

Accessibility Audit
Prepared by Caitlin Bartholic
May 3rd, 2026

WCAG 2.2 Accessibility Audit

Website: [REDACTED] home page

Date: May 3rd, 2026

Reviewer: Caitlin Bartholic

Executive Summary

This report describes the conformance of [REDACTED] website with W3C's Web Content Accessibility Guidelines (WCAG 2.2). Through manual and evaluation tools several barriers were detected and specific supplementary issues were pinpointed for remediation.

Based on this evaluation, [REDACTED] website does not meet WCAG 2.2 Level AA. Barriers are listed in order of severity in the conformance summary below.

Evaluation Methods and Tools Used

1. Manual Evaluation based on [WebAIM's WCAG 2.2](#)
2. Keyboard Testing
3. VoiceOver Screen Reader
4. ARC Toolkit
5. WebAIM Contrast Checker
6. axe DevTools
7. WebAIM WAVE

Severity Ratings

Accessibility barriers are rated using a four-point severity scale.

1. **Critical:** Content or functionality is completely inaccessible to some users.
2. **Significant:** Content or functionality is very difficult for some users.
3. **Moderate:** Content or functionality may require additional time or effort for some users because accessibility or usability guidelines and best practices are not followed.
4. **Recommendation:** Content or functionality can be made more usable for everyone by aligning with best practices.

Global Considerations

WCAG Success Criterion: 1.1.1 Non-text Content [Level A]

Severity: Moderate

Remarks and Explanations: All SVGs that convey meaning need role="img" and a title or label. If it is purely decorative, it can be hidden from assistive technologies. The site logo and widget icons are missing this attribute, and some non-standard attributes are likely coming from the accessibility overlay.

WCAG Success Criterion: 1.3.1 Info and Relationships

Severity: Moderate

Remarks and Explanations: Some HTML heading elements have aria-level attributes that do not match the element's heading level. This is a barrier for screen reader users who rely on the screen reader to know the structure of the page. Headings may be announced differently on the site than in the screen reader Rotor.

WCAG Success Criterion: 2.4.6 Headings and Labels [Level AA]

Severity: Moderate

Remarks and Explanations: There are some empty and redundant "ghost" headings on the page: <h2></h2>. These were likely just spaces to change the visual layout of content but should be created with CSS padding or margin the surrounding div instead of using structural HTML tags. This can indicate to screen reader users that a section exists where there is no content.

WCAG 2.2 Conformance Summary

WCAG Success Criterion: 1.4.3 Contrast (Minimum) [Level AA]

Severity Level: Critical

Remarks and Explanations: Orange (EA8025) and white (FFFFFF) is a 2.75:1 Color Contrast, and the Orange on the darker menu strip (F7F7F7) is a 2.57:1 color contrast. This low color contrast is a barrier for people with low vision, and people who are viewing the site in bright light. Links in the bottom menu in their hover state match the background as both the background and link text is set to the same orange. In the carousel, the heading is a 3.09:1 contrast with the background. When using AccessiBe's vision impaired profile, it changes to 2.29:1 contrast.

Recommendation: Replace the orange text, and orange backgrounds/buttons with the new pink (D81B60). Be sure to make link text meet the color contrast requirement of 4.5:1 with the background. Darken the heading color in the carousel to match the darker headings elsewhere on the page.

WCAG Success Criterion: 2.4.7 Focus Visible [Level AA]

Severity Level: Critical

Remarks and Explanations: *This issue only happened sometimes. The inconsistency could be due to the overlay. More investigation needed to determine situational cause.* When navigating with keyboard, the visual focus indicator fails to render on the interactive containers (when the entire container is serving as a link instead of the link/button). There is a change of color for the hover state, but no visible focus indicator. The hover state is also inconsistent, sometimes showing two containers with active hover at once, and sometimes never activating. See video below for a demonstration.

Recommendation: The focus indicator should be visible on screen with any keyboard navigation, whether keyboard alone or with other assistive technology such as a screen reader. It should be consistent on all interactive elements on the site. The hover state is best practice for mouse users when hovering over a clickable block. (See recommendation for removing the link status from the entire container for accessibility tools).

