

City of Charlotte

Accessibility Assessment – Analysis of Findings

Jesus Lardizabal
Stephen Fisher

Provided by Knowbility, Inc.
www.knowbility.org

CONFIDENTIAL

Date: June 28, 2019

Executive Summary

Knowbility reviewed the City of Charlotte website and pages selected and approved by Altura and the City of Charlotte to determine if these materials meet the needs of their visitors with disabilities. To make that determination, Knowbility tested against the Web Content Accessibility Guidelines (WCAG), Version 2.1, Levels A and AA. WCAG is the global standard, developed by member of the World Wide Web Consortium (W3C), the standards makers for the web. Knowbility tested using a wide variety of browsers, devices, and assistive technologies - including screen readers. We determined that these pages contain significant barriers that could prevent equal access to information and function by people with disabilities. Our findings are arranged by issue and each issue references the applicable WCAG guideline(s) which are included as links in Appendix 1 of this document. In conclusion, we provide a more thorough discussion of the issues and detailed guidance for how to remove the accessibility barriers. We look forward to discussing our findings with you and appreciate the opportunity to do this important work with you.

Some areas of particular concern include poor color contrast throughout the entire site, poor or extraneous use of ARIA, problematic keyboard interactions with controls and links, and issues with poor visible focus or hover styles. There were also multiple issues with form labels and error messaging. Dynamic content is not announced for screen reader users. The date picker and Live Chat widgets are not accessible and present multiple issues in and of themselves.

Table of Contents

Executive Summary.....	2
Table of Contents.....	2
Scope.....	7
Approach and testing tools used	8
Technical evaluation	8
Functional evaluation	8
Tools.....	8
Disclaimer.....	8
Detailed issues list.....	9
Duplicate ID generated by JavaScript (CLT-1)	9
No label for input (CLT-2).....	12
When JavaScript is disabled, form does not submit query (CLT-3)	14
Unnecessary attributes used on input (CLT-4)	16
No error messages for required fields (CLT-5).....	18
Insufficient color contrast for icons (CLT-6).....	20
Insufficient color contrast on link focus (CLT-7)	21

Insufficient border color contrast on focus (CLT-8).....	23
Insufficient underline color contrast on link focus (CLT-9).....	25
Insufficient color contrast for content over background images (CLT-10).....	27
Color alone used on link hover (CLT-11).....	29
Insufficient color contrast on link focus (CLT-12)	31
Poor visible focus on links (CLT-14)	33
Insufficient border color contrast on focus (CLT-15).....	37
No visible focus on links (CLT-16)	39
Incorrect ARIA attributes on site navigation (CLT-17)	42
Buttons have no discernible text (CLT-18).....	44
Images of text (CLT-19)	46
Images missing alt text (CLT-20)	48
Repetitive alt text on decorative images (CLT-21).....	50
No visible focus on drop down menus (CLT-22)	52
Title attribute used on links (CLT-23).....	54
Image links open in new window without warning visitors (CLT-24).....	58
Insufficient color contrast for text (CLT-25).....	61
Duplicate IDs (CLT-26)	64
No visual indicator for link hover (CLT-27)	68
Translate widget overlay modal does not trap keyboard focus (CLT-28).....	70
Tabular data is inaccessible (CLT-29)	72
Link text doesn't clearly explain the function (CLT-30)	77
Unnecessary aria-label attribute on links (CLT-31).....	79
Links use the title attribute (CLT-32)	81
Text-based controls interfere with user agent text resizing (CLT-33)	85
Inputs without labels (CLT-34)	87
Focusable elements are not visible on screen (CLT-35)	89
Error messages not programmatically tied to form input (CLT-37)	91
Insufficient color contrast on navigation menu icon (CLT-38)	93
Disallowed ARIA attributes used on elements (CLT-39)	95
Carousel is not keyboard accessible (CLT-40).....	97

No clear way to pause or stop carousel (CLT-41)	98
Insufficient color contrast on carousel controls (CLT-42).....	99
Sub menus are problematic (CLT-43)	101
Insufficient color contrast on link banner (CLT-44)	103
Print link opens new window (CLT-45)	104
Unordered list contains no list items, only a submit button (CLT-46).....	106
Tabbed interface is not keyboard accessible (CLT-47)	108
Page title is insufficient (CLT-48).....	110
Duplicate content generated by JavaScript (CLT-49).....	112
Text in uppercase (CLT-50)	114
Zooming/scaling disabled in viewport meta tag (CLT-51)	116
ARIA attributes referencing ID that doesn't exist (CLT-52)	117
Checkboxes used instead of radio buttons (CLT-53)	119
Error messages announced one at a time (CLT-54).....	120
Datepicker not accessible (CLT-55).....	122
Empty heading (CLT-57).....	124
Unordered list used with no list items (CLT-58)	126
Visible focus removed on some input elements for Firefox/Mozilla browsers (CLT-59)	128
Text resize widget out of visible focus order (CLT-60)	130
Copy-and-paste share URL's purpose not announced (CLT-61)	132
Multiple H1s and other heading issues (CLT-62)	134
SVG images not marked with role="img" (CLT-63).....	138
Skip to form link does not work (CLT-64)	140
Content not marked as a list (CLT-65)	142
Errors in form submission not announced or displayed clearly (CLT-68).....	144
Required fields are only indicted by an asterisk (CLT-69)	146
File picker not keyboard accessible (CLT-70).....	148
Incorrect ARIA roles (CLT-71).....	149
Inline validation indicates error on field as soon as focused (CLT-72)	150
Redundant alt text with adjacent text on page (CLT-73).....	151
Additional information not programmatically available (CLT-74).....	153

Duplicate label text for different elements (CLT-75)	155
Insufficient color contrast for links (CLT-76)	157
Use of tabindex="-1" (CLT-77)	158
Purpose of select box unclear (CLT-78)	160
Positive tabindex values used (CLT-79)	162
HTML table used for layout (CLT-81)	164
Navigation overlay modal does not trap keyboard focus (CLT-82)	168
Label refers to id that doesn't exist (CLT-83)	170
Image from CSS rules without alt text (CLT-84)	171
Board specific question(s) break flow of answering (CLT-85)	172
Local links not working (CLT-86)	173
Can't submit with only the keyboard (CLT-87)	174
Vertical bar character is announced with title (CLT-88)	175
List of error messages displayed at the bottom of the page (CLT-89)	176
Content is clipped at 200% (CLT-90)	178
Error message list not linked to specific erroneous fields (CLT-91)	180
Unclear error messages (CLT-92)	182
New slides are not announced by screen readers (CLT-93)	183
Incorrect use of navigation role (CLT-94)	184
Role="main" placed on wrong area (CLT-95)	186
Frames have no titles and wrong DOCTYPE (CLT-96)	188
Tab order is different from the expected visual order (CLT-97)	190
Modal window does not trap focus (CLT-98)	191
Changes in page not announced (CLT-99)	192
Insufficient title on iframe (CLT-100)	194
Job listings table lacking scope attribute (CLT-101)	196
No visible focus on button (CLT-102)	198
Live chat modal does not trap keyboard focus (CLT-103)	201
Favorite link takes unregistered users away from page (CLT-104)	203
Live chat dialog is not announced by screen readers (CLT-105)	204
Live chat modal is not keyboard accessible (CLT-106)	206

Keyboard focus is not set to required field (CLT-115)	208
Low contrast on selected option text (CLT-118).....	209
Items are not fully announced by screen readers (CLT-119).....	210
Low contrast on text (CLT-120).....	212
No form instructions provided (CLT-121)	213
Error messages not programmatically associated with input fields (CLT-122)	215
Link is missing href attribute and not focusable (CLT-123)	217
Incomplete error messages (CLT-124).....	219
No text alternative on font icon on link (CLT-125)	221
Required form fields are not programmatically required (CLT-126).....	222
Low contrast on inactive date text (CLT-127)	225
No labels for input fields (CLT-128)	226
Link used instead of button (CLT-129).....	228
Low contrast on focus style (CLT-130).....	230
Live chat interactions not announced to screen readers (CLT-131).....	231
Removing item is not announced by the screen reader (CLT-132)	233
SVG decorative images announced by screen reader (CLT-133).....	234
Live chat session ends without warning (CLT-134).....	236
Bugs.....	237
Non required field gives error message that it is required (CLT-36)	237
Extraneous markup (CLT-66)	238
Interactive map isn't loading (CLT-67)	239
Conclusion.....	240
Contact Information.....	240
Appendix 1: Web Content Accessibility Guidelines	242

Scope

- 001 - charlottenc.gov - Header/Footer
- 002 - charlottenc.gov - Homepage
- 003 - charlottenc.gov - Services - CharMeck 311 (drop downs)
- 004 - charlottenc.gov - Grave Spaces Evergreen
- 005 - charlottenc.gov - Bicyclist Report on Aggressive Drivers
- 006 - charlottenc.gov - Report Broken Parking Meter
- 007 - charlottenc.gov - TRASH - Schedule a Bulky Item Pickup
- 008 - charlottenc.gov - Serve on Advisory Boards - Apply for a Board
- 009 - charlottenc.gov - HR - Jobs - See Available Jobs
- 010 - charlottenc.gov - Charlotte Future (carousel and interactive map)

Approach and testing tools used

We completed a technical and functional evaluation of representative pages and/or components of the City of Charlotte website, as selected by Altura and the City of Charlotte staff for compliance against WCAG 2.1 and documented our findings. By combining functional and technical testing, we ensured a comprehensive review when testing for accessibility.

Technical evaluation

By "technical evaluation" we mean the conformance assessment to the Standards for all applicable success criteria in the context of the audited web page(s). No automated tests were run.

Functional evaluation

By "functional evaluation" we mean the use of computer adaptive technology such as voice synthesizers, screen readers, and magnification software, to verify that the results interpreted by these tools corresponds to the result observed during a visual and technical validation of the web page.

Tools

- VoiceOver, Jaws, NVDA
- WebAim color contrast checker
- Firefox web developer toolbar
- Dragon Naturally Speaking
- Windows High Contrast mode
- Browsers
- Firefox
- Safari
- Chrome
- Internet Explorer
- Edge

Disclaimer

This audit was conducted between June 24 and 28, 2019 on pages and components included in the scope. Some success criteria may be open to interpretation due to the supporting information provided in the official guidelines. The team erred on the more restrictive interpretation where more than one interpretation is possible.

Detailed issues list

Duplicate ID generated by JavaScript (CLT-1)

Components	001 - charlottenc.gov - Header/Footer
Affected community	Vision, Motor
WCAG Success Criterion	4.1.1 Parsing (A)
WCAG Level	A
Priority	Medium

Description

There are several duplicate IDs in the global search form. This appears to be tied to the JavaScript in place, which appears to generate redundant code for the search — as well as the "placeholder" text noted in [🔗 CLT-2 Open](#)

Duplicate ID values could lead assistive technologies, such as screen readers or voice-to-text applications, to parse the information improperly or to be unable to translate the information correctly.

Current code

Inline code

```
<div webpartid="00000000-0000-0000-0000-000000000000" haspers="true"
id="WebPartWPQ1" width="100%" class="ms-WPBody " onlyformepart="true"
allowdelete="false" style=""><div
componentid="ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr"
id="ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr"><div id="SearchBox"
name="Control"><div class="ms-srch-sb ms-srch-sb-border"
id="ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr_sboxdiv"><input
type="text" value="Search..." maxlength="2048" accesskey="S" title="Search..."
id="ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr_sbox" autocomplete="off"
autocorrect="off" onkeypress="EnsureScriptFunc('Search.ClientControls.js', 'Srch.U',
function() {if (Srch.U.isEnterKey(String.fromCharCode(event.keyCode)))
{$find('ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr').search($get('ctl00
_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr_sbox').value);return
Srch.U.cancelEvent(event);}})"
onkeydown="EnsureScriptFunc('Search.ClientControls.js', 'Srch.U', function() {var
ctl =
$find('ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr');ctl.activateDefault
QuerySuggestionBehavior();})" onfocus="EnsureScriptFunc('Search.ClientControls.js',
'Srch.U', function() {var ctl =
$find('ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr');ctl.hidePrompt();ct
l.setBorder(true);})" onblur="EnsureScriptFunc('Search.ClientControls.js', 'Srch.U',
function() {var ctl =
$find('ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr'); if (ctl){
```

```

ctl.showPrompt(); ctl.setBorder(false);}})" class="ms-textSmall ms-srch-sb-prompt
ms-helperText"><a title="Search" class="ms-srch-sb-searchLink"
id="ctl100_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr_SearchLink"
onclick="EnsureScriptFunc('Search.ClientControls.js', 'Srch.U', function()
{$find('ctl100_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr').search($get('ctl100
_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr_sbox').value);})"
href="javascript: {}"></a><div class="ms-
qSuggest-container ms-shadow" id="AutoCompContainer"><div
id="ctl100_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr_AutoCompList"></div></di
v></div></div></div><noscript><div
id="ctl100_PlaceHolderSearchArea_SearchBoxScriptWebPart1_noscript">It looks like your
browser does not have JavaScript enabled. Please turn on JavaScript and try
again.</div></noscript><div
id="ctl100_PlaceHolderSearchArea_SearchBoxScriptWebPart1">

</div><div class="ms-clear"></div></div>

```

Generated code (via JavaScript)

```

<div id="MSOZoneCell_WebPartWPQ2" class="s4-wpccell-plain ms-webpartzone-cell ms-
webpart-cell-vertical ms-fullWidth ">
    <div class="ms-webpart-chrome ms-webpart-chrome-vertical ms-webpart-
chrome-fullWidth ">
        <div webpartid="b23a0ca3-27ee-413a-afff-c6e4def63c16"
haspers="false" id="WebPartWPQ2" width="100%" class="ms-WPBody " allowdelete="false"
style=""><div style="display: none;"></div><div
componentid="ctl100_SPWebPartManager1_g_b23a0ca3_27ee_413a_afff_c6e4def63c16_csr"
id="ctl100_SPWebPartManager1_g_b23a0ca3_27ee_413a_afff_c6e4def63c16_csr">
    <div id="SearchBox" name="Control" class="ms-floatLeft">
        <div class="ms-srch-
sbLarge ms-srch-sb-border"
id="ctl100_SPWebPartManager1_g_b23a0ca3_27ee_413a_afff_c6e4def63c16_csr_sboxdiv">
            <input type="text" value="" maxlength="2048" accesskey="S" title="Search..."
id="ctl100_SPWebPartManager1_g_b23a0ca3_27ee_413a_afff_c6e4def63c16_csr_sbox"
autocomplete="off" autocorrect="off" onkeypress="if
(Srch.U.isEnterKey(String.fromCharCode(event.keyCode))) {
$getClientControl(this).search(this.value);return Srch.U.cancelEvent(event); }"
onkeydown="var ctl =
$getClientControl(this);ctl.activateDefaultQuerySuggestionBehavior();" onfocus="var
ctl = $getClientControl(this);ctl.hidePrompt();ctl.setBorder(true);" onblur="var ctl
= $getClientControl(this);ctl.showPrompt();ctl.setBorder(false);" class="ms-
textLarge ms-srch-sbLarge-fullWidth ms-srch-sb-prompt ms-helperText">
            <a title="Search" class="ms-srch-sb-searchLink"
id="ctl100_SPWebPartManager1_g_b23a0ca3_27ee_413a_afff_c6e4def63c16_csr_SearchLink"
onclick="$getClientControl(this).search($get('ctl100_SPWebPartManager1_g_b23a0ca3_27e
e_413a_afff_c6e4def63c16_csr_sbox').value);" href="javascript: {}">
            
            </a>
            <div
class="ms-qSuggest-container ms-shadow" id="AutoCompContainer">
                <div
id="ctl100_SPWebPartManager1_g_b23a0ca3_27ee_413a_afff_c6e4def63c16_csr_AutoCompList"
></div>
                </div>
                </div>
                </div>
                <div
id="SearchOptions">
                </div>
                </div><noscript><div
id="ctl100_SPWebPartManager1_g_b23a0ca3_27ee_413a_afff_c6e4def63c16_noscript">It
looks like your browser does not have JavaScript enabled. Please turn on JavaScript
and try again.</div></noscript><div
id="ctl100_SPWebPartManager1_g_b23a0ca3_27ee_413a_afff_c6e4def63c16">

```

```
        </div><div class="ms-clear"></div></div>  
    </div>  
</div>
```

Solution

Remove the JavaScript that generates redundant code, so that all `id` values are unique.
If that isn't possible, ensure JavaScript generates unique `id` values.

Linked issues

- CLT-2

Screenshot(s)

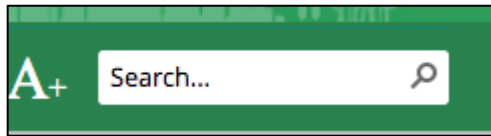


Figure 1a: - Screenshot of element with a duplicate ID

No label for input (CLT-2)

Components	001 - charlottenc.gov - Header/Footer
Affected community	Vision, Cognitive, Motor
WCAG Success Criterion	3.3.2 Labels or Instructions (A)
WCAG Level	A
Priority	Medium

Description

input field is missing a corresponding <label>. Instead, JavaScript is used to generate text from either the value or title attribute of the <input> to provide information about the purpose of the field (see [👤 CLT-4 Open](#)).

label elements natively provide information about the purpose of the field and do not have the same accessibility issues JavaScript-based solutions present (see [👤 CLT-3 Open](#)).

Current code

```
<input type="text" value="Search..." maxlength="2048" accesskey="S"
title="Search..." id="ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr_sbox"
autocomplete="off" autocorrect="off"
onkeypress="EnsureScriptFunc('Search.ClientControls.js', 'Srch.U', function() {if
(Srch.U.isEnterKey(String.fromCharCode(event.keyCode)))
{$find('ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr').search($get('ctl00
_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr_sbox').value);return
Srch.U.cancelEvent(event);}})"
onkeydown="EnsureScriptFunc('Search.ClientControls.js', 'Srch.U', function() {var
ctl =
$find('ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr');ctl.activateDefault
QuerySuggestionBehavior();})" onfocus="EnsureScriptFunc('Search.ClientControls.js',
'Srch.U', function() {var ctl =
$find('ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr');ctl.hidePrompt();ct
l.setBorder(true);})" onblur="EnsureScriptFunc('Search.ClientControls.js', 'Srch.U',
function() {var ctl =
$find('ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr'); if (ctl){
ctl.showPrompt(); ctl.setBorder(false);}})" class="ms-textSmall ms-srch-sb-prompt
ms-helperText">
```

Solution

Add a <label> using the for attribute with the input's ID as the value. If the design requires <label> be visually-hidden, ensure that it is still available for screen readers by using the correct CSS (e.g .sr-only).

Consider also removing the JavaScript that generates the "Search..." text and instead rely on the newly-added `<label>` to describe the purpose of the field. This will allow you to remove the unnecessary `title` and `value` attributes — see [🔗 CLT-4 Open](#) .

Recommended code

```
<label for="ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr_sbox"><span
class="sr-only">Search Site</search></label>
<input type="text" maxlength="2048" accesskey="S"
id="ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr_sbox" autocomplete="off"
autocorrect="off" class="ms-textSmall ms-srch-sb-prompt ms-helperText">

.sr-only {
  border:0 none;
  clip:rect(0px,0px,0px,0px);
  height:1px;
  margin:-1px;
  overflow:hidden;
  padding:0;
  position:absolute;
  width:1px
}
```

Linked issues

- CLT-1
- CLT-3
- CLT-4

Screenshot(s)

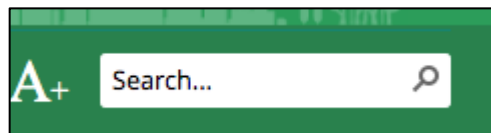


Figure 2a: - Screenshot of form input without label

When JavaScript is disabled, form does not submit query (CLT-3)

Components	001 - charlottenc.gov - Header/Footer
Affected community	Vision, Cognitive
WCAG Success Criterion	3.2.2 On Input (A)
WCAG Level	A
Priority	Medium

Description

When JavaScript is disabled, the global search form doesn't execute a query on submit. This appears due to the fact that there is no `<button>` or `<input>` used to submit. Instead the form relies on a link `onclick` event, which relies on JavaScript.

These issues affect any visitor, regardless of ability or device, who may have JavaScript disabled — or be on a slow network where downloads are limited. For these visitors, they will be unable to search the site. the `onclick` event may also not be recognized on some devices that rely on keyboard or keyboard-like devices such as switches.

Current code

```
<input type="text" value="Search..." maxlength="2048" accesskey="S"
title="Search..." id="ctl100_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr_sbox"
autocomplete="off" autocorrect="off"
onkeypress="EnsureScriptFunc('Search.ClientControls.js', 'Srch.U', function() {if
(Srch.U.isEnterKey(String.fromCharCode(event.keyCode)))
{$find('ctl100_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr').search($get('ctl100
_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr_sbox').value);return
Srch.U.cancelEvent(event);}})"
onkeydown="EnsureScriptFunc('Search.ClientControls.js', 'Srch.U', function() {var
ctl =
$find('ctl100_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr');ctl.activateDefault
QuerySuggestionBehavior();})" onfocus="EnsureScriptFunc('Search.ClientControls.js',
'Srch.U', function() {var ctl =
$find('ctl100_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr');ctl.hidePrompt();ct
l.setBorder(true);})" onblur="EnsureScriptFunc('Search.ClientControls.js', 'Srch.U',
function() {var ctl =
$find('ctl100_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr'); if (ctl){
ctl.showPrompt(); ctl.setBorder(false);}})" class="ms-textSmall "><a title="Search"
class="ms-srch-sb-searchLink"
id="ctl100_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr_SearchLink"
onclick="EnsureScriptFunc('Search.ClientControls.js', 'Srch.U', function()
{$find('ctl100_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr').search($get('ctl100
_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr_sbox').value);})"
href="javascript: {}"></a>
```

Solution

Add an `<input>` or `<button>` to submit the form and remove the `<a>` that relies on JavaScript `onclick`.

If you want to display a magnifying glass icon for the submit, use a CSS background image. You can also visually hide the `<button>` "Submit" text with CSS, while keeping it available for screen-reader users.

Recommended code

```
<input type="text" maxlength="2048" accesskey="S"
id="ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr_sbox" autocomplete="off"
autocorrect="off" class="ms-textSmall ">
<button class="search-submit"><span class="sr-only">Submit</span></button>

.search-submit {
    background: url(/images/magnifying-glass.png) no-repeat 0 0;
}

.sr-only {
    border:0 none;
    clip:rect(0px,0px,0px,0px);
    height:1px;
    margin:-1px;
    overflow:hidden;
    padding:0;
    position:absolute;
    width:1px
}
```

Linked issues

- CLT-2
- CLT-4

Screenshot(s)

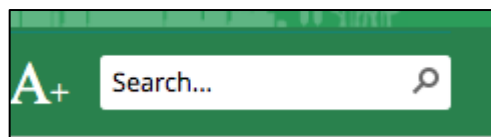



Figure 3a: - Screenshot of form that doesn't work without JavaScript


Unnecessary attributes used on input (CLT-4)

Components	001 - charlottenc.gov - Header/Footer
Affected community	Vision
WCAG Success Criterion	1.3.1 Info and Relationships (A), 4.1.2 Name, Role, Value (A)
WCAG Level	A
Priority	Medium

Description

The `title` attributed is applied to the search `<input>`. The `title` attribute doesn't have consistent support in screen readers. It may or may not be announced. In some cases, it may override the actual content. The content in a `title` is also not available to keyboard-only users.

The `value` attribute is also used, with a default value (`value="Search..."`). When JavaScript is disabled ( [CLT-3](#) Open) and a visitor focuses on that field, they must first delete the default value before proceeding with entering their search terms.

Further, one or both of these attributes appear to be used by JavaScript to present a type of "placeholder" text to provide visitors information about the field purpose (see  [CLT-2](#) Open).

Current code

```
<input type="text" value="Search..." maxlength="2048" accesskey="S"
title="Search..." id="ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr_sbox"
autocomplete="off" autocorrect="off"
onkeypress="EnsureScriptFunc('Search.ClientControls.js', 'Srch.U', function() {if
(Srch.U.isEnterKey(String.fromCharCode(event.keyCode)))
{$find('ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr').search($get('ctl00
_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr_sbox').value);return
Srch.U.cancelEvent(event);}})"
onkeydown="EnsureScriptFunc('Search.ClientControls.js', 'Srch.U', function() {var
ctl =
$find('ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr');ctl.activateDefault
QuerySuggestionBehavior();})" onfocus="EnsureScriptFunc('Search.ClientControls.js',
'Srch.U', function() {var ctl =
$find('ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr');ctl.hidePrompt();ct
l.setBorder(true);})" onblur="EnsureScriptFunc('Search.ClientControls.js', 'Srch.U',
function() {var ctl =
$find('ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr'); if (ctl){
ctl.showPrompt(); ctl.setBorder(false);}})" class="ms-textSmall ">
```


Solution

Remove the `title` and `value` attributes, as well as the JavaScript that appears to rely on one or both for the "placeholder" text.

Recommended code

```
<input type="text" maxlength="2048" accesskey="S"  
id="ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr_sbox" autocomplete="off"  
autocorrect="off" class="ms-textSmall ">
```

Linked issues

- CLT-2
- CLT-3

Screenshot(s)

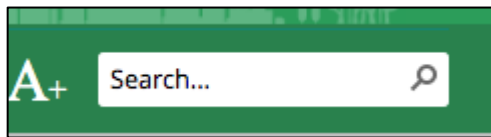


Figure 4a: - Screenshot of input with unnecessary attributes

No error messages for required fields (CLT-5)

Components	001 - charlottenc.gov - Header/Footer
Affected community	Vision, Cognitive
WCAG Success Criterion	3.3.1 Error Identification (A)
WCAG Level	A
Priority	Medium

Description

Visitors are unable to submit a search query without entering a search term. However, there is no cue for any visitors that the field is required before submitting, and there is no error message after.

This affects all users regardless of device or ability, who may be confused why the submit didn't work.

Current code

```
<input type="text" value="Search..." maxlength="2048" accesskey="S"
title="Search..." id="ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr_sbox"
autocomplete="off" autocorrect="off"
onkeypress="EnsureScriptFunc('Search.ClientControls.js', 'Srch.U', function() {if
(Srch.U.isEnterKey(String.fromCharCode(event.keyCode)))
{$find('ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr').search($get('ctl00
_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr_sbox').value);return
Srch.U.cancelEvent(event);}})"
onkeydown="EnsureScriptFunc('Search.ClientControls.js', 'Srch.U', function() {var
ctl =
$find('ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr');ctl.activateDefault
QuerySuggestionBehavior();})" onfocus="EnsureScriptFunc('Search.ClientControls.js',
'Srch.U', function() {var ctl =
$find('ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr');ctl.hidePrompt();ct
l.setBorder(true);})" onblur="EnsureScriptFunc('Search.ClientControls.js', 'Srch.U',
function() {var ctl =
$find('ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr'); if (ctl){
ctl.showPrompt(); ctl.setBorder(false);}})" class="ms-textSmall ms-srch-sb-prompt
ms-helperText">
```

Solution

Adding the boolean `required` attribute to the `input` field will provide native, client-side validation that ensures all visitors get a visual error message that the field is required. This attribute will also provide screen-reader users necessary context to know the field is required *before* submitting, as well as announcing the error message.

For keyboard users, sighted or not, the attribute will ensure focus returns to the `input` field if an error is submitted.

Recommended code

```
<input type="text" maxlength="2048" accesskey="S"  
id="ctl00_PlaceHolderSearchArea_SearchBoxScriptWebPart1_csr_sbox" autocomplete="off"  
autocorrect="off" class="ms-textSmall ms-srch-sb-prompt ms-helperText" required>
```

Screenshot(s)

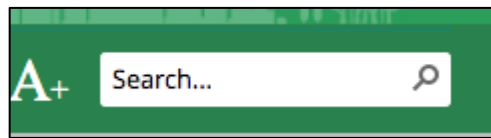


Figure 5a: - Screenshot of required field

Insufficient color contrast for icons (CLT-6)

Components	001 - charlottenc.gov - Header/Footer
Affected community	Vision
WCAG Success Criterion	1.4.3 Contrast (Minimum) (AA)
WCAG Level	AA
Priority	High

Description

For the social icons in the footer, the foreground color is #FFFFFF and the background color is #ADADAD. This is a color contrast ratio of 2.24:1, which makes it difficult for people with low vision to read the button text.

Minimum contrast should be 4.5:1.

Solution

Ensure the color contrast between the foreground and background colors on icons is at least 4.5:1.

Screenshot(s)




Figure 6a: - Screenshot of icons with poor color contrast

Insufficient color contrast on link focus (CLT-7)

Components	001 - charlottenc.gov - Header/Footer
Affected community	Vision, Cognitive, Motor
WCAG Success Criterion	1.4.3 Contrast (Minimum) (AA)
WCAG Level	AA
Priority	High

Description

When navigating through a page by using the tab key, sighted keyboard users can see their position on a web page in the form of a so-called focus outline or focus ring. This focus also provides sighted keyboard users with a cue that the element is interactive (e.g. a link or a button).

This site uses custom CSS that removes this default focus outline (see  [CLT-14](#) Open) and instead changes the link color to #BB1122. When used with the background color #007D3F, the color contrast ratio is 1.24:1. This makes it difficult for people with low vision to perceive the link focus.

Current code

```
a:hover, a:active, a:focus {  
    color: #b12;  
}  
  
a:focus {  
    outline: none;  
}
```

Solution

Ensure the color contrast between the link foreground and background colors is at least 4.5:1 at all link states.

Also consider allowing the default focus outline ( [CLT-14](#) Open).

Screenshot(s)

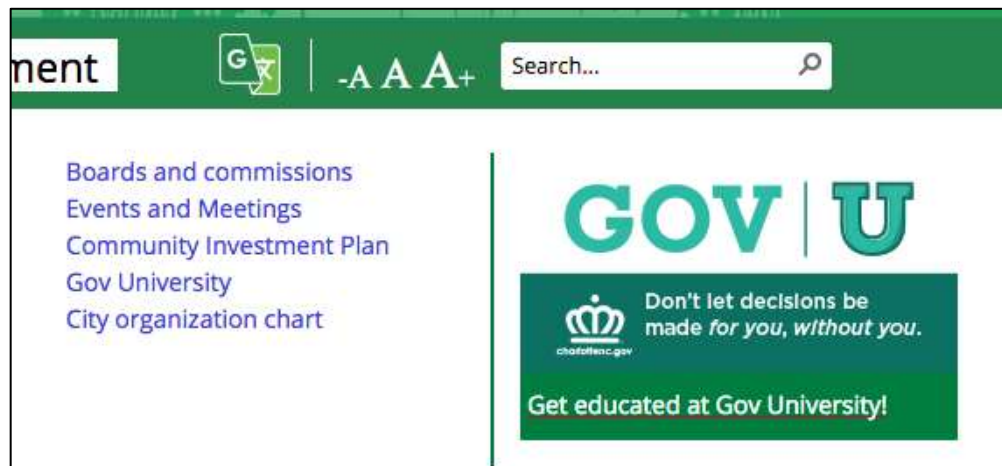



Figure 7a: - Screenshot of links with poor color contrast

Insufficient border color contrast on focus (CLT-8)

Components	001 - charlottenc.gov - Header/Footer
Affected community	Vision, Cognitive, Motor
WCAG Success Criterion	1.4.3 Contrast (Minimum) (AA)
WCAG Level	AA
Priority	High

Description

When navigating through a page by using the tab key, sighted keyboard users can see their position on a web page in the form of a so-called focus outline or focus ring. This focus also provides sighted keyboard users with a cue that the element is interactive (e.g. a link or a button).

This site uses custom CSS that removes this default focus outline (see  [CLT-14](#) Open) and instead applies a border that is #0072C6. When used with the background color #24824B, the color contrast ratio is 1.03:1.

This makes it difficult for people with low vision to perceive the link focus.

Current code

```
a:focus {
    outline: none;
}

a.goog-te-menu-value:focus {
    border: solid 2px #0072c6;
}

#universalnav {
    margin-bottom: 0;
    background: #24824b;
    border: none;
    margin-left: auto;
    margin-right: auto;
    width: 100%;
    max-width: 125rem;
    line-height: 5px;
}
```

Solution

Ensure the color contrast between border and background colors is at least 4.5:1 on focus styles.

Also consider allowing the default focus outline ( [CLT-14](#) Open).

Screenshot(s)




Figure 8a: - Screenshot of poor color contrast on focus style

Insufficient underline color contrast on link focus (CLT-9)

Components	002 - charlottenc.gov - Homepage
Affected community	Vision, Cognitive, Motor
WCAG Success Criterion	1.4.3 Contrast (Minimum) (AA)
WCAG Level	AA
Priority	High

Description

When navigating through a page by using the tab key, sighted keyboard users can see their position on a web page in the form of a so-called focus outline or focus ring. This focus also provides sighted keyboard users with a cue that the element is interactive (e.g. a link or a button).

