessible Solutions Branch, 2023), this warning occurs when the tool detects elements on the webpage with a negative tabindex value. I investigated the issue by first examining which elements on the page were generating the error; most of the problems were resulting from the product carousels, or more specifically, the *images* of products in the carousels. A typical mouse user exploring [wegmans.com](http://wegmans.com) might want to learn more about one of these products either by (a) clicking on the product or (b) clicking on the descriptive text of the product under its picture. The “tabindex” attribute controls which elements can be

|  |   |
|--|---|
|  | <p>selected with the TAB button and in what order the focus moves as the user repeatedly presses the button. At first, I thought the fANDI warning was suggesting that the tab order of page elements was incorrect (for example, navigating from item 1 to item 4 to item 2), or that certain elements were getting skipped (similar to the effect of “aria-hidden” observed in my manual inspection). I tested my theory by manually tabbing through every element on the page and found no issues with the order of elements. However, I noticed that for each item in the carousel, the descriptive text for the product could be TAB-focused (as could each product’s “+” (Add to Cart) button), but the image of the product could not. Each image was embedded within a &lt;button&gt;, and the tabindex property of the button had been manually set to “-1”. When setting tabindex, a value of 0 means the element can be reached with TAB, and a value of -1 means it cannot be reached with TAB. According to Developer.mozilla.org (MDN Contributors, 2023), a &lt;button&gt; has a default tabindex of 0, which means the developer intentionally made these elements not TAB-selectable.</p> <p>I would argue that this example constitutes a false alert. Although the images cannot be TAB-selected, this does not impede the functionality of the carousel, as users can still TAB to the product description, and both the image and product descriptions are hyperlinked to the same location. In this example, I believe the user’s inability to focus the selection on the button is actually the result of its “aria-hidden = true” property, not its “tabindex = -1” property.</p> |
|--|---|

Violation #2

|  |   |
|--|---|
| <p>WCAG<br/>Guideline:<br/>[1.4.3 (A)]</p> | <p>§ <b>Success Criterion 1.4.3 Contrast (Minimum)</b></p> <p>(Level AA)</p> <p>The visual presentation of <a href="#">text</a> and <a href="#">images of text</a> has a <a href="#">contrast ratio</a> of at least 4.5:1, except for the following:</p> <p><b>Large Text</b><br/><a href="#">Large-scale</a> text and images of large-scale text have a contrast ratio of at least 3:1;</p> <p><b>Incidental</b><br/>Text or images of text that are part of an inactive <a href="#">user interface component</a>, that are <a href="#">pure decoration</a>, that are not visible to anyone, or that are part of a picture that contains significant other visual content, have no contrast requirement.</p> <p><b>Logotypes</b><br/>Text that is part of a logo or brand name has no contrast requirement.</p> <p>(Campbell et al., 2023)</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin-left: auto; margin-right: auto;"> <p><a href="#">Understanding Contrast (Minimum)</a><br/><a href="#">How to Meet Contrast (Minimum)</a></p> </div> |
|--|---|

|  |   |
|--|---|
| <p>Type of Violation</p>                               | <p><b>Error and False Pass</b></p>  |
| <p>[Page of Violation] &gt;<br/>[Location on Page]</p> |    |
| <p>Description of Violation:</p>                       | <p>According to WCAG Success Criterion 1.4.3, colored elements containing text must meet a minimum contrast ratio of 4.5:1. In the above example, the color contrast between the pink “Save on Holiday Baking” button (background #ED174F) and the white text (#FFFFFF) is only 4.34:1, so the cANDI tool correctly identified the violation. However, to the left of the button, the website also displays several items using pink circles with white text to indicate the price with a Wegmans Shoppers Club card. These appear to also have the same color theme/color contrast. However, they were not detected by cANDI because the price labels were part of a larger image; cANDI stated that a manual contrast test would be required. I opened the image in a separate tab and analyzed the pink shade with the ColorZilla Chrome addon, which identified its code as #EE1651, which has a 4.31:1 color contrast with pure white. Therefore, these images would also fail the WCAG color contrast requirement, making this an example of a false pass.</p> <p>This violation was not detected during the manual inspection because the “Save on Holiday Baking” button was added to the website after the manual inspection was conducted. This element was part of the new</p> |

|  |  |
|--|--|
|  | site content introduced as part of Wegmans’ holiday-themed advertising campaign. |
|--|--|

