

Issues for SSSC Project - Public Website



Report generated on 12 September 2025

[Skip table of contents](#)

Table of contents

[Executive summary](#)

[Issues by priority](#)

[Breakdown of issues](#)

[Additional issues by priority](#)

[Breakdown of Additional issues](#)

[Links and references](#)

[ADR_22749-33](#) Text is truncated at default size **MEDIUM**

[ADR_22749-29](#) Functional image has inappropriate text alternative **MEDIUM**

[ADR_22749-48](#) Informative image has no text alternative **MEDIUM**

[ADR_22749-4](#) Heading not marked up semantically **MEDIUM**

[ADR_22749-35](#) Text uses heading semantics inappropriately **MEDIUM**

[ADR_22749-46](#) Accordion heading not marked up semantically **MEDIUM**

[ADR_22749-51](#) Buttons not associated with visual group label **MEDIUM**

[ADR_22749-34](#) Off-screen elements included in reading order **MEDIUM**

[ADR_22749-28](#) Focus indication relies on colour alone **MEDIUM**

[ADR_22749-2](#) Reflow requires users to scroll in multiple directions (no exceptions) **MEDIUM**

[ADR_22749-7](#) Reflow causes loss of content **MEDIUM**

[ADR_22749-20](#) Reflow causes loss of functionality **MEDIUM**

[ADR_22749-25](#) Content triggered by mouse hover or keyboard focus is not dismissable **MEDIUM**

[ADR_22749-16](#) In-page links do not move focus to the correct location **MEDIUM**

[ADR_22749-13](#) Navigation skip link does not move screen reader focus **MEDIUM**

[ADR_22749-18](#) Focus order includes off-screen interactive elements **MEDIUM**

[ADR_22749-22](#) Focus does not return to triggering element **MEDIUM**

[ADR_22749-30](#) Focus not set onto modal dialog **MEDIUM**

[ADR_22749-38](#) Non-interactive elements receive keyboard focus **MEDIUM**

[ADR_22749-45](#) Focus is not managed appropriately when page is dynamically updated (Pages 3 and 9) **MEDIUM**

[ADR_22749-53](#) Inoperable interactive elements receive keyboard focus **MEDIUM**

[ADR_22749-54](#) Focus is not managed appropriately when page is dynamically updated (Page 8) **MEDIUM**

[ADR_22749-26](#) Label insufficiently descriptive **MEDIUM**

[ADR_22749-12](#) Interactive elements have no visible focus indicator **MEDIUM**

[ADR_22749-14](#) Interactive elements have no visible focus indicator (Recite Me) **MEDIUM**

[ADR_22749-49](#) Accessible name does not include visual name **MEDIUM**

[ADR_22749-23](#) Expandable element is not programmatically identified **MEDIUM**

[ADR_22749-32](#) Interactive elements with no accessible name **MEDIUM**

[ADR_22749-40](#) Current page is not programmatically identified **MEDIUM**

[ADR_22749-42](#) Disabled element is not programmatically identified **MEDIUM**

[ADR_22749-50](#) Pressed element is not programmatically identified **MEDIUM**

[ADR_22749-43](#) Loading message not programmatically determined **MEDIUM**

[ADR_22749-44](#) Search results not programmatically identified correctly **MEDIUM**

[ADR_22749-55](#) Search results not programmatically identified **MEDIUM**

- ADR_22749-36** Decorative icon not hidden from assistive technology **LOW**
- ADR_22749-8** Unordered list not marked up semantically (Pages 1-3) **LOW**
- ADR_22749-11** Ordered list not marked up semantically **LOW**
- ADR_22749-39** Description list not semantically declared **LOW**
- ADR_22749-52** Unordered list not marked up semantically (Page 8) **LOW**
- ADR_22749-41** Anchor tag with no href has no role **LOW**

Executive Summary

This report details the conformance of the project SSSC Project - Public Website.

Based on this evaluation, the project currently **meets** the target conformance standard.

This report has found that there are:

- **No high priority** issues
- **33 medium priority** issues (all resolved)
- **6 low priority** issues (all resolved)
- **1 additional** issue (all resolved)

In this version of the report, there are:

- **0 issues** marked as **OPEN**
- **0 issues** marked as **PARTIALLY FIXED**
- **39 issues** (and 1 additional issue) marked as **RESOLVED**

The [review summary section](#) gives an overview of the issues found, whilst detailed recommendations for these issues are covered in the [review breakdown](#).

The [additional issue review summary section](#) gives an overview of the additional issues found, whilst detailed recommendations for these issues are covered in the [additional issue breakdown](#). They are not strictly required by the target conformance standard, but project accessibility and usability would benefit if they were addressed.

Issues by Priority

These are issues which fail the target conformance standard. They require resolving for the project to be conformant with the standard.

[Skip priority breakdown](#)

High priority (0 issues still open)

Can't use: some users are excluded from using part of the product.

Specific user group(s) are excluded from using part of the product. For example, widgets cannot be opened with keyboard or critical visual information is not communicated by screen readers.

No issues at this priority level.






















Medium priority (0 issues still open)

Causing problems: some users will experience problems using the product.

Specific user group(s) will experience problems but they are not prevented from using the product.

Nonetheless, they may find accessing and operating content frustrating, confusing, or otherwise challenging. For example, expanding and collapsing elements do not communicate this behaviour to screen reader users.

Issue	Success criteria Users affected	Status
ADR_22749-29 <u>Functional image has inappropriate text alternative</u>	1.1.1: Non-text content [2.0 A] 	RESOLVED
ADR_22749-48 <u>Informative image has no text alternative</u>	1.1.1: Non-text content [2.0 A] 	RESOLVED
ADR_22749-4 <u>Heading not marked up semantically</u>	1.3.1: Info and Relationships [2.0 A] 	RESOLVED
ADR_22749-35 <u>Text uses heading semantics inappropriately</u>	1.3.1: Info and Relationships [2.0 A] 	RESOLVED
ADR_22749-46 <u>Accordion heading not marked up semantically</u>	1.3.1: Info and Relationships [2.0 A] 	RESOLVED
ADR_22749-51 <u>Buttons not associated with visual group label</u>	1.3.1: Info and Relationships [2.0 A] 	RESOLVED
ADR_22749-34 <u>Off-screen elements included in reading order</u>	1.3.2: Meaningful Sequence [2.0 A] 	RESOLVED
ADR_22749-28 <u>Focus indication relies on colour alone</u>	1.4.1: Use of Color [2.0 A] 	RESOLVED

Issue	Success criteria Users affected	Status
ADR_22749-2 Reflow requires users to scroll in multiple directions (no exceptions)	1.4.10: Reflow [2.1 AA] 	RESOLVED
ADR_22749-7 Reflow causes loss of content	1.4.10: Reflow [2.1 AA] 	RESOLVED
ADR_22749-20 Reflow causes loss of functionality	1.4.10: Reflow [2.1 AA] 	RESOLVED
ADR_22749-25 Content triggered by mouse hover or keyboard focus is not dismissable	1.4.13: Content on Hover or Focus [AA] 	RESOLVED
ADR_22749-16 In-page links do not move focus to the correct location	2.1.1: Keyboard [2.0 A] 	RESOLVED
ADR_22749-13 Navigation skip link does not move screen reader focus	2.4.1: Bypass Blocks [2.0 A] 	RESOLVED
ADR_22749-18 Focus order includes off-screen interactive elements	2.4.3: Focus Order [2.0 A] 	RESOLVED
ADR_22749-22 Focus does not return to triggering element	2.4.3: Focus Order [2.0 A] 	RESOLVED
ADR_22749-30 Focus not set onto modal dialog	2.4.3: Focus Order [2.0 A] 	RESOLVED
ADR_22749-38 Non-interactive elements receive keyboard focus	2.4.3: Focus Order [2.0 A] 	RESOLVED
ADR_22749-45 Focus is not managed appropriately when page is dynamically updated (Pages 3 and 9)	2.4.3: Focus Order [2.0 A] 	RESOLVED
ADR_22749-53 Inoperable interactive elements receive keyboard focus	2.4.3: Focus Order [2.0 A] 	RESOLVED
ADR_22749-54 Focus is not managed appropriately when page is dynamically updated (Page 8)	2.4.3: Focus Order [2.0 A] 	RESOLVED
ADR_22749-26 Label insufficiently descriptive	2.4.6: Headings and Labels [2.0 AA] 	RESOLVED
ADR_22749-12 Interactive elements have no visible focus indicator	2.4.7: Focus Visible [2.0 AA]  	RESOLVED
ADR_22749-14 Interactive elements have no visible focus indicator (Recite Me)	2.4.7: Focus Visible [2.0 AA]  	RESOLVED
ADR_22749-49 Accessible name does not include visual name	2.5.3: Label in Name [2.1 A] 	RESOLVED
ADR_22749-23 Expandable element is not programmatically identified	4.1.2: Name, Role, Value [2.0 A] 	RESOLVED
ADR_22749-32 Interactive elements with no accessible name	4.1.2: Name, Role, Value [2.0 A] 	RESOLVED

Issue	Success criteria Users affected	Status
ADR_22749-40 Current page is not programmatically identified	4.1.2: Name, Role, Value [2.0 A] 	RESOLVED
ADR_22749-42 Disabled element is not programmatically identified	4.1.2: Name, Role, Value [2.0 A] 	RESOLVED
ADR_22749-50 Pressed element is not programmatically identified	4.1.2: Name, Role, Value [2.0 A] 	RESOLVED
ADR_22749-43 Loading message not programmatically determined	4.1.3: Status Messages [2.1 AA] 	RESOLVED
ADR_22749-44 Search results not programmatically identified correctly	4.1.3: Status Messages [2.1 AA] 	RESOLVED
ADR_22749-55 Search results not programmatically identified	4.1.3: Status Messages [2.1 AA] 	RESOLVED

Low priority (0 issues still open)

Minor and cosmetic issues: something is wrong, but it won't affect users too much.

Specific user group(s) will experience minor problems. They are not excluded from the product and won't face major obstacles. Nonetheless, these issues will make interaction bothersome and the product seem unpolished. For example, a small number of cosmetic images with redundant alt text.

Note that our priorities are context-sensitive. For example, if a page is swamped with occurrences of the same low priority issue, it may be bumped to medium priority.

Issue	Success criteria Users affected	Status
ADR_22749-36 Decorative icon not hidden from assistive technology	1.1.1: Non-text content [2.0 A] 	RESOLVED
ADR_22749-8 Unordered list not marked up semantically (Pages 1-3)	1.3.1: Info and Relationships [2.0 A] 	RESOLVED
ADR_22749-11 Ordered list not marked up semantically	1.3.1: Info and Relationships [2.0 A] 	RESOLVED
ADR_22749-39 Description list not semantically declared	1.3.1: Info and Relationships [2.0 A] 	RESOLVED
ADR_22749-52 Unordered list not marked up semantically (Page 8)	1.3.1: Info and Relationships [2.0 A] 	RESOLVED
ADR_22749-41 Anchor tag with no href has no role	4.1.2: Name, Role, Value [2.0 A] 	RESOLVED

Breakdown of issues

ADR_22749-29 **MEDIUM PRIORITY** **RESOLVED**

Functional image has inappropriate text alternative

Success Criteria	1.1.1: Non-text content [2.0 A]
Devices	None
Pages Affected	None
Users Affected	Vision
Component	Image
ADR Type	Web Audit

Recheck comment 26/06/25

This issue is fixed.

On page 1 on desktop and iOS, the “Scottish Social Services Council logo” functional image used in the footer region leads to the homepage. The text alternative of the functional image is “Scottish Social Services Council home page”, which appropriately indicates this functionality.

The screenshot shows the footer of the website. On the left is the Scottish Social Services Council logo. To its right are three columns: 'Navigation' with links like Home, About us, Registration, etc.; 'Useful' with links like Working at the SSSC, Access to information, etc.; and 'Contact Us' with the phone number 0345 60 30 891 and social media icons. Below the screenshot is a code snippet showing the HTML for the logo link. The `alt` attribute is highlighted in red and contains the text "Scottish Social Services Council home page".

```

<div class="footer__inner" > grid
  <div class="footer_gridArea footer_gridArea--one" > grid
    <a class="logo_footer_logo" href="/" > flex
      
    </a>
  </div>
  <div class="footer_gridArea footer_gridArea--two" > ... </div> grid
</div>
</div>
</section>

```

What happened

Non-text content must have appropriate alternatives. This is required for the content to be understood by all users. Images, icons and so on may otherwise be miscommunicated by assistive technology. There are functional images with inappropriate text alternatives.

We have classed the images as functional because they have functionality associated with them.

Where it happened

This can be found on page 1 on desktop and iOS, with the “Scottish Social Services Council logo” functional image used in the footer region. Activating the link leads to the homepage. However, the accessible name of the link does not indicate this functionality.



```

<div class="footer_gridArea footer_gridArea--one"> (grid)
  <a class="logo footer_logo" href="/"> (flex)
    
    </a>
  </div>
    
```

Steps to reproduce

1. Inspect the functional images.
2. Confirm that the images have inappropriate text alternatives.

Recommendation

Functional images need to communicate the action or outcome of the functionality, such as the destination of image links.

Images marked up with an `` tag must have an `alt` attribute. Use the attribute to provide an appropriate text alternative. For example:

```
<a href="/"></a>
```

Inline SVG images can use `<title>`, `<desc>`, and `aria-labelledby` for the same purpose. For example:

```

<svg aria-labelledby="uniqueTitleID uniqueDescID">
  <title id="uniqueTitleID">The title of the SVG</title>
  <desc id="uniqueDescID">A longer, more complete description for complex
graphics.</desc>
  ...
</svg>
    
```

For information on `<svg>` elements and icon fonts, visit [CSS Tricks - Accessible SVGs](#) and [24 Accessibility - SVG, Icon Fonts, and Accessibility: A Case Study](#).

For more information, see [Web Accessibility Tutorials - Functional Images](#), and [Alt Attribute Decision Tree](#).

[Return to priority table](#)

ADR_22749-48 **MEDIUM PRIORITY** **RESOLVED**

Informative image has no text alternative

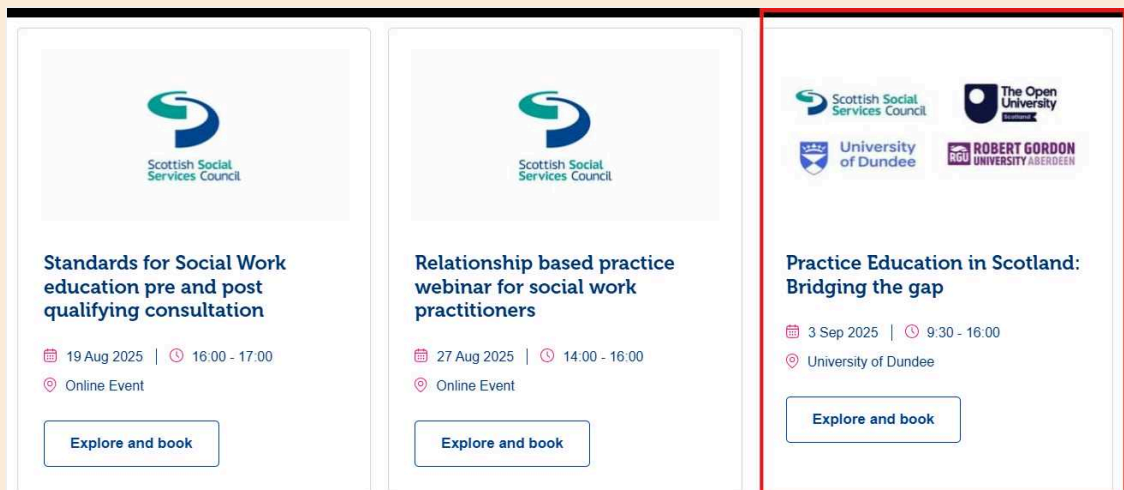
- Success Criteria** 1.1.1: Non-text content [2.0 A]
- Devices** Desktop, Mobile (iOS)
- Pages Affected** 06
- Users Affected** Vision
- Component** Image
- ADR Type** Web Audit

Recheck comment 14/07/25

This issue is fixed.

Fixed on page 6 on desktop and iOS, with the logos within each card. The cards themselves are each marked as a link to a specific event. The text content of the card link conveys the link destination; the logos themselves visually communicate the organisations involved in the event.

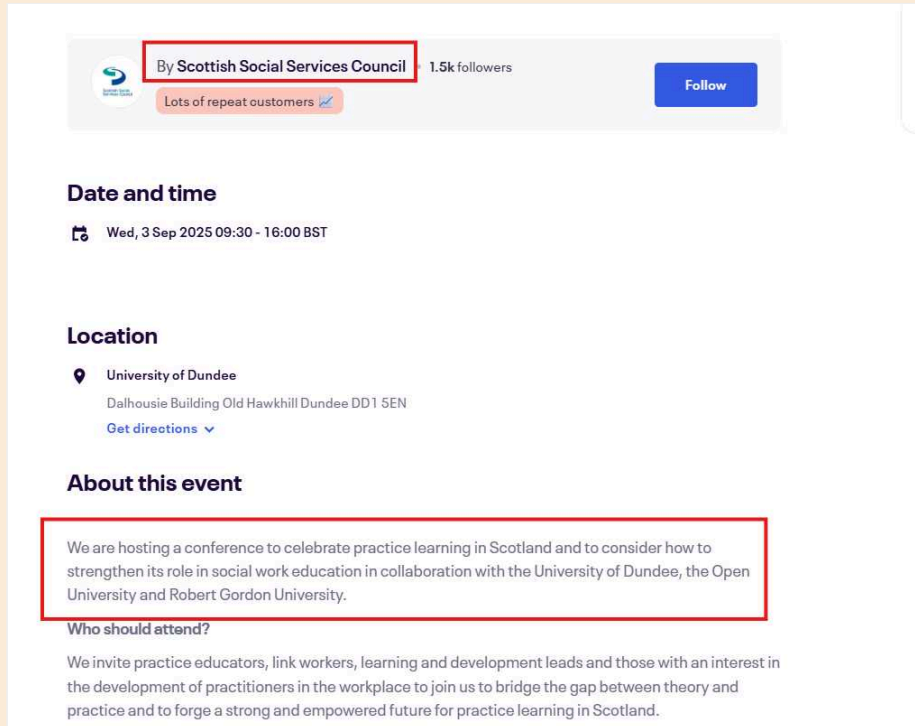
The information communicated by the logos is replicated on the event information page, linked by each card. For example, for the “Practice Education in Scotland: Bridging the gap” event:



```

▼ <a class="featuredCTA_item ctaItem" href="https://www.eventbrite.co.uk/e/practice-education-in-scotland-bridging-the-gap-registration-1426883259239"> flex
  ▼ <picture>
    
  </picture>
  <h2 class="ctaItem_heading heading heading--h3">Practice Education in Scotland: Bridging the gap</h2>
  ▶ <div class="ctaItem_meta"> flex
  ▶ <span class="mButton"> flex
  </a>
  
```

The link name communicates the destination “Practice Education in Scotland: Bridging the gap”. The organisations involved are then listed on this linked page:



As such, the logos can be considered to be functional here as they are part of a link. As the link destination is conveyed by text, these logos can be appropriately assigned blank alt text, for example, `alt=""`. In most cases reviewed, these logos are assigned a blank alt text value via an `alt` attribute. However, there are some instances of the SSC logo being used with alt text of "SSC logo". These instances can be instead provided with a blank `alt` attribute.

Recheck comment 27/06/25

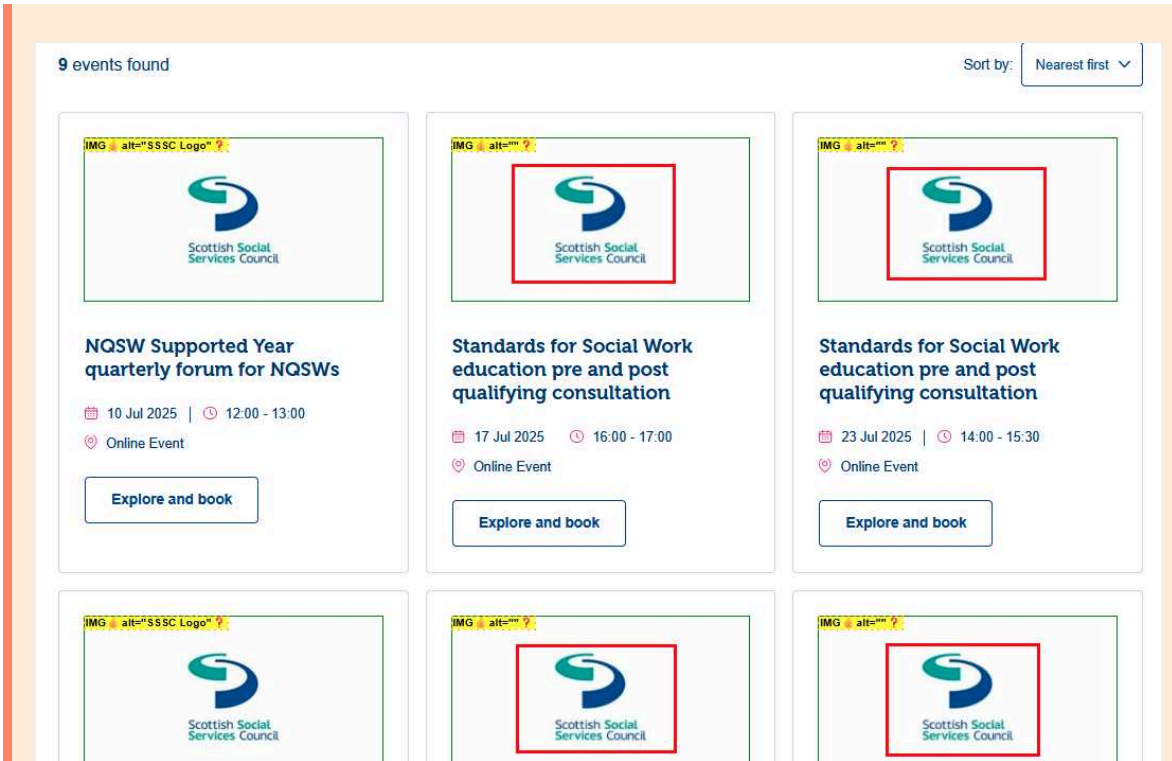
This issue is partially fixed.

Partially fixed: on page 6 on desktop and iOS, with the following informative images within each card:

- For the card image, such as the "Scottish Social Services Council" logo, only the logos above the "NQSW Supported Year quarterly forum for NQSWs" heading have alternative text. Logos in other cards do not have alternative text.
- The "Date", "Time", and "Location" icons under the heading of each card have appropriate text alternatives.
- The external domain icons within the "Explore and book" buttons in each card have been removed from the page. This was confirmed by Gabby Scott from MTC on 27/06/2025. She commented, "These will remain removed".

We trust in good faith that MTC has removed these inaccessible components and that they will not be reintroduced to the website.

If the elements are brought back, we trust that MTC understands the associated risk and will reintroduce them in a more accessible form.



```

<div class="postsBlock_filter"></div>
<div class="postsBlock_postGrid featuredCTA">
  <a class="featuredCTA_item ctaItem" href="https://www.eventbrite.co.uk/e/nqsw-supported-year-quarterly-forum-for-nqsw-tickets-1254447479179">
    <img alt="SSSC Logo" />
    <h2 class="ctaItem_heading heading--h3">NQSW Supported Year quarterly forum for NQSWs</h2>
    <div class="ctaItem_meta">
      <div class="ctaItem_meta_item ctaItem_meta_date">
        <span class="ctaItem_meta_itemLabel">
          <i class="mSvgIcon mSvgIcon--mask mSvgIcon--calendar" aria-hidden="true"></i>
          <span class="visuallyHidden">Date:</span>
        </span>
        10 Jul 2025
      </div>
      <div class="ctaItem_meta_item ctaItem_meta_time">
        <span class="ctaItem_meta_itemLabel">
          <i class="mSvgIcon mSvgIcon--mask mSvgIcon--clock" aria-hidden="true"></i>
          <span class="visuallyHidden">Time:</span>
        </span>
        12:00 - 13:00
      </div>
      <div class="ctaItem_meta_item ctaItem_meta_location">
        <span class="ctaItem_meta_itemLabel">
          <i class="mSvgIcon mSvgIcon--mask mSvgIcon--location-pin" aria-hidden="true"></i>
          <span class="visuallyHidden">Location:</span>
        </span>
        Online Event
      </div>
    </div>
    <span class="mButton"></span>
  </a>

```

Out of scope: on page 7 on desktop, the informative icon within the “View on map” and “Go to Contact Scotland” links have been removed from the page. This was confirmed by Gabby Scott from MTC on 27/06/2025. She commented, “These will remain removed”.

We trust in good faith that MTC has removed these inaccessible components and that they will not be reintroduced to the website.

If the elements are brought back, we trust that MTC understands the associated risk and will reintroduce them in a more accessible form.

What happened

Non-text content must have appropriate alternatives. This is required for the content to be understood by all users. Images, icons and so on may otherwise be miscommunicated by assistive technology. There are informative images without appropriate text alternatives.


We have classed the images as informative as they communicate unique or meaningful information.

Where it happened




This can be found on page 6 on desktop and iOS, with the below informative images within each card: * The card image such as the "Scottish Social Services Council" logo. * The "Date", "Time" and "Location" icons under the heading of each card. * The external domain icon within the "Explore and book" buttons in each card (opens in the same tab).


The images visually indicate information that is not otherwise available on the page. However, they do not have a text alternative.


2 events found






NQSW Supported Year quarterly forum for supervisors


 24 Jul 2025 |
  12:00 - 13:00
 Online Event

[Explore and book](#) 



NQSW Supported Year quarterly forum for NQSWs

 10 Jul 2025 |
  12:00 - 13:00
 Online Event

[Explore and book](#) 

```

<a class="featuredCTA_item ctaitem" href="https://www.eventbrite.co.uk/e/nqsw-supported-year-quarterly-forum-for-supervisors-tickets-1254453226369"> (#flex)
  <picture>
    
  </picture>
  <h2 class="ctaitem_heading heading heading-h3" data-element-id="headingsmap-9-0">
    NQSW Supported Year quarterly forum for supervisors
  </h2>
  <div class="ctaitem_meta"> (#flex)
    <div class="ctaitem_meta_item ctaitem_meta_date"> (#flex)
      <i class="msvgicon msvgicon--mask msvgicon--calendar" aria-hidden="true"></i>
      24 Jul 2025
      ::after
    </div>
    <div class="ctaitem_meta_item ctaitem_meta_time"> (#flex)
      <i class="msvgicon msvgicon--mask msvgicon--clock" aria-hidden="true"></i>
      12:00 - 13:00
    </div>
    <div class="ctaitem_meta_item ctaitem_meta_location"> (#flex)
      <i class="msvgicon msvgicon--mask msvgicon--location-pin" aria-hidden="true"></i>
      Online Event
    </div>
  </div>
  <span class="mbutton"> (#flex)
    <span class="mbutton_text">Explore and book</span>
    ::after
  </span>
</a>

```

This can be found on page 7 on desktop, with the informative icon within the “View on map” and “Go to Contact Scotland” links in the “Get in touch” section. The icon visually indicates that the links open a new domain (opens in the same tab). However, they do not have a text alternative.

Get in touch

Phone lines open Monday to Friday 9am – 4pm.
Your call may be recorded for training and reference purposes.

Phone

[0345 60 30 891](tel:03456030891)

Email

enquiries@sssc.uk.com

Write to us

Scottish Social Services Council
Compass House
11 Riverside Drive
Dundee
DD1 4NY

[View on map](#)

Media enquiries

For all media enquiries please contact:
media@sssc.uk.com

Do you use British Sign Language (BSL)?

BSL users can contact us via contact SCOTLAND-BSL, the online British Sign Language interpreting service.
Go to contact Scotland

[Go to Contact Scotland](#)

```

<h3 class="heading heading--h3">Write to us</h3>
<div class="getInTouch_address">
  <a class="mButton" href="https://www.google.com/maps/place/Scottish+Social+Services+C_q62b6q53?entry=ttu8g_ep=EggyMDI1MDQzMC4xIjYxMD5oASAFQAw%3D%3D"
    rel="noreferrer noopener nofollow">
    <span class="mButton_text">View on map</span>
    <img alt="external link icon" data-bbox="231 314 248 328" style="vertical-align: middle;"/>
  </a>
</div>
</div>
<div class="getInTouch_content">
  <h3>Media enquiries</h3>
  <p></p>
  <h3>Do you use British Sign Language (BSL)?</h3>
  <p></p>
  <a class="mButton" href="https://contactscotland-bsl.org/"
    rel="noreferrer noopener nofollow">
    <span class="mButton_text">Go to Contact Scotland</span>
    <img alt="external link icon" data-bbox="634 254 651 268" style="vertical-align: middle;"/>
  </a>
</div>

```

Steps to reproduce

1. Inspect the informative images.
2. Confirm that they do not have appropriate text alternatives.

Recommendation

Informative images that convey meaningful or unique information need a descriptive text alternative.

Images marked up with an `` tag must have an `alt` attribute. Use the attribute to provide an appropriate text alternative. For example:

```

```

Inline SVG images can use `<title>`, `<desc>`, and `aria-labelledby` for the same purpose. For example:

```

<svg aria-labelledby="uniqueTitleID uniqueDescID">
  <title id="uniqueTitleID">The title of the SVG</title>
  <desc id="uniqueDescID">A longer, more complete description for complex
  graphics.</desc>
  ...
</svg>

```

For information on `<svg>` elements and icon fonts, visit [CSS Tricks - Accessible SVGs](#) and [24 Accessibility - SVG, Icon Fonts, and Accessibility: A Case Study](#).

For more information, see [Web Accessibility Tutorials - Informative Images](#), and [Alt Attribute Decision Tree](#).

[Return to priority table](#)

ADR_22749-4 MEDIUM PRIORITY **RESOLVED**

Heading not marked up semantically

- Success Criteria** 1.3.1: Info and Relationships [2.0 A]
- Devices** None
- Pages Affected** None
- Users Affected** Vision
- Component** Heading
- ADR Type** Web Audit

Recheck comment 17/07/25

The issue is fixed.

Fixed: on page 5, on desktop and iOS, the “In this section” heading is correctly marked up as a level 2 heading using <h2>

```
<nav aria-labelledby='heading--contentwithNavigation-6878425713437'>  
  <h2 id='heading--contentwithNavigation-6878425713437' class='heading heading--h3'>In this section</h2>  
  <ul data-sidebar-links=''>  
    <li>  
      <a href='#introduction'>Introduction</a>  
    </li>  
    <li>  
      <a href='#employers'>Employers</a>  
    </li>  
    <li>  
      <a href='#member-of-the-public'>Member of the public</a>  
    </li>  
    <li>  
      <a href='#higher-education-institutes-hei'>Higher education institutes (HEI)</a>  
    </li>  
    <li>  
      <a href='#third-parties'>Third parties</a>  
    </li>  
  </ul>  
</nav>
```

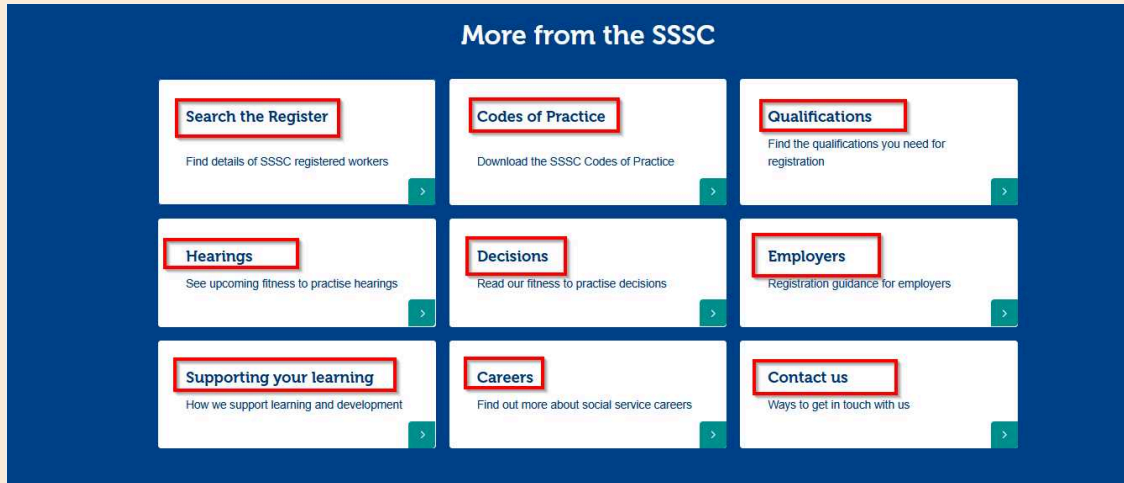
Recheck comment 26/06/25

This issue is partially fixed.

Fixed: on page 1 on desktop and iOS, the headings such as “Help with registering”, “Raising a concern”, “Supported Learning”, and “About the SSSC” are marked up semantically as headings.

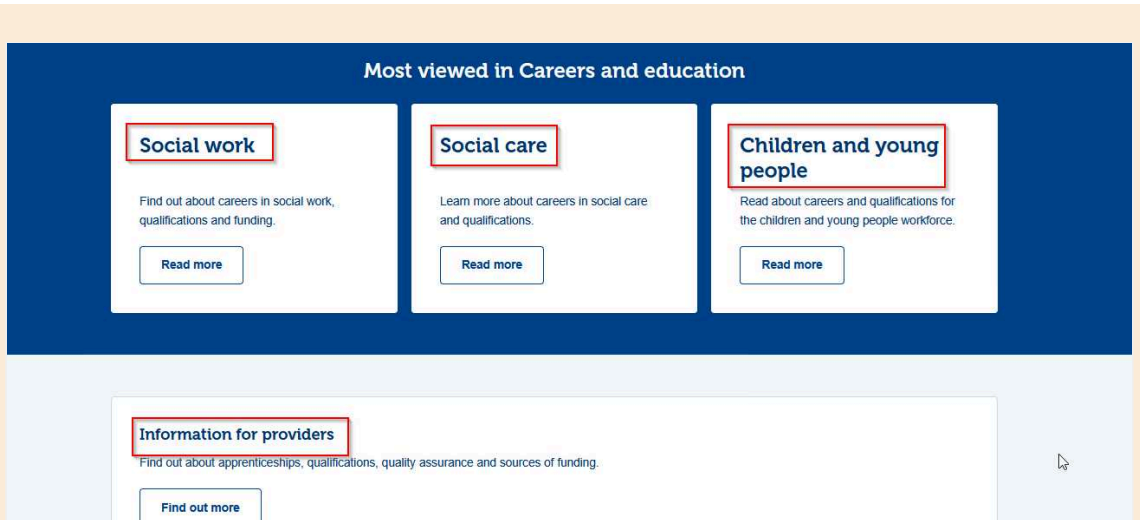
The screenshot displays the SSSC website's navigation menu and three main content cards. The cards are titled 'Help with registering', 'Raising a concern', and 'Supporting learning'. Each card includes a brief description and a button to access the relevant page. Below the cards is an 'About the SSSC' section. A developer tool overlay at the bottom shows the HTML structure, with red boxes highlighting the heading tags for each card and the 'About the SSSC' section.

Fixed: on page 1 on desktop and iOS, the card headings such as “Search the Register”, “Codes of Practice”, “Qualifications”, and so on are marked up semantically as headings.



```
<ul class="featuredCTA_list"> grid
  <li class="featuredCTA_listItem">
    <a class="featuredCTA_item ctaItem" href="https://www.sssc.uk.com/search-the-register/"> flex
      <h3 class="ctaItem_heading heading heading--h3">Search the Register</h3>
      <p>Find details of SSSC registered workers</p>
      <i class="mSvgIcon mSvgIcon--chevron-right-regular" aria-hidden="true"> </i> flex
    </a>
  </li>
  <li class="featuredCTA_listItem">
    <a class="featuredCTA_item ctaItem" href="https://www.sssc.uk.com/standards/codes-of-practice/"> flex
      <h3 class="ctaItem_heading heading heading--h3">Codes of Practice</h3>
      <p>Download the SSSC Codes of Practice</p>
      <i class="mSvgIcon mSvgIcon--chevron-right-regular" aria-hidden="true"> </i> flex
    </a>
  </li>
  <li class="featuredCTA_listItem">
    <a class="featuredCTA_item ctaItem" href="/registration/help-with-register-parts-fees-and-qualifications/"> flex
      <h3 class="ctaItem_heading heading heading--h3">Qualifications</h3>
      <p>Find the qualifications you need for registration</p>
      <i class="mSvgIcon mSvgIcon--chevron-right-regular" aria-hidden="true"> </i> flex
    </a>
  </li>
  <li class="featuredCTA_listItem"> </li>
  <li class="featuredCTA_listItem"> </li>
  <li class="featuredCTA_listItem"> </li>
  <li class="featuredCTA_listItem"> </li>
  <li class="featuredCTA_listItem"> </li>
  <li class="featuredCTA_listItem"> </li>
</ul>
```

Fixed: on page 2 on desktop and iOS, the visual headings under the “Most viewed in Careers and education” heading such as “Social work”, “Social care”, “Children and young people” and “Information for providers” are marked up semantically as headings.



```

<li class="featuredCTA_listItem">
  <a class="featuredCTA_item ctaItem" href="https://www.sssc.uk.com/careers-and-education/social-work/"> flex
  <h3 class="ctaItem_heading heading heading--h2">Social work</h3>
  <p></p>
  <span class="mButton"></span> flex
</a>
</li>
<li class="featuredCTA_listItem">
  <a class="featuredCTA_item ctaItem" href="https://www.sssc.uk.com/careers-and-education/social-care/"> flex
  <h3 class="ctaItem_heading heading heading--h2">Social care</h3>
  <p></p>
  <span class="mButton"></span> flex
</a>
</li>
<li class="featuredCTA_listItem">
  <a class="featuredCTA_item ctaItem" href="https://www.sssc.uk.com/careers-and-education/children-and-young-people/"> flex
  <h3 class="ctaItem_heading heading heading--h2">Children and young people</h3>
  <p></p>
  <span class="mButton"></span> flex
</a>
</li>
</ul>
</div>
</section>
<section id="information-for-providers" class="featuredCTA featuredCTA--none" style="--featuredCta--background-color: var(--muted-one--color); --feat-quick-links-label="Information for providers" tabindex="1">
  <div class="featuredCTA_wrapper">
    <div class="featuredCTA_list"> grid
      <div class="featuredCTA_listItem">
        <a class="featuredCTA_item ctaItem" href="/careers-and-education/information-for-providers/"> flex
        <h2 class="ctaItem_heading heading heading--h3">Information for providers</h2>
        <p></p>
        <span class="mButton"></span> flex
      </a>
    </div>
  </div>
</section>

```

Not Fixed: on page 5, on desktop and iOS, the “In this section” heading visual heading that titles the in-page links below it is not marked up as a heading.

In this section

- Introduction
- Employers
- Member of the public
- Higher education institutes (HEI)
- Third parties
- Guidance for survivors of child abuse while in care in Scotland

Introduction

Everyone should act if they have serious concerns about someone working in social care.

There are different ways of raising a concern with us depending on who you are.

We consider concerns about people working in social services already on our Register and people applying to register with us.

If you have a concern or wish to complain about a registered care service, you should contact the [Care Inspectorate](#).

Find out more by choosing the most suitable option for you from the list below.

Employers

```

<section class="contentWithNavigation" data-content-with-navigation=""> grid
  <aside class="contentWithNavigation_sidebar">
    <nav aria-labelledby="heading--contentWithNavigation-685d19f93de4e">
      <span id="heading--contentWithNavigation-685d19f93de4e" class="heading heading--h3">In this section</span>
    </nav>
  </aside>
  <article class="contentWithNavigation__content" data-content-with-navigation-content=""> </article>
</section>
</main>

```

Recheck comment 09/06/25

This issue was updated during additional testing on 09/06/2025.

The same issue has been observed on page 5. On this page, there is a heading that is not marked up as such.

What happened

If a relationship between elements is expressed with visual cues, it also needs to be programmatically determined. This enables assistive technology to communicate the relationships to users. Content may otherwise seem unstructured or disorganised.

There are elements that title or categorise other content. These elements have not been marked up as headings.

Where it happened

This can be seen on page 1 on desktop and iOS, with the headings such as “Help with registering”, “Raising a concern”, “Supported Learning” and “About the SSSC”. These are visual headings which title the content below them. However, they are not marked up semantically as headings.

Help with registering

Help and guidance on applying and being registered with us.

Get help registering

Raising a concern

Find out how we deal with concerns about registered workers.

Raise a concern

Supporting learning

Resources for your continuous professional learning (CPL).

Find learning

About the SSSC

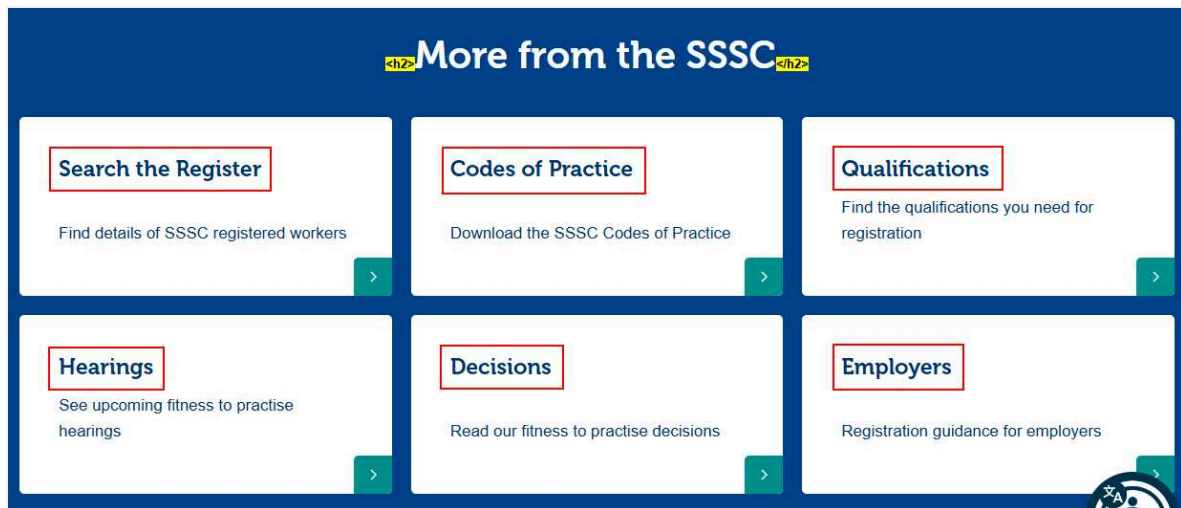
We are the regulator for the social work, social care and children and young people workforce in Scotland. Our vision, as the regulator of the social service workforce in Scotland, is that the people of Scotland can count on social work, social care and children and young people services being provided by a trusted, skilled, confident and valued workforce.

