

Sample Accessibility Audit

PREPARED FOR

Sample UK ecommerce store (anonymised)

SITE TESTED

Anonymised for this sample

REPORT DATE

18 August 2026

STANDARD TESTED AGAINST

WCAG 2.2 Level AA (EN 301 549 / EAA)

Note: Client and site details have been removed for this sample. All findings, severity ratings, and methodology are unchanged from the original audit.

Executive summary

We tested this site in two stages: an automated scan, followed by manual testing using keyboard-only navigation and NVDA, a free screen reader commonly used for UK compliance testing. This report covers the manual testing findings, which are the issues an automated scan cannot detect.

Manual testing identified 10 issues across the homepage, navigation, category pages, product pages, cart, and checkout. 1 is severe enough to be classed as a Blocker, meaning it can fully prevent a keyboard user from viewing or interacting with the product they came to buy. A further 4 are classed Serious, meaning they create major difficulty completing common tasks such as browsing, checking prices, or trusting what the cart contains.

The most significant finding is a navigation menu that does not close once a keyboard user has moved past it. On product pages, this menu visually covers the entire product, meaning a sighted keyboard user cannot see or buy the item they navigated to. The second most significant finding is that the cart preview incorrectly announces itself as empty immediately after an item has been added, which risks a screen reader user either abandoning their order or adding a duplicate item.

None of the issues in this report were flagged by the automated scan; they were identified during the manual testing stage of this audit.

1 BLOCKER	4 SERIOUS	4 MODERATE	1 MINOR
---------------------	---------------------	----------------------	-------------------

Methodology

Testing was carried out in the following stages, on the homepage, a category listing page, a product page, the cart, and checkout:

- **Automated scan**, a fast baseline check, run first, catching surface-level issues such as missing labels and colour contrast.
- **Keyboard-only navigation**, all pages and tasks tested using Tab, Shift+Tab, Enter, Space, and arrow keys only, with no mouse.
- **Screen reader testing**, the same pages and tasks tested again using NVDA, a free screen reader commonly used for UK compliance testing.
- **A full purchase attempt**, completed twice: once keyboard-only, once using NVDA with the screen unwatched, to test the experience of a screen reader user directly.

Severity is assessed by task impact, not by technical category:

- **BLOCKER**, prevents a task from being completed at all.
- **SERIOUS**, makes a task very difficult, confusing, or unreliable.
- **MODERATE**, a genuine barrier, workable with effort.
- **MINOR**, a real issue with low practical impact.

These ratings reflect our judgement of real-world impact on users, not a WCAG conformance level; WCAG itself has no built-in severity rating.

Findings (10 total)

1 No working skip link

SERIOUS

PAGE: ALL PAGES FOUND VIA: KEYBOARD

WHAT HAPPENS

The first Tab press on a fresh page load lands on the homepage slideshow's pause button, not on a skip link. There is no way to bypass the header and jump directly to page content.

WHO THIS AFFECTS

Keyboard users.

WHAT SHOULD HAPPEN

A skip link should be the first focusable element, and activating it should move focus into the main content area.

FIX

Add a skip link as the first element in the page, targeting the main content region, matching standard skip-link implementation patterns (visually hidden until focused, visible on focus, linking to a main landmark with a matching id).

2 Tab order breaks after closing a popup

MODERATE

PAGE: ALL PAGES (POPUPS) FOUND VIA: KEYBOARD

WHAT HAPPENS

A popup can be reached and closed with Escape, but the next Tab press afterwards jumps to a point past the main navigation, even on a freshly loaded page, rather than continuing from a predictable position.

WHO THIS AFFECTS

Keyboard users.

WHAT SHOULD HAPPEN

Closing a popup should return keyboard focus to the element that opened it, or to a predictable, documented position, not skip forward unpredictably.

FIX

Store a reference to the element that had focus before the popup opened, and return focus to it (or to the button that triggered the popup) as soon as the popup closes, whether closed with Escape or its close button.

3 Reaching page content requires tabbing through every navigation dropdown

SERIOUS

PAGE: CATEGORY, PRODUCT FOUND VIA: KEYBOARD

WHAT HAPPENS

The main navigation opens an automatic dropdown whenever a category receives keyboard focus. Every dropdown contains many sublinks, so reaching the actual page content requires tabbing through the full contents of every menu first.

WHO THIS AFFECTS

Keyboard users.

WHAT SHOULD HAPPEN

Keyboard users should be able to move past collapsed navigation items without being forced through every open submenu, or a working skip link should let them bypass navigation entirely.

FIX

Stop navigation submenus opening automatically when a category link receives keyboard focus. Only open a submenu when its parent link is activated with Enter, so Tab can move past collapsed items without forcing every submenu open first.

4 Main category menu does not close and visually covers the page

BLOCKER

PAGE: CATEGORY, PRODUCT FOUND VIA: KEYBOARD

WHAT HAPPENS

The final navigation dropdown, which contains a large number of sublinks, does not close once keyboard focus moves past it. It remains visually open over the page underneath: on the category page it covers the entire top row including filters, and on the product page it covers the entire product.