Violation #3

|  |   |
|--|---|
| <p>WCAG<br/>Guideline:<br/>[1.1.1 (A)]</p>             | <p>§ <b>Success Criterion 1.1.1 Non-text Content</b></p> <p>(Level A)</p> <div style="border: 1px solid black; padding: 2px; width: fit-content; margin-left: auto;"> <p><a href="#">Understanding Non-text Content</a><br/><a href="#">How to Meet Non-text Content</a></p> </div> <p>All <a href="#">non-text content</a> that is presented to the user has a <a href="#">text alternative</a> that serves the equivalent purpose, except for the situations listed below.</p> <p><b>Controls, Input</b><br/>If non-text content is a control or accepts user input, then it has a <a href="#">name</a> that describes its purpose. (Refer to <a href="#">Success Criterion 4.1.2</a> for additional requirements for controls and content that accepts user input.)</p> <p><b>Time-Based Media</b><br/>If non-text content is time-based media, then text alternatives at least provide descriptive identification of the non-text content. (Refer to <a href="#">Guideline 1.2</a> for additional requirements for media.)</p> <p><b>Test</b><br/>If non-text content is a test or exercise that would be invalid if presented in <a href="#">text</a>, then text alternatives at least provide descriptive identification of the non-text content.</p> <p><b>Sensory</b><br/>If non-text content is primarily intended to create a <a href="#">specific sensory experience</a>, then text alternatives at least provide descriptive identification of the non-text content.</p> <p><b>CAPTCHA</b><br/>If the purpose of non-text content is to confirm that content is being accessed by a person rather than a computer, then text alternatives that identify and describe the purpose of the non-text content are provided, and alternative forms of CAPTCHA using output modes for different types of sensory perception are provided to accommodate different disabilities.</p> <p><b>Decoration, Formatting, Invisible</b><br/>If non-text content is <a href="#">pure decoration</a>, is used only for visual formatting, or is not presented to users, then it is implemented in a way that it can be ignored by <a href="#">assistive technology</a>.</p> <p>(Campbell et al., 2023)</p> |
| <p>Type of Violation</p>                               | <p><b>Error and False Alert</b> (though I believe this is debatable, as outlined below)</p>   |
| <p>[Page of Violation] &gt;<br/>[Location on Page]</p> |   |

The image shows two screenshots of the Wegmans website with the ANDI accessibility tool overlaid. The top screenshot shows a banner for 'Happy Hanukkah' with the text 'holiday menus for'. A red circle highlights the 'Happy Hanukkah' logo, and a red arrow points to the ANDI code for that image. The code is: ` == $0`. The bottom screenshot shows a product grid with various items. A red circle highlights a 'Give the Gift of Wegmans!' button, and a red arrow points to the ANDI code for that image. The code is: ` == $0`

Description of Violation:

WCAG criterion 1.1.1 includes several accessibility requirements specific to non-text components of a webpage, including images. Per the guideline, images must have alt text that can communicate the same information as the image itself, unless the image meets one or more exception criteria. One exception of particular note for this example focuses on the image's purpose: if the image is decorative, then it must be included in the webpage in a way so technologies such as screen readers will skip over it.

In this example, the Star of David image visible next to the “Happy Hanukkah” graphic generated an error from gANDI because it does not include an accessible name (identifier), alt text, or title. The tool suggests that resolution of this error depends on whether or not the image qualifies as “decorative”. Upon inspection of the HTML code for the graphic, we see that the <img> tag includes text defining the pixel size and position of the image and includes a “src” property containing the hyperlink to the image. However, the tag does not include any properties defining an identifier for the image (such as “id”, a global HTML attribute). Additionally, the tag lacks alt text (such as “Star of David representing Judaism”) and a title, another global HTML attribute. We can see the same violation repeated on the image of the gift card next to the “Give the Gift of Wegmans” text.

I believe that categorization of the gANDI warning for the star as an actual error or a false positive ultimately depends on the web designer’s intent for the image. If the goal was to create a decoration around the message of “Happy Hanukkah”, then it could be argued that this meets the exemption outlined in the “Decoration, Formatting, Invisible” caveat of 1.1.1. However, the Star of David has significant meaning to many people. If the goal was to enhance the user’s emotional state upon seeing the image (for example, to promote a feeling of inclusion among Jewish users, which may not be achieved through the “Happy Hanukkah” text alone), then one could argue that the image does serve a specific purpose. The problem is that the programmer did not include any code specifying the image as one or the other. For example, if the images were intended to be meaningful, the author could have added an explicit property such as “aria-hidden=false”. On the other hand, I think the image of the graphic of the gift card would be difficult to categorize as anything other than decorative. This means the gANDI warning for this element would be best described as a False Warning.