Find out more

```

<section class="featuredCTA featuredCTA--none" style="--featuredCta--background-color: var(--white--bg-color); --featuredCta--column-count: 3;"
  <div class="featuredCTA_wrapper" (grid)
    <a class="featuredCTA_item ctaItem" href="/registration" (flex)
      <strong class="ctaItem_heading heading heading--h2">Help with registering</strong>
    <p>
      Help and guidance on applying and being registered with us.
    </p>
    <span class="mButton"> (flex)
  </a>
  <a class="featuredCTA_item ctaItem" href="https://www.ssscukcom.mtcsserver.com/dealing-with-concerns/raising-a-concern-with-us/" rel="noreferrer noopener nofollow"> (flex)
  <a class="featuredCTA_item ctaItem" href="https://www.ssscukcom.mtcsserver.com/supporting-your-learning/" rel="noreferrer noopener nofollow"> (flex)
</div>
</section>
<section class="featuredCTA featuredCTA--none" style="--featuredCta--background-color: var(--muted-one--color); --featuredCta--column-count: 1;"
  <div class="featuredCTA_wrapper" (grid)
    <a class="featuredCTA_item ctaItem" href="/about-us/" (flex)
      <strong class="ctaItem_heading heading heading--h3">About the SSSC</strong>
    <p>
      We are the regulator for the social work, social care and children and young people workforce in Scotland. Our vision, as the regulator of the social service workforce in Scotland, is that the people of Scotland can count on social work, social care and children and young people services being provided by a trusted, skilled, confident and valued workforce.
    </p>
    <span class="mButton"> (flex)
  </a>
</div>
</section>
  
```

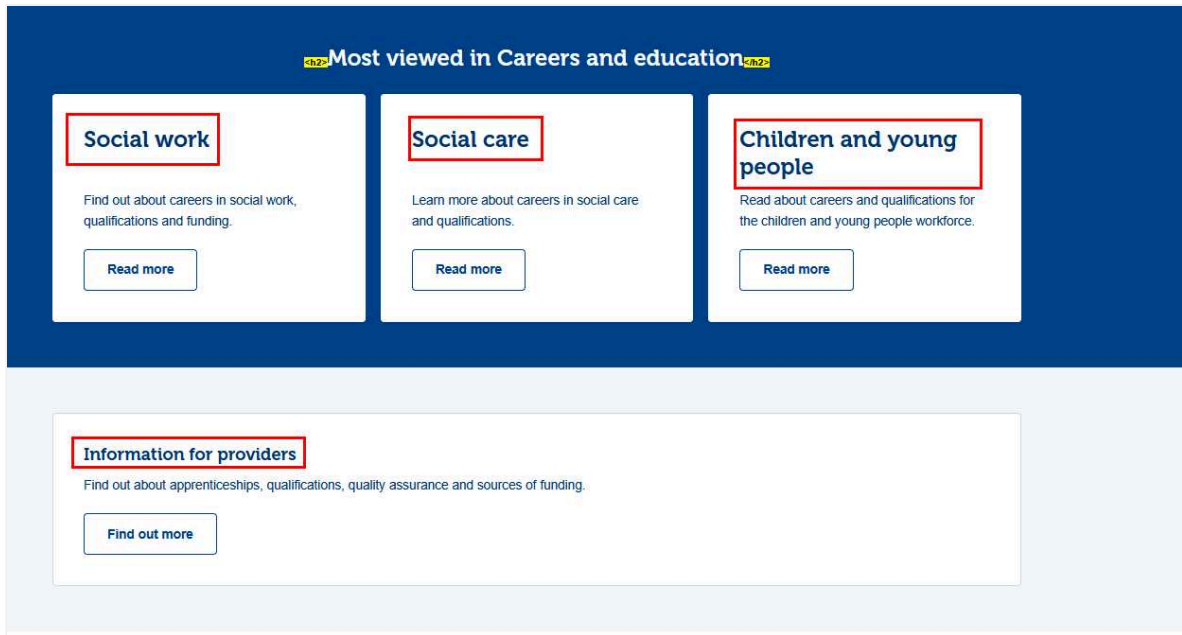
This can be seen on page 1 on desktop and iOS, with the card headings such as “Search the Register”, “Codes of Practice”, “Qualifications” and so on. These are visual headings which title the content below them. However, they are not marked up semantically as headings.



```

<section class="featuredCTA featuredCTA--simple" style="--featuredCta--background-color: var(--secondary-one--color); --featuredCta--column-count: 3;">
  <h2 class="featuredCTA_heading heading heading--h1">More from the SSSC</h2>
  <div class="featuredCTA_wrapper">
    <a class="featuredCTA_item ctaItem" href="https://www.ssscukcom.mtcservers.com/search-the-register/" rel="nofollow noopener nofollow">
      <strong class="ctaItem_heading heading heading--h3">Search the Register</strong>
      <p>Find details of SSSC registered workers</p>
      <i class="mSvgIcon mSvgIcon--chevron-right-regular" aria-hidden="true"></i>
    </a>
    <a class="featuredCTA_item ctaItem" href="https://www.ssscukcom.mtcservers.com/standards/codes-of-practice/" rel="nofollow noopener nofollow">
      <strong class="ctaItem_heading heading heading--h3">Codes of Practices</strong>
      <p>Download the SSSC Codes of Practice</p>
      <i class="mSvgIcon mSvgIcon--chevron-right-regular" aria-hidden="true"></i>
    </a>
    <a class="featuredCTA_item ctaItem" href="/registration/help-with-register-parts-fees-and-qualifications/">
      <strong class="ctaItem_heading heading heading--h3">Qualifications</strong>
      <p>Find the qualifications you need for registration</p>
      <i class="mSvgIcon mSvgIcon--chevron-right-regular" aria-hidden="true"></i>
    </a>
    <a class="featuredCTA_item ctaItem" href="/dealing-with-concerns/upcoming-hearings/"></a>
    <a class="featuredCTA_item ctaItem" href="/dealing-with-concerns/impairment-decisions/"></a>
    <a class="featuredCTA_item ctaItem" href="https://www.ssscukcom.mtcservers.com/registration/employers/" rel="nofollow noopener nofollow"></a>
    <a class="featuredCTA_item ctaItem" href="https://www.ssscukcom.mtcservers.com/supporting-your-learning/" rel="nofollow noopener nofollow"></a>
    <a class="featuredCTA_item ctaItem" href="https://www.ssscukcom.mtcservers.com/careers-and-education/" rel="nofollow noopener nofollow"></a>
    <a class="featuredCTA_item ctaItem" href="https://www.ssscukcom.mtcservers.com/contact-us/" rel="nofollow noopener nofollow"></a>
  </div>
</section>
    
```

This can also be seen on page 2 on desktop and iOS, with the visual headings under the “Most viewed in Careers and education” heading such as “Social work”, “Social care”, “Children and young people” and “Information for providers”. These are visual headings which title the content below them. However, they are not marked up semantically as headings.



```

<section id="most-viewed-in-careers-and-education" class="featuredCTA featuredCTA--none" style="--featuredcta--background-color: var(--secondary-one-color); --featuredcta--column-count: 3;" data-quick-links-label="Most viewed in Careers and education">
  <h2 class="featuredCTA_heading heading heading--h2">Most viewed in Careers and education</h2>
  <div class="featuredCTA_wrapper"> (g18)
    <a class="featuredCTA_item ctatitem" href="https://www.ssscuk.com/mtcserver.com/careers-and-education/social-work/" rel="noreferrer noopener nofollow"> (Fix)
      <strong class="ctatitem_heading heading heading--h2">Social work</strong>
    </a>
    <p></p>
    <span class="mbutton"> (Fix)
  </div>
  <a class="featuredCTA_item ctatitem" href="https://www.ssscuk.com/mtcserver.com/careers-and-education/social-care/" rel="noreferrer noopener nofollow"> (Fix)
    <strong class="ctatitem_heading heading heading--h2">Social care</strong>
  </a>
  <p></p>
  <span class="mbutton"> (Fix)
</div>
</div>
<section id="information-for-providers" class="featuredCTA featuredCTA--none" style="--featuredcta--background-color: var(--muted-one-color); --featuredcta--column-count: 1;" data-quick-links-label="Information for providers">
  <div class="featuredCTA_wrapper"> (g18)
    <a class="featuredCTA_item ctatitem" href="/careers-and-education/information-for-providers/"> (Fix)
      <strong class="ctatitem_heading heading heading--h2">Information for providers</strong>
    </a>
    <p></p>
    <span class="mbutton"> (Fix)
  </div>
</section>

```

This can also be seen on page 5, on desktop and iOS, with the "In this section" heading. This is a visual heading that titles the in-page links below it. However, it is not marked up as a heading.

In this section

- Introduction
- Employers
- Member of the public
- Higher education institutes (HEI)
- Third parties
- Guidance for survivors of child abuse while in care in Scotland

Introduction

Everyone should act if they have serious concerns about someone working in social care.

There are different ways of raising a concern with us depending on who you are.

We consider concerns about people working in social services already on our Register and people applying to register with us.

If you have a concern or wish to complain about a registered care service, you should contact the [Care Inspectorate](#).

Find out more by choosing the most suitable option for you from the list below.

Employers

Raise a concern as a representative of an employer of social service workers

+ What we will consider

```

<aside class="contentwithNavigation_sidebar">
  <nav aria-labelledby="heading--contentwithNavigation-6846b5f039d0e">
    <span id="heading--contentwithNavigation-6846b5f039d0e" class="heading heading--h3">In this section</span>
    <ul data-sidebar-links="">
  </nav>
</aside>
    
```

Steps to reproduce

1. Visually note the headings.
2. Inspect the elements in the DOM.
3. Note that the headings have not been marked up, such as `<h1>`, `<h2>`, and so on.

Recommendation

When elements title or categorise other content, use headings with appropriate heading levels. Make sure that the heading levels used correctly communicate the structure of the page, such as the parent and child relationship between the headings. For example:

```

<h1>Cooking techniques</h1>
<!-- some content here -->
  <h2>Cooking with oil</h2>
  <!-- some content here -->
    <h3>Saut ing</h3>
    <!-- some content here -->
    <h3>Deep frying</h3>
    <!-- some content here -->
  <h2>Cooking with butter</h2>
  <!-- some content here -->
    
```

For more information, see [WAI Tutorials - Headings](#).

[Return to priority table](#)

ADR_22749-35 **MEDIUM PRIORITY** **RESOLVED**

Text uses heading semantics inappropriately

Success Criteria 1.3.1: Info and Relationships [2.0 A]

Devices None

Pages Affected None

Users Affected Vision

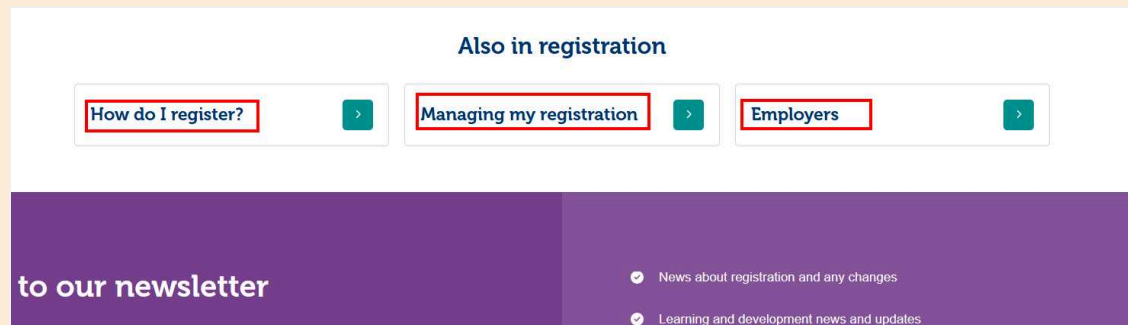
Component Heading

ADR Type Web Audit

Recheck comment 27/06/25

This issue is fixed.

Fixed: on page 3 on desktop and iOS, the three links under the “Also in registration” heading, such as “How do I register?” are not marked up as headings.

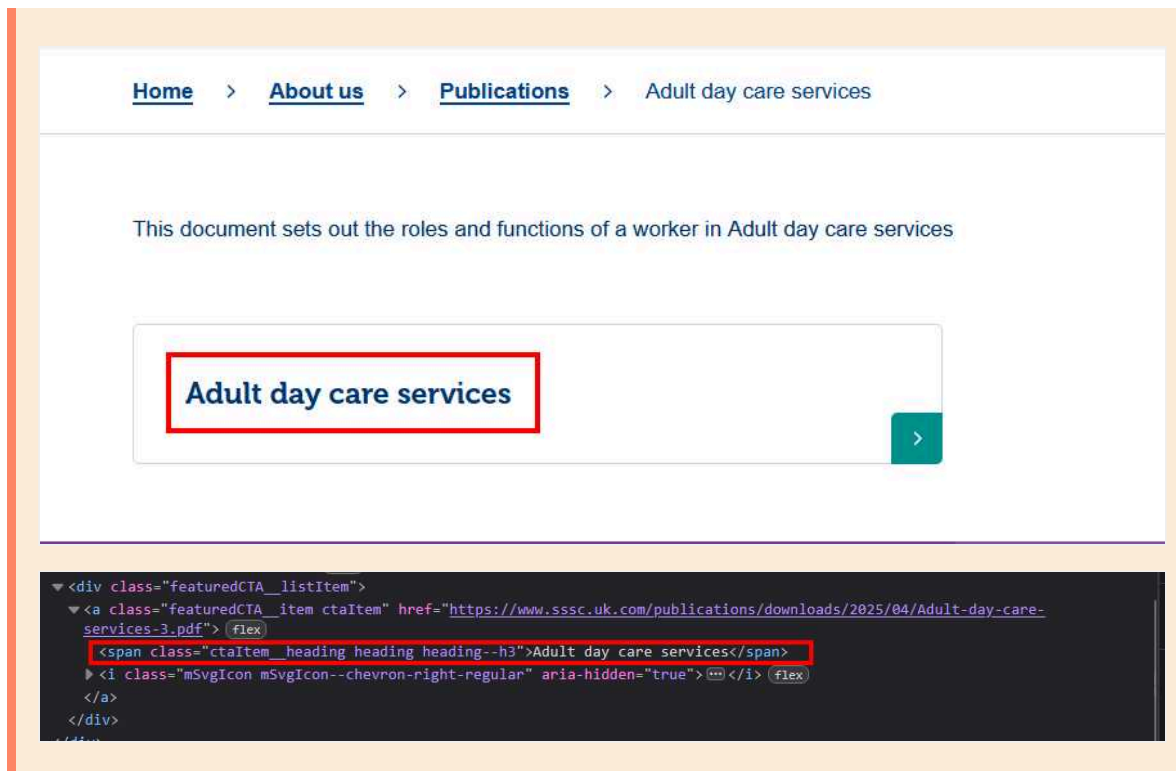


```

<div class="featuredCTA_wrapper">
  <ul class="featuredCTA_list">
    <li class="featuredCTA_listItem">
      <a class="featuredCTA_item ctaItem" href="/registration/how-do-i-register/">
        <span class="ctaItem_heading heading heading--h3">How do I register?</span>
        <i class="mSvgIcon mSvgIcon--chevron-right-regular" aria-hidden="true"></i>
      </a>
    </li>
    <li class="featuredCTA_listItem">
      <a class="featuredCTA_item ctaItem" href="/registration/managing-my-registration/">
        <span class="ctaItem_heading heading heading--h3">Managing my registration</span>
        <i class="mSvgIcon mSvgIcon--chevron-right-regular" aria-hidden="true"></i>
      </a>
    </li>
    <li class="featuredCTA_listItem">
      <a class="featuredCTA_item ctaItem" href="/registration/employers/">
        <span class="ctaItem_heading heading heading--h3">Employers</span>
        <i class="mSvgIcon mSvgIcon--chevron-right-regular" aria-hidden="true"></i>
      </a>
    </li>
  </ul>
</div>
</section>

```

Fixed: on page 4 on desktop and iOS, the “Adult day care services” link is not marked up as a heading.



What happened

If a relationship between elements is expressed with visual cues, it also needs to be programmatically determined. This enables assistive technology to communicate the relationships to users. Similarly, content must not use structural semantics inappropriately. Content may otherwise seem unstructured or disorganised.

There are elements contained in heading tags that do not title or categorise other content.

Where it happened

This can be seen on page 3, on desktop and iOS, with the three links under the “Also in registration” heading, such as “How do I register?”. The links are incorrectly marked up as level 3 headings.

Showing results 1 to 10 of 100 rows

< Previous 1 2 3 4 5 ... 10 Next >

Also in registration

How do I register? >

Managing my registration >

Employers >

```

<li class="featuredCTA_listItem">
  <a class="featuredCTA_item ctaItem" href="/registration/how-do-i-register/"> flex
    <h3 class="ctaItem_heading heading heading--h3">How do I register?</h3>
    <i class="msvgIcon msvgIcon--chevron-right-regular" aria-hidden="true"> ... </i> flex
  </a>
</li>
  
```

This can also be seen on page 4, on desktop and iOS, with the “Adult day care services” link. The link is incorrectly marked up as a level 2 heading.

Adult day care services

[Home](#) > [About us](#) > [Publications](#) > Adult day care services

This document sets out the roles and functions of a worker in Adult day care services

Adult day care services >

```

<a class="featuredCTA_item ctaItem" href="https://www.sssc.uk.com/publications/downloads/2025/04/Adult-day-care-services-3.pdf" rel="noreferrer noopener nofollow">
  flex
  <h2 class="ctaItem_heading heading heading--h2">Adult day care services</h2>
  <i class="msvgIcon msvgIcon--chevron-right-regular" aria-hidden="true"> ... </i> flex
</a>
  
```

Steps to reproduce

1. Observe that the content is not titling or categorising other content.
2. Inspect the content and review the DOM.
3. Note the inappropriate use of heading tags.

Recommendation

Use CSS to style the content. Do not use heading tags, such as <h1>, <h2>, and so on, unless the text elements title or categorise other content. Use neutral tags that do not have any structural meaning, such as <div>, , and <p> tags. For example:

`<h1>Contact us</h1>`

`<p>Use the form below to get in contact</p>`

For more information, see [MDN - Styling Text](#), [WAI Tutorials - Headings](#) and [WAI Tutorials - Content Structure](#).

[Return to priority table](#)

ADR_22749-46 **MEDIUM PRIORITY** **RESOLVED**

Accordion heading not marked up semantically

Success Criteria 1.3.1: Info and Relationships [2.0 A]

Devices None

Pages Affected None

Users Affected Vision

Component Accordion or Disclosure, Heading

ADR Type Web Audit

Recheck comment 27/06/25

This issue is fixed.

Fixed: on page 5 on desktop and iOS, the accordions, such as “What we will consider” are semantically marked up as headings.

In this section

[Introduction](#)

Employers

Member of the public

Higher education institutes (HEI)

Third parties

Guidance for survivors of child abuse while in care in Scotland

Raise a concern as a representative of an employer of social service workers

- What we will consider

The matters that we consider are:

- misconduct
- deficient professional practice
- health
- a decision by another regulator
- a conviction.

+ What you need to tell us

+ Referring a worker not registered with us

+ Matters you should not refer to us

```

<details name="what-we-will-consider " open=""> (container)
  <summary> (flex)
    <i class="mSvgIcon mSvgIcon--square-plus" aria-hidden="true"></i>
    <i class="mSvgIcon mSvgIcon--square-minus" aria-hidden="true"></i> (flex)
    <h3 class="heading heading--h3">what we will consider</h3>
  </summary>
  <div class="details__inner"></div>
</details>
<details name="tell-us "> (container)
  <summary> (flex)
    <i class="mSvgIcon mSvgIcon--square-plus" aria-hidden="true"></i> (flex)
    <i class="mSvgIcon mSvgIcon--square-minus" aria-hidden="true"></i>
    <h3 class="heading heading--h3">what you need to tell us</h3>
  </summary>
  <div class="details__inner"></div>
</details>
<details name="Referring-a-worker "> (container)
  <summary> (flex)
    <i class="mSvgIcon mSvgIcon--square-plus" aria-hidden="true"></i> (flex)
    <i class="mSvgIcon mSvgIcon--square-minus" aria-hidden="true"></i>
    <h3 class="heading heading--h3">Referring a worker not registered with us</h3>
  </summary>
  <div class="details__inner"></div>
</details>
<details name="not-refer "></details> (container)
<details name="Employer-advice-line "></details> (container)
<details name="Employer-referral-form "></details> (container)
<h2 id="member-of-the-public" class="wp-block-heading" tabindex="-1"></h2>
<p></p>
    
```

Fixed: on page 8 on desktop and iOS, the main accordions, such as “1. What best describes you?”, and the nested accordions within the “4. Details for this level of job role?” accordion, such as “Qualification” are semantically marked up as headings.

The screenshot shows a web form with four numbered steps, each with a plus icon to its left:

1. What best describes you?
2. Select the type of service you work in?
3. Select the level of your job role?
4. Details for this level of job role?

Below the steps are four expandable sections, each with a plus icon to its left:

- Qualification
- Registration
- Current fees
 - These fees are applicable prior to 1st April 2026
 - Annual Declaration Fee**
 - £15.00
 - Application Fee**
 - £15.00
- Future fees

At the bottom of the form are two buttons: "Back" and "Start again".

```

<details id="details--registerPart" name="help-with-register"> (container)
  <summary> (flex)
    <i class="mSvgIcon mSvgIcon--mask mSvgIcon--square-plus" aria-hidden="true"></i>
    <i class="mSvgIcon mSvgIcon--mask mSvgIcon--square-minus" aria-hidden="true"></i>
    <h3 id="details_label--registerPart" class="heading heading--h3">1. What best describes you?</h3>
  </summary>
  <div class="details_inner"> (flex) </div>
</details>
<details id="details--service" name="help-with-register" aria-disabled="false"> (container)
  <summary> (flex)
    <i class="mSvgIcon mSvgIcon--mask mSvgIcon--square-plus" aria-hidden="true"></i>
    <i class="mSvgIcon mSvgIcon--mask mSvgIcon--square-minus" aria-hidden="true"></i>
    <h3 id="details_label--service" class="heading heading--h3">2. Select the type of service you work in?</h3>
  </summary>
  <div class="details_inner"> (flex) </div>
</details>
<details id="details--role" name="help-with-register" aria-disabled="false"> (container)
  <summary> (flex)
    <i class="mSvgIcon mSvgIcon--mask mSvgIcon--square-plus" aria-hidden="true"></i>
    <i class="mSvgIcon mSvgIcon--mask mSvgIcon--square-minus" aria-hidden="true"></i>
    <h3 id="details_label--role" class="heading heading--h3">4. Details for this level of job role?</h3>
  </summary>
  <div class="details_inner">
    <details name=""> (container)
      <summary> (flex)
        <i class="mSvgIcon mSvgIcon--mask mSvgIcon--square-plus" aria-hidden="true"></i>
        <i class="mSvgIcon mSvgIcon--mask mSvgIcon--square-minus" aria-hidden="true"></i>
        <h4 id="details_label--role" class="heading heading--h3">Qualification</h4>
      </summary>
      <div class="details_inner"> (flex) </div>
    </details>
  </div>
</details>
</div>
</details>

```

What happened

If a relationship between elements is expressed with visual cues, it also needs to be programmatically determined. This enables assistive technology to communicate the relationships to users. Content may otherwise seem unstructured or disorganised.

There are elements that title or categorise other content. These elements have not been marked up as headings.

Where it happened

This can be seen on page 5, on desktop and iOS, with the accordions, such as “What we will consider”. The accordion headings title the content within them when expanded. However, they are not marked up as such.

In this section

- Introduction
- Employers
- Member of the public
- Higher education institutes (HEI)
- Third parties
- Guidance for survivors of child abuse while in care in Scotland

We consider concerns about people working in social services already on our Register and people applying to register with us.

If you have a concern or wish to complain about a registered care service, you should contact the [Care Inspectorate](#).

Find out more by choosing the most suitable option for you from the list below.

Employers

Raise a concern as a representative of an employer of social service workers

What we will consider

The matters that we consider are:

- misconduct
- deficient professional practice
- health
- a decision by another regulator
- a conviction.

What you need to tell us

```
<details name="what-we-will-consider" open=""> container  
  <summary> flex  
    <i class="mSvgIcon mSvgIcon--square-plus" aria-hidden="true"> </i>  
    <i class="mSvgIcon mSvgIcon--square-minus" aria-hidden="true"> </i> flex  
    <span class="heading heading--h3">What we will consider</span>  
  </summary>  
  <div class="details_inner"> </div>  
</details>
```

This can also be seen on page 8, on desktop and iOS, with the main accordions, such as “1. What best describes you?”, and the nested accordions within the “4. Details for this level of job role?” accordion, such as “Qualification”. The accordion headings title the content within them when expanded. However, they are not marked up as such.

```

<details name="help-with-register" open=""> (container)
  <summary> (flex)
    <i class="mSvgIcon mSvgIcon--mask mSvgIcon--square-plus" aria-hidden="true"></i>
    <i class="mSvgIcon mSvgIcon--mask mSvgIcon--square-minus" aria-hidden="true"></i> (flex)
    <span class="heading heading--h3">1. What best describes you?</span>
  </summary>
  <div class="details_inner">
    </div>
</details>

```

Steps to reproduce

1. Visually note the headings.
2. Inspect the elements in the DOM.
3. Note that the headings have not been marked up, such as `<h1>`, `<h2>`, and so on.

Recommendation

When elements title or categorise other content, use headings with appropriate heading levels. Make sure that the heading levels used correctly communicate the structure of the page, such as the parent and child relationship between the headings.

Browsers automatically apply `role="presentation"` onto anything contained in `<button>` elements. This is because the HTML and ARIA specifications do not support `<button>` elements with semantically-rich child elements. Make sure that `<button>` elements are used as child elements, rather than parent elements.

In this case, contain the `<button>` elements within the headings. For example:

```

<h1>ACME Rockets</h1>
<h2><button aria-expanded="...">Our History</button></h2>
<h2><button aria-expanded="...">Our Products</button></h2>

```

For more information, see [WAI Tutorials - Headings](#) and [MDN - Button Role - All descendants are presentational](#).

[Return to priority table](#)

ADR_22749-51 **MEDIUM PRIORITY** **RESOLVED**

Buttons not associated with visual group label

Success Criteria 1.3.1: Info and Relationships [2.0 A]

Devices None

Pages Affected None

Users Affected Vision

Component Button

ADR Type Web Audit

Recheck comment 27/06/25

This issue is fixed.

On page 8 on desktop and iOS, the “1. What best describes you?”, “2. Select the type of service you work in?”, and “3. Select the level of your job role?” visual group labels are now associated with the buttons within them.

1. What best describes you?

social care worker
Someone who works in social care such as care at home, housing support, care home and adult day care services, or inspects these types of services.

children and young people's worker
Someone who works with children and young people in day care of children, residential child care and residential school care accommodation services or inspects these types of services.

social worker
A social worker is someone who has a social work degree

social work student
A social work student is someone who has been accepted on a social work degree course

```

<div class="accordion">
  <div class="accordion_wrapper">
    <details id="details--registerPart" name="help-with-register" open=""> <container>
      <summary> <flex>
        <i class="mSvgIcon mSvgIcon--mask mSvgIcon--square-plus" aria-hidden="true"></i>
        <i class="mSvgIcon mSvgIcon--mask mSvgIcon--square-minus" aria-hidden="true"></i> <flex>
        <h3 id="details_label--registerPart" class="heading heading--h3">1. What best describes you?</h3>
      </summary>
      <div class="details_inner">
        <ul class="buttonGroup buttonGroup--stacked list--noStyle" role="group" aria-labelledby="details_label--registerPart">
          <li>
            <button class="mButton mButton--alignLeft" type="button" aria-pressed="false"> <event> <inline-flex>
              <span class="mButton_text">
                <strong class="heading heading--h3">social care worker</strong>
                <span></span>
              </span>
            </button>
          </li>
          <li></li>
          <li></li>
          <li></li>
        </ul>
      </div>
    </details>
  </div>
</div>

```

What happened

If a relationship between elements is expressed with visual cues, it also needs to be programmatically determined. This enables assistive technology to communicate the relationships to users. Content may otherwise seem unstructured or disorganised.

There are buttons that are not associated with their visual group label. The purpose of the buttons may be unclear to users.

Where it happened

This can be seen on page 8, on desktop and iOS, with the “1. What best describes you?”, “2. Select the type of service you work in?”, and “3. Select the level of your job role?” visual group labels for the relevant buttons. The buttons are not associated with their visual group label.

1. What best describes you?

social care worker

Someone who works in social care such as care at home, housing support, care home and adult day care services, or inspects these types of services.

children and young people’s worker

Someone who works with children and young people in day care of children, residential child care and residential school care accommodation services or inspects these types of services.

social worker

A social worker is someone who has a social work degree

social work student

A social work student is someone who has been accepted on a social work degree course

```

<summary> <flex>
  <i class="mSvgIcon mSvgIcon--mask mSvgIcon--square-plus" aria-hidden="true"></i>
  <i class="mSvgIcon mSvgIcon--mask mSvgIcon--square-minus" aria-hidden="true"></i> <flex>
  <span class="heading heading--h3">1. What best describes you?</span>
</summary>
<div class="details_inner">
  <div class="buttonGroup buttonGroup--stacked"> <flex>
    <button class="mButton mButton--alignLeft mButton--primary" type="button"> <event> <flex>
      <span class="mButton_text">
        <span class="heading heading--h3">social care worker</span>
        <span></span>
      </span>
    </button>
    <button class="mButton mButton--alignLeft" type="button"> </button> <event> <flex>
    <button class="mButton mButton--alignLeft" type="button"> </button> <event> <flex>
    <button class="mButton mButton--alignLeft" type="button"> </button> <event> <flex>
  </div>
</div>
</details>
    
```

Steps to reproduce

1. Observe the visual group label for the buttons.
2. Inspect the buttons and review the DOM.
3. Observe that the visual group label is not associated with the buttons.
4. Observe that there is not an alternative labelling method that serves the same purpose.
5. Observe that the buttons do not make sense as stand-alone elements.

Recommendation

Any relationship presented through visual cues must also use the appropriate semantics. Make sure that elements are associated with their visual labels. This can be done with `<fieldset>` and `<legend>`. For example:

```

<fieldset>
<legend>1. What best describes you?</legend>
  <button aria-pressed="true">social care worker...</button>
  ...
</fieldset>
    
```

If this is not possible, you can use `aria-labelledby`. This can be used with custom buttons created with ARIA. For example:

```
<h3 id="buttonsLabel">1. What best describes you?</h3>
<div role="group" aria-labelledby="buttonsLabel">
  <!-- custom buttonss -->
</div>
```

Avoid using multiple labelling techniques on the same element. For example, using `<label>` and `aria-label` together will cause the `aria-label` to overwrite the `<label>`. As well, using `aria-labelledby` will overwrite any other labelling techniques.

For more information, see [WAI Tutorials - Grouping Controls](#) and [Labelling Controls](#).

[Return to priority table](#)

ADR_22749-34 **MEDIUM PRIORITY** **RESOLVED**

Off-screen elements included in reading order

Success Criteria 1.3.2: Meaningful Sequence [2.0 A]

Devices None

Pages Affected None

Users Affected Vision

Component Modal Dialog

ADR Type Web Audit

Recheck comment 26/06/25

This issue is fixed.

On Page 1 on desktop and iOS, when the navigation mega menu is collapsed, the off-screen `<nav>` region is now properly hidden from assistive technologies with the `inert` attribute.

The image displays three parts of a website accessibility audit:

- Top Screenshot:** Shows the main header of the Scottish Social Services Council website. The navigation menu is closed. The main heading is "Supporting a trusted, skilled, confident and valued workforce". Below it, there are two buttons: "More about us" and "Register now".
- Middle Screenshot:** Shows the same website with the navigation menu open. The menu items are: "About us", "Registration", "Dealing with concerns", "Supporting your learning", "Careers and education", and "standards". The "About us" item is highlighted with a red box.
- Bottom Screenshot:** Shows the HTML code for the navigation menu. A red box highlights the attribute `inert=""` on the `mainMenu` container, which is a common accessibility issue as it prevents screen readers from navigating through the menu.

What happened

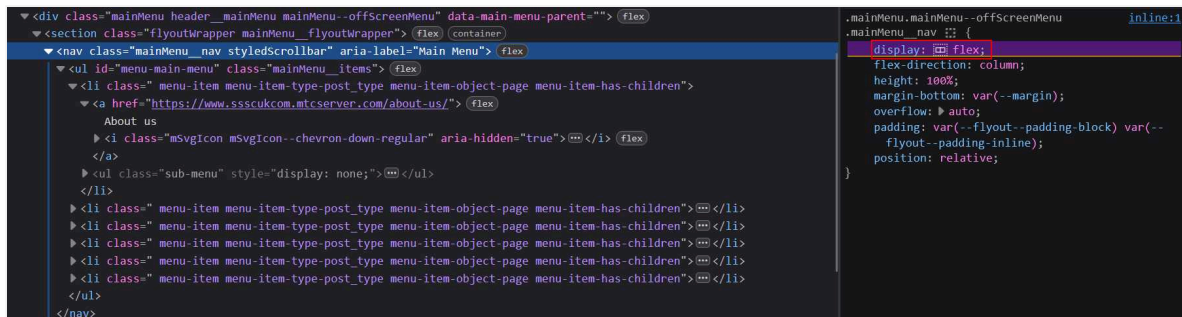
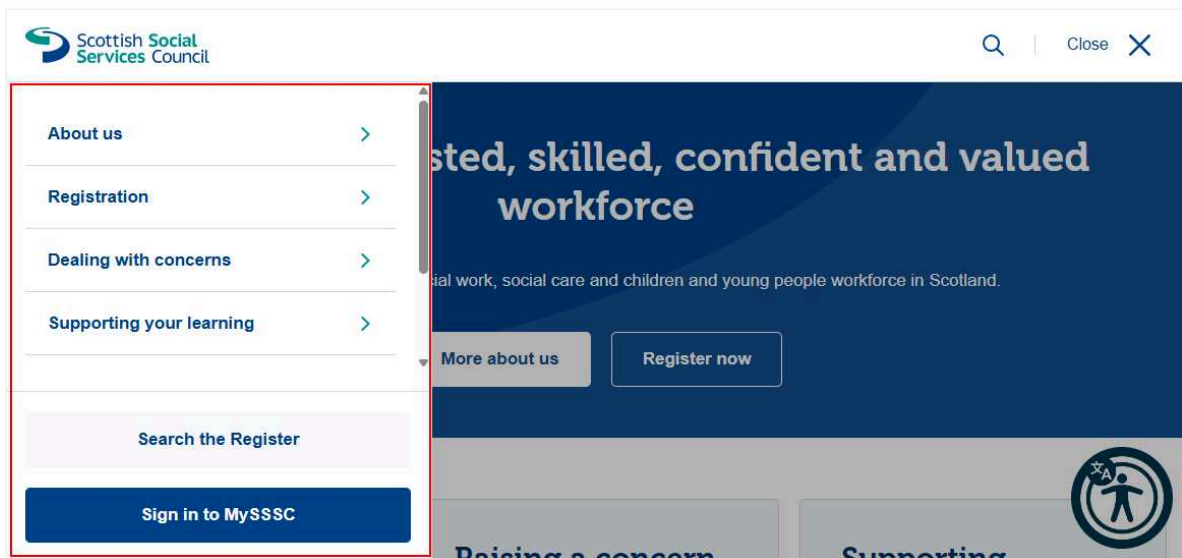
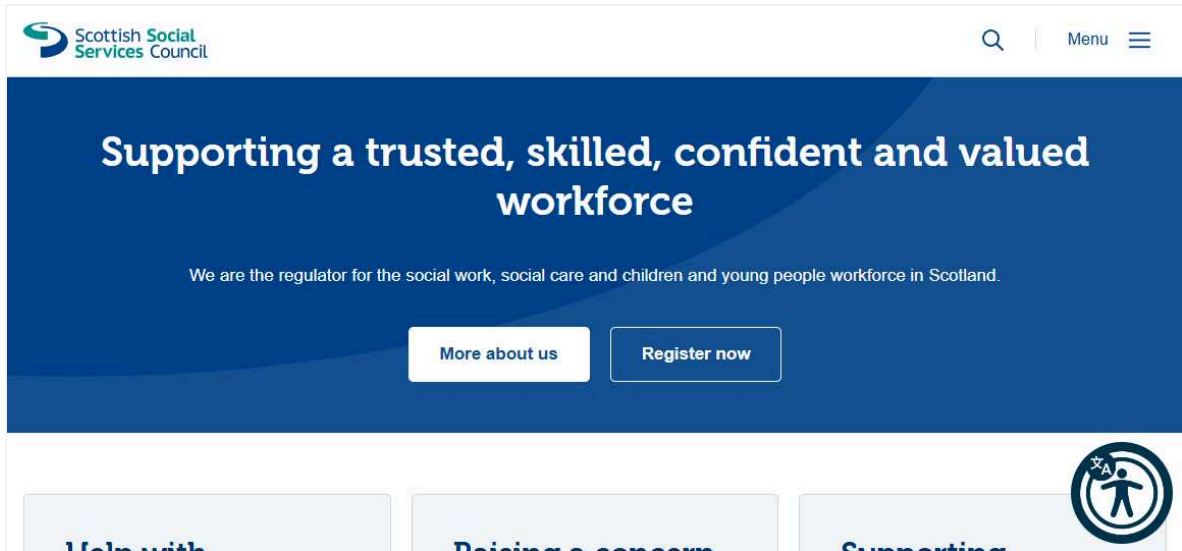
When content has a meaningful order, an appropriate reading order needs to be programmatically determined. This enables those using assistive technology to effectively navigate the content and understand it. Content may otherwise seem unstructured or disorganised.

There are modal dialogs where the programmatic reading order is not logical. When the modal dialog is collapsed, the background elements are still programmatically in the reading order, despite not being part of the visual reading order whilst the modal closed.

Where it happened

This can be seen on page 1 on desktop and iOS, with the navigation mega menu at the top of the page.

When the menu is collapsed, the off-screen `<nav>` region is still inappropriately included in the reading order. That is, it is not hidden from assistive technology.



Steps to reproduce

1. Using a screen reader, activate the modal dialog.
2. Observe the screen reader cursor location.
3. Attempt to navigate away from the modal dialog by swiping or using arrow keys.
4. Observe that the screen reader cursor can access the background elements.

Recommendation

The reading order is determined by the location of elements in the Document Object Model (DOM). Make sure that the elements are placed in the DOM in a logical sequence that screen reader users to navigate through the page in meaningful order.

In this case, when the menu is closed, ensure its contents are programmatically hidden from assistive technologies. When the menu is open, make sure that focus is set onto the menu.

As well, make sure that keyboard and screen reader focus is restricted to the menu. This can be achieved with the following:

- Send the focus to the first element on the modal container. This can be done using the JavaScript `.focus()` method.
- Hide all the other content of the page with `aria-hidden="true"`.
- Remove the interactive elements in the background from the focus order with `tabindex="-1"`.
- Revert the above when the modal is closed. Removing the `aria-hidden` and `tabindex` properties.
- When the dialog is dismissed, use the `.focus()` method to set focus back onto the element that opened it.

The `aria-hidden` and `tabindex` can be supplemented with an inert polyfill. For examples, see [GitHub - WICG - Inert](#), and [inert-polyfill](#).

For more information, see [WAI ARIA - Dialog Example](#), [MDN - aria-hidden](#), [Yale University - Web Accessibility - Focus and Keyboard Operability](#), [MDN - Keyboard](#), [Keyboard Navigable JavaScript widgets](#), [Google Developer - DOM Order Matters](#), and [Using Tabindex](#).

[Return to priority table](#)

ADR_22749-28 **MEDIUM PRIORITY** **RESOLVED**

Focus indication relies on colour alone

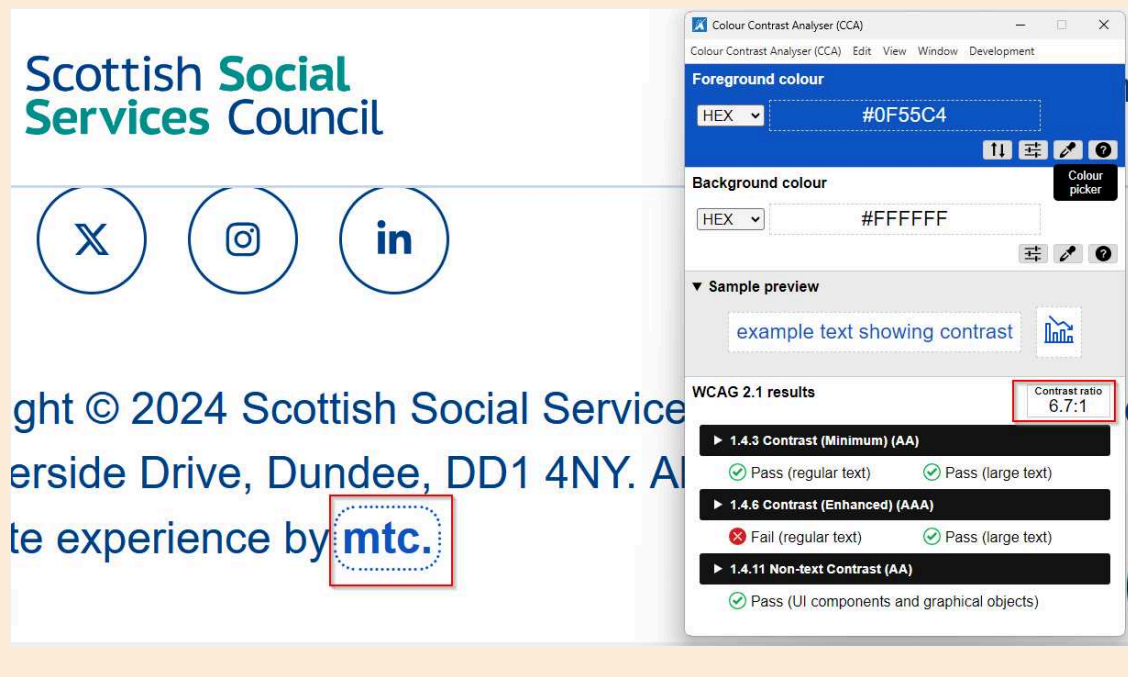
Success Criteria	1.4.1: Use of Color [2.0 A]
Devices	None
Pages Affected	None
Users Affected	Vision, Motor
Component	Link
ADR Type	Web Audit

Recheck comment 26/06/25

This issue is fixed.

On page 1 on desktop, when the “MTC” link at the bottom of the page receives keyboard focus, the link has a blue outline that indicates focus.

Note that the blue focus indicator has a sufficient contrast ratio of 6.7:1 against the white background.



What happened

Understanding content and operating interactive elements must not rely on colour. There is information that is visually communicated exclusively through colour. This may particularly disadvantage low vision and colour blind users who may be excluded by the visual reliance on colour.

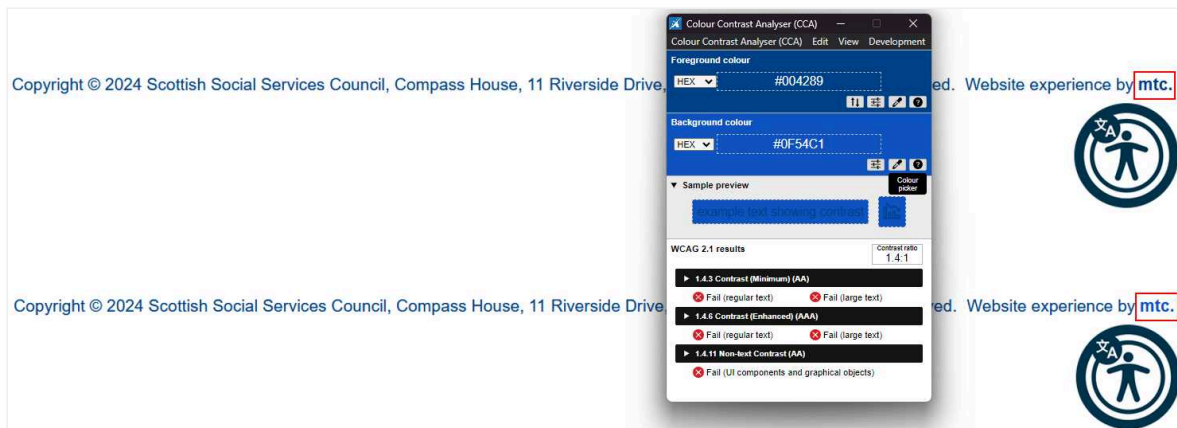
There are interactive elements which receive keyboard focus, but the only visual cue provided relies entirely on colour. A focus indicator relies exclusively on colour if:

- the only difference between the element when it does not have focus and when it does have focus is a change of colour, and
- the contrast ratio between the focussed and unfocussed colours is less than 3:1.