This site uses custom CSS that removes this default focus outline (see  [CLT-14](#) Open) and relies on a link color change — for hover state, as well.

The underline color of #BB1122 with the background color #078673 results in a 1.1:1 color contrast ratio. This makes it difficult for people with low vision to perceive the link focus.

Current code

```
a:hover, a:active, a:focus {  
    color: #b12;  
}  
  
a:focus {  
    outline: none;  
}
```

Solution

Ensure the color contrast between the link foreground and background colors is at least 4.5:1 at all link states.

Also consider allowing the default focus outline ( [CLT-14](#) Open).

Linked issues

- CLT-14

Screenshot(s)



Figure 9a: - Screenshot of poor color contrast on focus style

Insufficient color contrast for content over background images (CLT-10)

Components	002 - charlottenc.gov - Homepage
Affected community	Vision
WCAG Success Criterion	1.4.3 Contrast (Minimum) (AA)
WCAG Level	AA
Priority	High


Description

For content that appears over background images — text *and* linked images — the color contrast varies. This includes the contrast with foreground colors of link states, such as the "offsite link" icon that appears due to CSS generated content.

Because the contrast is inconsistent, it can make it difficult for people with low vision to read text or perceive the "offsite link" warning.

Solution

Ensure the color contrast between foreground and background colors (or images) is at least 4.5:1 — including across all link states.

Also consider allowing the default focus outline ( [CLT-14](#) Open).

Linked issues

- CLT-14

Screenshot(s)



Figure 10a: - Screenshot of poor color contrast for text

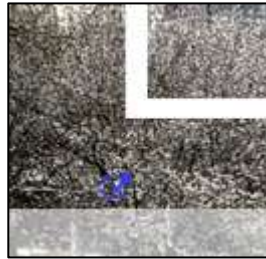


Figure 10b: - Screenshot of poor color contrast for text



Figure 10c: - Screenshot of poor color contrast for text

Color alone used on link hover (CLT-11)

Components	001 - charlottenc.gov - Header/Footer
Affected community	Vision
WCAG Success Criterion	1.4.1 Use of Color (A), 1.4.3 Contrast (Minimum) (AA)
WCAG Level	A
Priority	High

Description

On hover, the "font resize" links are only indicated with a text color change. This is because of custom CSS that removes the default underline for links.

Further, the color change is from a foreground color of #FFFFFF to #6FBE44. With a background color of #24824B, the color contrast is insufficient at 2.08:1.

Both of these issues make it difficult for people with low vision to perceive the link or its hover state.

Current code

```
div.tool-text a:hover {  
    color: #6fbe44;  
    text-decoration: none;  
}
```

Solution

Don't rely on color alone to provide visitors the visual cue that something is a link — at all links states. Remove the `text-decoration: none;` rule, so that the default underline appears on hover. Further, be sure that where you *do* use color for links (or links states), the color contrast is at least 4.5:1.

Also consider keeping the default focus `outline` for links (see  [CLT-14](#) Open).

Linked issues

- CLT-14

Screenshot(s)




Figure 11a: - Screenshot of elements identified by color alongw

Insufficient color contrast on link focus (CLT-12)

Components	002 - charlottenc.gov - Homepage, 010 - charlottenc.gov - Charlotte Future (carousel and interactive map)
Affected community	Vision, Cognitive, Motor
WCAG Success Criterion	1.4.3 Contrast (Minimum) (AA)
WCAG Level	AA
Priority	High

Description

When navigating through a page by using the tab key, sighted keyboard users can see their position on a web page in the form of a so-called focus outline or focus ring. This focus also provides sighted keyboard users with a cue that the element is interactive (e.g. a link or a button).

This site uses custom CSS that removes this default focus outline (see  [CLT-14](#) Open). For the calendar links, the CSS changes the focus color to #BB1122. With the background color #0071BC, the color contrast ratio is 1.27:1.

This makes it difficult for people with low vision to perceive the link focus.

Current code

```
a:hover, a:active, a:focus {  
    color: #b12;  
}  
  
a:focus {  
    outline: none;  
}
```

Solution

Ensure the color contrast between the link foreground and background colors is at least 4.5:1 at all link states.

Also consider allowing the default focus outline ( [CLT-14](#) Open).

Linked issues

- CLT-14

Screenshot(s)



Figure 12a: - Screenshot of poor color contrast on link focus

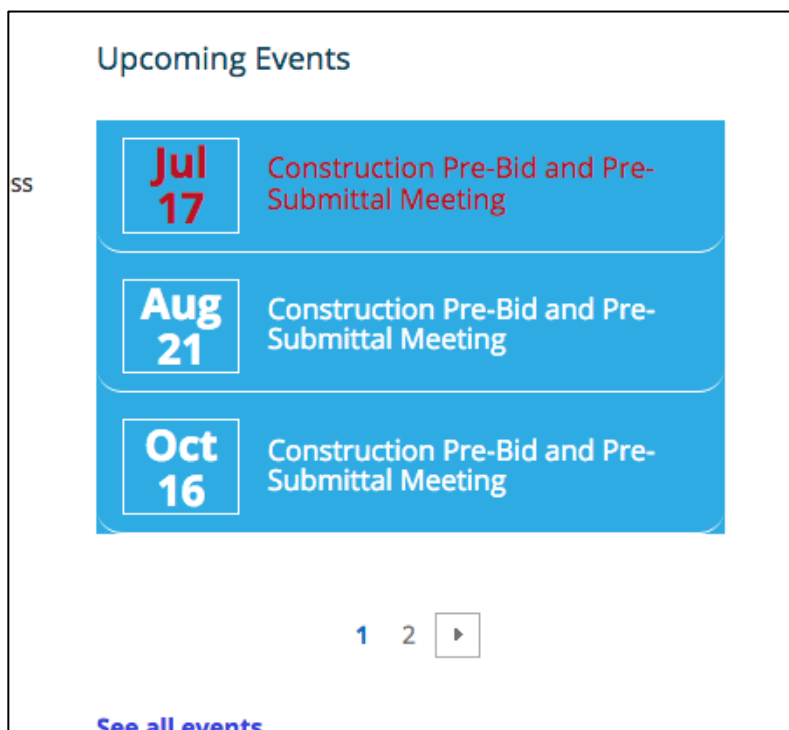






Figure 12b: - Screenshot of poor color contrast on link focus

Poor visible focus on links (CLT-14)

Components	001 - charlottenc.gov - Header/Footer, 002 - charlottenc.gov - Homepage, 010 - charlottenc.gov - Charlotte Future (carousel and interactive map)
Affected community	Vision, Cognitive, Motor
WCAG Success Criterion	2.1.1 Keyboard (A), 2.4.7 Focus Visible (AA)
WCAG Level	AA
Priority	Medium

Description

When navigating through a page by using the tab key, sighted keyboard users can see their position on a web page in the form of a so-called focus outline or focus ring. This focus also provides sighted keyboard users with a cue that the element is interactive (e.g. a link or a button).

This site uses custom CSS that removes this default focus outline ( [CLT-14](#) Open), instead relying on a subtle underline that is not immediately obvious. Further, some of the focus colors do not have sufficient contrast with the background colors (see  [CLT-9](#) Open  [CLT-10](#) Open  [CLT-12](#) Open).

Poor visible focus causes confusion, because sighted keyboard-only users may struggle to know where the focus is at on the page as they are navigating it. This could also cause confusion for sighted screen-reader users (e.g. someone with low vision or with dyslexia) who navigate with the keyboard.

Solution

Avoid removing the default focus `outline` for links ( [CLT-14](#) Open).

If the default browser focus style isn't used, provide a clearly visible focus style for interactive elements — this includes ensuring minimum color contrast of 4.5:1.

Recommended code

```
a:hover, a:active, a:focus {
  color: #b12;
}

a:focus {
  outline: none;
}
```

Linked issues

- CLT-11
- CLT-9
- CLT-10
- CLT-12

Screenshot(s)



Figure 14a: - Screenshot of poor visible focus on links



Figure 14b: - Screenshot of poor visible focus on links



Figure 14c: - Screenshot of poor visible focus on links

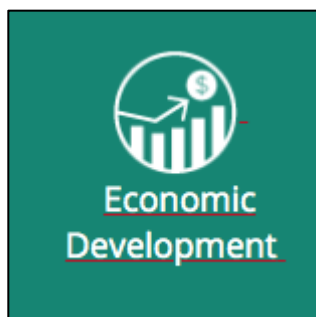


Figure 14d: - Screenshot of poor visible focus on links

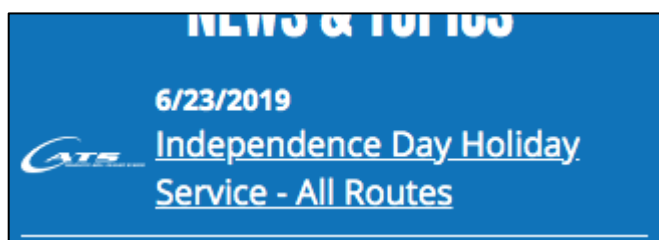


Figure 14e: - Screenshot of poor visible focus on links

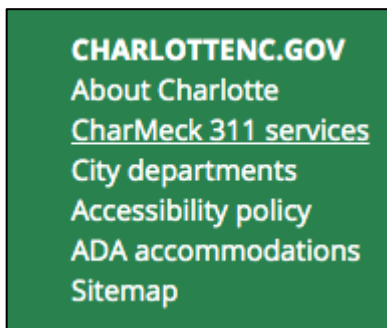


Figure 14f: - Screenshot of poor visible focus on links



Figure 14g: - Screenshot of poor visible focus on links



Figure 14h: - Screenshot of poor visible focus on links

Insufficient border color contrast on focus (CLT-15)

Components	001 - charlottenc.gov - Header/Footer
Affected community	Vision, Cognitive, Motor
WCAG Success Criterion	1.4.3 Contrast (Minimum) (AA), 2.4.7 Focus Visible (AA)
WCAG Level	AA
Priority	High

Description

When navigating through a page by using the tab key, sighted keyboard users can see their position on a web page in the form of a so-called focus outline or focus ring. This focus also provides sighted keyboard users with a cue that the element is interactive (e.g. a link or a button).

This site uses custom CSS that removes this default focus outline (see [👤 CLT-14 Open](#)) and instead applies a border that is #0072C6. When used with the background color #249E57, the color contrast ratio is 1.44:1.

This makes it difficult for people with low vision to perceive the link focus.

Current code

```
a:focus {
    outline: none;
}

.header-graphic-link:focus {
    border: 2px solid #0072c6;
}

.header-graphic-bg {
    background-color: #249e57 !important;
    height: 50px;
    display: block;
}
```

Solution

Ensure the color contrast between border and background colors is at least 4.5:1 on focus styles.

Also consider allowing the default focus outline ([👤 CLT-14 Open](#)).

Screenshot(s)




Figure 15a: - Screenshot of poor color contrast on focus style

No visible focus on links (CLT-16)

Components	002 - charlottenc.gov - Homepage, 003 - charlottenc.gov - Services - CharMeck 311 (drop downs), 004 - charlottenc.gov - Grave Spaces Evergreen, 010 - charlottenc.gov - Charlotte Future (carousel and interactive map)
Affected community	Vision, Cognitive, Motor
WCAG Success Criterion	2.1.1 Keyboard (A), 2.4.7 Focus Visible (AA)
WCAG Level	AA
Priority	High

Description

When navigating through a page by using the tab key, sighted keyboard users can see their position on a web page in the form of a so-called focus outline or focus ring. This focus also provides sighted keyboard users with a cue that the element is interactive (e.g. a link or a button).


This site uses custom CSS that removes this default focus outline (see  [CLT-14](#) Open). As such, some links do not display visible focus when navigating with a keyboard.

No visible focus causes confusion, because sighted keyboard-only users may not know where they are on the page, in general, or what link they are on, specifically. This could also cause confusion for sighted screen-reader users (e.g. someone with low vision or with dyslexia) who navigate with the keyboard.

Current code

```
a:focus {  
    outline: none;  
}
```

Solution

Avoid removing the default focus `outline` for links ( [CLT-14](#) Open). If the default browser focus style isn't used, provide a clearly visible focus style for interactive elements.

Linked issues

- CLT-22

Screenshot(s)



Figure 16a: - Screenshot of element without visible focus



Figure 16b: - Screenshot of element without visible focus



Figure 16c: - Screenshot of element without visible focus



Figure 16d: - Screenshot of element without visible focus



Figure 16e: - Screenshot of element without visible focus



Figure 16f: - Screenshot of element without visible focus



Figure 16g: - Screenshot of element without visible focus

Incorrect ARIA attributes on site navigation (CLT-17)

Components	001 - charlottenc.gov - Header/Footer
Affected community	Vision, Motor
WCAG Success Criterion	3.2.2 On Input (A), 4.1.2 Name, Role, Value (A)
WCAG Level	A
Priority	Medium

Description

The ARIA roles in the site navigation drop-down are used incorrectly. The `<div>` element has a `role="menu"` attribute, however, the links do not have the proper ARIA roles as required (`role="menuitem"`).

Screen-reader users, as well as sighted keyboard-only users, may expect to be able to navigate menus [with the following patterns](#):

- Down Arrow: When focus is in a menu, moves focus to the next item, optionally wrapping from the last to the first.
- Up Arrow: When focus is in a menu, moves focus to the previous item, optionally wrapping from the first to the last.
- Home: If arrow key wrapping is not supported, moves focus to the first item in the current menu or menubar.
- End: If arrow key wrapping is not supported, moves focus to the last item in the current menu or menubar.
- Tab: Moves focus to the next element in the tab sequence, and if the item that had focus is not in a menubar, closes its menu and all open parent menu containers.
- Shift + Tab: Moves focus to the previous element in the tab sequence, and if the item that had focus is not in a menubar, closes its menu and all open parent menu containers.

However, it is also worth noting that ARIA menus are for application menus, such as the menu bar for Word on a desktop computer, *not website navigation*.

Solution

Don't apply ARIA menu roles to site navigation, which means removing instances of `role="menu"`, as well as the trigger links that include ARIA properties for [menu buttons](#):

- `role="button"`

- aria-expanded
- aria-haspopup

Instead, use CSS and/or JavaScript to support the keyboard navigation necessary to expand and contract the sub-navigation — a "fly out," which works well for both mouse and keyboard users. Refer to this tutorial for details: <https://www.w3.org/WAI/tutorials/menus/flyout/>

If it's not possible to change from a menu to a fly out, follow the proper methodology for menus: <https://www.w3.org/TR/wai-aria-practices-1.1/#menu>

Screenshot(s)



Figure 17a: - Screenshot of elements with incorrect ARIA attributes

Buttons have no discernible text (CLT-18)

Components	001 - charlottenc.gov - Header/Footer, 003 - charlottenc.gov - Services - CharMeck 311 (drop downs)
Affected community	Vision
WCAG Success Criterion	4.1.2 Name, Role, Value (A)
WCAG Level	A
Priority	Medium

Description

The hamburger or mobile menu icons do not have text to indicate their respective meaning. Without that text, screen reader users are not able to discern the purpose of a button.

Current code

```
<button id="topnav-toggle" type="button" class="navbar-toggle" data-  
toggle="collapse" data-target="#topnav-collapse" aria-expanded="false" aria-  
controls="topnav-collapse">  
  <span class="icon-bar"></span>  
  <span class="icon-bar"></span>  
  <span class="icon-bar"></span>  
</button>
```

Solution

Ensure that each `button` element provides discernible text that describes its meaning. For example, one method is to use of `aria-label` attribute with a meaningful value.

Recommended code

```
<button aria-label="Menu" id="topnav-toggle" type="button" class="navbar-toggle" data-  
toggle="collapse" data-target="#topnav-collapse" aria-expanded="false" aria-  
controls="topnav-collapse">  
  <span class="icon-bar"></span>  
  <span class="icon-bar"></span>  
  <span class="icon-bar"></span>  
</button>
```

Screenshot(s)




Figure 18a: - Screenshot of buttons without text

Images of text (CLT-19)

Components	002 - charlottenc.gov - Homepage, 010 - charlottenc.gov - Charlotte Future (carousel and interactive map)
Affected community	Vision
WCAG Success Criterion	1.1.1 Non-text content (A), 1.4.5 Images of Text (AA)
WCAG Level	A
Priority	Medium

Description

The main "hero" images are of text, which should be avoided because it is inaccessible to screen-reader users — especially if there isn't `alt` text ( [CLT-20](#) Open).

This is also a problem on the Charlotte Future page, where photos have text embedded.

Images with text are also problematic for visitors with low vision, who may increase the browser zoom level. Images are not resized with zoom, so they actually become *less* readable.

The only exceptions for images with text are:

- Brand logos
- If you can't achieve the effect with text (for example, presenting a particular example of typography)
- If the presentation can't be achieved with the technology used to design the website
- Purely decorative text

Solution

Replace the hero promotional images with text and style with CSS to achieve the same typographic presentation.

For the Charlotte Future images, position text over the photos with CSS, to achieve the same typographic presentation.

If that isn't possible, ensure the `<image>` has `alt` text that reflects the text in the images.

Screenshot(s)



Figure 19a: - Screenshot of images of text



Figure 19b: - Screenshot of images of text

Images missing alt text (CLT-20)

Components	002 - charlottenc.gov - Homepage
Affected community	Vision
WCAG Success Criterion	1.1.1 Non-text content (A)
WCAG Level	A
Priority	Medium

Description

Informational images do not have an `alt` attribute.

This makes it difficult for screen-reader users, who won't know if the image contains important information that they need to interact with the site.

Solution

Add descriptive `alt` text to informational images. Ensure that decorative images have an empty or null value for `alt` (e.g. `alt=""`).

Refer to WAI's alt text decision tree for guidance on when and what kind of `alt` text should be used for images <https://www.w3.org/WAI/tutorials/images/decision-tree/>

Screenshot(s)



Figure 20a: - Screenshot of image without alt text



Figure 20b: - Screenshot of image without alt text

Repetitive alt text on decorative images (CLT-21)

Components	001 - charlottenc.gov - Header/Footer
Affected community	Vision
WCAG Success Criterion	1.1.1 Non-text content (A)
WCAG Level	A
Priority	Medium

Description

The images that appear in the expanded site navigation have `alt` text that is the same as the text immediately following the images. This will create a repetitive experience for screen-reader users.

Further these are decorative images, so they don't need descriptive `alt` text.

Current code

```
<a href="https://charlottenc.gov/water"><div  
class="mega-menu-campaign-caption">Pay your water bill</div></a>
```

Solution

Because these are decorative images — they do not provide more information than is already available in text on the page — use `{{alt=""}}`.

Recommended code

```
<a href="https://charlottenc.gov/water"><div class="mega-menu-  
campaign-caption">Pay your water bill</div></a>
```

Screenshot(s)

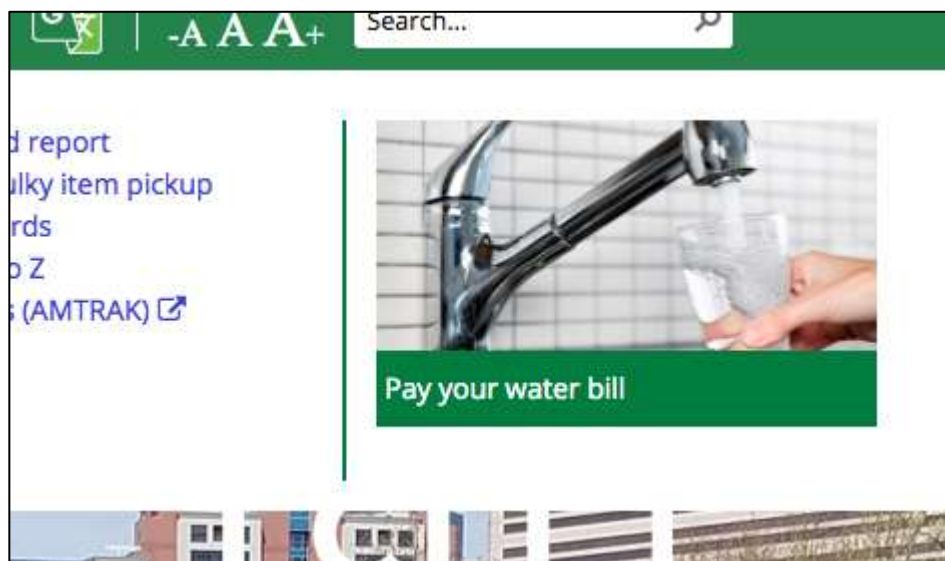


Figure 21a: - Screenshot of image with repetitive alt text

No visible focus on drop down menus (CLT-22)

Components	003 - charlottenc.gov - Services - CharMeck 311 (drop downs), 010 - charlottenc.gov - Charlotte Future (carousel and interactive map)
Affected community	Vision, Cognitive, Motor
WCAG Success Criterion	2.1.1 Keyboard (A), 2.4.7 Focus Visible (AA)
WCAG Level	A
Priority	Medium

Description

When navigating through drop down menu options with the tab key, sighted keyboard users can see their position on a web page in the form of a focus outline or focus ring. This focus also provides sighted keyboard users with a cue that the element is interactive (e.g. a link or a button).

This site uses CSS rules that removes this default focus outline on the drop down menus.

No visible focus causes confusion, because sighted keyboard-only users may not know where they are on the page, in general, or what link they are on, specifically. This could also cause confusion for sighted screen-reader users (e.g. someone with low vision or with dyslexia) who navigate with the keyboard.

Current code

```
a:focus {
    outline: none;
}
.dropdown-toggle:focus {
    outline: 0;
}
```

Solution

Avoid removing the default focus outline for links. If the default browser focus style isn't used, provide a clearly visible focus style for interactive elements.

Recommended code

```
a:focus {
    /* outline: none; */
}
.dropdown-toggle:focus {
    /* outline: 0; */
}
```

}

Linked issues

- CLT-43
- CLT-16

Screenshot(s)



Figure 22a: - Screenshot of menu with no visible focus

Title attribute used on links (CLT-23)

Components	001 - charlottenc.gov - Header/Footer, 002 - charlottenc.gov - Homepage
Affected community	Vision
WCAG Success Criterion	1.3.1 Info and Relationships (A), 4.1.2 Name, Role, Value (A)
WCAG Level	A
Priority	Low

Description

Several links use the `title` attribute.

Unfortunately, the support between browser and screen-reader combinations for the `title` attribute is inconsistent. Some screen readers will ignore the `title` altogether. Others override the link text, while others announced the `title` *in addition to* the link text.

This may cause repetitive content for screen-reader users, which creates a cumbersome experience.

Current code

Site Navigation

```
<li class="dropdown unav "><a href="#" class="dropdown-toggle" aria-expanded="false"
aria-haspopup="true" role="button" data-toggle="dropdown" aria-label="Services"
title="Services">Services</a>...</li>
<li class="dropdown unav "><a href="#" class="dropdown-toggle" aria-expanded="false"
aria-haspopup="true" role="button" data-toggle="dropdown" aria-label="Resident"
title="Resident">Resident</a>...</li>
<li class="dropdown unav "><a href="#" class="dropdown-toggle" aria-expanded="false"
aria-haspopup="true" role="button" data-toggle="dropdown" aria-label="Business"
title="Business">Business</a>...</li>
<li class="dropdown unav "><a href="#" class="dropdown-toggle" aria-expanded="false"
aria-haspopup="true" role="button" data-toggle="dropdown" aria-label="Visitor"
title="Visitor">Visitor</a>...</li>
<li class="dropdown unav "><a href="#" class="dropdown-toggle" aria-expanded="false"
aria-haspopup="true" role="button" data-toggle="dropdown" aria-label="City
Government" title="City Government">City Government</a>...</li>
```

Search Submit Link

```
<a title="Search" class="ms-srch-sb-searchLink"
id="ctl00_SPWebPartManager1_g_b23a0ca3_27ee_413a_ffff_c6e4def63c16_csr_SearchLink"
onclick="$getClientControl(this).search($get('ctl00_SPWebPartManager1_g_b23a0ca3_27e
e_413a_ffff_c6e4def63c16_csr_sbox').value);" href="javascript: {}">

</a>
```

Footer Icon Links

```

<a href="https://twitter.com/CLTgov"><div id="twittericon" class="socialicon icon-
twitter" title="Follow CharMeck on Twitter"></div></a>
<a href="https://www.facebook.com/CLTgov"><div id="facebookicon" class="socialicon
icon-facebook" title="Like Charmeck on Facebook"></div></a>
  <a href="https://www.youtube.com/user/CharlotteGOVchannel"><div id="youtubeicon"
class="socialicon icon-youtube" title="View our Youtube channel"></div></a>


```

Solution

Remove the `title` attribute from links.

For links that contain images, such as the search submit, rely on the image `alt` text to provide screen-reader users context for the link.

For links that display icons without text, such as the footer social links, add text to the link and hide it visually — while still keeping it available for screen readers to announce.

Note: This solution also removes the unnecessary `aria-label` attributed logged in  [CLT-31](#) Open

Recommended code

Site Navigation

```

<li class="dropdown unav "><a href="#" class="dropdown-toggle" aria-expanded="false"
aria-haspopup="true" role="button" data-toggle="dropdown">Services</a>...</li>
<li class="dropdown unav "><a href="#" class="dropdown-toggle" aria-expanded="false"
aria-haspopup="true" role="button" data-toggle="dropdown">Resident</a>...</li>
<li class="dropdown unav "><a href="#" class="dropdown-toggle" aria-expanded="false"
aria-haspopup="true" role="button" data-toggle="dropdown">Business</a>...</li>
<li class="dropdown unav "><a href="#" class="dropdown-toggle" aria-expanded="false"
aria-haspopup="true" role="button" data-toggle="dropdown">Visitor</a>...</li>
<li class="dropdown unav "><a href="#" class="dropdown-toggle" aria-expanded="false"
aria-haspopup="true" role="button" data-toggle="dropdown">City
Government</a>...</li>

```

Search Submit Link

```

<a class="ms-srch-sb-searchLink"
id="ctl00_SPWebPartManager1_g_b23a0ca3_27ee_413a_ffff_c6e4def63c16_csr_SearchLink"
onclick="$getClientControl(this).search($get('ctl00_SPWebPartManager1_g_b23a0ca3_27e
e_413a_ffff_c6e4def63c16_csr_sbox').value);" href="javascript: {}">

</a>

```

Footer Icon Links

```

<a href="https://twitter.com/CLTgov"><div id="twittericon" class="socialicon icon-
twitter"><span class="sr-only">Twitter</span></div></a>
<a href="https://www.facebook.com/CLTgov"><div id="facebookicon" class="socialicon
icon-facebook"><span class="sr-only">Facebook</span></div></a>
  <a href="https://www.youtube.com/user/CharlotteGOVchannel"><div id="youtubeicon"
class="socialicon icon-youtube"><span class="sr-only">YouTube</span></div></a>

```

```

.sr-only {
  border:0 none;
  clip:rect(0px,0px,0px,0px);
  height:1px;

```

```
margin:-1px;
overflow:hidden;
padding:0;
position:absolute;
width:1px
}
```

Linked issues

- CLT-31

Screenshot(s)



Figure 23a: - Screenshot of link with title attribute



Figure 23b: - Screenshot of link with title attribute

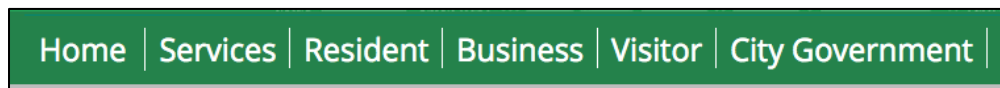


Figure 23c: - Screenshot of link with title attribute



Figure 23d: - Screenshot of link with title attribute



Figure 23e: - Screenshot of link with title attribute

Image links open in new window without warning visitors (CLT-24)

Components	002 - charlottenc.gov - Homepage, 003 - charlottenc.gov - Services - CharMeck 311 (drop downs), 010 - charlottenc.gov - Charlotte Future (carousel and interactive map)
Affected community	Vision
WCAG Success Criterion	3.2.2 On Input (A)
WCAG Level	A
Priority	High

Description

The linked images in the home hero use the `target="_blank"` attribute, forcing the link to open in a new window. There is no warning to visitors that this will happen, though there is a cue that the link destination is offsite (via CSS generated content).

This also occurs with the share links that appear on interior pages, as well as links to maps.

It can be a disorienting experience to have a new tab/window open. This is particularly problematic for screen-reader users who may try to navigate to the previous content but are unable to because it's in a different tab/window. This may also cause confusion in visitors with cognitive issues, who may not perceive that the link has opened in a new tab/window.

Current code

Image Links

```
<a href="http://www.clt250.com" target="_blank">
  
</a>
```

Text Links

```
<a href="https://maps.google.com/?q=600+East Fourth Street Room 280 Charlotte North
Carolina 28202" target="_blank"><span class="glyphicon glyphicon-map-
marker"></span>Get map</a>
```

Solution

Do not force the opening of new window. Let the user decide when, and if, a new window is required.

Should that not be possible, when dealing with linked images, add the warning to the `` descriptive `alt` text (needs to be added, per [👤 CLT-20 Open](#)).

When dealing with text links, use a foreground image with `alt` text.

Recommended code

Image Links

```
<a href="http://www.clt250.com">  
    
</a>
```

Or

```
<a href="http://www.clt250.com" target="_blank">  
    
</a>
```

Text Links

```
<a href="https://maps.google.com/?q=600+East+Fourth+Street+Room+280+Charlotte+North  
Carolina+28202" target="_blank"><span class="glyphicon glyphicon-map-  
marker"></span>Get map </a>
```

Screenshot(s)



Figure 24a: - Screenshot of link opening in a new window

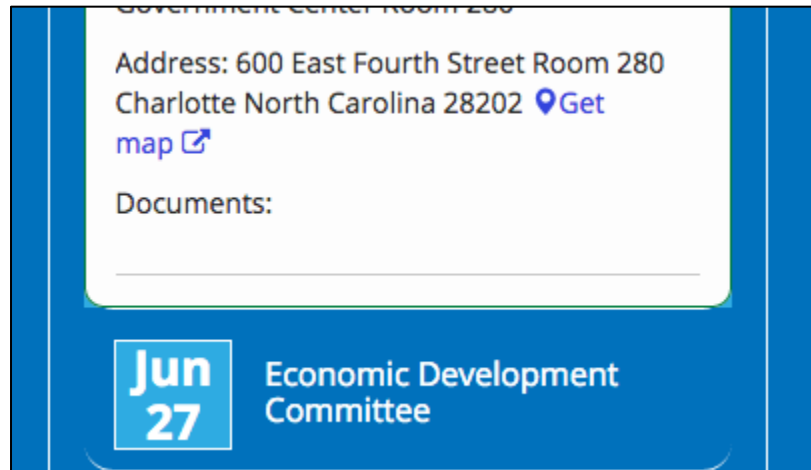


Figure 24b: - Screenshot of link opening in a new window



Figure 24c: - Screenshot of link opening in a new window

Insufficient color contrast for text (CLT-25)

Components	002 - charlottenc.gov - Homepage, 005 - charlottenc.gov - Bicyclist Report on Aggressive Drivers, 010 - charlottenc.gov - Charlotte Future (carousel and interactive map)
Affected community	Vision
WCAG Success Criterion	1.4.3 Contrast (Minimum) (AA)
WCAG Level	AA
Priority	High

Description

For the date boxes, the foreground color is #FFFFFF and the background color is #2EABE2. This is a color contrast ratio of 2.61:1.

For the File upload instructions (Bicycle Report form), the foreground text color is #9E9E9E and the background color is #EEEEEE. This combination has a color contrast ratio of 2.3:1.

For content that appears in the tabbed interface, the foreground text is the same color as the background: #FFFFFF. This is a contrast ratio of 0.

These contrast ratios are difficult for people with low vision to read the text. For a 0 ratio, no visitors — low vision or not — are able to read the text.

Solution

Ensure the color contrast between text foreground and background colors is at least 4.5:1.

