

R N I B

See differently

Shelter

Desktop and Mobile WCAG 2.1 Audit Report of selected pages from The Shelter website (England) and The Online shop

14 April 2021

by:

Suzette D'Cruz: Accessibility Consultant

Peer reviewed by:

Lori Di Bon Conyers: User Experience Evaluator

Please note:

This report may only be reproduced in its entirety and not in part.

RNIB (Royal National Institute of Blind People)
User Experience Team, Northminster House, Northminster,
Peterborough, Cambridgeshire, PE1 1YN
Email: nppd@rnib.org.uk

Royal National Institute of Blind People

A registered charity in England and Wales (226227), Scotland (SC039316) and the Isle of Man (1226). Also operating in Northern Ireland. A company incorporated by Royal Charter. Registered in England and Wales (RC000500). Registered office 105 Judd Street London WC1H 9NE

Table of Contents

1	Executive Summary	5
2	Website Accessibility audit	6
2.1	Using the audit report	6
3	Methodology	6
3.1	Web Content Accessibility Guidelines.....	6
3.2	The audit process	7
3.2.1	WCAG 2.1 Conformance	9
3.3	Pages audited.....	10
3.3.1	Shelter (England) website	10
3.3.2	Online shop	10
3.4	Summary of the results	12
3.4.1	WCAG 2.1 Check Point Conformance Summary	13
4	Detailed analysis	17
4.1	Desktop issues - Must	18
4.1.1	Issue: Unable to access areas via keyboard	18
4.1.2	Issue: Illogical heading structure	20
4.1.3	Issue: Elements that visually convey heading meaning do not match semantic heading markup	24
4.1.4	Issue: Missing form label	26
4.1.5	Issue: Form errors or status message not conveyed to assistive technology	30
4.1.6	Issue: Empty button	35
4.1.7	Issue: Increment/decrement form control incorrectly marked up for keyboard and screen reader	37
4.1.8	Issue: Main menu pop up is incorrectly marked up for screen reader	39
4.1.9	Issue: Missing or incorrect document language	42
4.1.10	Issue: Illogical reading order	44
4.1.11	Issue: Role and state of link is not conveyed to assistive technology	47
4.1.12	Issue: Missing skip link	49

4.1.13	Issue: Moving content plays automatically and cannot be controlled	50
4.1.14	Issue: Linked image missing alternative text	52
4.1.15	Issue: 'Shop here' dialog incorrect position in tab order	55
4.1.16	Issue: Content missing when magnified to 300% or greater	56
4.1.17	Issue: Status messages not conveyed to assistive technology	58
4.2	Desktop issues - Should	60
4.2.1	Issue: Decorative images missing alt attribute	60
4.2.2	Issue: Low colour contrast	63
4.2.3	Issue: Low graphic colour contrast	69
4.2.4	Issue: Error messages do not help users correct their errors	71
4.2.5	Issue: Missing autocomplete attribute on input field	74
4.2.6	Issue: Autosuggest functionality is not exposed to the screen reader	76
4.2.7	Issue: Fieldset missing legend	79
4.2.8	Issue: Navigation lists not marked up correctly	81
4.2.9	Issue: Role and state of accordion component not conveyed to assistive technology	82
4.2.10	Issue: Currently selected state it not announced by the screen reader	83
4.2.11	Issue: Non-descriptive page titles	86
4.2.12	Issue: Use of reCAPTCHA (version 31 March 2020)	88
4.3	Desktop issues - Advisory	90
4.3.1	Issue: Generic link name	90
4.3.2	Issue: Slider inaccessible by keyboard and screen reader	95
4.3.3	Issue: Incorrect use of 'breadcrumb' component for screen reader users	97
4.3.4	Issue: No announcement of link name when element receives focus by the screen reader	98
4.3.5	Issue: Incorrect ARIA references marked up for screen reader	99

4.4	Mobile only issues - Must.....	101
4.4.1	Issue: Main menu items inaccessible with the screen reader 101	
4.4.2	Issue: Hidden elements receive keyboard focus	103
5	Further Information	106
5.1	What happens next?.....	106
5.1.1	Claiming WCAG 2.1 Conformance	106
5.1.2	Arranging for the site recheck.....	106
5.1.3	Making changes to a site which carries a WCAG 2.1 conformance claim	106
6	Appendix 1: WCAG 2.1 guidelines and checkpoints for levels A and AA	107
6.1.1	Guideline 1.1: Text Alternatives	107
6.1.2	Guideline 1.2: Time-based Media	108
6.1.3	Guideline 1.3: Adaptable.....	109
6.1.4	Guideline 1.4: Distinguishable.....	109
6.1.5	Guideline 2.1: Keyboard Accessible	113
6.1.6	Guideline 2.2: Enough Time.....	114
6.1.7	Guideline 2.3: Seizures	115
6.1.8	Guideline 2.4: Navigable	116
6.1.9	Guideline 2.5 Input Modalities	116
6.1.10	Guideline 3.1: Readable	118
6.1.11	Guideline 3.2: Predictable.....	118
6.1.12	Guideline 3.3: Input Assistance.....	119
6.1.13	Guideline 4.1: Compatible.....	119
7	Contact details	121

1 Executive Summary

This report provides details of any accessibility or design issues, which will need to be addressed in order for the website to conform to W3C's Web Content Accessibility Guidelines (WCAG 2.1) level AA.

A desktop and mobile WCAG 2.1 Level AA accessibility audit review was carried out on [selected pages](#) from two websites: The Shelter website (England) and The Online shop. These pages were selected by the client. For further details see email from Andrea Dahlen (22 March 2021, 13:17).

It is important to note that, at this time the client did not want to carry out a full audit of these two websites. Therefore, representative sample of pages from the websites was not used, but a selection of pages was reviewed (as mentioned above).

Based on this evaluation, the two websites: The Shelter website (England) and The Online shop, do not meet WCAG 2.1 AA Conformance Level. A detailed review of the results is available in the Detailed Analysis section along with our recommended solutions. A total of **36 issues** have been raised.

Note:

- On the Shelter website, the search component's functionality could not be tested as this element is still under development. Presently, for any search criteria, e.g. 'donate', the following message is presented: 'Oops! We couldn't find the page you were looking for'.
- On the Online shop no test payment method was provided by the client. So, it was not possible to audit the final page of making a purchase without incurring a payment charge to an actual credit card.

Suzette D'Cruz
Accessibility Consultant
RNIB Digital Accessibility Consultancy

2 Website Accessibility audit

This section provides information on how to use the audit report, the methodology used and a summary of the results.

2.1 Using the audit report

The evaluation results in this report are based on evaluation conducted on the following date(s): 06 to 14 April 2021.

The website may have changed since the above dates.

Please remember that we can only carry out a detailed assessment of a representative sample of pages from any one website, unless the site is very small.

Therefore, the examples we provide in the "Detailed Analysis" section are just that, examples, and should not be regarded as a definitive list of all instances of any particular accessibility problem. Please use these examples to help you identify and correct all instances of each issue, not simply deal with the instances referred to in the report.

While we make every effort to identify all existing accessibility issues, the size of most commercial websites means that it is impossible for us to guarantee that we have found every problem that exists, particularly where issues may only occur on one or two pages deep within a site.

Once we have completed the audit, it is the site owner/manager's responsibility to ensure that the standard of accessibility is not degraded during content or design changes.

3 Methodology

3.1 Web Content Accessibility Guidelines

The World Wide Web Consortium's (W3C) commitment to lead the Web to its full potential includes promoting a high degree of accessibility for people with disabilities. The Web Accessibility Initiative (WAI), part of the W3C, in co-ordination with organisations around the world, is pursuing

web accessibility through five primary areas of work: Technology, Guidelines, Tools, Education & outreach and Research & development. RNIB is an active member of the W3C, and all of the tests that we carry out during an audit are based on WAI recommendations. Version 2.1 of the **WAI Web Content Accessibility Guidelines (WCAG)** was published in June 2018 by the WAI and can be found at <https://www.w3.org/TR/WCAG21/>.

3.2 The audit process

The audit tests a representative sample of pages against all of the Level A and Level AA checkpoints in the WCAG 2.1 guidelines. When all requirements are met, the site can claim conformance to level AA of the WCAG 2.1 standard.

RNIB's Web Accessibility Consultants conduct all of our website accessibility audits. The site is first checked using an automated checking tool to pinpoint any accessibility problems that can be detected programmatically and highlight aspects of the site that need to be examined more closely.

A representative sample of pages from the website is then examined using a combination of semi-automatic evaluation tools and manual evaluation in detail.

Note: in this case a representative sample of pages was not used so the websites cannot be said to be compliant to WCAG 2.1 AA.

A selection of browser and access software used, such as:

Tools: WAVE, colour contrast, accessibility and developer toolbar

Browser used to support the audit: Chrome and Safari

Assistive Technology: NVDA, JAWS, VoiceOver

Browser used to support assistive technology software: Chrome for JAWS, Safari for VoiceOver, NVDA for Firefox

We perform code, technical and manual reviews producing in-depth audit reports containing examples, solutions and strategies for improvement.

Where no issues are found, the web site will be able to claim WCAG 2.1 Level AA conformance. More commonly we find that assessed sites don't fully reach the required standard, and remedial work is needed.

3.2.1 WCAG 2.1 Conformance

Each issue found in the Audit is assigned a **priority level** based on its severity. The priority levels are: **must, should, advisory** and **recommended**.

To conform to WCAG 2.1 AA the website must address **all** the issues raised (except from recommended), regardless their assigned priority level. The priority levels are only to prioritise the issues raised.

Must issue: a critical issue is detected that means user cannot complete or it is very difficult to complete the task/goal.

Should issue: a major issue is detected that is heavily affecting the user's experience/performance in completing the task/goal.

Advisory issue: a minor issue is detected, but there are ways to overcome it or avoid it. The issue marginally affects the user's experience/performance in completing the task/goal. These issues are mainly usability problems that can be also be addressed to create a more accessible user experience on the web.

Recommended issue: this is not a WCAG 2.1 checkpoint violation but if you are able to address this issue it will help to improve the overall accessibility and usability of the website.

3.3 Pages audited

The following list of representative pages were audited:

3.3.1 Shelter (England) website

- 01 - Homepage <https://shelter-rebrand-uat.netlify.app/>
- 02 - Donate <https://shelter-rebrand-uat.netlify.app/donate>
- 03 – iRaiser <https://donate.shelter.org.uk/b?cid=397&amount=1000>
- 04 - support us https://shelter-rebrand-uat.netlify.app/support_us

3.3.2 Online shop

- 05 - online shop homepage - <https://cardshop.shelter.org.uk/>
- 06 - online shop sign up - <https://cardshop.shelter.org.uk/account/register>
- 07 - online shop captcha - <https://cardshop.shelter.org.uk/challenge>
- 08 - online shop collection <https://cardshop.shelter.org.uk/collections/christmas-gifts>
- 09 - online shop product <https://cardshop.shelter.org.uk/collections/christmas-gifts/products/milk-chocolate>
- 10 - online shop cart <https://cardshop.shelter.org.uk/cart>
- 11 - online shop information (custom url)
- 12 - online shop payment (custom url) - (do not finalise purchase)

Note: Some areas of the two websites are not in scope for this audit, see Executive Summary for full details. These areas are essential for full use of the site and the site itself cannot be considered compliant until

these areas have been addressed and a full sample of pages has been audited.

3.4 Summary of the results

Main issues:

- Unable to access areas via keyboard
- Illogical heading structure
- Missing form label
- Form errors not conveyed to assistive technology
- Main menu pop up is incorrectly marked up for screen reader

These and other issues are discussed in full within the Detailed Analysis section, along with our recommended solutions. If any solution would create difficulties or may not work within the authoring environment you use, please come back to us to discuss alternatives. We often find that on rechecking the site alternative solutions have been implemented and either don't solve the issue or create other issues.

We hope that you find the following analysis useful and informative. If you have any questions, please do get in touch, we are here to help.

The following are the results of the WCAG Success Criteria:

Level A	Level AA
Pass: 6	Pass: 7
Fail: 14	Fail: 10
N/A: 10	N/A: 3
Total: 30	Total: 20

3.4.1 WCAG 2.1 Check Point Conformance Summary

Checkpoint	Description	Pass/Fail
1.1.1: Non-text Content (Level A)	Provide text alternatives for non-text content	FAIL
1.2.1: Audio-only and Video-only (Pre-recorded) (Level A)	Provide an alternative to video only and audio only content	N/A
1.2.2: Captions (Pre-recorded) (Level A)	Provide captions for videos with audio	N/A
1.2.3: Audio Description or Media Alternative (Pre-recorded) (Level A)	Videos with audio has a second alternative	N/A
1.2.4: Captions (Live) (Level AA)	Live videos have captions	N/A
1.2.5: Audio Description (Pre-recorded) (Level AA)	Users have access to audio description for video content	N/A
1.3.1: Info and Relationships (Level A)	Logical structure	FAIL
1.3.2: Meaningful Sequence (Level A)	Present content in a meaningful order	FAIL
1.3.3: Sensory Characteristics (Level A)	Use more than one sense for instructions	PASS
1.3.4 Orientation (Level AA) [WCAG 2.1]	Do not restrict to portrait or landscape orientation	PASS
1.3.5 Identify Input Purpose (Level AA) [WCAG 2.1]	The purpose of the input can be determined programmatically	FAIL
1.4.1: Use of Colour (Level A)	Don't use presentation that relies solely on colour	FAIL
1.4.2: Audio Control (Level A)	Don't play audio automatically	N/A
1.4.3 Contrast (Minimum) (Level AA)	Contrast ratio between text and background at least 4.5:1	FAIL

1.4.4 Resize Text (Level AA)	Text to be resized to 200% without loss of content or function	PASS
1.4.5: Images of Text (Level AA)	Avoid using images of text where possible, if used they must have appropriate alt text/image description and be usable with magnification	FAIL
1.4.10 Reflow (Level AA) [WCAG 2.1]	Must be usable when resized to 320 pixels/400% zoom	FAIL
1.4.11 Non-Text Contrast (Level AA) [WCAG 2.1]	Contrast ratio of interface graphics is at least 3:1	FAIL
1.4.12 Text Spacing (Level AA) [WCAG 2.1]	Text has enough line height and spacing	FAIL
1.4.13 Content on Hover or Focus (Level AA) [WCAG 2.1]	Content triggered on focus or hover is not obscured and can be dismissed	FAIL
2.1.1: Keyboard (Level A)	Accessible by keyboard only	FAIL
2.1.2: No Keyboard Trap (Level A)	Don't trap keyboard users	PASS
2.1.4 Character Key Shortcuts (Level A) [WCAG 2.1]	If using single character key shortcuts, allow users to disable/remap them	N/A
2.2.1: Timing Adjustable (Level A)	Time limits have user controls	N/A
2.2.2: Pause, Stop, Hide (Level A)	Provide user controls for moving content	FAIL
2.3.1: Three Flashes or Below Threshold (Level A)	No content flashes more than three times per second	N/A
2.4.1: Bypass Blocks (Level A)	Provide a "Skip to Content" link	FAIL
2.4.2: Page Titled (Level A)	Helpful and clear page title	FAIL
2.4.3: Focus Order (Level A)	Logical order	FAIL
2.4.4: Link Purpose (In Context) (Level A)	Every link's purpose is clear from its context	FAIL

2.4.5: Multiple Ways (Level AA)	Offer several ways to find pages	PASS
2.4.6: Headings and Labels (Level AA)	Use clear headings and labels	FAIL
2.4.7: Focus Visible (Level AA)	Keyboard focus is visible and clear	PASS
2.5.1 Pointer Gestures (Level A) [WCAG 2.1]	If using complex gestures, allow users a simple alternative	N/A
2.5.2 Pointer Cancellation (Level A) [WCAG 2.1]	Do not trigger actions on the down event	N/A
2.5.3 Label in Name (Level A) [WCAG 2.1]	Interface components must have same name as visual label	PASS
2.5.4 Motion Actuation (Level A) [WCAG 2.1]	If triggering actions my device motion e.g. shake, provide simple alternative	N/A
3.1.1: Language of Page (Level A)	Page has a language assigned	FAIL
3.1.2: Language of Parts (Level AA)	Tell users when the language on the page changes	N/A
3.2.1: On Focus (Level A)	Elements do not change when they receive focus	PASS
3.2.2: On Input (Level A)	Elements do not change when they receive input	PASS
3.2.3: Consistent Navigation (Level AA)	Use menus consistently	PASS
3.2.4: Consistent Identification (Level AA)	Use icons and buttons consistently	PASS
3.3.1: Error Identification (Level A)	Clearly identify input errors	FAIL
3.3.2: Labels or Instructions (Level A)	Label elements and give instructions	FAIL
3.3.3: Error Suggestion (Level AA)	Suggest fixes when users make errors	FAIL
3.3.4: Error Prevention (Legal,	Reduce the risk of input errors for sensitive data	PASS

Financial, Data) (Level AA)		
4.1.1: Parsing (Level A)	No major code errors	PASS
4.1.2: Name, Role, Value (Level A)	Build all elements for accessibility	FAIL
4.1.3 Status Messages (Level AA) [WCAG 2.1]	Assistive Technology users must be notified of dynamic content	FAIL

4 Detailed analysis

All issues found in the sample pages checked are described in this section, along with information about which WCAG 2.1 checkpoint(s) each issue fails against.

The issues are categorised based on their assigned priority level, from the more severe to the least severe.

4.1 Desktop issues - Must

4.1.1 Issue: Unable to access areas via keyboard

Priority level: Must

Occurrences: Online shop collection (modal), Online shop: homepage, information/shipping/payment

Issue on mobile: Yes

WCAG 2.1 violation(s): 2.1.1 (A), 1.4.12 (AA)

Description:

There are instances where elements are not reachable and operable via the keyboard.

Also, on the 'Collection' modal, the scrollbar is inaccessible by keyboard when the user overrides text spacing for the browsers. Text spacing can improve the reading experience for certain users. Keyboard users are unable to view the full content of the modal (even with using up/down arrow keys).

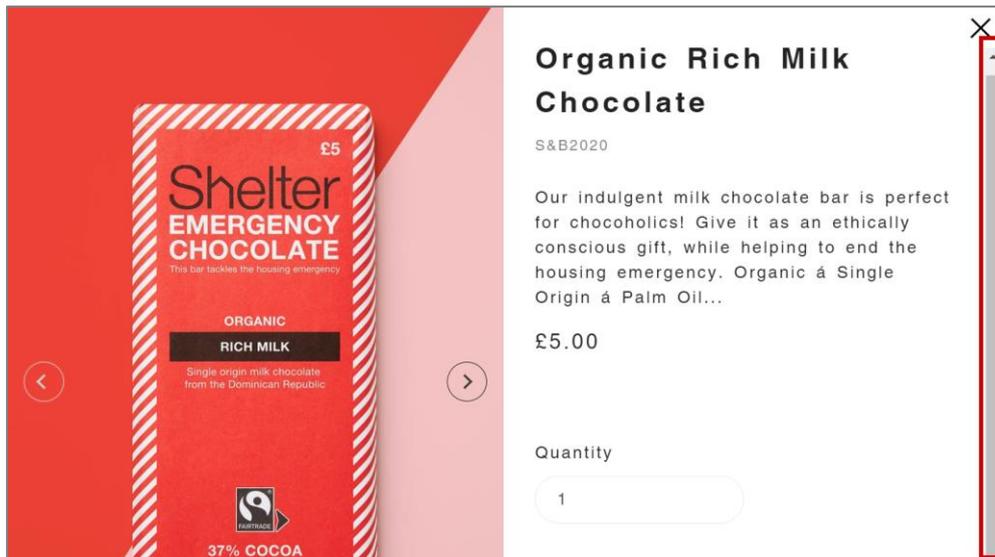
Impact on users:

Keyboard and screen reader users will not be able to use this functionality.

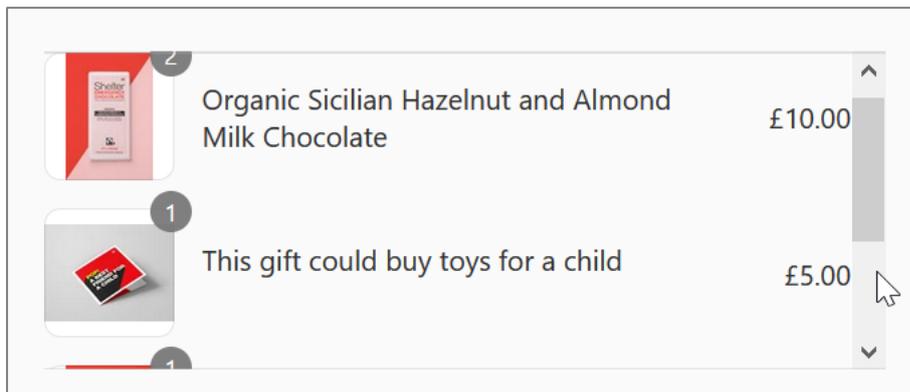
Screenshots:



Online shop collection (modal) - carousel navigation buttons - inaccessible by keyboard



Online shop collection (modal) - Scrollable content inaccessible by keyboard (using text spacing)



Online shop information/shipping/payment – the whole panel listing items for review - inaccessible area by keyboard



Online shop homepage – 'close' button inaccessible by keyboard

Recommendations:

Ensure all functionality that can be achieved by mouse/touch can also be achieved using the keyboard and expresses interactivity to screen reader users (e.g. the user can access all actionable elements on the page with the Tab key for desktop).

For further information about keyboard accessibility see:

<http://webaim.org/techniques/keyboard/>
<https://www.w3.org/TR/WCAG20-TECHS/G202.html>

4.1.2 Issue: Illogical heading structure

Priority level: Must

Occurrences: All pages e.g. Homepage, Donate, iRaiser (Donate to Shelter), Support us; Online shop: homepage, Captcha, Sign up, Collection, Product, Cart, Information, review

Issue on mobile: Yes

WCAG 2.1 violation(s): 1.3.1 (A), 2.4.1 (A), 2.4.6 (AA)

Description:

Page headings (<h1> to <h6>) do not follow a logical order with missing heading levels.

Also, on some pages headings contains no content.