WHO THIS AFFECTS

Keyboard users.

WHAT SHOULD HAPPEN

A dropdown should close once focus moves away from it and its contents, so it never remains open and obscuring unrelated page content.

FIX

Add a focusout handler on the dropdown that closes it as soon as focus moves to an element outside the dropdown and its menu items, and confirm it also closes on Escape.

5 Product prices are never announced

SERIOUS

PAGE: CATEGORY, PRODUCT FOUND VIA: NVDA

WHAT HAPPENS

NVDA announces product names when tabbing through category listings and individual product pages, but never announces the price, on either page type.

WHO THIS AFFECTS

Screen reader users.

WHAT SHOULD HAPPEN

Price is essential product information and should be included in what is announced for each product.

FIX

Move the price text inside the same link or container that receives focus for each product, or connect it with aria-describedby, so it's included in what gets announced when the product is reached.

6 Cart preview announces "empty" immediately after an item is added

SERIOUS

PAGE: PRODUCT (CART PREVIEW) FOUND VIA: NVDA

WHAT HAPPENS

Adding an item to the basket opens a slide-out cart preview. NVDA announces the preview opening, and in the same announcement states the cart is empty, even though the item is visibly present.

WHO THIS AFFECTS

Screen reader users.

WHAT SHOULD HAPPEN

The announcement should reflect the cart's actual contents. An incorrect announcement is worse than no announcement, since it actively misleads the user.

FIX

Update the cart preview's announced text to read from the cart's actual item count at the moment it opens, rather than a default or stale empty-state string, and re-test the announcement immediately after adding an item.

7 Changing quantity resets keyboard focus to the top of the page

MODERATE

PAGE: CART FOUND VIA: KEYBOARD

WHAT HAPPENS

Adjusting an item's quantity in the cart preview moves keyboard focus back to the start of the page after each change, requiring a full re-tab down to the cart for every adjustment.

WHO THIS AFFECTS

Keyboard users.

WHAT SHOULD HAPPEN

Focus should remain on or near the control just used, so repeated adjustments don't each require re-navigating the whole page.

FIX

After a quantity change is processed, programmatically return focus to the quantity input that was just changed, instead of letting default focus behaviour fall back to the top of the page.

8 Updated total is not announced after a change

MODERATE

PAGE: CART FOUND VIA: NVDA

WHAT HAPPENS

The cart's total is announced when the preview first opens, but changing a quantity or removing an item does not trigger an announcement of the new total.

WHO THIS AFFECTS

Screen reader users.

WHAT SHOULD HAPPEN

Any change to the order total should be announced, so a screen reader user always knows the current total.

FIX

Wrap the total amount in an `aria-live="polite"` region and update its text whenever a quantity change or removal changes the total, so the new total is announced automatically.

9 Help text on phone and security code fields is not read aloud

MODERATE

PAGE: CHECKOUT FOUND VIA: NVDA

WHAT HAPPENS

"More info" buttons beside the phone number and card security code fields announce only "expanded" when activated. The explanatory content they are meant to reveal is never read aloud.

WHO THIS AFFECTS

Screen reader users.

WHAT SHOULD HAPPEN

Activating a help disclosure should announce, or make immediately readable, the actual help content, not just its expanded/collapsed state.

FIX

Connect each "More info" button to its help text with `aria-describedby`, or move the revealed content so it is included in what's announced on expand, so the explanation itself is read aloud, not just the expanded state.

10 "MM/YY" card expiry format is misread

MINOR

PAGE: CHECKOUT FOUND VIA: NVDA

WHAT HAPPENS

The expiry date field's "MM/YY" placeholder is read aloud by NVDA as "millimetre, YY," rather than as a date format, at the point in checkout where accuracy matters most.

WHO THIS AFFECTS

Screen reader users.

WHAT SHOULD HAPPEN

Format hints should be written or marked up so assistive technology reads them as intended (for example, "month, slash, year"), not misinterpreted as an abbreviation.

FIX

Add a visually hidden label on the expiry field that reads clearly as a date format, for example "Expiry date, month then year", instead of relying on the "MM/YY" placeholder text alone.

Recommended next steps

In priority order:

FIX NOW

- 1 **Fix the navigation dropdown that fails to close (Finding 4) first**, this is the only Blocker and directly prevents purchases on affected pages.
- 2 Add a working skip link and reduce how much of the navigation a keyboard user must pass through to reach content (Findings 1 and 3).

PLAN

- 3 Correct the cart preview's empty-state announcement and ensure quantity/total changes are announced (Findings 6, 7, 8).
- 4 Ensure prices are announced alongside product names (Finding 5).
- 5 Address the remaining checkout field issues (Findings 9 and 10) before the next release.

We recommend a short re-test once these changes are made, to confirm each fix behaves correctly for keyboard and screen reader users, not just visually.