

# Accessibility Audit Report

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

<https://www.breakthroughpilates.com>

Technical Accessibility Findings & Remediation Report

Accessibility Risk Score

36

out of 100

Report Date: 2026-04-12  
Standard: WCAG 2.1 Level AA  
Prepared by: ADA Audit Report

Canonical audit: 5d1e2ece-4ae7-4afe-b060-a4bb1123a600  
Pages scanned: 25  
Audit completed: 2026-04-12 05:04 UTC

## Your review demo

Scan or visit the link below



[adaauditreport.com/  
accessibility-review-demo](http://adaauditreport.com/accessibility-review-demo)

key: 5d1e2ece

# Executive Summary



This audit identified 8 underlying accessibility issues affecting 28 instances across 25 pages on [www.breakthroughpilates.com](http://www.breakthroughpilates.com).

These are not independent failures — they are systemic issues originating from templates, components, or design tokens that propagate across multiple pages. Most sites resolve 70–90% of instances by fixing a small number of templates or components.

5 originate in site-controlled markup. 3 appear associated with third-party markup — verify ownership before assigning.

## Violations by Severity



Total: 28 violations across 8 rule categories

## Engineering Impact Summary

This audit identified 8 systemic issues. Most originate from shared templates and components, meaning a small number of targeted fixes will resolve the majority of instances.

### Estimated remediation scope:

- 3 component-level fixes — affects multiple pages
- 2 page-specific fixes — isolated to individual pages
- 3 vendor issues — requires vendor contact, not developer time

### Recommended execution order:

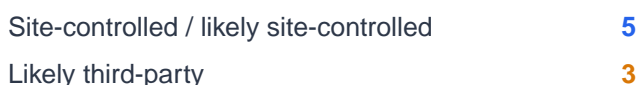
1. Fix template-level issues first (highest instance resolution per fix)
2. Fix component-level issues (shared across pages)
3. Address page-specific issues
4. Contact vendors for third-party issues

Key insight: You are fixing systems, not pages. Effort is significantly lower than instance counts suggest.

## Confidence Distribution



## Ownership Distribution



## Top Priority Findings

Some findings below are verified technical failures. Others are likely issues or require manual review. These categories are separated throughout the report.

### Verified high-priority findings (1):

- Elements must meet minimum color contrast ratio thresholds — 4 instances (SERIOUS)

### Likely findings requiring review (1):

- Links must have discernible text — 10 instances (SERIOUS, likely)

### Third-party / vendor-attributed findings (3):

- Buttons must have discernible text (YouTube Embed Player) — 2 instances
- Elements must only use permitted ARIA attributes (YouTube Embed Player) — 3 instances
- Frames must have an accessible name (YouTube Embed Player) — 1 instance

## How This Audit Was Produced

This report combines automated accessibility testing (axe-core engine) with rule-level adjudication of findings from the rendered page output. Each finding is classified by severity, confidence, ownership, and priority. Automated testing detects approximately 30-40% of WCAG issues. This report is a technical remediation document and should be used as a prioritized engineering worklist, not as a complete determination of accessibility compliance or a legal opinion.

Accessibility risk score: The score is a weighted technical risk indicator, not a legal determination of ADA compliance. Critical issues carry a 15-point penalty per instance, serious issues 8 points, moderate issues 3 points, and minor issues 1 point. 60% of the final score reflects the weighted penalty total; 40% reflects the pass/fail ratio across all automated checks.

### What This Report Is

- WCAG 2.1 AA-aligned technical audit
- Prioritized remediation plan with developer-ready fixes
- Ownership-classified findings (your code vs. third-party)

### What This Report Is Not

- Not a legal opinion or compliance certification
- Not a substitute for manual accessibility testing
- Not a guarantee of ADA compliance

## What You Need to Do

Issues are grouped into four action buckets so you know what to address first and who owns each fix.

**Fix Now** 2 issue categories (2 developer)

Verified failures that block access or comprehension for assistive technology users.

**Fix Next** 1 issue category (1 developer)

Confirmed structural issues or likely barriers that degrade usability but may not fully block access.

**Review Before Assigning** 2 issue categories (2 developer)

Findings that need manual verification, involve design tradeoffs, or have mixed confidence.

**Raise With Plugin / Vendor Provider** 3 issue categories (3 vendor)

Issues that appear associated with third-party or platform-generated markup.

# Section 1: 5 Site-Controlled Issues

These 5 issue categories were observed in site-controlled markup. Some may come from shared templates or components — verify in code before estimating scope. Each step includes the fix, what to avoid, and how to verify.