Where it happened

This can be found on page 1 on desktop, with the “MTC” link at the bottom of the page. When the link receives keyboard focus, the only visual cue provided is a change of colour from dark blue (#004289) to a lighter blue (#0F54C1).

However, the difference in contrast between these colours is 1.4:1, which is insufficient for focus indication. Instead, it should be at least 3:1.



Steps to reproduce

1. Use a keyboard to navigate through the above.
2. Observe that the visual cue when the interactive elements receive keyboard focus relies exclusively on a change of colour.
3. Confirm that the contrast ratio between the focussed and unfocussed colours is less than 3:1.

Recommendation

Make sure that information, including focus state information, is not exclusively visually communicated through colour. Colour can be used to communicate information. However, colour must not be the only visual means of conveying information. This success criterion requires that there be visual cues that are available to all users. For more information, see [Princeton - Colour Alone](#).

Make sure that elements provide a visual cue that is not just a change of colour when they receive keyboard focus. This can be achieved using the CSS `focus` pseudo class and `outline` properties. For more information, see [CSS Tricks - Focusing on Focus Styles](#), [GOV - Understanding focus state styles](#), and [Web Dev - Style Focus](#).

Another solution is making sure that default browser indication is supported. This is sometimes undone with CSS reset sheets. Do not overrule browser focus indication without implementing a custom focus indication. For example, do not use `outline: none`. For more information, see [a11y Project - Never Remove CSS Outlines](#).

If it is not possible to use any of those solutions, adjust the focussed and unfocussed colours so that the contrast between them is at least 3:1. The change of focus state would not then be relying on colour alone. The contrast between the two states would be sufficient for the change to be perceivable regardless of colour.

[Return to priority table](#)

ADR_22749-2 **MEDIUM PRIORITY** **RESOLVED**

Reflow requires users to scroll in multiple directions (no exceptions)

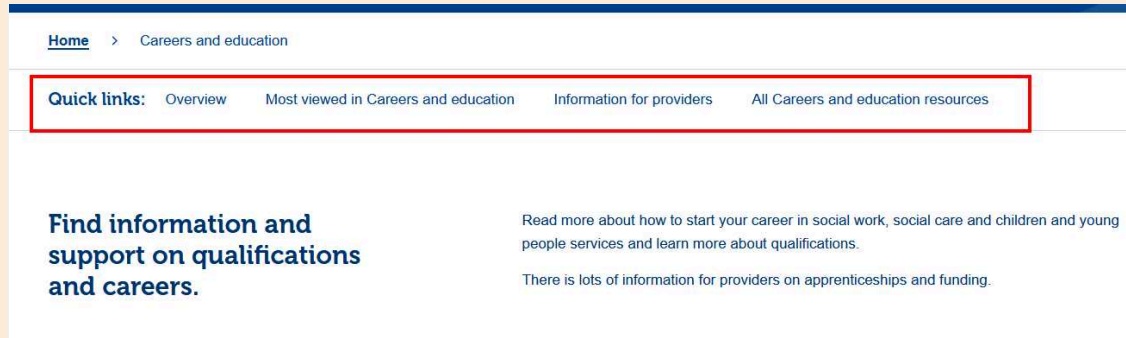
Success Criteria	1.4.10: Reflow [2.1 AA]
Devices	None
Pages Affected	None
Users Affected	Vision
Component	Link, Breadcrumb
ADR Type	Web Audit

Recheck comment 27/06/25

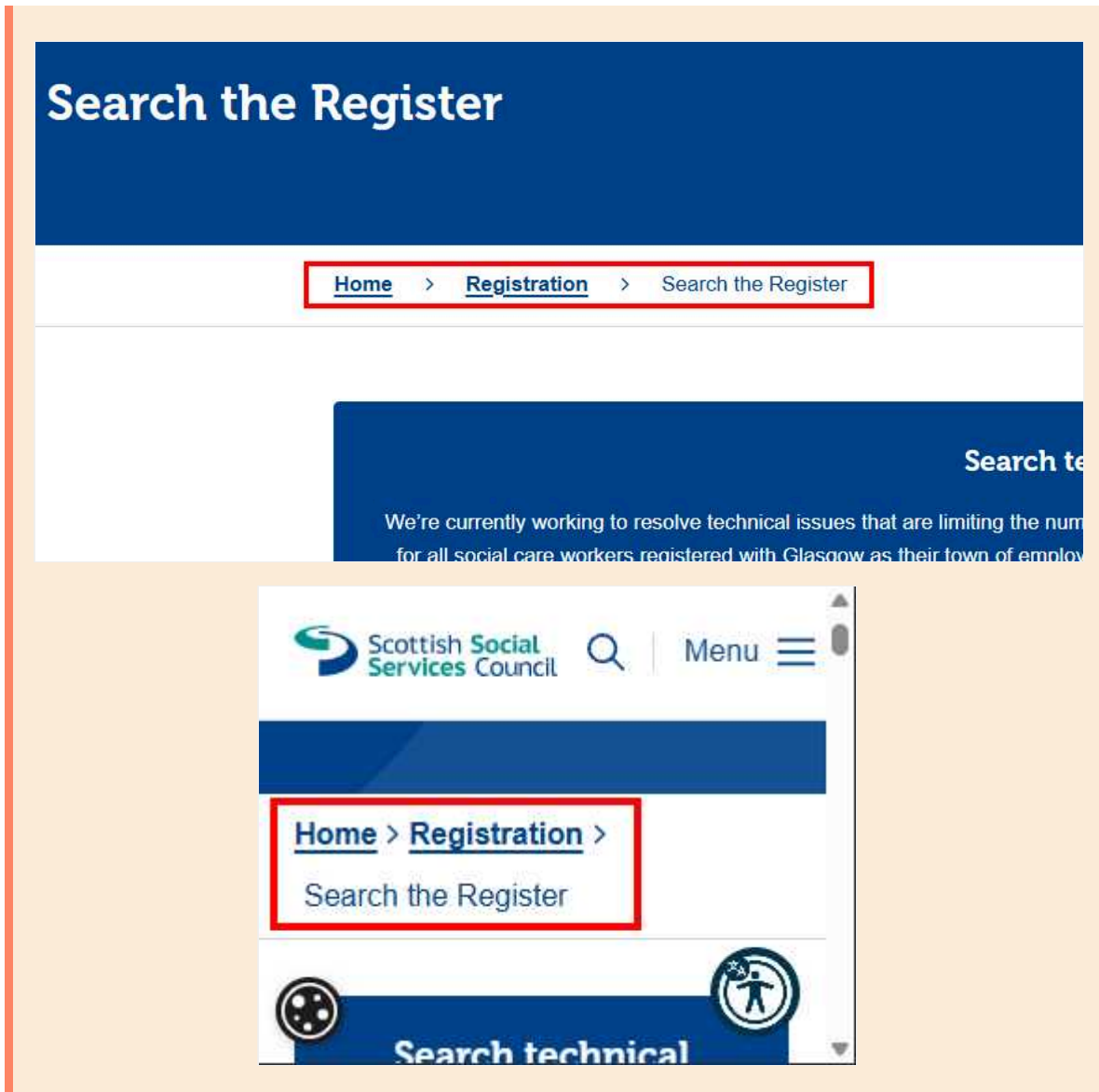
This issue is fixed. :

Fixed with comments: on page 2 on desktop, at 320 x 256px and 100% zoom, the “Quick links” list “Overview”, “Most viewed in Careers and education”, “Information for providers” and “All Careers and education resources do not require to scroll both horizontally and vertically to access all the links.

Note that these fixes have resulted in another issue where there is a loss of content because a vertical white gradient graphic overlaps with a part of the link text. This has been raised as a separate instance in ADR_22749 - 7



Fixed: on page 3 on desktop, at 320 x 256px and 100% zoom, users do not require to scroll both horizontally and vertically to access the full breadcrumbs navigation links.



What happened

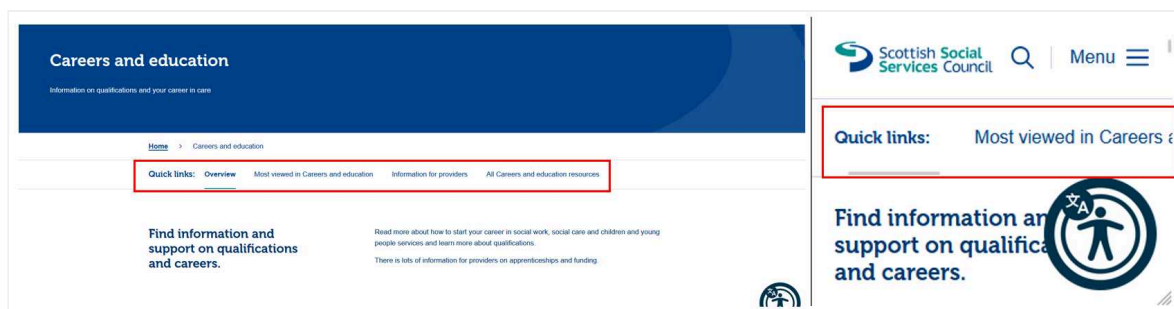
Users must be able to access content and functionality when the page is reflowed. As well, when the page is reflowed, users should not need to navigate up, down, left, and right. The content should be stacked such users only need to scroll in one direction, such as up and down.

This means when the screen size is 320 x 256px and 100% zoom, or 1280 x 1024px and 400% zoom. This is particularly important for low vision users as it can allow them to make content more readable.

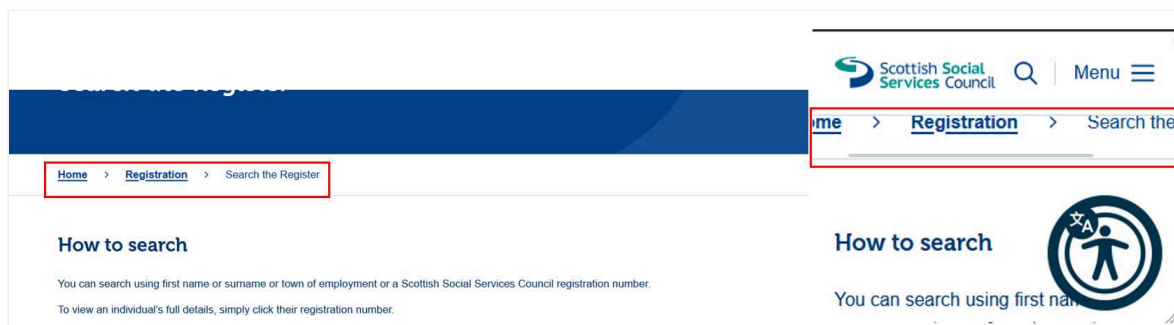
You do not need to reflow content when it would change the meaning of the content. For example, data tables, maps and diagrams, video, games, presentations, and toolbars. However, there are pages without such content that require scrolling in multiple directions when reflowed.

Where it happened

This can be seen on page 2, with the “Quick links” list “Overview”, “Most viewed in Careers and education”, “Information for providers” and “All Careers and education resources”. When the viewport is adjusted to 320 x 256px and 100% zoom, users are required to scroll both horizontally and vertically to access all the links within the “Quick links” list.



This can also be seen on page 3 with the breadcrumbs navigation links. When the viewport is adjusted to 320 x 256px and 100% zoom, users are required to scroll both horizontally and vertically to access the full breadcrumbs list.



Steps to reproduce

1. Observe the functionality at default screen size and zoom.
2. Set the screen size to 320 x 256px and 100% zoom.
3. Alternatively, set the screen size to 1280 x 1024px and 400% zoom.
4. Observe that users are required to scroll in more than one direction.
5. Observe that the content in scope does not meet the exemption criteria.

Recommendation

When the page is reflowed, make sure that the content is stacked such users only need to scroll in one direction. When the viewport is adjusted, make sure that users are not required to scroll in multiple directions (such as up and down, as well as left and right) to access the content and functionality.

As noted, there are some exceptions. You do not need to reflow content when it would change the meaning of the content. For example, a table does not need to be reflowed because the tabular data may not make sense when reflowed. However, any other content must be reflowed.

Use relative units, such as **em**, **ex**, **rem**, **%**, **vw** and modern, flexible layouts techniques to achieve this.

Note that decorative content can be lost, such as images that are not functional or informative. Similarly, the design of the page can change. For example, a mega menu can be converted into a burger menu. If you need to tweak the design to mitigate the need for users to scroll in multiple directions, that is acceptable.

For more information, see [MDN - Relative Units](#), [CSS Tricks - font-size](#), [CSS Tricks - Guide to Flexbox](#), [MDN - Basic Concepts of Flexbox](#), and [CUBE CSS](#). As well, testing can be done with [Web Developer Toolbar](#) or browser dev tools, such as [Chrome - Device Mode](#).

[Return to priority table](#)

ADR_22749-7 **MEDIUM PRIORITY** **RESOLVED**

Reflow causes loss of content

Success Criteria 1.4.10: Reflow [2.1 AA]

Devices None

Pages Affected None

Users Affected Vision

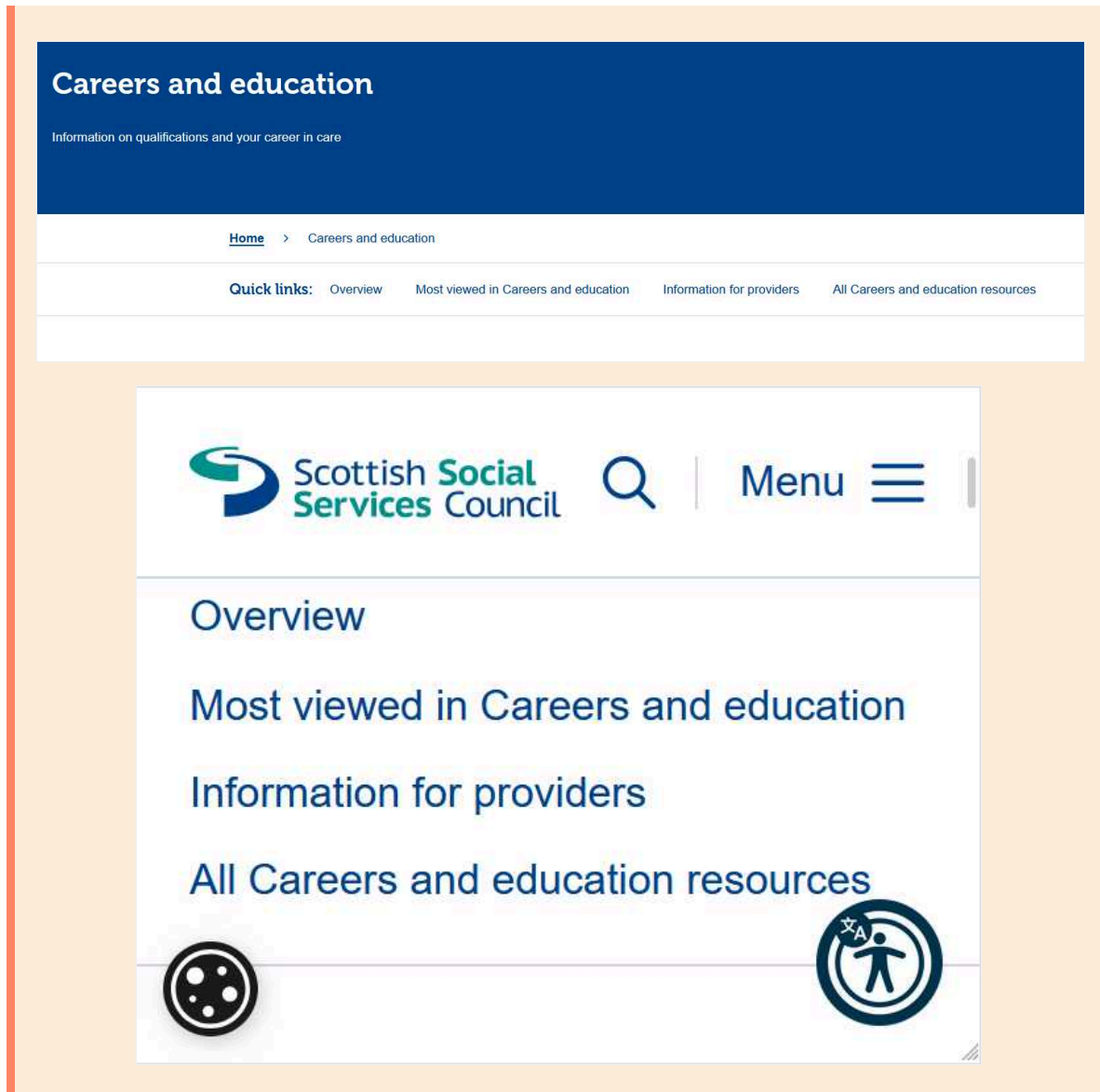
Component Link, Heading, List

ADR Type Web Audit

Recheck comment 17/07/25

The issue is fixed.

Fixed: on page 2, on desktop, the “Quick links” list containing the “Overview”, “Most viewed in Careers and education”, “Information for providers” and “All Careers and education resources” links is not cut off. At 320 x 256 px and 100% zoom, there is no vertical white gradient graphic that overlaps the link text.



Recheck comment 04/07/25

This issue was updated in a recheck.

A new instance of the same issue has been observed on page 2. On this page, there is a loss of content because the vertical white gradient graphic overlaps part of the text.

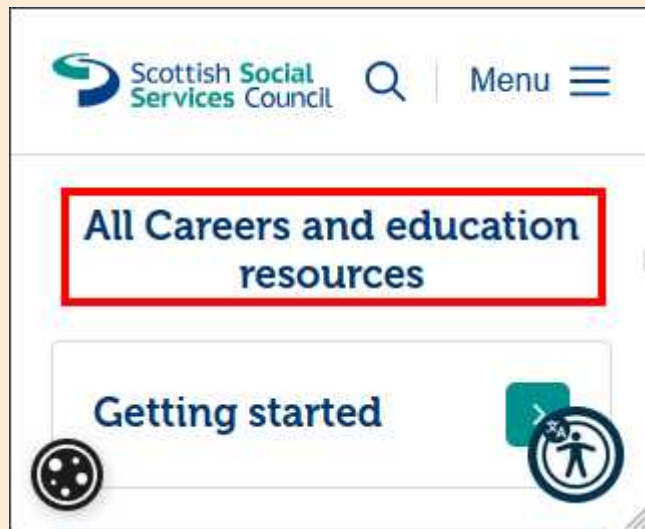
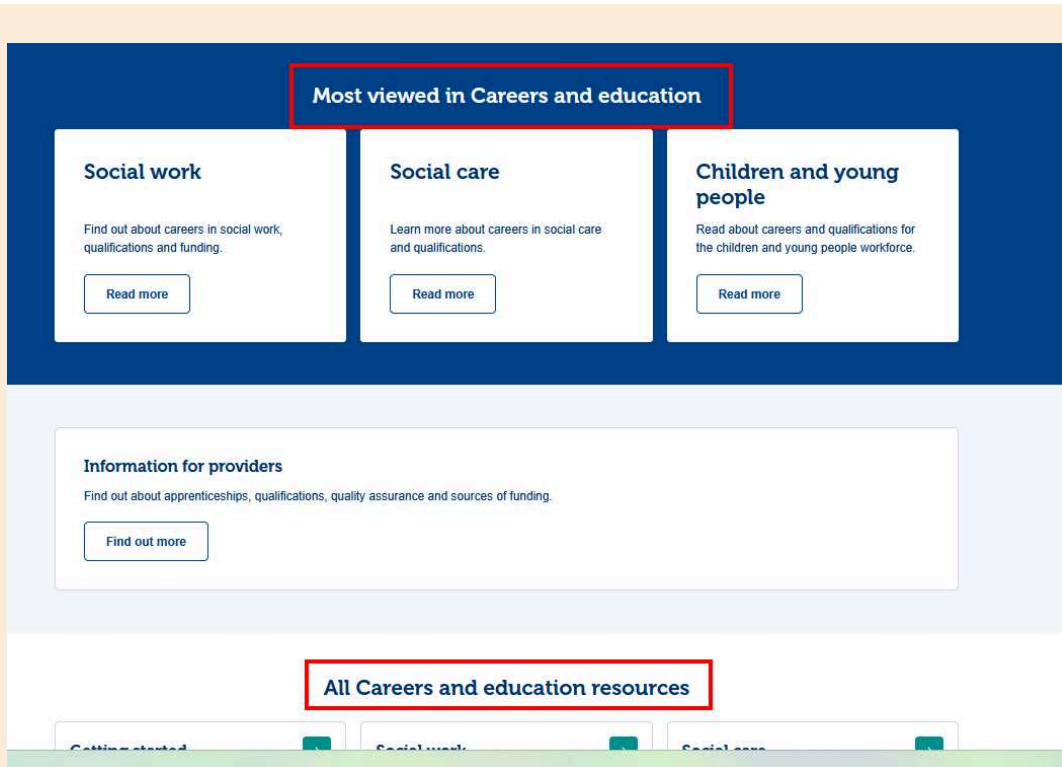
These were not previously raised due to changes to the website. Originally, the pages did not have this behaviour. This can be observed in issues ADR_22749 - 2.

Recheck comment 27/06/25

This issue is partially fixed.

This issue is marked as partially fixed due to the new instance added during the recheck.

Fixed: on page 2 on desktop, at 320 x 256px and 100% zoom, the heading text “Most viewed in careers and education” and “All Careers and education resources” are not cut off in any way.

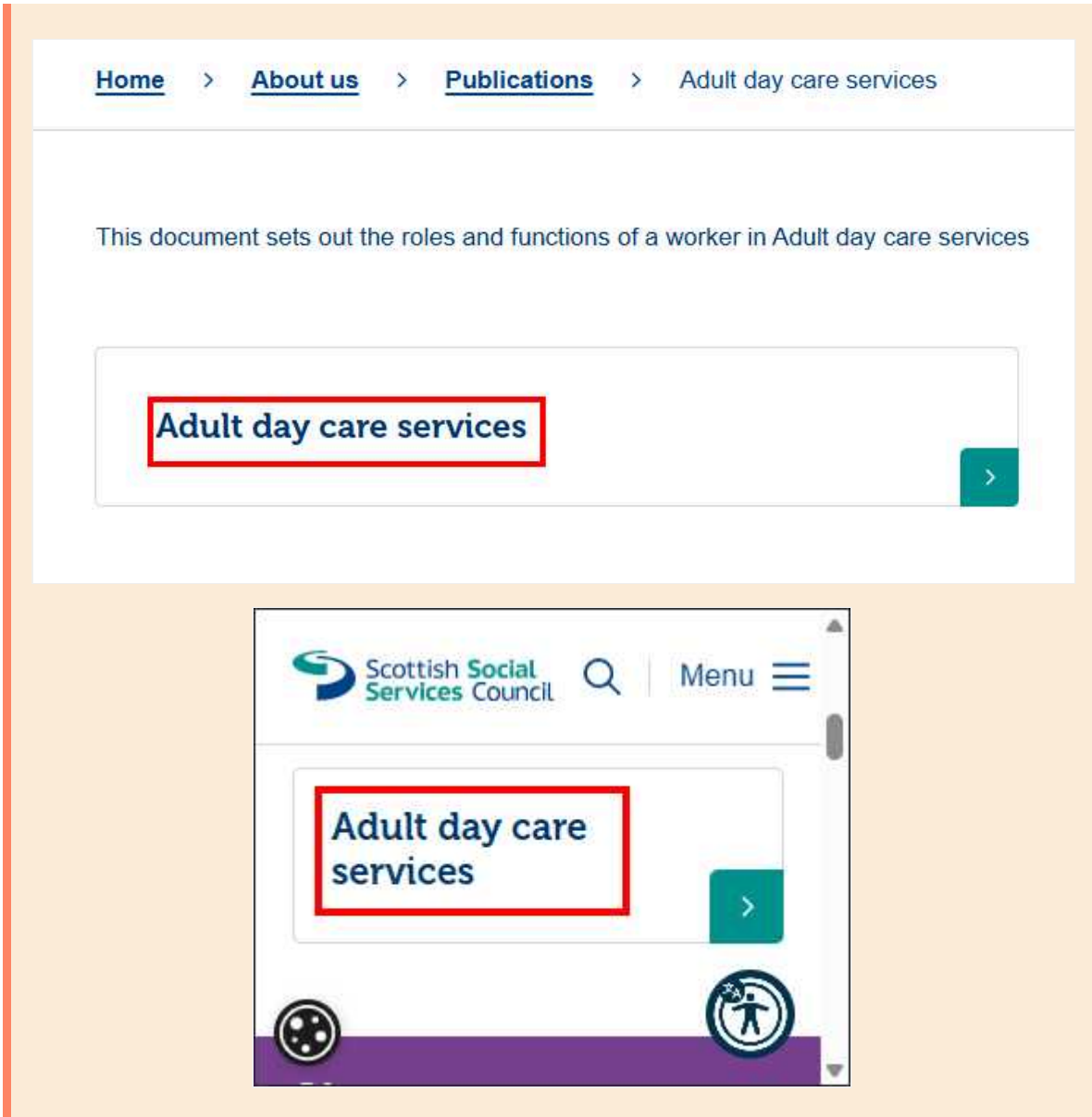


Fixed: on page 3 on desktop, At 320 x 256px and 100% zoom, the "GLENROTHES" text is not cut off.

Individual name	Town of employment	Register part(s)	Registration status	Registration number	Removal date	Suspended from	Suspended to	Specialist award
Amy Margaret Jean Binnie	Glenrothes	social worker	Registered	3005321				No
Amy Margaret Blackwood	Hamilton	children and young people's worker	Registered	3105752				No
Amy Margaret Campbell	Glasgow	children and young people's worker	Registered	4110025				No
Amy Margaret Carle	GLASGOW	social worker	Registered	3078173				No



Fixed: on page 4 on desktop, At 320 x 256px and 100% zoom, the “Adult day care services” link text is not cut off.



Recheck comment 09/06/25

This issue was updated during additional testing on 09/06/25.

The same issue has been observed on pages 3 and 4. On these pages, there is lost content when the page is reflowed.

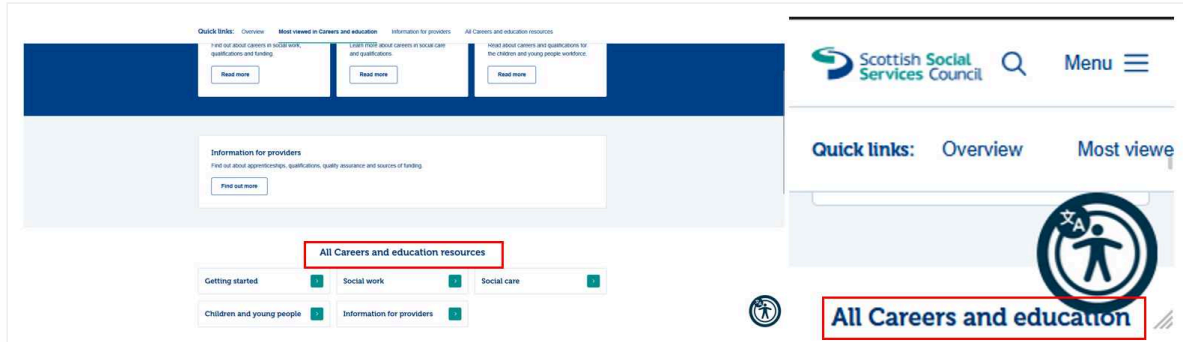
What happened

Users must be able to access content and functionality when the page is reflowed.

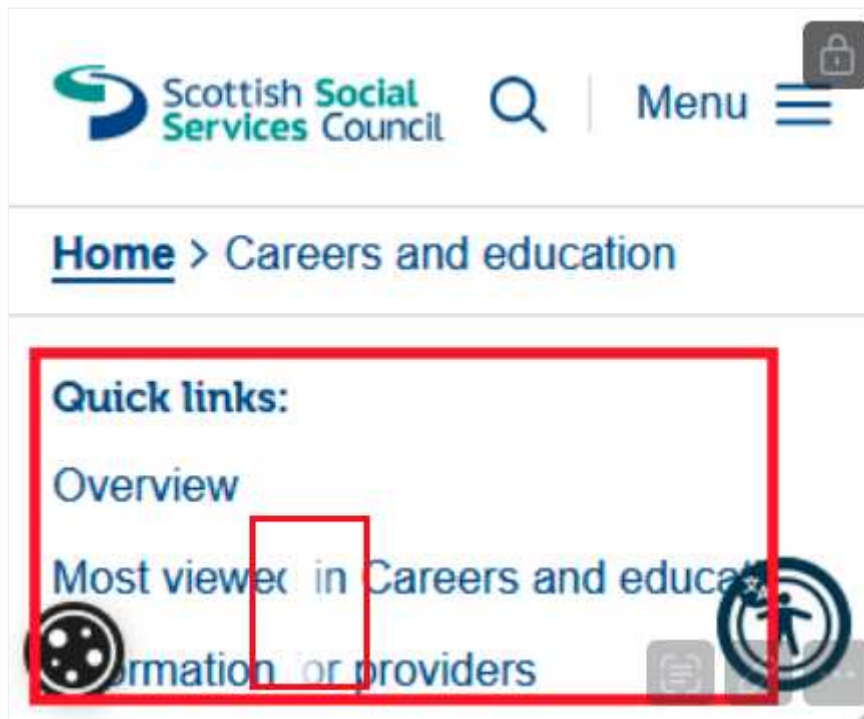
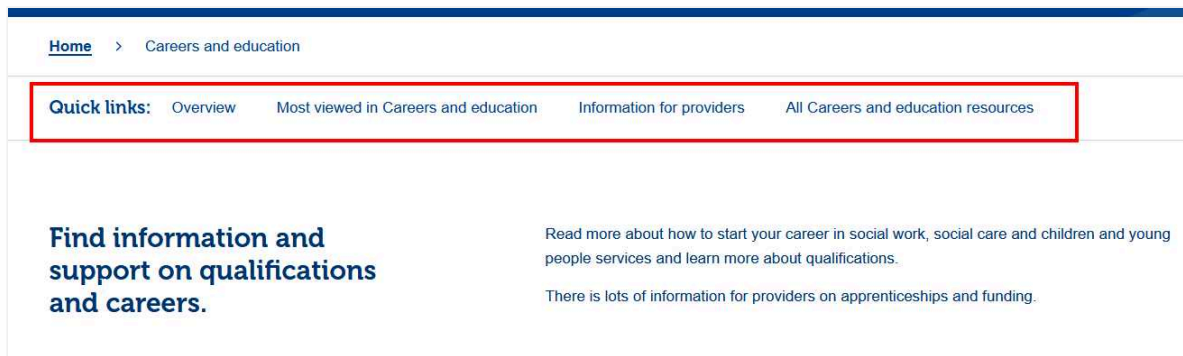
This means when the screen size is 320 x 256px and 100% zoom, or 1280 x 1024px and 400% zoom. This is particularly important for low vision users as it can allow them to make content more readable. There is meaningful or unique content that is lost when reflowed.

Where it happened

This can be seen on page 2 on desktop with the heading text “Most viewed in careers and education” and “All Careers and education”. At 320 x 256px and 100% zoom, the heading text is partially cut off by the “Recite me” icon.



This can be seen on page 2 on desktop with the “Quick links” list “Overview”, “Most viewed in Careers and education”, “Information for providers” and “All Careers and education resources”. At 320 x 256 px and 100% zoom, there’s a loss of content because the vertical white gradient graphic overlaps part of the text.



This can also be seen on page 3, on desktop, with the “GLENROTHERS” town of employment text within the search results. At 320 x 256px and 100% zoom, the “GLENROTHERS” text is partially cut off.

Individual name	Town of employment	Register part(s)	Registration status	Registration number	Removal date	Suspended from	Suspended to	Specialist award
Augustine Benny	Pitlochry	social care worker	Registered	4156943				No
Amy Margaret Jean Binnie	GLENROTHES	social worker	Registered	3005321				No

Suspended To

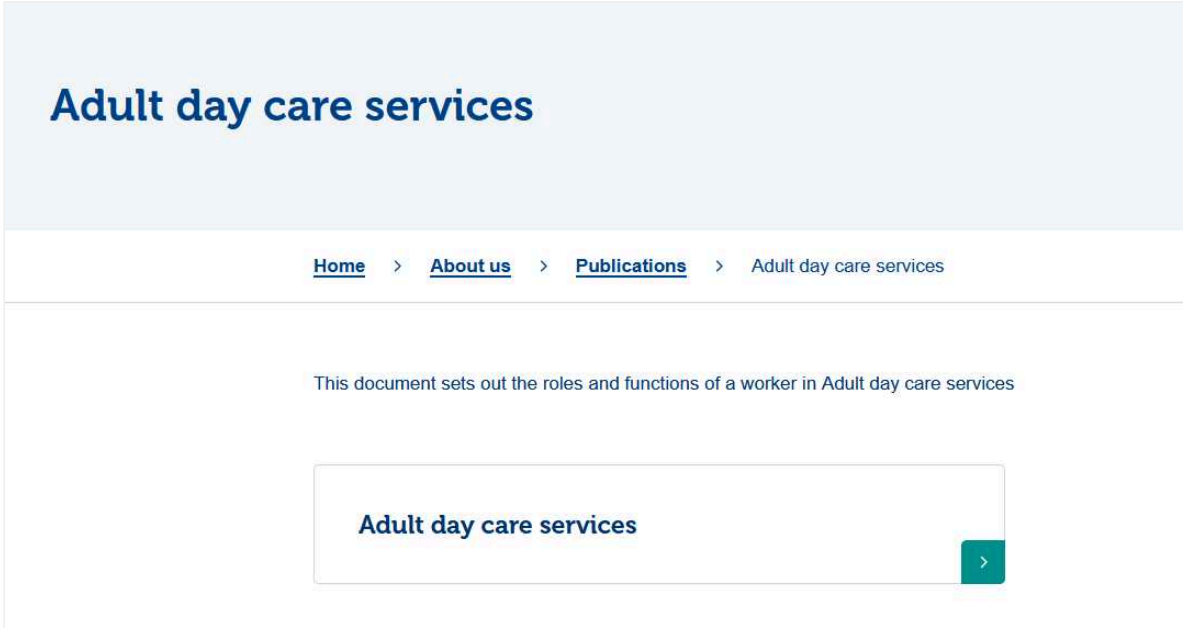
Specialist Award? No

Individual Name Amy
Margaret
Jean Binnie

Town of employment GLENROTHES

Register Part(s) social

This can also be seen on page 3, on desktop, with the “Adult day care services” link. At 320 x 256px and 100% zoom, the link text is partially cut off.



Steps to reproduce

- 1. Observe the content at default screen size and zoom.
- 2. Set the screen size to 320 x 256px and 100% zoom.
- 3. Alternatively, set the screen size to 1280 x 1024px and 400% zoom.
- 4. Observe that the meaningful content is lost.
- 5. Observe that there is no alternative way to access the same content.

Recommendation

Make sure that content is available when the page is reflowed. When the viewport is adjusted, make sure that the content is still available on the page. As well, make sure that users are not required to scroll in multiple directions (such as up and down, as well as left and right) to access the content.

Use relative units, such as **em**, **ex**, **rem**, **%**, **vw** and modern, flexible layouts techniques to achieve this.

Note that decorative content can be lost, such as images that are not functional or informative. Similarly, the design of the page can change. For example, a mega menu can be converted into a burger menu. If you need to tweak the design to allow users to access the meaningful content, that is acceptable.

For more information, see [MDN - Relative Units](#), [CSS Tricks - font-size](#), [CSS Tricks - Guide to Flexbox](#), [MDN - Basic Concepts of Flexbox](#), and [CUBE CSS](#). As well, testing can be done with [Web Developer Toolbar](#) or browser dev tools, such as [Chrome - Device Mode](#).

[Return to priority table](#)

ADR_22749-20 **MEDIUM PRIORITY** **RESOLVED**

Reflow causes loss of functionality

Success Criteria 1.4.10: Reflow [2.1 AA]

Devices None

Pages Affected None

Users Affected Vision

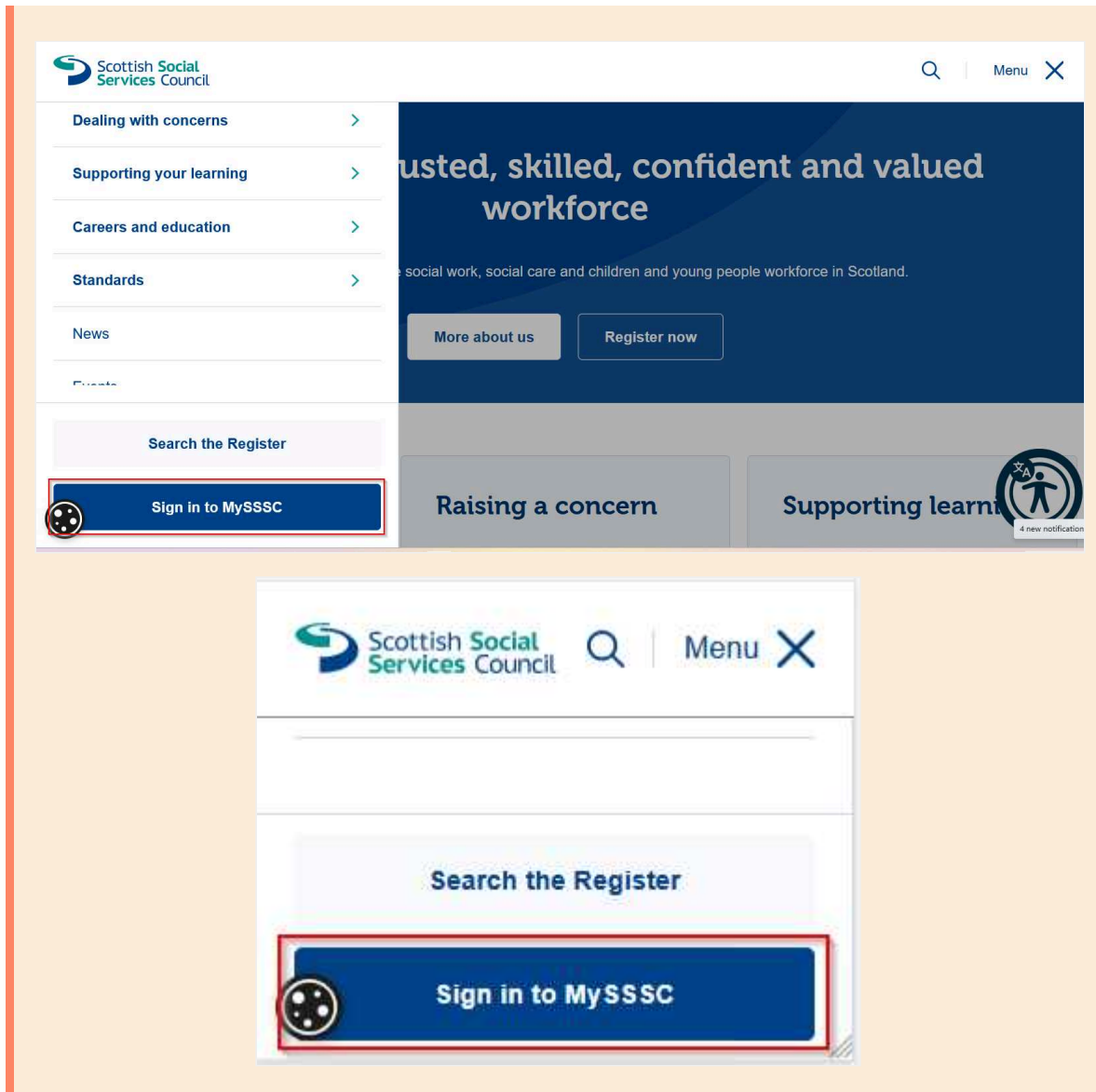
Component Link

ADR Type Web Audit

Recheck comment 26/06/25

This issue is fixed.

On page 1 on desktop, when the page is 320 x 256px and 100% zoom, the "Sign in to MySSSC" link is not lost from the page and can be activated.



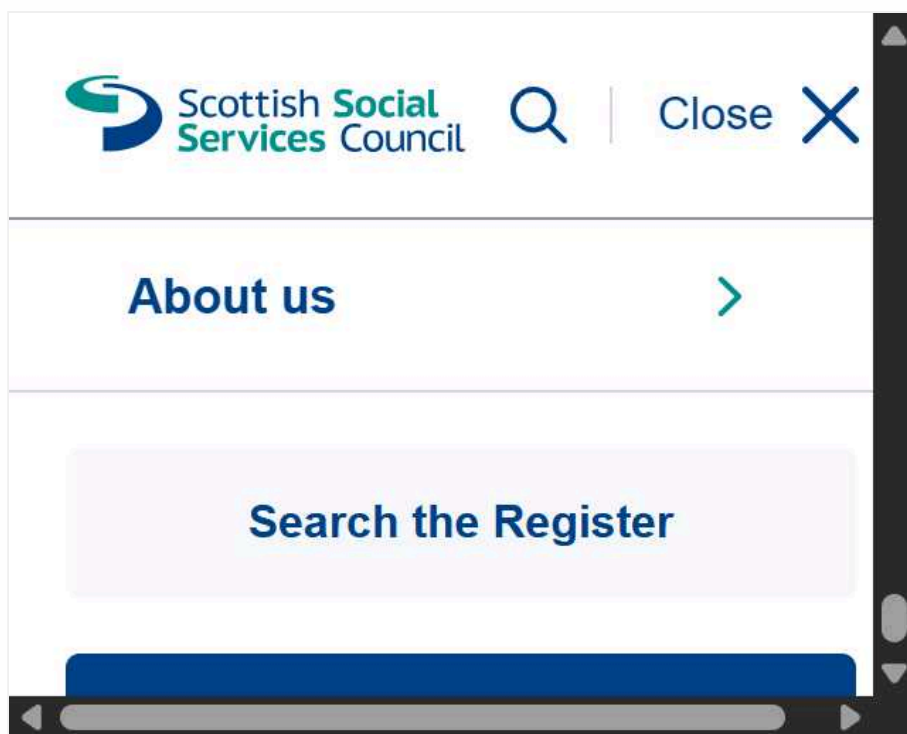
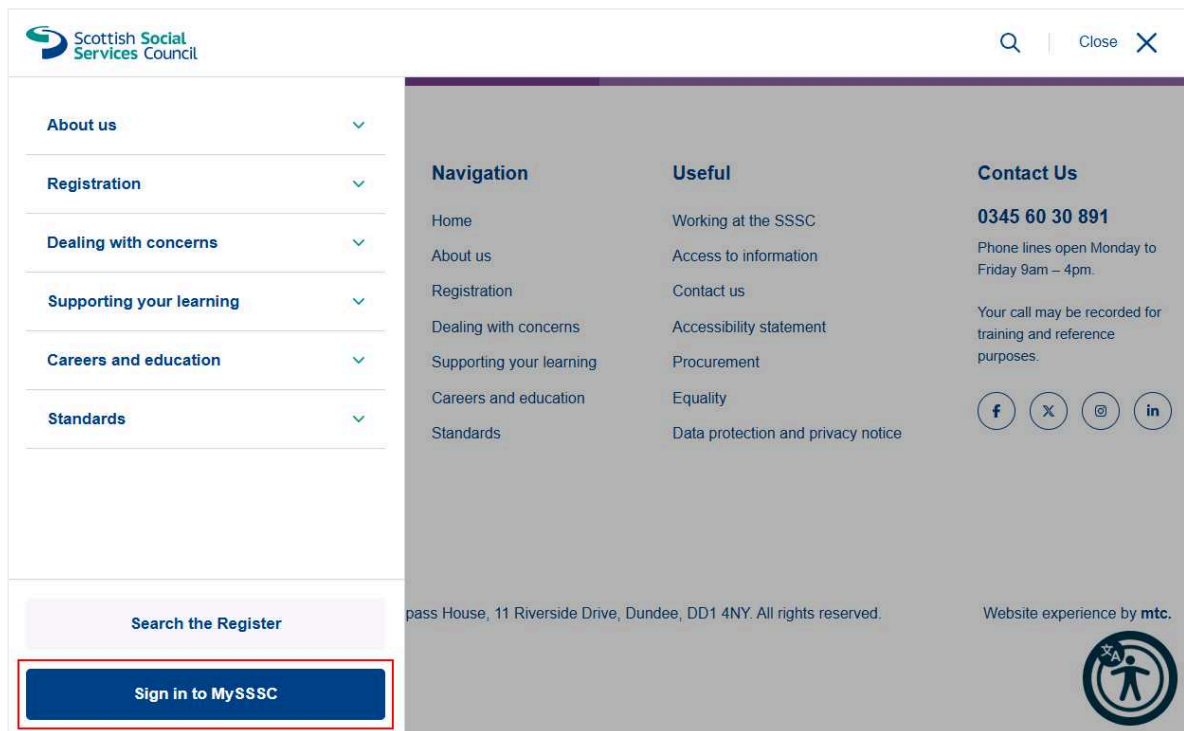
What happened

Users must be able to access content and functionality when the page is reflowed.