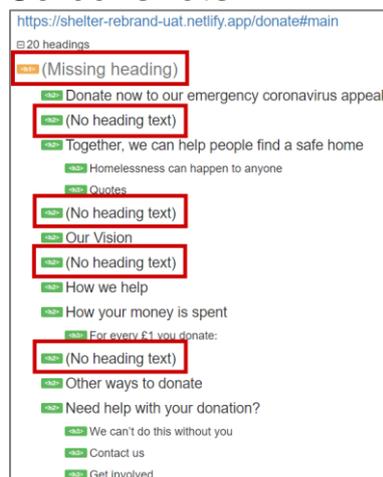
While on some pages there are duplicate headers. On the homepage, one of these headings, has no content description following on from the heading.

Impact on users:

Screen reader users use the heading structure to navigate and understand the page content. With an illogical heading structure, they will find this more difficult.

Empty headings can add to further confusion in understanding and navigating through the page.

Screenshots:



Donate – missing heading levels

https://donate.shelter.org.uk/b/my-donation?ns_ira_cr_arg=lyMjl

13 headings

- <h1> (No heading text)
- <h1> Your donation
 - <h2> Make your donation
 - <h3> I would like to make a single donation:
 - <h3> I would like to make a monthly donation:
 - <h4> (No heading text)
 - <h2> Your details
 - <h3> (Missing heading)
 - <h4> (No heading text)
 - <h2> Your donation could be worth £2.50 more with Gift Aid
 - <h2> Payment
 - <h3> (Missing heading)
 - <h4> Donate with PayPal
 - <h4> Or Donate by card
 - <h4> (No heading text)

iRaiser – missing heading levels

<https://shelter-rebrand-uat.netlify.app/>

11 headings

- <h1> Build more social housing
 - <h2> Home is Everything
 - <h3> Get Housing Help
 - <h3> The fight for home
 - <h3> We can't do this without you
 - <h4> Stay up-to-date
 - <h3> Contact us
 - <h3> Get involved
 - <h3> About Shelter
 - <h3> Preferences & terms
 - <h2> Home is everything

Homepage – duplicate header

https://shelter-rebrand-uat.netlify.app/support_us

▣ 21 headings

<h1> Your donation helps us campaign to end homelessness

<h1> Your donation helps us campaign to end homelessness

<h2> Will you give your support to Shelter?

<h3> Online shop

<h3> Events

<h3> Campaigns

Support us – duplicate header

<https://cardshop.shelter.org.uk/>

▣ 6 headings

<h1>

<h2> (Missing heading)

<h3> Contact Us

<h3> Get Involved

<h3> About Shelter

<h3> Preferences and Terms

<h3> Learn more about Shelter

Online shop homepage – missing heading level and incorrect content for H1

<https://cardshop.shelter.org.uk/account/register>

▣ 6 headings

<h1> Create Account

<h2> (Missing heading)

<h3> Contact Us

<h3> Get Involved

<h3> About Shelter

<h3> Preferences and Terms

<h3> Learn more about Shelter

Online shop sign up – missing heading level

<https://cardshop.shelter.org.uk/challenge>

5 headings

<h1> (Missing heading)

<h2> (Missing heading)

<h3> Contact Us

<h3> Get Involved

<h3> About Shelter

<h3> Preferences and Terms

<h3> Learn more about Shelter

Online shop captcha – missing heading levels

Recommendations:

Structure content using an appropriate and logical heading structure. Each page should have a single <h1> that describes the purpose of the page, followed by headings in sequential order e.g. <h2>, <h3> etc. Doing so will prompt the screen reader to announce them as heading, announce the level and allow the user to navigate the page by skipping between headings.

Ensure that all headings contain informative content and add headings to the pages, outlined above, for ease of navigation and to enhance context of the content displayed for screen reader users.

Also, avoid duplicating, except when the structure provides differentiation between them. Or consider removing the duplicate heading or change one of the heading's text.

For a more comprehensive understanding of properly structuring headings please see:

<https://www.w3.org/WAI/tutorials/page-structure/headings/>

4.1.3 Issue: Elements that visually convey heading meaning do not match semantic heading markup

Priority level: Must

Occurrences: iRaiser (Donate to Shelter)

Issue on mobile: Yes

WCAG 2.1 violation(s): 1.3.1 (A), 2.4.1 (A), 2.4.6 (AA)

Description:

There are numerous text elements that appear to be a heading but are not using the semantic heading markup.

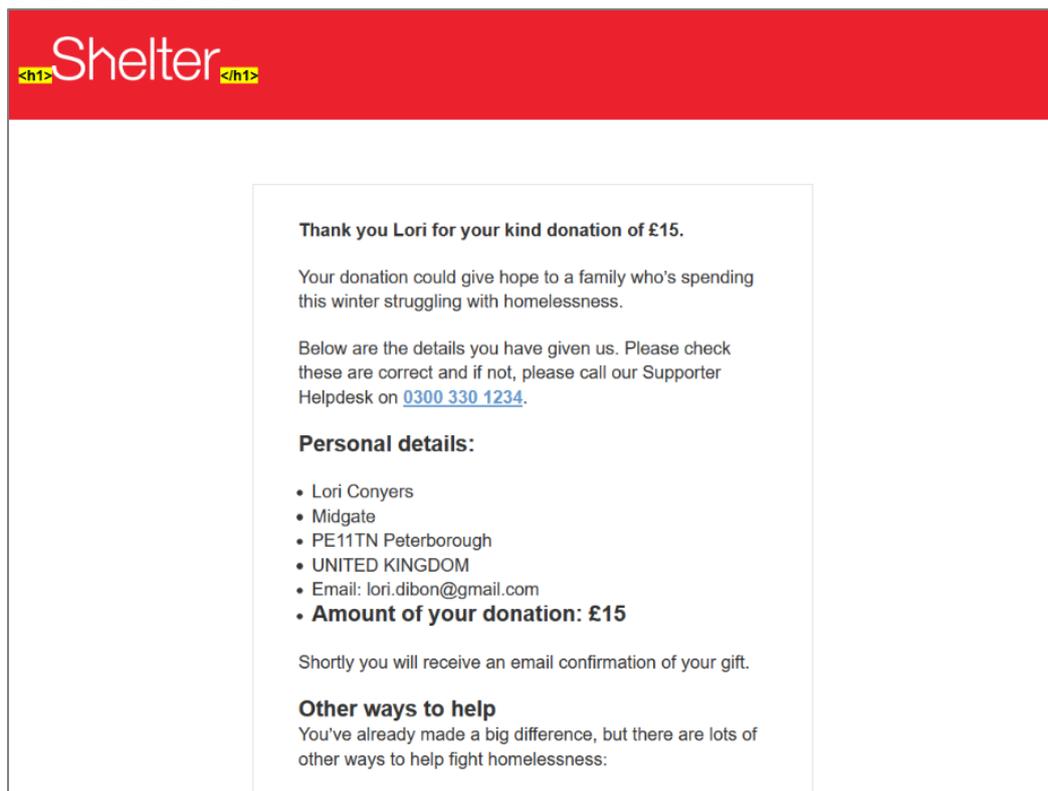
Heading elements (<h1>-<h6>) provide important document structure, outlines, and navigation functionality to assistive technology users. If heading text is not a true heading, this information and functionality will not be available for that text.

The Shelter logo has been marked up as an H1, this is incorrect and unhelpful for navigation. Ideally, the 'Thank you...' needs to be marked up as an H1.

Impact on users:

Screen reader users may be unaware of the informative importance of these text elements as they do not have the same prominence as they do visually.

Screenshots:



iRaiser (Donate to Shelter) – missing headings

Recommendations:

Ensure bold, large and image text element content is perceivable to screen reader users (e.g. the reason of its importance in large text). Convert these elements into one of the semantic heading tags (<h1>-<h6>) to enhance navigation for screen reader users.

For further information on semantic heading markup and understanding of properly structuring headings see:

<https://www.w3.org/WAI/tutorials/page-structure/headings/>

<https://bbc.github.io/gel/foundations/headings/>

4.1.4 Issue: Missing form label

Priority level: Must

Occurrences: All pages Shelter (search component), Donate (multiple form labels); iRaiser (Donate to Shelter); Online shop: sign up, cart

Issue on mobile: Yes

WCAG 2.1 violation(s): 1.1.1 (A), 1.3.1 (A), 2.4.6 (AA), 3.3.2 (A), 4.1.2 (A)

Description:

There are a number of instances where text fields do not have an associated label element (<label>) to provide a text label to assistive technologies. Labels without a 'for' attribute that references the ID of the input. Because of this, the inputs effectively do not have a label. A label element that references the ID of the input is required to provide an accessible name for assistive technology.

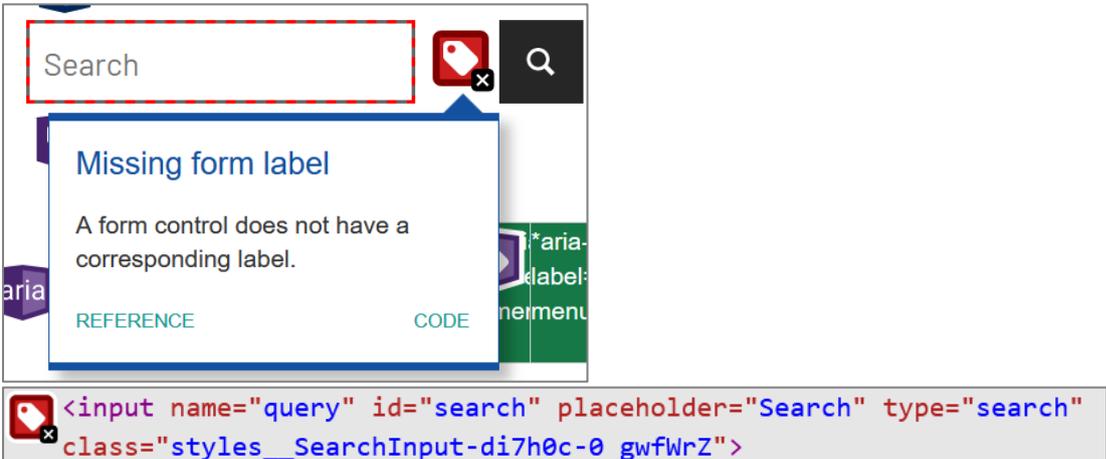
Also, on the Donate page, multiple form labels are present on a form control. Therefore, it does not have a corresponding label.

Note: placeholder text is not a sufficient alternative, all inputs must have an accessible label.

Impact on users:

Screen reader users will be unable to identify the purpose of the form input and will have difficulty successfully completing forms.

Screenshots:



The screenshot shows a search input field with the placeholder text "Search". A red dashed box highlights the input field. A blue error message box is overlaid on the input, stating "Missing form label" and "A form control does not have a corresponding label." Below the error message, there are two buttons: "REFERENCE" and "CODE". The "CODE" button is selected, and the corresponding HTML code is displayed in a code block below the error message. The code is:

```
<input name="query" id="search" placeholder="Search" type="search" class="styles__SearchInput-di7h0c-0 gwfwRz">
```

All pages – form control missing form label

Multiple form labels

A form control has more than one label associated with it.

REFERENCE CODE

```

<span class="styles__Currency" style="font-size: 1.2em; font-weight: bold;">£
</span>
<input type="number" step=".01" name="amount-holder" value="" placeholder="30"
id="amount-holder" pattern="[0-9]*(\.[0-9]{0,2})?" class="styles__Input-sc-3eimrs-1 bQchrs">

```

Donate – Multiple form labels

Select missing label

A select element does not have an associated label

REFERENCE CODE

iRaiser (Donate to Shelter)

```

<input type="password" value name="customer[password]" id="create_password" placeholder="Password" class="error">
<label for="password" class="hidden-label">Password</label>

```

Online shop sign up – incorrect associated text label – ‘for’ and ‘id’ don’t match



Online shop cart – form controls missing form labels

Want to be kept updated on our work, campaigns, fundraising activities, events, volunteering opportunities, and products and services?

	Yes	No
Email	<input type="radio"/>	<input checked="" type="radio"/>
Phone	<input checked="" type="radio"/>	<input type="radio"/>
Text	<input type="radio"/>	<input checked="" type="radio"/>
Post	<input type="radio"/>	<input checked="" type="radio"/>

iRaiser – radio buttons not associated to their labels - however, these are intermittently announced.

Recommendations:

Ensure all form fields have an associated text label (<label>) with a single form label. To add a label for screen-reader users, use a <label> element with text that describes the fields purpose with a for attribute that references the id of the field.

When providing forms for user input, it is essential that fields are well labelled so that users are aware what data is expected from them. Ensure the placeholder text matches the label or provide instructions with `aria-describedby`. While placeholder text can be used to support the user in entering information correctly, it should not be used as a replacement for a properly positioned text label. The placeholder used to display the fields purpose to sighted users, is currently only displayed when a user is not focused inside it and an always present label will serve as a reminder of the expected input of the field.

Further investigation is required as to why the labels are not always announced, for the first set of radio buttons, on the iRaiser form.

For further information on accessible forms please see:

<https://www.w3.org/TR/WCAG20-TECHS/H44.html>

<https://www.w3.org/WAI/tutorials/forms/>

https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA/ARIA_Techniques/Using_the_aria-describedby_attribute

<https://design-system.service.gov.uk/components/radios/>

For **working examples** of the **search component**, see:

<https://luna.sainsburys.co.uk/components/search>

4.1.5 Issue: Form errors or status message not conveyed to assistive technology

Priority level: Must

Occurrences: Homepage; All pages (footer email form): Online shop: footer email sign up form, Sign up

Issue on mobile: Yes

WCAG 2.1 violation(s): 3.3.1 (A), 1.4.1 (A)

Description:

The error messages provided when a user makes a mistake completing a form are not conveyed to assistive technology. Not all mandatory fields are labelled on all areas of the forms.

On the 'Online shop' homepage the error message is announce correctly for the sign up form at the footer of the page. However, the visual message displayed disappears quickly. Or, where part of the email address has been inputted, the error message displayed floats above the input field and obscures the field.

On the 'Online shop' sign up page the error messages are not announced. Also, before the error messages are provided, the user is taken to the Challenge page to submit the CAPTCHA form correctly. This must be performed every time the sign up form is submitted for validation.

Also, on the Online shop forms, the input field changes colour from a grey box to a red box. Here colour alone is used to indicate an error has occurred with a list of error messages at the top of the form.

Impact on users:

Screen reader users will find it difficult to read error messages and correct their mistakes. All users would benefit with the indication of mandatory fields to complete the form.

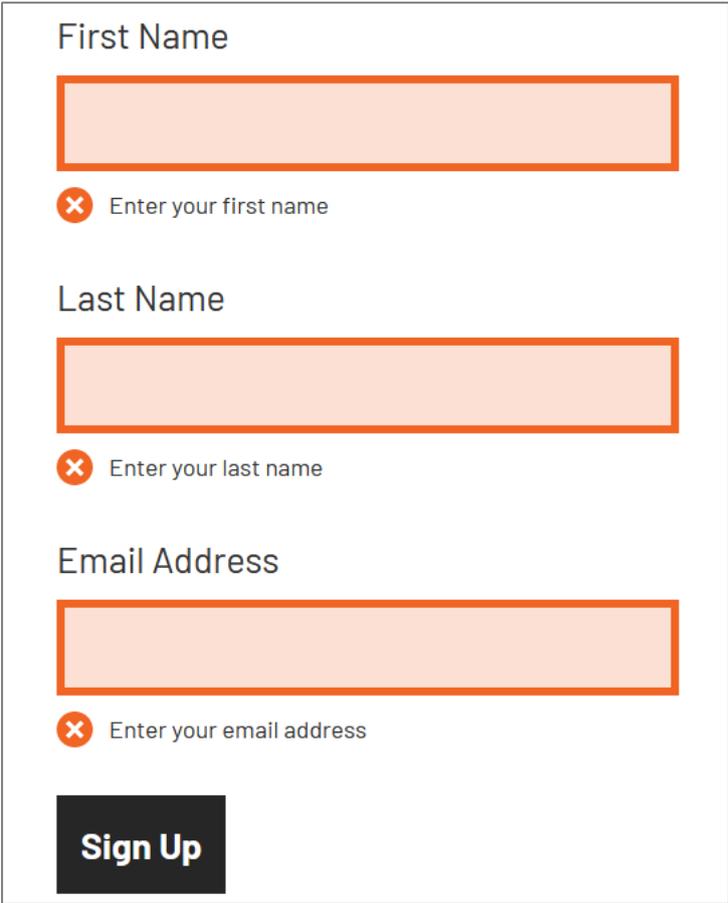
Cognitive impaired and low vision users could become confused or disoriented arising from the error message as there was insufficient time to read the message on the Online shop homepage.

Also, the overlap-floating of the error message could affect all users as well.

On the 'Online shop' sign up, having to submit the CAPCHA every time to submit the form before all error messages are corrected maybe laborious for all users.

Non-sighted users and users with certain visual impairments will not be able to perceive the change of colour of the input field outline. For example, some users who have colour blindness might find it difficult to identify the red input field outline in relation to the rest of the other input fields on the form (and a green outline for correctly inputted fields). On some inputs, a tick and cross are available within each field to indicate correct and incorrect fields (iRaiser form). However, this does not help some users.

Screenshots:



The screenshot shows a sign-up form with three input fields: 'First Name', 'Last Name', and 'Email Address'. Each field has a red border and a red 'x' icon with the text 'Enter your first name', 'Enter your last name', and 'Enter your email address' respectively. Below the fields is a black 'Sign Up' button.

Homepage – form error messages not announced

Your details

All fields are required except those marked as 'optional'

Email

I am making a donation on behalf of an organisation

Title

First name

Last name

Mobile number (optional)

iRaiser (Donate to Shelter) – form error messages not announced

Email Address Sign Up

! Please fill in this field.

Online shop homepage – form error messages disappears quickly

opportunities, and products and services?

Please enter an email address. Sign Up

Online shop homepage – form error messages overlaps with the input field

Create Account

- The **password** can't be blank
- The **email** can't be blank

First Name

Last Name

suzette.dacruz@mib.org.uk

.....

Create

Online shop sign up – incorrect form error messages (i.e. despite inputs being correct)

Create Account

- The **password** can't be blank
- The **email** can't be blank

First Name

Last Name

Email

Password

Create

Online shop sign up – form error messages not announced; red outline input field may not be perceivable

Recommendations:

Ensure all form error messages are readable by assistive technology.
See:

<https://www.w3.org/WAI/tutorials/forms/notifications/>.

Avoid using colour alone to convey information. Provide additional visual and textual cues to inform users of the error message near the input field as well as providing the list above the form.

Ensure that all users are aware of mandatory fields. Perhaps by including an asterisk symbol besides the text field label (available on some input fields). You should also include an instruction sentence to precede the form to inform users of the purpose of this asterisk. Asterisk should be coded in a label e.g. `<label for="first_name">XXX*</label>` so it is announced to screen reader users when they tab to the field. On the 'iRaiser' page there are instructions at the top of the page to inform users of mandatory form. This is a working example for the other form to perhaps adopt.

Also, for the radio buttons on the 'iRaiser' page, indicate the radio group is required by adding text 'required' in the group label.

Also, ensure error messages do not disappear quickly nor overlap with its input field.

Consider adding the CAPTCHA challenge after all form error validation has been performed before creating an account.

For further information on how to make accessible form validation and presenting error messages please see:

<https://bbc.github.io/gel/components/form-fields/>
https://developer.mozilla.org/en-US/docs/Learn/HTML/Forms/Form_validation
<https://design-system.service.gov.uk/components/error-message/>
<https://design-system.service.gov.uk/components/error-summary/>

For a **working example**, please see (follow the journey through this forms):

<https://bank.barclays.co.uk/olb/registration/registerAppContainer.do#/user-details>

4.1.6 Issue: Empty button

Priority level: Must

Occurrences: All pages

Issue on mobile: Yes

WCAG 2.1 violation(s): 1.1.1 (A), 1.3.1 (A), 2.4.6 (AA), 3.3.2 (A), 4.1.2 (A)

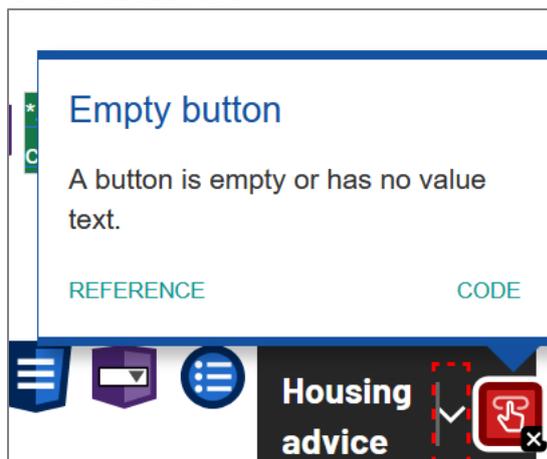
Description:

The image buttons used on this page does not contain content (empty).

Impact on users:

Screen reader users will be unable to identify the purpose of this button and will have difficulty successfully completing the task.

Screenshots:



Homepage – pop up menu – empty button



Homepage – search – empty button

Recommendations:

Ensure all buttons contains a descriptive label.

Use `aria-label` to provide an accessible label. For example, place the `aria-label` on the button element.

Further information on labelling buttons can be found:

<https://www.w3.org/TR/WCAG20-TECHS/ARIA6.html>

<https://webaim.org/techniques/forms/controls#button>

4.1.7 Issue: Increment/decrement form control incorrectly marked up for keyboard and screen reader

Priority level: Must

Occurrences: All pages with Donate component

Issue on mobile: Yes

WCAG 2.1 violation(s): 3.3.2 (A), 4.1.3 (A)

Description:

The spin button form control has a number of issues:

- Incorrect input value is announced by the screen reader, see screenshot
- While inputting values, these are incorrectly announced as well. For example, when inputting the value 100, the value is announced as 30 when using the up/down arrows to change the values. The screen reader is announcing the placeholder text and not the inputted value. Visually, the placeholder text is no longer visible.
- The value with this form control is not associated to its currency. So only the value entered is announced.
- For keyboard users there is no indication that this form control is a spin button to perform incremental/decremental value changes. It's just by exploring with the up/down arrow that this functionality is apparent.
- There are no instructions of using this component nor does the label for it help with how to use it.
- On the iRaiser (Donate to Shelter) page this component works correctly. However, for keyboard only users there's no visual indication that incremental/ decremental value changes are available.