## Step 1 of 5: Link component lacks accessible naming

Your Developer

Likely Issue (High)

SERIOUS | 10 instances | Multiple pages (5)

**For owners:** Some links on your site have no accessible name (the text a screen reader announces). A screen reader user hears "link" but has no idea where it goes. The accessible name should come from visible link text, or from the alt text of an image inside the link. This is not the same as image alt text — this is about labeling a clickable link.

**For developers:** Likely issue: Give each link a clear accessible name. Preferred sources (in order): (1) visible text inside the <a> tag, (2) alt text on an <img> inside the link, (3) aria-label as a last resort when no visible text exists.

### Root cause

Origin: Links containing only icons, images with empty alt, or dynamically generated anchors without accessible text

**This is not 10 separate issues — this is one underlying problem propagating across 10 instances.**

Icon links and social media links are usually template-level. Product card links may be component-level.

WCAG: 2.4.4 | Level: AA | Priority: Must fix | Changes UI: No

Scope: Repeated across 5 pages — may be a shared component or template pattern.

Example from: <https://www.breakthroughpilates.com>

### Before (captured from your site):

```
<a href="/" id="1709627566" file="false">
```

Element is in tab order and does not have accessible text

Element does not have text that is visible to screen readers

aria-label attribute does not exist or is empty

aria-labelledby attribute does not exist, references elements that do not exist or references elements that are empty

Element has no title attribute

Affected pages: <https://www.breakthroughpilates.com>, <https://www.breakthroughpilates.com/contact>, <https://www.breakthroughpilates.com/workshops>, <https://www.breakthroughpilates.com/HostYourWorkshop>, <https://www.breakthroughpilates.com/FAQ>

### Impact:

When a screen reader user encounters the link and hears "link" with no indication of where it goes.

### Exceptions:

- Links that are not in tab order (tabindex="-1") or are visually hidden duplicates of accessible links may be acceptable
- Check if the link has a computed accessible name from nested content not visible in the HTML snippet

### Remediation:

Locate: Use your browser DevTools: right-click the link !' Inspect !' check the Accessibility tab for "Name." If empty, the link needs visible text or an aria-label.

Owner: Link accessible names come from the HTML structure. A developer must update the template code.

### After (fix pattern):

```
<!-- Replace with a label that describes where this link goes -->  
<a aria-label="Breakthroughpilates home" href="/" id="1709627566" file="false">
```

### Avoid this bad fix:

Do not generate aria-labels from URL paths or slugs. Do not add aria-label when visible link text already provides the name. Prefer visible text over hidden labels.

### How to verify:

DevTools: Inspect link !' Accessibility tab !' check "Name" field. Screen reader: Tab to link and verify announcement describes the destination.

### Expected outcome:

Screen readers announce link destinations. No visual change unless visible text is added.

## Step 2 of 5: Theme color tokens fail minimum contrast ratio

Your Developer

Confirmed (High)

SERIOUS | 4 instances | Multiple pages (4)

**For owners:** Some text on your website is hard to read because the text color is too similar to the background color. This affects people with low vision, older adults, and anyone in bright sunlight.

**For developers:** Update CSS color variables in your theme to meet WCAG 2.1 AA contrast ratio of 4.5:1 for normal text.

### Root cause

Origin: CSS color variables or inline styles in theme/template

**This is not 4 separate issues — this is one underlying problem propagating across 4 instances.**

Typically 2-4 color pairings in the theme CSS cause all instances. Fix the variables, not each element.

WCAG: 1.4.3 | Level: AA | Priority: Must fix | Changes UI: Yes

Scope: Repeated across 4 pages — may be a shared component or template pattern.

Example from: <https://www.breakthroughpilates.com/contact>

### Before (captured from your site):

```
<a href="https://breakthroughpilates.com/">Home</a>
```

Element has insufficient color contrast of 2.77 (foreground color: #ad0000, background color: #000000, font size: 10.5pt (14px), font weight: normal). Expected contrast ratio of 4.5:1

Affected pages: <https://www.breakthroughpilates.com/contact>, <https://www.breakthroughpilates.com/workshops>, <https://www.breakthroughpilates.com/HostYourWorkshop>, <https://www.breakthroughpilates.com/FAQ>

### Impact:

When text is difficult to read for users with low vision, aging eyes, or in bright ambient light.

### Exceptions:

- Inactive UI elements, placeholder text, and purely decorative text are exempt
- Large text (18pt+ or 14pt+ bold) only requires 3:1 ratio
- Logos and brand names have no contrast requirement

### Remediation:

Locate: Use your browser DevTools: right-click the text ! Inspect ! check Computed styles for "color" and "background-color." Adjust in your CSS.