This means when the screen size is 320 x 256px and 100% zoom, or 1280 x 1024px and 400% zoom. This is particularly important for low vision users as it can allow them to make content more readable. There is meaningful or unique functionality that is lost when reflowed.

Where it happened

This can be seen on page 1 on desktop, with the “Sign in to MySSSC” link in the mega menu. When the page is subject to 320 x 256px and 100% zoom, the “Sign in to MySSSC” link is lost from the page, and cannot be scrolled into view.



Steps to reproduce

1. Observe the functionality at default screen size and zoom.
2. Set the screen size to 320 x 256px and 100% zoom.
3. Alternatively, set the screen size to 1280 x 1024px and 400% zoom.
4. Observe that the functionality is lost.
5. Observe that there is no alternative way to access the same functionality.

Recommendation

Make sure that functionality is available when the page is reflowed. When the viewport is adjusted, make sure that the functionality is still available on the page. As well, make sure that users are not required to scroll in multiple directions (such as up and down, as well as left and right) to access the functionality.

Use relative units, such as **em**, **ex**, **rem**, **%**, **vw** and modern, flexible layouts techniques to achieve this.

Note that decorative content can be lost, such as images that are not functional or informative. Similarly, the design of the page can change. For example, a mega menu can be converted into a burger menu. If you need to tweak the design to allow users to access the functionality, that is acceptable.

For more information, see [MDN - Relative Units](#), [CSS Tricks - font-size](#), [CSS Tricks - Guide to Flexbox](#), [MDN - Basic Concepts of Flexbox](#), and [CUBE CSS](#). As well, testing can be done with [Web Developer Toolbar](#) or browser dev tools, such as [Chrome - Device Mode](#).

[Return to priority table](#)

ADR_22749-25 **MEDIUM PRIORITY** **RESOLVED**

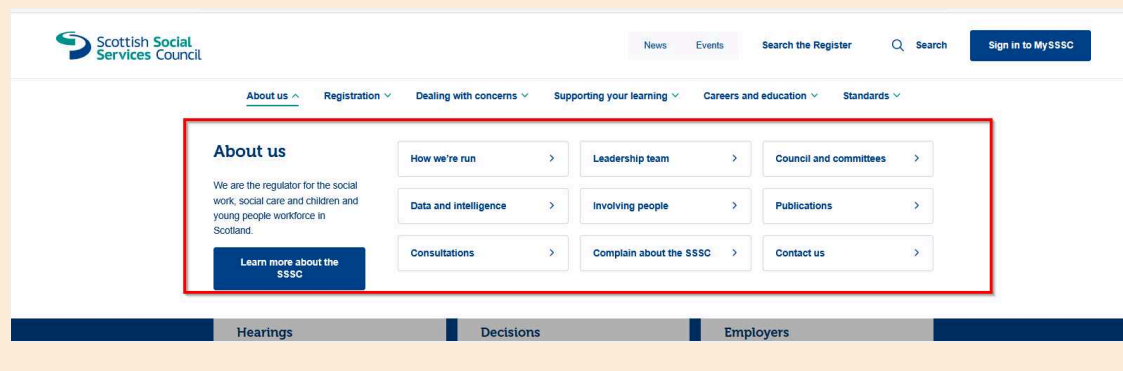
Content triggered by mouse hover or keyboard focus is not dismissable

Success Criteria	1.4.13: Content on Hover or Focus [AA]
Devices	None
Pages Affected	None
Users Affected	Vision
Component	Mega Menu
ADR Type	Web Audit

Recheck comment 26/06/25

This issue is fixed.

On Page 1 on desktop, the mega menu no longer expands on mouse hover. Instead, a button must be clicked to expand the menu. Additionally, the menu can now be collapsed by pressing the Escape (Esc) key.



What happened

Where content is shown when an element receives pointer hover or keyboard focus, the following must be true:

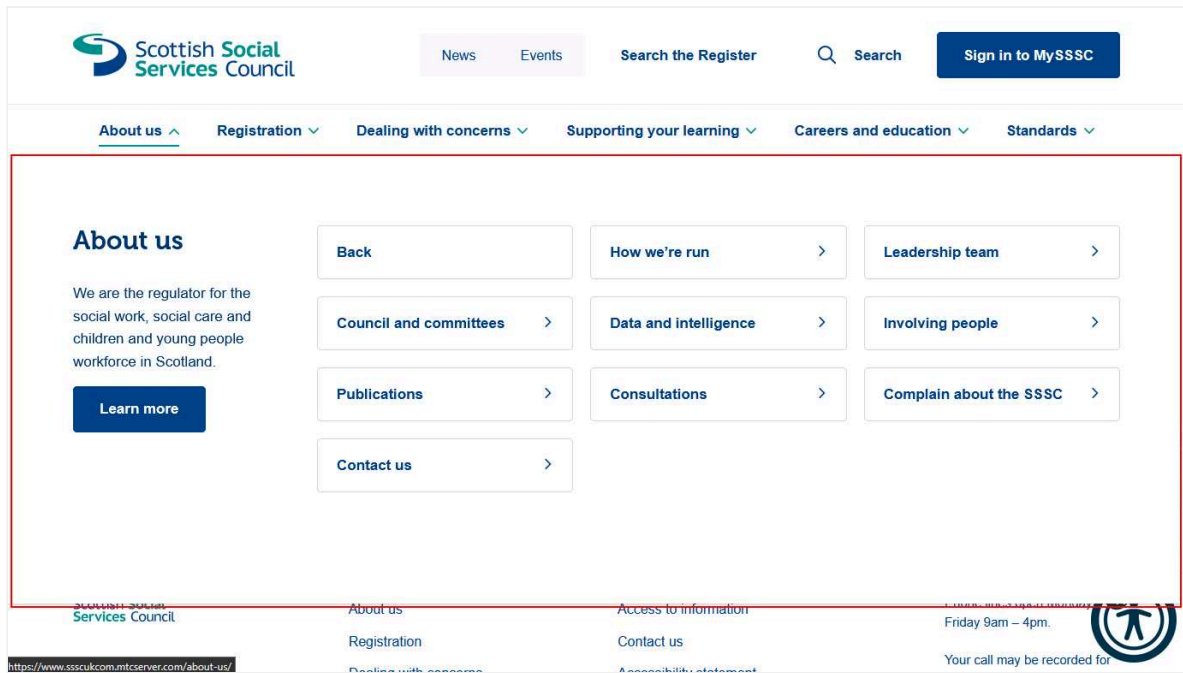
- When the new content obscures other content, it must be possible to dismiss the content without moving the mouse pointer or keyboard focus
- Users must be able to move the mouse pointer over the new content without it being dismissed
- The content must remain persistently visible until the user moves their mouse or keyboard focus onto other elements

There is additional content on mouse hover or keyboard focus that obscures other content and cannot be dismissed.

Where it happened

This can be seen on page 1 on desktop, with the mega menu at the top of the page. On mouse hover, the mega menu expands to show a second layer of navigation links. The menu obscures other content on the page, such as the main heading, the breadcrumb component and the “Quick links”.

However, the menu cannot be dismissed without moving the mouse pointer or keyboard focus, such as by pressing Escape.



Steps to reproduce

1. Navigate onto the trigger element.
2. Observe that additional content is added to the page.
3. Observe that the additional content obscures other content.
4. Observe that you cannot dismiss the additional content without moving the mouse or keyboard focus.

Recommendation

Make sure that users can dismiss the additional content. This is best done by allowing users to dismiss the content using Escape.

Alternatively, refactor the component such that it requires activation, such as mouse click or tabbing onto it and activating it with Return or Space. This success criterion considers components that are manually toggled to be out of scope.

For more information, see [W3C - Making content on focus or hover hoverable, dismissible, and persistent](#) and [Inclusive Components - Tooltips and Toggletips](#).

[Return to priority table](#)

ADR_22749-16 **MEDIUM PRIORITY** **RESOLVED**

In-page links do not move focus to the correct location

Success Criteria	2.1.1: Keyboard [2.0 A]
Devices	None
Pages Affected	None
Users Affected	Motor
Component	Link
ADR Type	Web Audit

Recheck comment 26/06/25

This issue is fixed.

On Page 2 on desktop and iOS, the in-page “Quick links” such as the “Overview”, “Most viewed in Careers and education”, “Information for providers”, and “All Careers and education resources” now function correctly. Upon activation, the page scrolls to the correct section, and the keyboard focus moves appropriately to the targeted content.

For example, when the “All Careers and education resources” link is activated, the page scrolls, and the focus moves to the “Careers and education resources” section. Upon pressing the Tab key, the focus is placed on the “Getting started” link.

Home > Careers and education

Quick links: Overview Most viewed in Careers and education Information for providers **All Careers and education resources** 1

Find information and support on qualifications and careers. Read more about how to start your career in social work, social care and children and young people services and learn more about qualifications. There is lots of information for providers on apprenticeships and funding.

All Careers and education resources 2

Getting started 3 > Social work > Social care > Children and young people > Information for providers > Funding for training >

```
>> document.activeElement
< div section id="all-careers-and-education-resources" class="featuredCTA featuredCTA--small style="--featuredCTA--background.edcta--column-count: 3;" data-quick-links-label="All careers and education resources" tabindex="-1">
>>
```

What happened

Interactive elements must be entirely compatible with a keyboard.

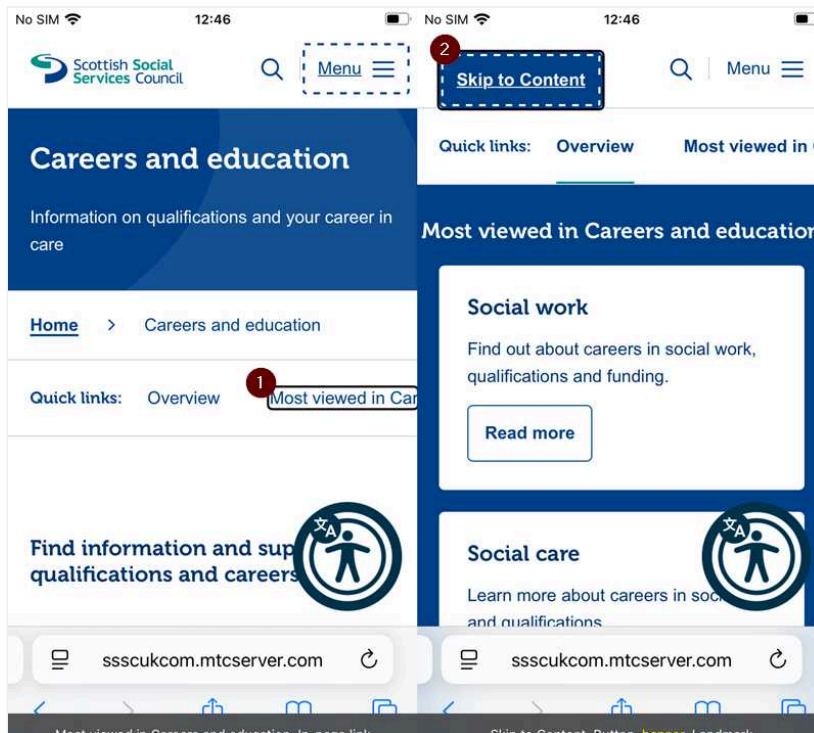
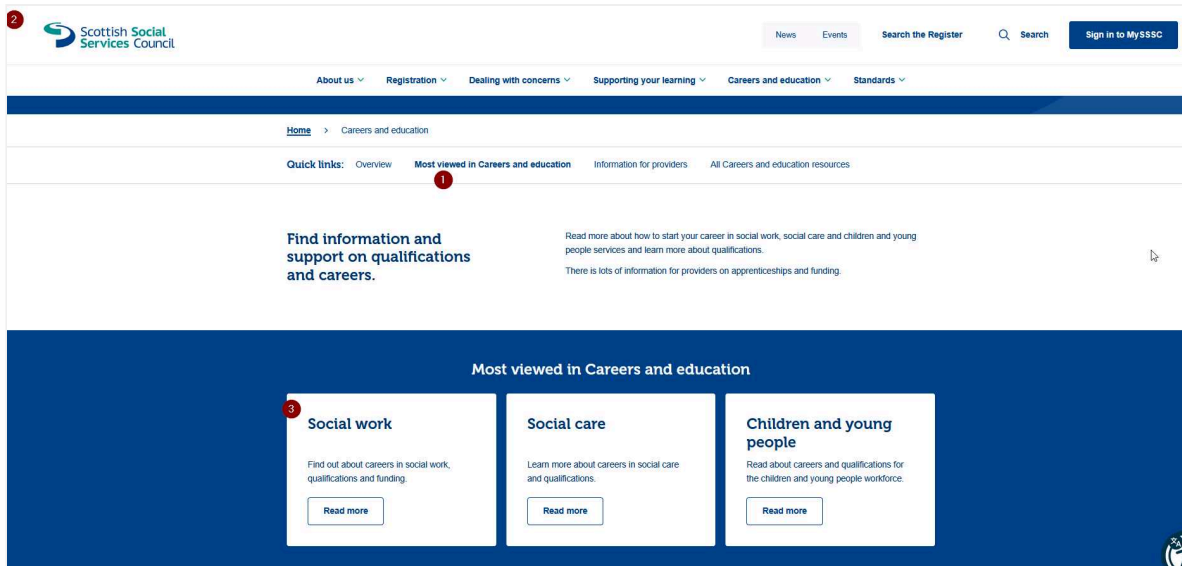
Users should never need to use a mouse. Otherwise, screen reader, keyboard, and switch users, may be unable to control the interactive element. There are same-page links that do not move focus when activated. Instead, focus remains on the same page link. As a result, the same page functionality cannot be accessed with a keyboard.

Note that this has been raised as medium priority, rather than high priority, because it does not block the user journey. As well, users can manually traverse the page to access the elements. However, all users should be able to access the same-page links, and the shortcuts that they provide.

Where it happened

This can be seen on page 2 on desktop and iOS, with the in-page “Quick links” such as “Overview”, “Most viewed in Careers and education”, “Information for providers” and “All Careers and education resources”. Upon activating the link, the page scrolls to the correct section.

However, keyboard focus and the screen reader cursor are not placed on appropriate elements. On desktop, keyboard focus is lost onto the `<body>` element. After Tab is pressed, focus then moves to the first element in the appropriate section. On iOS, the screen reader cursor is moved to the “Skip to Content” button at the top of the page.



```

<nav aria-labelledby="heading--quicklinks-68273f0068535" (flex)
  <h2 id="heading--quicklinks-68273f0068535" class="heading heading--h4"> (flex)
    <ul data-quick-links-nav="" (flex)
      <li>
        <a href="#overview"> (overflow)
      </li>
      <li>
        <a href="#most-viewed-in-careers-and-education"> (overflow)
      </li>
      <li> (flex)
      </li>
      <li> (flex)
      </li>
    </ul>
  </nav>
</div>
</section>
<div id="overview" class="headingText headingText_half" data-quick-links-label="Overview"> (flex)
<section id="most-viewed-in-careers-and-education" class="featuredCTA featuredCTA--none" style="--featuredCta--background-color: var(--secondary-one-color); --featuredCta--column-count: 3;" data-quick-links-label="Most viewed in Careers and education"> (flex)
<section id="information-for-providers" class="featuredCTA featuredCTA--none" style="--featuredCta--background-color: var(--muted-one-color); --featuredCta--column-count: 1;" data-quick-links-label="Information for providers"> (flex)
<section id="all-careers-and-education-resources" class="featuredCTA featuredCTA--small" style="--featuredCta--background-color: var(--white--bg-color); --featuredCta--column-count: 3;" data-quick-links-label="All Careers and education resources"> (flex)
</main>
    
```

```
>> document.activeElement
< body class="wp-singular page-templat_...child-theme-chop-child">
>> |
```

Steps to reproduce

1. Observe the same page links.
2. Navigate onto the links with a keyboard using Tab.
3. Attempt to activate the links with Return or Space.
4. Observe that keyboard focus does not move.

Recommendation

When receiving keyboard focus, make sure that components respond to keyboard commands. In this case, when the same page links are activated, make sure that focus is moved to the relevant part of the page. This can be done with same-page links or using the `focus` method.

As well, use `tabindex="-1"` to allow elements to receive programmatic focus when they are not naturally in the focus order, such as headings. That is, when you are not trying to set focus onto buttons or links, which should be in the focus order already. For example:

```
<ul>
  <li><a href="#SecA">Jump to Terms Starting with A</a></li>
  ...
</ul>
<h2 id="SecA" tabindex="-1">Terms Starting with A</h2>
<dl>
  <dt>ACME</dt>
  <dd>The company name</dd>
  <dt>Aircrafts</dt>
  <dd>What the company produces</dd>
</dl>
```

For examples, see [MDN - focus method](#) and [Dev - Managing Focus for Accessibility](#). For more information, see [MDN - Keyboard Accessibility](#), [Google Developer - Using Tabindex](#), [Yale - Web Accessibility - Links](#), [University of Washington - Links and buttons](#), and [A11y Project - Valid and Accessible Links](#).

[Return to priority table](#)

ADR_22749-13 **MEDIUM PRIORITY** **RESOLVED**

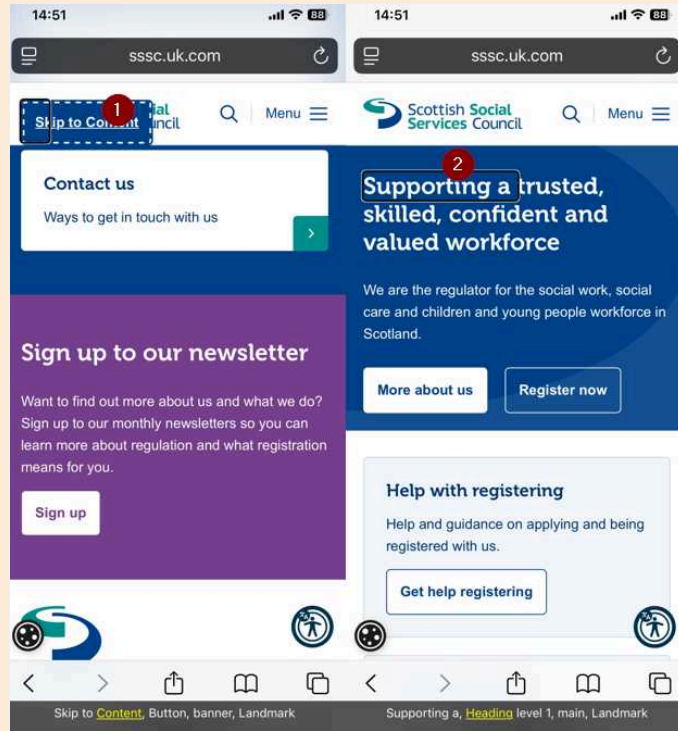
Navigation skip link does not move screen reader focus

Success Criteria	2.4.1: Bypass Blocks [2.0 A]
Devices	None
Pages Affected	None
Users Affected	Motor
Component	Link
ADR Type	Web Audit

Recheck comment 27/06/25

This issue is fixed.

On page 1 on iOS, upon activating the skip links such as “Skip to content” and “Skip to footer” with a screen reader, the focus is moved appropriately to the target content.



```

<header class="header header--fixed"> grid
  <a class="mButton skipToLink skipToLink--Content" role="button" href="#site_content"> flex
    <span class="mButton__text">Skip to Content</span>
  </a>
  <a class="mButton skipToLink skipToLink--Footer" role="button" href="#site_footer"> flex
    <span class="mButton__text">Skip to Footer</span>
  </a>
  <div class="header__wrapper"> ... </div>
  <!--.wrapper.header__wrapper-->
  <div class="header__bottomBar bottomBar bottomBar--borderTop"> ... </div> grid
  <!--.header__bottomBar-->
</header>
<section class="searchDialog searchDialog--hidden"> ... </section>
<div id="siteWrapper" data-v-app=""> ... </div> flex
<!--.siteWrapper-->
  
```

What happened

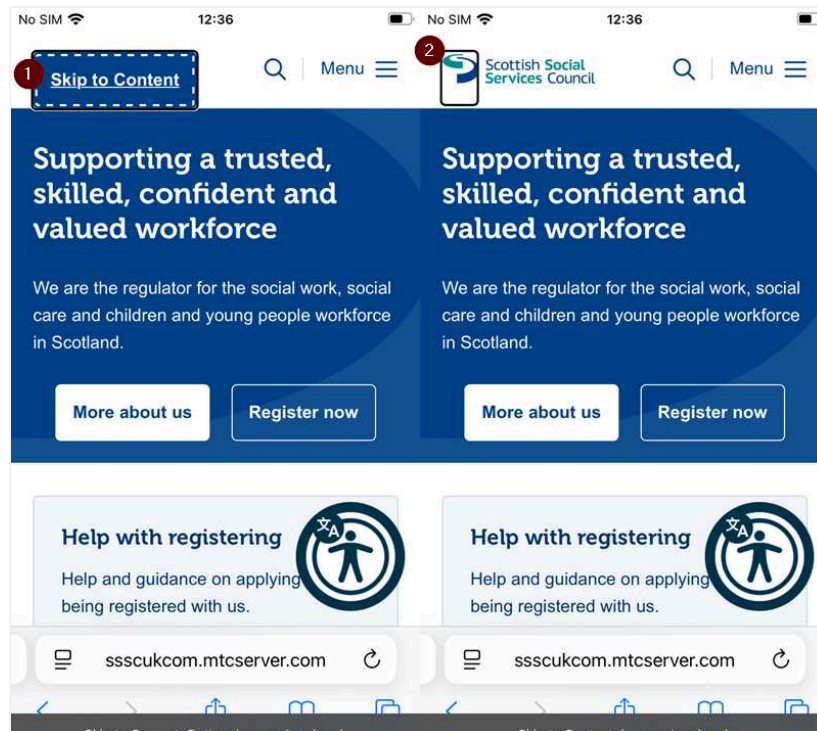
There must be a mechanism for users to bypass repeated blocks of content, such as navigation menus.

There are skip links. However, after activating them with a screen reader on a mobile device, focus does not bypass the content and move to the target. It remains on the skip link. The inoperable skip link may make navigating the pages much more time consuming.

Where it happened

This issue can be seen on page 1 on iOS with the Skip links such as “Skip to content” and “Skip to footer”. Upon activating the skip link with a screen reader, focus is not moved and stays on the button.

Note: this works correctly on desktop with a screen reader.



```

    <a class="mButton skipToLink skipToLink--Content" role="button" href="#site_content">
      <span class="mButton_text">Skip to Content</span>
    </a>
    <a class="mButton skipToLink skipToLink--Footer" role="button" href="#site_footer"></a>
    <div class="header_wrapper"></div>
    <!--_wrapper.header_wrapper-->
    <div class="header_bottomBar bottomBar bottomBar--borderTop"></div>
    <!--.header_bottomBar-->
  </header>
</section>
  <section class="searchDialog searchDialog--hidden">

```

Steps to reproduce

1. Activate the skip link with a screen reader on mobile.
2. Observe that the screen reader focus has not been correctly updated.

Recommendation

Make sure that skip links send keyboard focus to the appropriate place. For example, navigation skip links should send focus to the `<main>` container or the first element in the DOM after the navigation menu, such as a heading element. For example:

```

<body>
  <a href="#main">Skip to main content</a>
  <nav role="navigation">
    <ul>
      <li><a href="#">Home</a></li>
      <li><a href="#">About</a></li>
    </ul>
  </nav>
  <main id="main">
</main>
</body>

```

Skip links can be hidden by default. This can be achieved with a CSS class that visually hides them, but keeps them present in the DOM. However, they must become visible when receiving keyboard focus. This can be achieved using the CSS **focus** pseudoclass.

For more information on skip links, see [a11y Project - Skip Links](#), [WCAG - G1 Adding a link at the top of each page](#) and [Axess Labs - Your Skip Links are Broken](#).

[Return to priority table](#)

ADR_22749-18 **MEDIUM PRIORITY** **RESOLVED**

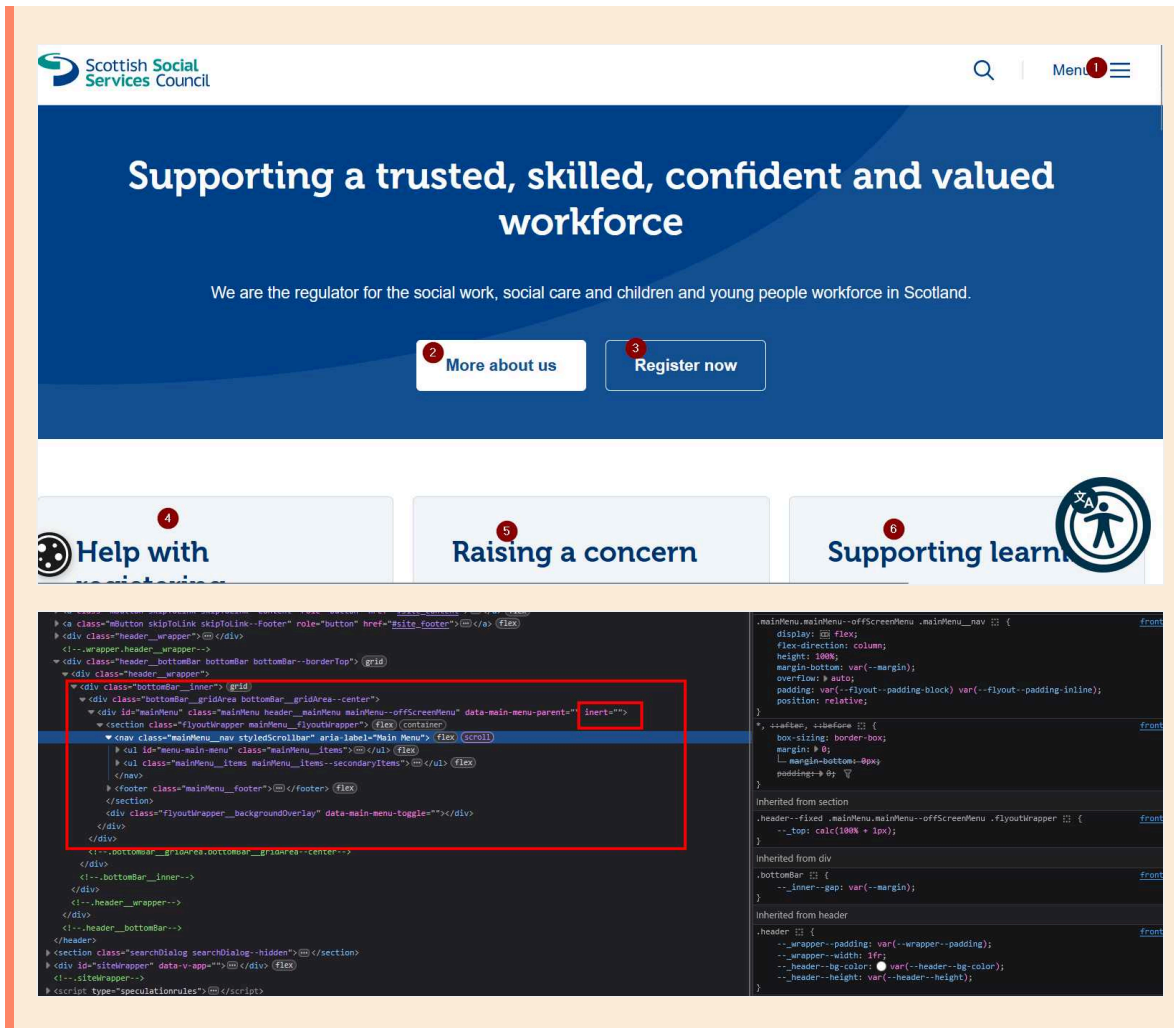
Focus order includes off-screen interactive elements

Success Criteria	2.4.3: Focus Order [2.0 A]
Devices	None
Pages Affected	None
Users Affected	Motor
Component	Link, Mega Menu
ADR Type	Web Audit

Recheck comment 26/06/25

This issue is fixed.

On page 1 on desktop and iOS, when the menu is collapsed, the interactive elements such as “About us”, “Registration” and so on are appropriately removed from the focus order by using the `inert` attribute on the parent `<div>`.



What happened

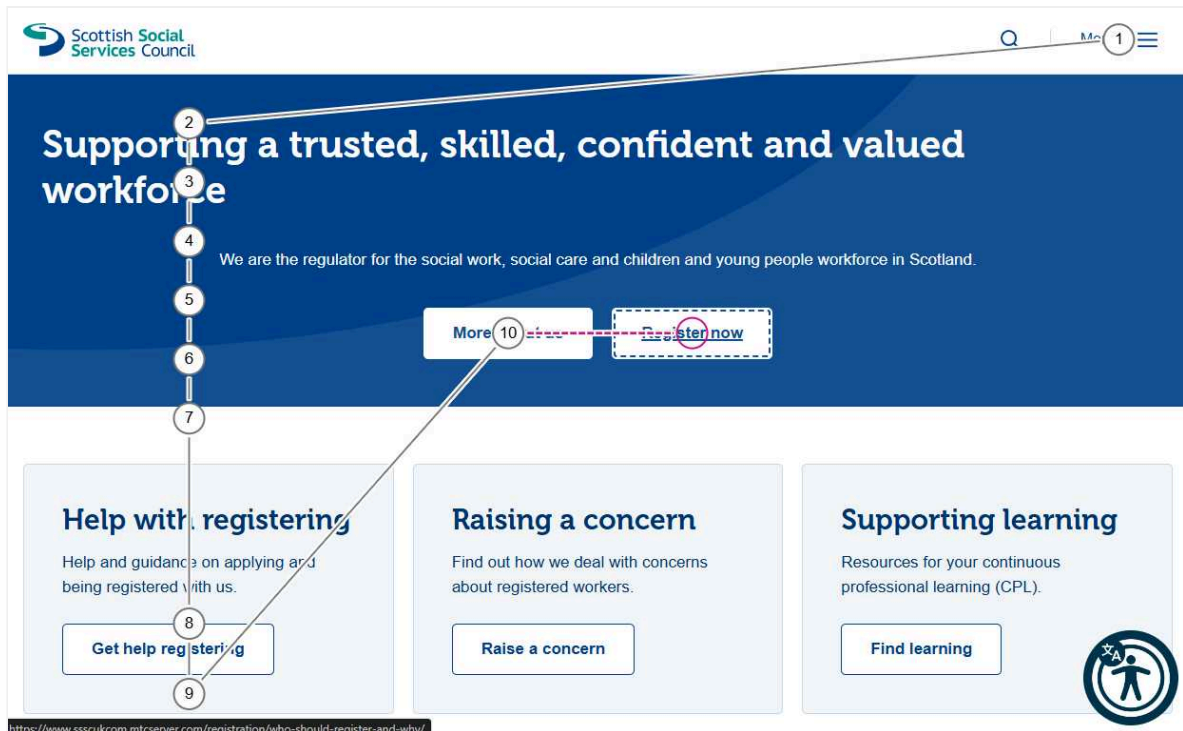
The keyboard focus order must be meaningful. It does not need to mirror the visual order. However, interactive elements must receive focus in a way that makes some sort of sense. The focus order must not be totally illogical or make the operability of components more challenging.

There are off-screen interactive elements that receive focus. This may particularly disadvantage keyboard users who often navigate by tabbing through the interactive elements. That invisible interactive elements receive focus may confuse or frustrate users.

Where it happened

This can be seen on page 1 on desktop and iOS, with the navigation mega menu at the top of the page.

When the menu is collapsed, the interactive elements such as "About us", "Registration" and so on are still inappropriately included in the focus order. That is, instead of removed from the focus order.



https://www.ssscukcom.mtcserver.com/registration/who-should-register-and-why/

```

<div class="mainMenu header_mainMenu mainMenu--offScreenMenu" data-main-menu-parent=""> flex
  <section class="flyoutWrapper mainMenu_flyoutWrapper"> flex container
    <nav class="mainMenu_nav styledScrollbar" aria-label="Main Menu"> flex
      <ul id="menu-main-menu" class="mainMenu_items"> flex
        <li class=" menu-item menu-item-type-post_type menu-item-object-page menu-item-has-children">
          <a href="https://www.ssscukcom.mtcserver.com/about-us/"> flex
            About us
            <i class="mSvgIcon mSvgIcon--chevron-down-regular" aria-hidden="true"> </i> flex
          </a>
          <ul class="sub-menu" style="display: none;"> </ul>
        </li>
        <li class=" menu-item menu-item-type-post_type menu-item-object-page menu-item-has-children">
          <a href="https://www.ssscukcom.mtcserver.com/registration/"> flex
            Registration
            <i class="mSvgIcon mSvgIcon--chevron-down-regular" aria-hidden="true"> </i> flex
          </a>
          <ul class="sub-menu" style="display: none;"> </ul>
        </li>
        <li class=" menu-item menu-item-type-post_type menu-item-object-page menu-item-has-children"> </li>
        <li class=" menu-item menu-item-type-post_type menu-item-object-page menu-item-has-children"> </li>
        <li class=" menu-item menu-item-type-post_type menu-item-object-page menu-item-has-children"> </li>
        <li class=" menu-item menu-item-type-post_type menu-item-object-page menu-item-has-children"> </li>
      </ul>
    </nav>
    <footer class="mainMenu_footer"> </footer> flex
  </section>
  <div class="flyoutWrapper_backgroundOverlay" data-main-menu-toggle=""></div>
</div>
    
```

Steps to reproduce

1. Using just your keyboard, tab through the above.
2. Observe that focus goes onto elements that are not visible on the page.

Recommendation

The focus order is determined by the location of focusable elements in the Document Object Model (DOM). Make sure that the interactive elements are placed in the DOM in a logical sequence that allows keyboard users to tab through the page in meaningful order.

In this case, make sure that off-screen elements do not receive keyboard focus.

If they are not meant to be interactive at any point, remove the elements from the DOM, comment them out, or remove them from the focus order. If they are meant to be interactive at some point, make sure that they are excluded from the focus order and included in the focus order as appropriate.

Removing elements from the focus order can be achieved with the HTML `hidden` or CSS `display:none`. Both will hide the element from all users. This same can be programmatically achieved by using both `aria-hidden="true"` and `tabindex="-1"` on the same element. For example:

```
<button hidden> ... </button>  
<button class="visuallyOffScreen" aria-hidden="true" tabindex="-1"> ... </button>
```

However, note that `aria-hidden="true"` and `tabindex="-1"` will not visually remove elements from the page.

Removing these attributes will re-insert the elements into the focus order based on their location in the DOM, allowing users to navigate onto them with a keyboard. Make sure that this is done as appropriate, such as when the user toggles a parent control that shows and hides these interactive elements.

For more information, see [Yale University - Web Accessibility - Focus and Keyboard Operability](#), [MDN - Keyboard](#), [Keyboard Navigable JavaScript widgets](#), [Google Developer - Using Tabindex](#), [MDN - aria-hidden](#), and [MDN - hidden](#).

[Return to priority table](#)

ADR_22749-22 **MEDIUM PRIORITY** **RESOLVED**

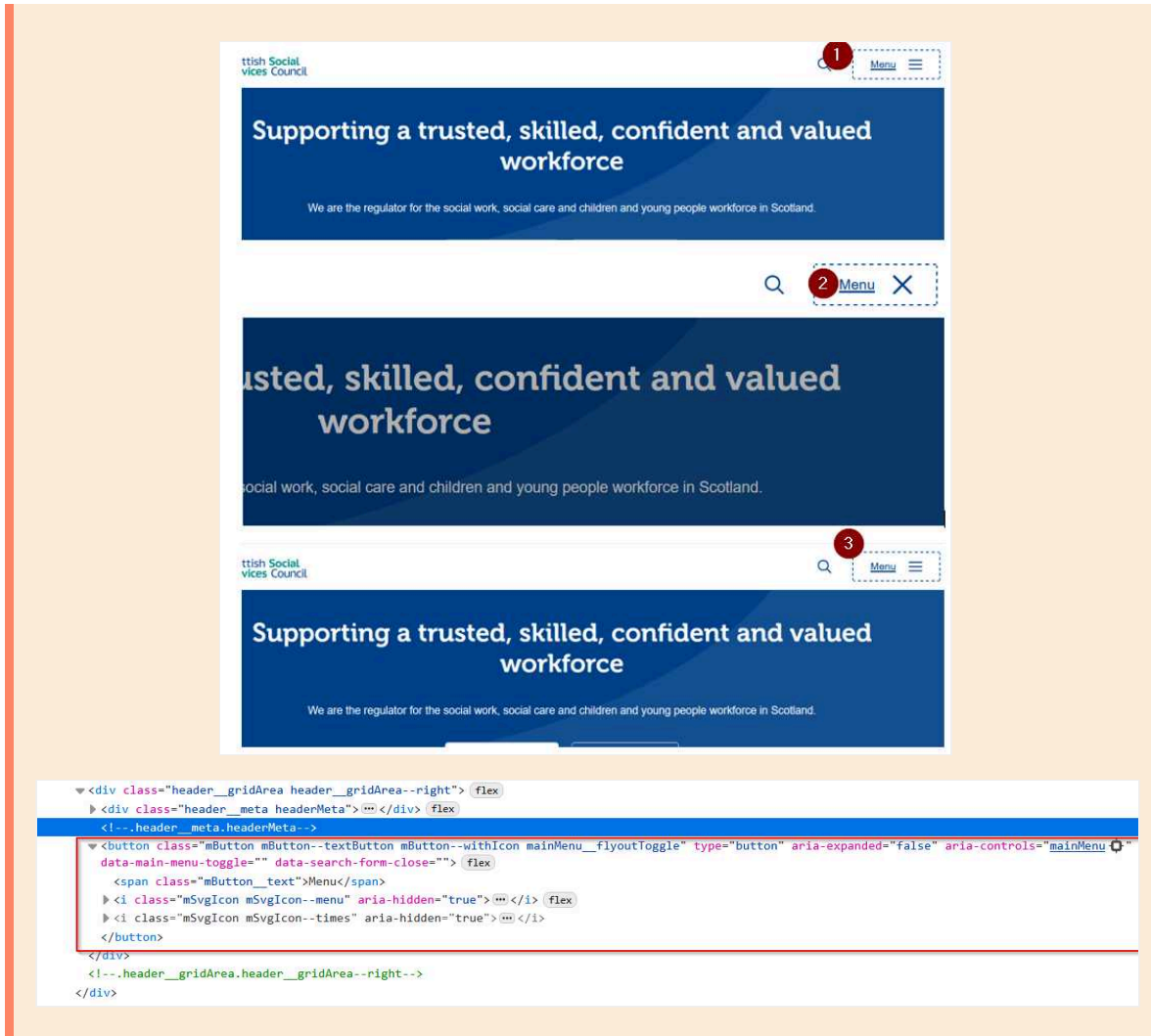
Focus does not return to triggering element

Success Criteria	2.4.3: Focus Order [2.0 A]
Devices	None
Pages Affected	None
Users Affected	Motor
Component	Button
ADR Type	Web Audit

Recheck comment 26/06/25

This issue is fixed.

On page 1 on desktop, the focus is returned to the triggering element. The “Menu” button expands to display a mega menu. On closing this menu by activating the “Close” button, the focus is returned to the triggering “Menu” button.



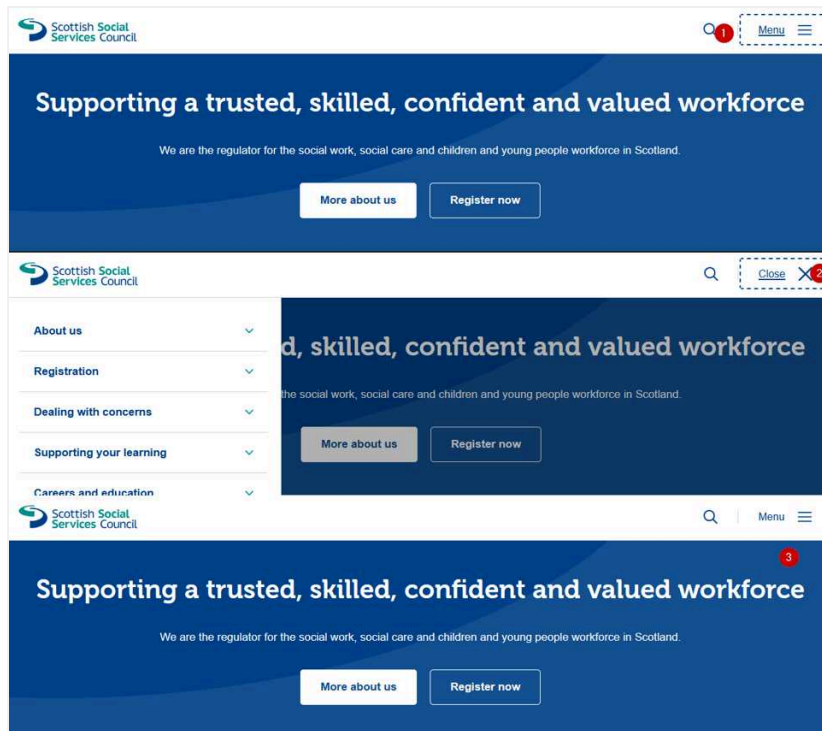
What happened

The keyboard focus order must be meaningful. It does not need to mirror the visual order. However, interactive elements must receive focus in a way that makes some sort of sense. The focus order must not be totally illogical or make the operability of components more challenging.

There are elements that control new content being opened and closed. When the new content is closed, the focus does not return to the element that triggered the content. This may particularly disadvantage keyboard users. The illogical focus order may make the page difficult to understand and interact with.

Where it happened

This can be seen on page 1 on desktop, with the “Menu” button. The “Menu” button expands to display a mega menu. On closing this menu by activating the “Close” button, focus is lost to the `<body>` element. That is instead of being returned to the triggering “Menu” button.



```
>> document.activeElement  
< > <body class="home wp-singular page-te...-child-theme-chop-child"> 🗑
```

Steps to reproduce

1. Using just your keyboard, tab onto the above elements and activate them.
2. Observe that new content can be added and removed from the page.
3. Using just your keyboard, close or dismiss the new content.
4. Observe that focus does not return to the element that triggered the content.