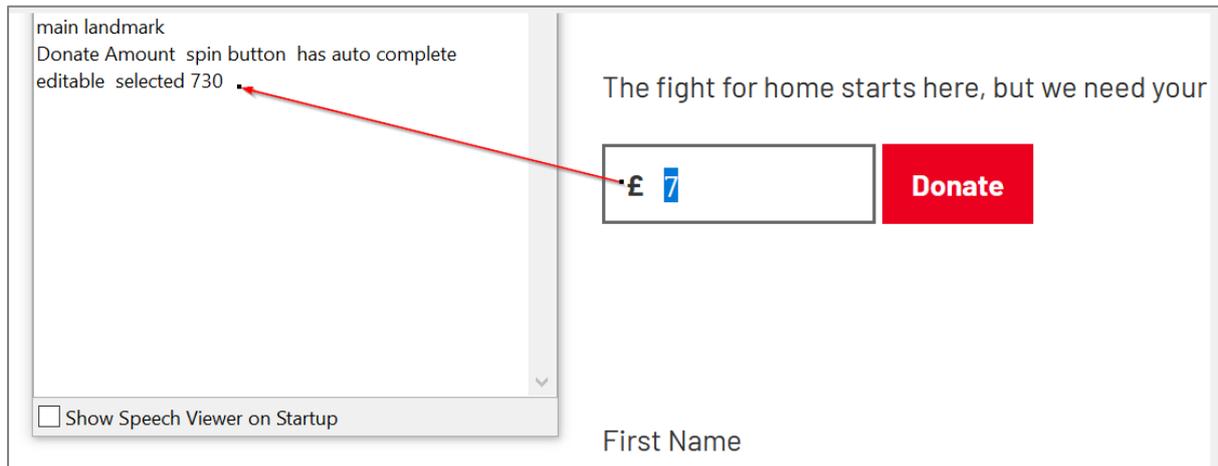
Impact on users:

Screen reader users will be unaware of the correct incremental or decremental values following interaction with the button. They may be

confused by the values announced as there's no association to its currency type.

Also, users who are restricted to using only a keyboard and/or screen reader will be unable to use the spin control's functionality.

Screenshots:



All pages with Donate component – form control with spin button incorrectly marked up

Recommendations:

Ensure the correct inputted values of the spin button form control are announced to the screen reader user. Also, associate the currency type with the value inputted so this is also announced to the screen reader user.

Consider making the spin button visual within the form control so keyboard users are aware of this functionality, as in the screenshot below:



Visual example of spin button with form control

Or provide adequate information on how to interact with it above the control.

For a **working example** see:

<https://www.barclays.co.uk/loans/budget-planner/>

4.1.8 Issue: Main menu pop up is incorrectly marked up for screen reader

Priority level: Must

Occurrences: All pages: Shelter & Online shop

Issue on mobile: Yes

WCAG 2.1 violation(s): 1.4.13 (AA), 4.1.2 (A)

Description:

There are a number of issues relating to the pop up menu component:

- The label and the component have been split. Therefore, receiving keyboard focus twice.
- When the label receives keyboard focus, it announces “Menu, Housing Advice, to move through items use up and down arrow”. However, this does not offer any such functionality, i.e. it’s inaccessible.
- When the pop up button receives focus after tabbing forward from the previous label, this button is unlabelled, but announces the correct state of the collapsible/expandable button. The unlabelled button issue has been raised under: [‘Empty button’](#).
- The pop up button does not collapse when focus is moved away from the expanded sub menu panel i.e. moves to the next main menu item, ‘Get Help’. It can only be closed by re-triggering the pop up button.

On the Online shop when the menu receives focus it’s announced as collapsed. However, on keyboard focus it automatically opens. The user has to navigate through the sub menu items to reach the next main menu item and for the pop up to collapse.

Impact on users:

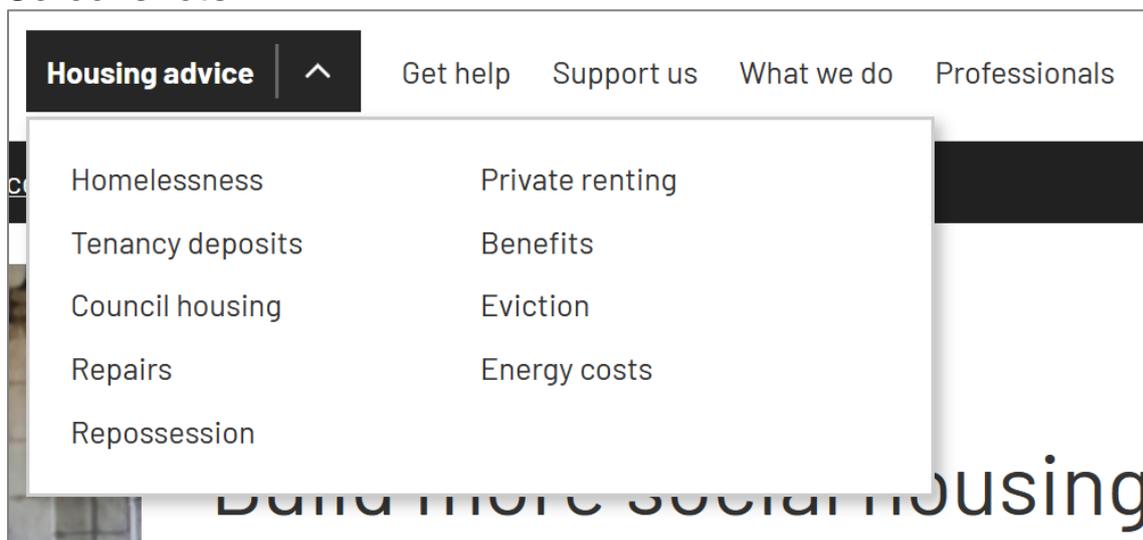
On the Shelter website, keyboard and screen reader users do not have control of activating the sub menu without tabbing twice. Screen reader users maybe unaware of the contents of the sub menu and could be confused with having to tab twice to open the sub menu.

As the expanded sub menu panel remains open, keyboard users have to navigate back from the menu item 'Get Help' to the pop up button to close it. Or find that the expanded panel, when navigating forward, obscures certain parts of the page.

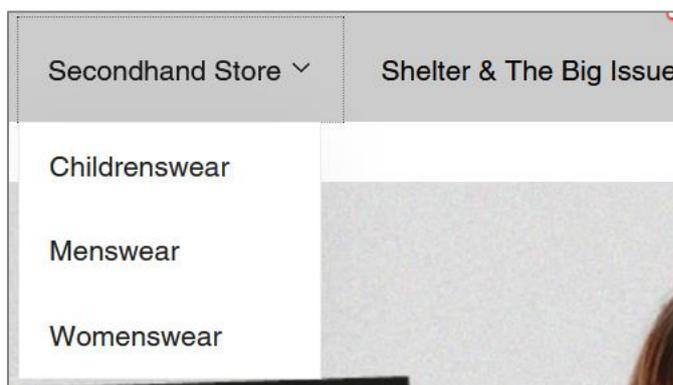
On the Online shop the screen readers could get confused by the announcement collapsed when the pop up is already expanded.

It is laborious having to tab through the sub menu list for keyboard and screen reader users to reach the next main menu item. Fortunately, at this present time, it's a short list comprising of three submenu items.

Screenshots:



All pages - Main menu pop up – split component



Online shop homepage – Main menu pop up – incorrectly marked up

Recommendations:

Ensure the pop up component is combined with its label and pop up button so screen reader users will know how to interact with the

component. This will also help keyboard users by limiting the number of tabbing strokes.

Also, ensure the expanded sub menu panel closes once focus is moved away from it.

For further information see:

<https://www.w3.org/TR/wai-aria-practices-1.1/#menu>.

For a **working example** see:

<https://www.w3.org/WAI/tutorials/menus/flyout/>.

Also, see the **working examples** from Adobe:

Submenus that need to be triggered to open:

<https://adobe-accessibility.github.io/Accessible-Mega-Menu/>

Explanation on mega menu:

<http://blogs.adobe.com/accessibility/2013/05/adobe-com.html>

4.1.9 Issue: Missing or incorrect document language

Priority level: Must

Occurrences: iRaiser (Donate to Shelter); All pages: Online shop

Issue on mobile: Yes

WCAG 2.1 violation(s): 3.1.1 (A)

Description:

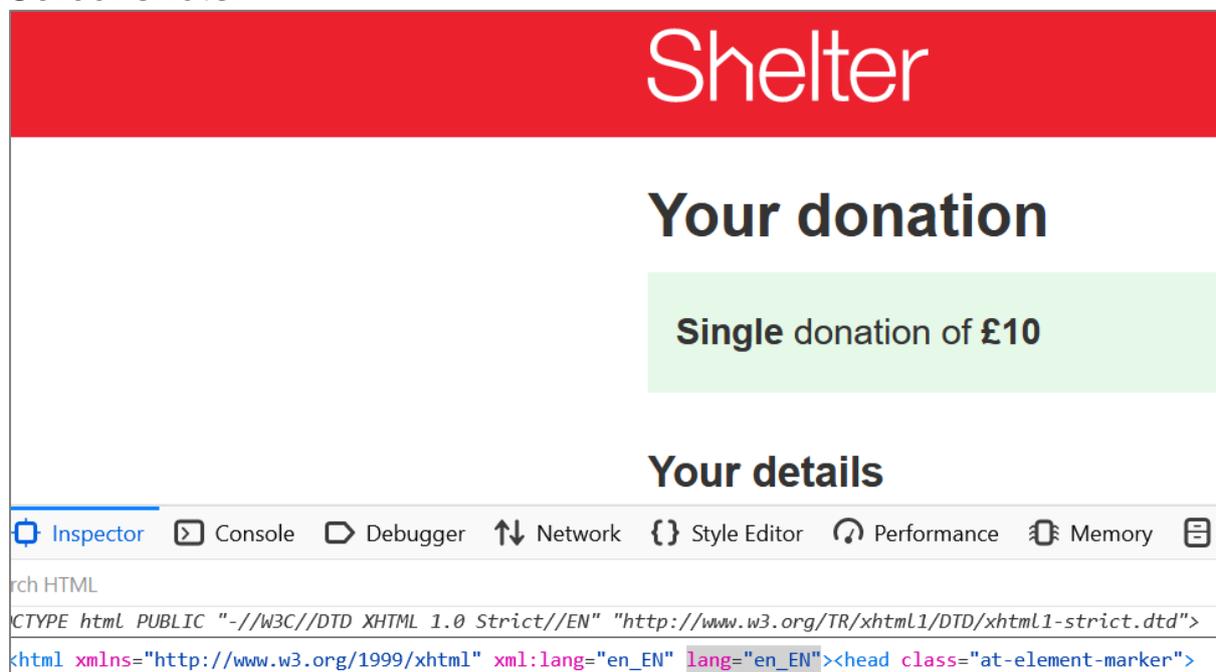
On all pages of the Online shop, the language of the document is missing.

On the iRaiser page, the language of the document used is incorrect. The code “en_EN” has been used which is meaningless as it’s an invalid language code (using a suffix code “EN”).

Impact on users:

Screen readers may not render the content of the page accurately without a language code and with an incorrect use of the code.

Screenshots:



iRaiser (Donate to Shelter) – invalid language code

Recommendations:

Ensure the default language of the document is set correctly. For example, as `lang="en"`, this specifies that language used is British English.

For further information see the following URL:

<https://www.w3.org/WAI/WCAG21/Techniques/html/H57>

For a **working example**, see all other pages tested (e.g. homepage, donate) in this audit as they have been coded correctly.

4.1.10 **Issue: Illogical reading order**

Priority level: Must

Occurrences: iRaiser (Donate to Shelter), Online shop information, shipping

Issue on mobile: Yes

WCAG 2.1 violation(s): 1.3.2 (A), 2.4.3 (A)

Description:

The reading order is illogical on a number of elements on the iRaiser page. The focus must follow the reading order (left to right, top to bottom).

This is probably caused by the elements using a tabindex greater than zero.

A similar issue occurs on the 'Online shop Information' page. Also, when the page loads, following on from the journey of the previous page (Cart), the focus is placed in the middle of the page rather than at the top of the page.

Impact on users:

Users who are restricted to using only a keyboard may be confused by the unexpected behaviour.

The focus does not follow the reading order and to move sequentially on a page (left to right, top to bottom).

Screenshots:

Single donation of £10 [Change](#) 26

Your details

All fields are required except those marked as 'optional'

Email 1

2 I'm making a donation on behalf of an organisation

iRaiser (Donate to Shelter) – reading order of links not left to right - 'Change' component not the first element to receive keyboard focus

Post 19

You can [change your preferences](#) 27 at any time by giving us a call on [0300 330 1234](tel:03003301234) or emailing info@shelter.org.uk 28. If you've already told us you want to receive updates, we'll keep contacting you in the same ways. And don't worry – although Shelter and Shelter Trading activities will use your details, we will never pass them on to third parties for marketing. [Read our privacy policy](#) 29 for more details.

Your donation could be worth £2.50 more with Gift Aid

Please read the declaration below and tick the box if you are able and happy to add Gift Aid to your donation

20 I would like Shelter to reclaim the tax I have paid on this

iRaiser (Donate to Shelter) – reading order of links not left to right

Recommendations:

Ensure the reading order is logical. Content that needs the user's attention before reading the page must be at the top of the DOM order e.g. the 'Change' component to set the donation amount.

Also, ensure the focus order is logical and follows the reading order (left to right, top to bottom). Place the focus at the top of the page, before the skip link, when the page loads, especially when it's part of a user journey, such as making a purchase.

For best practice, ensures tabindex attribute values are not greater than 0.

For further information see:

<https://www.w3.org/TR/2016/NOTE-WCAG20-TECHS-20161007/H4>

<http://www.w3.org/TR/WCAG20-TECHS/F44.html>

4.1.11 Issue: Role and state of link is not conveyed to assistive technology

Priority level: Must

Occurrences: iRaiser (Donate to Shelter)

Issue on mobile: Yes

WCAG 2.1 violation(s): 4.1.2 (A), 4.1.3 (AA)

Description:

The correct state of the for 'Change' link in its collapsible/expandable position needs to be exposed to the screen reader user.

Also, when the expanded form has been submitted by triggering the 'Confirm change' link the status of the change in value is not announced to the screen reader user e.g. 'Monthly donation of £50'. The new changes are visually displayed.

Impact on users:

The screen reader user is unaware of the status of the link, whether it's expanded or collapsed. Therefore, they could be unaware of the 'donation giving' form when the link is triggered. Triggering the link, the keyboard focus is taken to the next focusable element rather than to the newly exposed form.

After making changes to the form and triggering the link to confirm changes, screen reader users will be unaware if these changes have been made as there's no announcement.

Screenshots:



iRaiser – link – collapsed states not announced to the screen reader user

Your donation

Monthly



donation of £ 2000

Confirm change

iRaiser – link – expanded states not announced to the screen reader user

Recommendations:

Ensure the 'Change' link is correctly marked up. An aria attribute of `aria-expanded` could be used to communicate the state of this link to screen-reader users.

Also, ensure focus is moved into the exposed form after activating the trigger element, 'Change' link (i.e. first form control).

Furthermore, ensure status messages that is added, as a result of donation changes, are conveyed to assistive technology users.

Consider providing instructions on how to interact with the link and where the form will be displayed to change the value of the donation.

For further information on using `aria-expanded` see:

https://www.w3.org/WAI/GL/wiki/Using_aria-expanded_to_indicate_the_state_of_a_collapsible_element

https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA/ARIA_Techniques

Use alert aria role for status messages. For further information, see:

https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA/ARIA_Techniques/Using_the_alert_role

<http://www.w3.org/TR/wai-aria-practices-1.1/#alert>

4.1.12 Issue: Missing skip link

Priority level: Must

Occurrences: All pages: Online Shop

Issue on mobile: Yes

WCAG 2.1 violation(s): 2.4.1 (A)

Description:

A skip link is not present. A skip link is a mechanism that enables users who are restricted to using only a keyboard to bypass blocks of content that are repeated on multiple web pages e.g. header navigation.

Impact on users:

Users who are only able to use a keyboard are unable to bypass repeated content, forcing them to navigate links and inputs before using the tab key before gaining focus on the main content.

Screenshots:

N/A

Recommendations:

Provide a skip link that enables users to bypass blocks of repeated content and gain focus on the main content of the page.

Note: The skip link should be the first link in the DOM. The link is not required to be visible at all times but must be visible when it receives keyboard focus.

For further information please visit URLs:

<http://www.w3.org/TR/WCAG20-TECHS/G1.html>

<https://design-system.service.gov.uk/components/skip-link/>

For a **working example**, see the Shelter website, as this has been implemented correctly:

<https://shelter-rebrand-uat.netlify.app/>

4.1.13 Issue: Moving content plays automatically and cannot be controlled

Priority level: Must

Occurrences: Online shop homepage

Issue on mobile: Yes

WCAG 2.1 violation(s): 2.2.2 (A)

Description:

There is an instance where moving content plays automatically on page load and cannot be controlled (pause/play):

Impact on users:

Users with cognitive impairments may find the moving content distracting and that it affects their ability to concentrate on other content.

Screenshots:



Online shop homepage – carousel plays automatically, pause mechanism not provided

Recommendations:

Avoid moving content that plays automatically. All users must be able to control moving content (pause/stop). See:

<https://www.w3.org/WAI/WCAG21/Techniques/general/G4.html>

<https://www.w3.org/WAI/WCAG21/Techniques/general/G186.html>

For further information on accessible carousels:

<http://www.w3.org/WAI/tutorials/carousels/>
<https://bbc.github.io/gel/components/carousels/>

For a **working example** of a carousel see:

<http://www.w3.org/WAI/tutorials/carousels/working-example/>

4.1.14 Issue: Linked image missing alternative text

Priority level: Must

Occurrences:

Issue on mobile: Yes

WCAG 2.1 violation(s): 1.1.1 (A), 1.4.5 (AA), 2.4.4 (A)

Description:

There are a number of instances where an image is nested in a link and does not have alternative text (alt attribute) and/or is incorrectly marked up for the image. For example, the Shelter logo image link.

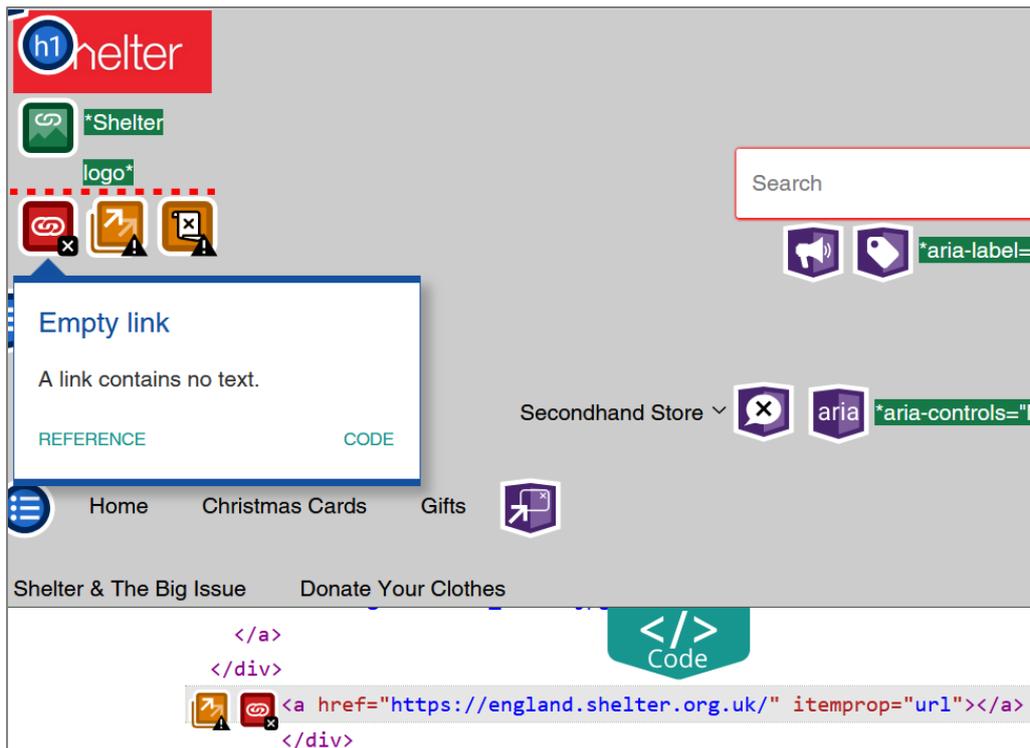
Also, on the carousel, there are linked images containing text that have alternative text, which does not match the text found in the image. This is a combination of incorrect alt text (e.g. Christmas gift or alt=""), missing alt text or target of alt text marked up incorrectly. All these seem to conflict with one another for announcing the correct link name for screen reader users.

Impact on users:

Screen reader users may not be able to understand the purpose of the image.

Also, for the carousel, screen reader users will not be able to understand the text contained in the image as this visual presentation is unavailable in the alt text.

Screenshots:



Online shop homepage - linked images missing alt text – however there is another alt text available for this image – this is conflicting for the screen reader



Online shop homepage – linked images have various alt text for each image including alt=""

Recommendations:

Ensure linked images have alternative text (alt attribute) that describes the link destination and image. Complex images require an accessible alternative, e.g. described by using text only.

Note: The alt attribute should not be used to provide a text alternative for complex images if the description is too long. If possible, avoid images of text.

For further information please see the following URL:

<https://www.w3.org/WAI/tutorials/images/functional/>

<http://www.w3.org/WAI/tutorials/images/complex/>

For a **working example** of a carousel see:

<http://www.w3.org/WAI/tutorials/carousels/working-example/>

4.1.15 Issue: 'Shop here' dialog incorrect position in tab order

Priority level: Must

Occurrences: All pages: Online shop

Issue on mobile: Yes

WCAG 2.1 violation(s): 2.4.3 (A)

Description:

The 'Shop here' dialog is after the main content in the tab order. The content of the dialog does not receive focus until after the main content, after the footer.

Impact on users:

Keyboard only users will have to navigate the main content to interact with the 'Shop here' dialog.

Screenshots:



All pages: Online shop – 'Shop here' dialog

Recommendations:

Ensure the 'Shop here' dialog is near the top of the tab order to enable users to navigate to it before the main content. Place the cookie consent dialog near the top of the DOM order.