Owner: Text color values are defined in your theme CSS. A developer must update the color variables.

### After (fix pattern):

```
<!-- Current: #ad0000 on #000000 = 2.77:1 ratio (requires 4.5:1) -->
<!-- Constraint: foreground color must meet 4.5:1 ratio against #000000 -->
<!-- Select a compliant value from your design system. Do not use arbitrary colors. -->
```

### Avoid this bad fix:

Do not suggest arbitrary replacement colors without checking they work with the site's design system. State the required ratio and let the developer choose compliant colors.

### How to verify:

DevTools: Use color contrast checker in Accessibility panel. Manual: verify text is readable at arm's length. Automated: re-run axe-core contrast check.

### Expected outcome:

Text becomes readable for users with low vision and in bright environments. No layout changes. Colors change to meet minimum contrast ratio.

## Step 3 of 5: In-text links distinguished only by color

Your Developer

Verify Manually (Med)

SERIOUS | 1 instance | Single page

**For owners:** A link inside a paragraph of text is only distinguished by color. Visitors who are color-blind can't tell what's a link and what isn't — they may miss important links entirely.

**For developers:** Verify first: Add a non-color visual indicator to links within text blocks, such as an underline (text-decoration: underline).

### Root cause

Origin: Links in text blocks distinguished only by color, without underline or other non-color indicator

CSS change — add text-decoration to links within paragraph context.

WCAG: 1.4.1 | Level: AA | Priority: Stakeholder review | Changes UI: Yes

Scope: Isolated to a single page.

Example from: <https://www.breakthroughpilates.com/HostYourWorkshop>

### Before (captured from your site):

```
<a href="https://breakthroughpilates.com/">Home</a>
```

The link has insufficient color contrast of 1.52:1 with the surrounding text. (Minimum contrast is 3:1, link text: #b8b8b8, surrounding text: #949494)

The link has no styling (such as underline) to distinguish it from the surrounding text

### Impact:

When color-blind users cannot identify links within paragraphs of text.

### Exceptions:

- If links have underline-on-hover, a border-bottom, or a distinct font weight/style visible in the rendered page, this may be a false positive
- The scanner checks computed styles but may miss hover/focus states

### Remediation:

Owner: This requires changes to the site code or theme templates.

### After (guidance):

Verify this issue in the rendered DOM before applying any fix. The remediation direction is described in the developer guidance above. Re-scan after any change to confirm resolution.

### Avoid this bad fix:

Do not rely solely on color to distinguish links. Adding text-decoration: none to links in paragraphs makes them invisible to color-blind users.

### How to verify:

Visual: verify links in text paragraphs have a non-color indicator (underline, border, icon). Test with browser in grayscale mode.

### Expected outcome:

Links in paragraphs are visually distinguishable without color. Underlines or other indicators appear.

## Step 4 of 5: Viewport prevents user zoom

Your Developer

Confirmed (High)

MODERATE | 5 instances | Multiple pages (5)

**For owners:** Your website prevents visitors from zooming in on their phone or tablet. People with low vision zoom in to read text, and blocking zoom makes your site unusable for them.

**For developers:** Remove maximum-scale=1 and user-scalable=no from the viewport meta tag. Allow users to zoom up to at least 200%.

### Root cause

Origin: Viewport meta tag includes maximum-scale=1 or user-scalable=no, preventing zoom

**This is not 5 separate issues — this is one underlying problem propagating across 5 instances.**

Single line change in the <head> template.

WCAG: 1.4.4 | Level: AA | Priority: Must fix | Changes UI: No

Scope: Repeated across 5 pages — may be a shared component or template pattern.

Example from: <https://www.breakthroughpilates.com>

### Before (captured from your site):

```
<meta id="view" name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=no">
```

user-scalable=no on <meta> tag disables zooming on mobile devices

Affected pages: <https://www.breakthroughpilates.com>, <https://www.breakthroughpilates.com/contact>, <https://www.breakthroughpilates.com/workshops>, <https://www.breakthroughpilates.com/HostYourWorkshop>, <https://www.breakthroughpilates.com/FAQ>

### Impact:

When users with low vision cannot zoom the page to increase text size.

### Exceptions:

- No exceptions — preventing zoom always fails WCAG
- Remove maximum-scale and user-scalable restrictions

### Remediation:

Owner: This requires changes to the site code or theme templates.