## 2 Top Three Accessibility Improvements

|                                 |  |
|---------------------------------|--|
| <p>Type of Improvement (#1)</p> | <p>Wegmans uses many promotional videos throughout the Careers section pages of their website. During the Manual Accessibility evaluation, I identified that several of these videos made captions available to viewers, but none of them appeared to offer descriptive text. This violates WCAG Success Criterion 1.2.4 Audio Description (Prerecorded), which requires “audio description for any prerecorded video” (Campbell et al., 2023).</p>  |
| <p>Improvement (#1)</p>         | <p>Descriptive text would increase the amount of information communicated through video to viewers using screen readers. As an example, the “Life at Wegmans: Customer Service” video found at the top of their Careers homepage showcases employees performing different types of duties in the store, such as baking bread, pulling shopping carts, or slicing prosciutto. The captions translate what the employees say in the videos into text, but most of this content features generic statements (such as “it’s a cool place to work”) that could be found in the marketing materials for any major company.</p> <p>I think including descriptive text in this context would enhance the experience of potential applicants who use screen reader technology. Most job seekers know that companies put their best foot forward in their promotional content, so they may look for clues on the company’s website about what the job is really like. For a person that cannot see the video, hearing only the statements in the captions would probably make the Wegmans environment seem generic and fail to distinguish it as a unique workplace. If the same user had the ability to imagine themselves baking bread or slicing prosciutto, it might enhance their interest in applying for a job.</p> |
| <p>Justification (#1)</p>       | <p>As with many other businesses, Wegmans undoubtedly wants to attract quality candidates for working roles in their stores. Adding descriptive text to their recruitment video content could potentially widen their applicant pool of visually impaired users by allowing these users to better imagine working in the store. By making their content more inclusive, Wegmans can potentially gain a recruitment edge over other potential employers for those individuals.</p>  |

|                                 |  |
|---------------------------------|--|
| <p>Type of Improvement (#2)</p> | <p>Many products showcased around Wegmans' site feature a linked submenu that includes specific product information such as ingredients, nutrition, and warnings. This menu can be accessed by selecting any product image or its associated text description. During the Manual Accessibility evaluation, I noticed that visually impaired web users may struggle to navigate this submenu at high zoom factors due to its improper reflow. According to WCAG Success Criterion 1.4.10, the content of any page should condense into a single column at a zoom factor of 400% so users don't need to use horizontal scrolling to see everything on the page (AG WG Participants, 2023). Unfortunately, not only do these detail menus require horizontal scrolling, but the generated scroll bar is also very thin and difficult to select with a cursor (either with a mouse or a touchpad).</p> |
| <p>Improvement (#2)</p>         | <p>According to the WCAG 2.2 Understanding Docs, developers can tackle the risk of reflow problems by employing methods of "responsive web design". These techniques use the page's CSS code to contain the page's content within the page's width, even as the user increases or decreases the zoom factor (AG WG Participants, 2023). If applied to the above product submenus on the Wegmans site, users would be able to view ingredients, product instructions, nutrition facts, and other pertinent information by scrolling in the vertical direction only. This would reduce the risk of users getting frustrated with having to constantly scroll in multiple directions or clicking and missing the narrow scroll bar.</p>   |
| <p>Justification (#2)</p>       | <p>Many users choose to increase the magnification of a webpage because they struggle to see its content at the 100% default zoom value; they expect that by changing the zoom level, reading will become easier, not harder. These users would likely feel jarred if increasing the magnification of the webpage reduced the amount of content they could easily see. Therefore, for visually impaired users, keeping content contained within a single column decreases cognitive load.</p>  |