Screenshot(s)

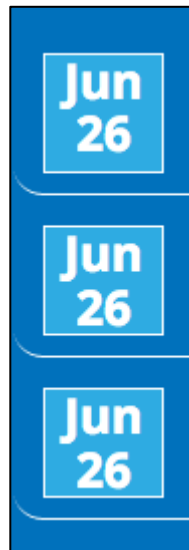


Figure 25a: - Screenshot of text with poor color contrast

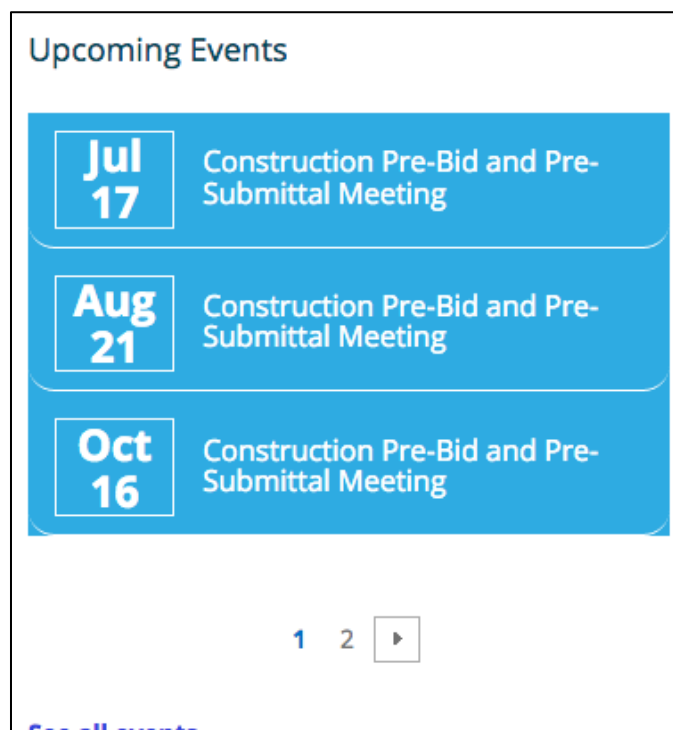


Figure 25b: - Screenshot of text with poor color contrast

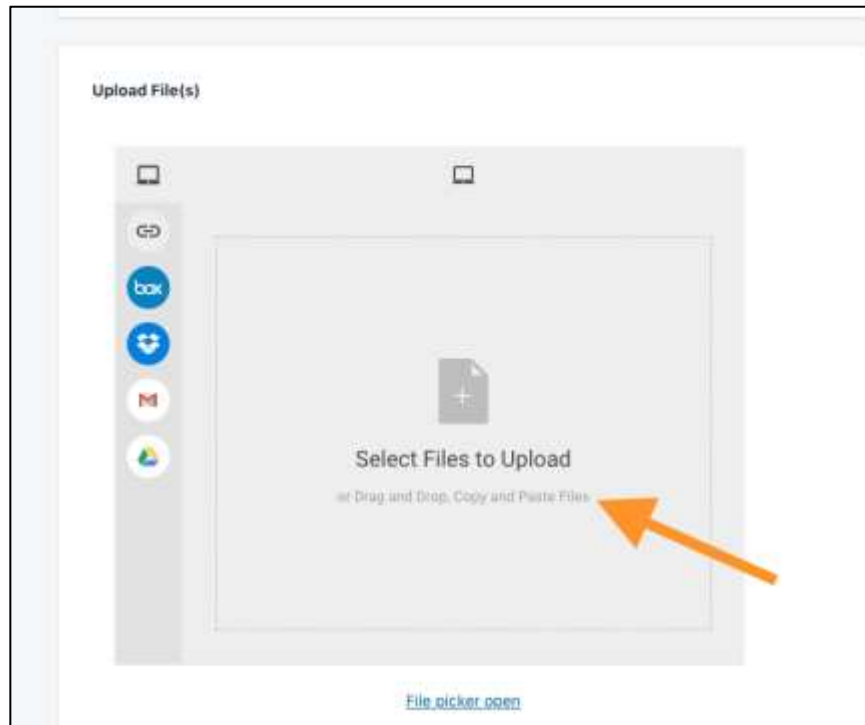


Figure 25c: - Screenshot of text with poor color contrast

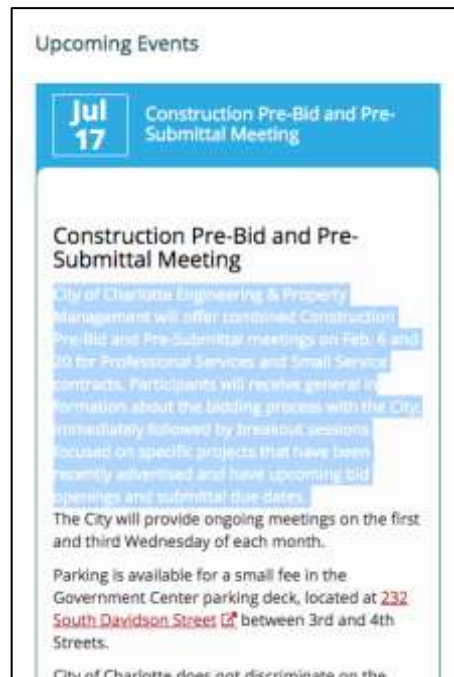


Figure 25d: - Screenshot of text with poor color contrast

Duplicate IDs (CLT-26)

Components	002 - charlottenc.gov - Homepage, 009 - charlottenc.gov - HR - Jobs - See Available Jobs
Affected community	Vision, Motor
WCAG Success Criterion	4.1.1 Parsing (A)
WCAG Level	A
Priority	Medium

Description

There are duplicate IDs in the feed wrapper.

Duplicate ID values could lead assistive technologies, such as screen readers or voice-to-text applications, to parse the information improperly or to be unable to translate the information correctly.

Current code

News & Topics

```
<div class="coc-home-webpart-wrapper coc-home-newsfeed-wrapper">
  <h2 class="coc-home-webpart-title">News & Topics</h2>
  <div class="ms-webpart-zone ms-fullWidth">
    <div id="MSOZoneCell_WebPartWPQ5" class="s4-wpcell-plain ms-webpartzone-
cell ms-webpart-cell-vertical ms-fullWidth ">
      <div class="ms-webpart-chrome ms-webpart-chrome-vertical ms-
webpart-chrome-fullWidth ">
        <div webpartid="af81146d-f6ca-4249-b02b-ad23a0de5de4"
haspers="false" id="WebPartWPQ5" width="100%" class="ms-WPBody " allowdelete="false"
style="">
          <div style="display: none;"></div>
          <div
componentid="ctl00_SPWebPartManager1_g_af81146d_f6ca_4249_b02b_ad23a0de5de4_csr"
id="ctl00_SPWebPartManager1_g_af81146d_f6ca_4249_b02b_ad23a0de5de4_csr">
            <div class="city-newsfeed-list-narrow-wrapper">
              <!--<div class="city-newsfeed-list-narrow-
title"><i class="fa fa-newspaper-o"></i> News and Topics</div-->
              <div
id="ctl00_SPWebPartManager1_g_af81146d_f6ca_4249_b02b_ad23a0de5de4_csr1_group"
name="Group" class="ms-srch-group " data-queryruleid="00000000-0000-0000-0000-
000000000000">
                <div id="GroupBorderTop"></div>
```

This Week

```
<div class="coc-home-webpart-wrapper">
  <h2 class="coc-home-webpart-title">This Week</h2>
```



```

<div class="ms-webpart-zone ms-fullWidth">
  <div id="MSOZoneCell_WebPartWPQ6" class="s4-wpcell-plain ms-webpartzone-
cell ms-webpart-cell-vertical ms-fullWidth ">
    <div class="ms-webpart-chrome ms-webpart-chrome-vertical ms-
webpart-chrome-fullWidth ">
      <div webpartid="8bde47fb-adc4-4be8-b92e-f364e91659b5"
haspers="false" id="WebPartWPQ6" width="100%" class="ms-WPBody " allowdelete="false"
style="">
        <div style="display: none;"></div>
        <div
componentid="ctl00_SPWebPartManager1_g_8bde47fb_adc4_4be8_b92e_f364e91659b5_csr"
id="ctl00_SPWebPartManager1_g_8bde47fb_adc4_4be8_b92e_f364e91659b5_csr">
          <div class="city-event-list-narrow-wrapper ms-
srch-result">
            <!--<div class="city-event-list-narrow-
title"><i class="fa fa-calendar"></i> Upcoming Events</div-->
            <div class="city-event-list-narrow-inner
charmeck-accordion-panel-group panel-group" role="tablist" aria-
multiselectable="true">
              <div
id="ctl00_SPWebPartManager1_g_8bde47fb_adc4_4be8_b92e_f364e91659b5_csr1_group"
name="Group" class="ms-srch-group " data-queryruleid="00000000-0000-0000-0000-
000000000000">
                <div
id="GroupBorderTop"></div>

```

Solution

Ensure all ID values are unique.

Recommended code

News & Topics

```

<div class="coc-home-webpart-wrapper coc-home-newsfeed-wrapper">
  <h2 class="coc-home-webpart-title">News & Topics</h2>
  <div class="ms-webpart-zone ms-fullWidth">
    <div id="MSOZoneCell_WebPartWPQ5" class="s4-wpcell-plain ms-webpartzone-
cell ms-webpart-cell-vertical ms-fullWidth ">
      <div class="ms-webpart-chrome ms-webpart-chrome-vertical ms-
webpart-chrome-fullWidth ">
        <div webpartid="af81146d-f6ca-4249-b02b-ad23a0de5de4"
haspers="false" id="WebPartWPQ5" width="100%" class="ms-WPBody " allowdelete="false"
style="">
          <div style="display: none;"></div>
          <div
componentid="ctl00_SPWebPartManager1_g_af81146d_f6ca_4249_b02b_ad23a0de5de4_csr"
id="ctl00_SPWebPartManager1_g_af81146d_f6ca_4249_b02b_ad23a0de5de4_csr">
            <div class="city-newsfeed-list-narrow-wrapper">
              <!--<div class="city-newsfeed-list-narrow-
title"><i class="fa fa-newspaper-o"></i> News and Topics</div-->
              <div
id="ctl00_SPWebPartManager1_g_af81146d_f6ca_4249_b02b_ad23a0de5de4_csr1_group"
name="Group" class="ms-srch-group " data-queryruleid="00000000-0000-0000-0000-
000000000000">

```

```
<div id="NewsBorderTop"></div>
```

This Week

```
<div class="coc-home-webpart-wrapper">
  <h2 class="coc-home-webpart-title">This Week</h2>
  <div class="ms-webpart-zone ms-fullWidth">
    <div id="MSOZoneCell_WebPartWPQ6" class="s4-wpcell-plain ms-webpartzone-
cell ms-webpart-cell-vertical ms-fullWidth ">
      <div class="ms-webpart-chrome ms-webpart-chrome-vertical ms-
webpart-chrome-fullWidth ">
        <div webpartid="8bde47fb-adc4-4be8-b92e-f364e91659b5"
haspers="false" id="WebPartWPQ6" width="100%" class="ms-WPBody " allowdelete="false"
style="">

          <div style="display: none;"></div>
          <div
componentid="ctl00_SPWebPartManager1_g_8bde47fb_adc4_4be8_b92e_f364e91659b5_csr"
id="ctl00_SPWebPartManager1_g_8bde47fb_adc4_4be8_b92e_f364e91659b5_csr">
            <div class="city-event-list-narrow-wrapper ms-
srch-result">

              <!--<div class="city-event-list-narrow-
title"><i class="fa fa-calendar"></i> Upcoming Events</div>-->
              <div class="city-event-list-narrow-inner
charmeck-accordion-panel-group panel-group" role="tablist" aria-
multiselectable="true">

                <div
id="ctl00_SPWebPartManager1_g_8bde47fb_adc4_4be8_b92e_f364e91659b5_csr1_group"
name="Group" class="ms-srch-group " data-queryruleid="00000000-0000-0000-0000-
000000000000">

                  <div
id="CalendarBorderTop"></div>
```

Screenshot(s)



Figure 26a: - Screenshot of elements with duplicate ids



Figure 26b: - Screenshot of elements with duplicate ids



Figure 26c: - Screenshot of elements with duplicate ids

No visual indicator for link hover (CLT-27)

Components	001 - charlottenc.gov - Header/Footer
Affected community	Cognitive
WCAG Success Criterion	
WCAG Level	
Priority	Medium

Description

The top-level site navigation links, which trigger the drop-down, do not have a visual indicator on hover. These links should display consistent behavior across states — the focus state displays an underline.

Most visitors expects links to display a visual indicator on hover and focus (both, often, the same). Inconsistency with this pattern could cause confusion for people with cognitive impairments.

Current code

```
.nav > li > a:focus, .nav > li > a:hover {  
    text-decoration: none;  
}
```

Solution

Apply the same underline to the hover state as appear on focus, by removing the hover pseudo-class from the CSS rule.

Note: If you remove the `.nav > li > a:focus` rule entirely, there won't be a need for the rule that adds the underline back on the focus state:

```
#universalnav .navbar-nav > li > a:focus {  
    text-decoration: underline;  
}
```

Recommended code

```
.nav > li > a:focus {  
    text-decoration: none;  
}
```

Screenshot(s)



Figure 27a: - Screenshot of elements without visual hover style



Figure 27b: - Screenshot of elements without visual hover style

Translate widget overlay modal does not trap keyboard focus (CLT-28)

Components	001 - charlottenc.gov - Header/Footer
Affected community	Vision, Cognitive, Motor
WCAG Success Criterion	1.3.2 Meaningful Sequence (A), 2.1.2 No Keyboard Trap (A), 2.4.3 Focus Order (A), 3.2.1 On Focus (A)
WCAG Level	A
Priority	High

Description

There are several issues with the keyboard accessibility of the translate widget:

- When the overlay modal is triggered, focus is not set inside the modal.
- If keyboard users navigate with the Tab key, focus leaves the modal after the last language option.
- The same thing happens if users navigate using Shift+Tab at the first language option.
- Keyboard users don't have an option to close the widget, either with a button or the escape key.

This will create problems for screen-reader, keyboard or other assistive-technology users who launch the modal but may not be able to operate it or close out of it. If these users can tab out of the modal, they are easily confused about their location within the page. Further, when the modal closes, these users expect to be at the location that triggered the modal.

Solution

1. Ensure that keyboard focus is set by default to the first element in the modal
2. Focus should be kept inside the modal until the visitor closes it
3. When the modal is closed, the focus should return to the link that triggered the modal.
4. The escape key should close the modal.

Refer to the W3C modal example for a thorough code sample and explanation on how to build accessible modals: <https://www.w3.org/TR/wai-aria-practices/examples/dialog-modal/dialog.html>

Screenshot(s)



Figure 28a: - Screenshot of modal that doesn't capture keyboard focus

Tabular data is inaccessible (CLT-29)

Components	004 - charlottenc.gov - Grave Spaces Evergreen
Affected community	Vision, Cognitive
WCAG Success Criterion	1.3.1 Info and Relationships (A), 4.1.2 Name, Role, Value (A)
WCAG Level	A
Priority	Medium

Description

There are multiple problematic issues with the tables on the page.

- Due to each table having multiple headers, it is complex and requires additional context to explain the information in the table. The current structure might be confusing to users, sighted or not, to understand the context of table data.
- Header cells are not consistently associated with the data cells, making it difficult for screen reader users to relate the information.

Current code

```
<table class="table table-hover" unselectable="on">
  <thead unselectable="on">
    <tr unselectable="on">
      <th unselectable="on">SALE OF GRAVE/ NICHE SPACE:</th>
      <th unselectable="on">
      </th>
      <th unselectable="on">COST</th>
    </tr>
  </thead>
  <tbody unselectable="on">
    <tr unselectable="on">
      <th scope="row" unselectable="on">Adult</th>
      <td unselectable="on">Grave Space</td>
      <td unselectable="on">$1200.00</td>
    </tr>
    ...
    <tr unselectable="on">
      <th scope="row" unselectable="on">Cremains</th>
      <td unselectable="on">In-Ground Space</td>
      <td unselectable="on">$600.00</td>
    </tr>
    ...
  </tbody>
</table>

<table class="table table-hover" unselectable="on">
```



```

<thead unselectable="on">
  <tr unselectable="on">
    <th unselectable="on">OPENING AND CLOSING FEES</th>
    <th unselectable="on">
</th>
    <th unselectable="on">COST</th>
  </tr>
</thead>
<tbody unselectable="on">
  <tr unselectable="on">
    <th scope="row" unselectable="on">Adult</th>
    <td unselectable="on">Weekdays</td>
    <td unselectable="on">$920.00</td>
  </tr>
  <tr unselectable="on">
    <td unselectable="on"></td>
    <td unselectable="on">Weekdays after 3:30 and Saturdays </td>
    <td unselectable="on">$1380.00</td>
  </tr>
  ...
  <tr unselectable="on">
    <th scope="row" unselectable="on">Infant/Child</th>
    <td unselectable="on">Weekdays</td>
    <td unselectable="on">$360.00</td>
  </tr>
  <tr unselectable="on">
    <td unselectable="on"></td>
    <td unselectable="on">Weekdays after 3:30 and Saturdays </td>
    <td unselectable="on">$540.00</td>
  </tr>
  ...
  <tr unselectable="on">
    <th scope="row" unselectable="on">Cremations</th>
    <td unselectable="on"> </td>
    <td unselectable="on"> </td>
  </tr>
  <tr unselectable="on">
    <td unselectable="on"> </td>
    <td unselectable="on">Weekdays</td>
    <td unselectable="on">$420.00</td>
  </tr>
  <tr unselectable="on">
    <td unselectable="on"> </td>
    <td unselectable="on">Weekdays after 3:30 and Saturdays</td>
    <td unselectable="on">$630.00</td>
  </tr>
  ...
  <tr unselectable="on">
    <th scope="row" unselectable="on">Disinterment</th>
    <td unselectable="on"></td>
    <td unselectable="on"> </td>
  </tr>
  <tr unselectable="on">
    <td unselectable="on">Adult</td>
    <td unselectable="on"></td>
    <td unselectable="on">$1800.00</td>
  </tr>

```

```
...
</tbody></table>
```

Solution

Update the table markup

- Ensure all header cells use the `th` element with the `scope` attribute.
- Associate the table headings to the cells in each respective row using the `scope="row"` attribute.
- If table header cells are for columns, use the `scope="col"` attribute.
- Use `colspan` on header cells

Refer to <https://www.w3.org/WAI/tutorials/tables/two-headers/> for details on creating an accessible data table with two headers.

Refer to <https://www.w3.org/WAI/tutorials/tables/caption-summary/> for details on the accessible implementation of the `<caption>` element.

Recommended code

```
<table class="table table-hover" unselectable="on">
  <thead unselectable="on">
    <tr unselectable="on">
      <th unselectable="on">SALE OF GRAVE/ NICHE SPACE:</th>
      <th unselectable="on">
      </th>
      <th unselectable="on">COST</th>
    </tr>
  </thead>
  <tbody unselectable="on">
    <tr unselectable="on">
      <th scope="row" unselectable="on">Adult</th>
      <td unselectable="on">Grave Space</td>
      <td unselectable="on">$1200.00</td>
    </tr>
    ...
    <tr unselectable="on">
      <th scope="row" unselectable="on">Cremains</th>
      <td unselectable="on">In-Ground Space</td>
      <td unselectable="on">$600.00</td>
    </tr>
    ...
  </tbody>
</table>

<table class="table table-hover" unselectable="on">
  <thead unselectable="on">
    <tr unselectable="on">
      <th colspan="2" unselectable="on">OPENING AND CLOSING FEES</th>
      <th unselectable="on">
```

```

        <th unselectable="on">COST</th>
    </tr>
</thead>
<tbody unselectable="on">
    <tr unselectable="on">
        <th scope="row" unselectable="on">Adult</th>
        <td unselectable="on">Weekdays</td>
        <td unselectable="on">$920.00</td>
    </tr>
    <tr unselectable="on">
        <td unselectable="on"></td>
        <td unselectable="on">Weekdays after 3:30 and Saturdays </td>
        <td unselectable="on">$1380.00</td>
    </tr>
    ...
    <tr unselectable="on">
        <th scope="row" unselectable="on">Infant/Child</th>
        <td unselectable="on">Weekdays</td>
        <td unselectable="on">$360.00</td>
    </tr>
    <tr unselectable="on">
        <td unselectable="on"></td>
        <td unselectable="on">Weekdays after 3:30 and Saturdays </td>
        <td unselectable="on">$540.00</td>
    </tr>
    ...
    <tr unselectable="on">
        <th scope="row" unselectable="on">Cremations</th>
        <td unselectable="on"> </td>
        <td unselectable="on"> </td>
    </tr>
    <tr unselectable="on">
        <td unselectable="on"> </td>
        <td unselectable="on">Weekdays</td>
        <td unselectable="on">$420.00</td>
    </tr>
    <tr unselectable="on">
        <td unselectable="on"> </td>
        <td unselectable="on">Weekdays after 3:30 and Saturdays</td>
        <td unselectable="on">$630.00</td>
    </tr>
    ...
    <tr unselectable="on">
        <th scope="row" unselectable="on">Disinterment</th>
        <td unselectable="on"></td>
        <td unselectable="on"> </td>
    </tr>
    <tr unselectable="on">
        <td unselectable="on">Adult</td>
        <td unselectable="on"></td>
        <td unselectable="on">$1800.00</td>
    </tr>
    ...
</tbody></table>

```

Screenshot(s)

SALE OF GRAVE/ NICHE SPACE:		COST
Adult	Grave Space	\$1200.00
Infant/Child	Grave Space	\$500.00
Cremains	In-Ground Space	\$600.00
	Niche-Single	\$800.00
	Niche-Double	\$1200.00
Perpetual Care Fee	(one time fee on all above sales)	\$150.00
OPENING AND CLOSING FEES		COST
Adult	Weekdays	\$920.00
	Weekdays after 3:30 and Saturdays	\$1380.00
	Sundays & Holidays	\$1725.00
	Entombment Fee	\$820.00
Infant/Child	Weekdays	\$380.00
	Weekdays after 3:30 and Saturdays	\$540.00
	Sundays & Holidays	\$675.00
Cremations	Weekdays	\$420.00
	Weekdays after 3:30 and Saturdays	\$630.00
	Sundays & Holidays	\$785.00
	Second Right of Interment	\$450.00
Disinterment		
Adult		\$1800.00
Infant/Child		\$800.00

Figure 29a: - Screenshot of tabular data

Link text doesn't clearly explain the function (CLT-30)

Components	002 - charlottenc.gov - Homepage
Affected community	Vision, Cognitive
WCAG Success Criterion	2.4.4 Link Purpose (In Context) (A)
WCAG Level	A
Priority	Medium

Description


The "font resizer" links don't provide context for screen-reader users who navigate the page by tabbing through links. Link text should clearly state what is the destination of the link. Also, because these are links that trigger JavaScript (vs. navigating to a destination), screen-reader users may be unaware these links provide functionality instead of navigation.

These links could also cause confusion for *all* visitors, especially those with cognitive disabilities, who may not understand what the letters represent.

Current code

```
<div class="tool-text">
  <a class="coc-fontresize-reduce" href="javascript:resizeText(-1);"><span
class="reduce-minus">-</span>A</a>
  <a class="coc-fontresize-normal" href="javascript:resizeText(0);">A</a>
  <a class="coc-fontresize-increase" href="javascript:resizeText(1);">A<span
class="reduce-plus">+</span></a>
</div>
```

Solution

Most site visitors who want to resize text, utilize the native functionality in browsers — whether that is changing their default font size or using the zoom. As such, consider removing the "font resizer" links entirely ( [CLT-33](#) Open).

If that isn't possible, consider:

- Changing the links to `button` elements, which provide native cues to screen-reader users about interactivity (vs. navigation with links).
- Ensure that the `button` text is descriptive of the functionality.
- If you need to visually hide `button` text, utilize CSS rules that ensure the content is still announced by screen readers.

- For the button text that doesn't offer context for screen-reader users (e.g. the letter "A" and the plus/minus signs), be sure to hide with `aria-hidden="true"`.

Recommended code

```
<div class="tool-text">
  <button class="coc-fontresize-reduce" onclick="javascript:resizeText(-
1);"><span class="sr-only">Increase font-size</span><span aria-hidden="true"><span
class="reduce-minus">-</span>A</span></button>
  <button class="coc-fontresize-normal"
onclick="javascript:resizeText(0);"><span class="sr-only">Reset font-
size</span><span aria-hidden="true">A</span></button>
  <button class="coc-fontresize-increase"
onclick="javascript:resizeText(1);"><span class="sr-only">Decrease font-
size</span><span aria-hidden="true">A<span class="reduce-
plus">+</span></span></button>
</div>

.sr-only {
  border:0 none;
  clip:rect(0px,0px,0px,0px);
  height:1px;
  margin:-1px;
  overflow:hidden;
  padding:0;
  position:absolute;
  width:1px;
}
```

Linked issues

- CLT-33

Screenshot(s)



Figure 30a: - Screenshot of unclear link text

Unnecessary aria-label attribute on links (CLT-31)

Components	001 - charlottenc.gov - Header/Footer
Affected community	Vision
WCAG Success Criterion	1.3.1 Info and Relationships (A), 4.1.2 Name, Role, Value (A)
WCAG Level	A
Priority	Medium

Description

The top-level site navigation links have the `aria-label` attribute, which is redundant to the link text. This may cause repetitive content for screen-reader users, which creates a cumbersome experience — especially when combined with the unnecessary `title` attribute noted in [🔗 CLT-23 Open](#).

Current code

```
<li class="dropdown unav "><a href="#" class="dropdown-toggle" aria-expanded="false"
aria-haspopup="true" role="button" data-toggle="dropdown" aria-label="Services"
title="Services">Services</a>...</li>
<li class="dropdown unav "><a href="#" class="dropdown-toggle" aria-expanded="false"
aria-haspopup="true" role="button" data-toggle="dropdown" aria-label="Resident"
title="Resident">Resident</a>...</li>
<li class="dropdown unav "><a href="#" class="dropdown-toggle" aria-expanded="false"
aria-haspopup="true" role="button" data-toggle="dropdown" aria-label="Business"
title="Business">Business</a>...</li>
<li class="dropdown unav "><a href="#" class="dropdown-toggle" aria-expanded="false"
aria-haspopup="true" role="button" data-toggle="dropdown" aria-label="Visitor"
title="Visitor">Visitor</a>...</li>
<li class="dropdown unav "><a href="#" class="dropdown-toggle" aria-expanded="false"
aria-haspopup="true" role="button" data-toggle="dropdown" aria-label="City
Government" title="City Government">City Government</a>...</li>
```

Solution

Remove the `aria-label` attribute from links.

Note: This solution also removes the unnecessary `title` attribute noted in [🔗 CLT-23 Open](#).

Recommended code

```
<li class="dropdown unav "><a href="#" class="dropdown-toggle" aria-expanded="false"
aria-haspopup="true" role="button" data-toggle="dropdown">Services</a>...</li>
```

```
<li class="dropdown unav "><a href="#" class="dropdown-toggle" aria-expanded="false"
aria-haspopup="true" role="button" data-toggle="dropdown">Resident</a>...</li>
<li class="dropdown unav "><a href="#" class="dropdown-toggle" aria-expanded="false"
aria-haspopup="true" role="button" data-toggle="dropdown">Business</a>...</li>
<li class="dropdown unav "><a href="#" class="dropdown-toggle" aria-expanded="false"
aria-haspopup="true" role="button" data-toggle="dropdown">Visitor</a>...</li>
<li class="dropdown unav "><a href="#" class="dropdown-toggle" aria-expanded="false"
aria-haspopup="true" role="button" data-toggle="dropdown">City
Government</a>...</li>
```

Linked issues

- CLT-23

Screenshot(s)



Figure 31a: - Screenshot of links with unnecessary attributes

Links use the title attribute (CLT-32)

Components	001 - charlottenc.gov - Header/Footer, 003 - charlottenc.gov - Services - CharMeck 311 (drop downs), 004 - charlottenc.gov - Grave Spaces Evergreen, 009 - charlottenc.gov - HR - Jobs - See Available Jobs
Affected community	Vision
WCAG Success Criterion	1.3.1 Info and Relationships (A), 4.1.2 Name, Role, Value (A)
WCAG Level	A
Priority	Low

Description

The `title` attribute has been used on some links. Support for that attribute is uneven across screen reader and browser combinations. In some cases, the title is ignored altogether. In other cases, it overrides the link text. In yet other cases, it is announced as well as the link text. It may create a cumbersome experience for screen reader users.

Current code

```
<a href="#" class="dropdown-toggle" aria-expanded="false" aria-haspopup="true"
role="button" data-toggle="dropdown" aria-label="City Government" title="City
Government">City Government</a>
```

```
<a title="Manages design and construction of City infrastructure projects, regulates
development, cares for the tree canopy and landscaping and manages real estate and
facilities portfolios" href="/engineering">Engineering & Property Management</a>
```

Solution

Remove the `title` attribute from links.

Recommended code

```
<a href="#" class="dropdown-toggle" aria-expanded="false" aria-haspopup="true"
role="button" data-toggle="dropdown" aria-label="City Government"></a>
```

```
<a href="/engineering">Engineering & Property Management</a>
```

Screenshot(s)



Figure 32a: - Screenshot of links using the title attribute



Figure 32b: - Screenshot of links using the title attribute

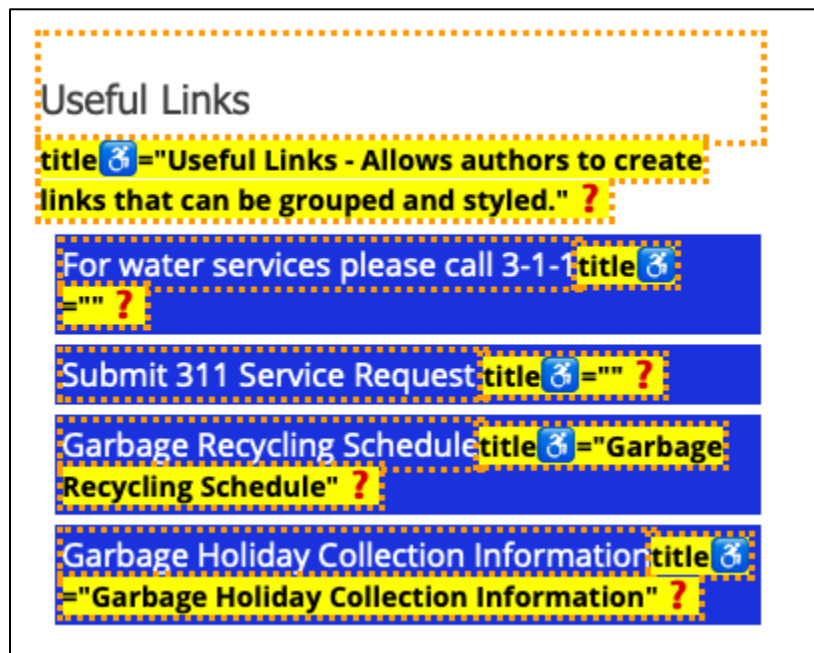


Figure 32c: - Screenshot of links using the title attribute



Figure 32d: - Screenshot of links using the title attribute



Figure 32e: - Screenshot of links using the title attribute



Figure 32f: - Screenshot of links using the title attribute

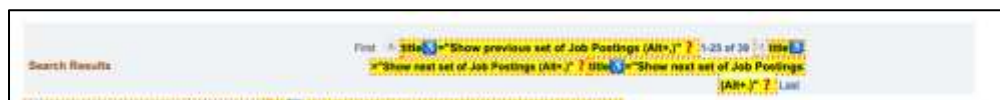


Figure 32g: - Screenshot of links using the title attribute



Figure 32h: - Screenshot of links using the title attribute



Figure 32i: - Screenshot of links using the title attribute

Text-based controls interfere with user agent text resizing (CLT-33)

Components	001 - charlottenc.gov - Header/Footer
Affected community	
WCAG Success Criterion	1.4.4 Resize Text (AA)
WCAG Level	AA
Priority	Medium

Description

The "font resizer" links do not effectively resize text, causing issues with text blocks overlaying each other, which makes the text illegible.

Also, these "font resizer" controls interfere with visitors' ability to scale content effectively. Most site visitors who want to resize text, utilize the native functionality in browsers — whether that is changing their default font size or using the zoom.

Solution

Remove the "font resizer" controls entirely.

Linked issues

- CLT-60
- CLT-30

Screenshot(s)



Figure 33a: - Screenshot of text based controls



Figure 33b: - Screenshot of resized text



Figure 33c: - Screenshot of resized text



Figure 33d: - Screenshot of resized text

Inputs without labels (CLT-34)

Components	006 - charlottenc.gov - Report Broken Parking Meter
Affected community	Vision
WCAG Success Criterion	3.3.2 Labels or Instructions (A)
WCAG Level	A
Priority	High

Description

Several `<inputs>` have no label, programmatic or otherwise. This means screen reader users may not easily know, or at all, what is the input's purpose.