4.1.16 Issue: Content missing when magnified to 300% or greater

Priority level: Must

Occurrences: Online shop collection

Issue on mobile: Yes

WCAG 2.1 violation(s): 1.4.10 (AA)

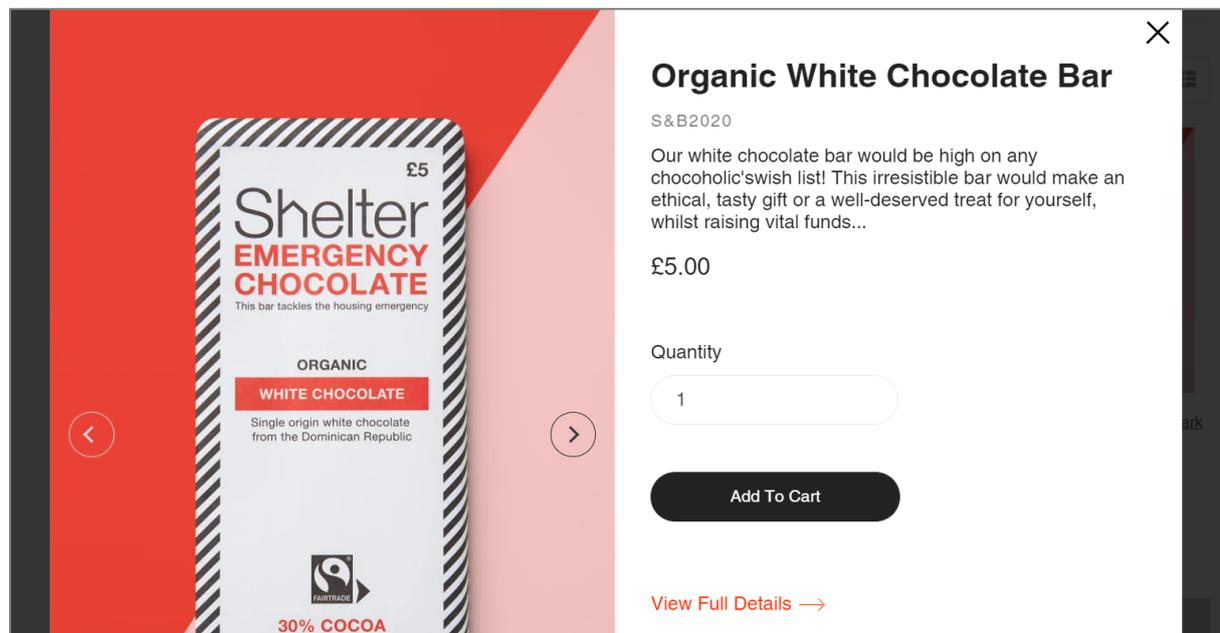
Description:

Page content is missing when the page is magnified to 300% or greater using browser zoom functionality. The modal's content is absent when zooming to 300% or greater.

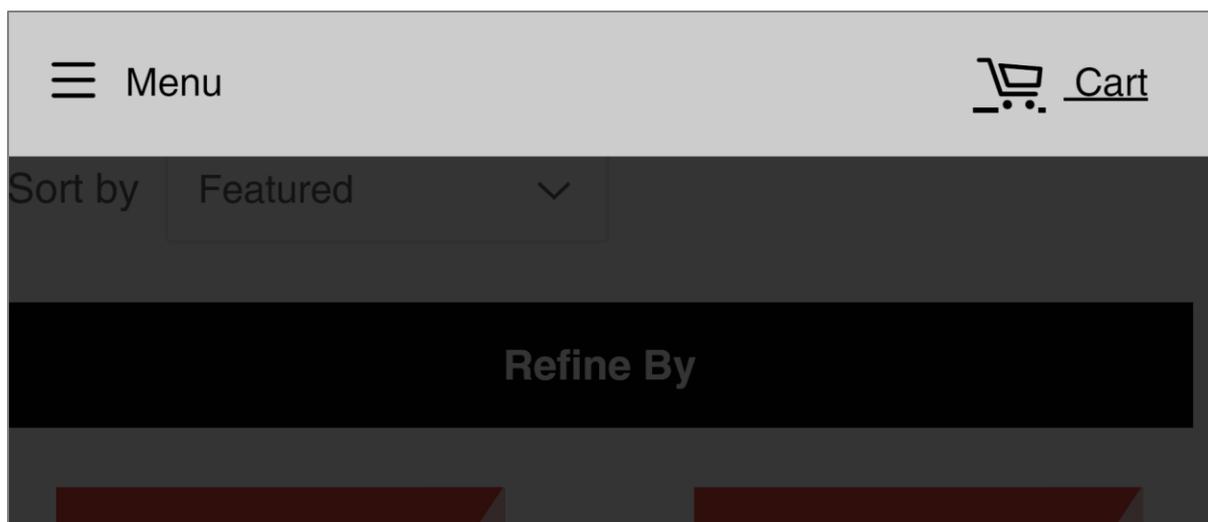
Impact on users:

Users with visual impairments who need to zoom the page, in order to understand content fully may have difficulty continuing the journey as the page is missing content.

Screenshots:



Online shop collection (100% magnification)



Online shop collection (300% magnification) – modal content missing

Recommendations:

Ensure all content is usable, when the browser's feature is zoomed 300%, without dropping any content or functions. See:

<http://www.w3.org/WAI/WCAG21/Understanding/reflow#techniques>

<https://www.w3.org/WAI/WCAG21/Techniques/general/G179.html>

For **working example** see:

<https://luna.sainsburys.co.uk/components/modal#examples>

Also, for **working examples**, see the modals on the **Online shop Information** page. These resize correctly without loss of functionality.

4.1.17 **Issue: Status messages not conveyed to assistive technology**

Priority level: Must

Occurrences: Online shop cart

Issue on mobile: Yes

WCAG 2.1 violation(s): 4.1.3 (A)

Description:

On the Cart page, the status message, 'Your cart is currently empty', is injected into the page. It appears in the middle of the page when an item has been removed from the shopping cart. This message is not conveyed to assistive technology on removing an item. The message only appears after removing all items from the cart.

Also, when various items are listed in the cart and when one of these has been removed from the cart, visually a progress indicator appears, briefly, on the page and then the item is removed from the page (it no longer appears in the list). This confirmation is also not conveyed to the screen reader user.

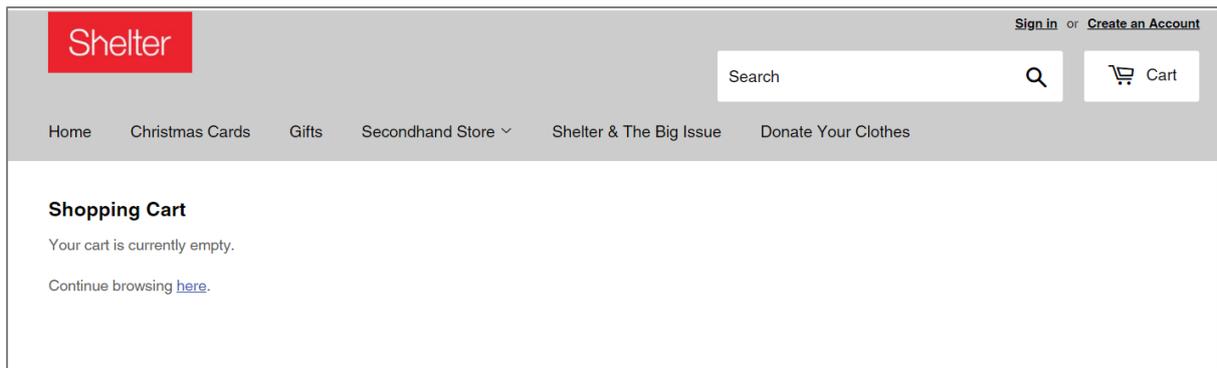
Impact on users:

Assistive technology users will be unaware of the presence of the status message.

Some users, such as cognitive impaired users may not realise that a status message has appeared.

Also, when one item is removed from a list of many items in the cart, screen reader users may be unaware that their action was successful without exploring the list of items remaining.

Screenshots:



Online shop cart - status message not conveyed

Recommendations:

Ensure content that is added dynamically, such as status messages, are conveyed to assistive technology users.

See: <https://www.w3.org/TR/wai-aria-practices-1.1/#alert>

4.2 Desktop issues - Should

4.2.1 Issue: Decorative images missing alt attribute

Priority level: Should

Occurrences: All pages: Shelter; Online shop: Homepage, Collection, Product, Cart

Issue on mobile: Yes

WCAG 2.1 violation(s): 1.1.1 (A)

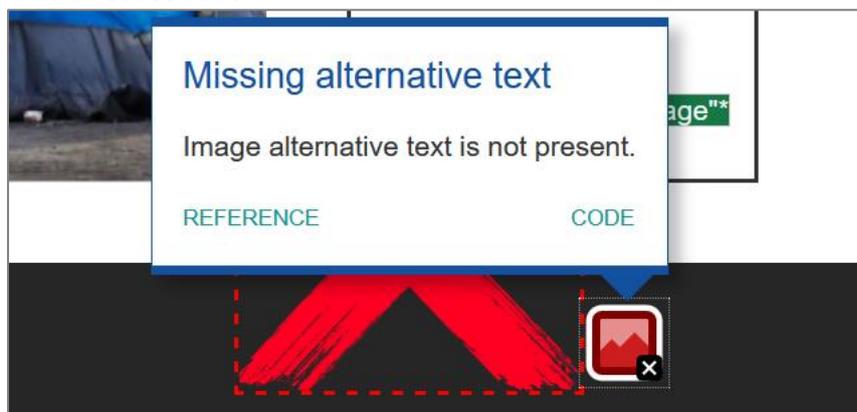
Description:

There are instances where an image that does not convey information (decorative) does not have an appropriate alt attribute. Decorative images must have an empty alt attribute to enable assistive technology to ignore them.

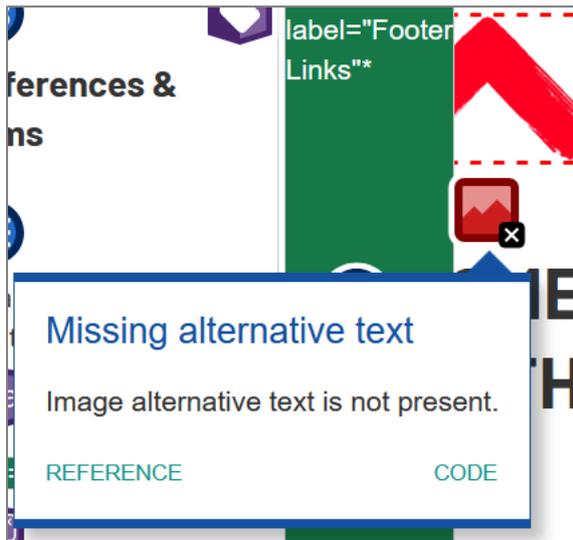
Impact on users:

Screen reader users will encounter images without alternative text and will not be able to understand their purpose.

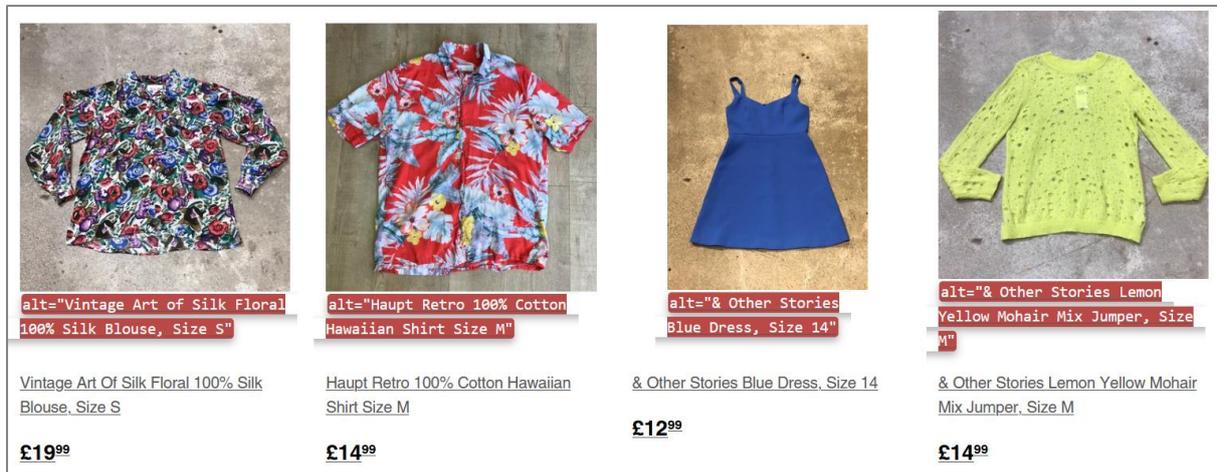
Screenshots:



Homepage – decorative image missing appropriate alt attribute



Homepage – decorative image missing appropriate alt attribute



Online shop homepage – decorative images missing appropriate alt attribute



Online shop product – decorative images missing appropriate alt attribute – also all images have the same alt text

Recommendations:

Ensure all decorative images have an empty alt attribute (`alt=""`).

Note: If an image is deemed part of the logo and/or branding, then give the image an alternative text (Shelter homepage).

For further information, see:

<https://www.w3.org/WAI/tutorials/images/decorative/>

<https://www.w3.org/WAI/tutorials/images/informative/>

4.2.2 Issue: Low colour contrast

Priority level: Should

Occurrences: Donate, iRaiser (Donate to Shelter), Support Us; Online shop: homepage, sign up, collection (modal), product, cart, information, review; Almost all pages

Issue on mobile: Yes

WCAG 2.1 violation(s): 1.4.3 (AA)

Description:

There are parts that do not have enough contrast between foreground text and background colours. These colour combinations fall below the minimum ratio.

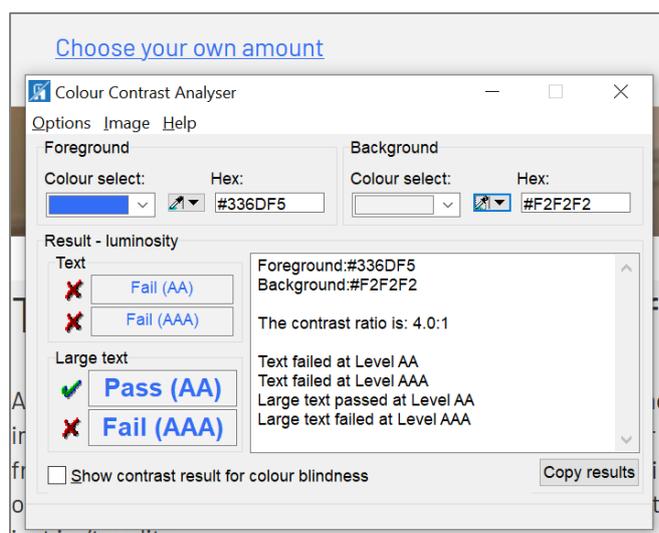
Note: Minimum ratio for text 18pt/14pt bold is 3:1. Below 14pt bold the minimum ratio is 4.5:1.

Impact on users:

Text with low colour contrast makes it more difficult to read for users with visual impairments.

Screenshots:

Note: This is not an exhaustive list



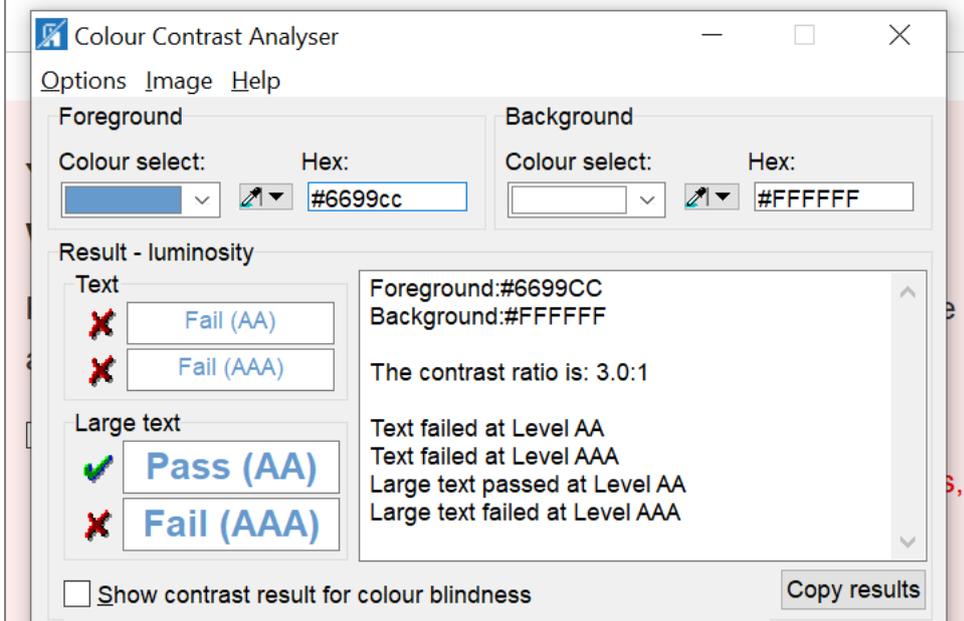
Donate – Contrast failure

Be careful, this payment form is using a **TEST** account (Ingenico)

[Dear donor if you are seeing this message, click here in order to inform the website administrator](#)

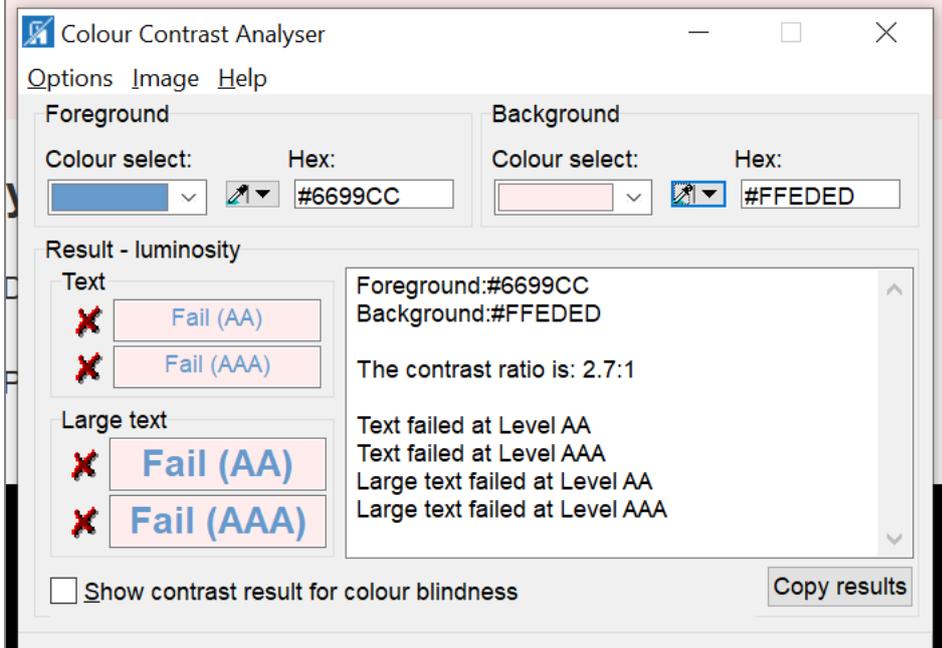
iRaiser (Donate to Shelter) – Contrast failure – perhaps this banner is only available in the development portal

You can change your preferences at any time by giving us a call on [0300 330 1234](tel:03003301234), or emailing info@shelter.org.uk. If you've already told us you want to receive updates, we'll keep contacting you in the same ways. And don't worry – although Shelter and Shelter Trading activities will use your details, we will never pass them on to third parties for marketing. [Read our privacy policy](#) for more details.