|                                 |  |
|---------------------------------|--|
| <p>Type of Improvement (#3)</p> | <p>During the Manual Accessibility evaluation, parsing the wegmans.com homepage with the JAWS screen reader revealed that each product listing included a small grey box containing information about the product's physical location in the store, but this text was not focusable with the screen reader. According to WCAG Success Criterion 4.1.2, webpage components whose function has been intentionally altered through code modification must still be able to serve their intended purpose when a screen reader brings them into focus (AG WG Participants, 2023). For this example, the boxes had been modified by explicitly setting the "aria-hidden" property to "true".</p> <p>If a blind customer wanted to visit a grocery store, they might use assistive technology to prepare their shopping list ahead of time. However, as Heidi Joshi reveals in her interview with Will Butler (Butler, 2021), blind shoppers will typically still shop with another person that walks with them and helps them pick out items. This type of user may prefer to know where in the store to find the products they want prior to arriving at the store, rather than spending the extra time locating items with their assistant. By hiding the locations of supermarket items from a screen reader, the web designer has made this preparatory task impossible for this user.</p> |
| <p>Improvement (#3)</p>         | <p>To improve this process and the way the web page functions, I would create a dedicated level within the heading structure specifically for listing product location information. In the manual inspection, I identified that the Wegmans header structure seldom used &lt;h3&gt; level tags. If the descriptive text of a product was categorized as a &lt;h2&gt; tag, then the &lt;h3&gt; could be used for storing the location data. For example, instead of</p> <pre>&lt;h1&gt; Holiday Baking Essentials &lt;/h1&gt; &lt;h2&gt; Wegmans All Purpose Unbleached Flour &lt;/h2&gt; &lt;span aria-hidden="true"&gt; Aisle 12B &lt;/span&gt;</pre> <p>the programmer could instead use a method such as the following:</p> <pre>&lt;h1&gt; Holiday Baking Essentials &lt;/h1&gt; &lt;h2&gt; Wegmans All Purpose Unbleached Flour &lt;/h2&gt; &lt;h3&gt; Aisle 12B &lt;/h3&gt;</pre>  |

|                    |   |
|--------------------|---|
| Justification (#3) | Visually impaired users are likely to have a drastically different experience when grocery shopping at a Wegmans as compared to a user without visual impairment. However, all users should have equal access to the information and cues that make this process easier. Having the ability to know where in the store to locate an item benefits all users, and withholding this information from one user group can potentially make their shopping unnecessarily more difficult. |
|--------------------|---|

## 3 Methodological Reflection

### 3.1 Reflection upon Automated Accessibility Evaluation

One of the aspects I found most interesting about using an automated inspection tool was just how much it reminds me of running a compiler. Both ANDI and a compiler examine code for specific types of patterns that they recognize as errors, much like the way a spelling and grammar checker works in a word processor. We know that writing code requires a programmer to be extremely precise with logic. A computer cannot infer what a programmer wants the computer to do; it will only attempt to do exactly what the code tells it to do. In a similar fashion, it appears that tools such as ANDI are examining web pages and looking for specific things. However, compilers will stop your code from running when they think they have found a mistake. If ANDI can't infer whether an error actually exists, it generates a warning telling us to manually check for the error ourselves. In this regard, accessibility tools can be considered much more generous than compilers!

I also found it very interesting to see how there are limits to an automated checker's ability to detect color contrast violations. For some elements, such as simple buttons, the contrast is easy to check because there are only two colors to compare: the text color and the color of the background. It makes sense that the tool wouldn't be checking color contrast on an image that's made up of many different colored pixels. The computer would have to somehow calculate color ratios/scores for every pixel and aggregate those values into a single average score, and I can't see how such a value would be at all meaningful. However, as seen in Violation #2 above, problems can arise when simple images that fail the contrast test are merged into background images. To address this issue in the example of the price tags for the flour and sugar, I think developers need to be particularly conscious of the risks of combining individual images and with choosing which ones to set as backgrounds.

### 3.2 Comparison: Manual & Automated Accessibility Evaluations

One of my key takeaways from performing and comparing the manual and automated inspections is that each analysis revealed a variety of types of issues of which a person using a screen reader would likely not be aware. For example, with the aisle numbers hidden via the “aria-hidden” property, a blind user exploring the Wegmans site would have no way to know that that information is even available on the site. If a user doesn’t know that information is being hidden from them, they can’t know to request help with getting it or know that they’ll need to find a workaround on their own. In this example, a blind user would become more reliant on a sighted user to identify where to find products in the store. Users shouldn’t have to do extra work; pertinent information should be available to them from the beginning, even if they need to take extra steps to be able to use it.