### After (guidance):

No site-specific code transform is safe to auto-generate for this rule. Follow the developer guidance above; the pattern depends on the exact element context.

#### Avoid this bad fix:

Do not just remove the viewport meta tag entirely. Keep it but remove the restrictive parts (maximum-scale=1, user-scalable=no).

#### How to verify:

Mobile: pinch-to-zoom on the page — it should zoom in. DevTools: check viewport meta tag no longer has maximum-scale=1 or user-scalable=no.

#### Expected outcome:

Users can zoom the page. Low-vision users can increase text size.

## Step 5 of 5: Page template missing primary heading (H1)

Your Developer

Verify Manually (Low)

MODERATE | 2 instances | Single page

**For owners:** Some pages have no main heading (H1) at all. The H1 is the page title that tells visitors and search engines what the page is about. Without it, screen reader users cannot quickly identify the page topic, and search engines may not understand your content.

**For developers:** Verify first: Add a single <h1> element to each page that describes the page's primary topic. Check whether one already exists in dynamically rendered or framework-injected content before adding a new one.

### Root cause

Origin: Page template does not render an H1 — or H1 is hidden/empty

**This is not 2 separate issues — this is one underlying problem propagating across 2 instances.**

Add or fix the H1 in the page template.

Priority: Should fix | Changes UI: Possibly

Scope: Isolated to a single page.

### Impact:

Verify in the rendered page with assistive technology to determine actual user impact..

### Exceptions:

- Verify in the rendered page
- If the element functions correctly with keyboard and screen reader, the finding may be a false positive

### Remediation:

Owner: This requires changes to the site code or theme templates.

#### Avoid this bad fix:

Do not add an H1 with placeholder or generic text like "Home" or "Welcome." The H1 must accurately describe the page's primary content.

#### How to verify:

DevTools: Run "document.querySelector('h1')" to confirm H1 exists. Screen reader: press H key to jump to headings and verify page has a clear main heading.

#### Expected outcome:

Assistive technology behavior normalizes for the affected element. Verify with screen reader after fix.

# Section 2: 3 Issues to Raise With App Vendors

---

The following issues appear associated with third-party or platform-generated markup based on DOM patterns (class names, data attributes, script signatures). Verify ownership in the rendered DOM and source before assigning remediation responsibility. If confirmed, contact each vendor listed below.

## 1. YouTube Embed Player

6 total instances | 3 issue types

### Issues found:

- Some buttons on your website have no accessible name (the label a screen reader announces).
- An element on your website has an accessibility label that isn't allowed on that type of element.
- An embedded video or widget on your website has no title or description.

### Copy and paste this email to their support team:

```
Subject: Accessibility (ADA/WCAG) Issues in Your YouTube Embed Player on Our Website

Hi YouTube Embed Player Support,

We recently conducted a WCAG 2.1 Level AA accessibility audit of our website (www.breakthroughpilates.com) and identified several accessibility violations originating from your app/widget on our site.

The specific issues are:
- Some buttons on your website have no accessible name (the label a screen reader announces) (2 instances across 2 pages)
- An element on your website has an accessibility label that isn't allowed on that type of element (3 instances across 2 pages)
- An embedded video or widget on your website has no title or description (1 instances across 1 pages)

These issues affect 2 pages on our site and create barriers for visitors who use screen readers and other assistive technology.

Can you let us know:
1. Are you aware of these accessibility issues?
2. Do you have a timeline for an accessibility update?
3. Are there any settings or configuration options we can adjust to improve accessibility?

We'd appreciate a response within the next two weeks. Thank you for your attention to this important issue.

Best regards,
[Your Name]
www.breakthroughpilates.com
```

# WCAG Success Criteria Failed

---

The following WCAG 2.1 success criteria were found to be violated. Each criterion is a specific, testable requirement from the Web Content Accessibility Guidelines.

## 1.4.1 — Use of Color (Level A)

User Impact: Medium

Color must not be the only visual means of conveying information, indicating an action, prompting a response, or distinguishing a visual element.

Why it matters: Color-blind users (8% of men) cannot distinguish information conveyed only by color, such as red/green error indicators.

Remediation: Always pair color with another indicator: icons, text labels, patterns, or underlines.

## 1.4.3 — Contrast (Minimum) (Level AA)

User Impact: High

Text and images of text must have a contrast ratio of at least 4.5:1 (3:1 for large text).

Why it matters: Low contrast text is the #2 most common accessibility issue on the web. It affects users with low vision, aging eyes, and anyone in bright sunlight.