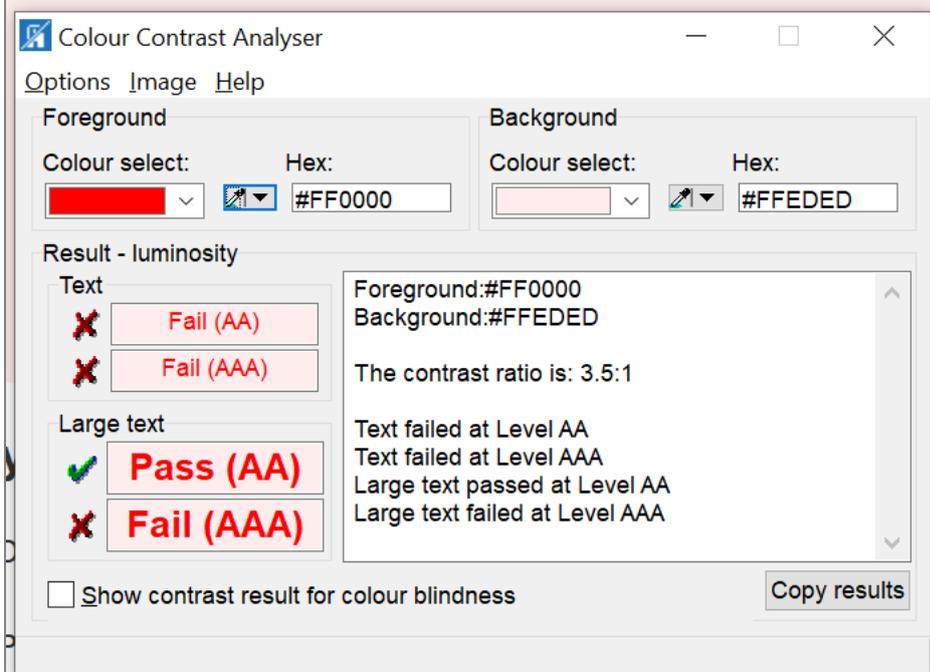
iRaiser (Donate to Shelter) – Contrast failure

cancel your declaration, please contact us on [0300 330 1234](tel:03003301234) so that we can adjust our records. [Find out more.](#)

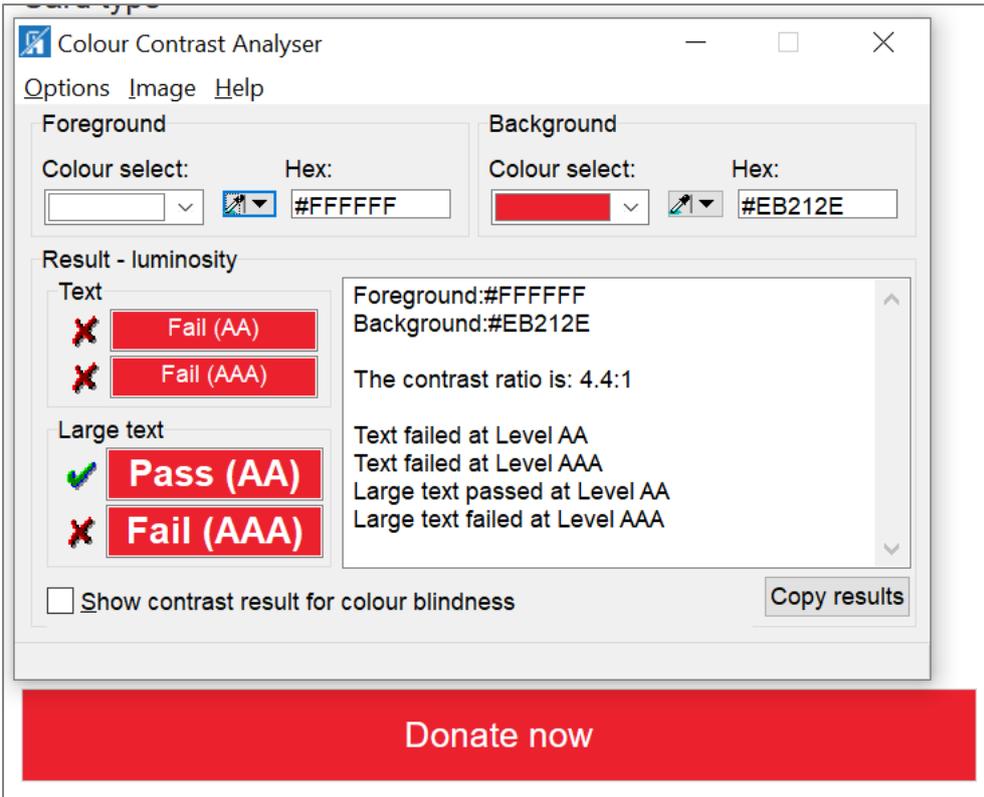


iRaiser (Donate to Shelter) – Contrast failure

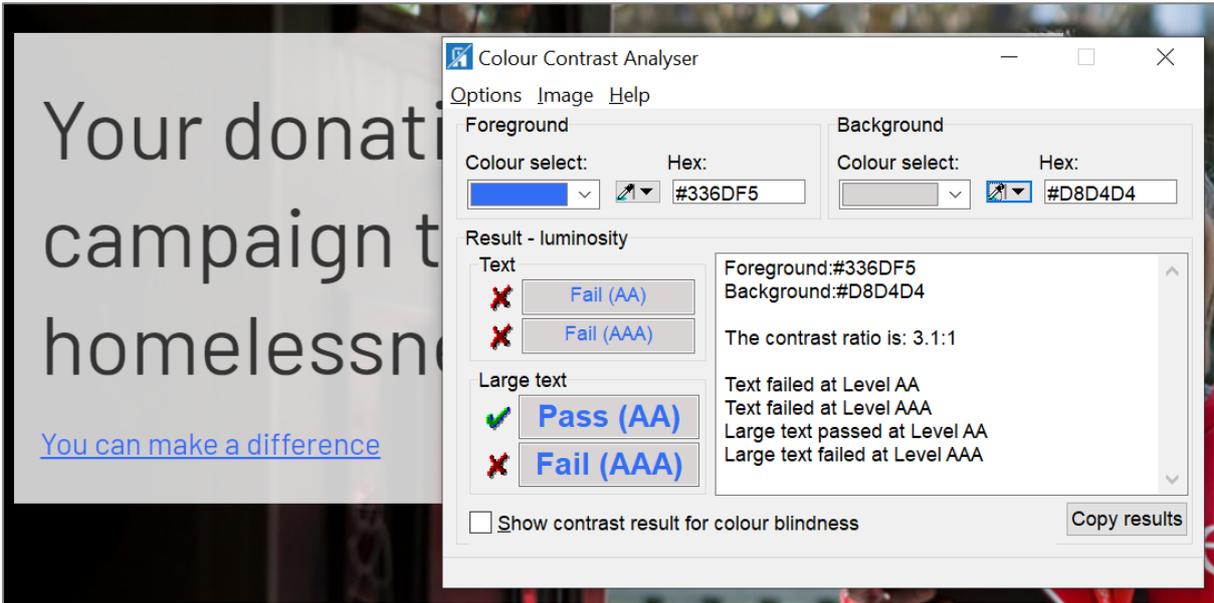
I would like Shelter to reclaim the tax I have paid on this donation, any donation I have made in the previous four years, and any donations I may make in the future.



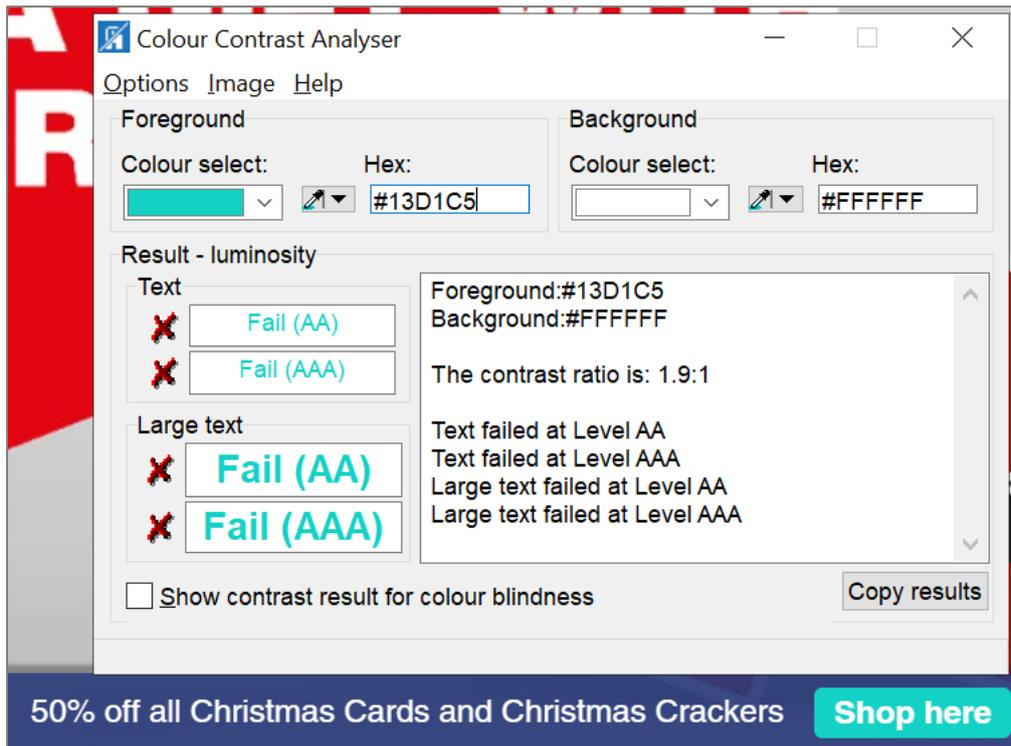
iRaiser (Donate to Shelter) – Contrast failure



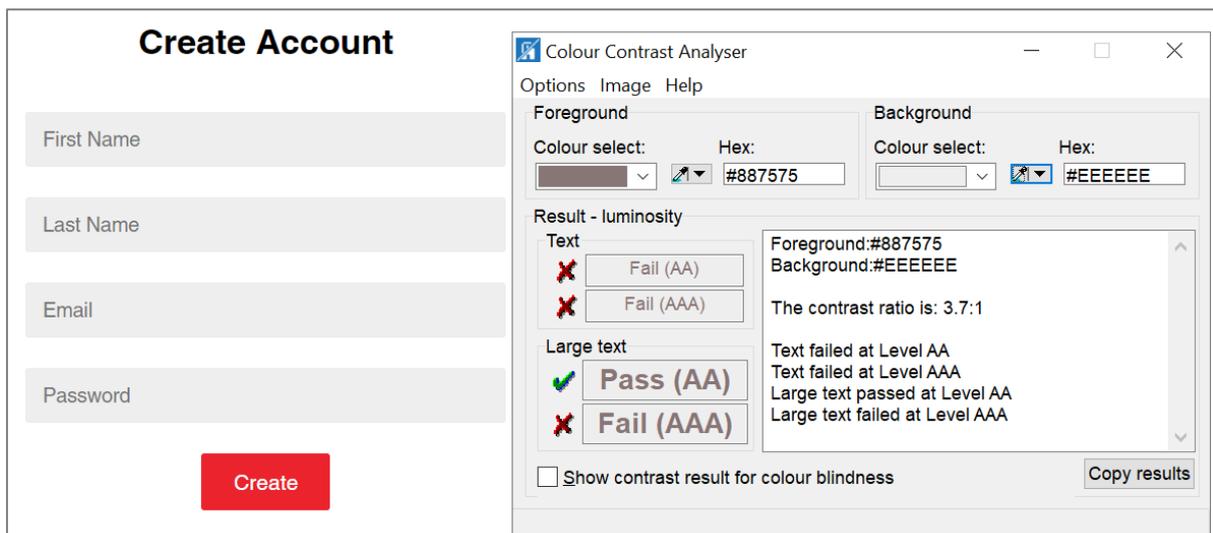
iRaiser (Donate to Shelter) – Contrast failure



Support Us – Contrast failure



Online shop homepage – Contrast failure



Online shop captcha – placeholder text - Contrast failure

Recommendations:

Ensure foreground text and background colours meet the minimum contrast ratio.

You can check contrast using the Colour Contrast Analyser tool:

<https://developer.paciellogroup.com/resources/contrastanalyser/>

Or select a compliant colour scheme, if necessary, with:

<http://colorsafe.co/>

4.2.3 Issue: Low graphic colour contrast

Priority level: Should

Occurrences: Donate, Online shop: captcha, sign up, collection, information, review

Issue on mobile: Yes

WCAG 2.1 violation(s): 1.4.11 (AA)

Description:

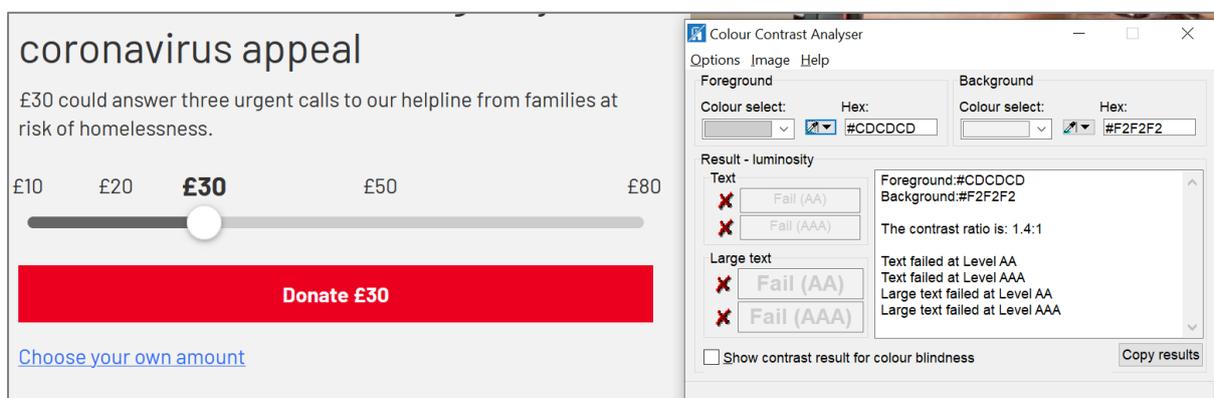
There are instances where user interface graphic components fall below the minimum ratio (3:1).

Impact on users:

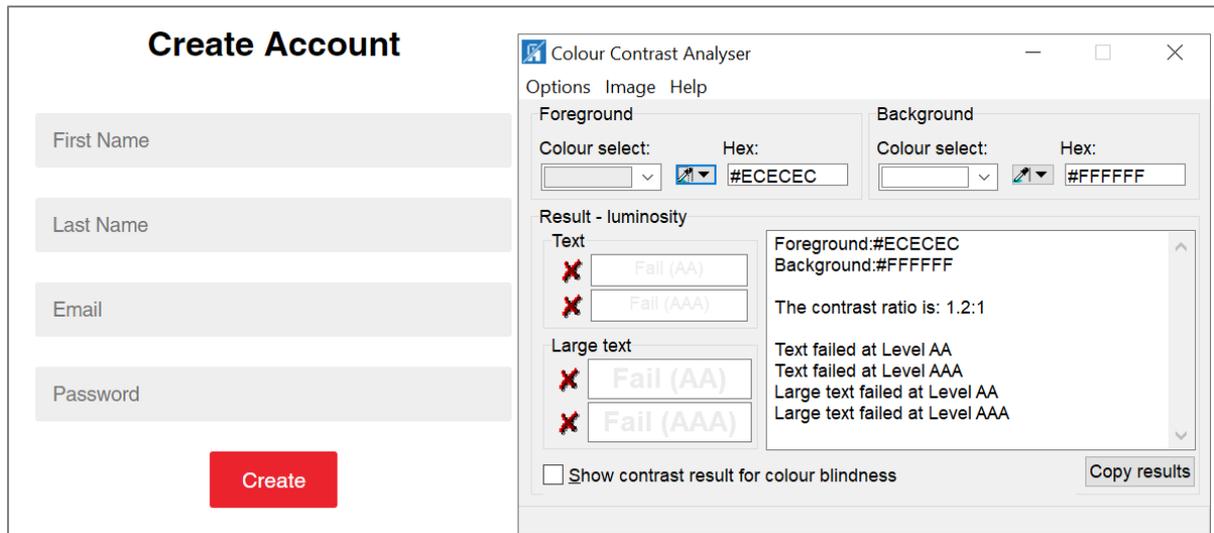
User interface graphic components with low contrast makes it more difficult to identify and understand how to interact with for users with visual impairments.

Screenshots:

Note: This is not an exhaustive list



Donate - Contrast failure on slider for draggable element & light grey incremental/disabled part of the bar



Online shop captcha – outline of input fields - Contrast failure

Recommendations:

Ensure the colour of user interface graphic elements meets the minimum contrast ratio.

You can check contrast using the Colour Contrast Analyser tool:

<https://developer.paciellogroup.com/resources/contrastanalyser/>

Or select a compliant colour scheme, if necessary, with:

<http://colorsafe.co/>

<http://contrast-finder.tanaguru.com/>

4.2.4 Issue: Error messages do not help users correct their errors

Priority level: Should

Occurrences: Homepage, iRaiser (Donate to Shelter); Online shop: all pages for footer form email sign up, sign up, information

Issue on mobile: Yes

WCAG 2.1 violation(s): 3.3.3 (AA)

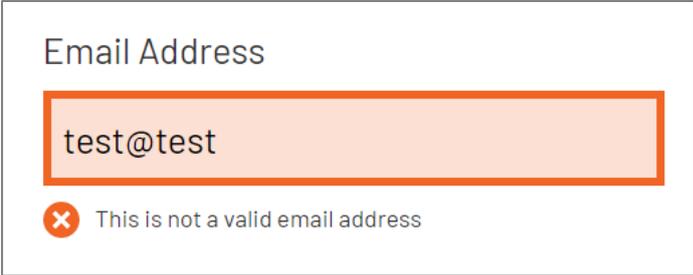
Description:

When a user makes a mistake with submitting a form, error messages are provided. However, the error message for the email input field does not provide enough information to help users correct the error.

Impact on users:

Users may find it difficult to correct their errors as the error message(s) do not provide the information to help them.

Screenshots:



The screenshot shows a form with the label "Email Address". The input field contains the text "test@test" and is highlighted with a thick orange border. Below the input field, there is an error message: a red circle with a white 'x' icon followed by the text "This is not a valid email address".

Homepage – form error messages not helpful

Your details

All fields are required except those marked as 'optional'

Email

I am making a donation on behalf of an organisation

Title

First name

Last name

Mobile number (optional)

iRaiser (Donate to Shelter) – form error messages not helpful

Email Address Sign Up

! Please fill in this field.

Online shop homepage – form error messages not helpful

Create Account

- The **password** can't be blank
- The **email** can't be blank

Online shop sign up – incorrect error messages

Recommendations:

Provide examples of correct information for data entries, e.g. give a text suggestion to rectify error. For example, if there is an error for the provided email then the error message can provide an example of a valid email address.

For further details please see:

<https://design-system.service.gov.uk/patterns/email-addresses/#error-messages>

<https://design-system.service.gov.uk/components/error-message/>

<https://design-system.service.gov.uk/components/error-summary/>

4.2.5 Issue: Missing autocomplete attribute on input field

Priority level: Should

Occurrences: All forms where applicable e.g. Homepage, iRaiser (Donate to Shelter); Online shop sign up, information, payment

Issue on mobile: Yes

WCAG 2.1 violation(s): 1.3.5 (AA)

Description:

Input fields that collect user data do not have the autocomplete attribute (e.g. lists relevancy or from search history). Autocomplete helps cognitive impaired, motor impaired, and assistive technology users to complete forms more easily as common fields are prefilled with the user's data.

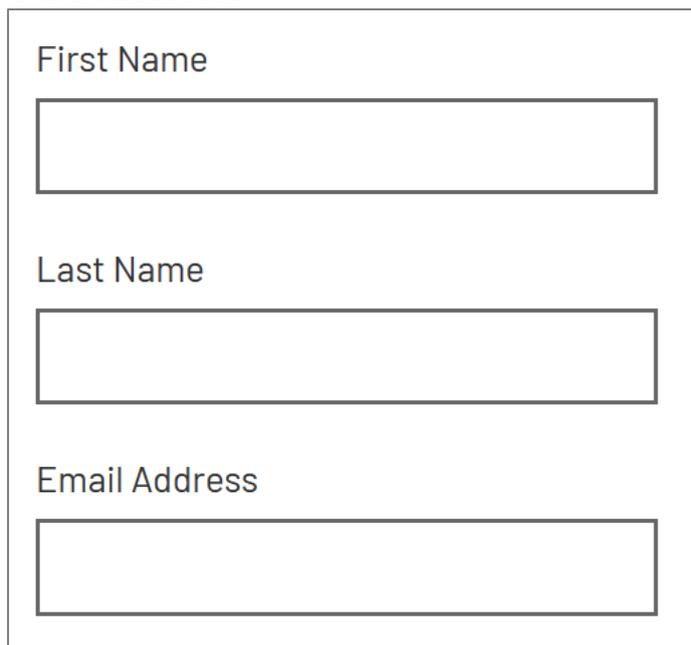
Autocomplete is only mandatory for fields that collect user data e.g. first name, last name, address, email, etc.

Note: this issue has been downgraded to a 'Should' priority as the mark-up uses HTML5 and autocomplete functionality can be determined by this.

Impact on users:

Users will not be able to benefit from autocomplete, which makes completing forms easier and faster as it predicts the rest of a word as the user types and helps them make more accurate search queries avoiding spelling mistakes.

Screenshots:



A screenshot of a web form with three input fields. The first field is labeled 'First Name', the second 'Last Name', and the third 'Email Address'. Each field is an empty rectangular box with a thin black border.

Homepage– missing autocomplete values

Recommendations:

Enable the autocomplete attribute on appropriate input fields. Allow the autocomplete attribute using valid HTML5 tokens:

<https://www.w3.org/TR/html52/sec-forms.html#sec-autofill>

<https://developer.mozilla.org/en-US/docs/Web/HTML/Attributes/autocomplete>

Types of personal data identified:

<https://www.w3.org/TR/WCAG21/#input-purposes>

For an example of an accessible autocomplete see:

<https://alphagov.github.io/accessible-autocomplete/examples/>

<https://luna.sainsburys.co.uk/components/autocomplete>

4.2.6 Issue: Autosuggest functionality is not exposed to the screen reader

Priority level: Should

Occurrences: iRaiser (Donate to Shelter), Online shop (search component)

Issue on mobile:

WCAG 2.1 violation(s): 2.1.1 (A), 4.1.2 (A)

Description:

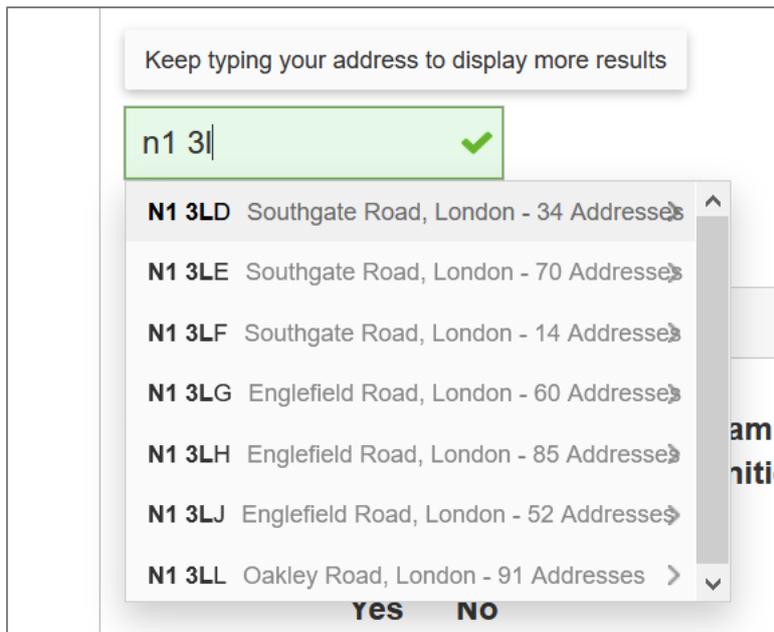
Autosuggest elements on the postcode field does not function correctly. When the field receives focus, the screen reader does not announce that this field has the autosuggest functionality. On inputting characters for a search its results presented was not announced by the screen reader. Keyboard and screen readers cannot tab to this list but can only arrow down to this list. Also, when the total number are found for a named street, this is inaccessible by keyboard, neither tabbing nor arrowing down.

On the Online shop, the search component's autosuggest functionality is also not announced to the screen reader user.