Recommendation

Make sure that focus returns to the element that triggered the content. This can be achieved with the `focus` method.

For examples, see [MDN - focus method](#) and [Dev - Managing Focus for Accessibility](#).

For more information, see [WCAG - G59 Placing the interactive elements in an order](#), [Yale University - Web Accessibility - Focus and Keyboard Operability](#), [MDN - Keyboard](#), [Keyboard Navigable JavaScript widgets](#), [Google Developer - DOM Order Matters](#), and [Using Tabindex](#).

[Return to priority table](#)

ADR_22749-30 **MEDIUM PRIORITY** **RESOLVED**

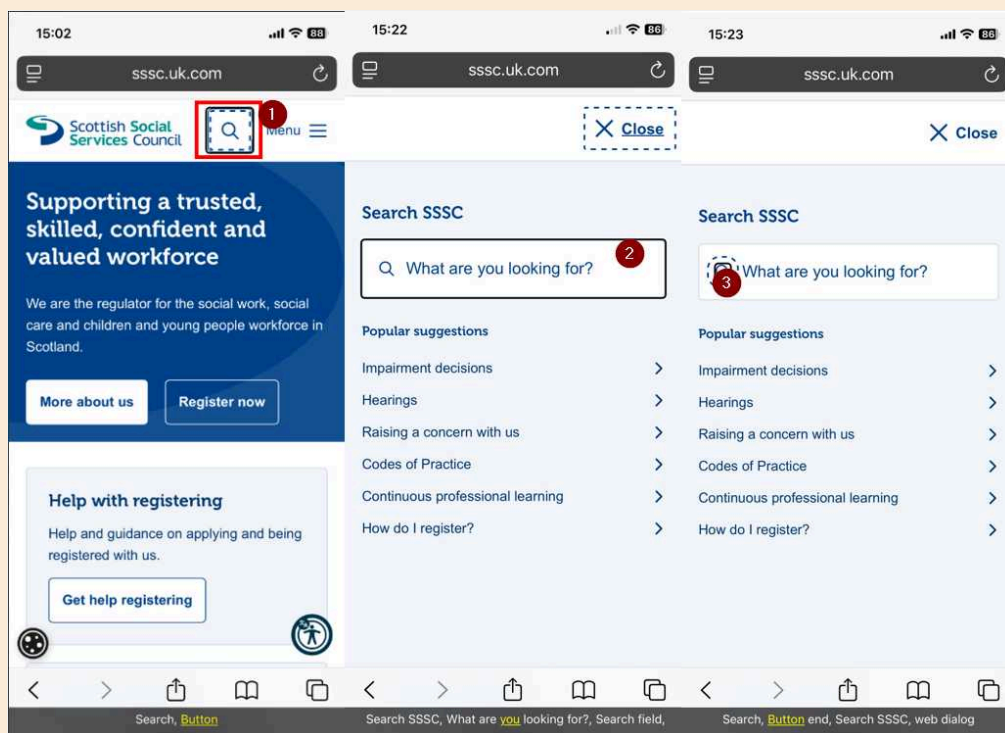
Focus not set onto modal dialog

Success Criteria	2.4.3: Focus Order [2.0 A]
Devices	None
Pages Affected	None
Users Affected	Motor
Component	Modal Dialog
ADR Type	Web Audit

Recheck comment 27/06/25

This issue is fixed.

On page 1 on iOS, when the search button is activated, the “Search SSSC” modal dialog is added to the page, including a search input and search suggestion links. The screen reader cursor is appropriately set to the search field. When swiping forward, the cursor goes to the search button followed by the suggestions.



What happened

The screen reader focus order must be meaningful. It does not need to mirror the visual order. However, interactive elements must receive the screen reader cursor in a way that makes some sort of sense. The focus order must not be totally illogical or make the operability of components more challenging.

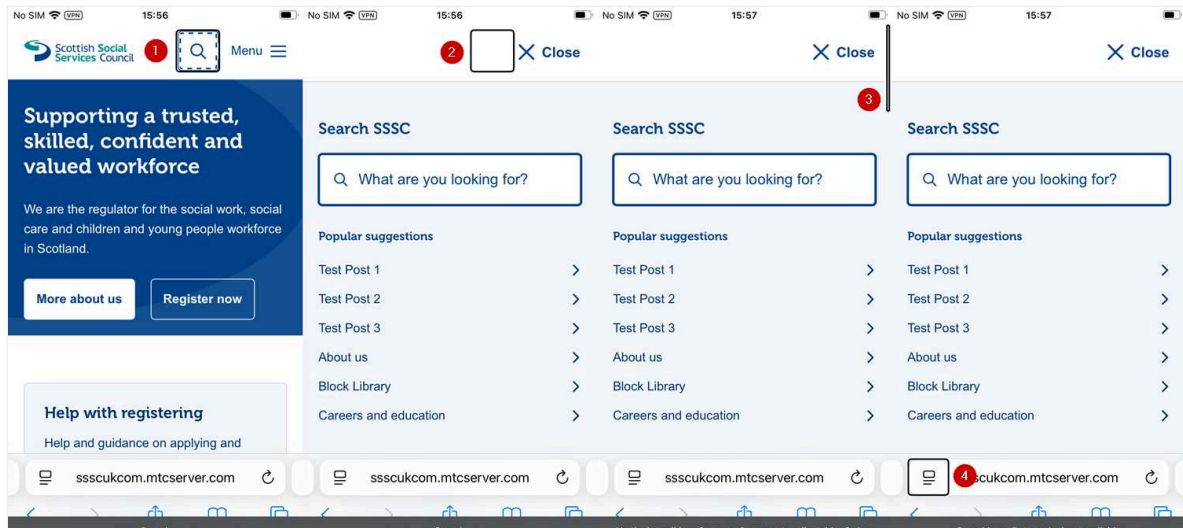
There are modal dialogs where the focus is not set onto the component when it is opened. Instead, the screen reader cursor remains in the background. This may particularly disadvantage keyboard users. The illogical focus order may make the modal dialog challenging to access and operate.

Where it happened

This can be seen on page 1 on iOS, with the “Search” button. When the button is activated, the “Search SSSC” modal dialog is added to the page, including a search input and search suggestion links.

However, the screen reader cursor is not managed appropriately. Instead, the screen reader cursor remains on the “search” button in the background. On the next swipe, the scrollbar receives the screen reader cursor, followed by the browser menu buttons.

Note that this behaviour does not occur the next time that the “Search” button is activated from the homepage, where the screen reader cursor is appropriately set to the search field.



Steps to reproduce

1. Using just your keyboard, activate the modal dialog.
2. Swipe to navigate around the page.
3. Observe that focus is not automatically set onto the dialog.

Recommendation

The screen reader focus order is determined by the location of focusable elements in the Document Object Model (DOM). Make sure that the interactive elements are placed in the DOM in a logical sequence that allows keyboard users to tab through the page in a meaningful order.

In this case, when the modal dialog is open, make sure that focus is set onto the dialog. This can be achieved with the `focus` method. As well, if you need the non-interactive elements to be capable of receiving focus through the `focus` method, use `tabindex="-1"`. For example:

```

<!-- background content -->
<div aria-hidden="true">
  <p>Example plain text in background</p>
  <button tabindex="-1">Example button in background</button>
  ...
</div>
<!-- modal dialog container -->
<div role="dialog" aria-labelledby="dialogName" aria-modal="true">
  <h2 id="dialogName" tabindex="-1">Modal Dialog Heading</h2>
  ...
</div>
    
```

For more information, see [WAI ARIA - Dialog Example](#), [MDN - focus method](#), [Yale University - Web Accessibility - Focus and Keyboard Operability](#), [MDN - Keyboard](#), [Keyboard Navigable JavaScript widgets](#), and [Google Developer - Using Tabindex](#).

[Return to priority table](#)

ADR_22749-38 **MEDIUM PRIORITY** **RESOLVED**

Non-interactive elements receive keyboard focus

Success Criteria 2.4.3: Focus Order [2.0 A]**Devices** None**Pages Affected** None**Users Affected** Motor**Component** Not Applicable**ADR Type** Web Audit

Recheck comment 27/06/25

This issue is fixed.

On page 3 on desktop, the non-interactive “...” element within the pagination component does not receive keyboard focus.



```
> <li class="page-numbers">...</li>
▼ <li class="page-numbers">
  <div class="page-link page-link--dots" data-type="ellipsis" index="5"> flex
    <span aria-hidden="true">...</span>
  </div>
</li>
▶ <li class="page-numbers">...</li>
▶ <li class="page-item page-item--next">...</li>
</ul>
</nav>
</section>
<!-->
```

What happened

The keyboard focus order must be meaningful. It does not need to mirror the visual order. However, interactive elements must receive focus in a way that makes some sort of sense. The focus order must not be totally illogical or make the operability of components more challenging.

There are elements that receive focus that are not interactive. This is illogical as only interactive elements should receive focus. This may particularly disadvantage keyboard users. The illogical focus order may make the page difficult to understand and interact with.

Where it happened

This can be seen on page 3, on desktop, with the non-interactive “...” element within the pagination component. The element has been incorrectly added to the focus order using `tabindex="0"`.

Showing results 1 to 10 of 100 rows

```

  <li class="page-numbers">...</li>
  <li class="page-numbers disabled page-numbers--dots">
    <a class="page-link" tabindex="0" flex>
      </a>
    </li>
  
```

Steps to reproduce

1. Using just your keyboard, tab through the above.
2. Observe that focus goes onto elements that are not interactive.

Recommendation

The focus order is determined by the location of focusable elements in the Document Object Model (DOM). Make sure that the interactive elements are placed in the DOM in a logical sequence that allows keyboard users to tab through the page in meaningful order.

In this case, make sure that the non-interactive elements do not receive keyboard focus. For example, make sure to avoid using `tabindex="0"` on elements that should not receive focus. If you need the non-interactive elements to be capable of receiving focus through the `focus` method, use `tabindex="-1"`. For example:

```
<h1 tabindex="-1">Main Heading</h1>
```

Using `tabindex="-1"` allows elements to be used as a target for the `focus` method without adding them to the focus order. For more information, see [Yale University - Web Accessibility - Focus and Keyboard Operability](#), [MDN - Keyboard](#) and [Google Developer - Using Tabindex](#).

[Return to priority table](#)

ADR_22749-45 **MEDIUM PRIORITY** **RESOLVED**

Focus is not managed appropriately when page is dynamically updated (Pages 3 and 9)

Success Criteria 2.4.3: Focus Order [2.0 A]

Devices None

Pages Affected None

Users Affected Motor

Component Button

ADR Type Web Audit

Recheck comment 27/06/25

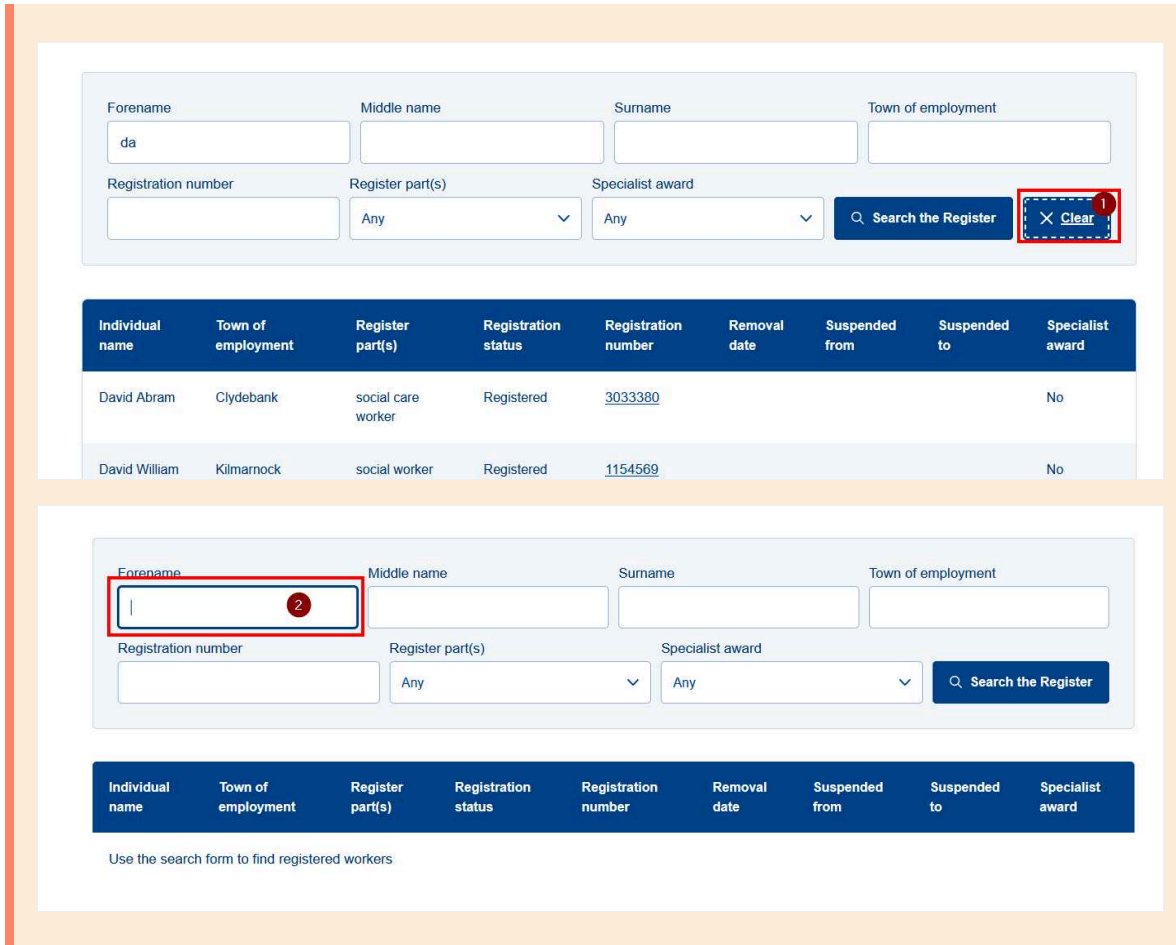
This issue is fixed.

Fixed: on pages 3 and 9, on desktop and iOS, focus is managed appropriately when the page is dynamically updated. On activating the pagination buttons, the search results are dynamically updated, and the page scrolls to the top of the updated results. The keyboard focus is set on the first interactive element within the newly updated content.



Individual name	Town of employment	Register part(s)	Registration status	Registration number	Removal date	Suspended from	Suspended to	Specialist award
Daisy Adeola	Edinburgh	social care worker	Registered	4098941				No
Damilola Yetunde Aderemi	Aberdeen	social care worker	Registered	4158871				No
David Oluseun Adesanya	Stirling	social care worker	Registered	4115715				No
David Oluwadare Adetoro	Luton	children and young people's worker	Registered	4124173				No
Damilare Andrew Adetunji	Glasgow, Manchester	social care worker	Registered	4162060				No

Fixed: on page 3, on desktop, when the "Clear" button is activated, the search results table is cleared, and the "Clear" button is removed from the page. The keyboard focus is set to the "Forename" input field, which is appropriate.



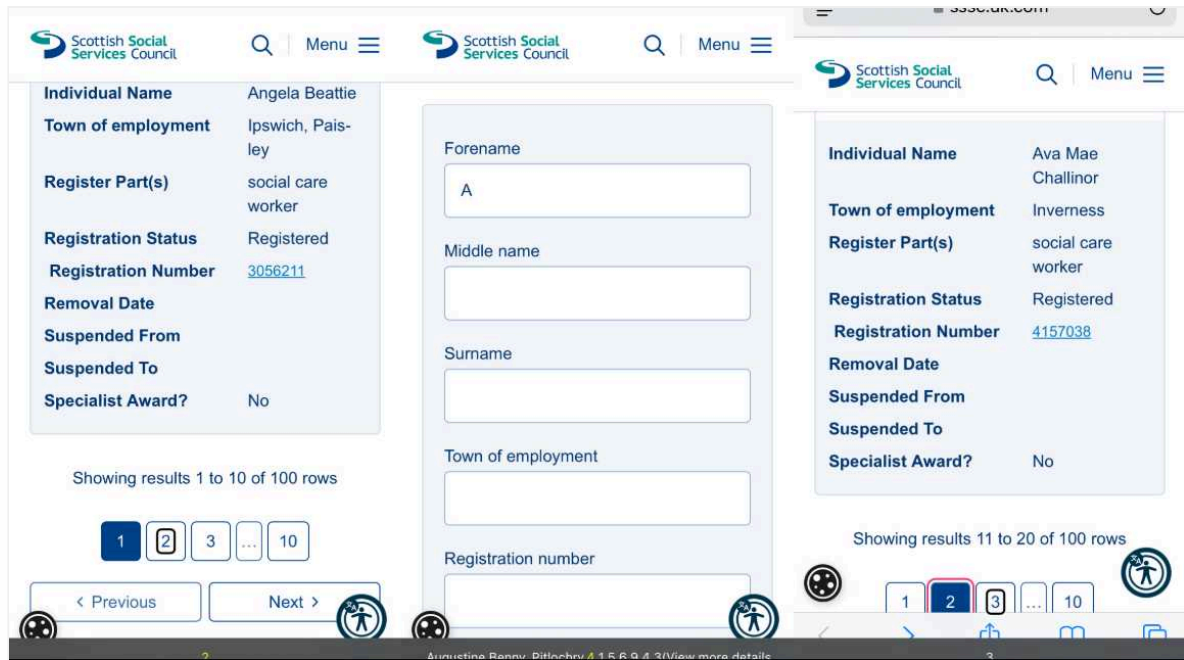
What happened

The keyboard focus order must be meaningful. It does not need to mirror the visual order. However, interactive elements must receive focus in a way that makes some sort of sense. The focus order must not be totally illogical or make the operability of components more challenging.

There are instances where the focus order is not managed when the page is dynamically updated. This may particularly disadvantage keyboard users who often navigate by tabbing through the interactive elements. The illogical focus order may make the page difficult to understand and interact with.

Where it happened

This can be seen on pages 3 and 9, on desktop and iOS, with the pagination buttons. On activation, the search results are dynamically updated and the page scrolls to the top of the updated results. However, focus remains on the activated pagination button and is not set to a logical element, such as the first search result within the newly updated content. When tabbing or swiping forward, focus is set to the next pagination button.



This can also be seen on page 3, on desktop, with the “Clear” button. On activation, the search results table is cleared and the “Clear” button is removed from the page. Focus is not set to a logical element. Instead, focus is set to the `<body>` element. When tabbing forward, focus is set to the “How do I register?” link.

Note that on iOS, the screen reader cursor is set to the adjacent “Use the search form to find registered workers” text which is appropriate.

Also in registration

[How do I register?](#)

[Managing my registration](#)

[Employers](#)

Steps to reproduce

1. Observe that the pages are dynamically updated.
2. Using just your keyboard, tab through the above.
3. Activate the dynamic functionality.
4. Continue to tab through the page.
5. Observe that the keyboard focus has not been appropriately managed.

Recommendation

The focus order is determined by the location of focusable elements in the Document Object Model (DOM). Make sure that the interactive elements are placed in the DOM in a logical sequence that allows keyboard users to tab through the page in meaningful order.

As well, make sure that focus is managed appropriately. Use the focus method sparingly when the page has a meaningful update. As well, use `tabindex="-1"` to allow elements to receive programmatic focus when they are not naturally in the focus order, such as headings.

When the entire page changes due to SPA functionality, make sure that the focus is managed. For example, when the user selected some “Continue” or “Next” button and the page is updated, set focus onto the heading of the brand new content.

```
<h2 tabindex="-1">SPA Page Heading</h2>
<p>When the below button is pressed, the the page updates with new content.</p>
<p>The focus method sets focus onto the new heading.</p>
<button>Next Page</button>
```

For examples, see [MDN - focus method](#) and [Dev - Managing Focus for Accessibility](#).

For more information, see [WCAG - G59 Placing the interactive elements in an order](#), [Yale University - Web Accessibility - Focus and Keyboard Operability](#), [MDN - Keyboard, Keyboard Navigable JavaScript widgets](#), [Google Developer - DOM Order Matters](#), and [Using Tabindex](#).

[Return to priority table](#)

ADR_22749-53 **MEDIUM PRIORITY** **RESOLVED**

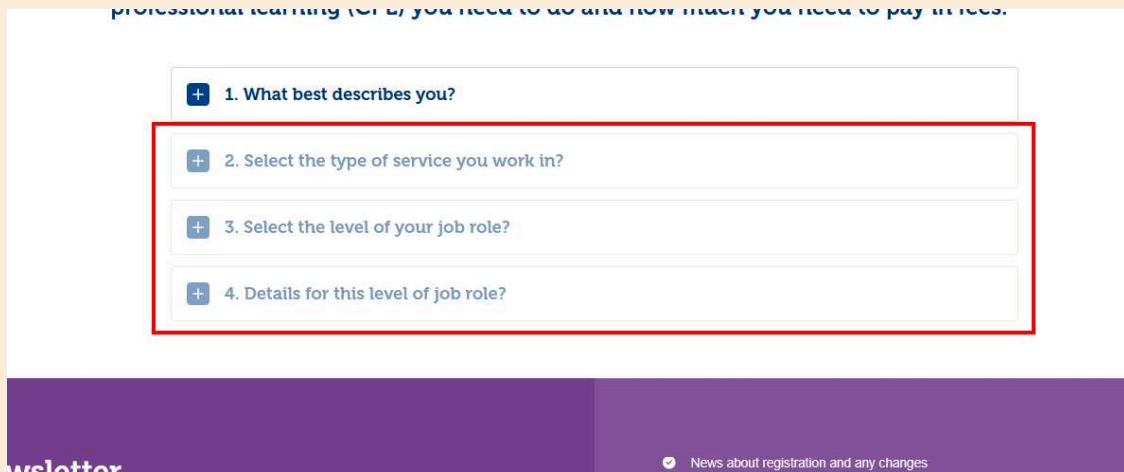
Inoperable interactive elements receive keyboard focus

Success Criteria	2.4.3: Focus Order [2.0 A]
Devices	None
Pages Affected	None
Users Affected	Motor
Component	Accordion or Disclosure
ADR Type	Web Audit

Recheck comment 27/06/25

This issue is fixed.

On page 8 on desktop, the disabled accordion buttons and the buttons within the accordion content are now removed from the keyboard focus when in their disabled state.



```

<div class="dynamic365form">
  <!-->
  <div class="accordion">
    <div class="accordion_wrapper">
      <details id="details--registerPart" name="help-with-register"> </details> (container)
      <details id="details--service" name="help-with-register" aria-disabled="true" tabindex="-1"> </details> (container)
      <details id="details--role" name="help-with-register" aria-disabled="true" tabindex="-1"> </details> (container)
      <details id="details--information" name="help-with-register" aria-disabled="true" tabindex="-1"> </details> (container)
    </div>
  </div>
</div>
</main>
  
```

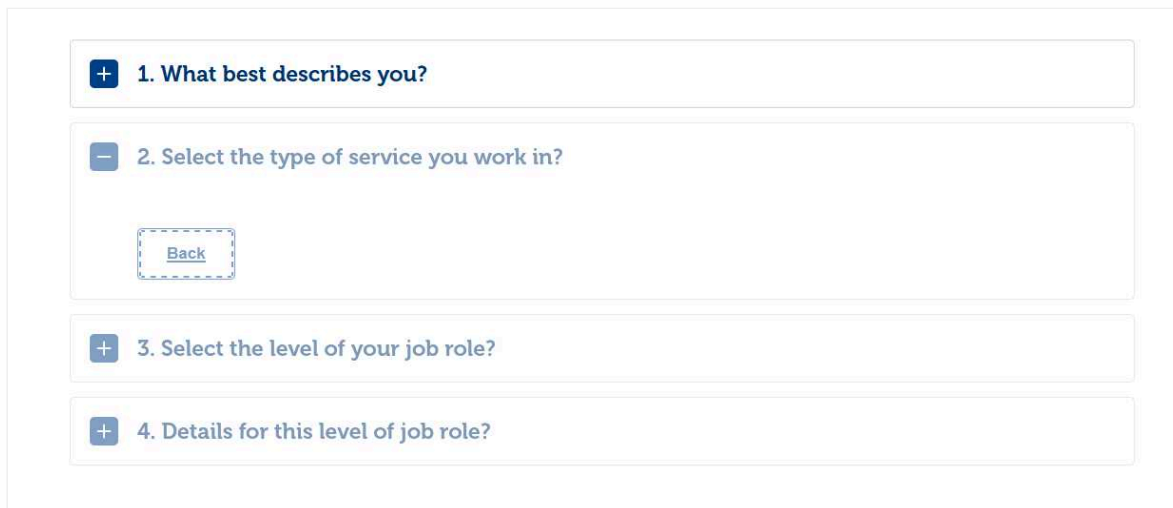
What happened

The keyboard focus order must be meaningful. It does not need to mirror the visual order. However, interactive elements must receive focus in a way that makes some sort of sense. The focus order must not be totally illogical or make the operability of components more challenging.

There are inoperable interactive elements that receive keyboard focus. This is illogical as only operable interactive elements should receive focus. This may particularly disadvantage keyboard users. The illogical focus order may make the page difficult to understand and interact with.

Where it happened

This can be seen on page 8, on desktop, with the disabled accordion buttons and the buttons within the accordion content. The disabled elements receive keyboard focus. Note that the disabled accordion buttons can be activated to expand them.



```

    <details name="help-with-register">...</details> container
    <details name="help-with-register" aria-disabled="true" open=""> container
      <summary>...</summary> flex
      <div class="details_inner">
        <div class="buttonGroup buttonGroup--stacked"> flex
          <!-->
          </div>
          <div class="buttonGroup buttonGroup--wrap"> flex
            <button class="mButton mButton--primary" type="button"> event flex
              <span class="mButton__text">Back</span>
            </button>
          </div>
        </div>
      </details>
    <details name="help-with-register" aria-disabled="true">...</details> container
    <details name="help-with-register" aria-disabled="true">...</details> container
  </div>
  
```

Steps to reproduce

1. Using just your keyboard, tab through the above.
2. Observe that focus goes onto elements that are not operable.

Recommendation

The focus order is determined by the location of focusable elements in the Document Object Model (DOM). Make sure that the interactive elements are placed in the DOM in a logical sequence that allows keyboard users to tab through the page in meaningful order.

In this case, make sure that the inoperable interactive elements do not receive keyboard focus.

Use `disabled`, or `aria-disabled` and `tabindex="-1"`. These techniques remove elements from the focus order and programmatically determine the elements as inoperable. Make sure that the attributes are used on the interactive elements themselves, rather than parent or child elements. For example:

```
<button disabled>This is disabled and excluded from the focus order</button>  
<button aria-disabled="true" tabindex="-1">This is disabled and excluded from the focus  
order</button>  
<span role="button" aria-disabled="true" tabindex="-1">This is disabled and excluded from  
the focus order</span>
```

When an element become operable, make sure that these attributes get removed or updated. For example:

```
<button>This is enabled and included from the focus order</button>  
<button aria-disabled="false">This is enabled and included from the focus order</button>  
<span role="button" aria-disabled="false" tabindex="0">This is enabled and included from  
the focus order</span>
```

For more information, see [MDN - disabled attribute](#) and [Google Developer - Using tabindex](#). As well, for guidance on reviewing accessibility properties, see [Chrome - Accessibility properties](#), [Edge - Accessibility Pane](#), and [Firefox - Accessibility Inspector](#).

[Return to priority table](#)

ADR_22749-54 **MEDIUM PRIORITY** **RESOLVED**

Focus is not managed appropriately when page is dynamically updated (Page 8)

Success Criteria	2.4.3: Focus Order [2.0 A]
Devices	None
Pages Affected	None
Users Affected	Motor
Component	Button
ADR Type	Web Audit

Recheck comment 27/06/25

This issue is fixed. **Fixed:** on page 8, on desktop, with the buttons within the “1. What best describes you?”, “2. Select the type of service you work in?”, and “3. Select the level of your job role?” accordions. On activation, the accordion is collapsed, and the next relevant accordion is expanded. Focus is set to the expanded accordion button.

1. What best describes you?

social care worker 1

Someone who works in social care such as care at home, housing support, care home and adult day care services, or inspects these types of services.

children and young people's worker

Someone who works with children and young people in day care of children, residential child care and residential school care accommodation services or inspects these types of services.

social worker

A social worker is someone who has a social work degree

social work student

A social work student is someone who has been accepted on a social work degree course

+ 1. What best describes you?

- 2. Select the type of service you work in? 2

care inspector

day care of children services

residential child care services

residential school care accommodation service

Back

+ 3. Select the level of your job role?

Fixed: on page 8 on desktop, with the "Back" button within the "2. Select the type of service you work in?" and "3. Select the level of your job role?" accordions. On activation, the accordion is collapsed, and the previous relevant accordion is expanded. Focus is then set on the expanded accordion button.

- 2. Select the type of service you work in?

- adult day care services
- care at home services
- care home service for adults
- care inspector
- housing support services

[Back](#)

+ 3. Select the level of your job role?

- 1. What best describes you? 2

- social care worker**
Someone who works in social care such as care at home, housing support, care home and adult day care services, or inspects these types of services.
- children and young people's worker**
Someone who works with children and young people in day care of children, residential child care and residential school care accommodation services or inspects these types of services.
- social worker**
A social worker is someone who has a social work degree
- social work student**

Fixed: on page 8 on desktop, with the "Back" and "Start again" buttons within the "4. Details for this level of job role?" accordion. On activation, the page scrolls down accordingly, and the focus is set to the relevant accordion button. When the "start again" button is activated, the focus is set to the first accordion button, and when the "back" button is activated, the focus is placed on the previous accordion.

+ **3. Select the level of your job role?**

- **4. Details for this level of job role?**

- **Qualification**

There are no qualifications for this register part

+ **Registration**

+ **Current fees**

+ **Future fees**

Back

Start again 1

- **1. What best describes you?** 2

social care worker

Someone who works in social care such as care at home, housing support, care home and adult day care services, or inspects these types of services.

children and young people's worker

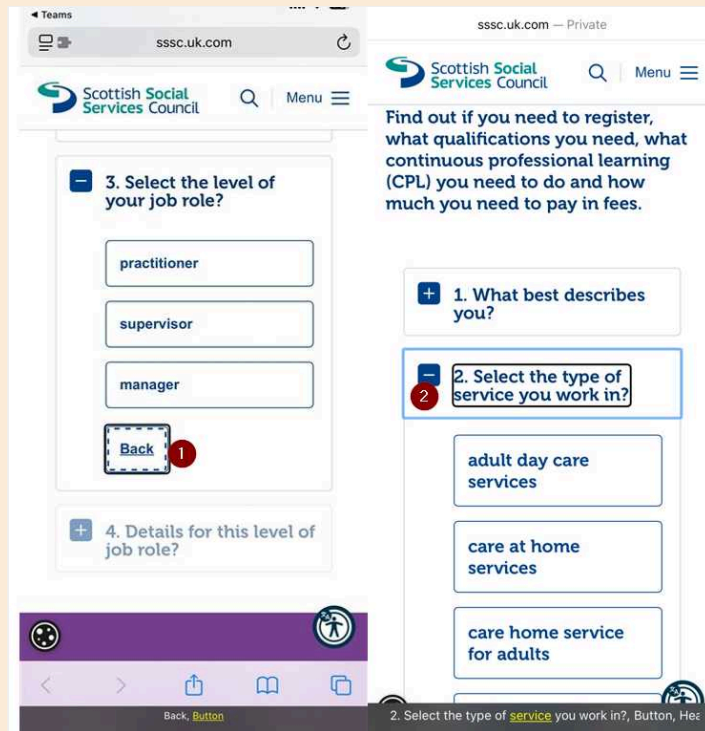
Someone who works with children and young people in day care of children, residential child care and residential school care accommodation services or inspects these types of services.

social worker

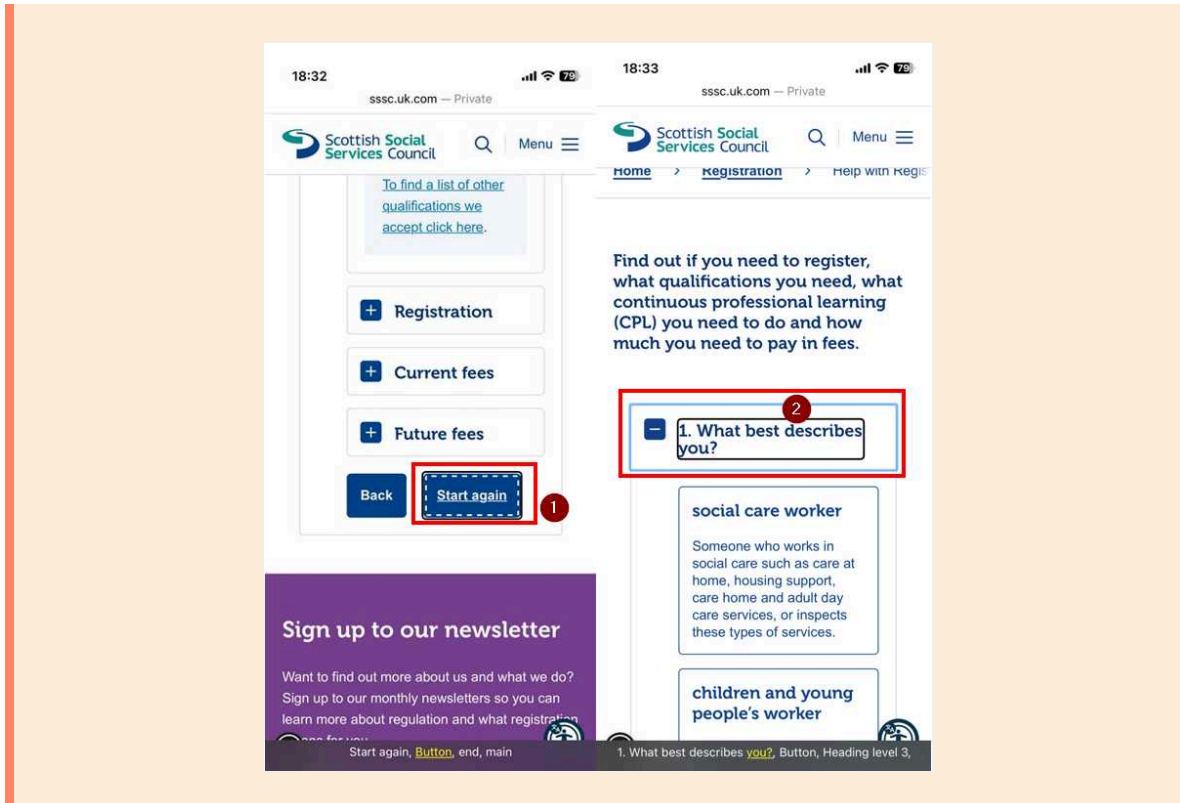
A social worker is someone who has a social work degree

social work student

Fixed: on page 8 on iOS, with the "Back" button within the "2. Select the type of service you work in?", "3. Select the level of your job role?" and "4. Details for this level of job role?" accordions. On activation, the accordion is collapsed, and the previous relevant accordion is expanded. Focus is then set to the expanded accordion button.



Fixed: on page 8 on iOS, with the “Start again” button within the “4. Details for this level of job role?” accordion. On activation, the accordion is collapsed, and the “1. What best describes you?” accordion is expanded. Focus is then set to the first accordion button.



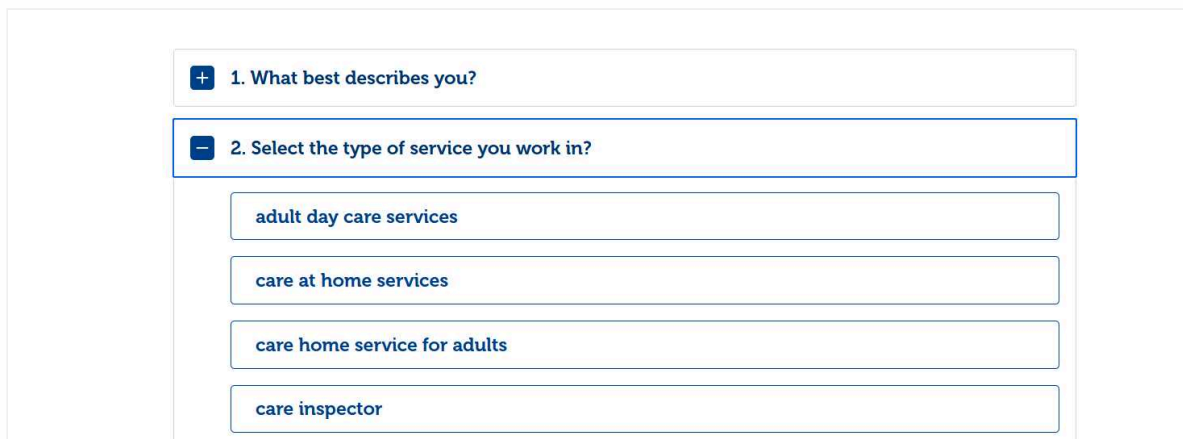
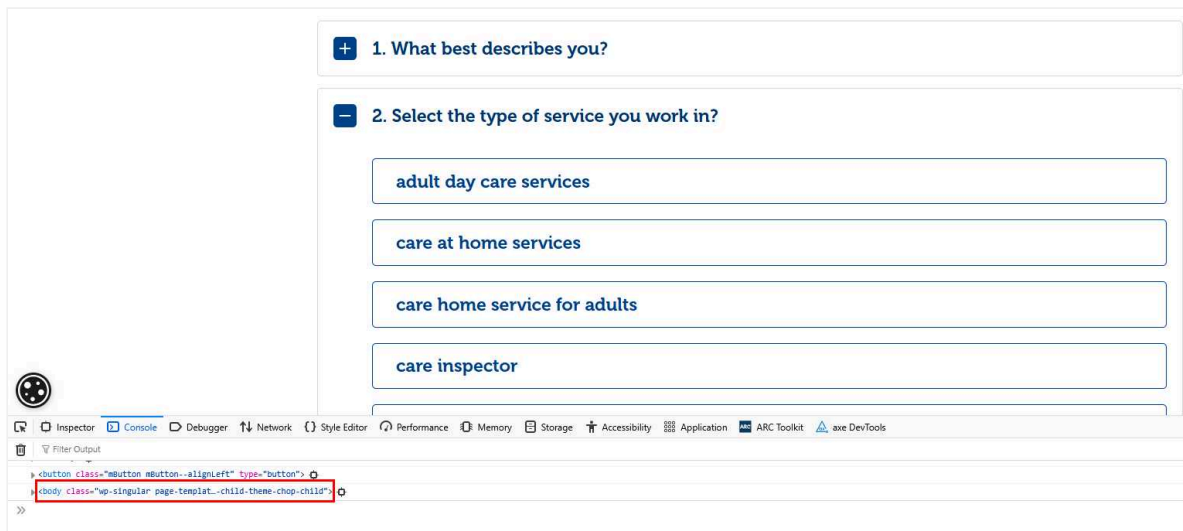
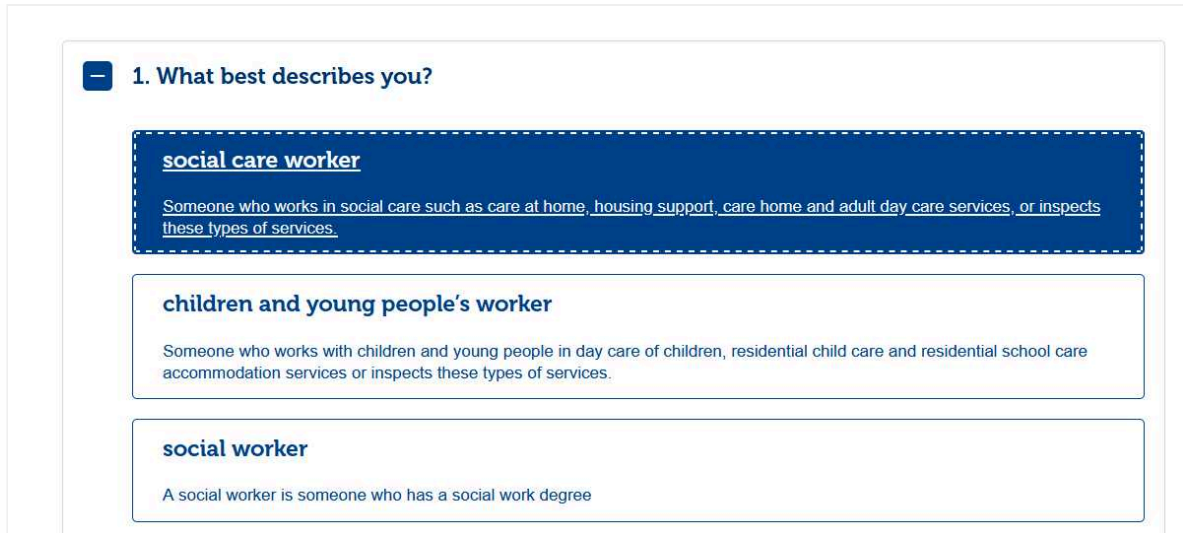
What happened

The keyboard focus order must be meaningful. It does not need to mirror the visual order. However, interactive elements must receive focus in a way that makes some sort of sense. The focus order must not be totally illogical or make the operability of components more challenging.

There are instances where the focus order is not managed when the page is dynamically updated. This may particularly disadvantage keyboard users who often navigate by tabbing through the interactive elements. The illogical focus order may make the page difficult to understand and interact with.

Where it happened

This can be seen on page 8, on desktop, with the buttons within the “1. What best describes you?”, “2. Select the type of service you work in?”, and “3. Select the level of your job role?” accordions. On activation, the accordion is collapsed and the next relevant accordion is expanded. Focus is not set to a logical element such as the accordion button. Instead, focus is set to the `<body>` element. Note that when tabbing forward, focus is then correctly set to the accordion button.



This can also be seen on page 8, on desktop, with the “Back” button within the “2. Select the type of service you work in?” and “3. Select the level of your job role?” accordions. On activation, the accordion is collapsed and the previous relevant accordion is expanded. Focus is not set to a logical element such as the accordion button. Instead, focus is set to the `<body>` element. Note that when tabbing forward, focus is then incorrectly set to the next disabled accordion button.

+ 2. Select the type of service you work in?

- 3. Select the level of your job role?

supervisor

support worker

practitioner

manager

Back

+ 4. Details for this level of job role?