WCAG Success Criterion: 1.3.1 Info and Relationships [Level A]

Severity Level: Critical

Remarks and Explanations: Slide 1 of the carousel excludes paragraph text from screen reader access. This is a barrier for screen reader users who have no way to access the text on slide 1.

Recommendation: Check the aria-hidden attributes, and ensure that the active slide doesn't have any CSS properties hiding it from accessibility tools. Ensure that aria-hidden="true" is removed from active slides. Watch video demonstrating screen reader interaction with the carousel. (Opens in YouTube).

Note: Even when AccessiBe's "Blind Users" function is activated, the paragraph text on slide 1 is not accessible to a screen reader.

WCAG Success Criterion: 4.1.2 Name, Role, Value [Level A], 2.4.4 Link Purpose (In Context) [Level A]

Severity Level: Critical

Remarks and Explanations: The previous and next buttons in the carousel are programmed as links instead of buttons, and they have no text or aria labels describing their function. This is a barrier to screen reader users who only hear "Button, group" and not the purpose of the button. (See above video demonstrating screen reader interaction with the carousel for more detail).

Recommendation: Replace anchor elements (<a>) with native HTML button elements (<button>). Assign a clear accessible name to each button using an aria-label. Ensure buttons are included in the tab order so keyboard users can navigate the carousel.

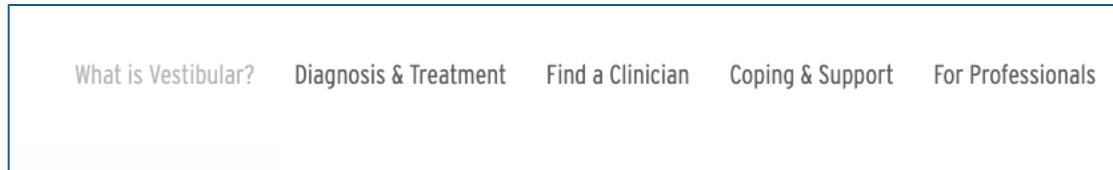
WCAG Success Criterion: 2.4.7 Focus Visible [Level AA]

Severity Level: Critical

Remarks and Explanations: *This issue only happened sometimes. The inconsistency could be due to the overlay. More investigation needed to determine situational cause.* Menu items at the top of the site do not have the native focus indicator that the rest of the page does. The text only changes color slightly in the hover/focus state.

Recommendation: Identify and resolve any CSS or JavaScript conflicts that are intermittently suppressing the existing focus indicator.

The image below shows the menu items at the top of the site. “What is Vestibular?” has keyboard focus and appears lighter than the other items, but there is no visual focus indicator around the text.



Note: Using AccessiBe’s “Keyboard Navigation (Motor)” function, these menu items do have a clear visible focus indicator.

WCAG Success Criterion: 2.1.1 Keyboard [Level A]

Severity Level: Critical

Remarks and Explanations: *This issue only happened sometimes. The inconsistency could be due to the overlay. More investigation needed to determine situational cause.* The previous and next arrows in the carousel will not activate with a keyboard while using a screen reader. They do respond to keyboard alone, but when the screen reader is activated, the buttons are not selectable. This is a barrier to everyone who uses screen readers.

Recommendation: Use native HTML <button> elements which should automatically handle mouse clicks, enter, and space keys without additional custom scripting.

Note: Using AccessiBe’s “Blind Users” function, the arrows are responsive with the screen reader.

WCAG Success Criterion: 4.1.2 Name, Role, Value [Level A], 1.3.1 Info and Relationships [Level A]