Remediation: Increase contrast between text and background colors. Use tools like WebAIM Contrast Checker. Large text (18pt+) needs only 3:1 ratio.

## 1.4.4 — Resize Text (Level AA)

User Impact: Medium

Text can be resized up to 200% without assistive technology and without loss of content or functionality.

Why it matters: Users with low vision need to enlarge text. Fixed-size containers that clip or hide text when zoomed deny access.

Remediation: Use relative units (em, rem, %) for font sizes and container dimensions. Test at 200% zoom.

## 2.4.4 — Link Purpose (In Context) (Level A)

User Impact: High

The purpose of each link can be determined from the link text alone or from the link together with its context.

Why it matters: Screen reader users often navigate by links. "Click here" and "Read more" links without context are meaningless.

Remediation: Use descriptive link text that makes sense out of context. Avoid "click here", "read more", "learn more" without additional context via aria-label.

## 4.1.2 — Name, Role, Value (Level A)

User Impact: High

All UI components must have accessible names, roles, and states communicated to assistive technology.

Why it matters: Custom widgets (dropdowns, modals, tabs) built with divs and spans are invisible to screen readers without ARIA attributes.

Remediation: Use native HTML elements when possible. For custom widgets, add appropriate ARIA roles, labels, and state attributes.

# Appendix: Per-Instance Repair Checklist

---

Every instance found by the audit is listed below. Use this as a checklist to locate, fix, and verify each issue individually.

## button-name — 2 instances on 2 pages [App / Plugin Vendor]

### #1 Button missing accessible name [App vendor]

Page: /workshops

```
<button class="ytmVideoInfoLink ytmVideoInfoChannelAvatar"><img alt="thumbnail-image" class="ytCoreImageHost ytmVideoI...
```

No accessible name. Add visible text or aria-label to the button.

### #2 Button missing accessible name [App vendor]

Page: /HostYourWorkshop

```
<button class="ytmVideoInfoLink ytmVideoInfoChannelAvatar"><img alt="thumbnail-image" class="ytCoreImageHost ytmVideoI...
```

No accessible name. Add visible text or aria-label to the button.

## link-name — 10 instances on 5 pages [Your Developer]

### #1 Link missing accessible name [Developer]

Page: /

Link target: /

```
<a href="/" id="1709627566" file="false">
```

No accessible name. Add visible text, alt text on the linked image, or aria-label.

### #2 Link missing accessible name [Developer]

Page: /

Link target: <https://www.gratzpilates.com/>

```
<a href="https://www.gratzpilates.com/" id="1743416346" class="" target="_blank" file="false">
```

No accessible name. Add visible text, alt text on the linked image, or aria-label.

### #3 Link missing accessible name [Developer]

Page: /contact

Link target: /

```
<a href="/" id="1709627566" file="false">
```

No accessible name. Add visible text, alt text on the linked image, or aria-label.

### #4 Link missing accessible name [Developer]

Page: /contact

Link target: <https://www.gratzpilates.com/>

```
<a href="https://www.gratzpilates.com/" id="1743416346" class="" target="_blank" file="false">
```

No accessible name. Add visible text, alt text on the linked image, or aria-label.

### #5 Link missing accessible name [Developer]

Page: /workshops

Link target: /

```
<a href="/" id="1709627566" file="false">
```

No accessible name. Add visible text, alt text on the linked image, or aria-label.

### #6 Link missing accessible name [Developer]

Page: /workshops

Link target: <https://www.gratzpilates.com/>

```
<a href="https://www.gratzpilates.com/" id="1743416346" class="" target="_blank" file="false">
```

No accessible name. Add visible text, alt text on the linked image, or aria-label.

### #7 Link missing accessible name [Developer]

Page: /HostYourWorkshop

Link target: /

```
<a href="/" id="1709627566" file="false">
```

No accessible name. Add visible text, alt text on the linked image, or aria-label.

## #8 Link missing accessible name[Developer]

Page: /HostYourWorkshop

Link target: <https://www.gratzpilates.com/>

```
<a href="https://www.gratzpilates.com/" id="1743416346" class="" target="_blank" file="false">
```

No accessible name. Add visible text, alt text on the linked image, or aria-label.

## #9 Link missing accessible name[Developer]

Page: /FAQ

Link target: /

```
<a href="/" id="1709627566" file="false">
```

No accessible name. Add visible text, alt text on the linked image, or aria-label.

## #10 Link missing accessible name[Developer]

Page: /FAQ

Link target: <https://www.gratzpilates.com/>

```
<a href="https://www.gratzpilates.com/" id="1743416346" class="" target="_blank" file="false">
```

No accessible name. Add visible text, alt text on the linked image, or aria-label.

