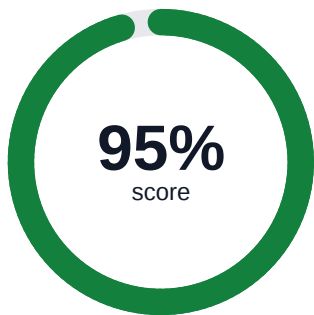




● Compliant

Your website is highly accessible

Your website meets WCAG accessibility standards, ensuring every visitor including those with disabilities can fully access your content and services.

9 Issues

0 Critical • 9 Moderate • 0 Minor

7 Auto-Fixed

Issues detected by category



Content

1

11.1% of total issues



Cognitive

1

11.1% of total issues



Low Vision

0

0.0% of total issues



Navigation

5

55.6% of total issues



Mobility

1

11.1% of total issues



Other

1

11.1% of total issues

ACCESSIBILITY COMPLIANCE ACHIEVED

Your website is now compliant with accessibility standards.

Compliant

- ✓ Continuous website protection by WebAbility.
- ✓ Real-time accessibility improvements.
- ✓ Meets WCAG 2.1 AA compliance standards.
- ✓ Maintains ongoing legal accessibility compliance.

Monitor Your Accessibility



Issues by Severity

Complete technical report, with affected elements and remediation steps.

Moderate issues: 9

Moderate Links that open in new tabs should indicate this to users WCAG 3.2.5 Need Action	
FIX	Add a clear visual indicator and screen reader-friendly text to the link. For example, append text like `(opens in new tab)` within a visually hidden span (`<span class="sr-only">(opens in new tab)</span>`) or use an icon with an appropriate `aria-label` or `title` attribute. Example: ` <a >(opens="" >link="" <span="" a>`.<="" class="sr-only" href="..." in="" new="" rel="noopener noreferrer" span><="" tab)<="" target="_blank" td="" text="">
CONTEXT	<code></code>

Moderate Ensure the document has at most one main landmark Uncategorized Need Action	
FIX	Review the page structure and identify all instances of the ` <main>` element. Consolidate the primary content into a single `<main>` element. If there are other significant content areas, consider using other appropriate landmark roles like `<section>` with an `aria-label` or `role="region"` if they are distinct and important enough to be navigable.</section></main></main>
CONTEXT	<code><main id="main-content"></code>

Moderate Ensure tabindex attribute values are not greater than 0 WCAG 2.4.3 Auto Fixed	
FIX	Remove the `tabindex="1"` attribute from the skip link. Elements that are naturally focusable (like links and buttons) do not need a `tabindex` value greater than 0. If an element needs to be focusable but isn't naturally, use `tabindex="0"`. For elements that should not be focusable, use `tabindex="-1"`.
CONTEXT	<code></code>

Moderate

Ensure all page content is contained by landmarks

Auto Fixed

WCAG 1.3.1

FIX	Identify the content that is not within a landmark. Wrap this content in the most semantically appropriate HTML5 landmark element or assign an ARIA landmark role. For example, if it's the primary content of the page, use <code><main></code> . If it's a navigation menu, use <code><nav></code> . Ensure all meaningful content is contained within a landmark.
CONTEXT	<code>Enhance accessibility audits & fixes with ABILYO</code>

Moderate

Ensure landmarks are unique

Auto Fixed

WCAG 1.3.1

FIX	For each landmark element that is not unique (e.g., multiple <code><nav></code> elements), add a unique <code>aria-label</code> attribute. For example, <code><nav aria-label="Main navigation"></code> and <code><nav aria-label="Footer navigation"></code> . This provides a clear, descriptive name for each landmark, allowing users to easily identify and navigate to specific sections.
CONTEXT	<code><main id="main-content"></code>

Moderate

Ensure the main landmark is at top level

Auto Fixed

WCAG 1.3.1

FIX	Review the HTML structure and ensure the <code><main></code> element is a direct child of the <code><body></code> element or another appropriate top-level container. Avoid nesting <code><main></code> within other landmark elements.
CONTEXT	<code><main class="min-h-screen bg-gray-50 p-8 lg:py-28 lg:p-16"></code>

Moderate

SVG elements should have accessible names

Auto Fixed

WCAG 1.1.1

FIX	Add an accessible name to the SVG element. You can do this by: 1. Using <code><title></code> and <code><desc></code> elements within the SVG, and referencing them with <code>aria-labelledby</code> on the <code><svg></code> . 2. Using <code>aria-label</code> directly on the <code><svg></code> element. 3. Using <code>aria-labelledby</code> to reference an existing visible text element.
CONTEXT	<code><svg class="asw-skip-icon" viewBox="0 0 28 28" fill="none" stroke="currentColor" stroke-width="2"><rect x="6" y="4" width="16" height="20" rx="2"></rect><line x1="10" y1="10" x2="18"</code>

Moderate Ensure all elements with a role attribute use a valid value Auto Fixed WCAG 4.1.2	
FIX	Remove the invalid <code>role="null"</code> attribute. If the element is a navigation landmark, use <code>role="navigation"</code> or simply use the <code><nav></code> HTML5 element, which implicitly has a navigation role. If the element does not require a specific ARIA role, remove the <code>role</code> attribute entirely.
CONTEXT	<code><nav class="transition-all bg-transparent duration-300 w-full" style="display:flex;padding:8px 16px;justify-content:space-between;align-items:center;height:57px" role="null" aria-label="Primary Navigation"></code>

Moderate Ensure role attribute has an appropriate value for the element Auto Fixed WCAG 4.1.2	
FIX	Remove the <code>role="navigation"</code> attribute from the <code><a></code> element. The <code><a></code> element, when used as a skip link, already has an implicit role of <code>link</code> . The <code>aria-label</code> is sufficient to describe its purpose. The <code>role</code> attribute should be reserved for cases where the semantic meaning of the HTML element needs to be augmented or overridden.
CONTEXT	<code></code>

Still Have Questions About Accessibility Compliance?

Talk to Our Accessibility Experts >