Current code

```
<div class="item control-group tag-input-text " aria-required="true">
    <input name="input_T8EMno" type="text" value=""
class="validate['required'] fc-field-error" id="input_T8EMno" maxlength=""
validationregex="" placeholder="" style="width:320px;:width:320px;;" aria-
required="true">

    </div>
```

Solution

Use the `aria-label` attribute to provide a label for the elements.

Recommended code

```
<div class="item control-group tag-input-text " aria-required="true">
    <input name="input_T8EMno" type="text" value=""
class="validate['required'] fc-field-error" id="input_T8EMno" maxlength=""
validationregex="" placeholder="" style="width:320px;:width:320px;;" aria-
required="true" aria-label="Ticket Number">

    </div>
```

Screenshot(s)

If yes, please provide the ticket number:

What was the problem with the meter?

Please describe any other problems you experienced when using this meter:

Figure 34a: - Screenshot of inputs without labels

Focusable elements are not visible on screen (CLT-35)

Components	001 - charlottenc.gov - Header/Footer
Affected community	Vision, Cognitive, Motor
WCAG Success Criterion	2.1.1 Keyboard (A)
WCAG Level	A
Priority	High

Description

Starting at 170% zoom, as well as views narrower than approximately 766px, there are invisible tab stops:

- After the skip navigation link, there is an invisible tab stop before the linked logo. Nothing happens if selected.
- After the hamburger menu, there are nine invisible tab stops before the hero linked image — it appears these stops are the visibly-hidden primary navigation and site search.

This could be confusing to keyboard users, sighted or not.

Solution

Ensure the visibly-hidden navigation and search are also hidden from keyboard navigation with `display:none` — unless users trigger the navigation modal.

Screenshot(s)



Figure 35a: - Screenshot of invisible focusable elements



Figure 35b: - Screenshot of code inspector

Error messages not programmatically tied to form input (CLT-37)

Components	006 - charlottenc.gov - Report Broken Parking Meter, 008 - charlottenc.gov - Serve on Advisory Boards - Apply for a Board
Affected community	Vision
WCAG Success Criterion	
WCAG Level	
Priority	High

Description

Inline validation of errors displays a small text bubble stating the field is required. This is not programmatically tied to the input field, hence screen readers won't be able to announce this information.

Current code

```
<textarea name="textarea_Mn5k6z" class="validate['required'] fc-field-error"
id="textarea_Mn5k6z" placeholder="" type=""
style="width:320px;height:100px;;width:320px;height:100px;;" aria-
required="true"></textarea>
...
<div class="err"><p>This field is required.</p></div>
```

Solution

Use the `aria-describedby` attribute to link the field to the error message. The error message container will require a unique ID which can be referenced in the ARIA attribute.

Recommended code

```
<textarea name="textarea_Mn5k6z" class="validate['required'] fc-field-error"
id="textarea_Mn5k6z" placeholder="" type=""
style="width:320px;height:100px;;width:320px;height:100px;;" aria-required="true"
aria-describedby="err-prob-meter"></textarea>
...
<div class="err"><p id="err-prob-meter">This field is required.</p></div>
```

Screenshot(s)

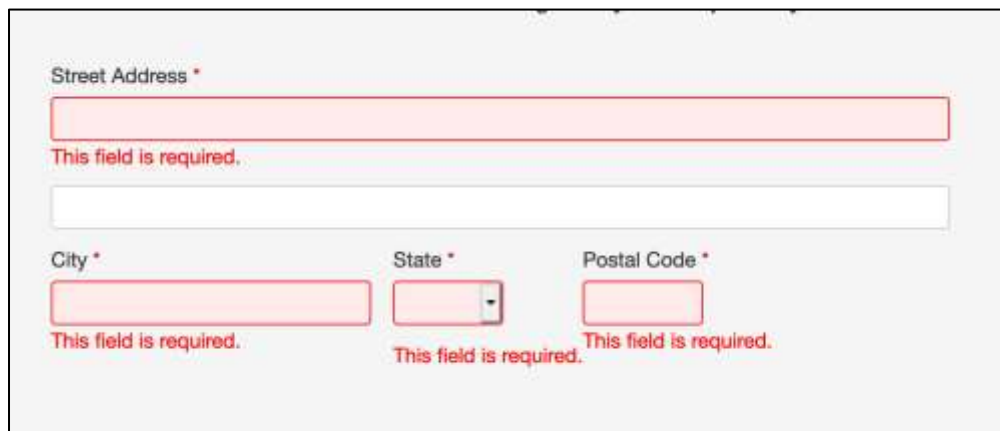


Please describe any other problems you experienced when using this meter:

This field is required.

The screenshot shows a light blue rectangular container. At the top, it contains the text "Please describe any other problems you experienced when using this meter:". Below this text is a large, empty white rectangular input field. To the right of the bottom right corner of this input field, there is a small red rectangular box with the white text "This field is required.".

Figure 37a: - Screenshot of error message



Street Address *

This field is required.

City *

This field is required.

State *

This field is required.

Postal Code *

This field is required.

The screenshot shows a light gray rectangular container. At the top, it contains the text "Street Address *". Below this text is a long, empty white rectangular input field. To the right of the bottom right corner of this input field, there is a small red rectangular box with the white text "This field is required.". Below this input field is another empty white rectangular input field. Below this second input field, there are three input fields arranged horizontally. The first is labeled "City *" and is an empty white rectangular input field with a red "This field is required." message below it. The second is labeled "State *" and is a white rectangular dropdown menu with a red "This field is required." message below it. The third is labeled "Postal Code *" and is an empty white rectangular input field with a red "This field is required." message below it.

Figure 37b: - Screenshot of error messages

Insufficient color contrast on navigation menu icon (CLT-38)

Components	001 - charlottenc.gov - Header/Footer, 010 - charlottenc.gov - Charlotte Future (carousel and interactive map)
Affected community	Vision
WCAG Success Criterion	1.4.3 Contrast (Minimum) (AA)
WCAG Level	AA
Priority	High

Description

Starting at 170% zoom, as well as views narrower than approximately 766px, the "hamburger" icon is available to trigger site navigation.

For main site navigation, the foreground color is #888888 and the background color is #24824B. This is a color contrast ratio of 1.35:1.

For interior page navigation, the foreground color is #888888 and the background color is #BDBDBD. This is a color contrast ratio of 1.88:1. On hover, the background color changes to #DDDDDD, which is a color contrast ratio of 2.6:1.

These contrast ratios are difficult for people with low vision to read the icon text.

Current code

```
.navbar-default .navbar-toggle .icon-bar {
    background-color: #888;
}

#universalnav {
    margin-bottom: 0;
    background: #24824b;
    border: none;
    margin-left: auto;
    margin-right: auto;
    width: 100%;
    max-width: 125rem;
    line-height: 5px;
}
```

Recommended code

Ensure the color contrast between the foreground and background colors on icons is at least 4.5:1.

Screenshot(s)



Figure 38a: - Screenshot of element with poor color contrast



Figure 38b: - Screenshot of element with poor color contrast

Disallowed ARIA attributes used on elements (CLT-39)

Components	005 - charlottenc.gov - Bicyclist Report on Aggressive Drivers, 006 - charlottenc.gov - Report Broken Parking Meter
Affected community	Vision
WCAG Success Criterion	4.1.2 Name, Role, Value (A)
WCAG Level	A
Priority	High

Description

The ARIA attribute `aria-required` has been applied to `<div>` element. This attribute is not allowed on that element. Using non-allowed attribute may, at best, not have any effect. It may, however, cause conflicts and decrease accessibility.

Current code

```
<div class="item control-group tag-textarea " aria-required="true">
```

Solution

Remove the `aria` attribute from the `div`. Do not use disallowed attributes on elements.

Recommended code

```
<div class="item control-group tag-textarea ">
```

Screenshot(s)

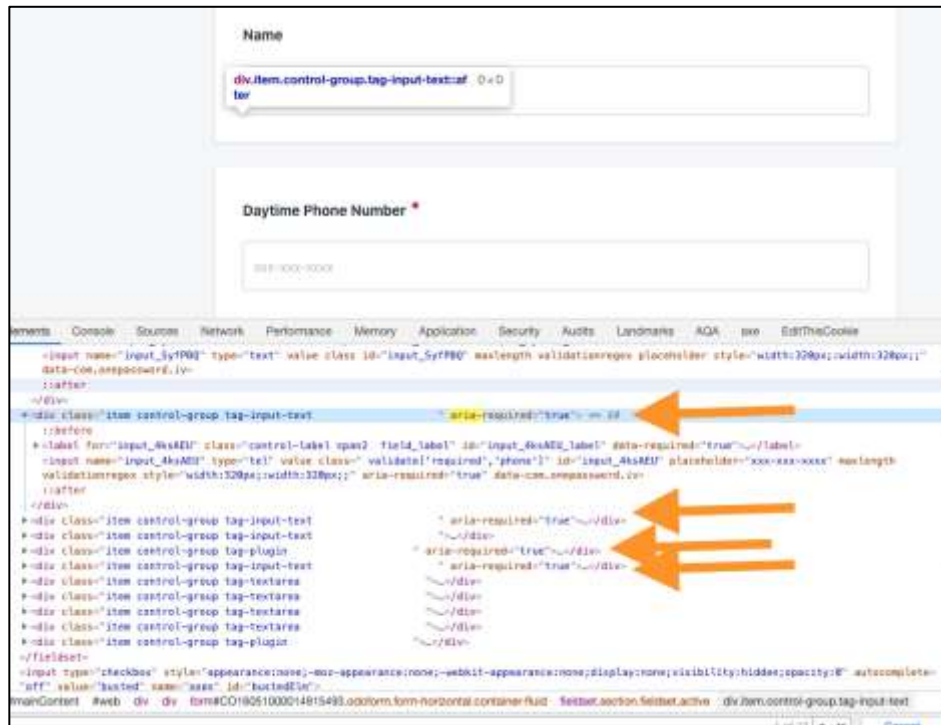


Figure 39a: - Screenshot of elements with disallowed ARIA attributes and code inspector window

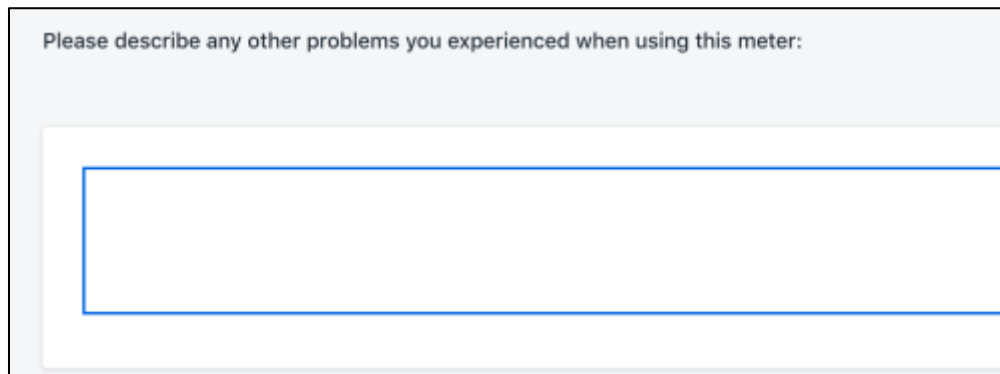


Figure 39b: - Screenshot of element with disallowed ARIA attribute

Carousel is not keyboard accessible (CLT-40)

Components	002 - charlottenc.gov - Homepage
Affected community	Vision, Motor
WCAG Success Criterion	2.1.1 Keyboard (A)
WCAG Level	A
Priority	High

Description

Starting at 170% zoom, as well as views narrower than approximately 766px, the campaign links display in a carousel that features Next and Previous triggers.

These triggers are not accessible by keyboard navigation. As such, keyboard users, sighted or not, cannot control the carousel.

Solution

Ensure that carousels are keyboard-friendly. Refer to <https://www.w3.org/WAI/tutorials/carousels/> for an in-depth tutorial on how to make carousels accessible.

Linked issues

- CLT-41

Screenshot(s)



Figure 40a: - Screenshot of carousel

No clear way to pause or stop carousel (CLT-41)

Components	002 - charlottenc.gov - Homepage
Affected community	Cognitive
WCAG Success Criterion	2.2.2 Pause, Stop, Hide (A)
WCAG Level	A
Priority	High

Description

Starting at 170% zoom, as well as views narrower than approximately 766px, the campaign images display in a carousel. This carousel of scrolls automatically and continuously with no clear option for visitors to pause or stop the content.

Content that moves or auto-updates can be a barrier to anyone who has trouble tracking moving objects. It can also cause problems for screen readers. Moving content can also be a severe distraction for some people. Certain groups, particularly those with attention deficit disorders, find blinking content distracting, making it difficult for them to concentrate on other parts of the Web page.

Solution

Add a pause `<button>` that allows users to stop the carousel movement.

Refer to <https://www.w3.org/WAI/tutorials/carousels/> for an in-depth tutorial on how to make carousels accessible

Linked issues

- CLT-40

Screenshot(s)



Figure 41a: - Screenshot of carousel

Insufficient color contrast on carousel controls (CLT-42)

Components	002 - charlottenc.gov - Homepage
Affected community	Vision
WCAG Success Criterion	1.4.3 Contrast (Minimum) (AA)
WCAG Level	AA
Priority	High


Description

Starting at 170% zoom, as well as views narrower than approximately 766px, the campaign links display in a carousel that features Next and Previous icons. These icons are transparent at all states and, when paired with the background images in the carousel, have varying contrasts.

Because the contrast is inconsistent, it can make it difficult for people with low vision to perceive the Next and Previous triggers.

Solution

Ensure the color contrast between foreground and background colors (or images) is at least 4.5:1.

Also consider allowing the default focus outline ( [CLT-14](#) Open).

Screenshot(s)



Figure 42a: - Screenshot of control with poor contrast

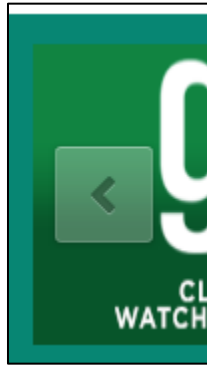


Figure 42b: - Screenshot of control with poor contrast



Figure 42c: - Screenshot of control with poor contrast

Sub menus are problematic (CLT-43)

Components	003 - charlottenc.gov - Services - CharMeck 311 (drop downs), 010 - charlottenc.gov - Charlotte Future (carousel and interactive map)
Affected community	Vision, Cognitive, Motor
WCAG Success Criterion	2.1.1 Keyboard (A), 2.4.7 Focus Visible (AA)
WCAG Level	A
Priority	High

Description

There are multiple issues with the main navigation of the page:

- No visible focus on active links so sighted keyboard users cannot see their position on a web page.
- Getting to links for the secondary submenu is not possible with keyboard or screen readers:
- When attempting to reach an item with secondary submenu with a keyboard, the tab order goes to the next link item that is not does not contain a secondary submenu option. This blocks access to those links.
- When attempting to navigate to the secondary submenu with a screen reader, the main submenu closes unexpectedly blocking access to those links.

Solution

Don't apply ARIA menu roles to site navigation, which means removing instances of `role="menu"`, as well as the trigger links that include ARIA properties for menu buttons:

- `role="button"`
- `aria-expanded`
- `aria-haspopup`

Instead, use CSS and/or JavaScript to support the keyboard navigation necessary to expand and contract the sub-navigation

Follow the coding examples on the W3C Flyout Menus tutorial to update this code to work well with both the mouse and the keyboard.

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

Linked issues

- CLT-22

Screenshot(s)



Figure 43a: - Screenshot of sub menus

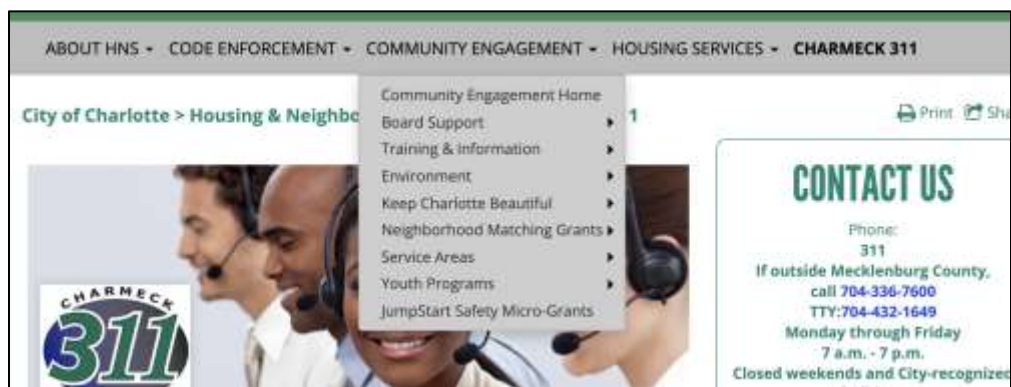


Figure 43b: - Screenshot of sub menus

Insufficient color contrast on link banner (CLT-44)

Components	004 - charlottenc.gov - Grave Spaces Evergreen, 010 - charlottenc.gov - Charlotte Future (carousel and interactive map)
Affected community	Vision
WCAG Success Criterion	1.4.3 Contrast (Minimum) (AA)
WCAG Level	AA
Priority	High

Description

There is insufficient color contrast between the background color for the hover/focus state and menu background color for the links. The minimum contrast ratio must be 3:1.

The foreground color is #CECECE and the background color is #bdbdbd. The contrast ratio is 1.19:1. Low contrast ratios may create difficulty for users with all levels of visual acuity.

Current code

```
...
<a class="" tabindex="0"
href="/Engineering/LandscapeManagement/Cemeteries/Pages/Locations.aspx">Locations</a>
>
#topnav ul.nav>li>a:visited {
    color: #000;
    background: #bdbdbd;
}

#topnav .navbar-nav>li>a:focus, #topnav .navbar-nav>li>a:hover {
    background: #cecece;
}
```

Solution

Ensure the color contrast between foreground and background colors is at least 3:1

Screenshot(s)



Figure 44a: - Screenshot of banner links with poor contrast

Print link opens new window (CLT-45)

Components	003 - charlottenc.gov - Services - CharMeck 311 (drop downs), 010 - charlottenc.gov - Charlotte Future (carousel and interactive map)
Affected community	Vision
WCAG Success Criterion	3.2.2 On Input (A)
WCAG Level	A
Priority	Medium

Description

When selecting to print the page, a new window appears and then the print dialog box appears. There is no warning to visitors that this will happen.

It can be a disorienting experience to have a new tab/window open. This is particularly problematic for screen-reader users who may try to navigate to the previous content but are unable to because it's in a different tab/window. This may also cause confusion in visitors with cognitive issues, who may not perceive that the link has opened in a new tab/window.

Solution

I would recommend using a print style sheet and having that style sheet linked within the web page. This would eliminate the need for a pop-up window for printing. An example of a print style sheet can be found here <https://github.com/msikma/h5bp-print-styles>

Do not force the opening of new window. Let the user decide when, and if, a new window is required.

Should that not be possible, when dealing with linked images, add the warning to the `` descriptive alt text.

Screenshot(s)

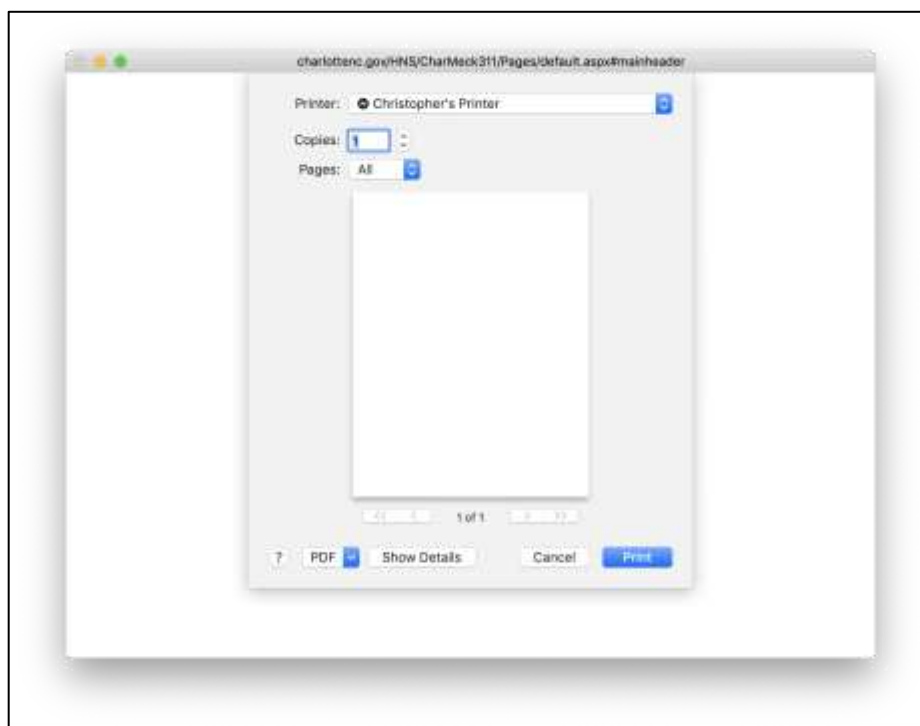


Figure 45a: - Screenshot of print function

Unordered list contains no list items, only a submit button (CLT-46)

Components	005 - charlottenc.gov - Bicyclist Report on Aggressive Drivers
Affected community	Vision, Cognitive
WCAG Success Criterion	1.3.1 Info and Relationships (A)
WCAG Level	A
Priority	High

Description

The submit button for the report form is wrapped in an unordered list (which contains no list items. Lists (ordered and unordered) do not allow `input` as descendants, and must contain list items within it to make sense, particularly for screen reader users.

Current code

```
<ul class="pager module_actions">
  <input type="submit" value="Submit" class="action_button submit primary">
</ul>
```

Solution

Remove the `ul` from around the `input`. Replace the `ul` with a `div` if needed for styling/layout purposes.

Recommended code

```
<div class="pager module_actions">
  <input type="submit" value="Submit" class="action_button submit primary">
</div>
```

Linked issues

- CLT-58

Screenshot(s)

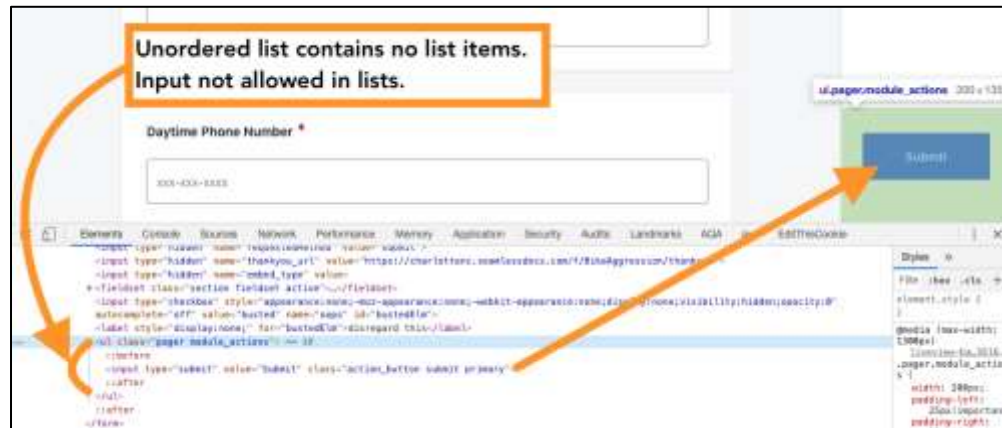


Figure 46a: - Screenshot of incorrect list

Tabbed interface is not keyboard accessible (CLT-47)

Components	002 - charlottenc.gov - Homepage, 010 - charlottenc.gov - Charlotte Future (carousel and interactive map)
Affected community	Vision, Motor
WCAG Success Criterion	4.1.2 Name, Role, Value (A)
WCAG Level	A
Priority	High

Description

The events content uses ARIA tab roles, but the keyboard accessibility doesn't follow expected patterns:

- Tab: When focus moves into the tab list, places focus on the active tab element. When the tab list contains the focus, moves focus to the next element in the tab sequence, which is the `tabpanel` element.
- Right Arrow: When a tab has focus, moves focus to the next tab. If focus is on the last tab, moves focus to the first tab.
- Left Arrow: When a tab has focus, moves focus to the previous tab. If focus is on the first tab, moves focus to the last tab.
- Delete: When focus is on the Joke tab, removes the tab from the tab list and places focus on the previous tab.

This can cause issues for screen-reader users, who hear information about tabs, but are unable to execute expected behavior. This can also cause problems for keyboard users, who are also unable to navigate as expected.

Solution

Ensure that tabbed interfaces are keyboard-friendly. Refer to <https://www.w3.org/TR/wai-aria-practices/examples/tabs/tabs-2/tabs.html> for an in-depth tutorial on how to make accessible tabbed interfaces.

Screenshot(s)



Figure 47a: - Screenshot of tabbed interface

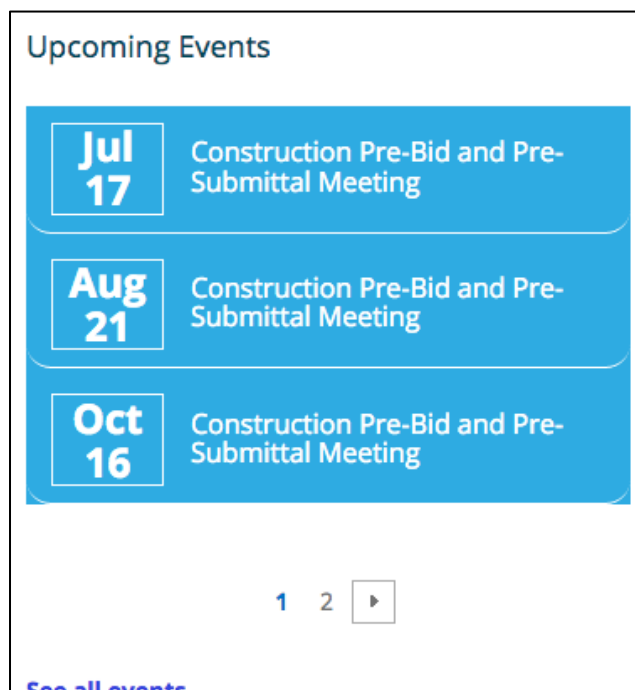


Figure 47b: - Screenshot of tabbed interface

Page title is insufficient (CLT-48)

Components	003 - charlottenc.gov - Services - CharMeck 311 (drop downs), 009 - charlottenc.gov - HR - Jobs - See Available Jobs
Affected community	Vision, Cognitive
WCAG Success Criterion	2.4.2 Page Titled (A)
WCAG Level	A
Priority	Low

Description

The page title is "CharMeck 311 > Home". Since

The page is not the secondary level of the Home page. Also, "Home" does not adequately describe the content on the page.

A page title needs to be unique, concise and descriptive enough to provide a clear understanding of the page's context and domain. This practice is important since people who are blind and use screen reading software will hear the page title as the first thing and that allows them to know that they are in the right place. Users with cognitive disabilities may use the page title as it appears in the browser tab to verify they are on the right track

Current code

```
<title>
    CharMeck 311 >
    Home
</title>
```

Solution

I recommend assigning a unique, descriptive title to pages that clearly provide their context to screen-reader users.

Recommended code

```
<title>CharMeck 311 - Housing & Neighborhood Services</title>
```

Screenshot(s)



Figure 48a: - Screenshot of page title tab

Duplicate content generated by JavaScript (CLT-49)

Components	002 - charlottenc.gov - Homepage
Affected community	Vision
WCAG Success Criterion	4.1.1 Parsing (A), 4.1.2 Name, Role, Value (A)
WCAG Level	A
Priority	Medium

Description

The "This Week" content is announced by screen readers twice. This appears to be due to JavaScript, which is generating duplicate content.

In addition to causing repetitive content for screen-reader users, which is a cumbersome experience, the duplicate content could also cause parsing issues for assistive technologies.

Solution

Remove the JavaScript that generates redundant content.

Screenshot(s)



Figure 49a: - Screenshot of duplicate content

Text in uppercase (CLT-50)

Components	002 - charlottenc.gov - Homepage, 009 - charlottenc.gov - HR - Jobs - See Available Jobs
Affected community	Vision
WCAG Success Criterion	4.1.2 Name, Role, Value (A)
WCAG Level	A
Priority	Low

Description

Text, such as the word "update" in the news section, is written in uppercase. This may trigger screen readers to announce the words as if they were acronyms and spell out one letter at a time. This may be confusing to screen-reader users.

Current code

```
<a href="https://charlottenc.gov/newsroom/releases/Pages/Central-Division---Homicide-in-the-300-block-of-North-Graham-Street.aspx">***UPDATE*** Central Division - Homicide in the 300 block of North Graham Street</a>
```

Solution

Write words in lower case or sentence case. Use CSS `text-transform: uppercase` to change the visual display.

Recommended code

```
<a href="https://charlottenc.gov/newsroom/releases/Pages/Central-Division---Homicide-in-the-300-block-of-North-Graham-Street.aspx">***<span class="txt-upper">Update</span>*** Central Division - Homicide in the 300 block of North Graham Street</a>

.txt-upper {
    text-transform:uppercase;
}
```

Screenshot(s)



Figure 50a: - Screenshot of text in uppercase



Figure 50b: - Screenshot of text in uppercase

Zooming/scaling disabled in viewport meta tag (CLT-51)

Components	005 - charlottenc.gov - Bicyclist Report on Aggressive Drivers, 006 - charlottenc.gov - Report Broken Parking Meter
Affected community	Vision
WCAG Success Criterion	1.4.4 Resize Text (AA), 1.4.10 Reflow (AA) (WCAG 2.1)
WCAG Level	AA
Priority	High

Description

The viewport meta element suggests to browsers that the content should not be zoomed by pinching on touch input devices. This pattern makes it difficult for visitors with low vision that need to enlarge the page to view the content.

For more information and resources about this issue, please see:

- [Accessibility in Resizing Text](#), article by Tim Wright
- [A11y Project Checklist: Ensure that viewport zoom is not disabled](#)

Current code

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

Solution

Do not prevent users' ability to zoom or scale the content to suit their needs.

Recommended code

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

ARIA attributes referencing ID that doesn't exist (CLT-52)

Components	006 - charlottenc.gov - Report Broken Parking Meter
Affected community	Vision
WCAG Success Criterion	3.3.2 Labels or Instructions (A)
WCAG Level	A
Priority	High

Description

Several elements have an `aria-label` attribute applied that refers to an `id` that does not exist on the page. This could cause conflicts for some assistive technologies. It also means the intention of providing a label for the option group is falling short - no labelling information is available.

Current code

```
<div class="" id="div_CUte39" type="" style=""><p><strong>Did you receive a
ticket?</strong></p>
</div>

...
<div class="item control-group tag-input-checkbox " role="group" aria-
labelledby="input_Gycet3_label" aria-required="true">
    <div class="option_group">
        <div class="option_item"><input name="input_Gycet3[]"
id="Yes_0" value="Yes " type="checkbox" class=" validate['group[\`input_Gycet3\`']]'
style=""> <label class="option_label radio_label" for="Yes_0">Yes </label></div>
        <div class="option_item"><input name="input_Gycet3[]"
id="No1" value="No" type="checkbox" class=" validate['group[\`input_Gycet3\`']]'
style=""> <label class="option_label radio_label" for="No1">No</label></div>
    </div>
</div>
```

Solution

It would be better to avoid using ARIA in this case and group the elements with a `<fieldset>`, using a `<legend>` to label the entire checkbox group.