## color-contrast — 4 instances on 4 pages [Your Developer]

### #1 Insufficient color contrast[Developer]

Page: /contact

```
<a href="https://breakthroughpilates.com/">Home</a>
```

#ad0000 on #000000 — Measured: 2.77:1; Required: 4.5:1

### #2 Insufficient color contrast[Developer]

Page: /workshops

```
<span class="current">Workshops</span>
```

#737373 on #000000 — Measured: 4.42:1; Required: 4.5:1

### #3 Insufficient color contrast[Developer]

Page: /HostYourWorkshop

```
<h2>Built for Teaching</h2>
```

#ffffff on #ece8df — Measured: 1.22:1; Required: 3:1

### #4 Insufficient color contrast[Developer]

Page: /FAQ

```
<span class="current">FAQ</span>
```

#737373 on #000000 — Measured: 4.42:1; Required: 4.5:1

## aria-prohibited-attr — 3 instances on 2 pages [App / Plugin Vendor]

### #1 Invalid ARIA attribute[Elementor (Page Builder)]

Page: /workshops

```
<div class="html5-video-player ytp-hide-controls ytp-exp-bottom-control-flexbox ytp-modern-caption ytp-livebadge-color...">
```

Element has a prohibited ARIA attribute.

### #2 Invalid ARIA attribute[Elementor (Page Builder)]

Page: /HostYourWorkshop

```
<div class="html5-video-player ytp-hide-controls ytp-exp-bottom-control-flexbox ytp-modern-caption ytp-livebadge-color...">
```

Element has a prohibited ARIA attribute.

### #3 Invalid ARIA attribute[Elementor (Page Builder)]

Page: /HostYourWorkshop

```
<div class="html5-video-player ytp-hide-controls ytp-exp-bottom-control-flexbox ytp-modern-caption ytp-livebadge-color...">
```

Element has a prohibited ARIA attribute.

## frame-title — 1 instance on 1 page [App / Plugin Vendor]

### #1 Iframe missing title [App vendor]

Page: /workshops

```
<iframe src="https://www.youtube.com/embed/hhm2DNalP1E" style="position:absolute;top:0;left:0;width:100%;height:100%;b...
Embedded frame has no title attribute.
```

## link-in-text-block — 1 instance on 1 page [Your Developer]

### #1 link in text block [Developer]

Page: /HostYourWorkshop

```
<a href="https://breakthroughpilates.com/">Home</a>
```

The link has insufficient color contrast of 1.52:1 with the surrounding text. (Minimum contrast is 3:1, link text: #b8b8)

## meta-viewport — 5 instances on 5 pages [Your Developer]

### #1 Zoom disabled on mobile [Developer]

Page: /

```
<meta id="view" name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=no">
user-scalable=no on <meta> tag disables zooming on mobile devices
```

### #2 Zoom disabled on mobile [Developer]

Page: /contact

```
<meta id="view" name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=no">
user-scalable=no on <meta> tag disables zooming on mobile devices
```

### #3 Zoom disabled on mobile [Developer]

Page: /workshops

```
<meta id="view" name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=no">
user-scalable=no on <meta> tag disables zooming on mobile devices
```

### #4 Zoom disabled on mobile [Developer]

Page: /HostYourWorkshop

```
<meta id="view" name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=no">
user-scalable=no on <meta> tag disables zooming on mobile devices
```

### #5 Zoom disabled on mobile [Developer]

Page: /FAQ

```
<meta id="view" name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=no">
user-scalable=no on <meta> tag disables zooming on mobile devices
```

## page-has-heading-one — 2 instances on 1 page [Your Developer]

### #1 Page missing main heading (H1) [Developer]

Page: /contact

```
<html xmlns="http://www.w3.org/1999/xhtml" lang="en" class="">
```

Page has no H1 heading.

### #2 Page missing main heading (H1) [Developer]

Page: /contact

```
<html dir="ltr" lang="en">
```

Page has no H1 heading.