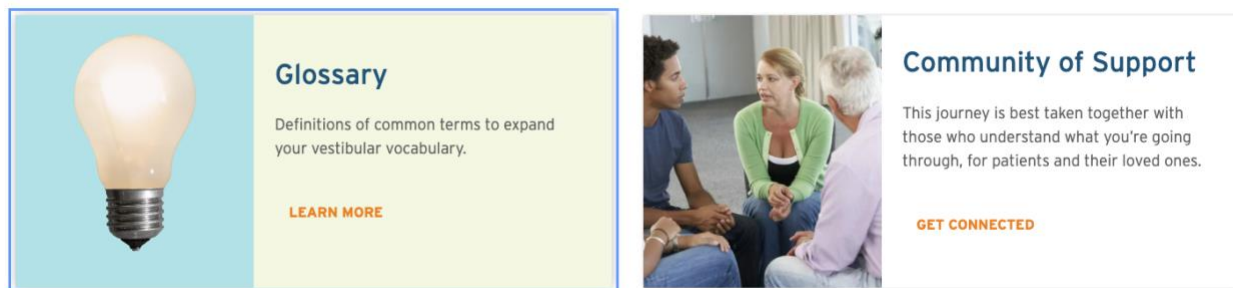
Severity Level: Significant

Remarks and Explanations: aria-hidden=”true” is used on elements that contain focusable child elements. There are multiple links in sections on the site that have a hover state and are selectable by mouse, but not by keyboard. When navigating with keyboard/screen reader, the entire block including heading and text and link are read as the link and selectable by the keyboard, receiving the visible focus indicator. This prevents keyboard users from seeing the focus indicator move specifically to the link. Furthermore, screen readers announce the entire block of text as one long link, preventing users from navigating efficiently by heading or individual link.

You can use AccessiBe's "Highlight Links" function to visually see how links are currently programmed. Watch video demonstrating how a screen reader interacts with these container links. (Opens in YouTube).

Recommendation: Headings and text should not be in the focus order. Change the focus to the actual link and make that selectable. The hover state the click-ability of the block for mouse users can remain the same to give mouse users a larger target. The AccessiBe Overlay may be causing the issue. Keep the Heading as a real <3> that is unhidden. Keep the text as <p> unhidden. Make the link focusable. Use stretched link CSS to keep the entire container clickable.

Below is one example of the focus indicator on the full container instead of the link.



WCAG Success Criterion: 1.3.1 Info and Relationships [Level A], 2.4.3 Focus Order [Level A], 4.1.2 Name, Role, Value [Level A]

Severity Level: Significant

Remarks and Explanations: Non-active element in tab order. In the carousel on the home page, the container holding the photo, heading, text, and button for each person's story is in the tab order (tabindex="0") with no assigned role (role="..."). This is a barrier for keyboard and other assistive technology users who will navigate to the container with a focus indicator, and not know that selecting it is the same as the "next" button. This is also a barrier for screen reader users who will have no indication where they are navigating or ended up. These same issues occurred when activating AccessiBe's Keyboard Navigation (Motor). [Watch video demonstrating the carousel keyboard accessibility issue.](#) (Opens in YouTube).

Recommendation: The container should be removed from the tab order, and only the buttons to read the story, previous, and next should be in the tab order. Slides that are not currently visible on screen should be hidden from the keyboard and screen readers. When the user selects the next button, the carousel should move to the next slide and remove the hidden attributes from the new slide, adding hidden attributes to the previous slide.

WCAG Success Criterion: 1.4.11 Non-Text Contrast [Level AA]

Severity Level: Significant

Remarks and Explanations: The orange button color on top of the light blue background is a 2.41:1 contrast, which does not meet the 3:1 contrast requirement. This is a barrier for users with low vision, color blindness, or viewing in a bright environment.

Recommendation: Changing the orange buttons to the new pink color, the buttons will meet contrast requirements at 4.32:1 contrast ratio. However, even in its hover state, the button must meet contrast requirements both with the background, and with the text on the button. The text must maintain a 4.5:1 contrast ratio against the button and hover state of the button, and the button must have a ratio of at least 3:1 against the background. A 2px solid border in a contrasting color can be added around the button if needed to meet requirements.

WCAG Success Criterion: 2.4.4 Link Purpose (In Context) [Level A]

Severity Level: Moderate

Remarks and Explanations: Many links (once the container link issues are resolved) have the same text. For example “More”. This is a barrier for screen reader users who tab through links on a website and won’t know the destination of the link they are selecting.

Recommendation: Change the link text to descriptive text. Descriptive text hyperlinks should be used anywhere they can for the benefit of clarity for all users, but if the link text needs to remain the same and makes sense in context for sighted users, an aria-label can be used to provide the necessary context to assistive technology. For example, `<a href="/symptoms" aria-label="Learn More About Common Symptoms">More</a>` For the same example, the “More” link could be changed to the descriptive text “Common Vestibular Symptoms”.