Recommended code

```
<fieldset>
<div class="" id="div_CUte39" type="" style=""><legend><strong>Did you receive a
ticket?</strong></p>
</div>
```

```

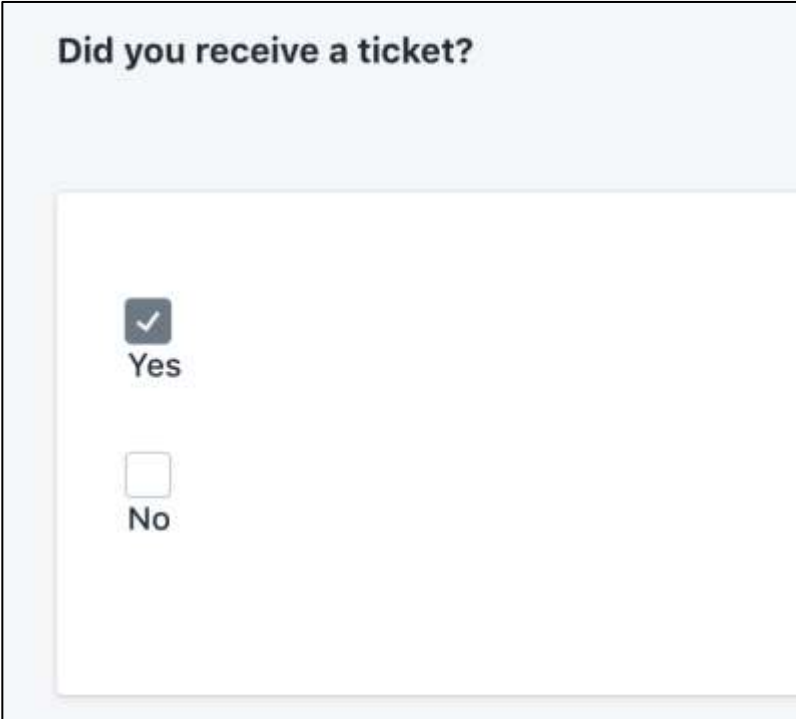
...
<div class="item control-group tag-input-checkbox" >
    <div class="option_group">
        <div class="option_item"><input required
name="input_Gycet3[]" id="Yes_0" value="Yes " type="checkbox" class="
validate['group[\'input_Gycet3\']']" style=""> <label class="option_label
radio_label" for="Yes_0">Yes </label></div>
        <div class="option_item"><input required
name="input_Gycet3[]" id="No1" value="No" type="checkbox" class="
validate['group[\'input_Gycet3\']']" style=""> <label class="option_label
radio_label" for="No1">No</label></div>
    </div>
</div>
</fieldset>

```

Linked issues

- CLT-53

Screenshot(s)



The screenshot shows a light blue rectangular box containing the question "Did you receive a ticket?". Below the question is a white rectangular area with a thin grey border. Inside this area, there are two radio button options. The first option is "Yes", which has a dark grey radio button with a white checkmark inside. The second option is "No", which has a white radio button with a grey border.

Figure 52a: - Screenshot of elements with poor ARIA attribute references

Checkboxes used instead of radio buttons (CLT-53)

Components	006 - charlottenc.gov - Report Broken Parking Meter
Affected community	Vision
WCAG Success Criterion	1.3.1 Info and Relationships (A), 4.1.2 Name, Role, Value (A)
WCAG Level	A
Priority	Medium

Description

Checkboxes were used for users to complete the "did you receive a ticket" question. As this is a yes or no selection, mutually exclusive, the elements should be radio buttons. Alternatively, a drop down list.

Solution

Use radio buttons instead of checkboxes.

Linked issues

- CLT-52

Screenshot(s)


A screenshot of a web form with a light blue header bar containing the text "Did you receive a ticket?". Below the header is a white rectangular area containing two checkboxes. The first checkbox is checked and has the word "Yes" below it. The second checkbox is unchecked and has the word "No" below it.

Figure 53a: - Screenshot of checkboxes used instead of radio buttons

Error messages announced one at a time (CLT-54)

Components	006 - charlottenc.gov - Report Broken Parking Meter
Affected community	Vision, Cognitive, Motor
WCAG Success Criterion	3.3.1 Error Identification (A)
WCAG Level	A
Priority	High

Description

If there are more than one error made on the form, and the form is submitted, only the first error message is listed. Then when the form is submitted again, the next error message is listed. This will make for a cumbersome experience for people.

Solution

Identify all the errors on the form at the same time. Provide an "error bucket" at the top of the page. Set focus on the heading for that block of content, so screen readers can immediately announce it. List each error message in a list. Link each error message to the erroneous field.

Recommended code

```
<h3 tabindex="-1">There are errors in the form</h3>
<ul>
<li><a href="#date">Event date cannot be left empty</a></li>
<li><a href="#fname">First name cannot be left empty</a></li>
<li><a href="#email">Email is not a valid email address</a></li>
</ul>
```


Screenshot(s)

The screenshot shows a web form with several input fields. At the top, there is a date picker with a red header that says "Date is required" and "019". The date picker shows a calendar for the month of January 2019. Below the date picker, there are two input fields for "Last Name" with a red asterisk indicating a required field. Below these, there are two input fields for "Street Address" with a red asterisk indicating a required field. Below the street address fields, there are three input fields for "City", "State", and "Zip". The "State" field has a dropdown menu with "Choose State" selected. Below the city, state, and zip fields, there is a single input field for "Daytime Phone Number" with a red asterisk indicating a required field.

Su	Mo	Tu	We	Th	Fr	Sa
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	1	2	3	4	5	6

Last Name *

Street Address *

City State Zip

Choose State

Daytime Phone Number *

Figure 54a: - Screenshot of error message

Datepicker not accessible (CLT-55)

Components	005 - charlottenc.gov - Bicyclist Report on Aggressive Drivers, 006 - charlottenc.gov - Report Broken Parking Meter
Affected community	Vision, Motor
WCAG Success Criterion	1.3.1 Info and Relationships (A), 2.1.1 Keyboard (A), 4.1.1 Parsing (A), 4.1.2 Name, Role, Value (A)
WCAG Level	A
Priority	Highest

Description

The Date and Time of Incident input field on the Bicyclist Report form does not allow direct input and instead relies on a datepicker/calendar widget for entry of date into the field.

Unfortunately, the pop-up for the calendar datepicker is not announced or available to screen reader users nor is it accessible to keyboard-only users who can see the screen but cannot use a mouse.

These problems are caused by a few key issues which include:

- Datepicker component/code is placed at the end of the DOM, just before the closing `body` tag.
- Datepicker code uses entirely non-semantic HTML which have no (0) focusable elements in it.
- Focus is not shifted to the datepicker at all when it is opened.
- Users are not made aware that there is a datapicker that requires interaction with to fill out the required field.

Additionally, the label for the field reads "Date and Time of Incident" yet there is no provision for providing a time which would be expected based on the label.

The problem is very similar on the report broken meter page.

Solution

- Allow entry of date into the field manually.
- Use semantic HTML for interactions with the datepicker (e.g. use `button` or `a` and in order to allow native keyboard focus and easier event handling).
- Place datepicker code directly after the field and show it when appropriate (also shift focus to and away from it properly).

More information on accessible datepickers:

- [Axe Lab: Accessible Datepickers](#)
- [WebAxe: Accessible Datepickers](#)
- [Deque: Accessible Datepicker](#)

Screenshot(s)

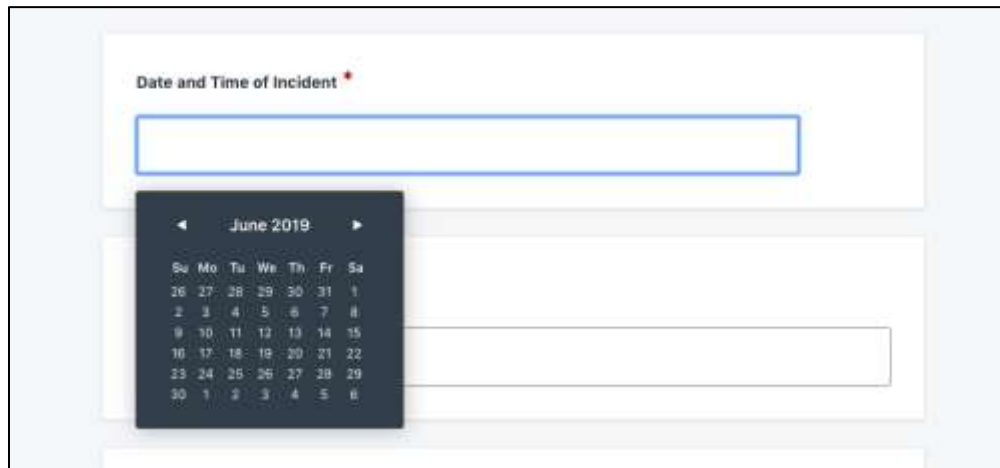


Figure 55a: - Screenshot of datepicker

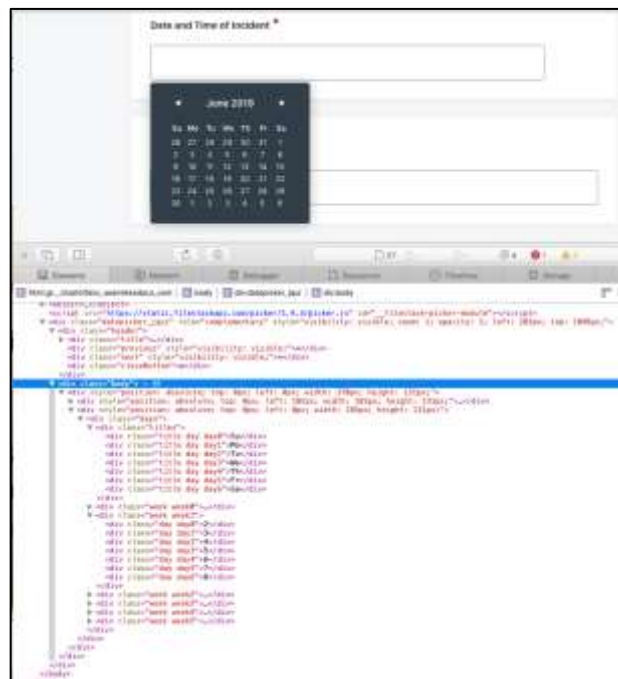


Figure 55b: - Screenshot of datepicker and code inspector

Empty heading (CLT-57)

Components	003 - charlottenc.gov - Services - CharMeck 311 (drop downs), 010 - charlottenc.gov - Charlotte Future (carousel and interactive map)
Affected community	Vision
WCAG Success Criterion	2.4.6 Headings and Labels (AA)
WCAG Level	AA
Priority	Medium

Description

Screen reader users often navigate a page going from heading to heading to get a feel for what is present on a page. Or they bring up a list of headings on a page. If a heading is empty, they will wonder what the heading is intended to be for, and may get confused

Current code

```
<h2 class="ms-rteElement-Heading2Blue" style=""></h2>
<h2 id="sharePageTitle"></h2>
<h2 style="text-align:justify;" class="ms-webpart-
titleText"><nobr><span></span><span id="WebPartCaptionWPQ13"></span></nobr></h2>
```

Solution

Do not deliver empty headings. Remove the empty h2 elements.

Screenshot(s)

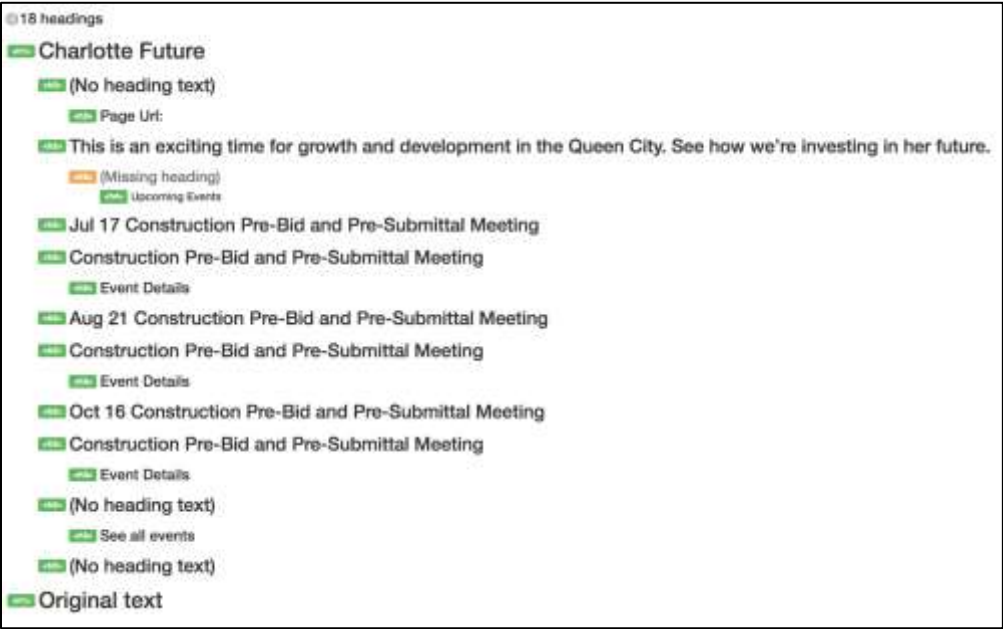


Figure 57a: - Screenshot of document outline



Figure 57b: - Screenshot of empty heading

Unordered list used with no list items (CLT-58)

Components	005 - charlottenc.gov - Bicyclist Report on Aggressive Drivers, 006 - charlottenc.gov - Report Broken Parking Meter
Affected community	Vision
WCAG Success Criterion	1.3.1 Info and Relationships (A)
WCAG Level	
Priority	Medium

Description

There are unordered lists with no list items. This may have an impact on screen reader users. Screen readers parse and announce lists in specific ways. If the list isn't marked up properly, it may create conflicts for the assistive technology.

Current code

```
<ul class="pager module_actions"><input type="submit" value="Submit"
class="action_button submit primary"></ul>
```

Solution

As the content inside the `` tag isn't a list, the best option is to remove the tag altogether, or use a semantically meaningful tag, such as `<p>`. Ensure the tag follows the patten of the rest of the page.

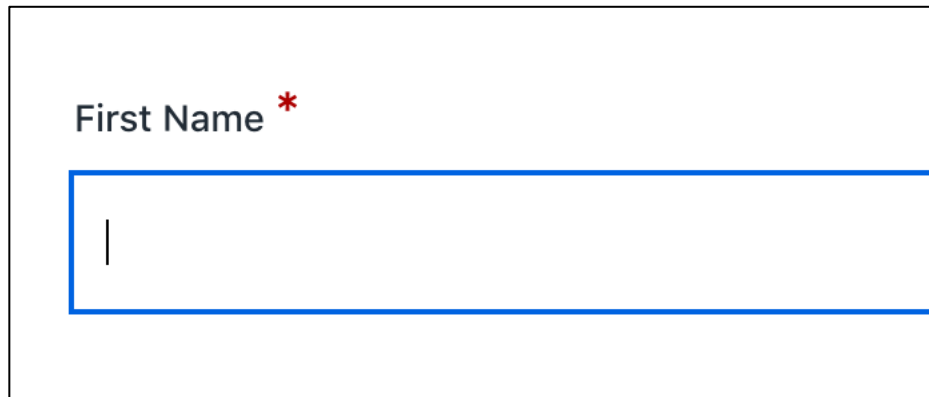
Recommended code

```
<p class="pager module_actions"><input type="submit" value="Submit"
class="action_button submit primary"></p>
```

Linked issues

- CLT-46

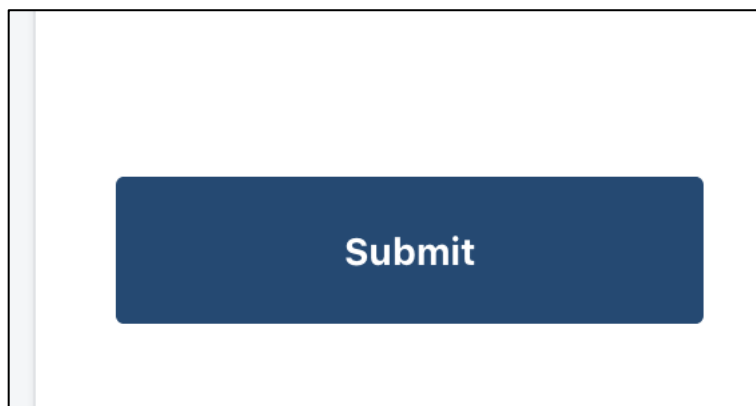
Screenshot(s)



First Name *

A screenshot of a web form element. It features a label "First Name" followed by a red asterisk, indicating a required field. Below the label is a rectangular input box with a blue border. A vertical cursor is positioned at the beginning of the input box.

Figure 58a: - Screenshot of element loaded as a list



Submit

A screenshot of a dark blue rectangular button with rounded corners. The word "Submit" is centered on the button in white text.

Figure 58b: - Screenshot of element loaded as a list

Visible focus removed on some input elements for Firefox/Mozilla browsers (CLT-59)

Components	005 - charlottenc.gov - Bicyclist Report on Aggressive Drivers
Affected community	Vision, Motor
WCAG Success Criterion	2.4.7 Focus Visible (AA)
WCAG Level	AA
Priority	High

Description

Keyboard-only users with normal and low vision rely on visible focus indicators (a.k.a. "focus outlines") to know where they are on a page and what action(s) are available to them.

A good set of visible keyboard focus styles have been provided for most instances on the Bicyclist Report on Aggressive Drivers page. However, the visible focus indication has been explicitly removed for Firefox and other Mozilla-based browsers using the `-moz` browser prefix in CSS.

Current code

```
button::-moz-focus-inner,  
input::-moz-focus-inner {  
  border:0;  
  padding:0  
}
```

Solution

Remove all instances of the `current code` from CSS.

Screenshot(s)



Figure 59a: - Screenshot of elements with no visible focus in Firefox

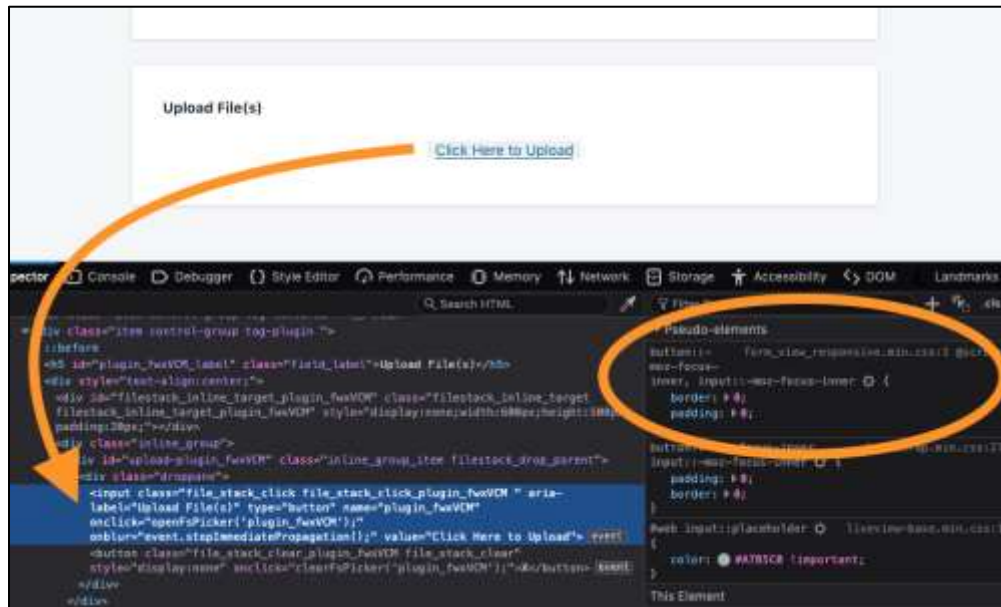


Figure 59b: - Screenshot of elements with no visible focus in Firefox

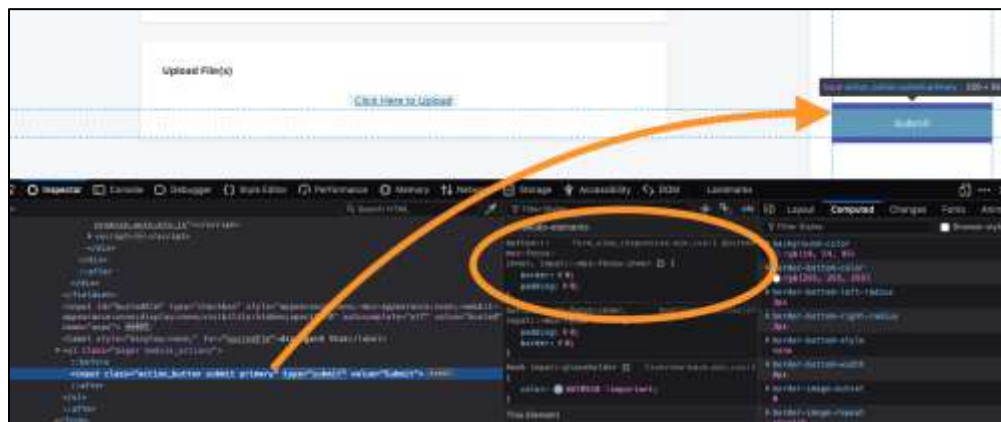


Figure 59c: - Screenshot of elements with no visible focus in Firefox

Text resize widget out of visible focus order (CLT-60)

Components	005 - charlottenc.gov - Bicyclist Report on Aggressive Drivers
Affected community	Vision, Motor
WCAG Success Criterion	2.4.3 Focus Order (A)
WCAG Level	A
Priority	High


Description

The text resize widget appears, visibly, in the top right section of the Bicyclist Report on Aggressive Drivers page. However, it is the last item on the page to receive focus.

This can be problematic for keyboard-only users who may want to use the resize widget but are forced to tab through to the bottom of the page to access the widget control.

Solution

At bare minimum, move code for widget to the top of the page to keep the focus in the expected order which aligns with its visual placement/prominence.

Since this issue relates closely to issue  [CLT-33](#) Open , it is recommended to remove the text-resize widgets, if possible, since many users who need this will already have other controls in place for zoom and/or resizing globally.

Linked issues

- CLT-33

Screenshot(s)

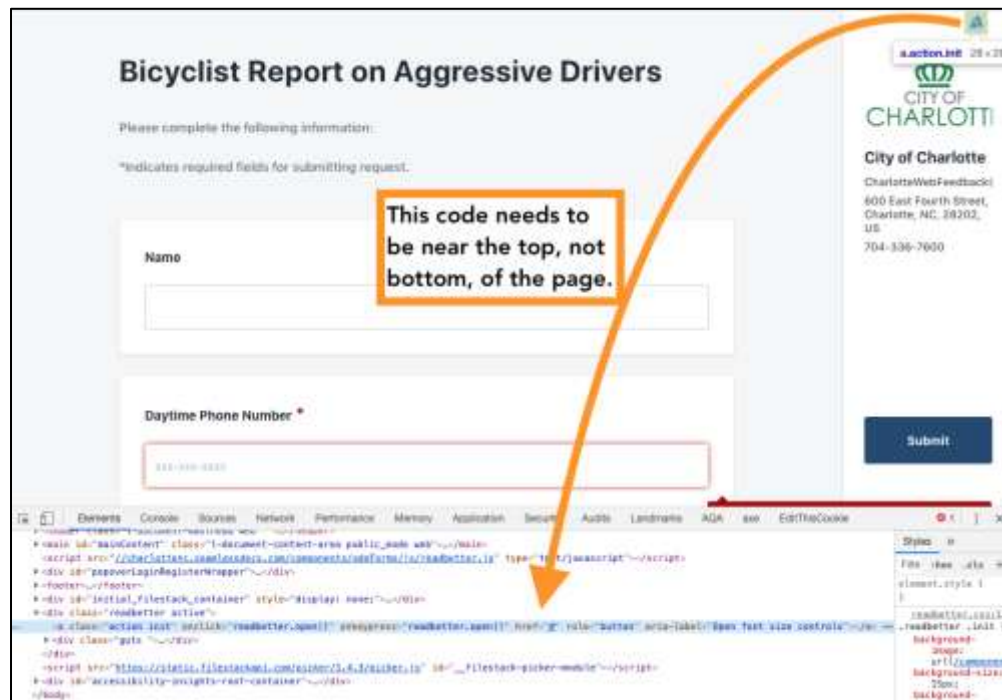


Figure 60a: - Screenshot of element out of focus order

Copy-and-paste share URL's purpose not announced (CLT-61)

Components	003 - charlottenc.gov - Services - CharMeck 311 (drop downs), 010 - charlottenc.gov - Charlotte Future (carousel and interactive map)
Affected community	Vision
WCAG Success Criterion	1.3.1 Info and Relationships (A), 3.2.2 On Input (A), 3.3.2 Labels or Instructions (A)
WCAG Level	A
Priority	Medium

Description

There is an input field that's for copy and pasting. The screen reader is not announcing its purpose (for copying and pasting). Also, there's no label or form element around this input element.

Current code

```
<h3>Page Url:</h3>
<input id="shareUrl" type="text" style="width:80%;">
```

Solution

I would recommend wrapping the input field with a `form` element and use a `label` to provide meaning and context to the input field. Also, use a `button` to provide copy-and-pasting for users rather than relying on providing the text in an input field.

Recommended code

```
<form>
<h3><label for="shareURL">Page Link for Sharing</label></h3>
<input id="shareUrl" type="text" style="width:80%;">
<button>Copy</button>
</form>
```

Screenshot(s)

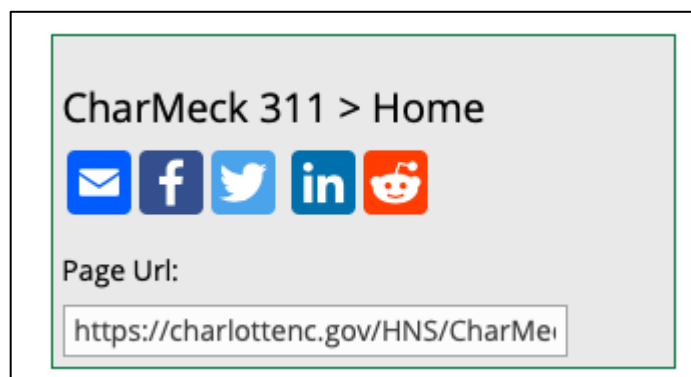


Figure 61a: - Screenshot of copy/paste not programmatically sharing purpose

Multiple H1s and other heading issues (CLT-62)

Components	005 - charlottenc.gov - Bicyclist Report on Aggressive Drivers, 007 - charlottenc.gov - TRASH - Schedule a Bulky Item Pickup, 008 - charlottenc.gov - Serve on Advisory Boards - Apply for a Board, 009 - charlottenc.gov - HR - Jobs - See Available Jobs, 010 - charlottenc.gov - Charlotte Future (carousel and interactive map)
Affected community	Vision
WCAG Success Criterion	1.3.1 Info and Relationships (A)
WCAG Level	A
Priority	High

Description

The Bicyclist Report on Aggressive Drivers page contains four headings, two of which are H1 elements. The other two headings are H4 elements. Headings for levels 2 and 3 (H2 and H3) are skipped.

This can be an issue for screen reader users who want to quickly determine the purpose of a page as well as the hierarchy of the page.

Solution

Use one heading level one (H1) per page according to the current W3C HTML5 spec (<https://www.w3.org/TR/html52/sections.html#headings-and-sections>).

Ensure any heading element is truly a heading element and use the appropriate nesting level for each in the document outline.

Headings are important landmarks for people who use assistive technologies when browsing the web. Several issues are important when using headings on a page:

- Heading content should be descriptive.
- The most important heading (topic of page) should be an H1 — best practice for accessibility is to limit it to one H1.
- Other headings must be used according to their hierarchy of the page, *not* the visual hierarchy.
- Nest headings by their rank (or level). The most important heading has the rank 1 (H1), the least important heading rank 6 (H6). Headings with an equal or higher rank start a new section, headings with a lower rank start new subsections that are part of the higher ranked section.

See <https://www.w3.org/WAI/tutorials/page-structure/headings/> for further information.

Further reading at: <https://www.w3.org/TR/WCAG20-TECHS/H42.html>

Screenshot(s)

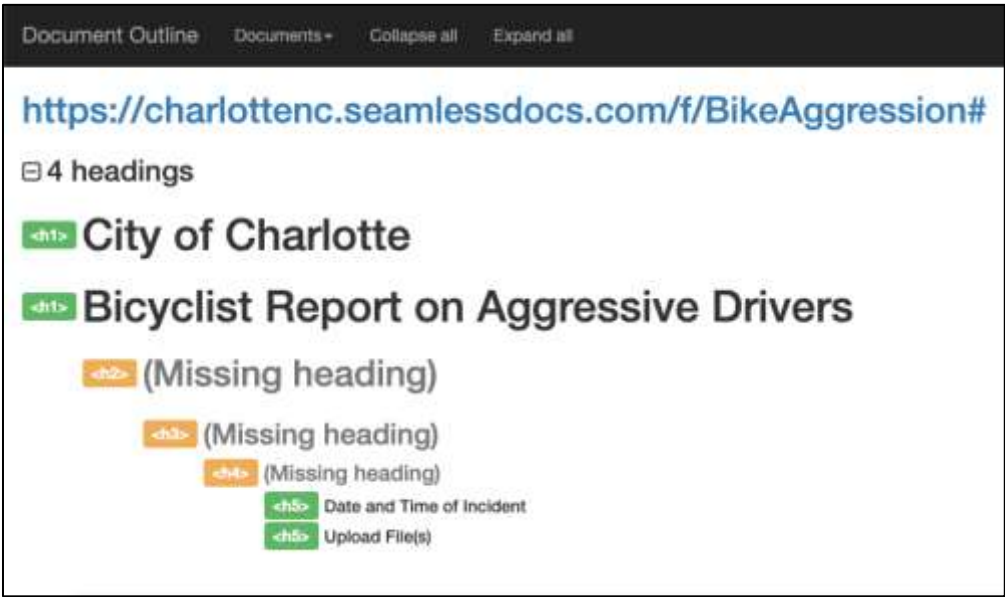


Figure 62a: - Screenshot of document outline

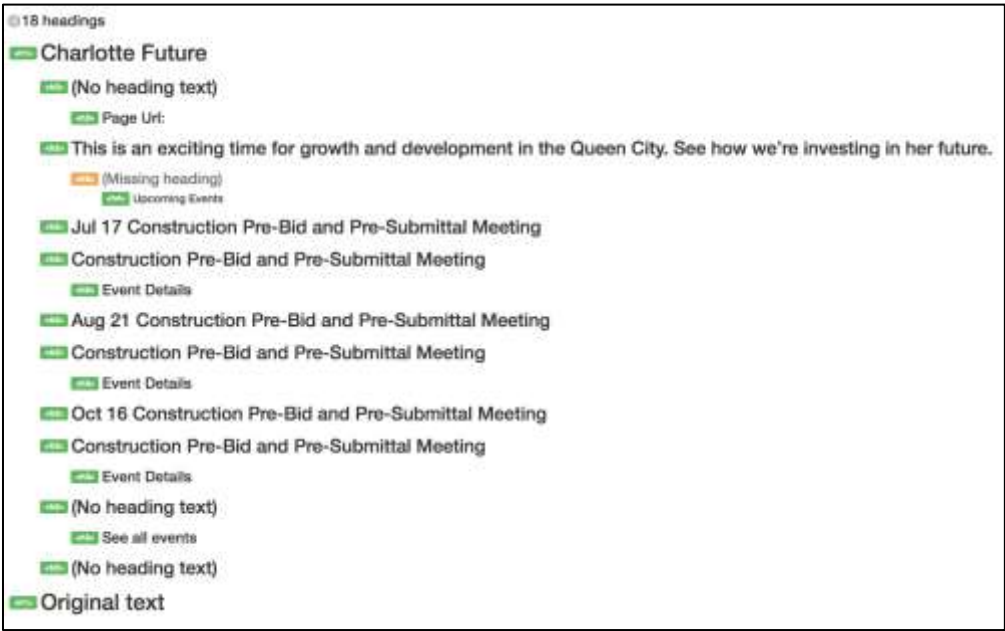


Figure 62b: - Screenshot of document outline

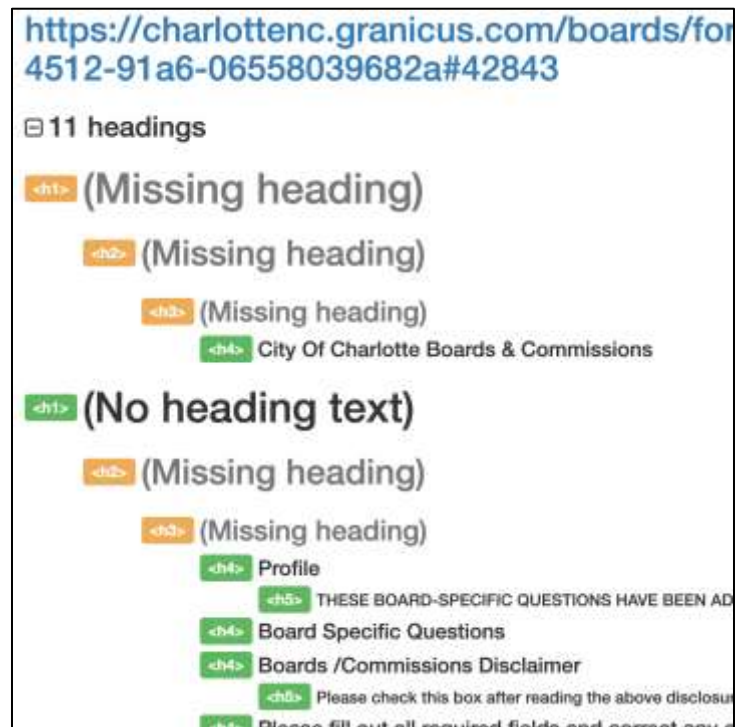


Figure 62c: - Screenshot of document outline

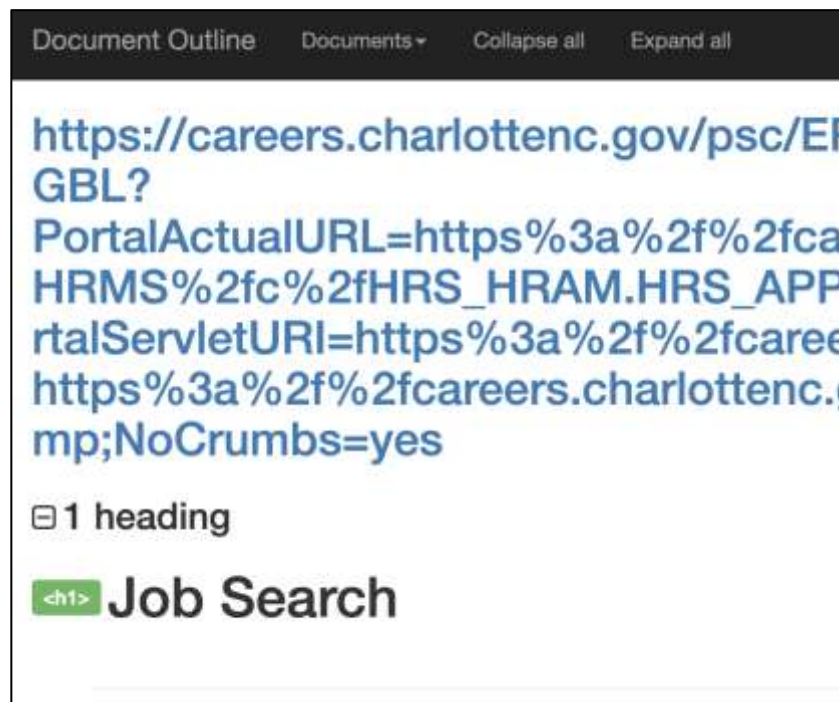


Figure 62d: - Screenshot of document outline

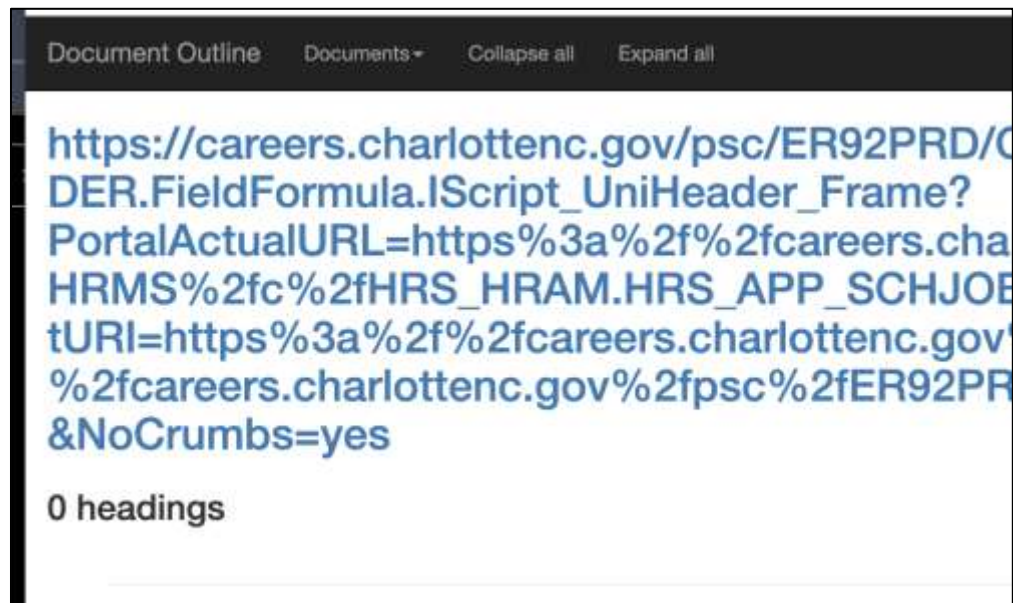


Figure 62e: - Screenshot of document outline

SVG images not marked with role="img" (CLT-63)

Components	003 - charlottenc.gov - Services - CharMeck 311 (drop downs), 010 - charlottenc.gov - Charlotte Future (carousel and interactive map)
Affected community	Vision
WCAG Success Criterion	1.1.1 Non-text content (A)
WCAG Level	A
Priority	Low

Description

The social media SVG icons do not use `{role="img"}` to identify that SVG's should be considered as a single image.