# Page-by-Page Results

---

## <https://www.breakthroughpilates.com>

Violations: 3 (0C / 2S / 1M / 0m) | Passed: 41 | Scan type: quick

## <https://www.breakthroughpilates.com/contact>

Violations: 6 (0C / 3S / 3M / 0m) | Passed: 50 | Scan type: quick

## <https://www.breakthroughpilates.com/workshops>

Violations: 7 (1C / 5S / 1M / 0m) | Passed: 47 | Scan type: quick

## <https://www.breakthroughpilates.com/HostYourWorkshop>

Violations: 8 (1C / 6S / 1M / 0m) | Passed: 48 | Scan type: quick

## <https://www.breakthroughpilates.com/FAQ>

Violations: 4 (0C / 3S / 1M / 0m) | Passed: 43 | Scan type: quick

## <https://www.breakthroughpilates.com/terms-and-conditions>

Violations: 0 (0C / 0S / 0M / 0m) | Passed: 0 | Scan type: deep

*Scan note: route returned no testable HTML surface (non-HTML response, redirect, or empty body).*

## <https://www.breakthroughpilates.com/privacy-policy>

Violations: 0 (0C / 0S / 0M / 0m) | Passed: 0 | Scan type: deep

*Scan note: route returned no testable HTML surface (non-HTML response, redirect, or empty body).*

## <https://www.breakthroughpilates.com/unlimited-january>

Violations: 0 (0C / 0S / 0M / 0m) | Passed: 0 | Scan type: deep

*Scan note: route returned no testable HTML surface (non-HTML response, redirect, or empty body).*

## <https://www.breakthroughpilates.com/copy-of-home>

Violations: 0 (0C / 0S / 0M / 0m) | Passed: 0 | Scan type: deep

*Scan note: route returned no testable HTML surface (non-HTML response, redirect, or empty body).*

## <https://www.breakthroughpilates.com/careers>

Violations: 0 (0C / 0S / 0M / 0m) | Passed: 0 | Scan type: deep

*Scan note: route returned no testable HTML surface (non-HTML response, redirect, or empty body).*

## <https://www.breakthroughpilates.com/careers-resume>

Violations: 0 (0C / 0S / 0M / 0m) | Passed: 0 | Scan type: deep

*Scan note: route returned no testable HTML surface (non-HTML response, redirect, or empty body).*

## <https://www.breakthroughpilates.com/studio-residency>

Violations: 0 (0C / 0S / 0M / 0m) | Passed: 0 | Scan type: deep

*Scan note: route returned no testable HTML surface (non-HTML response, redirect, or empty body).*

## <https://www.breakthroughpilates.com/what-to-expect>

Violations: 0 (0C / 0S / 0M / 0m) | Passed: 0 | Scan type: deep

*Scan note: route returned no testable HTML surface (non-HTML response, redirect, or empty body).*

## <https://www.breakthroughpilates.com/classical-pilates-mentorship>

Violations: 0 (0C / 0S / 0M / 0m) | Passed: 0 | Scan type: deep

*Scan note: route returned no testable HTML surface (non-HTML response, redirect, or empty body).*

## <https://www.breakthroughpilates.com/teacher-training>

Violations: 0 (0C / 0S / 0M / 0m) | Passed: 0 | Scan type: deep

*Scan note: route returned no testable HTML surface (non-HTML response, redirect, or empty body).*

## <https://www.breakthroughpilates.com/team>

Violations: 0 (0C / 0S / 0M / 0m) | Passed: 0 | Scan type: deep

*Scan note: route returned no testable HTML surface (non-HTML response, redirect, or empty body).*

## <https://www.breakthroughpilates.com/breakthrough-livestream-class>

Violations: 0 (0C / 0S / 0M / 0m) | Passed: 0 | Scan type: deep

*Scan note: route returned no testable HTML surface (non-HTML response, redirect, or empty body).*

## <https://www.breakthroughpilates.com/compare-memberships>

Violations: 0 (0C / 0S / 0M / 0m) | Passed: 0 | Scan type: deep

*Scan note: route returned no testable HTML surface (non-HTML response, redirect, or empty body).*

**<https://www.breakthroughpilates.com/check-in>**

Violations: 0 (0C / 0S / 0M / 0m) | Passed: 0 | Scan type: deep

*Scan note: route returned no testable HTML surface (non-HTML response, redirect, or empty body).*

**<https://www.breakthroughpilates.com/black-friday-special>**

Violations: 0 (0C / 0S / 0M / 0m) | Passed: 0 | Scan type: deep

*Scan note: route returned no testable HTML surface (non-HTML response, redirect, or empty body).*

**<https://www.breakthroughpilates.com/pilates-studio-rental>**

Violations: 0 (0C / 0S / 0M / 0m) | Passed: 0 | Scan type: deep

*Scan note: route returned no testable HTML surface (non-HTML response, redirect, or empty body).*

**<https://www.breakthroughpilates.com/new-client-practically-private-pilates-los-angeles>**

Violations: 0 (0C / 0S / 0M / 0m) | Passed: 0 | Scan type: deep

*Scan note: route returned no testable HTML surface (non-HTML response, redirect, or empty body).*