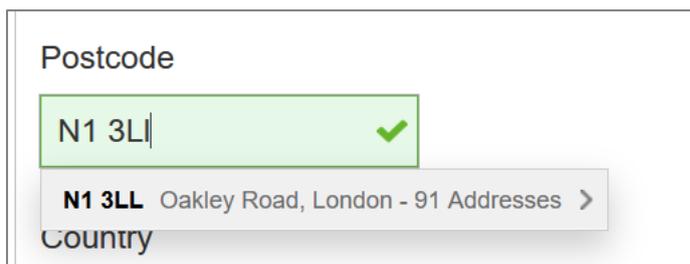
Impact on users:

Screen reader users will be unaware that an autosuggest list is available, e.g. for postcode and street name selection.

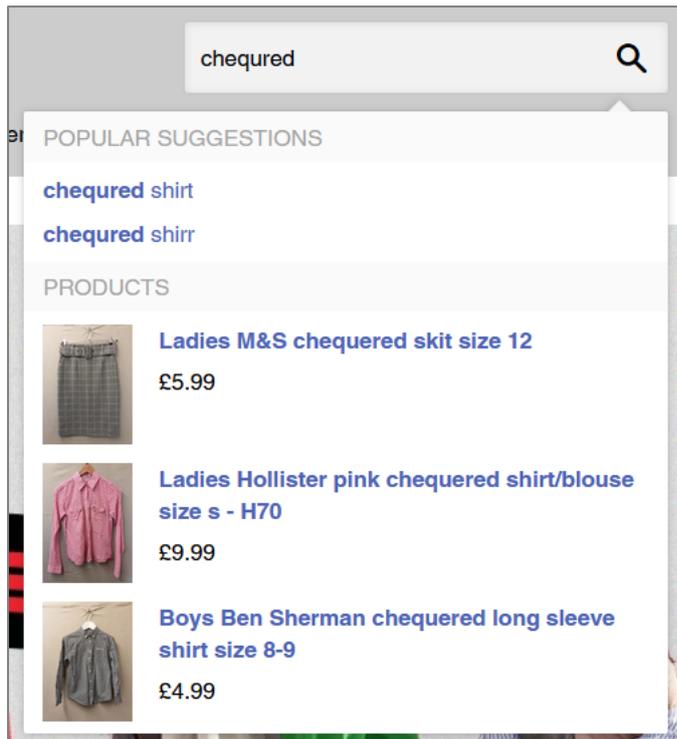
Screenshots:



iRaiser (Donate to Shelter) - autosuggest not announced for screen reader users



iRaiser (Donate to Shelter) - inaccessible autosuggest functionality for screen reader and keyboard users - When search narrow down to one street, unable to access suggestion by keyboard



Online shop homepage – search - autosuggest not announced for screen reader users

Recommendations:

Ensure screen reader users are made aware of the autosuggest result list. When navigating to this list ensure the items are announced to the screen reader.

Further information on making the autosuggest component accessible can be found at the following URLs:

<https://luna.sainsburys.co.uk/components/autocomplete>

<https://designnotes.blog.gov.uk/2017/04/20/were-building-an-autocomplete/>

<https://www.w3.org/TR/wai-aria-1.1/#ariahttps://www.w3.org/TR/wai-aria-1.1/ - aria-autocompleteautocomplete>

<https://www.w3.org/TR/wai-aria-1.1/#searchbox>

4.2.7 Issue: Fieldset missing legend

Priority level: Should

Occurrences: iRaiser (Donate to Shelter)

Issue on mobile: Yes

WCAG 2.1 violation(s): 1.1.1 (A), 1.3.1 (A), 3.3.2 (A)

Description:

There are instances where a fieldset has been used to group form fields that include a legend (<legend>). However, these are not identified by the screen reader on the first set of radio buttons. On the other set these are missing.

Also, the first set of radio buttons are not associated by their labels. This issue has been raised under, '[Missing form label](#)'. However, these are intermittently announced by the screen reader.

Impact on users:

A fieldset and legend helps screen reader users by providing context to a group of form fields. This context is lost when the fieldset does not include a descriptive legend.

Screenshots:

	Yes	No
Email	<input type="radio"/>	<input checked="" type="radio"/>
Phone	<input checked="" type="radio"/>	<input type="radio"/>
Text	<input type="radio"/>	<input checked="" type="radio"/>
Post	<input type="radio"/>	<input checked="" type="radio"/>

iRaiser - fieldset and legend not recognised by screen reader

Start date of your direct debit

<input checked="" type="radio"/> 25/4/2021	<input type="radio"/> 2/5/2021
<input type="radio"/> 9/5/2021	<input type="radio"/> 17/5/2021

iRaiser - fieldset missing legend

Recommendations:

Ensure all fieldset (<fieldset>) elements have a descriptive legend element (<legend>).

For further information on fieldset and native HTML:

<https://www.w3.org/WAI/tutorials/forms/grouping/>

<https://design-system.service.gov.uk/components/fieldset/>

<https://design-system.service.gov.uk/components/radios/>

4.2.8 Issue: Navigation lists not marked up correctly

Priority level: Should

Occurrences: Homepage

Issue on mobile: Yes

WCAG 2.1 violation(s): 1.3.1 (A)

Description:

On the homepage the navigation menu items are displayed as a list visually and are not announced to the screen reader as such. The list element is not structured correctly.

Impact on users:

Screen reader users might miss important information on the page if lists do not offer accurate information as they are not properly structured. They might find the page structure confusing and harder to navigate.

Screenshots:



Recommendations:

Ensure all functionality offer by lists coded in HTML are exposed to the screen reader user. Use the HTML ``, `` and `` elements to hold data in a list where visually they appear as a list. Screen reader software can then announce it correctly and tell their users up front how many items the list holds.

This issue requires further investigation to ascertain why the currently code list element is not exposed to the screen reader.

For further information see:

<http://www.w3.org/WAI/tutorials/page-structure/content/#lists>

<http://www.w3.org/TR/WCAG20-TECHS/H48.html>

For a **working example**, see the Online shop where this has been marked up correctly: <https://cardshop.shelter.org.uk/>

4.2.9 Issue: Role and state of accordion component not conveyed to assistive technology

Priority level: Should

Occurrences: Donate

Issue on mobile: Yes

WCAG 2.1 violation(s): 4.1.2 (A)

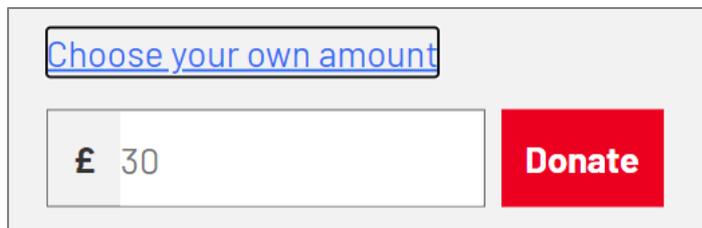
Description:

The role and the current state of the accordion component is not programmatically conveyed to assistive technology.

Impact on users:

Screen reader users will not be able to understand the purpose of the component, how to interact with it and may not be able to access the content it contains.

Screenshots:



Donate – Accordion component state not declared

Recommendations:

Ensure the role and current state of the component can be identified by assistive technology.

See: <https://www.w3.org/TR/wai-aria-practices-1.1/#accordion>

For **working examples:**

<https://design-system.service.gov.uk/components/accordion/>

<https://dequeuniversity.com/library/aria/tabpanels-accordions/sf-navigation>

4.2.10 Issue: Currently selected state is not announced by the screen reader

Priority level: Should

Occurrences: Online shop collection, All page: Shelter

Issue on mobile: Yes

WCAG 2.1 violation(s): 4.1.2 (A)

Description:

Users can use one of two icon links, to select the view of how the product items are displayed on a page, i.e. in a grid format of either four in a row or one per line. Visually, the user can tell:

- which selection state the page view is active, before making a selection
- when one of these links have been selected the page changes accordingly to the selected view.

For both these events, this information is not announced by the screen reader. No indication that the link setting, is currently selected, is given. Also, visually these two icon links are indistinguishable for the active state.

On the Shelter website, it has two versions: England and Scotland. On all pages when the current page is active, visually an underline appears on the country name. The active country, England, is not announced to the screen reader user.

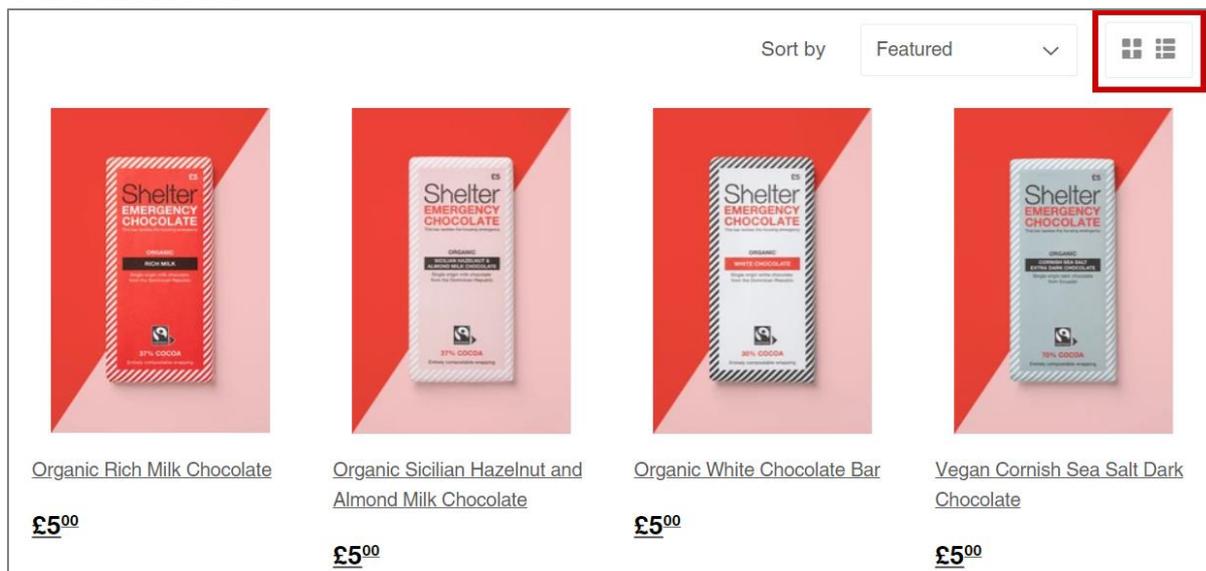
Impact on users:

Screen reader users will be unaware of the active display state. Also, when they have triggered one of the two icon links, they will not know if the action has changed the view as these icons do not offer a change in state. They will have to explore the page to gain this information.

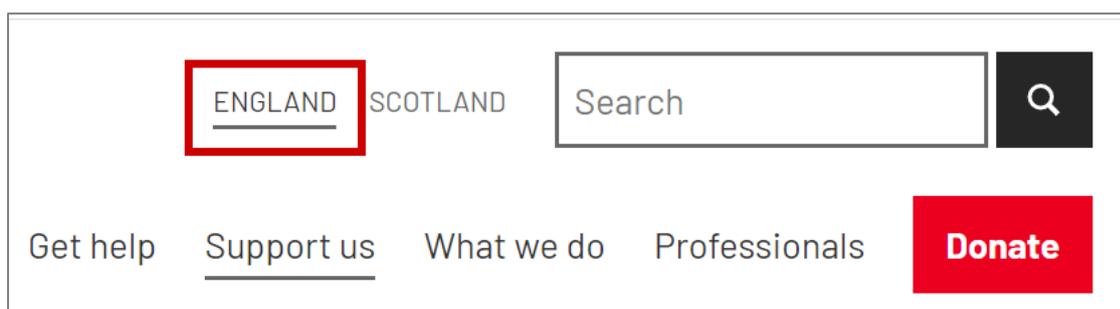
Also, some users, especially cognitive users, may not be able to differentiate the active state of these two icons as they appear the same.

On the Shelter website, screen reader users will not know that the current view of the page belongs to 'England', as it does visually.

Screenshots:



Online shop collection – Active state of display unknown and not announced



Shelter all pages – Current website/country not announced to screen reader users

Recommendations:

Where a series of related interface elements such as links or controls are present which control what content is visible on-screen, the currently selected or active element should be announced by screen readers. This should also be made visible to all users (e.g. on the collection page).

The `aria-current` attribute can be used to convey this visual information to non-visual users of screen readers. Add `aria-current` to the text which indicates the name of the page or step in the process.

The following resources provide more information:

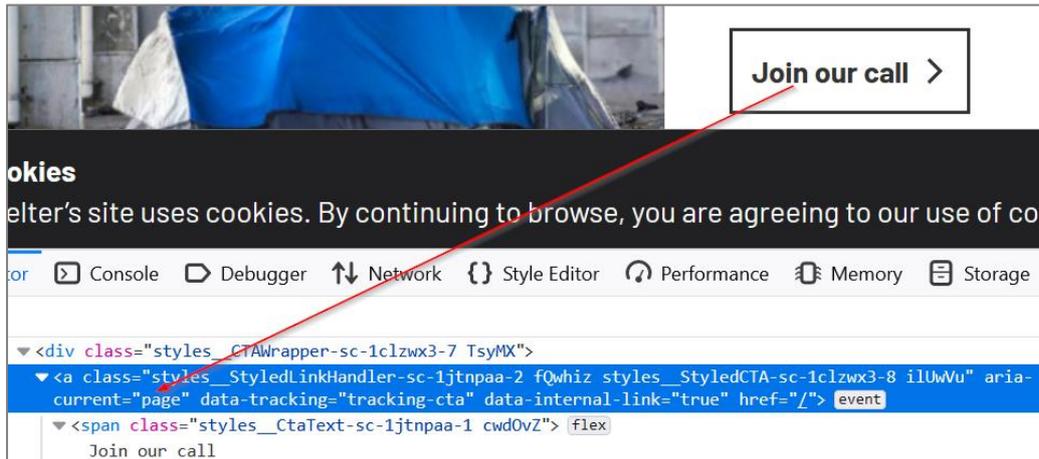
<https://tink.uk/using-the-aria-current-attribute/>

<http://design-patterns.tink.uk/aria-current/>

For a **working example**, please see how the current menu item is marked up:

<https://luna.sainsburys.co.uk/components/>

Also, for a **working example** see the Shelter homepage:



<https://shelter-rebrand-uat.netlify.app/>

4.2.11 Issue: Non-descriptive page titles

Priority level: Should

Occurrences: Donate, iRaiser (Donate to Shelter)

Issue on mobile: Yes

WCAG 2.1 violation(s): 2.4.2 (A)

Description:

The page title does not provide a clear description of the purpose of the page. The 'Donate' page has a similar name to 'Donate to Shelter'.

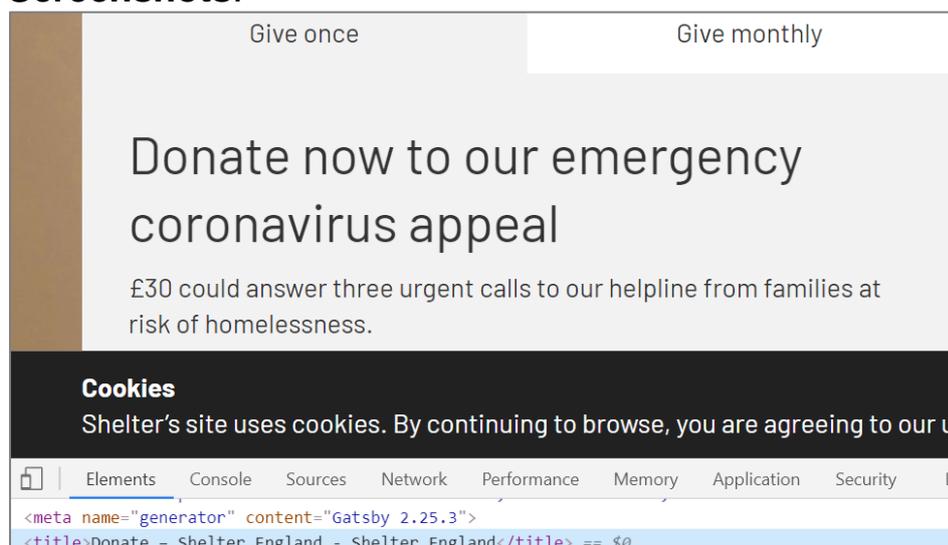
Also, the 'Thank you' page has the same title, 'Donate to Shelter', after submitting the iRaiser form. Rather than saying, e.g. 'Thanks for donation to Shelter'

Note: This similarity maybe due to the website been under development.

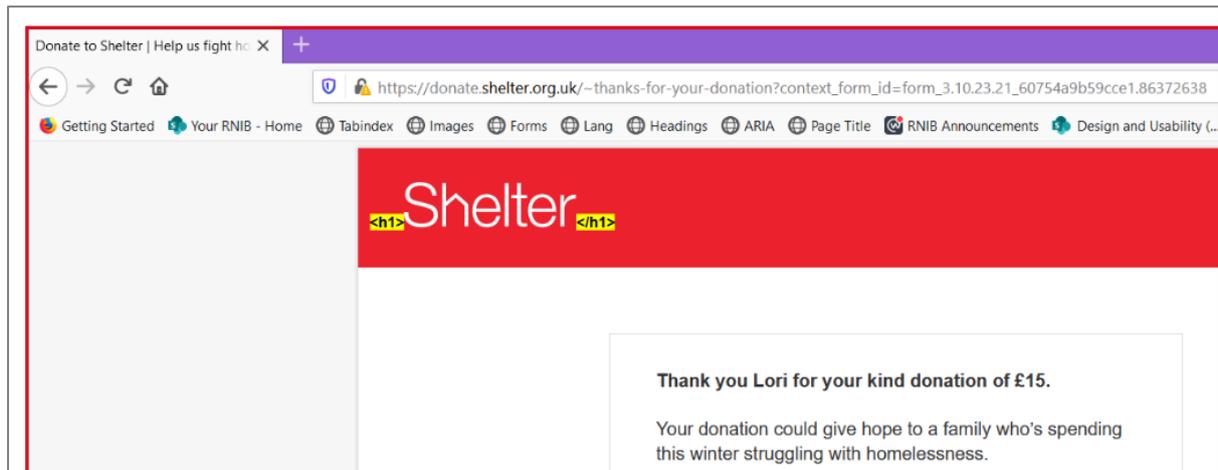
Impact on users:

Most users and especially screen reader users make use of the page title to assist in navigation. As the title doesn't describe the page accurately, users may find navigation more difficult.

Screenshots:



Donate - Non-descriptive page title – similar name to 'Donate to Shelter' page



iRaiser (Donate to Shelter) – incorrect page title

Recommendations:

Ensure all pages have a consistent, descriptive and unique page title (<title> element) and are succinct (below 70 characters)

For further details see:

<https://www.w3.org/WAI/WCAG21/Techniques/general/G88>

4.2.12 Issue: Use of reCAPTCHA (version 31 March 2020)

Priority level: Should

Occurrences: Online shop captcha

Issue on mobile: Yes

WCAG 2.1 violation(s): 1.1.1 (A)

Description:

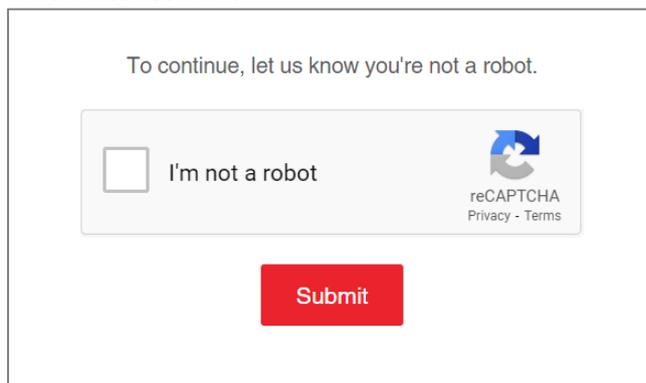
The form uses CAPTCHA (reCAPTCHA version 31 March 2020), which is required to submit a form (e.g. email sign up). The checkbox version of the CAPTCHA is accessible, however, under certain circumstances the CAPTCHA will revert to an image challenge instead of the default checkbox. Although the image challenge has an audio alternative, this can be difficult to use, especially for users with cognitive impairments.

Also, the outline of the checkbox interface graphic component falls below the minimum ratio (3:1).

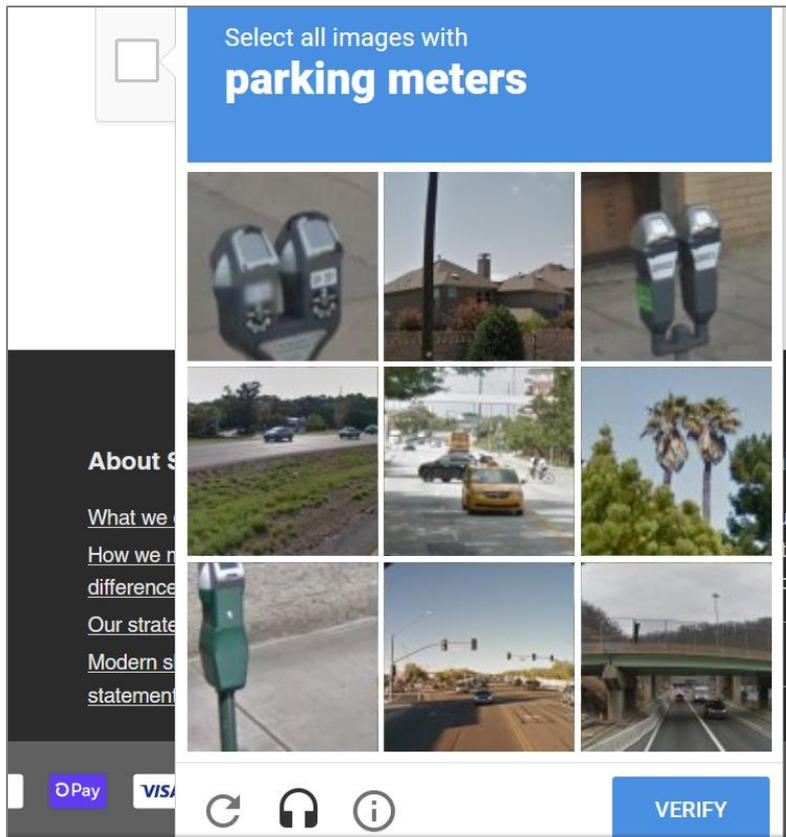
Impact on users:

Screen reader users may encounter an image challenge CAPTCHA, which can be difficult to complete successfully.