Also, the social media icons are delivered as an SVG icons without alternative text. Instead, text for the icon is posted next to the image and then hidden with CSS.

Current code

```
<a class="a2a_button_reddit"
href="https://www.addtoany.com/add_to/reddit?linkurl=https%3A%2F%2Fcharlottenc.gov%2FHN%2FCharMeck311%2FPages%2Fdefault.aspx&linkname=CharMeck%20311%20%3E%20Home&linknote=Official%20website%20for%20the%20city%20of%20Charlotte%2C%20North%20Carolina."
rel="nofollow noopener">
  <span class="a2a_svg a2a_s__default a2a_s_reddit" style="background-color:
rgb(255, 69, 0);">
    <svg focusable="false" xmlns="http://www.w3.org/2000/svg" viewBox="0 0 32
32"><path d="M28.543 15.774a2.953 2.953 0 0 0-2.951-2.949 2.882 2.882 0 0 0-1.9.713
14.075 14.075 0 0 0-6.85-2.044l11.38-4.349 3.768.884a2.452 2.452 0 1 0 .24-1.176l-
4.274-1a.6.6 0 0 0-.709.41-1.659 5.224a14.314 14.314 0 0 0-7.316 2.029 2.908 2.908 0
0 0-1.872-.681 2.942 2.942 0 0 0-1.618 5.4 5.109 5.109 0 0 0-.062.765c0 4.158 5.037
7.541 11.229 7.541s11.22-3.383 11.22-7.541a5.2 5.2 0 0 0-.053-.706 2.963 2.963 0 0 0
1.427-2.51zm-18.008 1.88a1.753 1.753 0 0 1 1.73-1.74 1.73 1.73 0 0 1 1.709 1.74
1.709 1.709 0 0 1-1.709 1.711 1.733 1.733 0 0 1-1.73-1.711zm9.565 4.968a5.573 5.573
0 0 1-4.081 1.272h-.032a5.576 5.576 0 0 1-4.087-1.272.6.6 0 0 1 .844-.854 4.5 4.5 0
0 0 3.238.927h.032a4.5 4.5 0 0 0 3.237-.927.6.6 0 1 1 .844.854zm-.331-3.256a1.726
1.726 0 1 1 1.709-1.712 1.717 1.717 0 0 1-1.712 1.712z" fill="#fff">
    </path></svg>
  </span>
  <span class="a2a_label">Reddit</span>
</a>
```

Solution

Add `role="img"` to the SVG. Add an `aria-label` to the SVG.

See <https://css-tricks.com/accessible-svg/> for further information about accessible SVG

Recommended code

```
<a class="a2a_button_reddit"
href="https://www.addtoany.com/add_to/reddit?linkurl=https%3A%2F%2Fcharlottenc.gov%2
FHNS%2FCharMeck311%2FPages%2Fdefault.aspx&linkname=CharMeck%20311%20%3E%20Home&linkn
ote=Official%20website%20for%20the%20city%20of%20Charlotte%2C%20North%20Carolina."
rel="nofollow noopener">
  <span class="a2a_svg a2a_s__default a2a_s_reddit" style="background-color:
rgb(255, 69, 0);">
    <svg role="img" aria-label="reddit" xmlns="http://www.w3.org/2000/svg"
viewBox="0 0 32 32"><path d="M28.543 15.774a2.953 2.953 0 0 0-2.951-2.949 2.882
2.882 0 0 0-1.9.713 14.075 14.075 0 0 0-6.85-2.044l1.38-4.349 3.768.884a2.452 2.452
0 1 0 .24-1.176l-4.274-1a.6.6 0 0 0-.709.41-1.659 5.224a14.314 14.314 0 0 0-7.316
2.029 2.908 2.908 0 0 0-1.872-.681 2.942 2.942 0 0 0-1.618 5.4 5.109 5.109 0 0 0-
.062.765c0 4.158 5.037 7.541 11.229 7.541s11.22-3.383 11.22-7.541a5.2 5.2 0 0 0-
.053-.706 2.963 2.963 0 0 0 1.427-2.51zm-18.008 1.88a1.753 1.753 0 0 1 1.73-1.74
1.73 1.73 0 0 1 1.709 1.74 1.709 1.709 0 0 1-1.709 1.711 1.733 1.733 0 0 1-1.73-
1.711zm9.565 4.968a5.573 5.573 0 0 1-4.081 1.272h-.032a5.576 5.576 0 0 1-4.087-
1.272.6.6 0 0 1 .844-.854 4.5 4.5 0 0 0 3.238.927h.032a4.5 4.5 0 0 0 3.237-.927.6.6
0 1 1 .844.854zm-.331-3.256a1.726 1.726 0 1 1 1.709-1.712 1.717 1.717 0 0 1-1.712
1.712z" fill="#fff">
    </path></svg></span>
</a>
```

Screenshot(s)



Figure 63a: - Screenshot of SVG images

Skip to form link does not work (CLT-64)

Components	005 - charlottenc.gov - Bicyclist Report on Aggressive Drivers
Affected community	Vision, Motor
WCAG Success Criterion	2.4.1 Bypass Blocks (A)
WCAG Level	A
Priority	High

Description

The Skip to form link does not link to the form, nor does it shift focus to it.

Skip links, also known as Bypass blocks, allow users to bypass repeated blocks of content such as header banners and navigation lists to directly access the unique content on a page. This is beneficial to both users of screen reader software as well as keyboard only users.

Current code

```
<a href="#mainContent" class="skip-to-main-link" style="position:absolute;top:-1000px;" onclick="return false;">Skip to form</a>
```

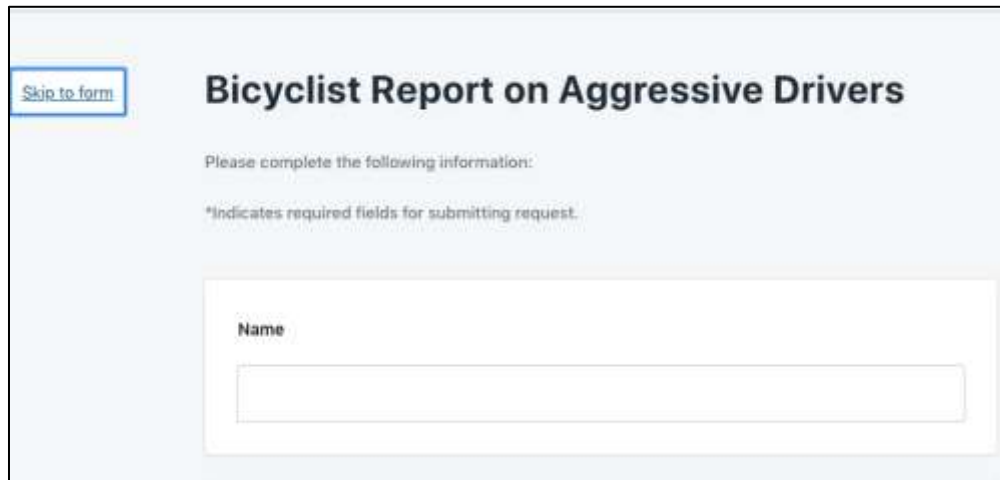
Solution

Remove `onclick="return false; "` from the "Skip to form" link. Update the links `href` value to the ID for the heading (H1) above the form instead of the `#mainContent` ID since the link is to skip someone to the form itself.

Recommended code

```
<a href="#h1_soxTqU" class="skip-to-main-link" style="position:absolute;top:-1000px;">Skip to form</a>
```

Screenshot(s)



The screenshot shows a web form titled "Bicyclist Report on Aggressive Drivers". In the top left corner, there is a blue-bordered button labeled "Skip to form". Below the title, the text "Please complete the following information:" is displayed. A note below that states "*Indicates required fields for submitting request." The form itself is a light gray box containing a label "Name" and an empty text input field below it.

Figure 64a: - Screenshot of skip link

Content not marked as a list (CLT-65)

Components	003 - charlottenc.gov - Services - CharMeck 311 (drop downs)
Affected community	Vision, Motor
WCAG Success Criterion	1.3.1 Info and Relationships (A)
WCAG Level	A
Priority	Medium

Description

The list of holidays is written out in `span` and `br` elements. Without proper structure from list elements like `ul`, assistive technologies are not able provide valuable information to users.

For example, if a list is marked up in HTML as a list (by using `` or ``, and `li` elements), screen readers are able to announce the list and the number of items inside that list to give the user more orientation.

Current code

```
<div class="ms-rtestate-field">
  <h3 class="ms-rteElement-Heading3Grey" style="text-align: center;">City-Observed
Holidays</h3>
  <p style="text-align: center;">New Years Day
    <br><span style="line-height: 1.6; font-size: 1em; background-color:
transparent;">Martin Luther King Jr. Day<br></span><span style="line-height: 1.6;
font-size: 1em; background-color: transparent;">Good Friday<br></span><span
style="line-height: 1.6; font-size: 1em; background-color: transparent;">Memorial
Day<br></span><span style="line-height: 1.6; font-size: 1em; background-color:
transparent;">Independence Day<br></span><span style="line-height: 1.6; font-size:
1em; background-color: transparent;">Labor Day<br></span><span style="line-height:
1.6; font-size: 1em; background-color: transparent;">Veteran's Day<br></span><span
style="line-height: 1.6; font-size: 1em; background-color:
transparent;">Thanksgiving Day + day after<br></span><span style="line-height: 1.6;
font-size: 1em; background-color: transparent;">Christmas + one day</span></p>
</div>
```

Solution

Use semantic HTML such as an unordered list to provide a solid, accessible experience.

Recommended code

```
<div class="ms-rtestate-field">
```

```
<h3 class="ms-rteElement-Heading3Grey" style="text-align: center;">City-Observed  
Holidays</h3>  
<ul>  
  <li>New Years Day</li>  
  <li>Martin Luther King Jr. Day</li>  
  <li>Good Friday</li>  
  <li>Memorial Day</li>  
  <li>Independence Day</li>  
  <li>Labor Day</li>  
  <li>Veteran's Day</li>  
  <li>Thanksgiving Day + day after</li>  
  <li>Christmas + one day</li>  
</ul>  
</div>
```

Screenshot(s)



Figure 65a: - Screenshot of list items not marked up as a list

Errors in form submission not announced or displayed clearly (CLT-68)

Components	005 - charlottenc.gov - Bicyclist Report on Aggressive Drivers
Affected community	Vision, Cognitive, Motor
WCAG Success Criterion	3.3.1 Error Identification (A)
WCAG Level	A
Priority	Highest

Description

When the Bicyclist Report on Aggressive Drivers form is submitted with erroneous and/or missing information, no clear indication is provided that there were errors on the form or which field(s) are invalid/erroneous.

Currently, the focus is shifted to the first input with an error, and that field, when blurred (not in keyboard focus), has a red outline instead of the blue focus outline. No error indication is provided in text, nor is any announcement (alert) made available to screen readers.

Solution

Provide clear error messaging and handling so that visible error messages are shown as well as programmatically tied to an error summary block and/or directly to the fields that the errors pertain to.

For examples and more information, see:

- W3C WAI Web Accessibility Tutorials > Forms > User Notifications
- [WebAIM: Usable and Accessible Form Validation](#)

Screenshot(s)

*Indicates required fields for submitting request.

Name

Daytime Phone Number *

Evening Phone Number *

Email Address

Only indication of error is color change in outline around input fields.

Figure 68a: - Screenshot of error indicator

Required fields are only indicted by an asterisk (CLT-69)

Components	007 - charlottenc.gov - TRASH - Schedule a Bulky Item Pickup
Affected community	Vision
WCAG Success Criterion	3.3.2 Labels or Instructions (A)
WCAG Level	A
Priority	High

Description

Required fields are only indicated using the font icon for an asterisk (*). This is announced as "star" by screen reader and does not clearly and programmatically indicate that the field is required.

Current code

```
<label for="addressString" id="addressStringLabel">Location Address <i class="fa fa-asterisk"></i></label>
...
<input aria-activedescendant="" aria-autocomplete="both" aria-expanded="false" aria-
haspopup="listbox" aria-owns="rbt-menu-m9oulj25yr" autocomplete="nope"
placeholder="Enter a Solid Waste Serviced Address..." role="combobox" class="rbt-
input-main form-control rbt-input " type="text" value="">
```

Solution

Use the required attribute on the <input> to programmatically signify the field is required.

Hide the asterisk from screen readers in the label using the aria-hidden attribute.

Recommended code

```
<label for="addressString" id="addressStringLabel">Location Address <span aria-
hidden="true"><i class="fa fa-asterisk"></i></span></label>
...
<input required aria-activedescendant="" aria-autocomplete="both" aria-
expanded="false" aria-haspopup="listbox" aria-owns="rbt-menu-m9oulj25yr"
autocomplete="nope" placeholder="Enter a Solid Waste Serviced Address..."
role="combobox" class="rbt-input-main form-control rbt-input " type="text" value="">
```

Screenshot(s)



*The address must be a Solid Waste serviced address.

Location Address *

Enter a Solid Waste Serviced Address...

Next ➔

Figure 69a: - Screenshot of required field

File picker not keyboard accessible (CLT-70)

Components	005 - charlottenc.gov - Bicyclist Report on Aggressive Drivers
Affected community	Vision, Motor
WCAG Success Criterion	2.1.1 Keyboard (A)
WCAG Level	A
Priority	Highest

Description

The "Upload File(s)" file-picker widget is entirely inaccessible for keyboard-only users. This is caused by use of non-semantic markup (`<div>`s and ``s) instead of links, buttons, and other keyboard-appropriate controls.

Solution

Use a native file upload function instead of relying on "fancy" third-party widgets that have neglected accessibility during their design and development.

Screenshot(s)

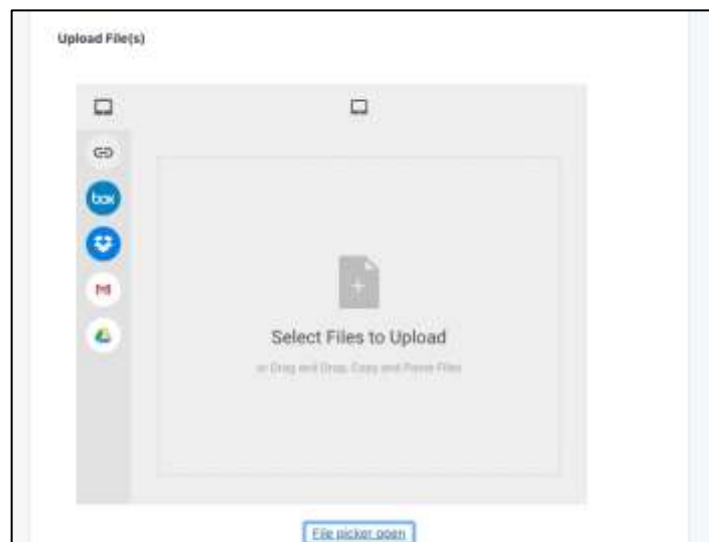


Figure 70a: - Screenshot of file picker

Incorrect ARIA roles (CLT-71)

Components	008 - charlottenc.gov - Serve on Advisory Boards - Apply for a Board
Affected community	Vision
WCAG Success Criterion	4.1.2 Name, Role, Value (A)
WCAG Level	A
Priority	High

Description

ARIA roles have been applied to several elements using incorrect values, such as:

- header
- save, print and submit
- main application area
- navigation menu

These are not actual roles described in the ARIA specifications. Assistive technologies will not be able to parse these roles. Screen readers will not know what to do with these elements.

Solution

Use native HTML elements. If roles must be defined with ARIA, only use the elements allowed.

For a list of ARIA roles, please see: <https://www.w3.org/WAI/PF/aria/roles>

Screenshot(s)

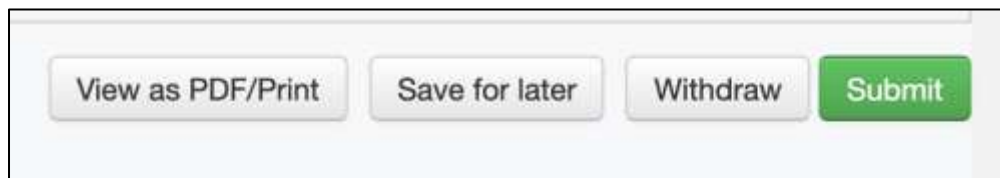


Figure 71a: - Screenshot of element with incorrect ARIA roles

Inline validation indicates error on field as soon as focused (CLT-72)

Components	008 - charlottenc.gov - Serve on Advisory Boards - Apply for a Board
Affected community	Cognitive
WCAG Success Criterion	3.3.1 Error Identification (A)
WCAG Level	A
Priority	High

Description

The required fields display a red border as soon as focus is set on them, before anything is typed in. This may be perceived as a "stop" sign for many users with cognitive impairments. Red means stop.

Solution

Do not throw an error before the field has been completed.

Screenshot(s)

A screenshot of a web form titled "Profile". It contains three input fields: "First Name *", "Last Name *", and "Email Address *". The "First Name" field is highlighted with a thick red border, indicating an inline validation error. The other fields have standard gray borders. The form is set against a light gray background.

Figure 72a: - Screenshot of inline validation indicator

Redundant alt text with adjacent text on page (CLT-73)

Components	005 - charlottenc.gov - Bicyclist Report on Aggressive Drivers
Affected community	Vision
WCAG Success Criterion	1.1.1 Non-text content (A)
WCAG Level	A
Priority	Low

Description

The logo for the City of Charlotte has an `alt` value of "City of Charlotte" which is appropriate. However, there is text adjacent to the logo in a heading (`h1`) that reads the same thing.

Current code

```

```

Solution

Consider using blank `alt` text (keep the `alt` attribute, but set its value to be empty, no space) or remove the `h1` text entirely.

Recommended code

```

```

Screenshot(s)



Figure 73a: - Screenshot of image with redundant alt text

Additional information not programmatically available (CLT-74)

Components	008 - charlottenc.gov - Serve on Advisory Boards - Apply for a Board
Affected community	Vision
WCAG Success Criterion	3.3.2 Labels or Instructions (A)
WCAG Level	A
Priority	High

Description

Additional information or requirements are displayed before some form inputs. This information isn't programmatically associated with the inputs and may easily be missed by screen reader users who navigate the form with their assistive technology in "forms mode".

This is particularly important for the information about the boards that have, or don't have a vacancy.

Current code

```
<p>
<b>Board Members must be residents of Mecklenburg County. Please provide your home
address below:</b>
</p>
...
<input class="address_1" data-jurisdiction-question-id="14198" data-readable-
name="Street Address" id="answers_for_14198" name="address_line_1" placeholder=""
required="" type="textfield" value="">
```

Solution

Use the `aria-describedby` attribute pointing to the `id` of the additional information.

Recommended code

```
<p id="address-info">
<b>Board Members must be residents of Mecklenburg County. Please provide your home
address below:</b>
</p>
...
<input aria-describedby="address-info" class="address_1" data-jurisdiction-question-
id="14198" data-readable-name="Street Address" id="answers_for_14198"
name="address_line_1" placeholder="" required="" type="textfield" value="">
```

Screenshot(s)

Board Members must be residents of Mecklenburg County. Please provide your home address below:

Street Address *


City *

State *

Postal Code *

Figure 74a: - Screenshot of additional information for form inputs

which Boards would you like to apply for?

 This board does not have a vacancy

Bechtler Arts Foundation Board ▼

Figure 74b: - Screenshot of additional information for form inputs

Duplicate label text for different elements (CLT-75)

Components	008 - charlottenc.gov - Serve on Advisory Boards - Apply for a Board
Affected community	Vision
WCAG Success Criterion	3.3.2 Labels or Instructions (A), 4.1.2 Name, Role, Value (A)
WCAG Level	A
Priority	High

Description

The primary and alternate phone number input fields have an `aria-label` of "telephone number input". This is not a very helpful label text, particularly as it is applied to both types of phone numbers. An assistive technology user may not be able to know the input's purpose or interact with it. Screen readers would announce "telephone number input" for both fields. Speech input users could not command their assistive software to "click Primary phone".

Additionally, a typo has been found in the ARIA attribute. It should be a hyphen rather than an underscore.

Current code

```
<input aria_label="telephone number input" class="phone" data-jurisdiction-question-id="14204" data-readable-name="Primary Phone" id="answers_for_14204" name="primary_phone" placeholder="(415) 123-4567" required="" type="textfield" value="" maxlength="14" autocomplete="off">
...
<input aria_label="telephone number input" class="phone" data-jurisdiction-question-id="14205" data-readable-name="Alternate Phone" id="answers_for_14205" name="alternate_phone" placeholder="(415) 123-4567" type="textfield" value="" maxlength="14" autocomplete="off">
```

Solution

Use distinct `aria-label` for distinct input fields.

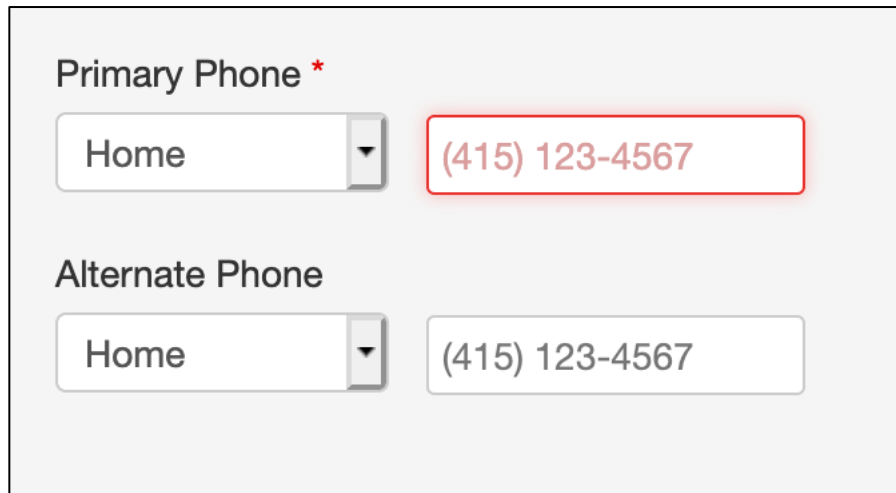
Write the attribute name properly.

Recommended code

```
<input aria-label="Primary phone" class="phone" data-jurisdiction-question-id="14204" data-readable-name="Primary Phone" id="answers_for_14204" name="primary_phone" placeholder="(415) 123-4567" required="" type="textfield" value="" maxlength="14" autocomplete="off">
```

```
...  
<input aria-label="Alternate phone" class="phone" data-jurisdiction-question-  
id="14205" data-readable-name="Alternate Phone" id="answers_for_14205"  
name="alternate_phone" placeholder="(415) 123-4567" type="textfield" value=""  
maxlength="14" autocomplete="off">
```

Screenshot(s)



The screenshot shows a form with two sections. The first section is titled "Primary Phone" with a red asterisk. It contains a dropdown menu with "Home" selected and a text input field containing "(415) 123-4567". The text input field is highlighted with a red border. The second section is titled "Alternate Phone" and also contains a dropdown menu with "Home" selected and a text input field containing "(415) 123-4567".

Figure 75a: - Screenshot of elements with duplicate label text

Insufficient color contrast for links (CLT-76)

Components	010 - charlottenc.gov - Charlotte Future (carousel and interactive map)
Affected community	Vision
WCAG Success Criterion	1.4.3 Contrast (Minimum) (AA)
WCAG Level	AA
Priority	High

Description

For the event pagination, the foreground color is #777777 and the background color is #FFFFFF. This is a color contrast ratio of 4.47:1, which makes it difficult for people with low vision to read the link text. Minimum contrast should be 4.5:1.

Solution

Ensure the color contrast between link foreground and background colors is at least 4.5:1.

Screenshot(s)



Figure 76a: - Screenshot of elements with poor color contrast

Use of tabindex="-1" (CLT-77)

Components	009 - charlottenc.gov - HR - Jobs - See Available Jobs
Affected community	Vision
WCAG Success Criterion	2.1.1 Keyboard (A)
WCAG Level	A
Priority	Medium

Description

The filtering options menu contains the `tabindex` attribute with a value of `-1` and removes the element from the default navigation flow.

Current code

```
<a tabindex="-1" role="treeitem" aria-setsize="6" aria-posinset="1" aria-level="2"
  id="#ICSetFieldHRS_APP_SCHJOB.TREECTLEVENT.S3" class="PTGBLFACET"

href="javascript:submitAction_win0(document.win0, '#ICSetFieldHRS_APP_SCHJOB.TREECTLE
VENT.S3');">Engineering &
  Property Mg.. (9)</a>
```

Solution

Do not use the `tabindex` attribute with a value of `-1` unless absolutely necessary to conduct programmatic manipulation of focus order.

Recommended code

```
<a role="treeitem" aria-setsize="6" aria-posinset="1" aria-level="2"
  id="#ICSetFieldHRS_APP_SCHJOB.TREECTLEVENT.S3" class="PTGBLFACET"

href="javascript:submitAction_win0(document.win0, '#ICSetFieldHRS_APP_SCHJOB.TREECTLE
VENT.S3');">Engineering &
  Property Mg.. (9)</a>
```

Linked issues

- CLT-79

Screenshot(s)



Figure 77a: - Screenshot of elements with negative tabindex

Purpose of select box unclear (CLT-78)

Components	008 - charlottenc.gov - Serve on Advisory Boards - Apply for a Board
Affected community	Vision, Hearing, Cognitive, Motor
WCAG Success Criterion	3.3.2 Labels or Instructions (A)
WCAG Level	A
Priority	High

Description

At the end of the form, a `<select>` is provided asking to "Select one of the following", with options of "answer 1", "answer 2", and "answer 3". The purpose of the question is unclear and the answers aren't self-explanatory either. People will not know what is expected of them.

Current code

```
<div class="citizen-facing">
<label for="answers_for_42842">
Select one of the following

</label>
<select class="" data-jurisdiction-question-id="42842" data-readable-name="Select
one of the following" id="answers_for_42842"
name="select_c46ddc72_baaa_4131_9e64_9c3e260fead5" placeholder="">
<option></option>
<option value="answer 1">
answer 1
</option>
<option value="answer 2">
answer 2
</option>
<option value="answer 3">
answer 3
</option>
</select>
<span class="help-block"></span>
</div>
```

Solution

Make sure the label and options for the `<select>` clearly identify the purpose of the question and the answers.

Screenshot(s)



Select one of the following

answer 1

A screenshot of a web form element. It features a light gray rectangular container. Inside the container, at the top, is the text "Select one of the following" in a dark gray font. Below this text is a white rectangular input field with a thin gray border. The input field contains the text "answer 1". To the right of the input field is a small, light gray square button with a dark gray downward-pointing triangle (dropdown arrow) inside it.

Figure 78a: - Screenshot of select box

Positive tabindex values used (CLT-79)

Components	009 - charlottenc.gov - HR - Jobs - See Available Jobs
Affected community	Vision, Motor
WCAG Success Criterion	2.1.1 Keyboard (A)
WCAG Level	A
Priority	High

Description

Positive `tabindex` attributes have been used, forcing a specific tab order. This means keyboard-only users may have difficulty navigating the page. It may even mean they can't reach some elements of the page at all.

Solution

Remove the positive `tabindex` attributes. Let the natural tab order prevail.