**<https://www.breakthroughpilates.com/corporate-wellness>**

Violations: 0 (0C / 0S / 0M / 0m) | Passed: 0 | Scan type: deep

*Scan note: route returned no testable HTML surface (non-HTML response, redirect, or empty body).*

**<https://www.breakthroughpilates.com/on-demand>**

Violations: 0 (0C / 0S / 0M / 0m) | Passed: 0 | Scan type: deep

*Scan note: route returned no testable HTML surface (non-HTML response, redirect, or empty body).*

**<https://www.breakthroughpilates.com/classes>**

Violations: 0 (0C / 0S / 0M / 0m) | Passed: 0 | Scan type: deep

*Scan note: route returned no testable HTML surface (non-HTML response, redirect, or empty body).*

# Recommendations & Next Steps

---

## 1. Update Color Contrast in Design System

Low contrast affects the largest population of any accessibility barrier, including users with low vision, older adults, and anyone in bright sunlight. Update your theme color variables to meet the 4.5:1 ratio for normal text and 3:1 for large text.

## 2. Escalate Third-Party Issues to Vendors

3 issue category(s) appear to originate from third-party apps or platform code. Use the email templates in Section 2 to contact each vendor. Track responses and consider alternatives if vendors do not commit to fixes.

## 3. Manually Verify Flagged Findings

2 finding(s) are marked "Verify Manually" because automated scanning alone cannot confirm them. Have a developer inspect these in the rendered page using DevTools accessibility panel, keyboard navigation, and a screen reader (VoiceOver, NVDA, or JAWS).

## 4. Run Screen Reader and Keyboard Checks on Critical Pages

Automated scans catch 30-40% of accessibility issues. For your highest-traffic pages (homepage, product pages, checkout, contact), test with a screen reader and keyboard. Focus on: can a user complete the task without seeing the screen?

## 5. Add Accessibility Checks to Your Development Process

Prevent new issues by adding axe-core or eslint-plugin-jsx-a11y to your build pipeline. Include accessibility acceptance criteria in feature requirements. Re-scan after fixes to verify resolution.

## Verification Notice

While these issues are detected using industry-standard automated tools, we recommend:

- Verifying key findings in a browser using the DevTools Accessibility panel
- Testing critical user flows using only a keyboard (Tab, Enter, Escape)
- Spot-checking with a screen reader (VoiceOver on Mac, NVDA on Windows, or TalkBack on Android)
- Re-scanning after fixes are applied to confirm resolution

Some accessibility requirements cannot be fully evaluated through automation and require manual testing with assistive technologies. Findings marked "Verify Manually" should be inspected in the rendered page before assigning remediation work.

## Methodology, Limits, and Legal Disclosures

Automated testing detects approximately 30-40% of WCAG 2.1 accessibility barriers. Many requirements, including keyboard navigation, screen reader compatibility, visual adjustments, and core user flow testing, can only be verified through manual review with assistive technologies. This report identifies issues detectable by automated scanning and should be used as a prioritized remediation worklist, not as a complete assessment of accessibility compliance.

This report is a technical remediation document provided for informational and remediation-planning purposes only. It does not constitute legal advice or a legal opinion regarding ADA or WCAG compliance.

Where a finding appears associated with platform, vendor, or injected markup, ownership attribution is provisional and based on DOM patterns (class names, data attributes, script signatures). Ownership should be verified in the rendered DOM and page source before assigning remediation responsibility.

# Next Steps

You have the findings. Three ways to move forward from here for [www.breakthroughpilates.com](http://www.breakthroughpilates.com).

## 1. Open your review demo

Your private review demo rebuilds up to four representative pages with the fixes applied, per-page fix notes, and carve-outs for theme/plugin/vendor work.



[adaauditreport.com/  
accessibility-review-demo](http://adaauditreport.com/accessibility-review-demo)

Key: 5d1e2ece

## 2. Ask about implementation

The \$49 report and review demo cover the audit artifact. Live-site implementation — theme edits, Shopify/WordPress updates, QA, re-scan verification — is a separate engagement, scoped per site after demo signoff.

[audit@adaauditreport.com](mailto:audit@adaauditreport.com)

Reply to your delivery email or write with the subject "Implementation quote" and your site URL.

## 3. Run a free scan on another site

Checking a client site, a second property, or a different domain? The free scan returns a homepage accessibility snapshot in seconds — no signup, no credit card.

[adaauditreport.com](http://adaauditreport.com)