Screenshots:



online shop captcha – CAPTCHA (default checkbox)



online shop captcha – CAPTCHA (image challenge)

Recommendations:

If possible, avoid the use of CAPTCHA.

Alternatively, opt for Invisible CAPTCHA. See:

<https://developers.google.com/recaptcha/docs/invisible>

4.3 Desktop issues - Advisory

4.3.1 Issue: Generic link name

Priority level: Advisory

Occurrences: Homepage (Cookies modal), iRaiser (Donate to Shelter);
Online shop: collection, product, cart, information all pages (basket has item[s])

Issue on mobile: Yes

WCAG 2.1 violation(s): 2.4.4 (A)

Description:

Wherever possible, links/buttons and labels for call to action should be as unique and descriptive as possible. There are instances on the examined pages where generic link text was used.

Impact on users:

On the iRaiser, the link text 'find out more' is used. This is not descriptive and does not provide the user with clear information about where the link will take them.

Also, when screen reader users access the shortcut commands via their screen reader for links and buttons, a list of names will be presented which can be frustrating to use as it's not clear where the link will take them i.e. what 'find out more' is related to. They will need to navigate and read the surrounding text on the page. Or navigate to the prior heading to determine the context of this "find out more" link. If this pattern is repeated multiple times in a page, it can become quite frustrating and time consuming for a screen reader user (e.g. on the Collection page).

On the online shop when there are items in the cart, a number is displayed, e.g. "1". When the cart receives focus, only the number is announced, e.g. "cart one". Screen reader users will not know what this number relates to. There's no context that it's related to the basket.

On every page an icon of the shopping page and a number from 0 upwards is displayed.

Also, on the 'Cart' page, the link text 'Remove' is used. This is not descriptive and does not provide the user which item will be removed from the shopping basket.

Screenshots:

Cookies

Shelter's site uses cookies. By continuing to browse, you are agreeing to our use of cookies. [Find out more](#)

Cookies modal – all pages – generic link name - links not understandable out of context

If your name, address or tax status should change in the future, or you wish to cancel your declaration, please contact us on [0300 330 1234](tel:03003301234) so that we can adjust our records. [Find out more](#).

iRaiser (Donate to Shelter) – all pages – generic link name - links not understandable out of context

Your donation

Single donation of £10

[Change](#)

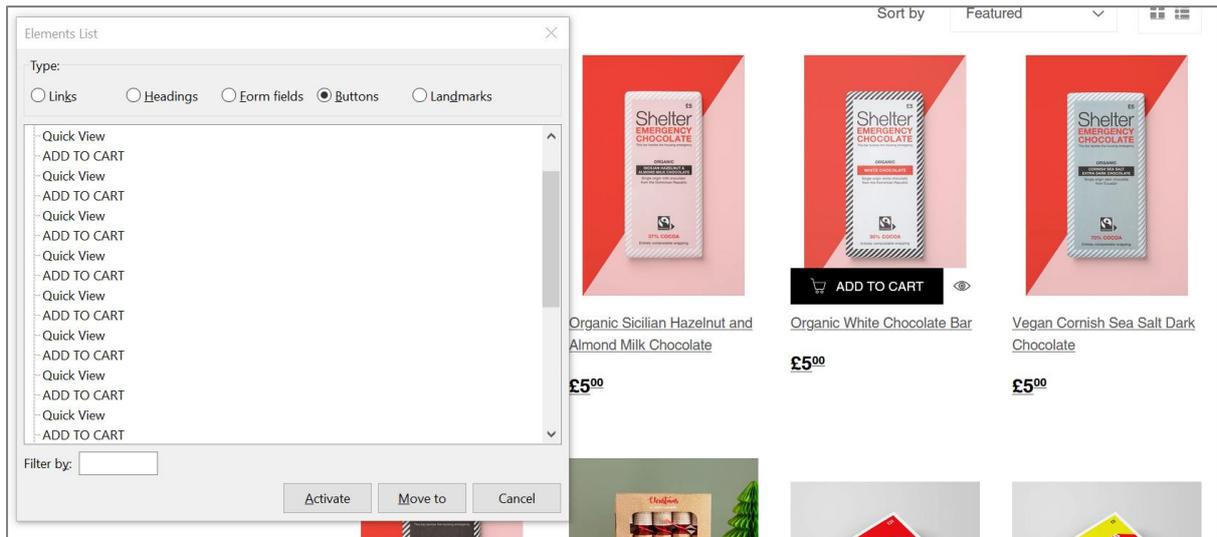
iRaiser (Donate to Shelter) – generic link name - links not understandable out of context

Your donation

Monthly donation of £

[Confirm change](#)

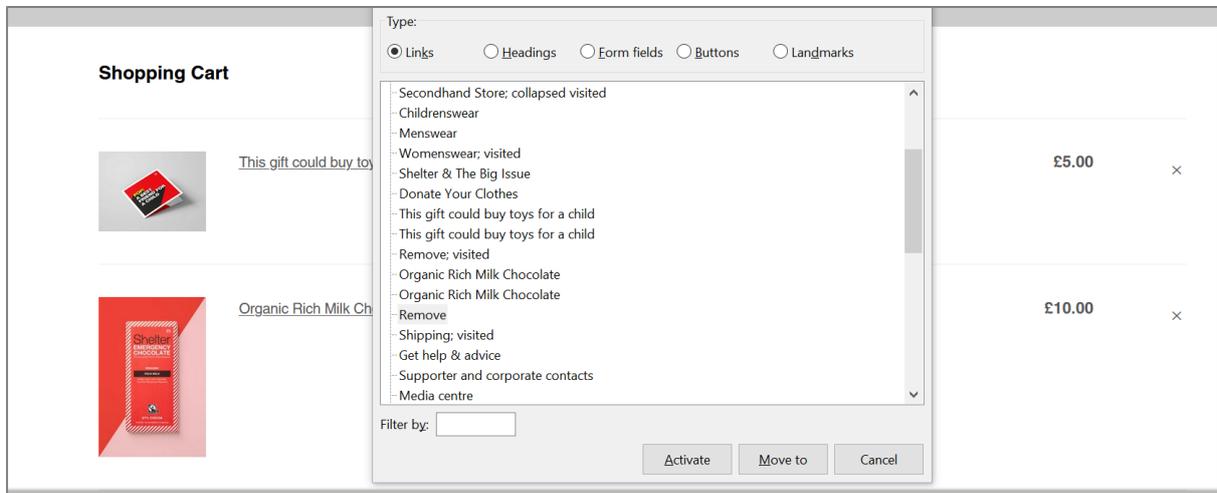
iRaiser (Donate to Shelter) – generic link name - links not understandable out of context



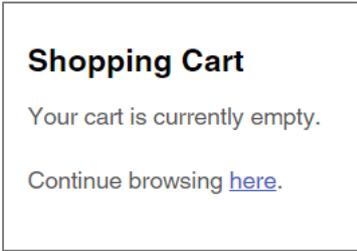
Online shop collection – Screen reader shortcut command view – generic button name links/buttons not understandable out of context



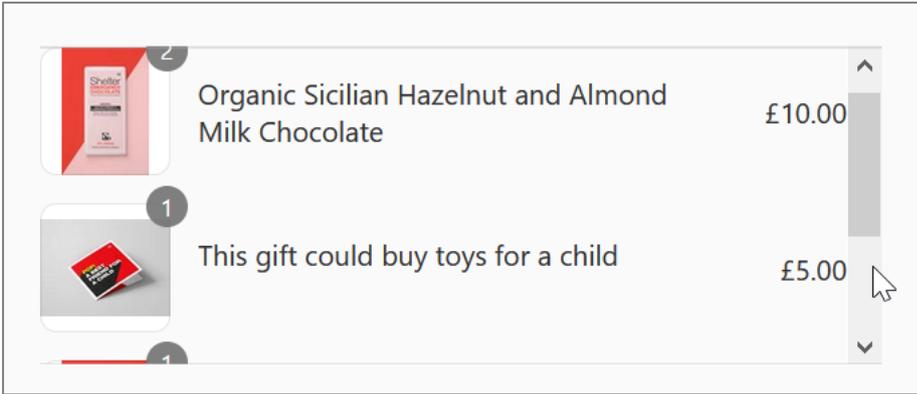
Online shop – all pages with items in basket – generic names used – announced as “cart one” (has no context)



Online shop cart – Screen reader shortcut command view – generic button name links/buttons not understandable out of context: “Remove”



Online shop cart – generic link name



Online shop information – generic link names

Recommendations:

The link text should be descriptive and provide the user with enough information as to where it will take them. For example, “Find out more to cancel your declaration or change your status”.

Alternatively, provide hidden label so screen reader users can understand the difference. For example, “Find out more [to cancel your declaration or change your status].”

See the below link for more information on hidden text for screen reader users:

<https://www.w3.org/WAI/tutorials/fundamentals/hiding/>

Invisible Content Just for Screen Reader Users - Techniques for hiding text:

<https://webaim.org/techniques/css/invisiblecontent/#techniques>

For further information on techniques of accessible links see:

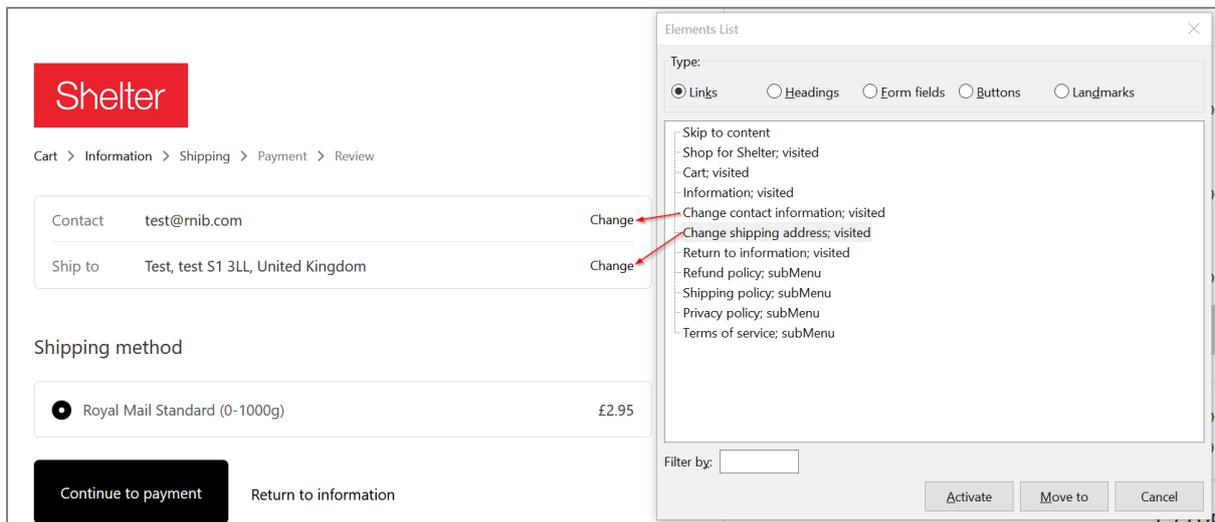
<https://webaim.org/techniques/hypertext/>

<https://www.w3.org/WAI/tips/writing/#make-link-text-meaningful>

For a **working example** of hidden text see the Online shop shipping/payment page where this has been implemented correctly:



Online shop shipping/payment



Online shop shipping/payment – screen reader shortcut command list view

4.3.2 Issue: Slider inaccessible by keyboard and screen reader

Priority level: Advisory

Occurrences: Donate

Issue on mobile: Yes

WCAG 2.1 violation(s): 1.3.1 (A), 2.1.1 (A), 1.4.3 (AA), 4.1.2 (A)

Description:

The Donate page has a slider component that allows mouse users to drag the slider element to donate from preset values. This slider element receives keyboard focus. Nonetheless, the component is inaccessible to keyboard and screen reader users to select from one of the amounts set.

However, there's an alternative method to manually donate an amount via 'Choose your amount' form just below the slider.

Also, the slider's control has low colour contrast. This issue has been raised under: ['Low graphic colour contrast'](#).

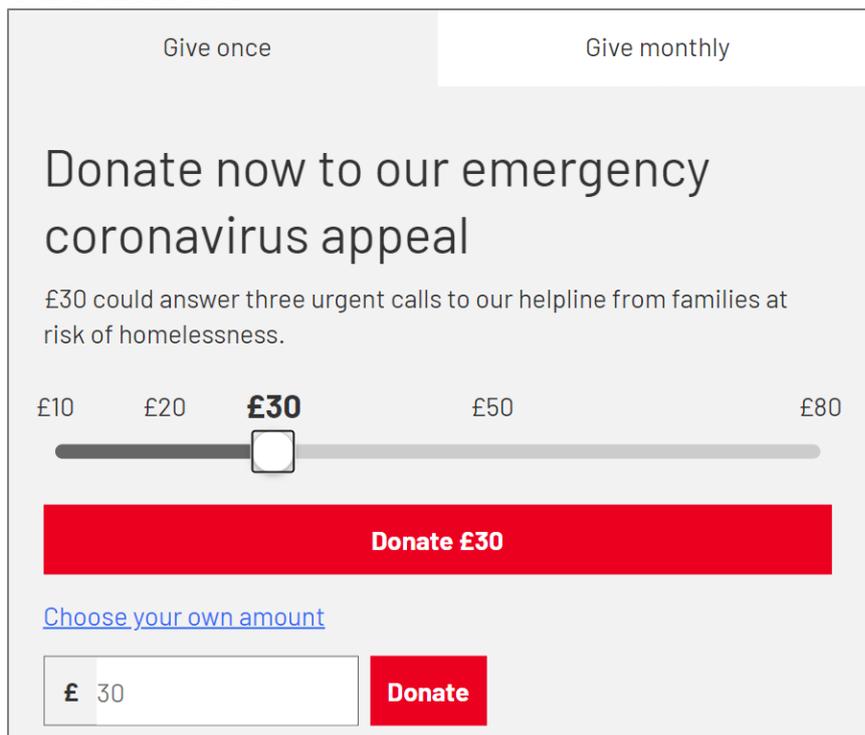
Note: This issue has been downgraded from a 'Must' to an 'Advisory' as an alternative method for this component is available, under the section 'Choose your amount'.

Impact on users:

Despite a text alternative available, some users who are restricted to using only a keyboard and/or screen reader may wish to use the slider component's functionality. Screen reader users are unaware that this feature is available.

Also, the slider controls with low colour contrast makes it more difficult to access for users with visual impairments.

Screenshots:



Give once Give monthly

Donate now to our emergency coronavirus appeal

£30 could answer three urgent calls to our helpline from families at risk of homelessness.

£10 £20 **£30** £50 £80

Donate £30

[Choose your own amount](#)

£ 30 **Donate**

Donate – slider

Recommendations:

Ensure all functionality and content that can be accessed and operated by mouse/touch can also be accessed and operated by keyboard only. This should also include:

- Managing focus control: when accessing the slider, the focus should be on the slider control
- Announcing the names, roles, states, and properties to the screen reader user
- Making it easy for people with low vision to see the slider's control

For a **working example** of an accessible **slider** component see:

<https://dequeuniversity.com/library/aria/custom-controls/sf-slider>

A **working example** can also be on the Online shop - Shelter Charity Christmas Gifts:

<https://cardshop.shelter.org.uk/collections/christmas-gifts>

4.3.3 Issue: Incorrect use of 'breadcrumb' component for screen reader users

Priority level: Advisory

Occurrences: Online shop: information to review

Issue on mobile: Yes

WCAG 2.1 violation(s): 1.3.1 (A)

Description:

On the checkout journey pages, the progress of the journey is indicated at the top of each page. It has a similar appearance to common breadcrumb navigation pattern with arrow icons used to separate navigation levels. The announcement of this component, by the screen reader is breadcrumb navigation. Its actual function is as a progress status indicator.

These are two different design patterns for different uses. Breadcrumb navigation is used to help users traverse pages grouped into sections. A progress indicator shows the users' progress through a set of pages, showing their current position and what pages are next in the journey.

Impact on users:

Screen reader users may be confused as the progress indicator is announced as a breadcrumb navigation. They may not understand the purpose of the progress indicator and its meaning.

Screenshots:



Cart > Information > Shipping > Payment > Review

Online shop: information to review – incorrect use of breadcrumb

Recommendations:

Ensure the progress status indicator is announce correctly by the screen reader.

For a **working example** of an accessible progress indicator, see:

<http://www.w3.org/WAI/tutorials/forms/multi-page/#using-step-by-step-indicator>

4.3.4 Issue: No announcement of link name when element receives focus by the screen reader

Priority level: Advisory

Occurrences: All pages: Online shop

Issue on mobile: No

WCAG 2.1 violation(s): 2.4.4 (A), 4.1.2 (A)

Description:

The link name, 'Sign in', is not announced by the screen reader when it receives keyboard focus. Tabbing forward to the next element and then back to the link, it's announced correctly.

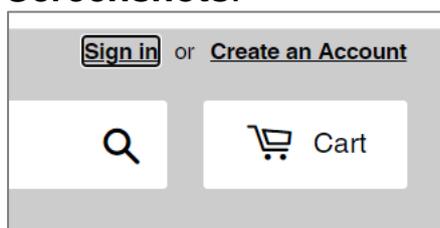
Impact on users:

By the lack of an announcement for the link, screen reader users will not know a Sign in link is available when it receives keyboard focus.

This can also be confusing for when the screen reader goes 'silent' on focusable elements. The screen reader user may have to explore the area to ascertain its purpose.

Tabbing backwards from the next focusable element is not an intuitive way of navigation.

Screenshots:



All pages – Online shop – link name not announced

Recommendations:

Ensure when links receive keyboard focus, their link name is announced by the screen reader.

This issue will require further investigation as to why the link name is only announced when tabbing backwards.

4.3.5 Issue: Incorrect ARIA references marked up for screen reader

Priority level: Advisory

Occurrences: Donate (broken ARIA), All pages (only use allowed ARIA attributes – review all page)

Issue on mobile: Yes

WCAG 2.1 violation(s): 1.3.1 (A), 4.1.2 (A)

Description:

On the Donate There are broken ARIA references where an `aria-labelledby` reference exists, but the target for the reference does not exist.

Also, on most of the pages reviewed, not all the 'role' of the ARIA attributes is allowed

Impact on users:

Screen reader users will not be able to announce the ARIA label or description if the element reference does not exist on the page.

Using ARIA attributes that are not allowed can restrict the accessibility of the page such as disabling whole parts of the application or having no effect on the accessibility of the application.

Screenshots:



Donate - – broken ARIA reference (x 2)

Recommendations:

Ensure the element referenced in the `aria-labelledby` attribute value is present within the page and presents a proper label or description.

See:

https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA/ARIA_Techniques/Using_the_aria-labelledby_attribute

<https://webaim.org/techniques/forms/advanced#labelledby>

Also, review all the pages, to ensure that each element with an ARIA role uses only ARIA attributes allowed for that role. Use appropriate ARIA roles, states, and properties

Information on ARIA roles and attributes can be found at the following URLs:

http://www.w3.org/TR/wai-aria-1.1/#state_prop_taxonomy
<http://www.w3.org/TR/wai-aria-1.1/#roles>

<https://www.w3.org/TR/wai-aria-1.1/#aria-hidden>

<https://www.w3.org/TR/html/editing.html#can-be-focused>

4.4 Mobile only issues - Must

4.4.1 Issue: Main menu items inaccessible with the screen reader

Priority level: Must

Occurrences: All pages (Shelter)

WCAG 2.1 violation(s): 1.3.1 (A), 2.1.1 (A), 4.1.2 (A)

Description:

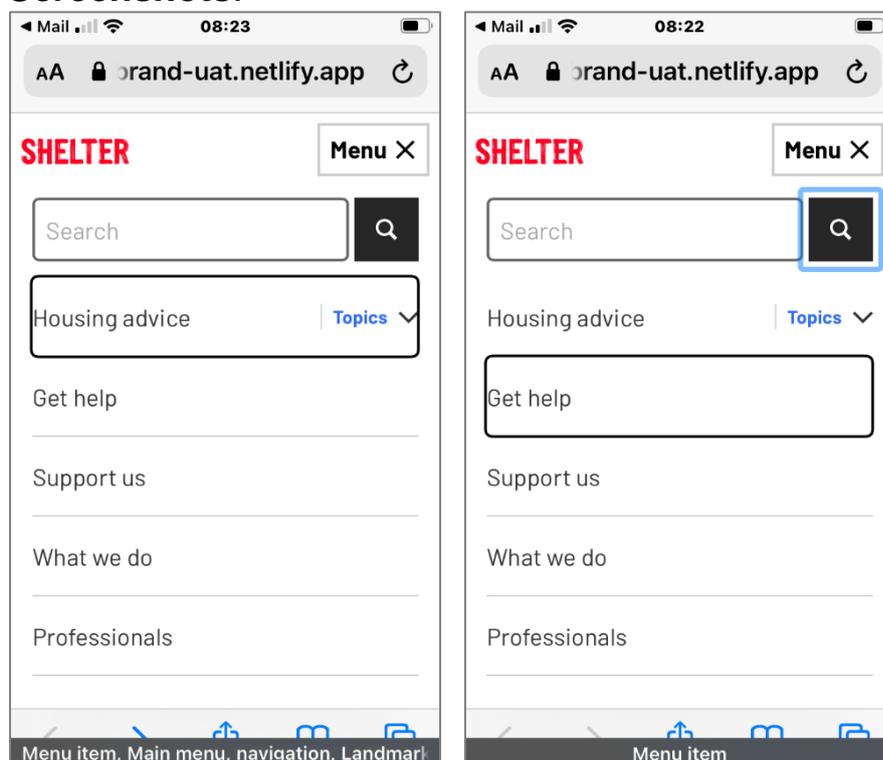
The main menu items when they receive focus cannot be operated when using the screen reader. Also, all the main menu items are announced as “menu item” rather than their actual name.

Note: the only main menu item accessible in the list is ‘Professionals’ and is announced correctly.

Impact on users:

Screen reader only users will not be able to interact with the main menu items to access the content of other pages of the website.