Linked issues

- CLT-77

Screenshot(s)

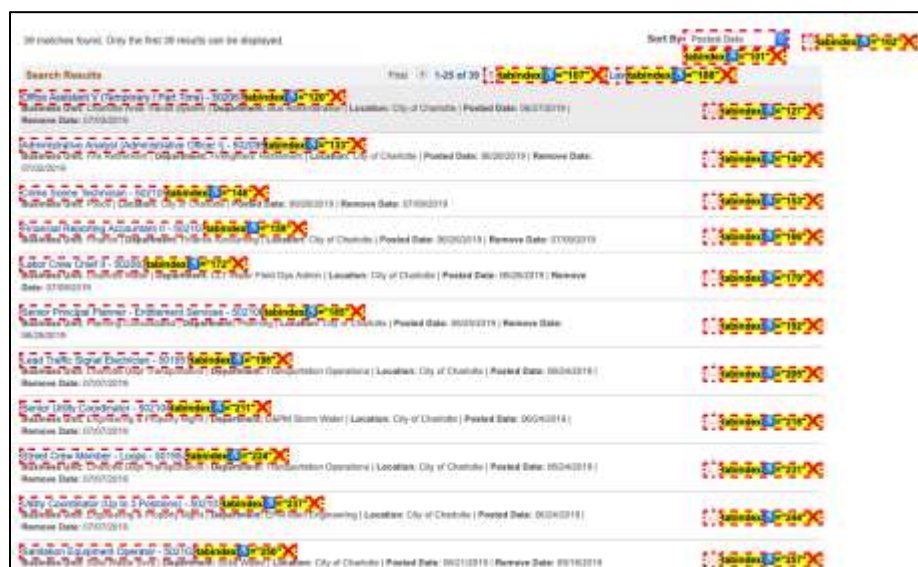


Figure 79a: - Screenshot of elements with positive tabindex

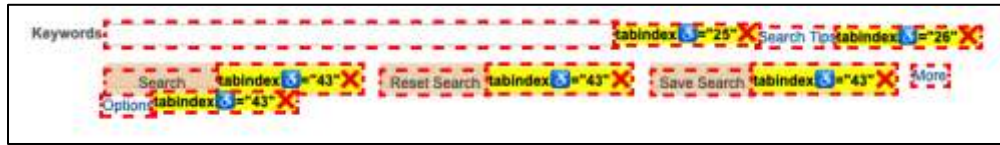


Figure 79b: - Screenshot of elements with positive tabindex



Figure 79c: - Screenshot of elements with positive tabindex

HTML table used for layout (CLT-81)

Components	002 - charlottenc.gov - Homepage, 009 - charlottenc.gov - HR - Jobs - See Available Jobs
Affected community	Vision
WCAG Success Criterion	1.3.1 Info and Relationships (A)
WCAG Level	A
Priority	High

Description

The filtering options have been laid out using an HTML table. This also occurs within the tabbed interface content.

These tables provide semantically meaningful information for screen reader users when they display data. For layout, each row and column and cell will be announced, creating a difficult experience for screen reader users.

Current code

```
<table cellpadding="0" cellspacing="0" cols="1" class="MUTENBO" width="260">
  <tbody>
    <tr>
      ...
```

Solution

Do not use HTML tables for layout. Use semantically meaningful code instead.

If that is not an option, apply `role="presentation"` to the table so assistive technologies know to ignore it.

Recommended code

```
<table role="presentation" cellpadding="0" cellspacing="0" cols="1" class="MUTENBO"
width="260">
  <tbody>
    <tr>
      ...
```

Screenshot(s)

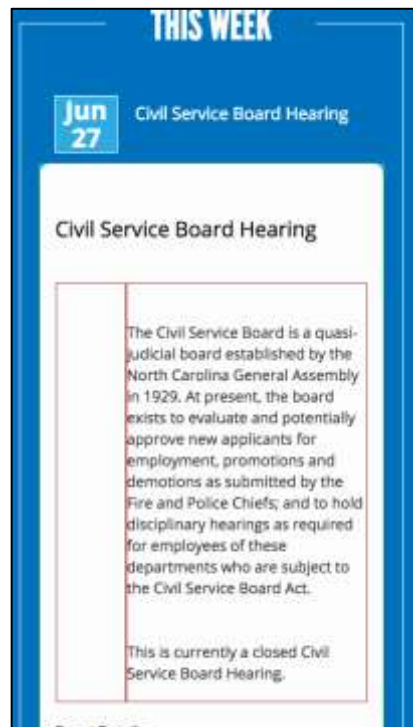


Figure 81a: - Screenshot of layout done with table

table.PSLEVEL1GRIDLABEL 884 x 20	be displayed.	Sort By: Posted Date
Search Results	First	11 of 11
Lead Traffic Social Electricity - 501951		

Figure 81d: - Screenshot of table used for layout

Navigation overlay modal does not trap keyboard focus (CLT-82)

Components	001 - charlottenc.gov - Header/Footer, 010 - charlottenc.gov - Charlotte Future (carousel and interactive map)
Affected community	Vision, Cognitive, Motor
WCAG Success Criterion	2.1.2 No Keyboard Trap (A)
WCAG Level	A
Priority	High

Description

The modal overlay for the main site navigation — starting at 170% and in views less than approximately 766px wide — does not trap keyboard focus. The user is able to tab through other parts of the page.

This also occurs on interior page navigation.

This will create problems for screen-reader, keyboard or other assistive-technology users who launch the modal but may not be able to operate it or close out of it. If these users can tab out of the modal they are easily confused about their location within the page.

Solution

1. Ensure that keyboard focus is set by default to the first heading element in the modal
2. The focus should be kept inside the modal until the visitor closes it
3. When the modal is closed, the focus should be returned to the button element that triggered the modal
4. Provide an escape route to get out of the modal. There should be options for keyboard-only and mouse input.

Refer to the W3C modal example for a thorough code sample and explanation on how to build accessible modals: <https://www.w3.org/TR/wai-aria-practices/examples/dialog-modal/dialog.html>

Screenshot(s)



Figure 82a: - Screenshot of modal that doesn't trap keyboard focus



Figure 82b: - Screenshot of modal that doesn't trap keyboard focus

Label refers to id that doesn't exist (CLT-83)

Components	008 - charlottenc.gov - Serve on Advisory Boards - Apply for a Board
Affected community	Vision
WCAG Success Criterion	3.3.2 Labels or Instructions (A)
WCAG Level	
Priority	High

Description

A `<label>` has been used but the `for` attribute refers to an `id` that isn't present on the page. This in effect creates an orphaned label and could cause issues for screen reader users.

Current code

```
<label for="boards">
Which Boards would you like to apply for?
<span class="asterisk">*</span>
</label>
```

Solution

Do not use a `<label>` for that text as it replicates already existing `aria-label`

Screenshot(s)

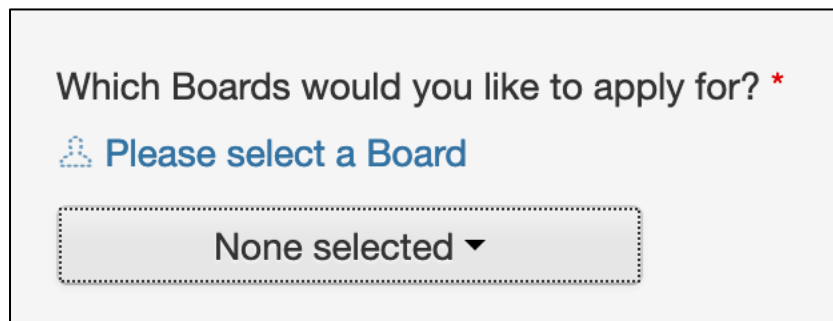


Figure 83a: - Screenshot of orphaned label

Image from CSS rules without alt text (CLT-84)

Components	009 - charlottenc.gov - HR - Jobs - See Available Jobs
Affected community	Vision
WCAG Success Criterion	1.1.1 Non-text content (A)
WCAG Level	A
Priority	Medium

Description

The logo image appears due to generated content through a CSS property. This logo is announced as "image" in screen readers. Users will not have any context as to what the image is about.

Current code

```
#clthdrlogo:before {  
  content: url(/cs/ER92PRD/cache/CLT_CROWN_WHITE92_1.PNG);  
  left: 0px;  
  position: relative;  
  top: 0px;  
}
```

Solution

Use an inline image with alt text.

Recommended code

```

```

Screenshot(s)



Figure 84a: - Screenshot of image loaded from CSS

Board specific question(s) break flow of answering (CLT-85)

Components	008 - charlottenc.gov - Serve on Advisory Boards - Apply for a Board
Affected community	Vision, Motor
WCAG Success Criterion	1.3.2 Meaningful Sequence (A)
WCAG Level	A
Priority	Medium

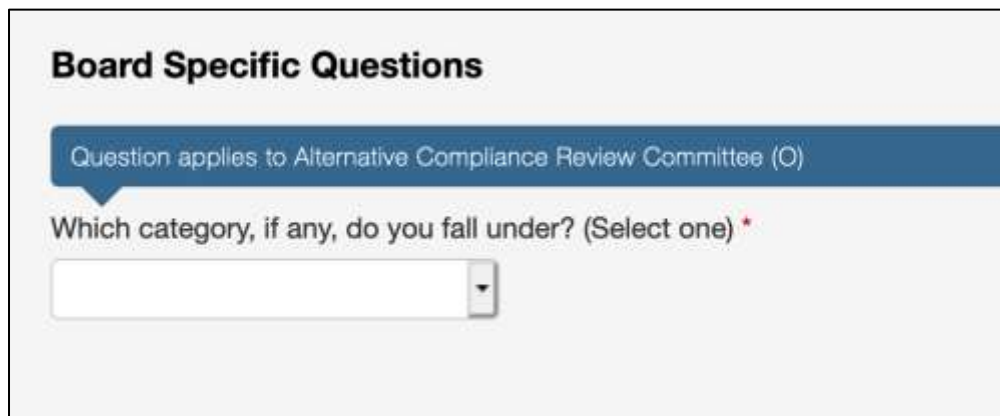
Description

Some boards have specific questions asked. These questions appear at the bottom of the form. When a user selects a board that has specific questions, a link is provided to go directly to those questions. This bypasses multiple questions about the individual's experience with other boards or employment. It would be easy for them to jump straight to the board specific question and submit before the other, required, questions are answered.

Solution

Consider revisiting the workflow. Either place the block of questions for board specific questions immediately after the question about which board the person wants to apply for, or place the question about which board the person wants to apply for at the end of the form.

Screenshot(s)



Board Specific Questions

Question applies to Alternative Compliance Review Committee (O)

Which category, if any, do you fall under? (Select one) *

Figure 85a: - Screenshot of board specific question

Local links not working (CLT-86)

Components	008 - charlottenc.gov - Serve on Advisory Boards - Apply for a Board
Affected community	Vision, Motor
WCAG Success Criterion	2.1.1 Keyboard (A)
WCAG Level	A
Priority	High

Description

There are jump links on the left sidebar to go to different sections of the form. Two of these links are not working. This will impact on keyboard-only users, sighted or not.

Current code

```
<a href="#Board Specific Questions">  
<span class="menu-item-text">  
Board Specific Questions  
</span>  
<span class="saved-text" style="display:none">Saved!</span>  
</a>
```

Solution

Ensure local links function properly.

Screenshot(s)

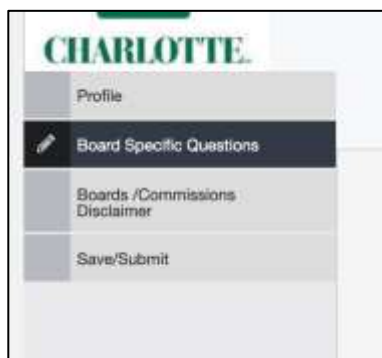


Figure 86a: - Screenshot of in-page links

Can't submit with only the keyboard (CLT-87)

Components	008 - charlottenc.gov - Serve on Advisory Boards - Apply for a Board
Affected community	Vision, Motor
WCAG Success Criterion	2.1.1 Keyboard (A)
WCAG Level	A
Priority	High

Description

The submit button is a link without an `href`. This means it cannot be activated via keyboard.

Current code

```
<a class="btn btn-success" data-toggle="button" data-  
url="/boards/apply/1449225/submit" id="continue" role="button" tabindex="0"  
type="button">  
Submit  
</a>
```

Solution

As this link performs the role of a button, use the `<button>` element instead. If that isn't possible, ensure the element can be activated with the keyboard. Typically, adding an `href` element would accomplish this.

Recommended code

```
<button class="btn btn-success" data-toggle="button" data-  
url="/boards/apply/1449225/submit" id="continue">  
Submit  
</button>
```

Screenshot(s)

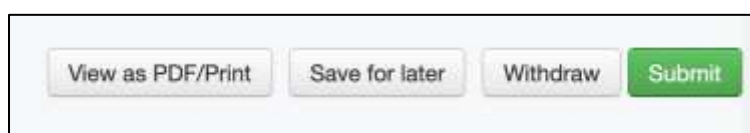


Figure 87a: - Screenshot of submit button

Vertical bar character is announced with title (CLT-88)

Components	009 - charlottenc.gov - HR - Jobs - See Available Jobs
Affected community	Vision
WCAG Success Criterion	1.3.1 Info and Relationships (A)
WCAG Level	A
Priority	Low

Description

The vertical bar character, "|", is announced by the screen reader with the rest of the main heading.

Current code

```
<font color="white"> | CITY OF CHARLOTTE</font>
```

Solution

I would recommend removing the vertical bar character. However, if the character is needed, use `aria-hidden` attribute to keep it from being announced by screen readers.

Recommended code

```
<span style="color:white;"> <span aria-hidden="true">|</span> CITY OF  
CHARLOTTE</span>
```

Screenshot(s)



Figure 88a: - Screenshot of vertical bar character

List of error messages displayed at the bottom of the page (CLT-89)

Components	008 - charlottenc.gov - Serve on Advisory Boards - Apply for a Board
Affected community	Vision, Motor
WCAG Success Criterion	3.3.1 Error Identification (A)
WCAG Level	A
Priority	High

Description

If the form is submitted with errors in it, a list of the errors is displayed at the bottom of the form. But keyboard focus is sent back to the top of the page. Keyboard-only users will have a more difficult time navigating the page and errors.

Current code

```
<h4>Please fill out all required fields and correct any errors before submitting.</h4>
```

Solution

Display the error messages at the top of the form. Use `tabindex="-1"` on the heading, and set focus to it when the page reloads.

Recommended code

```
<h4 tabindex="-1">Please fill out all required fields and correct any errors before submitting.</h4>
```


Screenshot(s)

Please fill out all required fields and correct any errors before submitting.

- Last Name
- Email Address
- Street Address
- City
- State
- Postal Code
- Is your mailing address the same as your home address?
- Provide preferred mailing address below; if your mailing address is the same as your home address, enter N/A
- Primary Phone
- What City Council district do you live in?
- Ethnicity
- Political Party
- Gender
- Date of Birth
- List any boards you are currently serving on:
- List any boards you have served on in the past:
- Current Employer:
- Has any formal charge of professional misconduct ever been sustained against you in any jurisdiction?
- Do you have any personal or business interest that could create a conflict (either real or perceived) if appointed?
- Have you completed the City's Civic Leadership Academy?
- How did you find out about the Charlotte Boards and Commissions vacancies?
- Please check this box after reading the above disclosure statement:
- Please type your name below:

[View as PDF/Print](#) [Save for later](#) [Withdraw](#) [Submit](#)

Figure 89a: - Screenshot of list of error messages

Content is clipped at 200% (CLT-90)

Components	009 - charlottenc.gov - HR - Jobs - See Available Jobs
Affected community	Vision
WCAG Success Criterion	1.4.4 Resize Text (AA)
WCAG Level	AA
Priority	Medium

Description

When the text is resized to 200%, the search form and job listings are clipped. The "Sign In" and "Register Here" items are hidden. The entire page is only viewable when the viewport is at least 1250 pixels wide.

Solution

Ensure that there is white space remaining around each text element when text is resized. Don't force the width of the widget to remain the same. Allow dynamic reflow of content.

Screenshot(s)

Job Search

Keywords

Filter by

Business Unit

Engineering & Property Mg..(9)

Aviation (8)

Charlotte Dept Transporta..(6)

Charlotte Area Transit Sy..(3)

Charlotte Water (3)

More...

Job Posted In

2019 (38)

2018 (1)

39 matches found. Only the first 3

Search Results

Office Assistant V (Temporary / Ps

Business Unit: Charlotte Area Trans

Remove Date: 07/03/2019

Administrative Analyst (Administra

Business Unit: Fire Retirement | De

07/02/2019

Crime Scene Technician - 502105

Business Unit: Police | Location: C

Financial Reporting Accountant II

Business Unit: Finance | Departme

Figure 90a: - Screenshot of text resized

179

Error message list not linked to specific erroneous fields (CLT-91)

Components	008 - charlottenc.gov - Serve on Advisory Boards - Apply for a Board
Affected community	Vision, Motor
WCAG Success Criterion	3.3.1 Error Identification (A)
WCAG Level	A
Priority	High

Description

The list of error messages clearly states what errors are on the form. But each list item isn't linked to the specific erroneous field it discusses. This means keyboard-only users, sighted or not, must navigate through the entire form to reach the erroneous input to fix it.

Current code

```
<li> Last Name</li>
```

Solution

Link each list item for the problematic fields to the erroneous field.

Recommended code

```
<li> <a href="#answers_for_14193">Last Name</a></li>
```

Screenshot(s)

Please fill out all required fields and correct any errors before submitting.

- Last Name
- Email Address
- Street Address
- City
- State
- Postal Code
- Is your mailing address the same as your home address?
- Provide preferred mailing address below; if your mailing address is the same as your home address, enter N/A
- Primary Phone
- What City Council district do you live in?
- Ethnicity
- Political Party
- Gender
- Date of Birth
- List any boards you are currently serving on:
- List any boards you have served on in the past:
- Current Employer:
- Has any formal charge of professional misconduct ever been sustained against you in any jurisdiction?
- Do you have any personal or business interest that could create a conflict (either real or perceived) if appointed?
- Have you completed the City's Civic Leadership Academy?
- How did you find out about the Charlotte Boards and Commissions vacancies?
- Please check this box after reading the above disclosure statement:
- Please type your name below:

[View as PDF/Print](#) [Save for later](#) [Withdraw](#) [Submit](#)

Figure 91a: - Screenshot of error messages

Unclear error messages (CLT-92)

Components	008 - charlottenc.gov - Serve on Advisory Boards - Apply for a Board
Affected community	Vision
WCAG Success Criterion	3.3.1 Error Identification (A)
WCAG Level	A
Priority	High

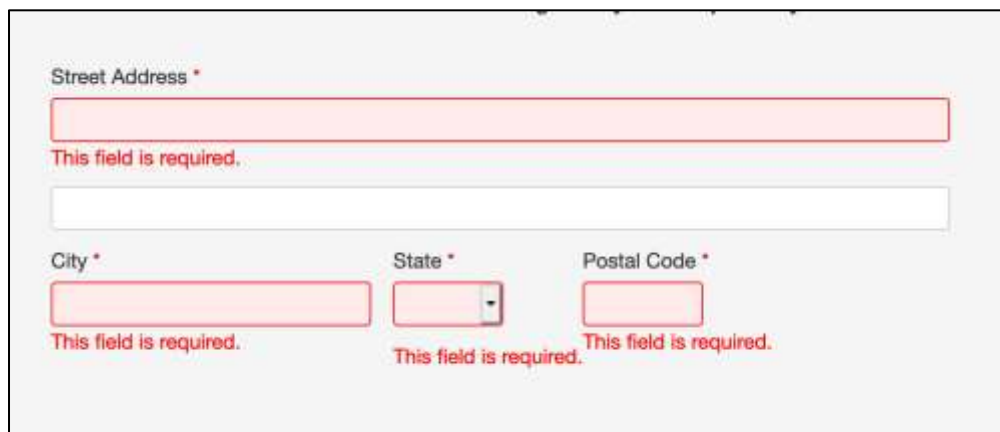
Description

There are multiple messages with the same text "this field is required". This isn't helpful if an individual cannot see which field the message applies to.

Solution

Ensure each error message clearly defines which erroneous field is targeted.

Screenshot(s)



The screenshot shows a form with the following fields and error messages:

- Street Address ***: A text input field with a red border and the error message "This field is required." below it.
- City ***: A text input field with a red border and the error message "This field is required." below it.
- State ***: A dropdown menu with a red border and the error message "This field is required." below it.
- Postal Code ***: A text input field with a red border and the error message "This field is required." below it.

Figure 92a: - Screenshot of unclear error messages

New slides are not announced by screen readers (CLT-93)

Components	010 - charlottenc.gov - Charlotte Future (carousel and interactive map)
Affected community	Vision
WCAG Success Criterion	3.2.2 On Input (A)
WCAG Level	A
Priority	High

Description

When using the carousel navigation, there is no information given to screen readers about the new slide being loaded. A screen reader user may trigger the links and not be aware that anything changed on the page.

Solution

Ensure that carousels are keyboard and screen-reader friendly. Changes in content/slides should be announced. Please refer to this tutorial about accessible carousels for further information:

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

Screenshot(s)

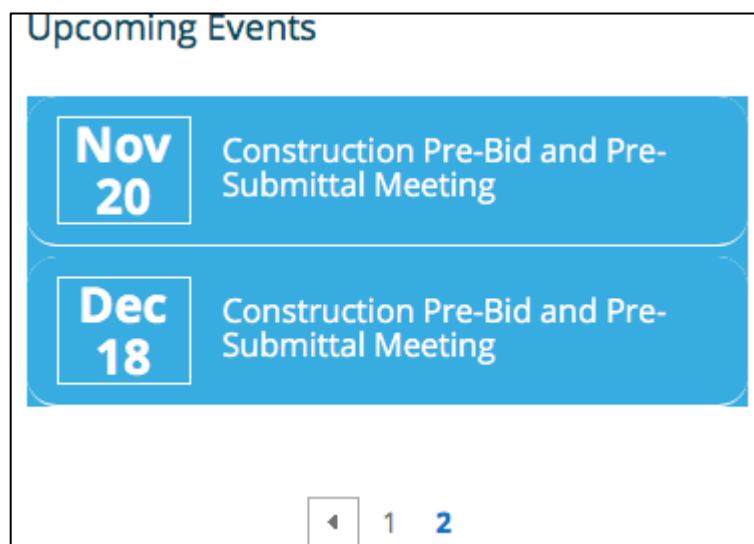


Figure 93a: - Screenshot of slide

Incorrect use of navigation role (CLT-94)

Components	009 - charlottenc.gov - HR - Jobs - See Available Jobs
Affected community	
WCAG Success Criterion	4.1.2 Name, Role, Value (A)
WCAG Level	A
Priority	Medium

Description

The "Sign In" and "Register Here" links and form buttons are identified as `role="navigation"`. This is an incorrect use of the navigation role – and they should not be marked with a `role=navigation`. The navigation role is used to identify a collection of navigation elements. See the definition of the [WAI-ARIA 1.1 navigation landmark](#) for more details.

Current code

```
<div name="SS_SIGNINOUT" id="SS_SIGNINOUT"
  style="display:inline-block;position:absolute;bottom:6px;right:0;white-
space:nowrap;" role="Navigation">
...
</div>
```

Solution

Remove the `role="navigation"` on the "Sign In" and "Register Here" and form buttons.

Recommended code

```
<div name="SS_SIGNINOUT" id="SS_SIGNINOUT"
  style="display:inline-block;position:absolute;bottom:6px;right:0;white-
space:nowrap;">
...
</div>
```


Screenshot(s)

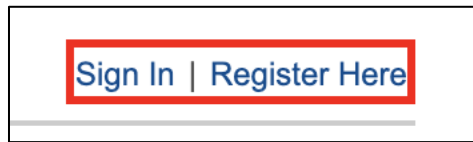


Figure 94a: - Screenshot of links



Figure 94b: - Screenshot of links

Role="main" placed on wrong area (CLT-95)

Components	009 - charlottenc.gov - HR - Jobs - See Available Jobs
Affected community	Vision
WCAG Success Criterion	1.3.1 Info and Relationships (A)
WCAG Level	A
Priority	Medium

Description

The `role="main"` attribute is placed on the outer most HTML table. This attribute with the role value should be placed on the main content of the site – in this case, the job listings.

The main landmark role is used to indicate the primary content of a document.

Current code

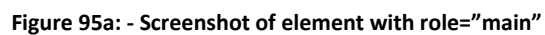
```
<table class="PSPAGECONTAINER" role="main" aria-labelledby="app_label">
...
</table>
```

Solution

I would recommend moving the `role="main"` to the `div` element that wraps around job listings on the page.

Recommended code

```
<div role="main" id="win0divHRS_SCH_WRK_HRS_SCH_GRPBOXCR">
...
</div>
```



Frames have no titles and wrong DOCTYPE (CLT-96)

Components	009 - charlottenc.gov - HR - Jobs - See Available Jobs
Affected community	Vision
WCAG Success Criterion	4.1.2 Name, Role, Value (A)
WCAG Level	A
Priority	High

Description

The frame elements do not use the title attribute. The title attribute is used to describe the frame's content.

Also, this document uses HTML5 doctype. The frame element is deprecated in HTML5.

Current code

```
<frameset rows="65,*">
<frame name="UniversalHeader" scrolling="no" frameborder="no" noresize
src="https://careers.charlottenc.gov/psc/ER92PRD/CAREERS/HRMS/s/WEBLIB_PORTAL.PORTAL
_HEADER.FieldFormula.IScript_UniHeader_Frame?PortalActualURL=https%3a%2f%2fcareers.c
harlottenc.gov%2fpsc%2fER92PRD%2fCAREERS%2fHRMS%2fc%2fHRS_HRAM.HRS_APP_SCHJOB.GBL&Po
rtalRegistryName=CAREERS&PortalServletURI=https%3a%2f%2fcareers.charlottenc.gov%2fps
p%2fER92PRD%2f&PortalURI=https%3a%2f%2fcareers.charlottenc.gov%2fpsc%2fER92PRD%2f&Po
rtalHostNode=HRMS&PortalIsPagelet=true&NoCrumbs=yes">
</frame>
<frame name="TargetContent" scrolling="yes" frameborder="no" noresize
src="https://careers.charlottenc.gov/psc/ER92PRD/CAREERS/HRMS/c/HRS_HRAM.HRS_APP_SCH
JOB.GBL?PortalActualURL=https%3a%2f%2fcareers.charlottenc.gov%2fpsc%2fER92PRD%2fCARE
ERS%2fHRMS%2fc%2fHRS_HRAM.HRS_APP_SCHJOB.GBL&PortalRegistryName=CAREERS&PortalServle
tURI=https%3a%2f%2fcareers.charlottenc.gov%2fpsp%2fER92PRD%2f&PortalURI=https%3a%2f%
2fcareers.charlottenc.gov%2fpsc%2fER92PRD%2f&PortalHostNode=HRMS&NoCrumbs=yes">
</frame>
</frameset>
```

Solution

Add a title to each of the frame elements.

Switch the DOCTYPE that allows for frame elements as shown here

<https://www.w3.org/TR/html4/sgml/framesetdtd.html>

Recommended code

```
<frameset rows="65,*">
```

```

<frame title="Job Search Header" name="UniversalHeader" scrolling="no"
frameborder="no" noresize
src="https://careers.charlottenc.gov/psc/ER92PRD/CAREERS/HRMS/s/WEBLIB_PORTAL.PORTAL_HEADER.FieldFormula.IScript_UniHeader_Frame?PortalActualURL=https%3a%2f%2fcareers.c
harlottenc.gov%2fpsc%2fER92PRD%2fCAREERS%2fHRMS%2fc%2fHRS_HRAM.HRS_APP_SCHJOB.GBL&Po
rtalRegistryName=CAREERS&PortalServletURI=https%3a%2f%2fcareers.charlottenc.gov%2fps
p%2fER92PRD%2f&PortalURI=https%3a%2f%2fcareers.charlottenc.gov%2fpsc%2fER92PRD%2f&Po
rtalHostNode=HRMS&PortalIsPagelet=true&NoCrumbs=yes">
</frame>
<frame title="Job Search Form and Listings" name="TargetContent" scrolling="yes"
frameborder="no" noresize
src="https://careers.charlottenc.gov/psc/ER92PRD/CAREERS/HRMS/c/HRS_HRAM.HRS_APP_SCH
JOB.GBL?PortalActualURL=https%3a%2f%2fcareers.charlottenc.gov%2fpsc%2fER92PRD%2fCARE
ERS%2fHRMS%2fc%2fHRS_HRAM.HRS_APP_SCHJOB.GBL&PortalRegistryName=CAREERS&PortalServle
tURI=https%3a%2f%2fcareers.charlottenc.gov%2fpsp%2fER92PRD%2f&PortalURI=https%3a%2f%
2fcareers.charlottenc.gov%2fpsc%2fER92PRD%2f&PortalHostNode=HRMS&NoCrumbs=yes">
</frame>
</frameset>

```

Screenshot(s)

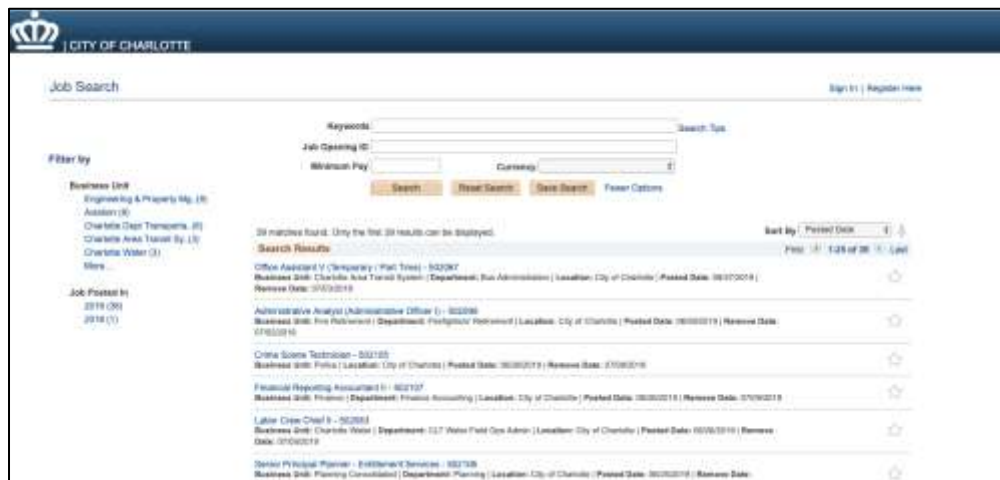


Figure 96a: - Screenshot of frame

Tab order is different from the expected visual order (CLT-97)

Components	009 - charlottenc.gov - HR - Jobs - See Available Jobs
Affected community	Motor
WCAG Success Criterion	1.3.2 Meaningful Sequence (A)
WCAG Level	A
Priority	Medium

Description

The tab order starts with "Sign In" and registration links. Then moves to the Filter list, skipping the search field. Then instead of going through the Filter list of links, the tab order jumps to the search form. Visually, this is jarring and a sighted keyboard-only user might get confused.

Solution

Ensure that keyboard tab order follows visual order. For an English-speaking site, this means from left to right and top to bottom.

Screenshot(s)



Figure 97a: - Screenshot of tab order

Modal window does not trap focus (CLT-98)

Components	009 - charlottenc.gov - HR - Jobs - See Available Jobs
Affected community	Vision, Motor
WCAG Success Criterion	2.1.2 No Keyboard Trap (A)
WCAG Level	A
Priority	High

Description

Keyboard focus is not trapped in the Search Tips modal. When displayed, the Search Tips modal window does not trap focus – allowing users to select elements outside of the modal.

This pattern could be confusing and create problems for screen reader users, keyboard users, or other assistive technology users who launch the modal but may not be able to operate it or close out of it.

Modals should trap both keyboard users and screen reader users until closed by users.

Solution

Refer to the W3C modal example for a thorough code sample and explanation on how to make the modal accessible. <https://www.w3.org/TR/wai-aria-practices/examples/dialog-modal/dialog.html>

Screenshot(s)

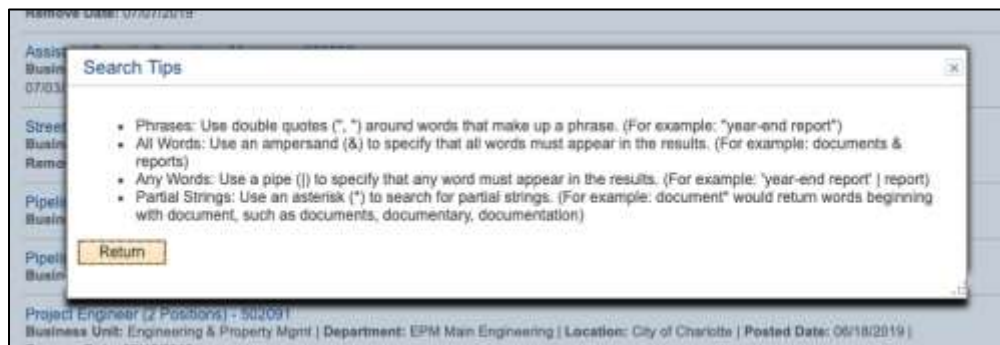


Figure 98a: - Screenshot of modal window

Changes in page not announced (CLT-99)

Components	009 - charlottenc.gov - HR - Jobs - See Available Jobs
Affected community	Vision
WCAG Success Criterion	3.2.2 On Input (A)
WCAG Level	A
Priority	High

Description

When submitting a search or selecting more options in the search or sorting job listings, changes in the page occur. For example, when submitting a search, search results are loaded – However there is no method for screen readers to know something changed. As a result screen readers cannot announce changes in pages. Screen reader users may not know something changed.