- 2. Select the type of service you work in?

adult day care services

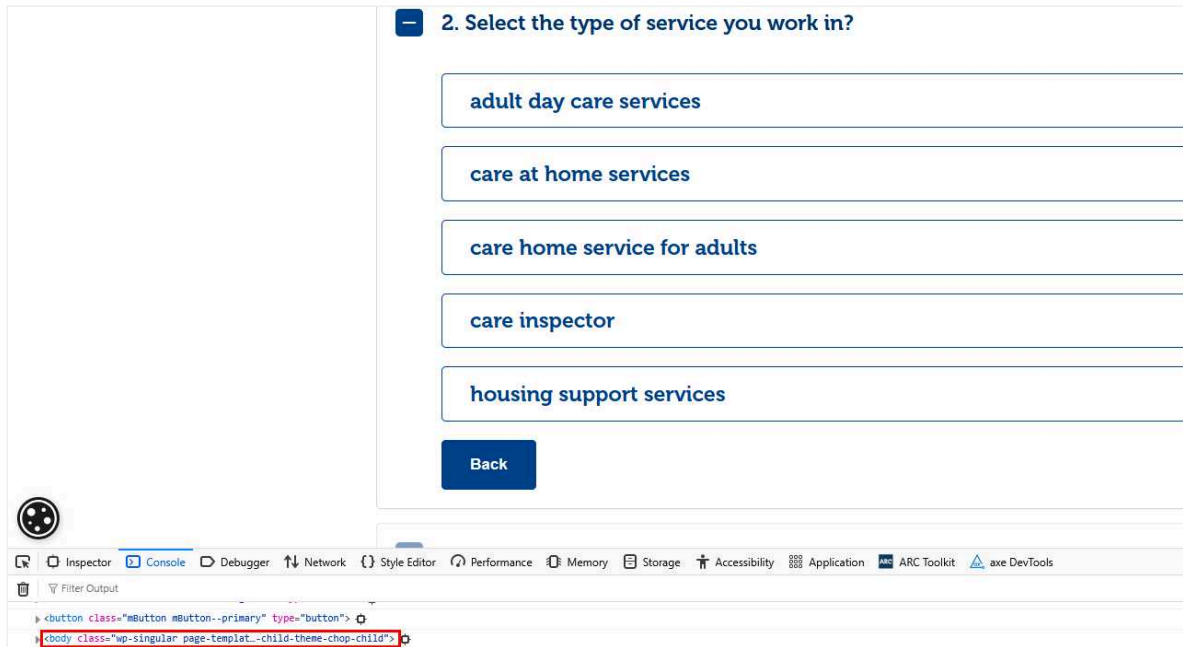
care at home services

care home service for adults

care inspector

housing support services

Back



The screenshot shows a web browser with a form. The form has a section titled "2. Select the type of service you work in?" with a minus sign icon. Below this are five text input fields containing "adult day care services", "care at home services", "care home service for adults", "care inspector", and "housing support services". A blue "Back" button is at the bottom of this section. Below the form is a developer console with a red box around the following code: `<body class="wp-singular page-templat...child-theme-chop-child">`

housing support services

Back

+ 3. Select the level of your job role?

+ 4. Details for this level of job role?

This can also be seen on page 8, on desktop, with the “Back” and “Start again” buttons within the “4. Details for this level of job role?” accordion. On activation, the page scrolls down and focus is not set to a logical element such as the relevant accordion button. Instead, focus is set to the `<body>` element. When tabbing forward, focus is set to the “Sign up” button in the footer.

+ Registration

+ Current fees

+ Future fees

Back Start again

Want to find out more about us and what we do? Sign up to our monthly newsletters so you can learn more about regulation and what registration means for you.

Sign up

- Learning and develop
- Fitness to practise
- Changes to legislation
- Consultations and eve

Navigation

- Home
- About us
- Registration
- Dealing with concerns
- Supporting your learning

Inspector Console Debugger Network Style Editor Performance Memory Storage Accessibility Application ARC Toolkit axe DevTools

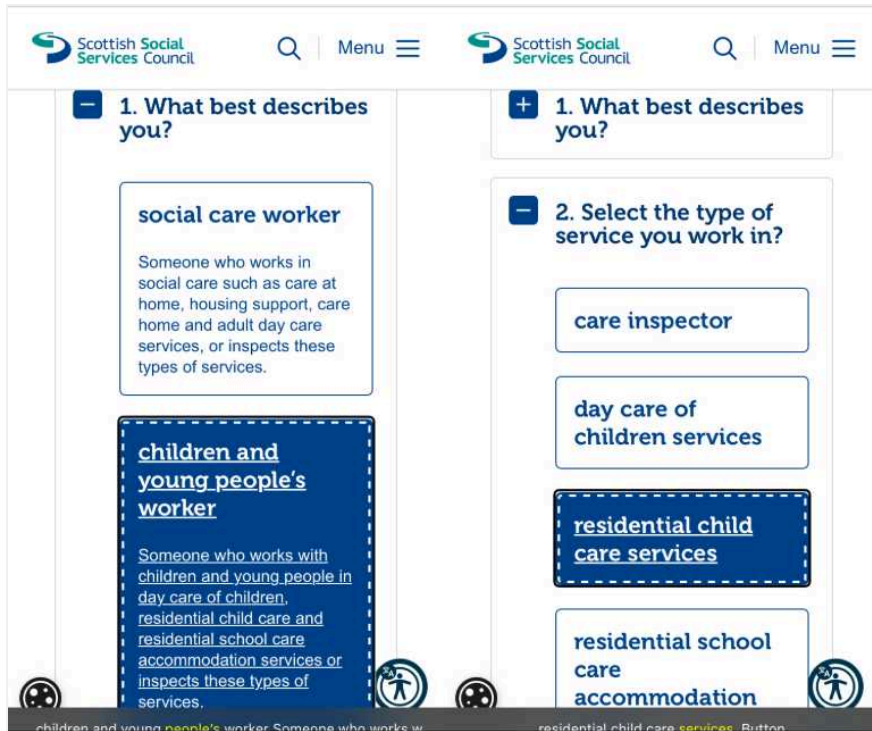
```
<button class="mButton mButton--primary" type="button">  
<body class="wp-singular page-templat--child-theme-chop-child">
```

Want to find out more about us and what we do? Sign up to our monthly newsletters so you can learn more about regulation and what registration means for you.

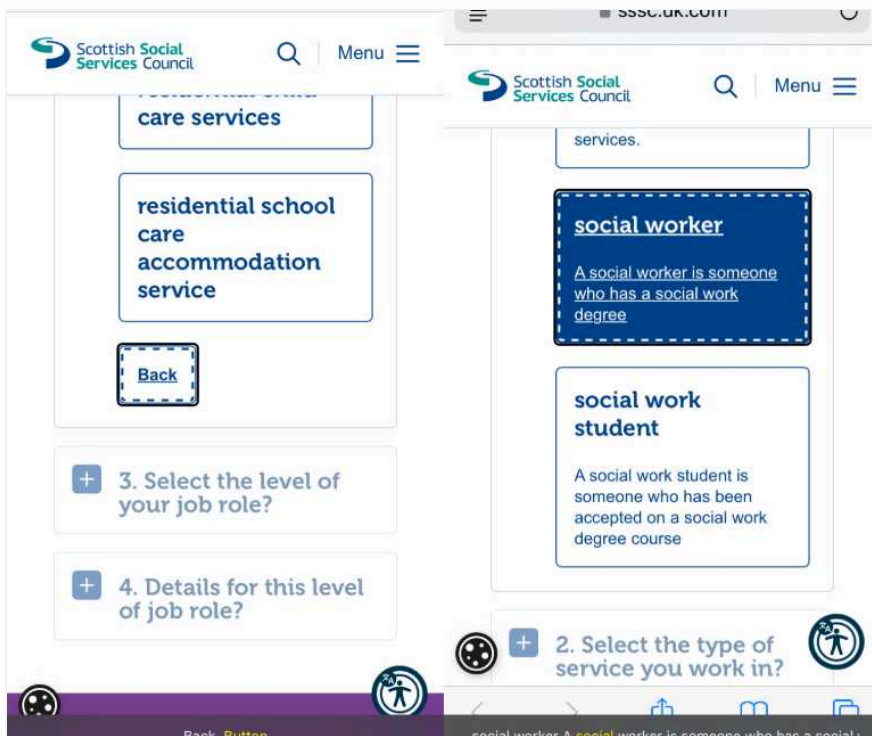
Sign up

- Learning and development news and updates
- Fitness to practise
- Changes to legislation
- Consultations and events

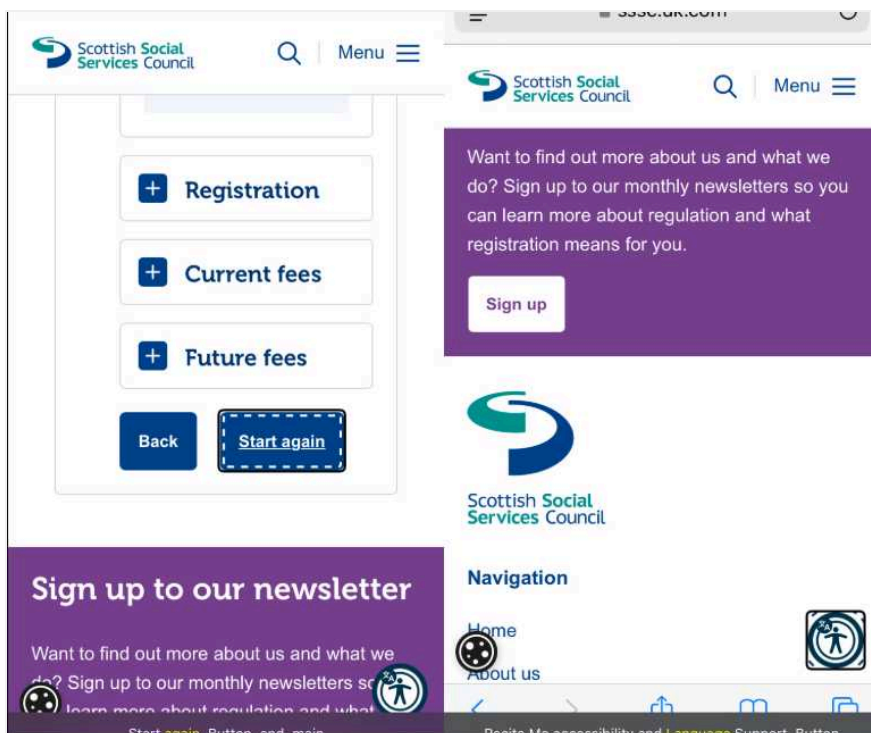
This can also be seen on page 8, on iOS, with the buttons within the “1. What best describes you?”, “2. Select the type of service you work in?”, and “3. Select the level of your job role?” accordions. On activation, the accordion is collapsed and the next relevant accordion is expanded. Focus is not set to a logical element such as the accordion button. Instead, focus is set to a random element within the next relevant accordion.



This can also be seen on page 8, on iOS, with the “Back” button within the “2. Select the type of service you work in?”; “3. Select the level of your job role?” and “4. Details for this level of job role?” accordions. On activation, the accordion is collapsed and the previous relevant accordion is expanded. Focus is not set to a logical element such as the accordion button. Instead, focus is set to a random element within the previous relevant accordion.



This can also be seen on page 8, on iOS, with the “Start again” button within the “4. Details for this level of job role?” accordion. On activation, the accordion is collapsed and the “1. What best describes you?” accordion is expanded. Focus is not set to a logical element such as the accordion button. Instead, focus is set to a random element on the page such as the “Recite Me” button.



Steps to reproduce

1. Observe that the pages are dynamically updated.
2. Using just your keyboard, tab through the above.
3. Activate the dynamic functionality.
4. Continue to tab through the page.
5. Observe that the keyboard focus has not been appropriately managed.

Recommendation

The focus order is determined by the location of focusable elements in the Document Object Model (DOM). Make sure that the interactive elements are placed in the DOM in a logical sequence that allows keyboard users to tab through the page in meaningful order.

As well, make sure that focus is managed appropriately. Use the focus method sparingly when the page has a meaningful update. As well, use `tabindex="-1"` to allow elements to receive programmatic focus when they are not naturally in the focus order, such as headings.

When the entire page changes due to SPA functionality, make sure that the focus is managed. For example, when the user selected some “Continue” or “Next” button and the page is updated, set focus onto the heading of the brand new content.

```
<h2 tabindex="-1">SPA Page Heading</h2>
<p>When the below button is pressed, the the page updates with new content.</p>
<p>The focus method sets focus onto the new heading.</p>
<button>Next Page</button>
```

For examples, see [MDN - focus method](#) and [Dev - Managing Focus for Accessibility](#).

For more information, see [WCAG - G59 Placing the interactive elements in an order](#), [Yale University - Web Accessibility - Focus and Keyboard Operability](#), [MDN - Keyboard](#), [Keyboard Navigable JavaScript widgets](#), [Google Developer - DOM Order Matters](#), and [Using Tabindex](#).

[Return to priority table](#)

ADR_22749-26 **MEDIUM PRIORITY** **RESOLVED**

Label insufficiently descriptive

Success Criteria 2.4.6: Headings and Labels [2.0 AA]

Devices None

Pages Affected None

Users Affected Vision

Component Link

ADR Type Web Audit

Recheck comment 27/06/25

This issue is fixed.

On page 3 on desktop and iOS, the links under the “Also in registration” heading are appropriately labeled. Each of these links goes to the following three pages: “How do I register?”, “Managing my registration” and the “Employers”. The labels of the links describe the related action.



```

<ul class="featuredCTA_list"> (grid)
  <li class="featuredCTA_listItem">
    <a class="featuredCTA_item ctaItem" href="/registration/how-do-i-register/"> (flex)
      <span class="ctaItem_heading heading heading-h3">How do I register?</span>
      <i class="mSvgIcon mSvgIcon--chevron-right-regular" aria-hidden="true"> (flex)
    </a>
  </li>
  <li class="featuredCTA_listItem">
    <a class="featuredCTA_item ctaItem" href="/registration/managing-my-registration/"> (flex)
      <span class="ctaItem_heading heading heading-h3">Managing my registration</span>
      <i class="mSvgIcon mSvgIcon--chevron-right-regular" aria-hidden="true"> (flex)
    </a>
  </li>
  <li class="featuredCTA_listItem">
    <a class="featuredCTA_item ctaItem" href="/registration/employers/"> (flex)
      <span class="ctaItem_heading heading heading-h3">Employers</span>
      <i class="mSvgIcon mSvgIcon--chevron-right-regular" aria-hidden="true"> (flex)
    </a>
  </li>
</ul>
</div>
</section>

```

What happened

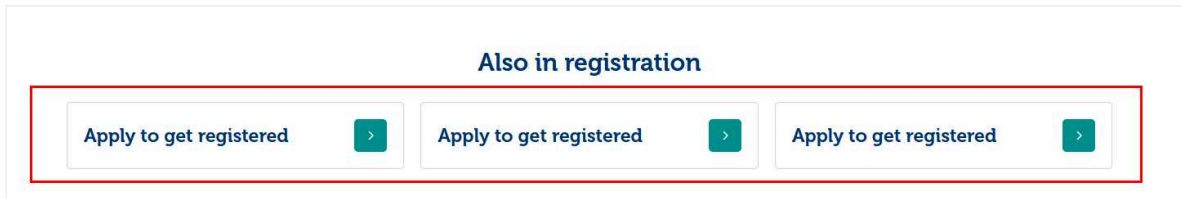
When headings and labels are used, they must describe the appropriate topic or purpose. In the context of this success criterion, the label is the visual signifier that communicates the nature of the form field. These visual signifiers must appropriately communicate the purpose of the form fields

There are labels that are insufficiently descriptive. This may particularly disadvantage users that find navigating the web cognitively challenging. Users may struggle to understand the purpose and operation of the relevant form fields. The insufficiently descriptive labels may confuse or misguide users.

Where it happened

This can be seen on page 3 on desktop and iOS, with the three “Apply to get registered” links under the “Also in registration” heading. Each of these links go to the following three pages “Impairment decisions”, “About us” and “Upcoming hearings”.

The “Apply to get registered” label is not descriptive of the related action.



```

<section class="featuredCTA featuredCTA--small" style="--featuredCta--background-color: var(--white-bg-color); --featuredCta--column-count: 3;">
  <h2 class="featuredCTA_heading heading heading--h2" data-element-id="headingsMap-12-0">Also in registration</h2>
  <div class="featuredCTA_wrapper">
    <a class="featuredCTA_item ctaItem" href="https://www.ssscukcom.mtcserver.com/impairment-decisions/" rel="noreferrer noopener nofollow">
      <strong class="ctaItem_heading heading heading--h3">Apply to get registered</strong>
      <i class="mSvgIcon mSvgIcon--chevron-right-regular" aria-hidden="true"></i>
    </a>
    <a class="featuredCTA_item ctaItem" href="https://www.ssscukcom.mtcserver.com/about-us/" rel="noreferrer noopener nofollow">
      <strong class="ctaItem_heading heading heading--h3">Apply to get registered</strong>
      <i class="mSvgIcon mSvgIcon--chevron-right-regular" aria-hidden="true"></i>
    </a>
    <a class="featuredCTA_item ctaItem" href="https://www.ssscukcom.mtcserver.com/hearings/" rel="noreferrer noopener nofollow">
      <strong class="ctaItem_heading heading heading--h3">Apply to get registered</strong>
      <i class="mSvgIcon mSvgIcon--chevron-right-regular" aria-hidden="true"></i>
    </a>
  </div>
</section>
    
```

Steps to reproduce

1. Observe the functionality associated with the labels.
2. Observe what the label communicates.
3. Observe that the label does not sufficiently describe the functionality.

Recommendation

Use clear and descriptive labels.

This success criterion does not necessarily mean the `<label>` element or the accessible name of the element. The requirement that labels are associated with form fields is covered by 1.3.1: Info and Relationships. The requirement that interactive elements have an accessible name is covered by 4.1.2 Name, Role, Value.

In the context of this success criterion, the label is the visual signifier that communicates the nature of the form field. If and when you use a visual signifier, make sure that it appropriately communicates the purpose of the form fields. Make sure that it is not vague and that it does not miscommunicate information.

For more information, see [WCAG - G130 Providing descriptive headings](#), [WAI - Writing for Accessibility](#), and [WAI Tutorials - Page Structure](#).

[Return to priority table](#)

ADR_22749-12 **MEDIUM PRIORITY** **RESOLVED**

Interactive elements have no visible focus indicator

Success Criteria	2.4.7: Focus Visible [2.0 AA]
Devices	None
Pages Affected	None
Users Affected	Vision, Motor
Component	Link, Breadcrumb
ADR Type	Web Audit

Recheck comment 26/06/25

This issue is fixed.

On page 2 on desktop, with the “Quick links”, such as “Overview”. These elements have a blue focus indicator when they receive keyboard focus. The current page link indicator has been removed from the current page links.

The screenshot shows a web page titled "Careers and education" with a "Quick links" section. The "Overview" link is highlighted with a blue focus indicator. Below the screenshot, a code editor displays the HTML structure of the quick links navigation and the CSS for the focus-visible state.

```

<div class="quickLinks_wrapper">
  ::before
  <nav aria-labelledby="heading--quicklinks-685d3bfe23d15" flex>
    <h2 id="heading--quicklinks-685d3bfe23d15" class="heading heading--h4">
      <ul data-quick-links-nav="" flex>
        <li>
          <a href="#overview">
            Overview
            :after
          </a>
        </li>
        <li>
          </li>
        <li>
          </li>
        <li>
          </li>
      </ul>
    </nav>
  ::after
  </div>
  quickLinks ul a:not(.noButton):focus-visible {
    text-decoration: underline;
    text-decoration-style: solid;
    text-decoration-thickness: auto;
    text-decoration-style: dashed;
    text-decoration-thickness: 2px;
    text-decoration-offset: 4px;
  }
  quickLinks ul a:not(.noButton):focus-visible, quickLinks ul a:not(.noButton):hover, quickLinks ul a:not(.noButton)[aria-current="page"] {
    font-weight: 700;
  }
  quickLinks ul a {
    color: var(--secondary-one-color);
    display: block;
    padding: calc(var(--margin)*1.15) calc(var(--margin)*1.14);
    position: relative;
  }
    
```

What happened

When they receive keyboard focus, interactive elements must provide a visual cue.

There are interactive elements that receive focus but which do not provide any visual cue. There is no visual difference between when the element is not receiving keyboard focus and when the element is receiving keyboard focus. Keyboard users may struggle to track their position on the page.

ADR_22749-14 **MEDIUM PRIORITY** **RESOLVED**

Interactive elements have no visible focus indicator (Recite Me)

Success Criteria	2.4.7: Focus Visible [2.0 AA]
Devices	None
Pages Affected	None
Users Affected	Vision, Motor
Component	Link
ADR Type	Web Audit

Recheck comment 27/06/25

This issue is fixed.

This can be seen on all pages on desktop with the “Recite Me” link. The “Recite Me” link that visually appears at the bottom of the page now has a visible focus indicator when navigated onto.

This is due to use of the CSS `outline: 2px dashed currentColor` property on the `<a>` tags, which displays the visible focus indicator.

The screenshot shows the SSSC website's navigation and contact information. The navigation menu includes links for Home, About us, Registration, Dealing with concerns, Supporting your learning, Careers and education, and Standards. The 'Useful' section lists links for Working at the SSSC, Access to information, Contact us, Accessibility statement, Procurement, Equality, and Data protection and privacy notice. The 'Contact Us' section provides the phone number 0345 60 30 891, operating hours from Monday to Friday 9am to 4pm, and a note that calls may be recorded. Social media icons for Facebook, X, Instagram, and LinkedIn are present, along with a circular icon containing a person and a speech bubble, which is highlighted with a red box. Below the website content, a browser developer tool is open, showing the HTML structure of the 'Recite Me' button and its CSS styles. The HTML shows a link with an 'event' attribute and an image alt text. The CSS shows a focus state with a dashed outline and a background color.

What happened

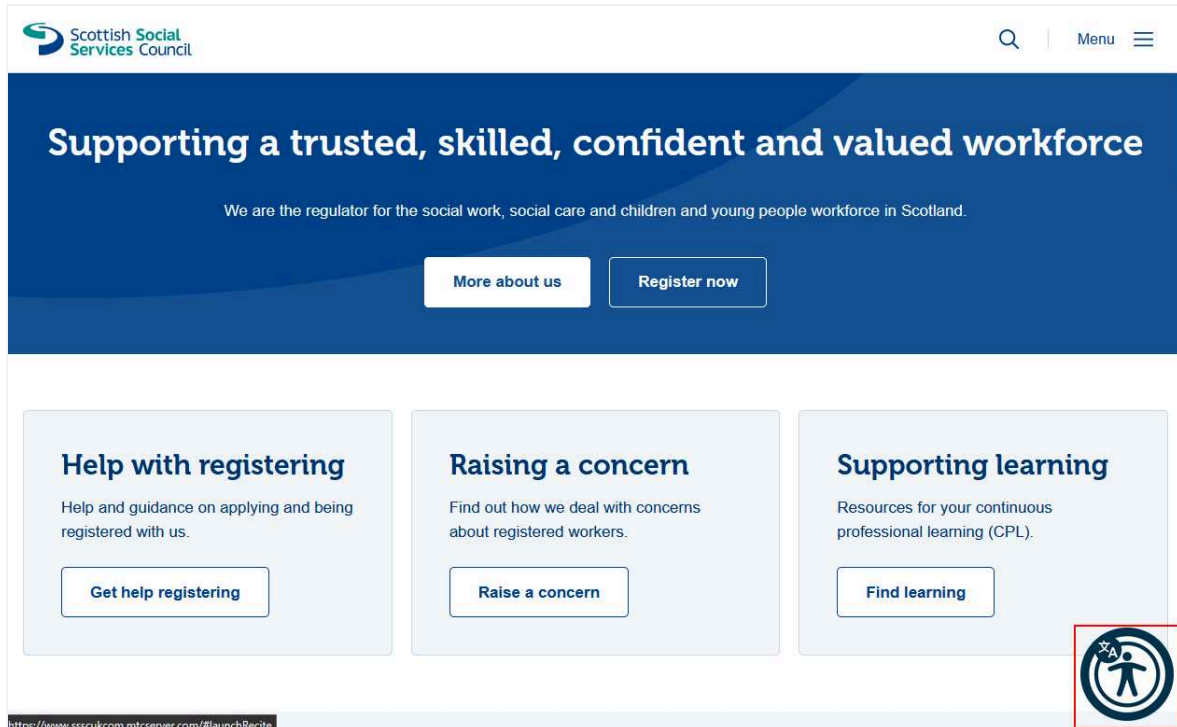
When they receive keyboard focus, interactive elements must provide a visual cue.

There are interactive elements that receive focus but which do not provide any visual cue. There is no visual difference between when the element is not receiving keyboard focus and when the element is receiving keyboard focus. Keyboard users may struggle to track their position on the page.

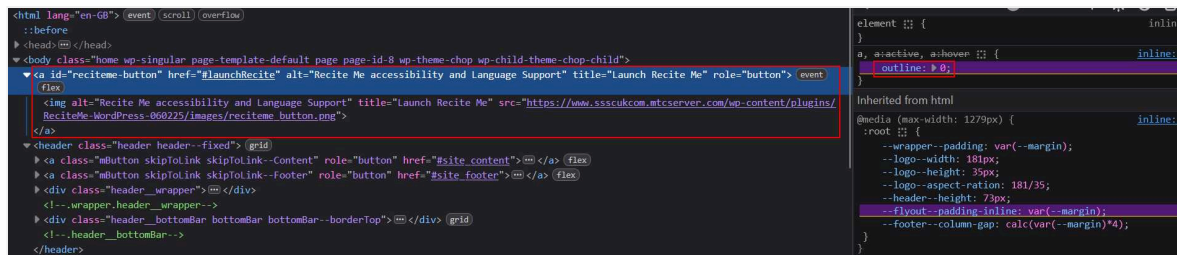
Where it happened

This can be seen on all pages on desktop with the “Recite Me” link. The “Recite Me” link visually appears at the bottom of the page. It is placed in the DOM at the top of the body, above the skip links. However, navigating the page using Tab, the interactive element does not have a visible focus indicator.

This is due to use of the CSS `outline: 0` property on the `<a>` tags, which removes the visible focus indicator.



https://www.ssscukcom.mtcserver.com/#launchRecite



Steps to reproduce

1. Use a keyboard to navigate through the above.
2. Observe that there is no visual cue when the elements receive keyboard focus.

Recommendation

Make sure that elements provide a visual cue when they receive keyboard focus. This can be achieved using the CSS **focus** pseudoclass and **outline** property. For more information, see [CSS Tricks - Focusing on Focus Styles](#), [GOV - Understanding focus state styles](#), and [Web Dev - Style Focus](#).

Another solution is making sure that default browser indication is supported. This is sometimes undone with CSS reset sheets. Do not overrule browser focus indication without implementing a custom focus indication. For example, do not use **outline: none**. For more information, see [a11y Project - Never Remove CSS Outlines](#).

[Return to priority table](#)

ADR_22749-49 **MEDIUM PRIORITY** **RESOLVED**

Accessible name does not include visual name

Success Criteria	2.5.3: Label in Name [2.1 A]
Devices	None
Pages Affected	None
Users Affected	Vision
Component	Input Field
ADR Type	Web Audit

Recheck comment 27/06/25

This issue is fixed.

On page 9 on desktop and iOS, the “Search SSSC” input field now has an accessible name of “Search SSSC”, provided by a visually hidden element.

The screenshot shows a search interface titled "Search SSSC". The search input field contains the text "Child". Below the input field, the HTML structure is shown, with the following elements highlighted:

```

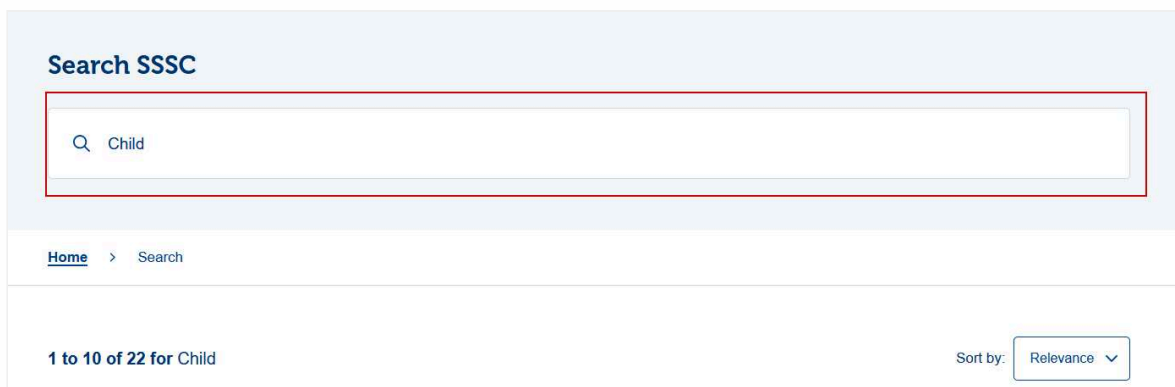
<label class="mTextField_label visuallyHidden" for="solr_site_search">Search SSSC</label>
<input id="solr_site_search" type="search" placeholder="What are you looking for?">
  
```

What happened

The accessible name of interactive elements must include the name that is visually communicated by on-screen text. This can otherwise particularly disadvantage voice recognition users, who use a say-what-you see interaction model. There are interactive elements where the accessible name does not include the visual name.

Where it happened

This can be seen on page 9 on desktop and iOS, with the “Search SSSC” input field. This input field has an accessible name of “What are you looking for?”, provided by a visually hidden `<label>` element. However, the “Search SSSC” visual label is not included in its accessible name.



```
<h1 class="subpageHero_heading heading heading--h2">Search SSSC</h1>
<search class="searchForm">
  <form action="/search" method="get" novalidate="novalidate">
    <div class="searchForm_input">
      <div class="row">
        <label class="mTextField_label visuallyHidden" for="solr_site_search">What are you looking for?</label>
        <div class="inputWrap">
          <input id="solr_site_search" type="search" placeholder="What are you looking for?">
          <button class="searchForm_submit"></button>
```

Steps to reproduce

1. Observe the visual name of the element communicated by on-screen text.
2. Inspect the element and review the accessible name.
3. Observe that the accessible name does not include the visual name.

Recommendation

Make sure that the accessible name includes the visual name. This is best achieved by creating the accessible name by referencing on-screen text. This can be done with `label` or `aria-labelledby`. Make sure that the `for` and `id` match correctly. For example:

```
<label for="firstName">First Name</label>
<input id="firstName" type="text" autocomplete="given-name">
```

Note that `aria-labelledby` is useful when you need to construct the label using multiple elements. For example:

```
<span id="firstPart">Last</span>
<span id="secondPart">Name</span>
<input aria-labelledby="firstPart secondPart" type="text" autocomplete="family-name">
```

If you cannot reference the visual name, make sure that you manually update the accessible name to include the visual name. Although not strictly required, it is best practice to have the visual name at the start of the accessible name and with the words in the same order.

For more information, see [WAI Tutorials - Labelling Controls](#), [WCAG - G208 Including the text of the visible label](#), and [G211 Matching the accessible name](#).

[Return to priority table](#)

ADR_22749-23 **MEDIUM PRIORITY** **RESOLVED**

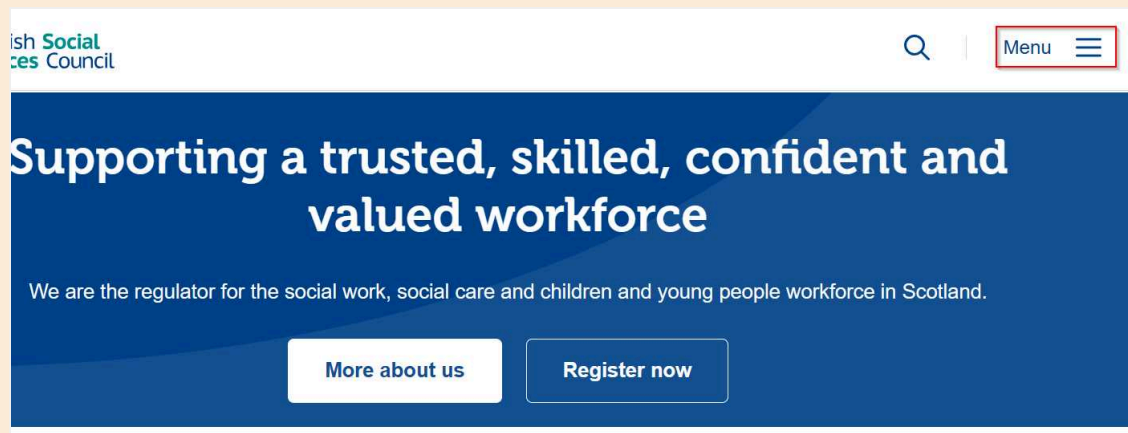
Expandable element is not programmatically identified

Success Criteria	4.1.2: Name, Role, Value [2.0 A]
Devices	None
Pages Affected	None
Users Affected	Vision
Component	Button
ADR Type	Web Audit

Recheck comment 26/06/25

This issue is fixed.

On desktop and iOS, the expandable nature of the “Menu” button has been programmatically determined.



```

<!--.header__meta.headerMeta-->
<button class="mButton mButton--textButton mButton--withIcon mainMenu_flyoutToggle" type="button" aria-expanded="false" aria-controls="mainMenu" data-main-menu-toggle="" data-search-form-close="" flex>
  <span class="mButton_text">Menu</span>
  <i class="mSvgIcon mSvgIcon--menu" aria-hidden="true"> flex
    <svg width="22" height="16" viewBox="0 0 22 16" fill="none" xmlns="http://www.w3.org/2000/svg">
      <path d="M1 1H21" stroke="#004289" stroke-width="2" stroke-linecap="round"></path>
      <path d="M1 8H21" stroke="#004289" stroke-width="2" stroke-linecap="round"></path>
      <path d="M1 15H21" stroke="#004289" stroke-width="2" stroke-linecap="round"></path>
    </svg>
  </i>
  <i class="mSvgIcon mSvgIcon--times" aria-hidden="true"> </i>
</button>
</div>
<!--.header__gridArea.header__gridArea--right-->
</div>
  
```

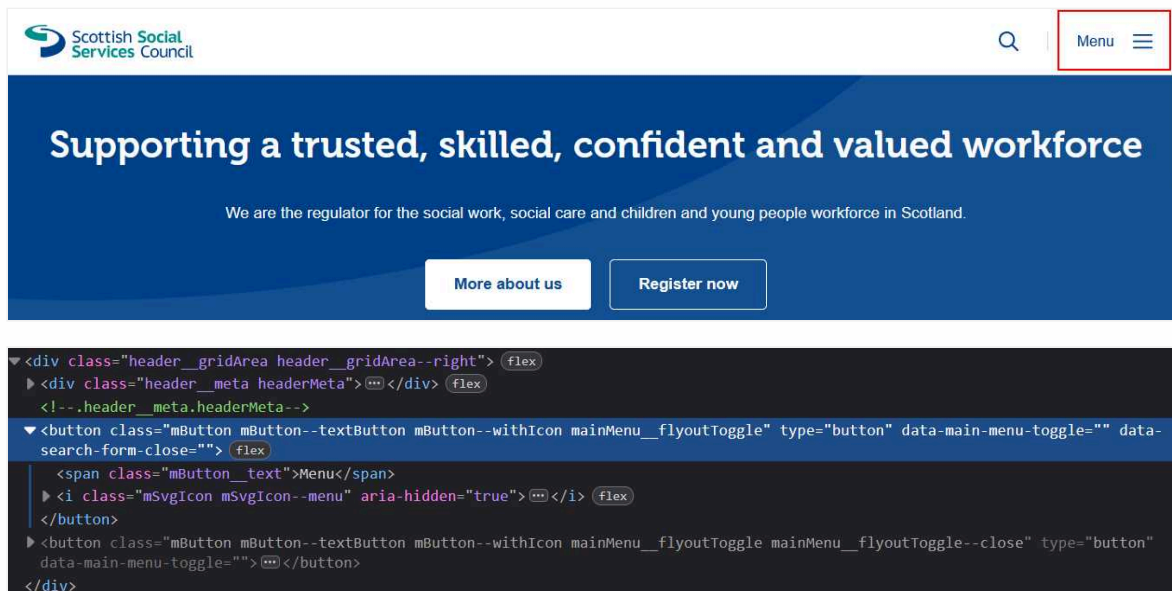
What happened

Interactive elements need a programmatically determined name, role, and value. These properties must also be accurate and appropriate.

There are expandable elements that are not programmatically determined as such. This may particularly disadvantage users that interface with the accessibility properties of elements, such as screen reader and voice recognition users. That the expandable state is not programmatically determined may confuse or mislead users.

Where it happened

This can be found on page 1 on desktop and iOS, with the “Menu” expandable button. Activating the button expands the mega menu. However, the expandable nature of the element has not been programmatically determined.



Steps to reproduce

1. Observe the expandable interactive elements.
2. Inspect the elements and review the DOM.
3. Review the computed accessibility properties.
4. Observe that the expandable state is not programmatically determined.
5. Observe that the state is not available through text.

Recommendation

Make sure that the expandable state is programmatically determined.

Use `aria-expanded` and update the `true` and `false` value appropriately. Make sure that the attribute is on the interactive elements themselves, rather than parent or child elements. For example:

```
<button aria-expanded="true">This is expanded</button>
<button aria-expanded="false">This is collapsed</button>
<span role="button" aria-expanded="true" tabindex="0">This is expanded</span>
<span role="button" aria-expanded="false" tabindex="0">This is collapsed</span>
```

For more information, see [WAI-ARIA Practices - Accordion](#), [WAI-ARIA - Disclosure](#), and [ADG - Marking elements expandable](#). As well, for guidance on reviewing accessibility properties, see [Chrome - Accessibility properties](#), [Edge - Accessibility Pane](#), and [Firefox - Accessibility Inspector](#).

[Return to priority table](#)

ADR_22749-32 MEDIUM PRIORITY RESOLVED

Interactive elements with no accessible name

Success Criteria 4.1.2: Name, Role, Value [2.0 A]

Devices None

Pages Affected None

Users Affected Vision

Component Button

ADR Type Web Audit

Recheck comment 27/06/25

This issue is fixed.

On page 1 on iOS, the search button has an accessible name. The `display: none` property previously used to hide the “search” text has been removed, and the search button has internal text from which to take its accessible name.

Showing results 1 to 10 of 100 rows

1

```

<li class="page-item page-item--previous disabled">...</li>
<li class="page-numbers active">
  <a class="page-link" tabindex="0">event flex
    1
  </a>
</li>

```

Steps to reproduce

1. Observe the elements visually highlighted are currently relevant.
2. Inspect the elements and review the DOM.
3. Review the computed accessibility properties.
4. Observe that the current state is not programmatically determined.
5. Observe that the state is not available through text.

Recommendation

Make sure that the current state is programmatically determined. Use `aria-current` or hidden text. The possible specific values are `page`, `step`, `location`, `date`, `time`. As well, `true` and `false` can be used as a catch-all value to determine whether an element is current or not. For example:

```

<ul>
  <li><a href="...">Home</a></li>
  <li><a href="..." class="highlighted" aria-current="page">Blog</a></li>
  <li><a href="...">Shop</a></li>
  <li><a href="...">Contact</a></li>
</ul>

```

Make sure that `aria-current` and `aria-selected` are not used interchangeably.

For example, `aria-current="date"` should be used for dates that are highlighted without the user interacting with them, such as today's date. When the user actively selects an element, it should use `aria-selected="true"`. The nuance is explained in [aria-current - examples and best practices](#).

For more information, see [ADG - Marking elements as the current one](#). As well, for guidance on reviewing accessibility properties, see [Chrome - Accessibility properties](#), [Edge - Accessibility Pane](#), and [Firefox - Accessibility Inspector](#).

[Return to priority table](#)

ADR_22749-42 **MEDIUM PRIORITY** **RESOLVED**

Disabled element is not programmatically identified

- Success Criteria** 4.1.2: Name, Role, Value [2.0 A]
- Devices** None
- Pages Affected** None
- Users Affected** Vision
- Component** Link
- ADR Type** Web Audit

Recheck comment 27/06/25

This issue is fixed.

On pages 3 and 9, on desktop and iOS, the disabled state of the “Previous” and “Next” pagination buttons when on the first or last search results page is now programmatically determined.



```
::before
<li class="page-item page-item--previous">
  <button class="page-link" aria-label="Previous Page" type="button" disabled="">event flex
    <i class="mSvgIcon mSvgIcon--mask mSvgIcon--chevron-left-regular" aria-hidden="true"></i>
    Previous
  </button>
</li>
<li class="page-numbers">...</li>
<li class="page-numbers">...</li>
<li class="page-numbers">...</li>
<li class="page-numbers">...</li>
<li class="page-numbers">...</li>
<li class="page-numbers">...</li>
<li class="page-numbers">...</li>
<li class="page-item page-item--next">
  <button class="page-link" aria-label="Next Page" type="button">event flex
    Next
    <i class="mSvgIcon mSvgIcon--mask mSvgIcon--chevron-right-regular" aria-hidden="true"></i>
  </button>
</li>
</ul>
</nav>
```

What happened

Interactive elements need a programmatically determined name, role, and value. These properties must also be accurate and appropriate.

There are disabled elements that are not programmatically determined as such. This may particularly disadvantage users that interface with the accessibility properties of elements, such as screen reader and voice recognition users. That the disabled state is not programmatically determined may confuse or mislead users.

Where it happened

This can be found on pages 3 and 9, on desktop and iOS, with the “Previous” and “Next” pagination links. When the links are disabled when on the first or last search results page, the disabled state of the link is not programmatically determined.



```

<li class="page-item page-item--previous disabled">
  <a class="page-link" tabindex="-1"> event flex
    ::before
      Previous
    </a>
</li>
  
```

Steps to reproduce

1. Observe the inoperable interactive elements.
2. Inspect the elements and review the DOM.
3. Review the computed accessibility properties.
4. Observe that the disabled state is not programmatically determined.

Recommendation

Make sure that any disabled elements have their disabled state programmatically determined, using either the `disabled` or `aria-disabled` attribute as appropriate. It is best practise to ensure that disabled elements are removed from the focus order, e.g., using `tabindex=-1`, but this is not a requirement.

Make sure that the attributes are used on the interactive elements themselves, rather than parent or child elements. For example:

```

<button disabled>This is disabled and excluded from the focus order</button>
<button aria-disabled="true" tabindex="-1">This is disabled and excluded from the focus order</button>
<span role="button" aria-disabled="true" tabindex="-1">This is disabled and excluded from the focus order</span>
  
```

When an element becomes operable, make sure that these attributes get removed or updated. For example:

```

<button>This is enabled and included in the focus order</button>
<button aria-disabled="false">This is enabled and included in the focus order</button>
<span role="button" aria-disabled="false" tabindex="0">This is enabled and included in the focus order</span>
  
```

Note that the `disabled` attribute can be used on native `<button>`, `<command>`, `<fieldset>`, `<keygen>`, `<optgroup>`, `<option>`, `<select>`, `<textarea>` and `<input>` elements. With any other elements, use the above `aria-disabled` and `tabindex="-1"` technique. For example:

```

<a href="..." aria-disabled="true" tabindex="-1">Disabled Link</a>
  
```

For more information, see [MDN - disabled attribute](#) and [Google Developer - Using tabindex](#). As well, for guidance on reviewing accessibility properties, see [Chrome - Accessibility properties](#), [Edge - Accessibility Pane](#), and [Firefox - Accessibility Inspector](#).

[Return to priority table](#)

--

ADR_22749-50 **MEDIUM PRIORITY** **RESOLVED**

Pressed element is not programmatically identified

Success Criteria 4.1.2: Name, Role, Value [2.0 A]

Devices None

Pages Affected None

Users Affected Vision

Component Button

ADR Type Web Audit

Recheck comment 27/06/25

This issue is fixed.

On page 8 on desktop and iOS, the pressed state of the buttons within the “1. What best describes you?”, “2. Select the type of service you work in?”, and “3. Select the level of your job role?” accordions is now programmatically determined.

1. What best describes you?

social care worker
Someone who works in social care such as care at home, housing support, care home and adult day care services, or inspects these types of services.

children and young people's worker
Someone who works with children and young people in day care of children, residential child care and residential school care accommodation services or inspects these types of services.

social worker
A social worker is someone who has a social work degree

social work student
A social work student is someone who has been accepted on a social work degree course

```

<li>
  <button class="mButton mButton--alignLeft mButton--primary" type="button" aria-pressed="true">event inline-flex
  <span class="mButton_text">
    <strong class="heading heading--h3">children and young people's worker</strong>
    <span></span>
  </span>
</button>
</li>
</li>
</li>
</ul>
</div>
/details>

```

What happened

Interactive elements need a programmatically determined name, role, and value. These properties must also be accurate and appropriate.