Screenshots:



All pages – main menu items inaccessible

Recommendations:

Ensure the main menu items can be operated by the screen reader (i.e. the link can be triggered to navigate to other parts of the website.). Also, when they receive focus the item name needs to be announced correctly

For a **working example** of an accessible navigation hamburger menu see:

<http://pauljadam.com/demos/hamburger-menu.html>

4.4.2 Issue: Hidden elements receive keyboard focus

Priority level: Must

Occurrences: All pages: Shelter and Online shop

WCAG 2.1 violation(s): 2.4.3 (A)

Description:

There are instances where elements that are visually hidden receives keyboard focus. This occurs when the site navigation panel is in the expanded state.

Also, when the main menu is in the collapsed state, the menu items receive focus when using the screen reader.

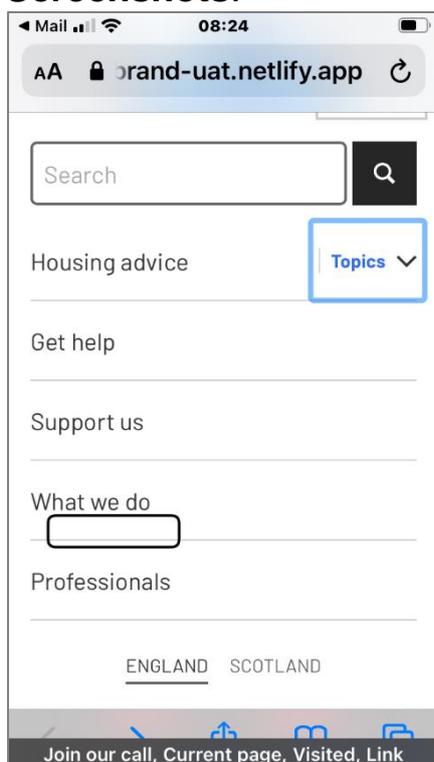
On the online shop, in the collapsed state, the menu items also receive focus when using the screen reader. However, in the expanded state the menu items are announced correctly and are accessible.

Impact on users:

Screen reader users could find it difficult to use the page's functionality. They can lose their position as the page is not contained. It is disorientating for screen reader users to encounter elements before they trigger a page to expose these elements.

Also, sighted users using a screen reader may be disorientated when accessing the page but cannot follow the Voiceover cursor on elements that are hidden.

Screenshots:



Homepage - main navigation – links receive focus when hidden

Recommendations:

Ensure elements that are visually hidden do not receive keyboard focus. See:

<https://www.w3.org/WAI/WCAG21/Techniques/html/H4.html>

When the panel is expanded, ensure focus is contained within the page. Swiping around should then only cycle through this panel until the user decides to close using the close button and/or the expanded/collapsed button on the menu.

To prevent screen reader software from focusing and announcing hidden elements, ensure that hidden elements are treated with `aria-hidden="true"`.

For further information on `aria-hidden="true"`, see:

https://a11y-guidelines.orange.com/web_EN/exemples/masquage/index.html

https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA/ARIA_Techniques/Using_the_aria-hidden_attribute

For a **working example** of a navigation main menu and carousel see:

<http://pauljadam.com/demos/hamburger-menu.html>

5 Further Information

5.1 What happens next?

5.1.1 Claiming WCAG 2.1 Conformance

This audit has highlighted issues that need to be corrected in order for the site to achieve WCAG 2.1 AA conformance – see Executive Summary for further details. You should arrange for the site to be re-checked once the necessary changes have been made.

5.1.2 Arranging for the site recheck

Once you have read the report and estimated how long it will take you to make the required changes to your website, please contact us to schedule the site recheck.

5.1.3 Making changes to a site which carries a WCAG 2.1 conformance claim

You are responsible for ensuring that any changes you make to your site do not reduce the standard of accessibility attained. If you are unsure, you can contact us for advice. Depending on the nature and extent of the changes you are proposing, there may be a charge for assessing these changes, to establish if there is any impact on the accessibility of the site and its continuing right to display a conformance claim. If you completely change the design of the site, you are strongly advised to arrange for a new audit to be carried out before the new site can display a WCAG 2.1 conformance claim.

6 Appendix 1: WCAG 2.1 guidelines and checkpoints for levels A and AA

6.1.1 Guideline 1.1: Text Alternatives

Provide text alternatives for any non-text content so that it can be changed into other forms people need, such as large print, braille, speech, symbols or simpler language.

1.1.1: Non-text Content (Level A)

- All non-text content that is presented to the user has a text alternative that serves the equivalent purpose, except for the situations listed below.
- Controls, Input: If non-text content is a control or accepts user input, then it has a name that describes its purpose. (Refer to Guideline 4.1 for additional requirements for controls and content that accepts user input.)
- Time-Based Media: If non-text content is time-based media, then text alternatives at least provide descriptive identification of the non-text content. (Refer to Guideline 1.2 for additional requirements for media.)
- Test: If non-text content is a test or exercise that would be invalid if presented in text, then text alternatives at least provide descriptive identification of the non-text content.
- Sensory: If non-text content is primarily intended to create a specific sensory experience, then text alternatives at least provide descriptive identification of the non-text content.
- CAPTCHA: 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.
- Decoration, Formatting, Invisible: If non-text content is pure decoration, 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 assistive technology.

6.1.2 Guideline 1.2: Time-based Media

Provide alternatives for time-based media.

1.2.1: Audio-only and Video-only (Pre-recorded) (Level A)

For pre-recorded audio-only and pre-recorded video-only media, the following are true, except when the audio or video is a media alternative for text and is clearly labelled as such:

- Pre-recorded Audio-only: An alternative for time-based media is provided that presents equivalent information for pre-recorded audio-only content.
- Pre-recorded Video-only: Either an alternative for time-based media or an audio track is provided that presents equivalent information for pre-recorded video-only content.

1.2.2: Captions (Pre-recorded) (Level A)

Captions are provided for all pre-recorded audio content in synchronized media, except when the media is a media alternative for text and is clearly labelled as such.

1.2.3: Audio Description or Media Alternative (Pre-recorded) (Level A)

An alternative for time-based media or audio description of the pre-recorded video content is provided for synchronized media, except when the media is a media alternative for text and is clearly labelled as such.

1.2.4: Captions (Live) (Level AA)

Captions are provided for all live audio content in synchronized media.

1.2.5: Audio Description (Pre-recorded) (Level AA)

Audio description is provided for all pre-recorded video content in synchronized media.

6.1.3 Guideline 1.3: Adaptable

Create content that can be presented in different ways (for example simpler layout) without losing information or structure.

1.3.1: Info and Relationships (Level A)

Information, structure, and relationships conveyed through presentation can be programmatically determined or are available in text.

1.3.2: Meaningful Sequence (Level A)

When the sequence in which content is presented affects its meaning, a correct reading sequence can be programmatically determined.

1.3.3: Sensory Characteristics (Level A)

Instructions provided for understanding and operating content do not rely solely on sensory characteristics of components such as shape, size, visual location, orientation, or sound.

Note: For requirements related to colour, refer to Guideline 1.4.

1.3.4 Orientation (Level AA) [WCAG 2.1]

Content does not restrict its view and operation to a single display orientation, such as portrait or landscape, unless a specific display orientation is essential.

Note: Examples where a particular display orientation may be essential are a bank check, a piano application, slides for a projector or television, or virtual reality content where binary display orientation is not applicable.

1.3.5 Identify Input Purpose (Level AA) [WCAG 2.1]

The purpose of each input field collecting information about the user can be programmatically determined when:

- The input field serves a purpose identified in the Input Purposes for User Interface Components section; and
- The content is implemented using technologies with support for identifying the expected meaning for form input data.

6.1.4 Guideline 1.4: Distinguishable

Make it easier for users to see and hear content including separating foreground from background.

1.4.1: Use of Colour (Level A)

Colour is not used as the only visual means of conveying information, indicating an action, prompting a response, or distinguishing a visual element.

Note: This success criterion addresses colour perception specifically. Other forms of perception are covered in Guideline 1.3 including programmatic access to colour and other visual presentation coding.

1.4.2: Audio Control (Level A)

If any audio on a Web page plays automatically for more than 3 seconds, either a mechanism is available to pause or stop the audio, or a mechanism is available to control audio volume independently from the overall system volume level.

Note: Since any content that does not meet this success criterion can interfere with a user's ability to use the whole page, all content on the Web page (whether or not it is used to meet other success criteria) must meet this success criterion. See Conformance Requirement 5: Non-Interference.

1.4.3: Contrast (Minimum) (Level AA)

The visual presentation of text and images of text has a contrast ratio of at least 4.5:1, except for the following:

- Large Text: Large-scale text and images of large-scale text have a contrast ratio of at least 3:1;
- Incidental: Text or images of text that are part of an inactive user interface component, that are pure decoration, that are not visible to anyone, or that are part of a picture that contains significant other visual content, have no contrast requirement.
- Logotypes: Text that is part of a logo or brand name has no minimum contrast requirement.

1.4.4: Resize text (Level AA)

Except for captions and images of text, text can be resized without assistive technology up to 200 percent without loss of content or functionality.

1.4.5: Images of Text (Level AA)

If the technologies being used can achieve the visual presentation, text is used to convey information rather than images of text except for the following:

- Customizable: The image of text can be visually customized to the user's requirements;
- Essential: A particular presentation of text is essential to the information being conveyed.

Note: Logotypes (text that is part of a logo or brand name) are considered essential.

1.4.10 Reflow (Level AA) [WCAG 2.1]

Content can be presented without loss of information or functionality, and without requiring scrolling in two dimensions for:

- Vertical scrolling content at a width equivalent to 320 CSS pixels;
- Horizontal scrolling content at a height equivalent to 256 CSS pixels.

Except for parts of the content which require two-dimensional layout for usage or meaning.

Note: 320 CSS pixels is equivalent to a starting viewport width of 1280 CSS pixels wide at 400% zoom. For web content which are designed to scroll horizontally (e.g. with vertical text), the 256 CSS pixels is equivalent to a starting viewport height of 1024px at 400% zoom.

Note: Examples of content which require two-dimensional layout are images, maps, diagrams, video, games, presentations, data tables, and interfaces where it is necessary to keep toolbars in view while manipulating content.

1.4.11 Non-text Contrast (Level AA) [WCAG 2.1]

The visual presentation of the following has a contrast ratio of at least 3:1 against adjacent colour(s):

- **User Interface Components:** Visual information required to identify user interface components and states, except for inactive components or where the appearance of the component is determined by the user agent and not modified by the author;

- **Graphical Objects:** Parts of graphics required to understand the content, except when a particular presentation of graphics is essential to the information being conveyed.

1.4.12 Text Spacing (Level AA) [WCAG 2.1]

In content implemented using markup languages that support the following text style properties, no loss of content or functionality occurs by setting all of the following and by changing no other style property:

- Line height (line spacing) to at least 1.5 times the font size;
- Spacing following paragraphs to at least 2 times the font size;
- Letter spacing (tracking) to at least 0.12 times the font size;
- Word spacing to at least 0.16 times the font size.

Exception: Human languages and scripts that do not make use of one or more of these text style properties in written text can conform using only the properties that exist for that combination of language and script.

1.4.13 Content on Hover or Focus (Level AA) [WCAG 2.1]

Where receiving and then removing pointer hover or keyboard focus triggers additional content to become visible and then hidden, the following are true:

- **Dismissible:** A mechanism is available to dismiss the additional content without moving pointer hover or keyboard focus, unless the additional content communicates an input error or does not obscure or replace other content;
- **Hoverable:** If pointer hover can trigger the additional content, then the pointer can be moved over the additional content without the additional content disappearing;
- **Persistent:** The additional content remains visible until the hover or focus trigger is removed, the user dismisses it, or its information is no longer valid.

Exception: The visual presentation of the additional content is controlled by the user agent and is not modified by the author.

Note: Examples of additional content controlled by the user agent include browser tooltips created through use of the HTML title attribute.

Note: Custom tooltips, sub-menus, and other nonmodal popups that display on hover and focus are examples of additional content covered by this criterion.

6.1.5 Guideline 2.1: Keyboard Accessible

Make all functionality available from a keyboard.

2.1.1: Keyboard (Level A)

All functionality of the content is operable through a keyboard interface without requiring specific timings for individual keystrokes, except where the underlying function requires input that depends on the path of the user's movement and not just the endpoints.

Note 1: This exception relates to the underlying function, not the input technique. For example, if using handwriting to enter text, the input technique (handwriting) requires path-dependent input but the underlying function (text input) does not.

Note 2: This does not forbid and should not discourage providing mouse input or other input methods in addition to keyboard operation.

2.1.2: No Keyboard Trap (Level A)

If keyboard focus can be moved to a component of the page using a keyboard interface, then focus can be moved away from that component using only a keyboard interface, and, if it requires more than unmodified arrow or tab keys or other standard exit methods, the user is advised of the method for moving focus away.

Note: Since any content that does not meet this success criterion can interfere with a user's ability to use the whole page, all content on the Web page (whether it is used to meet other success criteria or not) must meet this success criterion. See Conformance Requirement 5: Non-Interference.

2.1.4 Character Key Shortcuts (Level A) [WCAG 2.1]

If a keyboard shortcut is implemented in content using only letter (including upper- and lower-case letters), punctuation, number, or symbol characters, then at least one of the following is true:

- **Turn off:** A mechanism is available to turn the shortcut off;

- **Remap:** A mechanism is available to remap the shortcut to use one or more non-printable keyboard characters (e.g. Ctrl, Alt, etc);
- **Active only on focus:** The keyboard shortcut for a user interface component is only active when that component has focus.

6.1.6 Guideline 2.2: Enough Time

Provide users enough time to read and use content.

2.2.1: Timing Adjustable (Level A)

For each time limit that is set by the content, at least one of the following is true:

- Turn off: The user is allowed to turn off the time limit before encountering it; or
- Adjust: The user is allowed to adjust the time limit before encountering it over a wide range that is at least ten times the length of the default setting; or
- Extend: The user is warned before time expires and given at least 20 seconds to extend the time limit with a simple action (for example, "press the space bar"), and the user is allowed to extend the time limit at least ten times; or
- Real-time Exception: The time limit is a required part of a real-time event (for example, an auction), and no alternative to the time limit is possible; or
- Essential Exception: The time limit is essential and extending it would invalidate the activity; or
- 20 Hour Exception: The time limit is longer than 20 hours.

Note: This success criterion helps ensure that users can complete tasks without unexpected changes in content or context that are a result of a time limit. This success criterion should be considered in conjunction with Success Criterion 3.2.1, which puts limits on changes of content or context as a result of user action.

2.2.2: Pause, Stop, Hide (Level A)

For moving, blinking, scrolling, or auto-updating information, all of the following are true:

- Moving, blinking, scrolling: For any moving, blinking or scrolling information that (1) starts automatically, (2) lasts more than five seconds, and (3) is presented in parallel with other content, there is a mechanism for the user to pause, stop, or hide it unless the movement, blinking, or scrolling is part of an activity where it is essential; and
- Auto-updating: For any auto-updating information that (1) starts automatically and (2) is presented in parallel with other content, there is a mechanism for the user to pause, stop, or hide it or to control the frequency of the update unless the auto-updating is part of an activity where it is essential.

Note 1: For requirements related to flickering or flashing content, refer to Guideline 2.3.

Note 2: Since any content that does not meet this success criterion can interfere with a user's ability to use the whole page, all content on the Web page (whether it is used to meet other success criteria or not) must meet this success criterion. See Conformance Requirement 5: Non-Interference.

Note 3: Content that is updated periodically by software or that is streamed to the user agent is not required to preserve or present information that is generated or received between the initiation of the pause and resuming presentation, as this may not be technically possible, and in many situations could be misleading to do so.

Note 4: An animation that occurs as part of a preload phase or similar situation can be considered essential if interaction cannot occur during that phase for all users and if not indicating progress could confuse users or cause them to think that content was frozen or broken.

6.1.7 Guideline 2.3: Seizures

Do not design content in a way that is known to cause seizures.2.3

2.3.1: Three Flashes or Below Threshold (Level A)

Web pages do not contain anything that flashes more than three times in any one second period, or the flash is below the general flash and red flash thresholds.

Note: Since any content that does not meet this success criterion can interfere with a user's ability to use the whole page, all content on the Web page (whether it is used to meet other success criteria or not) must meet this success criterion. See Conformance Requirement 5: Non-Interference.

6.1.8 Guideline 2.4: Navigable

Provide ways to help users navigate, find content, and determine where they are.

2.4.1: Bypass Blocks (Level A)

A mechanism is available to bypass blocks of content that are repeated on multiple Web pages.

2.4.2: Page Titled (Level A)

Web pages have titles that describe topic or purpose.

2.4.3: Focus Order (Level A)

If a Web page can be navigated sequentially and the navigation sequences affect meaning or operation, focusable components receive focus in an order that preserves meaning and operability.

2.4.4: Link Purpose (In Context) (Level A)

The purpose of each link can be determined from the link text alone or from the link text together with its programmatically determined link context, except where the purpose of the link would be ambiguous to users in general.

2.4.5: Multiple Ways (Level AA)

More than one way is available to locate a Web page within a set of Web pages except where the Web Page is the result of, or a step in, a process.

2.4.6: Headings and Labels (Level AA)

Headings and labels describe topic or purpose.

2.4.7: Focus Visible (Level AA)

Any keyboard operable user interface has a mode of operation where the keyboard focus indicator is visible.

6.1.9 Guideline 2.5 Input Modalities

Make it easier for users to operate functionality through various inputs beyond keyboard.

2.5.1 Pointer Gestures (Level A) [WCAG 2.1]

All functionality that uses multipoint or path-based gestures for operation can be operated with a single pointer without a path-based gesture, unless a multipoint or path-based gesture is essential.

Note: This requirement applies to web content that interprets pointer actions (i.e. this does not apply to actions that are required to operate the user agent or assistive technology).

2.5.2 Pointer Cancellation (Level A) [WCAG 2.1]

For functionality that can be operated using a single pointer, at least one of the following is true:

- **No Down-Event:** The down-event of the pointer is not used to execute any part of the function;
- **Abort or Undo:** Completion of the function is on the up-event, and a mechanism is available to abort the function before completion or to undo the function after completion;
- **Up Reversal:** The up-event reverses any outcome of the preceding down-event;
- **Essential:** Completing the function on the down-event is essential.

Note: Functions that emulate a keyboard or numeric keypad key press are considered essential.

Note: This requirement applies to web content that interprets pointer actions (i.e. this does not apply to actions that are required to operate the user agent or assistive technology).

2.5.3 Label in Name (Level A) [WCAG 2.1]

For user interface components with labels that include text or images of text, the name contains the text that is presented visually.

Note: A best practice is to have the text of the label at the start of the name.

2.5.4 Motion Actuation (Level A) [WCAG 2.1]

Functionality that can be operated by device motion or user motion can also be operated by user interface components and responding to the motion can be disabled to prevent accidental actuation, except when:

- **Supported Interface:** The motion is used to operate functionality through an accessibility supported interface;
- **Essential:** The motion is essential for the function and doing so would invalidate the activity.

6.1.10 Guideline 3.1: Readable

Make text content readable and understandable. 3.1

3.1.1: Language of Page (Level A)

The default human language of each Web page can be programmatically determined.

3.1.2: Language of Parts (Level AA)

The human language of each passage or phrase in the content can be programmatically determined except for proper names, technical terms, words of indeterminate language, and words or phrases that have become part of the vernacular of the immediately surrounding text.

6.1.11 Guideline 3.2: Predictable

Make Web pages appear and operate in predictable ways. 3.2

3.2.1: On Focus (Level A)

When any component receives focus, it does not initiate a change of context.

3.2.2: On Input (Level A)

Changing the setting of any user interface component does not automatically cause a change of context unless the user has been advised of the behaviour before using the component.

3.2.3: Consistent Navigation (Level AA)

Navigational mechanisms that are repeated on multiple Web pages within a set of Web pages occur in the same relative order each time they are repeated, unless a change is initiated by the user.

3.2.4: Consistent Identification (Level AA)

Components that have the same functionality within a set of Web pages are identified consistently.

6.1.12 Guideline 3.3: Input Assistance

Help users avoid and correct mistakes.

3.3.1: Error Identification (Level A)

If an input error is automatically detected, the item that is in error is identified and the error is described to the user in text.

3.3.2: Labels or Instructions (Level A)

Labels or instructions are provided when content requires user input.

3.3.3: Error Suggestion (Level AA)

If an input error is automatically detected and suggestions for correction are known, then the suggestions are provided to the user, unless it would jeopardize the security or purpose of the content.

3.3.4: Error Prevention (Legal, Financial, Data) (Level AA)

For Web pages that cause legal commitments or financial transactions for the user to occur, that modify or delete user-controllable data in data storage systems, or that submit user test responses, at least one of the following is true:

1. Reversible: Submissions are reversible.
2. Checked: Data entered by the user is checked for input errors and the user is provided an opportunity to correct them.
3. Confirmed: A mechanism is available for reviewing, confirming, and correcting information before finalizing the submission.

6.1.13 Guideline 4.1: Compatible

Maximize compatibility with current and future user agents, including assistive technologies.

4.1.1: Parsing (Level A)

In content implemented using markup languages, elements have complete start and end tags, elements are nested according to their specifications, elements do not contain duplicate attributes, and any IDs are unique, except where the specifications allow these features.

Note: Start and end tags that are missing a critical character in their formation, such as a closing angle bracket or a mismatched attribute value quotation mark are not complete.

4.1.2: Name, Role, Value (Level A)

For all user interface components (including but not limited to: form elements, links and components generated by scripts), the name and role can be programmatically determined; states, properties, and values that can be set by the user can be programmatically set; and notification of changes to these items is available to user agents, including assistive technologies.

Note: This success criterion is primarily for Web authors who develop or script their own user interface components. For example, standard HTML controls already meet this success criterion when used according to specification.

4.1.3 Status Messages (Level AA) [WCAG 2.1]

In content implemented using markup languages, status messages can be programmatically determined through role or properties such that they can be presented to the user by assistive technologies without receiving focus.

7 Contact details

Accessibility Consultancy and User Experience Team
Royal National Institute of Blind People
Northminster House, Northminster, Peterborough, UK, PE1 1YN

Email: nppd@rnib.org.uk