Both the manual and automated inspections excelled at revealing gaps where page elements could not be selected. In fact, on multiple occasions, I found selection issues that I had missed in the manual inspection which were later detected in the automated inspection. Performing the manual inspection a second time revealed that the issues were visible via manual inspection all along! For example, during the manual inspection, I noticed that the aisle numbers couldn’t be tab-selected, but didn’t notice that the product photos couldn’t be selected either. This issue was caught during the automated inspection, and a follow-up manual inspection by tabbing through the elements revealed that it was a detectable problem.

Overall, both inspections reflected that the Wegmans site is reasonably well-designed from an accessibility perspective. Additionally, most of the issues analyzed could be fixed with relatively simple modifications, such as a few extra lines of code or simply tweaking a few words in the properties of select HTML elements.

### 3.3 Contrast: Manual vs. Automated Accessibility Evaluations

One of the interesting patterns that emerged in my findings from the manual inspection was a series of issues that could potentially harm a user’s experience through denial of useful or even essential information. For example, my manual inspection revealed that the text elements on a page were not properly organized into a hierarchy for a screen reader to interpret. If a user navigates Wegmans.com with a tool such as JAWS, they could potentially miss parts of the webpage because they weren’t organized into the correct content category. For example, if a page element had been grouped under the <h3> tag when it should have been coded as a <h2>, and the user only parsed the <h1> and <h2> elements, then that content could be inadvertently missed. In the Wegmans Careers section of the website, I identified that the lack of descriptive text in their marketing videos means that blind users may not be receiving the full scope of information that those videos share. In the product detail menus, I identified how a user exploring the webpage at 400% zoom might miss some of the information being displayed; they have to horizontally scroll to see it using a tiny scroll bar that’s difficult to see at that magnification level.

My findings in the automated inspection tended to reflect quality of life issues more than denials of information, but in most cases, workarounds were readily available. For example, ANDI flagged the hyperlinked product images in the carousels as not being tab-selectable, but a user could reach the same linked location by selecting the text located directly under the picture, and this text *was* tab-selectable. Similarly, a user having difficulty reading the pink button whose color contrast fell below the 3.5:1 ratio can move their cursor over the button to change the pink background to black, greatly increasing its readability.

**Sources:**

- Accessibility Guidelines Working Group (AG WG) Participants. (2023, December 7). *Understanding SC 1.2.4: Captions (Live) (Level AA)*. WCAG 2.2 Understanding Docs. <https://www.w3.org/WAI/WCAG22/Understanding/captions-live.html>
- Accessibility Guidelines Working Group (AG WG) Participants. (2023b, December 7). *Understanding SC 1.4.10: Reflow (Level AA)*. WCAG 2.2 Understanding Docs. <https://www.w3.org/WAI/WCAG22/Understanding/reflow.html>
- Accessibility Guidelines Working Group (AG WG) Participants. (2023, December 7). *Understanding SC 4.1.2: Name, Role, Value (Level A)*. WCAG 2.2 Understanding Docs. <https://www.w3.org/WAI/WCAG22/Understanding/name-role-value.html>
- Butler, W. (Host). (13 Apr 2021). *How Do Blind People Shop For Groceries?* [Audio podcast]. Retrieved from <https://www.bemyeyes.com/podcasts/how-do-blind-people-shop-for-groceries>
- Campbell, A., Adams, C., Montgomery, R. B., Cooper, M., & Kirkpatrick, A. (Eds.). (2023, December 7). *Web Content Accessibility Guidelines (WCAG) 2.2, W3C World Wide Web Consortium Recommendation 05 October 2023, Level A Success Criteria*. <https://www.w3.org/TR/WCAG22/>
- MDN Contributors. (2023, July 17). *tabindex*. MDN Web Docs. [https://developer.mozilla.org/en-US/docs/Web/HTML/Global\\_attributes/tabindex](https://developer.mozilla.org/en-US/docs/Web/HTML/Global_attributes/tabindex)
- Social Security Administration Accessible Solutions Branch. (2023, May 19). *Alert: Focusable Element Not In Tab Order*. ANDI Alerts. [https://www.ssa.gov/accessibility/andi/help/alerts.html?not\\_in\\_tab\\_order](https://www.ssa.gov/accessibility/andi/help/alerts.html?not_in_tab_order)
- Wegmans Food Markets. (2023). Wegmans.com. <https://www.wegmans.com/>