There are pressed elements that are not programmatically determined as such. This may particularly disadvantage users that interface with the accessibility properties of elements, such as screen reader and voice recognition users. That the pressed state is not programmatically determined may confuse or mislead users.

Where it happened

This can be found on page 8, on desktop and iOS, with the buttons within the “1. What best describes you?”, “2. Select the type of service you work in?”, and “3. Select the level of your job role?” accordions. Within each accordion, a single button can be pressed to select it. However, the pressed state is not programmatically determined.

-
1. What best describes you?

social care worker

Someone who works in social care such as care at home, housing support, care home and adult day care services, or inspects these types of services.

children and young people’s worker

Someone who works with children and young people in day care of children, residential child care and residential school care accommodation services or inspects these types of services.

social worker

A social worker is someone who has a social work degree

```

<button class="mButton mButton--alignLeft mButton--primary" type="button"> (event) (flex)
  <span class="mButton_text">
    <span class="heading heading--h3">social care worker</span>
    <span>
      Someone who works in social care such as care at home, housing support, care home and adult day care services, or inspects these types of services.
    </span>
  </span>
</button>

```

Steps to reproduce

1. Observe the pressable interactive elements.
2. Inspect the elements and review the DOM.
3. Review the computed accessibility properties.
4. Observe that the pressed state is not programmatically determined.
5. Observe that the state is not available through text.

Recommendation

Make sure that the pressed state is programmatically determined.

Use `aria-pressed` and update the `true` and `false` value appropriately. Make sure that the attribute is on the interactive elements themselves, rather than parent or child elements. For example:

```

<button aria-pressed="true">This is pressed</button>
<button aria-pressed="false">This is not pressed</button>
<span role="button" aria-pressed="true" tabindex="0">This is pressed</span>
<span role="button" aria-pressed="false" tabindex="0">This is not pressed</span>

```

For more information, see [WAI-ARIA Practices - Toggle Button](#), [WAI-ARIA - Disclosure](#), and [ADG - Marking elements activatable](#). As well, for guidance on reviewing accessibility properties, see [Chrome - Accessibility properties](#), [Edge - Accessibility Pane](#), and [Firefox - Accessibility Inspector](#).

[Return to priority table](#)

ADR_22749-43 **MEDIUM PRIORITY** **RESOLVED**

Loading message not programmatically determined

Success Criteria	4.1.3: Status Messages [2.1 AA]
Devices	None
Pages Affected	None
Users Affected	Vision
Component	Not Applicable
ADR Type	Web Audit

Recheck comment 27/06/25

This issue is fixed.

On page 3, when searching for results, a loading spinner dynamically appears on the page on desktop, whereas on iOS, no spinner is visible, but the page becomes dimmed and stays in a loading state for a considerable period. This loading status is programmatically determined by the "Loading, please wait..." message.

Other regulators

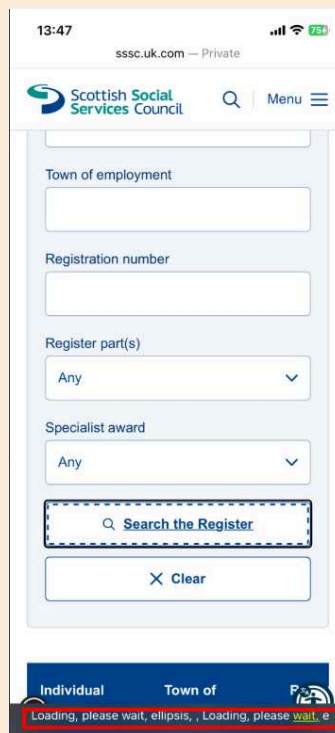
People in the social care and the children and young people workforce might register with another regulator, for example the Nursing and Midwifery Council or the General Teaching Council of Scotland, so they won't be on our Register. There are also other regulators covering the social care and children and young people workforce and other health professionals outside Scotland.

Forename: Middle name: Surname: Town of employment:

Registration number: Register part(s): Specialist award:

Individual name	Town of employment	Register part(s)	Registration status	Registration number	Removal date	Suspended from	Suspended to	Specialist award
-----------------	--------------------	------------------	---------------------	---------------------	--------------	----------------	--------------	------------------

Loading, please wait...



```

    <thead> </thead>
    <tbody> </tbody>
  </table>
</div>
</figcaption>
  <span id="dynamic365form_figcaption" class="dynamic365form_resultsText" role="status">Loading, please wait...</span>
</figcaption>
</figure>
<nav class="paginationNav" aria-labelledby="mtcPagination--174"> </nav>
</section>
<!-->
</div>
  
```

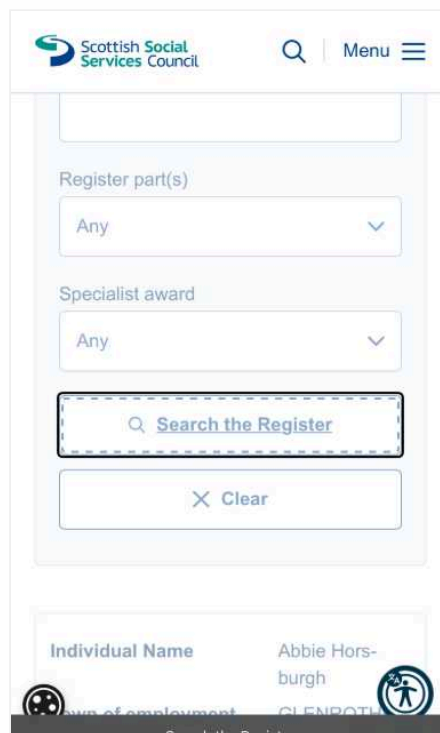
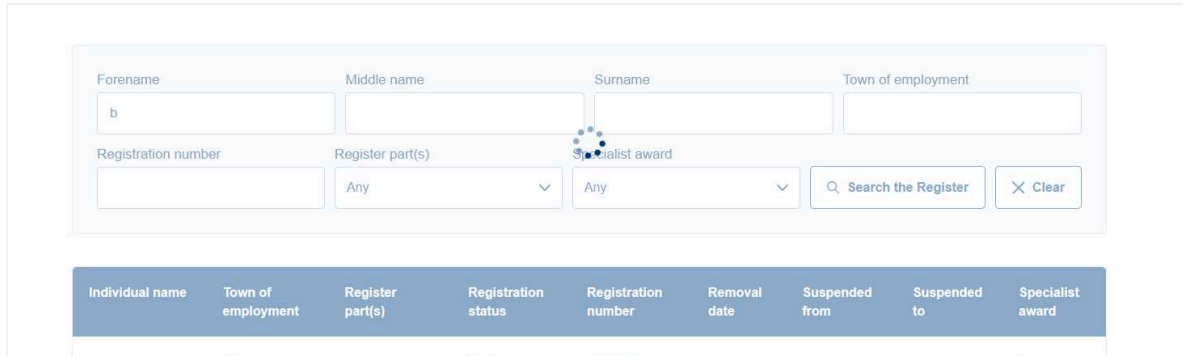
What happened

Dynamic updates need to be programmatically determined.

There are loading messages that are not programmatically determined as such. This may particularly disadvantage users that interface with the accessibility properties of elements, such as screen reader and voice recognition users. That the updates are not programmatically determined may confuse or mislead users.

Where it happened

This can be found on page 3, on desktop and iOS, with the loading state when searching for results. On desktop, a loading spinner dynamically appears on the page. On iOS, a loading spinner is not visible, but the page becomes dimmed and is in a loading state for a considerable period of time. The loading message is not programmatically determined.



```

<div class="loadingwrap loadingwrap--over"> {flex}
  <div class="loading">
    <svg class="svg-inline--fa fa-spinner icon" aria-hidden="true" focusable="false" data-prefix="fad" data-icon="spinner" role="img" xmlns="http://www.w3.org/2000/svg" viewBox="0 0 512 512">...</svg>
  </div>
</div>

```

Steps to reproduce

1. Observe the dynamic loading messages that are visually communicated.
2. Inspect the update and review the DOM.
3. Note that the loading messages are not programmatically determined.
4. Observe that focus is not moved and a new page is not loaded.
5. Observe that the update will not otherwise be immediately obvious to all users.

Recommendation

Make sure that the loading messages are programmatically determined.

For live regions to work, an empty container with live region semantics must be present on page load (before the region message is triggered). The live region message needs to be dynamically added to the container when triggered. For example, rather than toggling the visibility of the text using CSS. Also note, that you need to empty this container again, before dynamically adding a new message.

If the loading message visually communicates the progress, make sure that this is programmatically determined. As well, do the same when the loading message is completed. For example:

```
<span role="status"></span>
<span role="status">Loading, please wait</span>
<span role="status"></span>
<span role="status">Loading, 50% complete</span>
<span role="status"></span>
<span role="status">Loading complete</span>
```

You may wish to delay the status message. For example, if the loading is almost instantaneous, the status message may not be necessary. It is therefore acceptable to wait 3-5 seconds before triggering the status message, such that it only occurs when the loading is delayed or generally not instant.

Another solution is to move focus or prompt a page load. In such cases, you do not need to programmatically determine the status message as the change of context, such as focus moving to the added content or a new page being loaded, provides an acceptable call-to-action.

For more information, see [MDN - Using the status role](#) and [MDN - ARIA Live Regions](#).

[Return to priority table](#)

ADR_22749-44 **MEDIUM PRIORITY** **RESOLVED**

Search results not programmatically identified correctly

- Success Criteria** 4.1.3: Status Messages [2.1 AA]
- Devices** None
- Pages Affected** None
- Users Affected** Vision
- Component** Not Applicable
- ADR Type** Web Audit

Recheck comment 27/06/25

This issue is fixed.

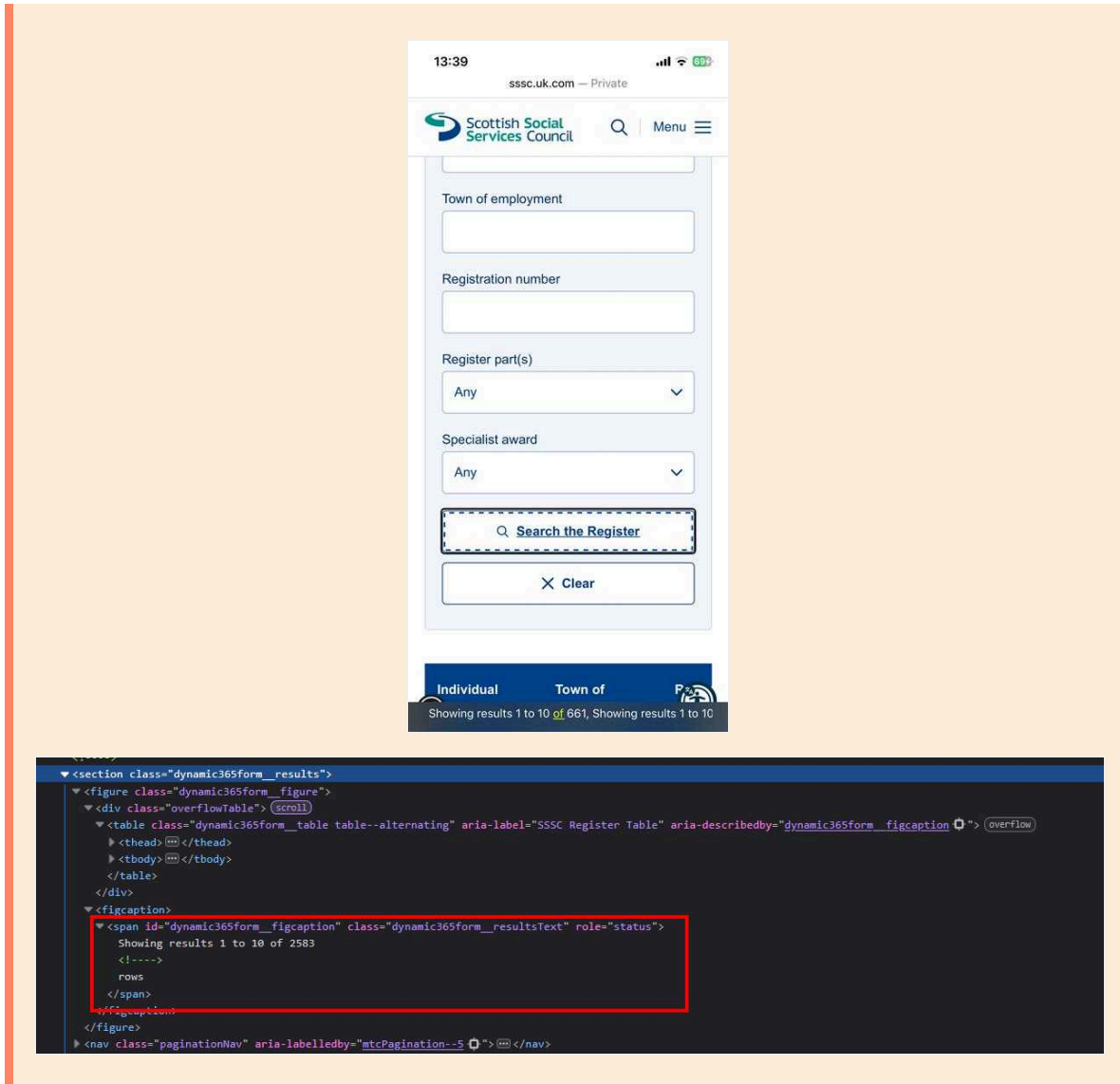
Fixed: on page 3 on desktop, the search result counts, such as “Showing results 21 to 30 of 3157 rows” are programmatically determined using `role="status"`.



```

<section class="dynamic365form_results">
  <figure class="dynamic365form_figure">
    <div class="overflowTable"> <div> scroll
      <figcaption>
        <span id="dynamic365form_figcaption" class="dynamic365form_resultsText" role="status">
          Showing results 21 to 30 of 3157
        </span>
      </figcaption>
    </figure>
    <nav class="paginationNav" aria-labelledby="mtcPagination--30">
  </section>
</div>
    
```

Fixed: on page 3 on iOS, the search result counts, such as “Showing results 1 to 10 of 661 rows” are programmatically determined using `role="status"`.



What happened

Dynamic updates need to be programmatically determined.

There are dynamic search results that are not programmatically correct. This may particularly disadvantage users that interface with the accessibility properties of elements, such as screen reader and voice recognition users. That the updates are not programmatically determined correctly may confuse or mislead users.

Where it happened

This can be found on page 3, on desktop, with the search results table when there are multiple results. The search result count, such as “Showing results 1 to 10 of 12 rows”, is associated with the table using `aria-describedby`. A live region has been defined on the table using `aria-live="polite"`. This results in all the search results being announced, followed by the search result count.

Abubakar Abdulkadir	GLASGOW, London, Glasgow	social care worker	Registered	4008396	No
Ahmed Nazir Abdullah	Glasgow	social care worker	Registered	4125701	No
Ayishat Abdulrahman	GLASGOW	social care worker	Registered	3119711	No

Showing results 1 to 10 of 12 rows

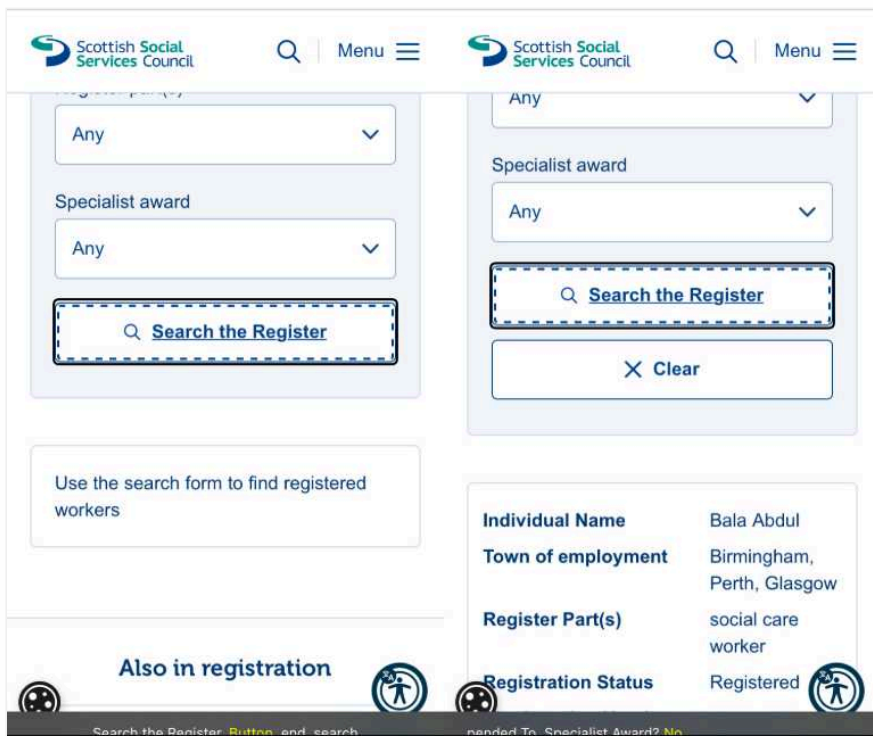
< Previous 1 2 Next >

```

<table class="dynamic365form_table table--alternating" aria-live="polite" aria-busy="false" aria-label="SSSC Register Table" aria-describedby="dynamic365form_figcaption">
  <tbody>
    <tr>
      <td>Abubakar Abdulkadir</td>
      <td>GLASGOW, London, Glasgow</td>
      <td>social care worker</td>
      <td>Registered</td>
      <td><a href="#">4008396</a></td>
      <td>No</td>
    </tr>
    <tr>
      <td>Ahmed Nazir Abdullah</td>
      <td>Glasgow</td>
      <td>social care worker</td>
      <td>Registered</td>
      <td><a href="#">4125701</a></td>
      <td>No</td>
    </tr>
    <tr>
      <td>Ayishat Abdulrahman</td>
      <td>GLASGOW</td>
      <td>social care worker</td>
      <td>Registered</td>
      <td><a href="#">3119711</a></td>
      <td>No</td>
    </tr>
  </tbody>
</table>
<figcaption id="dynamic365form_figcaption" class="dynamic365form_resultsText">Showing results 1 to 10 of 12 rows</figcaption>
</figure>

```

This can also be found on page 3, on iOS, with the search results table. The search result count, such as “Showing results 1 to 10 of 12 rows” and “No workers found in the register”, is associated with the table using `aria-describedby`. A live region has been defined on the table using `aria-live="polite"`. This results in the search results being announced. However, the search result count is not announced.



```

<table class="dynamic365form_table table--alternating" aria-live="polite" aria-busy="false" aria-label="SSSC Register Table" aria-describedby="dynamic365form_figcaption">
  <thead>
    <tr>
      <th></th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td></td>
      <td></td>
      <td></td>
      <td></td>
      <td></td>
      <td></td>
    </tr>
  </tbody>
</table>
<figcaption id="dynamic365form_figcaption" class="dynamic365form_resultsText">Showing results 1 to 1 of 1 rows</figcaption>
</figure>

```

Steps to reproduce

1. Observe the dynamic search results that are visually communicated.
2. Inspect the update and review the DOM.
3. Note that the dynamic search results are not programmatically determined correctly.
4. Observe that focus is not moved and a new page is not loaded.
5. Observe that the update will not otherwise be immediately obvious to all users.

Recommendation

Make sure that status messages are programmatically determined correctly. To achieve this, use a polite live region, such as `role="status"`, and `aria-atomic` to make sure that the entire text node is always announced.

For live regions to work, an empty container with live region semantics must be present on page load (before the region message is triggered). The live region message needs to be dynamically added to the container when triggered. For example, rather than toggling the visibility of the text using CSS. Also note, that you need to empty this container again, before dynamically adding a new message. For example:

```
<span role="status" aria-atomic="true"></span>  
<span role="status" aria-atomic="true">10 results found</span>  
<span role="status" aria-atomic="true"></span>  
<span role="status" aria-atomic="true">No results found</span>
```

Another solution is to move focus or prompt a page load. In such cases, you do not need to programmatically determine the status message as the change of context, such as focus moving to the added content or a new page being loaded, provides an acceptable call-to-action.

For more information, see [MDN - Using the status role](#) and [MDN - ARIA Live Regions](#).

[Return to priority table](#)

ADR_22749-55 **MEDIUM PRIORITY** **RESOLVED**

Search results not programmatically identified

Success Criteria	4.1.3: Status Messages [2.1 AA]
Devices	None
Pages Affected	None
Users Affected	Vision
Component	Not Applicable
ADR Type	Web Audit

Recheck comment 27/06/25

This issue is fixed.

On page 9 on desktop and iOS, the search result count is now programmatically identified using `role="status"`.

Search SSSC

Child

[Home](#) > Search

1 to 10 of 22 for Child

Residential child care workers
 ...Residential *child* care workers Home...
<https://www.sssc.uk.com/residential-child-care-workers/>

```

<section class="solrResults">
  <div class="solrResults_wrapper">
    <div class="solrResults_filter"> flex
    <div id="solrResults_resultsText" class="row solrResults_resultsText" role="status"> flex
      <b>1 to 10 of 22 for</b>
      <span>Child</span>
    </div>
    <div class="row"> ... </div> flex
  </div>
  <ul class="solrResults_list" aria-busy="false" aria-label="Search SSSC results" aria-describedby="solrResults_resultsText">
  </ul> grid
  <nav class="paginationNav" aria-labelledby="mtcPagination--4"> ... </nav>
</div>
  
```

What happened

Dynamic updates need to be programmatically determined.

There are dynamic search results that are not programmatically determined as such. This may particularly disadvantage users that interface with the accessibility properties of elements, such as screen reader and voice recognition users. That the updates are not programmatically determined may confuse or mislead users.

Where it happened

This can be found on page 9, on desktop and iOS, with the search results. When a search is completed, the results are dynamically updated. The search result count is not programmatically identified.

Note that a live region has been defined for the list of search results. However, this is not appropriate as it announces all the results and not the search results count.

Search SSSC

Q Child

[Home](#) > Search

1 to 10 of 22 for Child Sort by: Relevance ▾

[Residential child care workers](#)
 ...Residential *child* care workers Home...
<https://www.sssc.uk.com/residential-child-care-workers/>

```

<section class="solrResults">
  <div class="solrResults__wrapper">
    <div class="solrResults__filter"> flex
    <div id="solrResults_resultsText" class="row solrResults_resultsText"> flex
      <b>1 to 10 of 22 for</b>
      whitespace
      <span>child</span>
    </div>
  </div>

```

Steps to reproduce

1. Observe the dynamic search results that are visually communicated.
2. Inspect the update and review the DOM.
3. Note that the dynamic search results are not programmatically determined.
4. Observe that focus is not moved and a new page is not loaded.
5. Observe that the update will not otherwise be immediately obvious to all users.

Recommendation

Make sure that status messages are programmatically determined. To achieve this, use a polite live region, such as `role="status"`, and `aria-atomic` to make sure that the entire text node is always announced.

For live regions to work, an empty container with live region semantics must be present on page load (before the region message is triggered). The live region message needs to be dynamically added to the container when triggered. For example, rather than toggling the visibility of the text using CSS. Also note, that you need to empty this container again, before dynamically adding a new message. For example:

```

<span role="status" aria-atomic="true"></span>
<span role="status" aria-atomic="true">10 results found</span>
<span role="status" aria-atomic="true"></span>
<span role="status" aria-atomic="true">No results found</span>

```

Another solution is to move focus or prompt a page load. In such cases, you do not need to programmatically determine the status message as the change of context, such as focus moving to the added content or a new page being loaded, provides an acceptable call-to-action.

For more information, see [MDN - Using the status role](#) and [MDN - ARIA Live Regions](#).

[Return to priority table](#)

Showing results 1 to 10 of 3157 rows

< Previous 1 2 3 4 5 ... 316 Next >

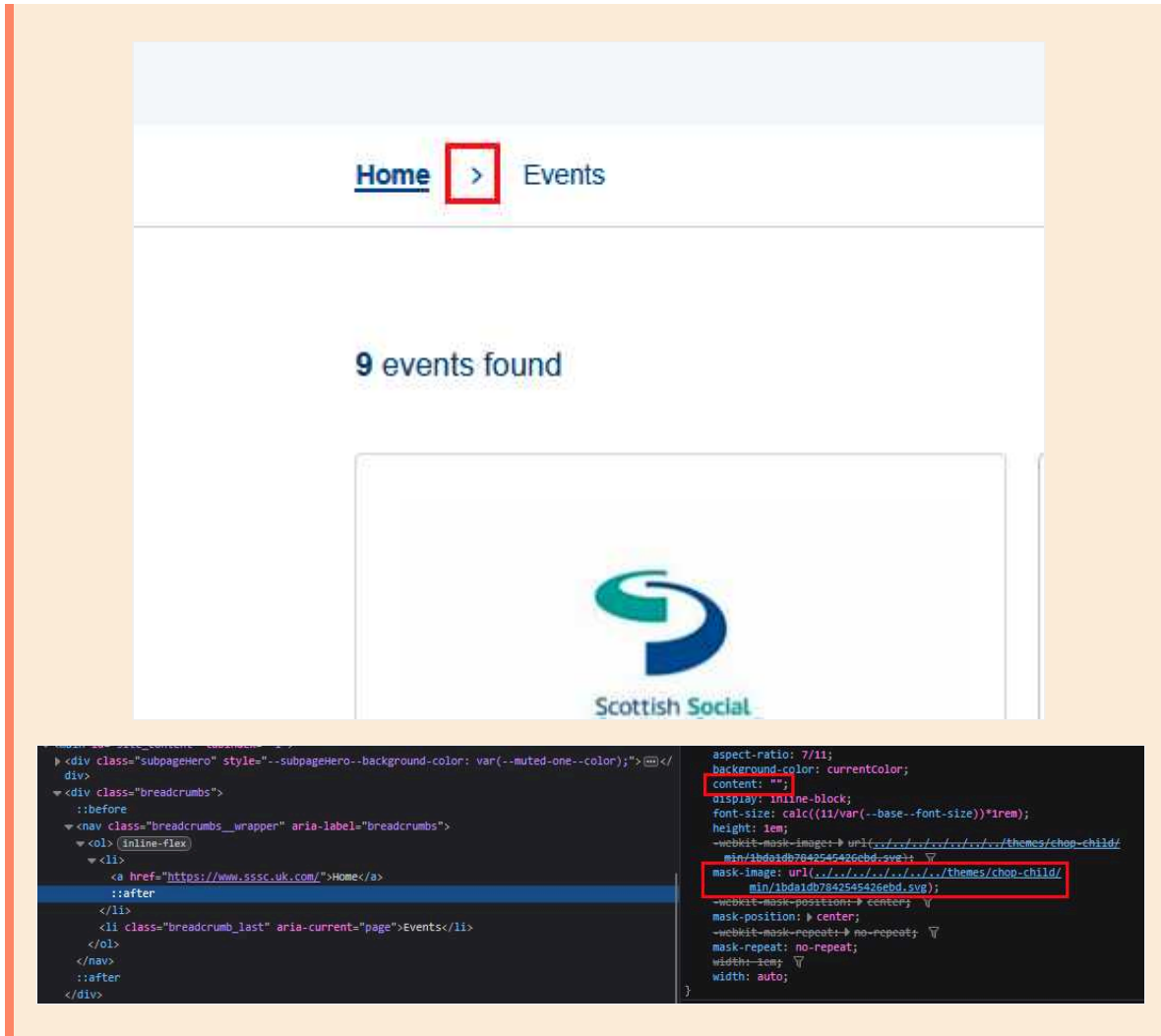
Also in registration

```
<li class="page-item page-item--previous">
  <button class="page-link" aria-label="Previous Page" type="button" disabled=""> event flex
    <i class="mSvgIcon mSvgIcon--mask mSvgIcon--chevron-left-regular" aria-hidden="true"></i>
    Previous
  </button>
</li>
<li class="page-numbers"> ... </li>
<li class="page-numbers"> ... </li>
<li class="page-numbers"> ... </li>
<li class="page-numbers"> ... </li>
<li class="page-numbers"> ... </li>
<li class="page-numbers"> ... </li>
<li class="page-numbers"> ... </li>
<li class="page-item page-item--next">
  <button class="page-link" aria-label="Next Page" type="button"> event flex
    Next
    <i class="mSvgIcon mSvgIcon--mask mSvgIcon--chevron-right-regular" aria-hidden="true"></i>
  </button>
</li>
</ul>
```

Fixed, with comments: on pages 4 and 6, on desktop, the decorative “>” icons in the breadcrumb region are not exposed to assistive technology. The `::after` pseudo-element has an empty `content` property.

We believe the instance may have been raised in error and apologise for the inconvenience.

Note, however, that for consistency and to ensure that decorative icons do not have content that is exposed to assistive technology, we recommend applying `aria-hidden="true"` to the icons as in other instances.



Recheck comment 11/06/25

This issue was updated in additional testing time on 11/06/2025.

A new instance of the same issue has been observed on pages 4 and 6. On this page, the “>” icon in the breadcrumbs is not hidden from assistive technologies. A new instance of the same issue has been observed on pages 3 and 9. On this page, the “<” and “>” icons in the “Previous” and “Next” buttons are not hidden from assistive technologies.

What happened

Non-text content must have appropriate alternatives. This is required for the content to be understood by all users. Images, icons and so on may otherwise be miscommunicated by assistive technology. There are decorative icons that have not been hidden from assistive technology.

We have classed the images as decorative as they do not communicate any information that is not otherwise available on the page.

Where it happened

This can be seen on page 3, on desktop and iOS, with the decorative “...” icon within the pagination component. The decorative icon is not hidden from assistive technology.

Note that this non-interactive element also incorrectly receives keyboard focus. This has been raised as a separate issue in ADR_22749 - 37.

Showing results 1 to 10 of 100 rows

< Previous 1 2 3 4 5 ... 10 Next >

```

<li class="page-numbers">... </li>
<li class="page-numbers disabled page-numbers--dots">
  <a class="page-link" tabindex="0"> flex
  -
  </a>
</li>
    
```

This can be seen on pages 3 and 9, on desktop, with the decorative “<” and “>” icons in the “Next” and “Previous” buttons. The decorative icons are not hidden from assistive technology.

<https://www.sssc.uk.com/sign-up-for-our-newsletters/>

< Previous 1 2 3 Next >

```

<ul class="pagination"> flex
  <!-->
  <li class="page-item page-item--previous disabled">
    <a class="page-link" tabindex="-1"> event flex
      ::before
      Previous
    </a>
  </li>
  <li class="page-numbers active"> ... </li>
  <li class="page-numbers"> ... </li>
  <li class="page-numbers"> ... </li>
  <li class="page-item page-item--next">
    <a class="page-link" tabindex="0"> event flex
      Next
      ::after
    </a>
  </li>
  <!-->
</ul>
    
```

This can be seen on pages 4 and 6, on desktop, with the decorative “>” icon in the breadcrumbs. The decorative icon is not hidden from assistive technology.

Events

Discover all the latest events on offer from the SSSC

[Home](#) > [Events](#)

```

<nav class="breadcrumbs_wrapper" aria-label="breadcrumbs">
  <ol> inline-flex
    <li>
      <a href="https://www.sssc.uk.com/">Home</a>
      <span>::after</span>
    </li>
    <li class="breadcrumb_last" aria-current="page">Events</li>
  </ol>
</nav>

```

Steps to reproduce

1. Observe the icons.
2. Observe that they do not visually communicate any more information than is otherwise available.
3. Observe that they do not provide any further context or clarity than the information on the page.
4. Inspect the icons.
5. Confirm that they have not been hidden from assistive technology.

Recommendation

Decorative images that do not convey meaning or unique information need to be programmatically hidden.

Images marked up with an `` tag must have an alt attribute. Use an empty `alt` attribute to programmatically hide decorative images. For example:

```
Error: The email address field cannot be left blank
```

Inline SVG images can use `aria-hidden` and `focusable="false"` for the same purpose. For example:

```
<svg aria-hidden="true" focusable="false">
...
</svg>
```

The `aria-hidden` attribute can also be used with icon fonts and CSS generated content. For example:

```
<div>
  <span aria-hidden="true">::before</span>
  Error: The email address field cannot be left blank
</div>
<div>
  <i class="fas fa-exclamation-triangle" aria-hidden="true"></i>
  Error: The email address field cannot be left blank
</div>
```

For decorative `<svg>` elements, use the `aria-hidden="true"` and `focusable="false"` attributes. For more information, visit [CSS Tricks - Accessible SVGs](#).

For decorative icon fonts and CSS-generated content, use the `aria-hidden="true"` attribute. For more information, visit [MDN - aria-hidden](#), [a11y Project - How to Hide Content](#), and [CloudFour - Hidden Content and Accessibility](#).

For more information, see [Web Accessibility Tutorials - Informative Images](#), and [Alt Attribute Decision Tree](#).

[Return to priority table](#)

ADR_22749-8 **LOW PRIORITY** **RESOLVED**

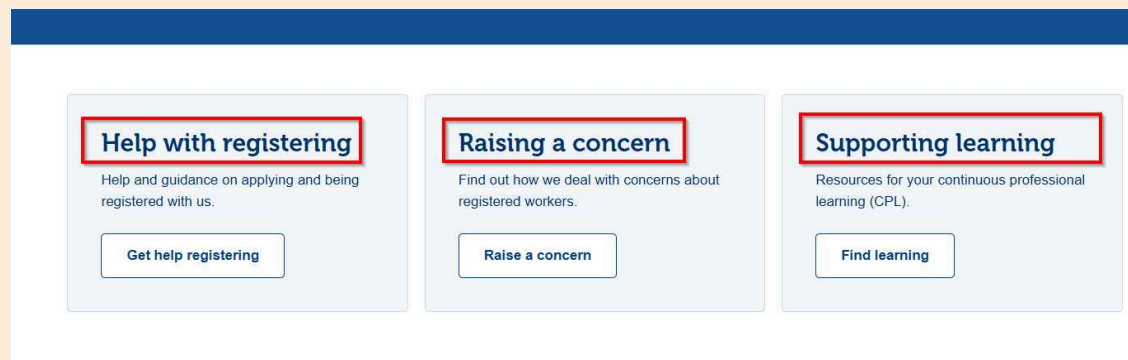
Unordered list not marked up semantically (Pages 1-3)

Success Criteria	1.3.1: Info and Relationships [2.0 A]
Devices	None
Pages Affected	None
Users Affected	Vision
Component	Link, List
ADR Type	Web Audit

Recheck comment 26/06/25

This issue is fixed.

Fixed: on page 1 on desktop and iOS, the list of card links “Help with registering”, “Raising a concern”, and “Supported Learning” is semantically marked up as unordered list items.



```

<div class="featuredCTA_wrapper">
  <ul class="featuredCTA_list"> <grid>
    <li class="featuredCTA_listItem">
      <a class="featuredCTA_item ctaItem" href="/registration"> <flex>
        <h2 class="ctaItem_heading heading heading--h2">Help with registering</h2>
        <p> </p>
        <span class="mButton"> </span> <flex>
      </a>
    </li>
    <li class="featuredCTA_listItem"> </li>
    <li class="featuredCTA_listItem"> </li>
  </ul>
</div>
</section>

```

Fixed: on page 1 on desktop and iOS, the lists of card links under the “More from the SSSC” heading, such as the “Search the Register” link, are marked up as unordered list items.



```

<div class="featuredCTA_wrapper">
  <ul class="featuredCTA_list">
    <li class="featuredCTA_listItem">
      <a class="featuredCTA_item ctaItem" href="https://www.sssc.uk.com/search-the-register/">
        <h3 class="ctaItem_heading heading heading--h3">Search the Register</h3>
        <p>Find details of SSSC registered workers</p>
        <i class="mSvgIcon mSvgIcon--chevron-right-regular" aria-hidden="true"></i>
      </a>
    </li>
    <li class="featuredCTA_listItem">
      <a class="featuredCTA_item ctaItem" href="https://www.sssc.uk.com/standards/codes-of-practice/">
        <h3 class="ctaItem_heading heading heading--h3">Codes of Practice</h3>
        <p>Download the SSSC Codes of Practice</p>
        <i class="mSvgIcon mSvgIcon--chevron-right-regular" aria-hidden="true"></i>
      </a>
    </li>
    <li class="featuredCTA_listItem"></li>
    <li class="featuredCTA_listItem"></li>
    <li class="featuredCTA_listItem"></li>
    <li class="featuredCTA_listItem"></li>
    <li class="featuredCTA_listItem"></li>
    <li class="featuredCTA_listItem"></li>
    <li class="featuredCTA_listItem"></li>
  </ul>
</div>
</section>
/main>
    
```

Fixed: on page 1 on desktop and iOS, the list of links under the “Popular suggestions” heading, such as the “Test Post 1” link, are marked up as unordered list items.

Search SSSC

Q What are you looking for?

Popular suggestions

- Impairment decisions >
- Hearings >
- Raising a concern with us >
- Codes of Practice >
- Continuous professional learning >
- How do I register? >

```

<div class="searchDialog_suggestions">
  <h3 class="heading heading--h5">Popular suggestions</h3>
  <ul class="searchDialog_list"> grid
    <li class="searchDialog_listItem">
      <a class="searchDialog_suggestion" href="https://www.sssc.uk.com/dealing-with-concerns/impairment-decisions/"> flex
        Impairment decisions
        <i class="mSvgIcon mSvgIcon--chevron-right-regular" aria-hidden="true"> </i> flex
      </a>
    </li>
    <li class="searchDialog_listItem"> </li>
    <li class="searchDialog_listItem"> </li>
    <li class="searchDialog_listItem"> </li>
    <li class="searchDialog_listItem"> </li>
  </ul>
</div>
</div>

```

Fixed: on page 2 on desktop and iOS, the list of cards links under the “Most viewed in Careers and education” and “All Careers and education resources” headings, such as “Social work”, “Social care”, “Children and young people”, and “Getting started” are marked up as unordered list items.



What happened

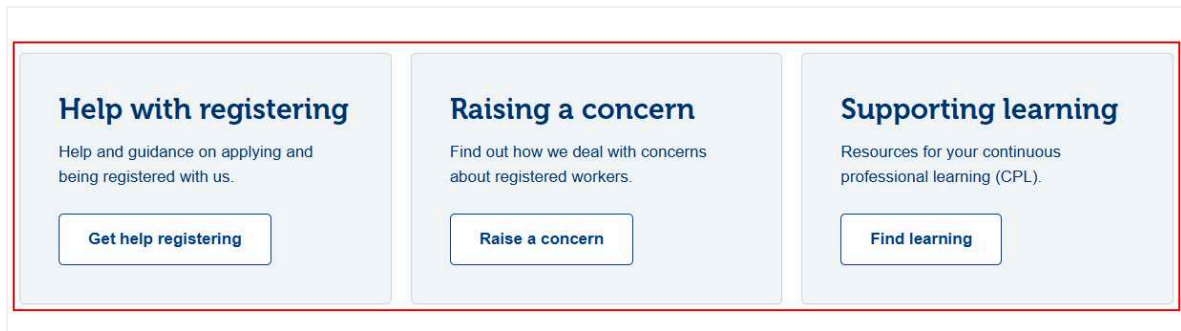
If a relationship between elements is expressed with visual cues, it also needs to be programmatically determined. This enables assistive technology to communicate the relationships to users. Content may otherwise seem unstructured or disorganised.

There are elements that are visually communicated as being interrelated, unordered list items. However, this structure has not been programmatically determined.

Where it happened

This can be seen on page 1 on desktop and iOS, with the list of card links “Help with registering”, “Raising a concern” and “Supported Learning”. The links are not marked up as unordered list items.

Note that if heading markup is added to the card headings, as per ADR_22749 - 4, list semantics will no longer be strictly required, but it is best practice.

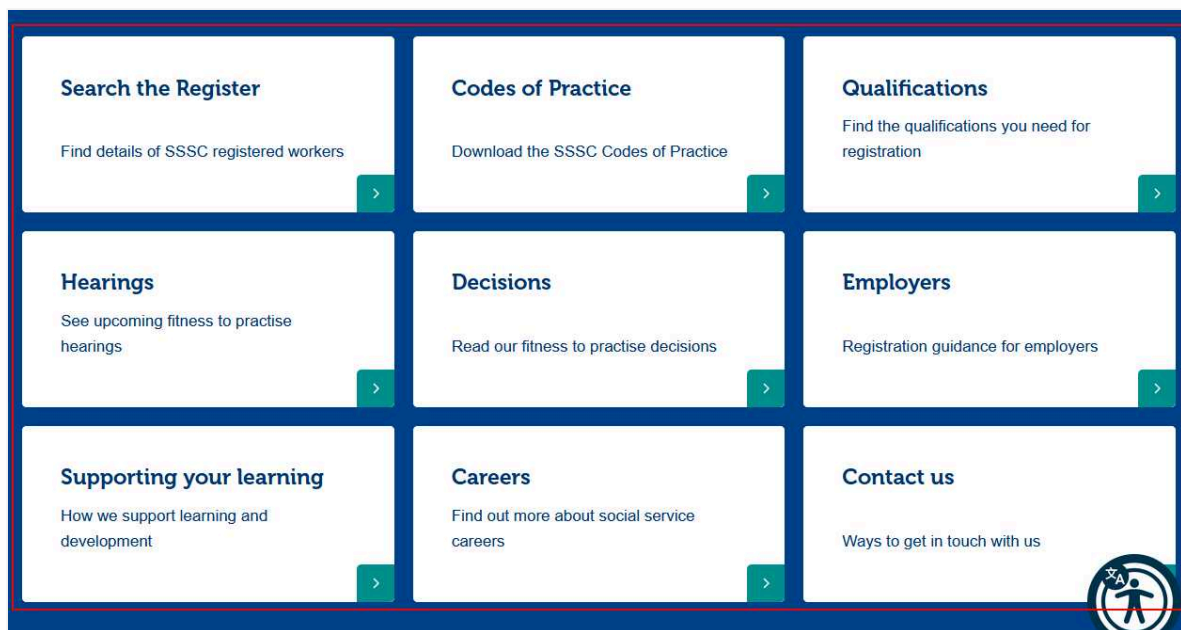


```

<section class="featuredCTA featuredCTA--none" style="--featuredCta--background-color: var(--white--bg-color); --featuredCta--column-count: 3;">
  <div class="featuredCTA_wrapper"> grid
    <a class="featuredCTA_item ctaItem" href="/registration"> flex
      <strong class="ctaItem_heading heading heading--h2">Help with registering</strong>
      <p> </p>
      <span class="mButton"> </span> flex
    </a>
    <a class="featuredCTA_item ctaItem" href="https://www.ssscukcom.mtcserver.com/dealing-with-concerns/raising-a-concern-with-us/" rel="nofollow noopener nofollow">
    </a> flex
    <a class="featuredCTA_item ctaItem" href="https://www.ssscukcom.mtcserver.com/supporting-your-learning/" rel="nofollow noopener nofollow"> </a> flex
  </div>
</section>
    
```

This can also be seen on page 1 on desktop and iOS, with the list of card links under the “More from the SSSC” heading, such as the “Search the Register” link. The card links are not marked up as unordered list items.

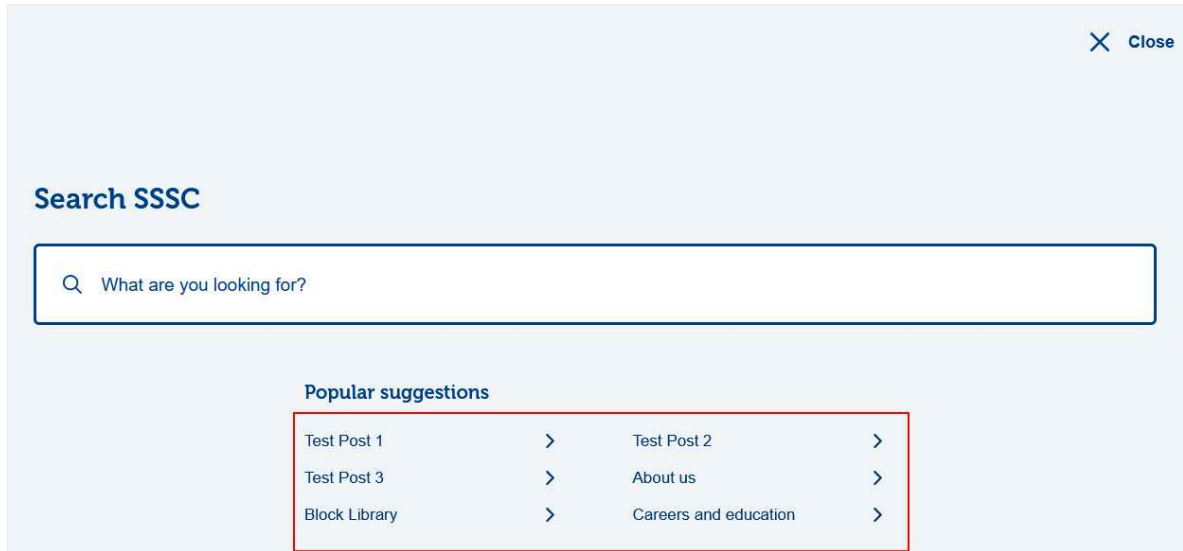
Note that if heading markup is added to the card headings, as per ADR_22749 - 4, list semantics will no longer be strictly required, but it is best practice.



```

<section class="featuredCTA featuredCTA--simple" style="--featuredCta--background-color: var(--secondary-one--color); --featuredCta--column-count: 3;">
  <h2 class="featuredCTA_heading heading heading--h1">More from the SSSC</h2>
  <div class="featuredCTA_wrapper"> grid
    <a class="featuredCTA_item ctaItem" href="https://www.ssscukcom.mtcserver.com/search-the-register/" rel="nofollow noopener nofollow"> flex
      <strong class="ctaItem_heading heading heading--h3">Search the Register</strong>
      <p>Find details of SSSC registered workers</p>
      <i class="mSvgIcon mSvgIcon--chevron-right-regular" aria-hidden="true"> </i> flex
    </a>
    <a class="featuredCTA_item ctaItem" href="https://www.ssscukcom.mtcserver.com/standards/codes-of-practice/" rel="nofollow noopener nofollow"> </a> flex
    <a class="featuredCTA_item ctaItem" href="/registration/help-with-register-parts-fees-and-qualifications/"> </a> flex
    <a class="featuredCTA_item ctaItem" href="/dealing-with-concerns/upcoming-hearings/"> </a> flex
    <a class="featuredCTA_item ctaItem" href="/dealing-with-concerns/impairment-decisions/"> </a> flex
    <a class="featuredCTA_item ctaItem" href="https://www.ssscukcom.mtcserver.com/registration/employers/" rel="nofollow noopener nofollow"> </a> flex
    <a class="featuredCTA_item ctaItem" href="https://www.ssscukcom.mtcserver.com/supporting-your-learning/" rel="nofollow noopener nofollow"> </a> flex
    <a class="featuredCTA_item ctaItem" href="https://www.ssscukcom.mtcserver.com/careers-and-education/" rel="nofollow noopener nofollow"> </a> flex
    <a class="featuredCTA_item ctaItem" href="https://www.ssscukcom.mtcserver.com/contact-us/" rel="nofollow noopener nofollow"> </a> flex
  </div>
</section>
    
```

This can also be seen on page 1 on desktop and iOS, with the list of links under the “Popular suggestions” heading, such as the “Test Post 1” link. The links are not marked up as unordered list items.



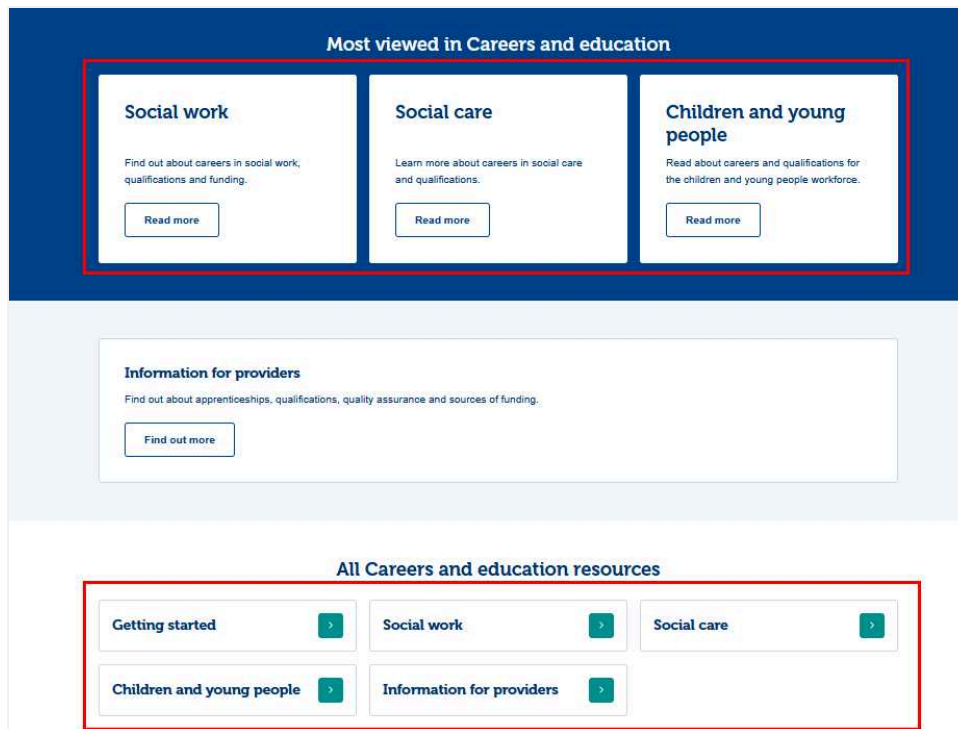
```

<div class="searchDialog_suggestions"> {grid}
  <h3 class="heading heading--h5">Popular suggestions</h3>
  <a class="searchDialog_suggestion" href="https://www.ssscukcom.mtcserver.com/test-post-1/"> {flex}
    Test Post 1
    <i class="mSvgIcon mSvgIcon--chevron-right-regular" aria-hidden="true"> {flex}
  </a>
  <a class="searchDialog_suggestion" href="https://www.ssscukcom.mtcserver.com/test-post-2/"> {flex}
  <a class="searchDialog_suggestion" href="https://www.ssscukcom.mtcserver.com/?p=1433"> {flex}
  <a class="searchDialog_suggestion" href="https://www.ssscukcom.mtcserver.com/about-us/"> {flex}
  <a class="searchDialog_suggestion" href="https://www.ssscukcom.mtcserver.com/block-library/"> {flex}
  <a class="searchDialog_suggestion" href="https://www.ssscukcom.mtcserver.com/careers-and-education/"> {flex}
</div>

```

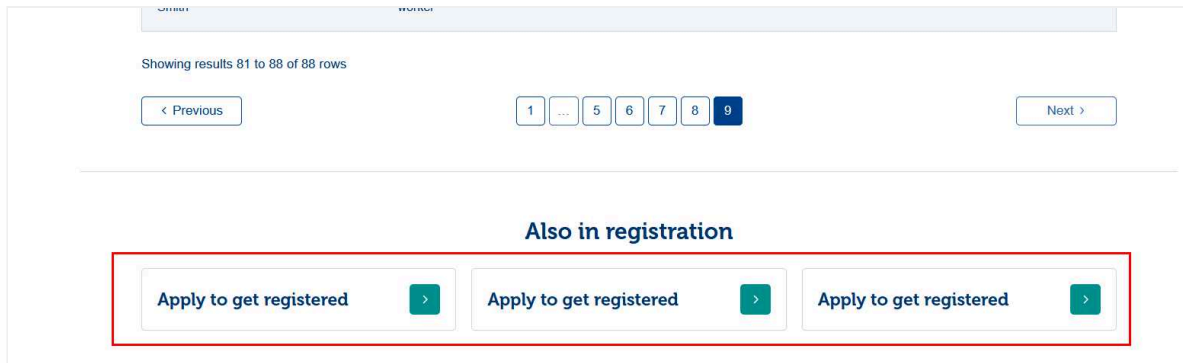
This can also be seen on page 2 on desktop and iOS, with the list of card links under the “Most viewed in Careers and education” and “All Careers and education resources” headings, such as “Social work”, “Social care”, “Children and young people”, “Getting started” and so on. The links are not marked up as unordered list items.

Note that if heading markup is added to the card headings, as per ADR_22749 - 4, list semantics will no longer be strictly required, but it is best practice.



```
<section id="most-viewed-in-careers-and-education" class="featuredCTA featuredCTA--none" style="--featuredCta--background-color: var(--secondary-one--color); --featuredCta--column-count: 3;" data-quick-links-label="Most viewed in Careers and education">
  <h2 class="featuredCTA_heading heading heading--h2">Most viewed in Careers and education</h2>
  <div class="featuredCTA_unwrapper" grid>
    <a class="featuredCTA_item ctaItem" href="https://www.ssscuk.com/mtcserver.com/careers-and-education/social-work/" rel="nofollow noopener nofollow"> {flex}
      <strong class="ctaItem_heading heading heading--h2">Social work</strong>
      <p> {flex}
        <span class="Button"> {flex}
      </p>
    </a>
    <a class="featuredCTA_item ctaItem" href="https://www.ssscuk.com/mtcserver.com/careers-and-education/social-care/" rel="nofollow noopener nofollow"> {flex}
      <strong class="ctaItem_heading heading heading--h2">Social care</strong>
      <p> {flex}
        <span class="Button"> {flex}
      </p>
    </a>
    <a class="featuredCTA_item ctaItem" href="https://www.ssscuk.com/mtcserver.com/careers-and-education/children-and-young-people/" rel="nofollow noopener nofollow"> {flex}
      <strong class="ctaItem_heading heading heading--h2">Children and young people</strong>
      <p> {flex}
        <span class="Button"> {flex}
      </p>
    </a>
  </div>
</section>
```

This can also be seen on page 3 on desktop and iOS, with the list of links underneath the “Also in registration”. All three links have the label “Apply to get registered”. The links are not marked up as unordered list items.



```

<section class="featuredCTA featuredCTA--small" style="--featuredCta--background-color: var(--white--bg-color); --featuredCta--column-count: 3;">
  <h2 class="featuredCTA_heading heading heading--h2">Also in registration</h2>
  <div class="featuredCTA_wrapper">
    <a class="featuredCTA_item ctaItem" href="https://www.ssscukcom.mtcserver.com/impairment-decisions/" rel="nofollow noopener nofollow">
      <strong class="ctaItem_heading heading heading--h3">Apply to get registered</strong>
      <i class="mSvgIcon mSvgIcon--chevron-right-regular" aria-hidden="true"></i>
    </a>
    <a class="featuredCTA_item ctaItem" href="https://www.ssscukcom.mtcserver.com/about-us/" rel="nofollow noopener nofollow">
      <strong class="ctaItem_heading heading heading--h3">Apply to get registered</strong>
      <i class="mSvgIcon mSvgIcon--chevron-right-regular" aria-hidden="true"></i>
    </a>
    <a class="featuredCTA_item ctaItem" href="https://www.ssscukcom.mtcserver.com/hearings/" rel="nofollow noopener nofollow">
      <strong class="ctaItem_heading heading heading--h3">Apply to get registered</strong>
      <i class="mSvgIcon mSvgIcon--chevron-right-regular" aria-hidden="true"></i>
    </a>
  </div>
</section>
</main>

```

Steps to reproduce

1. Observe the unordered list items.
2. Inspect the elements in the DOM.
3. Observe that the list items have not been marked up with `` and ``.

Recommendation

Mark up the unordered list using `` and ``. Make sure that each set of list items is contained in a single `` and that the list items themselves are contained within individual `` tags. Only use separate `` tags when the list items are contained logically and visually in separate, distinct lists. For example:

```

<ul>
  <li>Pears</li>
  <li>Apples</li>
  ...
</ul>

```

Nested lists can be implemented in a similar fashion. For example:

```
<ul>
  <li>Pears</li>
  <li>Apples
    <ul>
      <li>Granny Smith</li>
      <li>Golden Delicious</li>
    </ul>
  </li>
  <li>Oranges</li>
  ...
</ul>
```

For more information, see [WAI Tutorials - Unordered Lists](#).

[Return to priority table](#)

ADR_22749-11 **LOW PRIORITY** **RESOLVED**

Ordered list not marked up semantically

Success Criteria	1.3.1: Info and Relationships [2.0 A]
Devices	None
Pages Affected	None
Users Affected	Vision
Component	List
ADR Type	Web Audit

Recheck comment 26/06/25

This issue is fixed.

On page 2 on desktop, the breadcrumb navigation links are nested in a `<nav>` element with an appropriate accessible name and appropriately use the ordered list semantically.

The screenshot shows a web page titled "Careers and education" with the subtitle "Information on qualifications and your career in care". Below the title is a breadcrumb navigation bar: "Home > Careers and education". Underneath is a "Quick links" section with four items: "Overview", "Most viewed in Careers and education", "Information for providers", and "All Careers and education resources".

Below the screenshot is a code editor showing the HTML structure of the breadcrumb navigation. A red box highlights the following code:

```

<ol>
  <li>
    <a href="https://www.sssc.uk.com/">Home</a>
    ::after
  </li>
  <li class="breadcrumb_last" aria-current="page">Careers and education</li>
</ol>

```

What happened

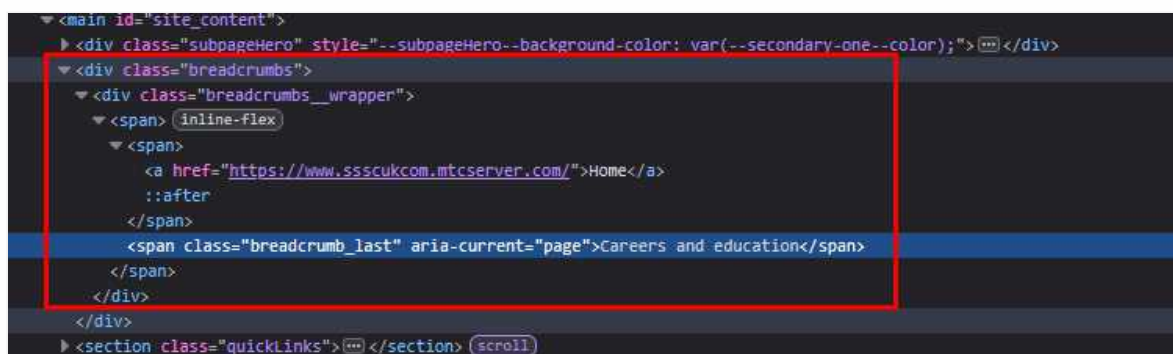
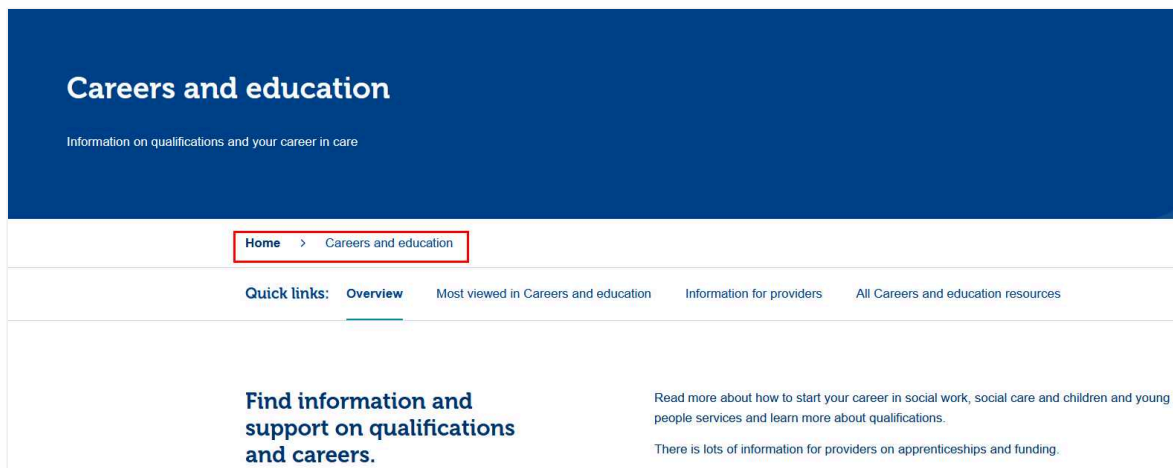
If a relationship between elements is expressed with visual cues, it also needs to be programmatically determined. This enables assistive technology to communicate the relationships to users. Content may otherwise seem unstructured or disorganised.

There are elements that are visually communicated as being interrelated, ordered list items. However, this structure has not been programmatically determined.

Where it happened

This can be found on page 2 with the breadcrumb navigation. The sequential items are not marked up semantically.

Note that while it is not a strict requirement for WCAG AA, it is recommended that the breadcrumbs are nested in a `<nav>` element with an appropriate accessible name.



Steps to reproduce

1. Observe the ordered list items.
2. Inspect the elements in the DOM.
3. Observe that the list items have not been marked up with `` and ``.

Recommendation

Mark up the ordered list using `` and ``. Make sure that each set of list items is contained in a single `` and that the list items themselves are contained within individual `` tags. Only use separate `` tags when the list items are contained logically and visually in separate, distinct lists. For example:

```
<ol>
  <li>Collect the ingredients</li>
  <li>Measure the ingredients</li>
  ...
</ol>
```

Nested lists can be implemented in a similar fashion. For example:

```
<ol>
  <li>Collect the ingredients</li>
  <li>Measure the ingredients
    <ul>
      <li>100g flour</li>
      <li>2 eggs</li>
    </ul>
  </li>
  <li>Mix the ingredients</li>
  ...
</ol>
```

For more information, see [WAI Tutorials - Ordered Lists](#).

[Return to priority table](#)

ADR_22749-39 **LOW PRIORITY** **RESOLVED**

Description list not semantically declared

- Success Criteria** 1.3.1: Info and Relationships [2.0 A]
- Devices** None
- Pages Affected** None
- Users Affected** Vision
- Component** List
- ADR Type** Web Audit

Recheck comment 27/06/25

This issue is fixed.

On page 3 on iOS, the search results are displayed as a table and correctly marked up as such, as it is marked up on the desktop.



```

<table class="dynamic365form_table table--alternating" aria-label="SSSC Register Table" aria-
describedby="dynamic365form_figcaption">
  <thead>
    <tr>
      <th>Individual name</th>
      <th>Town of employment</th>
      <th>Register part(s)</th>
      <th>Registration status</th>
      <th>Registration number</th>
      <th>Removal date</th>
      <th>Suspended from</th>
      <th>Suspended to</th>
      <th>Specialist award</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td data-label="Individual name">
        <span>Bernitha Ajanigo Timothy</span>
      </td>
      <td data-label="Town of employment">
        <span>Bellshill</span>
      </td>
      <td data-label="Register part(s)"></td>
      <td data-label="Registration status"></td>
      <td data-label="Registration number"></td>
      <td data-label="Removal date"></td>
      <td data-label="Suspended from"></td>
      <td data-label="Suspended to"></td>
      <td data-label="Specialist award"></td>
    </tr>
  </tbody>
</table>

```

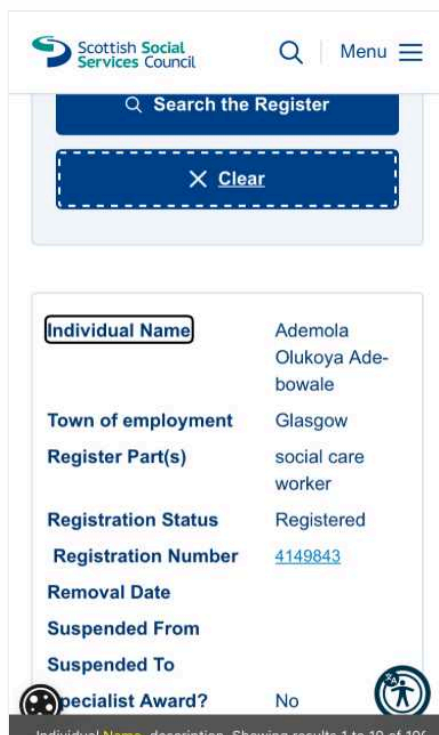
What happened

If a relationship between elements is expressed with visual cues, it also needs to be programmatically determined. This enables assistive technology to communicate the relationships to users. Content may otherwise seem unstructured or disorganised.

There are elements that are visually communicated as being key and value pairs. However, this structure has not been programmatically determined.

Where it happened

This can be seen on page 3, on iOS, with the search results. On desktop, the search results are displayed as a table and correctly marked up as such. On iOS, the search results are displayed as multiple description lists. However, they are still marked up as a table and not as description lists.



```

<table class="dynamic365form_table table--alternating" aria-live="polite" aria-busy="false" aria-label="SSSC Register Table" aria-describedby="dynamic365form_figcaption">
  <thead>
  </thead>
  <tbody>
    <tr>
      <td data-label="Individual Name">
        <subgrid>
          <::before>
            <span>Ademola Olukoya Adebowale</span>
          </td>
      <td data-label="Town of employment">
        <subgrid>
          </td>
      <td data-label="Register Part(s)">
        <subgrid>
          </td>
      <td data-label="Registration Status">
        <subgrid>
          </td>
      <td data-label="Registration Number">
        <subgrid>
          </td>
      <td data-label="Removal Date">
        <subgrid>
          </td>
      <td data-label="Suspended From">
        <subgrid>
          </td>
      <td data-label="Suspended To">
        <subgrid>
          </td>
      <td data-label="Specialist Award?">
        <subgrid>
          </td>
    </tr>
  </tbody>
</table>
  
```

Steps to reproduce

1. Observe description list items.
2. Inspect the elements in the DOM.
3. Observe that the list items have not been marked up using description list semantics, `<dl>`, `<dt>`, and `<dd>`.

Recommendation

Mark up the description list using `<dl>`, `<dt>`, and `<dd>`. Make sure that each set of list items is contained in a single `<dl>`, contain each individual term within a `<dt>` tag and each description within a `<dd>` tag. Only use separate `<dl>` tags when the list items are contained logically and visually in separate, distinct lists.

Mark up the list using description list semantics to communicate the relationship between terms and their descriptions. For example:

```
<dl>
  <dt>
    Term one
  </dt>
  <dd>
    Definition one
  </dd>
  <dt>
    Term two
  </dt>
  <dd>
    Definition two
  </dd>
</dl>
```

In this case, the table markup is acceptable, provided it includes appropriate table headers defining the terms.

For more information, see [WAI Tutorials - Description Lists](#).

[Return to priority table](#)

ADR_22749-52 LOW PRIORITY **RESOLVED**

Unordered list not marked up semantically (Page 8)

Success Criteria	1.3.1: Info and Relationships [2.0 A]
Devices	None
Pages Affected	None
Users Affected	Vision
Component	List
ADR Type	Web Audit

Recheck comment 17/07/25

The issue is fixed.

On page 8, on desktop and iOS, the buttons within the “1. What best describes you?”, “2. Select the type of service you work in?”, and “3. Select the level of your job role?” accordions are marked up as an unordered list correctly. The `` elements are now nested in an element with `role="group"` rather than this being on the list itself.

1. What best describes you?

social care worker

Someone who works in social care such as care at home, housing support, care home and adult day care services, or inspects these types of services.

children and young people's worker

Someone who works with children and young people in day care of children, residential child care and residential school care accommodation services or inspects these types of services.

social worker

A social worker is someone who has a social work degree

social work student

A social work student is someone who has been accepted on a social work degree course

```

<div role="group" aria-labelledby="details_label--registerPart">
  <ul class="buttonGroup buttonGroup--stacked list--noStyle">
    <li>
      <button class="mButton mButton--alignLeft" type="button" aria-pressed="false">
        <span class="mButton_text">
          <strong class="heading heading--h3">social care worker</strong>
          <span>
            Someone who works in social care such as care at home, housing support, care home and adult day care services, or inspects these types of services.
          </span>
        </span>
      </button>
    </li>
    <li></li>
    <li></li>
    <li></li>
  </ul>
</div>

```

```

<div role="group" aria-labelledby="details_label--service">
  <ul class="buttonGroup buttonGroup--stacked list--noStyle">
    <li>
      <button class="mButton mButton--alignLeft" type="button" aria-pressed="false">
        <span class="mButton_text">
          <strong class="heading heading--h3">adult day care services</strong>
        </span>
      </button>
    </li>
    <li></li>
    <li></li>
    <li></li>
  </ul>
</div>

```

```

<div class="details_inner">
  <div role="group" aria-labelledby="details_label--service">
    <ul class="buttonGroup buttonGroup--stacked list--noStyle">
      <li>
        <button class="mButton mButton--alignLeft" type="button" aria-pressed="false">
          <span class="mButton_text">
            <strong class="heading heading--h3">adult day care services</strong>
          </span>
        </button>
      </li>
      <li></li>
      <li></li>
      <li></li>
    </ul>
  </div>
</div>

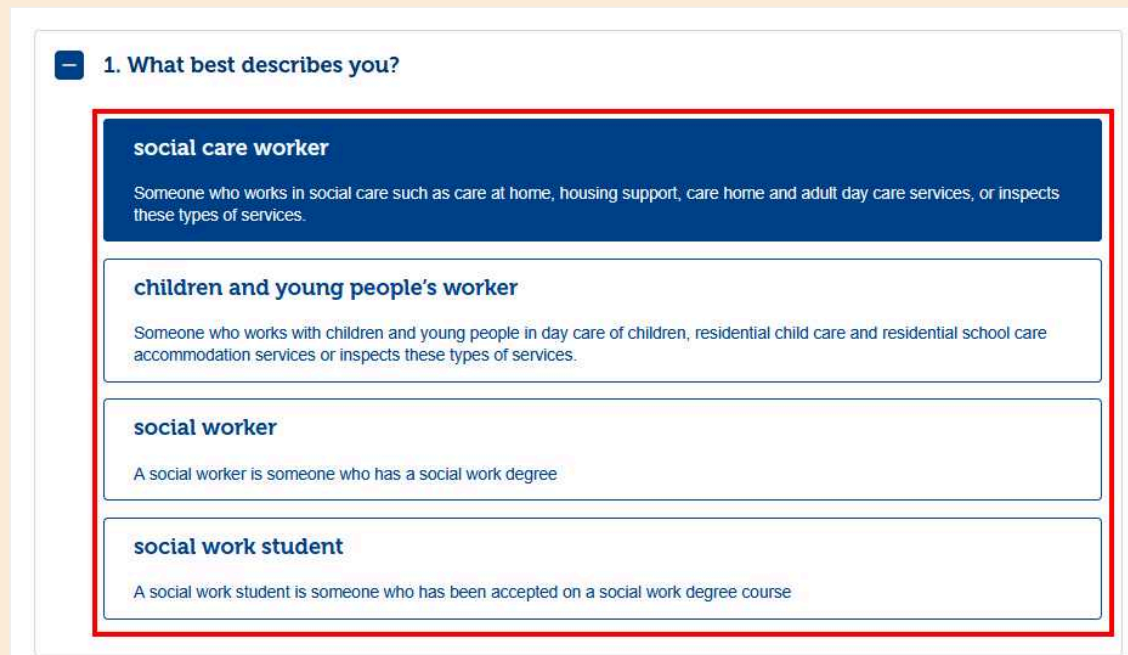
```

Recheck comment 27/06/25

This issue is partially fixed.

On page 8 on desktop and iOS, the buttons within the “1. What best describes you?”, “2. Select the type of service you work in?”, and “3. Select the level of your job role?” accordions are marked up as an unordered list. However, the `` has `role="group"` on it, which overrides the list semantics.

To fix the list semantics while maintaining the solution to ADR_22749 - 51, the list can be nested inside a `<div>` with `role="group"`.



```

<div class="details_inner">
  <ul class="buttonGroup buttonGroup--stacked list--noStyle" role="group" aria-labelledby="details_label--registerPart">
    <li>
      <button class="mButton mButton--alignLeft" type="button" aria-pressed="false"> event inline-flex
      <span class="mButton_text">
        <strong class="heading heading--h3">social care worker</strong>
        <span> </span>
      </span>
    </button>
  </li>
  <li> </li>
  <li> </li>
  <li> </li>
</ul>
</div>
</details>
    
```

What happened

If a relationship between elements is expressed with visual cues, it also needs to be programmatically determined. This enables assistive technology to communicate the relationships to users. Content may otherwise seem unstructured or disorganised.

There are elements that are visually communicated as being interrelated, unordered list items. However, this structure has not been programmatically determined.

Where it happened

This can be seen on page 8, on desktop and iOS, with the buttons within the “1. What best describes you?”, “2. Select the type of service you work in?”, and “3. Select the level of your job role?” accordions. The related buttons within each accordion are not marked up as an unordered list.

1. What best describes you?

social care worker

Someone who works in social care such as care at home, housing support, care home and adult day care services, or inspects these types of services.

children and young people’s worker

Someone who works with children and young people in day care of children, residential child care and residential school care accommodation services or inspects these types of services.

social worker

A social worker is someone who has a social work degree

social work student

A social work student is someone who has been accepted on a social work degree course

```

<div class="details_inner">
  <div class="buttonGroup buttonGroup--stacked"> flex
    <button class="mButton mButton--alignLeft mButton--primary" type="button"> event flex
      <span class="mButton_text">
        <span class="heading heading--h3">social care worker</span>
      </span>
    </button>
    <button class="mButton mButton--alignLeft" type="button"> event flex
    <button class="mButton mButton--alignLeft" type="button"> event flex
    <button class="mButton mButton--alignLeft" type="button"> event flex
  </div>
</div>

```

Steps to reproduce

1. Observe the unordered list items.
2. Inspect the elements in the DOM.
3. Observe that the list items have not been marked up with `` and ``.

Recommendation

Mark up the unordered list using `` and ``. Make sure that each set of list items is contained in a single `` and that the list items themselves are contained within individual `` tags. Only use separate `` tags when the list items are contained logically and visually in separate, distinct lists. For example:

```

<ul>
  <li>Pears</li>
  <li>Apples</li>
  ...
</ul>

```

Nested lists can be implemented in a similar fashion. For example:

```
<ul>
  <li>Pears</li>
  <li>Apples
    <ul>
      <li>Granny Smith</li>
      <li>Golden Delicious</li>
    </ul>
  </li>
  <li>Oranges</li>
  ...
</ul>
```

For more information, see [WAI Tutorials - Unordered Lists](#).

[Return to priority table](#)

ADR_22749-41 **LOW PRIORITY** **RESOLVED**

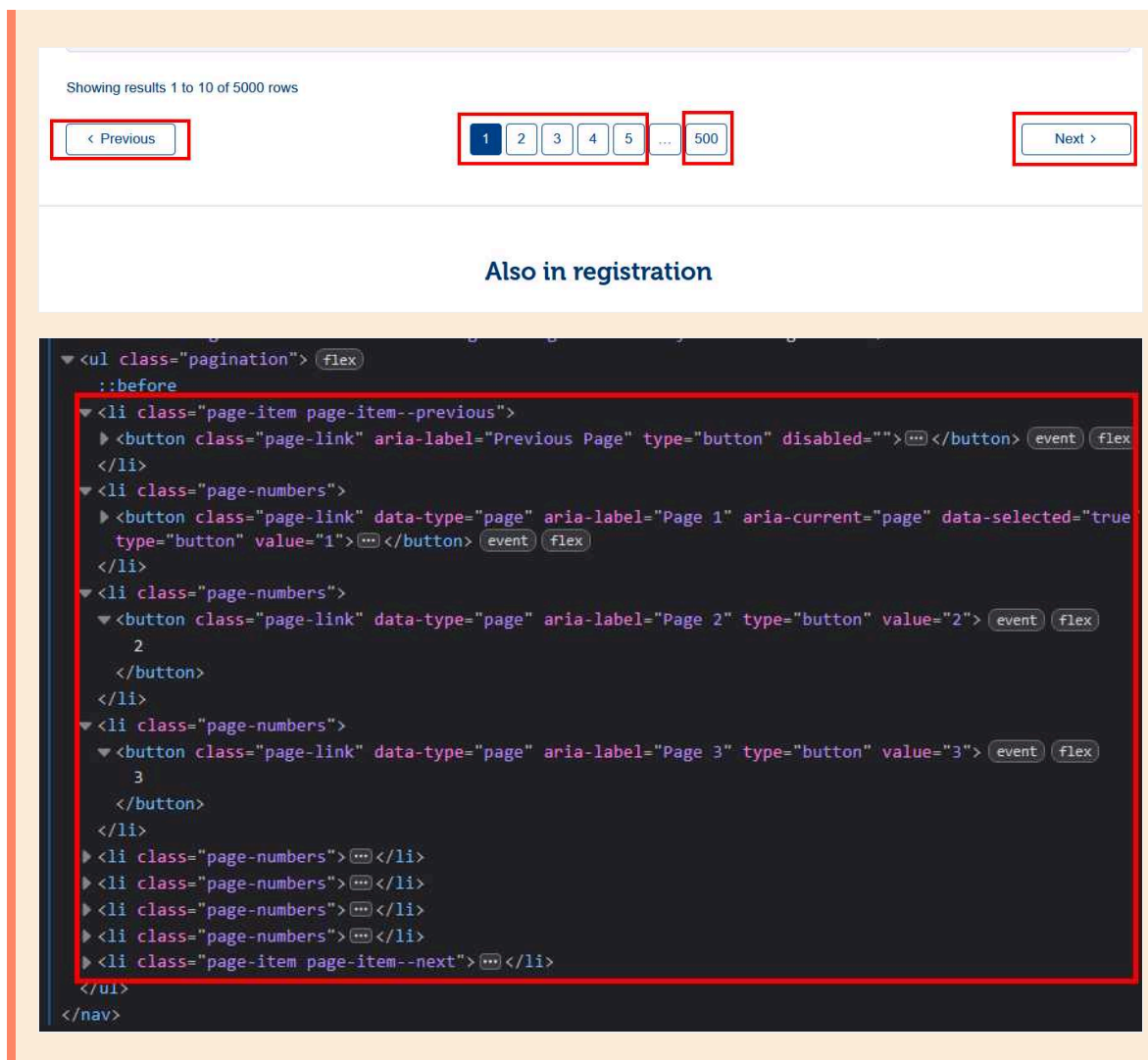
Anchor tag with no href has no role

Success Criteria	4.1.2: Name, Role, Value [2.0 A]
Devices	None
Pages Affected	None
Users Affected	Vision
Component	Link
ADR Type	Web Audit

Recheck comment 27/06/25

This issue is fixed.

On pages 3 and 9, on desktop and iOS, the pagination controls, such as “Previous” (when enabled), “2”, and “Next” (when enabled), are appropriately marked up as buttons.



What happened

Interactive elements need a programmatically determined name, role, and value. These properties must also be accurate and appropriate.

There are anchor tags with no `href` attribute, which means that they are technically not determined as links. This is because the [HTML specification determines anchor tags without href attributes as being placeholder elements](#). That is, rather than being true links.

This may particularly disadvantage users that interface with the accessibility properties of elements, such as screen reader and voice recognition users. That the link role is not programmatically determined may confuse or mislead users.

Where it happened

This can be found on pages 3 and 9, on desktop and iOS, with the pagination links such as “Previous” (when enabled), “2”, and “Next” (when enabled). The links do not have a `href` attribute defined. As such, the link role is not defined.

Note that the “...” element within the pagination is not included in this issue as it is non-interactive.

Showing results 1 to 10 of 100 rows

```

<li class="page-numbers">
  <a class="page-link" tabindex="0">
    2
  </a>
</li>
  
```

Steps to reproduce

1. Observe the anchor tags.
2. Inspect the elements and review the DOM.
3. Observe that the links do not have a `href` attribute.
4. Observe that the link role is not otherwise determined with `role="link"`.

Recommendation

Make sure that interactive elements have an appropriate role. In this case, make sure that anchor tags have `href` attributes that point to their destination. When anchor tags do not have a `href` attributes, browsers do not define them as links. For example:

```
<a href="https://www.example.com">Example Homepage</a>
```

If this is not possible, you can manually add the link role with `role="link"`.

That said, you may wish to convert the element into a button. If the element performs an action, rather than loading a new page or linking elsewhere on the current page, it is more semantically correct to have these as buttons. This is not a strict WCAG requirement, but it will improve the usability of the components. For example:

```
<button>Action</button>
<span tabindex="0" role="button">Action</span>
```

For more information, see [Yale - Web Accessibility - Links](#), [University of Washington - Links and buttons](#), [A11y Project - Valid and Accessible Links](#). As well, for guidance on reviewing accessibility properties, see [Chrome - Accessibility properties](#), [Edge - Accessibility Pane](#), and [Firefox - Accessibility Inspector](#).

[Return to priority table](#)

Additional Issues by Priority

These are issues do not fail the conformance standard. They do not require resolving for the project to be conformant with the standard.

[Skip additional issue priority breakdown](#)

Note that the **additional** and **language** issues listed in the table below are not included in the final count.



High priority (0 issues still open)

Can't use: some users are excluded from using part of the product.
 Specific user group(s) are excluded from using part of the product. For example, widgets cannot be opened with keyboard or critical visual information is not communicated by screen readers.

No issues at this priority level.

Medium priority (0 issues still open)

Causing problems: some users will experience problems using the product.
 Specific user group(s) will experience problems but they are not prevented from using the product.
 Nonetheless, they may find accessing and operating content frustrating, confusing, or otherwise challenging. For example, expanding and collapsing elements do not communicate this behaviour to screen reader users.

Issue	Success criteria Users affected	Status
ADR_22749-33 Text is truncated at default size	Additional Issue  	RESOLVED

Low priority (0 issues still open)

Minor and cosmetic issues: something is wrong, but it won't affect users too much.
 Specific user group(s) will experience minor problems. They are not excluded from the product and won't face major obstacles. Nonetheless, these issues will make interaction bothersome and the product seem unpolished. For example, a small number of cosmetic images with redundant alt text.
 Note that our priorities are context-sensitive. For example, if a page is swamped with occurrences of the same low priority issue, it may be bumped to medium priority.

No issues at this priority level.

Breakdown of Additional issues

ADR_22749-33 **MEDIUM PRIORITY** **RESOLVED**

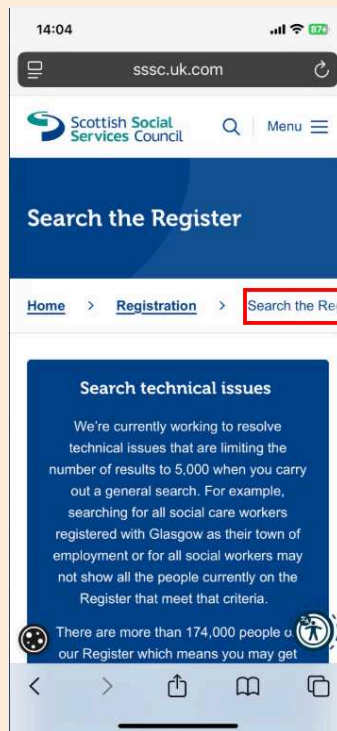
Text is truncated at default size

Success Criteria	Additional Issue
Devices	None
Pages Affected	None
Users Affected	Vision, Cognitive
Component	Plain Text
ADR Type	Web Audit

Recheck comment 27/06/25

This issue is fixed.

On page 3 on iOS, the breadcrumb region now works like a carousel, and the off-screen text can be scrolled onto the screen at the default text size and zoom levels.



What happened

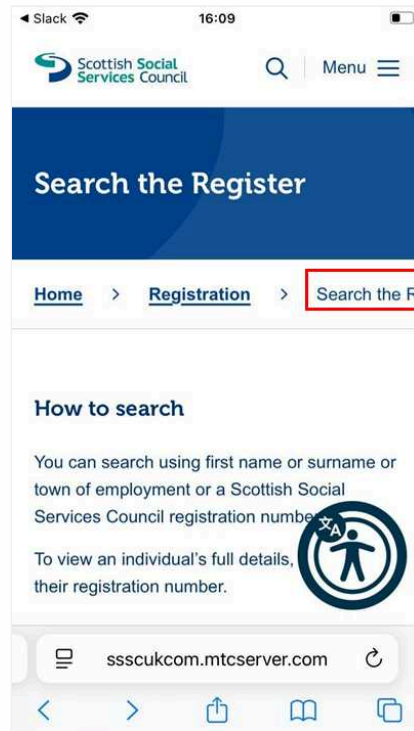
Users must be able to access content and functionality when the page is viewed with default text size and zoom settings. There should be no significant content or functionality that is partially unavailable.

There is text that is truncated by default.

This has been raised as a Medium Priority Additional Issue as it affects all users with missing text that may block understanding.

Where it happened

This can be seen on page 3 on iOS with the breadcrumb navigation text “Search the Register”. This text is partially cut off at default text size and zoom levels.



Steps to reproduce

1. Navigate onto the page with default text size and zoom settings. (Browser window 1280 x 1024px for desktop).
2. Observe that there is significant text that is truncated and unavailable to all users.

Recommendation

Make sure that all significant text is readable, and there is no loss of important content or functionality at default zoom.

Text truncation issues can be avoided by ensuring that all containers can scale dynamically to avoid text getting cut off. Use relative units for sizing, such as %, **em** or **rem** to allow elements to adjust to different screen sizes, resolutions, or zoom levels.

Additionally, ensure that text elements have enough space within their containers by adjusting container sizes or the **padding**, **margin** or **line-height** values for the elements. If you have dynamic or user-generated content that is getting cut off, provide mechanisms to control the text length that is generated to ensure it can be displayed fully.

For more information, see [CSS Tricks - Guide to Flexbox](#), [MDN - Basic Concepts of Flexbox](#), [CUBE CSS](#), [MDN - Relative Units](#), and [CSS Tricks - font-size](#).

[Return to additional issue priority table](#)

Links and references

These resources are complementary to the recommendations given in individual issues above. They can be useful for debugging, understanding how to retest issues yourselves, and develop new features in an accessible fashion from the start.

Understanding WCAG

- [WCAG Quick Reference](#)
- [Web Content Accessibility Guidelines — for People Who Haven't Read Them](#) on 24Ways
- [The full WCAG specification](#)

Accessibility resources and tutorials

- [W3C Web Accessibility Tutorials](#) for structure, images, tables and forms
- [The A11Y project](#): A community-driven effort to make web accessibility easier
- [A11Y Style Guide](#) for developers
- [Accessibility Developer Guide](#)
- Checking colour contrast: [Contrast-Ratio.com](#) (online), [Colorable](#) (online), [Accessible Color Matrix](#) (online), [Colour Contrast Analyser](#) (Windows), [Contrast](#) (macOS), [Stark](#) (Sketch plugin)

ARIA resources

- [WAI-ARIA Authoring Practices](#), your first point of call for checklists and examples of accessible components
- [Inclusive Components](#), in-depth resources for complex patterns
- [ARIA Browser support matrix](#) on CanIUse

For native app development

- [Apple Developer documentation for Accessibility](#)
- [Android accessibility overview](#)