Current code

```
<div id="win0divHRS_AGNT_RSLT_IGP$0">
  ...
</div>
```

Solution

Implement an `aria-live` attribute on the block of content where content is updated.

Recommended code

```
<div id="win0divHRS_AGNT_RSLT_IGP$0" aria-live="polite">
  ...
</div>
```

Screenshot(s)



Figure 99a: - Screenshot of content changing on the page

Keywords [Search Tips](#)

Figure 99b: - Screenshot of content changing on the page

39 matches found. Only the first 39 results can be displayed.

Sort By:

First 1-25 of 39 Last

Search Results

Apprenticeship - Aviation Fleet Maintenance - 501605 Business Unit: Aviation Department: Aviation Location: City of Charlotte Posted Date: 12/04/2018 Remove Date: 07/21/2019	
Lateral Police Trainee - 501625 Business Unit: Police Department: Recruit Training Unit Location: City of Charlotte Posted Date: 01/01/2019 Remove Date: 12/11/2019	

Figure 99c: - Screenshot of content changing on the page

First 1-11 of 11 Last

Figure 99d: - Screenshot of content changing on the page

Insufficient title on iframe (CLT-100)

Components	010 - charlottenc.gov - Charlotte Future (carousel and interactive map)
Affected community	Vision
WCAG Success Criterion	2.4.1 Bypass Blocks (A), 4.1.2 Name, Role, Value (A)
WCAG Level	A
Priority	Low

Description

The title for the `iframe` is "Page Viewer." This is not descriptive of the `iframe` contents, which could make it difficult for screen-reader users to know what is in the `iframe` and whether to navigate it — either to bypass it or enter it.

Current code

```
<iframe title="Page Viewer" id="MSOPageViewerWebPart_WebPartWPQ7"
name="MSOPageViewerWebPart_WebPartWPQ7" src="https://arcg.is/2cqUqGO"
ddf_src="https://arcg.is/2cqUqGO" width="100%" height="100%" frameborder="0">
    <div class="UserGeneric">The current browser does not
support Web pages that contain the IFRAME element. To use this Web Part, you must
use a browser that supports this element, such as Internet Explorer 7.0 or
later.</div>
</iframe>
```

Solution

Ensure the `iframe` title is descriptive of the contents.

Recommended code

```
<iframe title="Interactive map of the Charlotte Future Community Investment Plan"
id="MSOPageViewerWebPart_WebPartWPQ7" name="MSOPageViewerWebPart_WebPartWPQ7"
src="https://arcg.is/2cqUqGO" ddf_src="https://arcg.is/2cqUqGO" width="100%"
height="100%" frameborder="0">
    <div class="UserGeneric">The current browser does not
support Web pages that contain the IFRAME element. To use this Web Part, you must
use a browser that supports this element, such as Internet Explorer 7.0 or
later.</div>
</iframe>
```

Linked issues

- CLT-67

Job listings table lacking scope attribute (CLT-101)

Components	009 - charlottenc.gov - HR - Jobs - See Available Jobs
Affected community	Vision
WCAG Success Criterion	1.3.1 Info and Relationships (A), 4.1.2 Name, Role, Value (A)
WCAG Level	A
Priority	Medium

Description

The HTML table shows job listings. However, the headers are not associated with their cells. The header is being used as heading and pagination bar. This means that a screen reader user will have difficulty understanding the table's content.

Solution

I would recommend restructuring the HTML tables to allow the job listings to be an HTML table – while separating the pagination and heading. Then ensure you use the `scope` attribute in column and header rows, to allow screen readers to parse the information and announce correct relationships between table cells and their headers.

Recommended code

```
<table>
  <tr>
    <th scope="col">
      Job Title
    </th>
    <th scope="col">
      Job Description
    </th>
    <td scope="col">
      Posting Date
    </td>
    <td scope="col">
      Posting Remove Date
    </td>
    <th scope="col">
      Favorite
    </th>
  </tr>
  <tr>
    <td>
      Office Assistant V (Temporary / Part Time) - 502067
```

```

        </td>
    </tr>
    <td>
        <dl>
            <dt>Business Unit:</dt>
            <dd>Charlotte Area Transit System</dd>
            <dt>Department:</dt>
            <dd>Bus Administration</dd>
            <dt>Location:</dt>
            <dd>City of Charlotte</dd>
        </dl>
    </td>
    <td>
        06/27/2019
    </td>
    <td>
        07/03/2019
    </td>
    <td>
        <a name="HRS_FAVORITE_ICN$0" id="HRS_FAVORITE_ICN$0"
            onclick="javascript:cancelBubble(event);"
            href="javascript:submitAction_win0(document.win0,'HRS_FAVORITE_ICN$0');"></a>
    </td>
</table>

```

Screenshot(s)



Figure 101a: - Screenshot of job listing

No visible focus on button (CLT-102)

Components	001 - charlottenc.gov - Header/Footer
Affected community	Vision, Cognitive, Motor
WCAG Success Criterion	2.1.1 Keyboard (A), 2.4.7 Focus Visible (AA)
WCAG Level	A
Priority	High

Description

When navigating through a page by using the tab key, sighted keyboard users can see their position on a web page in the form of a so-called focus outline or focus ring. This focus also provides sighted keyboard users with a cue that the element is interactive (e.g. a link or a button).

The live chat uses custom CSS that removes this default focus outline from the `<button>`. As such, the `button` not display visible focus when navigating with a keyboard.

No visible focus causes confusion, because sighted keyboard-only users may not know where they are on the page, in general, or what link they are on, specifically. This could also cause confusion for sighted screen-reader users (e.g. someone with low vision or with dyslexia) who navigate with the keyboard.

Current code

```
.olark-launch-button {
  font-family: -apple-system,BlinkMacSystemFont,"Segoe
UI","Roboto","Oxygen","Ubuntu","Cantarell","Fira Sans","Droid Sans","Helvetica
Neue",sans-serif !important;
  font-size: 100% !important;
  position: fixed !important;
  bottom: 15px !important;
  right: 20px !important;
  left: auto !important;
  height: 48px !important;
  width: auto !important;
  z-index: 2147483646 !important;
  background: #2d2d2d !important;
  border: 0 !important;
  border-radius: 100px !important;
  box-shadow: 0 3px 15px 0 rgba(0,0,0,0.25) !important;
  box-sizing: border-box !important;
  padding: 0 20px !important;
  transition: all .4s,bottom .8s ease-in-out !important;
  cursor: pointer !important;
  outline: none !important;
  display: inline-block !important;
```

```

margin: 0 !important;
-webkit-font-smoothing: antialiased !important;
-webkit-tap-highlight-color: rgba(0,0,0,0) !important;
animation-name: loadBubble;
animation-iteration-count: 1;
animation-timing-function: ease-in-out;
animation-duration: .2s;
}

```

Solution

Avoid removing the default focus outline for buttons. If the default browser focus style isn't used, provide a clearly visible focus style for interactive elements.

Recommended code

```

.olark-launch-button {
    font-family: -apple-system,BlinkMacSystemFont,"Segoe
UI","Roboto","Oxygen","Ubuntu","Cantarell","Fira Sans","Droid Sans","Helvetica
Neue",sans-serif !important;
    font-size: 100% !important;
    position: fixed !important;
    bottom: 15px !important;
    right: 20px !important;
    left: auto !important;
    height: 48px !important;
    width: auto !important;
    z-index: 2147483646 !important;
    background: #2d2d2d !important;
    border: 0 !important;
    border-radius: 100px !important;
    box-shadow: 0 3px 15px 0 rgba(0,0,0,0.25) !important;
    box-sizing: border-box !important;
    padding: 0 20px !important;
    transition: all .4s,bottom .8s ease-in-out !important;
    cursor: pointer !important;
    display: inline-block !important;
    margin: 0 !important;
    -webkit-font-smoothing: antialiased !important;
    -webkit-tap-highlight-color: rgba(0,0,0,0) !important;
    animation-name: loadBubble;
    animation-iteration-count: 1;
    animation-timing-function: ease-in-out;
    animation-duration: .2s;
}

```

Screenshot(s)

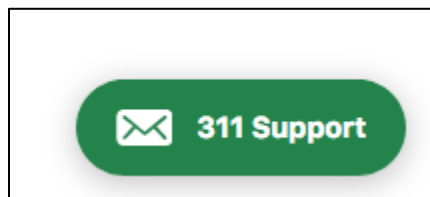


Figure 102a: - Screenshot of button with poor visible focus

Live chat modal does not trap keyboard focus (CLT-103)

Components	001 - charlottenc.gov - Header/Footer
Affected community	Vision, Cognitive, Motor
WCAG Success Criterion	2.1.2 No Keyboard Trap (A)
WCAG Level	A
Priority	High

Description

The modal overlay for the live chat does not trap keyboard focus. The user is able to tab through other parts of the page.

This will create problems for screen-reader, keyboard or other assistive-technology users who launch the modal but may not be able to operate it or close out of it. If these users can tab out of the modal they are easily confused about their location within the page.

Solution

1. Ensure that keyboard focus is set by default to the first heading element in the modal
2. The focus should be kept inside the modal until the visitor closes it
3. When the modal is closed, the focus should be returned to the button element that triggered the modal
4. Provide an escape route to get out of the modal. There should be options for keyboard-only and mouse input.

Refer to the W3C modal example for a thorough code sample and explanation on how to build accessible modals: <https://www.w3.org/TR/wai-aria-practices/examples/dialog-modal/dialog.html>

Linked issues

- CLT-105

Screenshot(s)

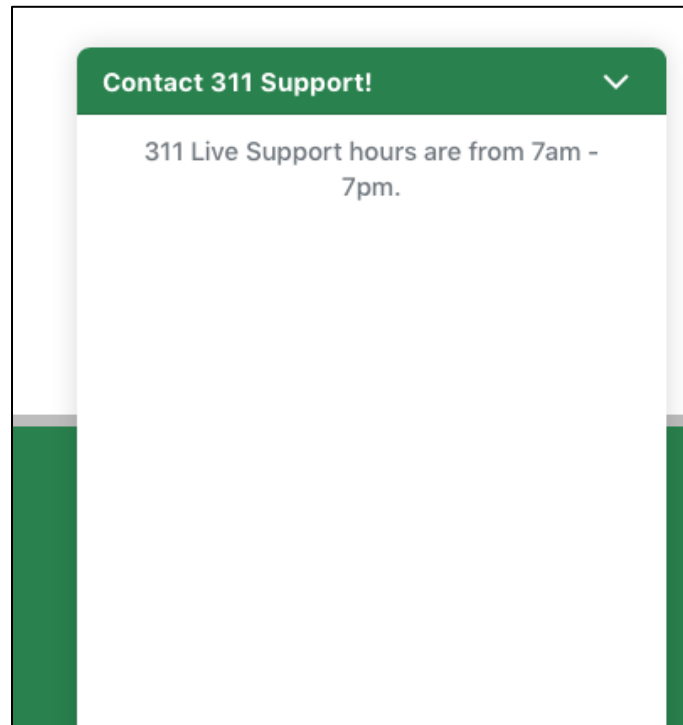


Figure 103a: - Screenshot of chat modal

Favorite link takes unregistered users away from page (CLT-104)

Components	009 - charlottenc.gov - HR - Jobs - See Available Jobs
Affected community	Vision, Cognitive
WCAG Success Criterion	3.2.2 On Input (A)
WCAG Level	A
Priority	Medium

Description

There are star icons that allow users to "favorite" job listings. If a user is not registered, they are automatically taking to the sign in page without any warning or understanding as to why they are at the new page. This could be a confusing experience for both screen reader users or visitors with cognitive impairments.

Solution

Disable the link if a user is not registered, or provide proper context before a user activates the link to manage user's expectations.

Screenshot(s)




Figure 104a: - Screenshot of links

Live chat dialog is not announced by screen readers (CLT-105)


Components	001 - charlottenc.gov - Header/Footer
Affected community	Vision
WCAG Success Criterion	3.2.2 On Input (A), 4.1.2 Name, Role, Value (A)
WCAG Level	A
Priority	High

Description

When the live chat dialog is triggered, it is not announced to screen readers. This appears to be related to the fact that the dialog doesn't have the correct:

- ARIA roles
- Keyboard trap ( [CLT-103](#) Open)
- Label

Similarly, when the live chat is ended (by the agent or the visitor), nothing is announced to screen readers.

This may cause confusion with screen-reader users, who may not know that something has changed on the site. This confusion is compounded by the fact that the dialog is not keyboard accessible ( [CLT-106](#) Open), which impacts screen-reader users who navigate with the keyboard

Solution

Ensure the live chat has the necessary properties and attributes to designate it as a dialog. Refer to the W3C modal example for a thorough code sample and explanation on how to build accessible modals: <https://www.w3.org/TR/wai-aria-practices/examples/dialog-modal/dialog.html>

Linked issues

- CLT-103
- CLT-106

Screenshot(s)

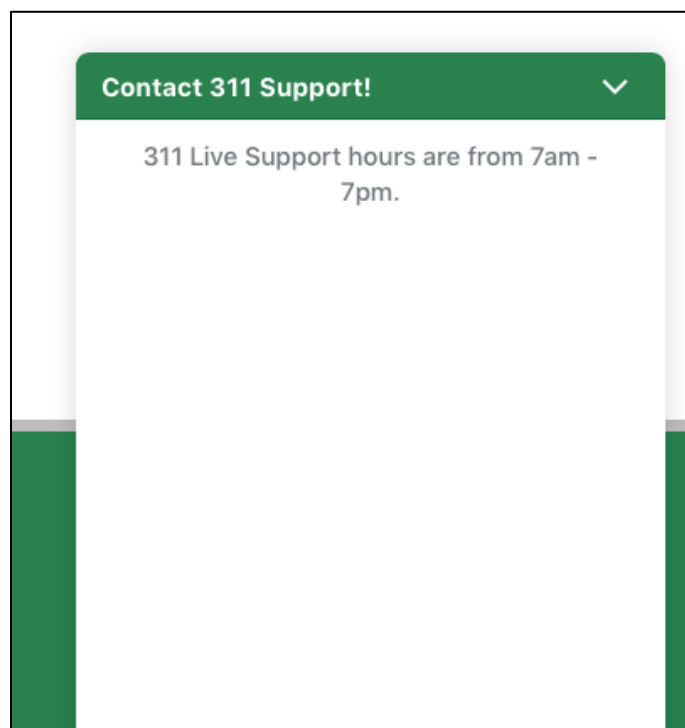



Figure 105a: - Screenshot of live chat

Live chat modal is not keyboard accessible (CLT-106)

Components	001 - charlottenc.gov - Header/Footer
Affected community	Vision, Motor
WCAG Success Criterion	3.2.2 On Input (A)
WCAG Level	A
Priority	High

Description

The live chat behaves like a dialog, but the keyboard accessibility doesn't follow expected patterns:

- Ensure that keyboard focus is set by default to the first heading element in the modal
- The focus should be kept inside the modal until the visitor closes it ( [CLT-103](#) Open)
- When the modal is closed, the focus should be returned to the button element that triggered the modal
- Provide an escape route to get out of the modal. There should be options for keyboard-only and mouse input.

This can cause issues for keyboard users, sighted or not, who are unable to navigate as expected with the dialog.

Unfortunately, these keyboard navigation problems continue once a chat starts:

- Focus is not set on the field to start a conversation
- Users have to Shift+Tab to discover the file upload button
- Users have to Shift+Tab *several* times to discover the "End Chat" option

These are major problems for screen-reader users who may not even be aware there is an option to upload a file or to end the chat.

Solution

Refer to the W3C modal example for a thorough code sample and explanation on how to build accessible modals: <https://www.w3.org/TR/wai-aria-practices/examples/dialog-modal/dialog.html>

Linked issues

- CLT-105

Screenshot(s)

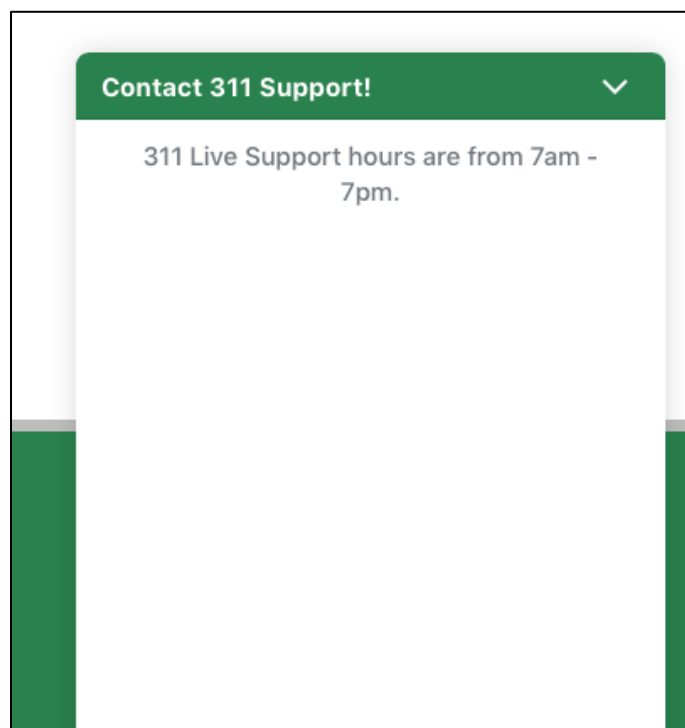


Figure 106a: - Screenshot of live chat

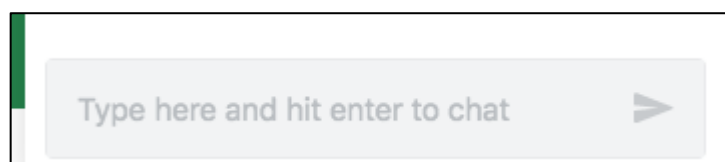


Figure 106b: - Screenshot of live chat

Keyboard focus is not set to required field (CLT-115)

Components	007 - charlottenc.gov - TRASH - Schedule a Bulky Item Pickup
Affected community	Vision, Motor
WCAG Success Criterion	3.3.1 Error Identification (A)
WCAG Level	A
Priority	Medium

Description

If errors are found in the form, at the end of the workflow, a list of errors are provided. These errors are clickable, but don't link to the specific erroneous field. They link to the page where the erroneous input is located. This makes it difficult for keyboard-only users, sighted or not, to find the field that needs correcting.

Solution

Set keyboard focus to the specific erroneous field. For a reference, visit <https://www.w3.org/TR/2016/NOTE-WCAG20-TECHS-20161007/G139>

Screenshot(s)



Figure 115a: - Screenshot of link to required field

Low contrast on selected option text (CLT-118)

Components	007 - charlottenc.gov - TRASH - Schedule a Bulky Item Pickup
Affected community	Vision
WCAG Success Criterion	1.4.3 Contrast (Minimum) (AA)
WCAG Level	AA
Priority	High

Description

There is insufficient contrast between the selected option text and background highlight color. The minimum required contrast ratio is 4.5:1.

- Item name:
The foreground color is #FF0000 and the background color is #337ab7. The contrast ratio is 1.13:1.
- "Not Collected" text:
The foreground color is #777 and the background color is #337ab7. The contrast ratio is 1.01:1.

Solution

Ensure the color contrast between text foreground and background colors is at least 4.5:1.

Screenshot(s)

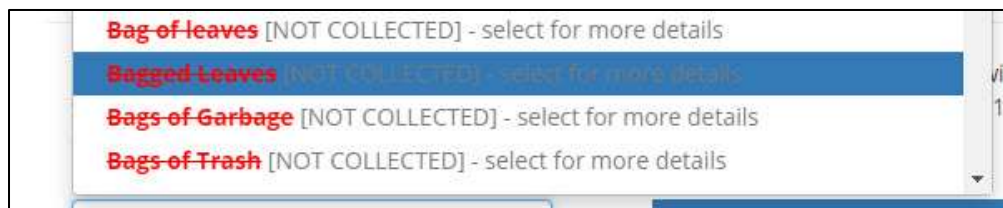


Figure 118a: - Screenshot of text with poor color contrast

Items are not fully announced by screen readers (CLT-119)

Components	007 - charlottenc.gov - TRASH - Schedule a Bulky Item Pickup
Affected community	Vision
WCAG Success Criterion	1.3.1 Info and Relationships (A), 4.1.2 Name, Role, Value (A)
WCAG Level	A
Priority	High

Description

The items in the autocomplete list are not fully announced by the screen reader such as the category and details. Screen reader users will be confused and might not know about important information.

Current code

```
<li aria-label="Automobile seats" aria-selected="true" id="rbt-menu-item-5"
role="option" class="active"><a class="dropdown-item active" href="#"><div><strong
style="color: red; text-decoration: line-through;"> Automobile seats</strong><span
class="text-muted"> [NOT COLLECTED] - select for more details</span></div></a></li>
```

Solution

Add all text for each item to the `aria-label` attribute.

Recommended code

```
<li aria-label="Automobile seats [NOT COLLECTED] - select for more details" aria-
selected="true" id="rbt-menu-item-5" role="option" class="active"><a
class="dropdown-item active" href="#"><div><strong style="color: red; text-
decoration: line-through;"> Automobile seats</strong><span class="text-muted"> [NOT
COLLECTED] - select for more details</span></div></a></li>
```

Screenshot(s)

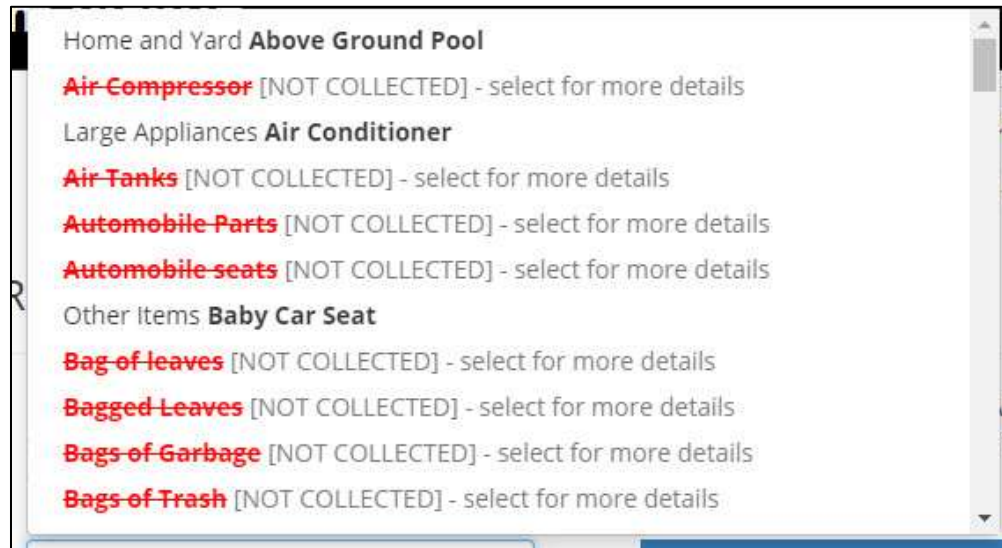


Figure 119a: - Screenshot of items not fully announced by screen readers

Low contrast on text (CLT-120)

Components	007 - charlottenc.gov - TRASH - Schedule a Bulky Item Pickup
Affected community	Vision
WCAG Success Criterion	1.4.3 Contrast (Minimum) (AA)
WCAG Level	AA
Priority	Medium

Description

There is insufficient contrast below the minimum required contrast ratio of 4.5:1. The foreground color is #777 and the background color is #FFFFFF. The contrast ratio is 4.47:1.

Solution

Ensure the color contrast between text foreground and background colors is at least 4.5:1.

Screenshot(s)

A small rectangular box containing the text "(max 15)". The text is dark gray and the background is white, illustrating low contrast.

Figure 120a: - Screenshot of text with low contrast

No form instructions provided (CLT-121)

Components	001 - charlottenc.gov - Header/Footer
Affected community	Cognitive
WCAG Success Criterion	1.3.1 Info and Relationships (A), 4.1.2 Name, Role, Value (A)
WCAG Level	A
Priority	High


Description

All three fields on the initial live chat interface are indicated as required, with asterisk icons. However, there are no instructions to indicate that all fields are required, or even that fields with an asterisk are required.

This makes it difficult for all visitors, regardless of their ability, to know which fields need to be completed in order to start chatting.

Solution

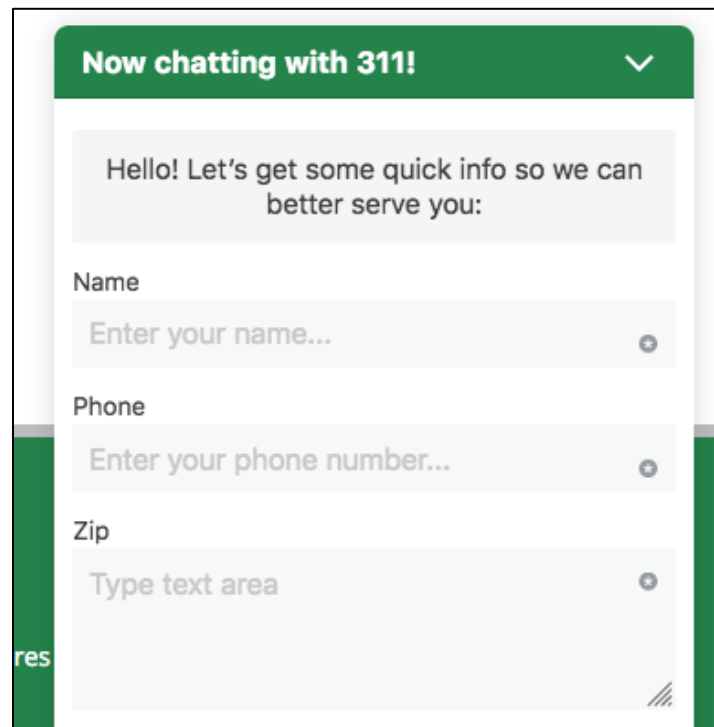
Add instructions before the form fields to indicate that all fields are required.

Make sure to indicate this programmatically by using the HTML `required` attribute on each required input ( [CLT-126](#) Open).

Linked issues

- CLT-126

Screenshot(s)



The screenshot shows a live chat interface. At the top, a green header bar contains the text "Now chatting with 311!" and a downward-pointing chevron icon. Below the header, a light gray message bubble contains the text "Hello! Let's get some quick info so we can better serve you:". Underneath the message, there are three input fields, each with a label and a placeholder text, and a small circular icon with a plus sign to the right of each field. The first field is labeled "Name" with the placeholder "Enter your name...". The second field is labeled "Phone" with the placeholder "Enter your phone number...". The third field is labeled "Zip" with the placeholder "Type text area". The entire form is set against a white background with a green vertical bar on the left side. The word "res" is partially visible on the left edge of the white area.

Now chatting with 311! ✓

Hello! Let's get some quick info so we can better serve you:

Name
Enter your name... +

Phone
Enter your phone number... +

Zip
Type text area +

res

Figure 121a: - Screenshot of live chat form

Error messages not programmatically associated with input fields (CLT-122)

Components	001 - charlottenc.gov - Header/Footer
Affected community	Vision
WCAG Success Criterion	3.3.2 Labels or Instructions (A), 4.1.2 Name, Role, Value (A)
WCAG Level	A
Priority	High

Description

If an error is made in the initial live chat form, error messages are displayed but the messages are not programmatically associated to the `input` fields.

While sighted users can easily see the error messages, screen-reader users may have difficulty identifying the what needs to change.

Current code

```
<div class="olark-survey-form-error-msg" data-reactid=".0.5.4.0.0.0.0"
role="alert">Please enter a valid phone number</div>

<input class="olark-survey-form-input olark-survey-form-text" data-lpignore="true"
aria-label="Phone" tabindex="0" type="text" placeholder="Enter your phone number..."
data-reactid=".0.5.4.0.0.1.0.0:$1.2">
```

Solution

Add a unique ID to the `<div>` containing the instructions. Add the `aria-describedby` attribute to the `<input>`, with the value reflecting the ID of the instructions `<div>`.

Recommended code

```
<div id="error-phone" class="olark-survey-form-error-msg" data-
reactid=".0.5.4.0.0.0.0" role="alert">Please enter a valid phone number</div>

<input aria-describedby="error-phone" class="olark-survey-form-input olark-survey-
form-text" data-lpignore="true" aria-label="Phone" tabindex="0" type="text"
placeholder="Enter your phone number..." data-reactid=".0.5.4.0.0.1.0.0:$1.2">
```

Screenshot(s)

The screenshot shows a mobile application interface for starting a chat. At the top, a green header bar contains the text "Now chatting with 311!" and a downward-pointing chevron icon. Below the header, a light red rectangular box displays the error message "Please complete all required fields" in red text. Underneath this message, there are three input fields, each with a red border indicating they are required and currently empty. The first field is labeled "Name" and contains the placeholder text "Enter your name...". The second field is labeled "Phone" and contains the placeholder text "Enter your phone number...". The third field is labeled "Zip" and contains the placeholder text "Type text area". To the right of each input field is a small red circle with a white plus sign. At the bottom of the form is a large green button with the text "Start chatting" in white. The entire form is set against a light gray background.

Figure 122a: - Screenshot of error message

Link is missing href attribute and not focusable (CLT-123)

Components	007 - charlottenc.gov - TRASH - Schedule a Bulky Item Pickup
Affected community	Vision, Motor
WCAG Success Criterion	2.1.1 Keyboard (A)
WCAG Level	A
Priority	High

Description

The link to remove items from the "Selected Bulk Items" list is missing the `href` attribute and cannot receive keyboard focus. The link functions as a button since there is no destination and it does an action.

Keyboard-only users, sighted or not, will have great difficulty or might not be able to reach the link.

Current code

```
<div class="bulky-button">
<a class="btn btn-xs btn-danger">
  <i class="fa fa-2x fa-times"></i>
</a>
</div>
```

Solution

Change the `<a>` element to `button` element. The `button` element will receive keyboard focus by default. Use a foreground image with alt text rather than a font icon.

If you need to keep it as a link, then add the `href` attribute to receive keyboard focus.

Recommended code

```
<div class="bulky-button">
<button class="btn btn-xs btn-danger">
  
</button>
</div>
```

Linked issues

- CLT-125
- CLT-129

Screenshot(s)



Figure 123a: - Screenshot of link without href



Figure 123b: - Screenshot of links without href

Incomplete error messages (CLT-124)

Components	001 - charlottenc.gov - Header/Footer
Affected community	Vision, Cognitive
WCAG Success Criterion	3.3.2 Labels or Instructions (A), 4.1.2 Name, Role, Value (A)
WCAG Level	A
Priority	Medium

Description

The error messages in the initial live chat form are incomplete, so that visitors have to submit multiple times in order to be notified that fields are required. For example, a visitor can submit the form with the required phone and zip code fields blank, but only an error message about the phone number is available. The user has to submit again, with the zip code field blank, in order to be notified that field is also required.

This can create frustration and confusion for all users, regardless of ability or device, who have to submit the form multiple times in order to understand which fields are necessary.

Solution

Provide all error messages when a visitor submits the live chat, so that visitors have the full context and instructions necessary to complete the form.

Linked issues

- CLT-126

Screenshot(s)

The screenshot shows a mobile interface for a live chat widget. At the top, a green header bar contains the text "Now chatting with 311!" and a downward-pointing chevron icon. Below the header, a light red rectangular box displays the error message "Please enter a valid phone number" in red text. The form consists of several fields: a "Name" field with the placeholder "Test First Name" and a small circular icon on the right; a "Phone" field with the placeholder "Enter your phone number..." and a red border, which is the source of the error message; a "Zip" field with the placeholder "Type text area" and a small circular icon on the right. A green "Start chatting" button is located at the bottom of the form. On the left side of the form, there is a vertical green bar with the text "es" and a small "Live Chat Widget" label.

Figure 124a: - Screenshot of error message

No text alternative on font icon on link (CLT-125)

Components	007 - charlottenc.gov - TRASH - Schedule a Bulky Item Pickup
Affected community	Vision
WCAG Success Criterion	1.1.1 Non-text content (A), 4.1.2 Name, Role, Value (A)
WCAG Level	A
Priority	Medium

Description

The close icon by font awesome does not have a text alternative. Screen readers don't announce the icon. Screen reader users will be confused or not know that the icon exists.

Current code

```
<i class="fa fa-2x fa-times"></i>
```

Solution

Use a `<button>` with with a foreground image and `alt` text

Recommended code

```

```

Linked issues

- CLT-123

Screenshot(s)

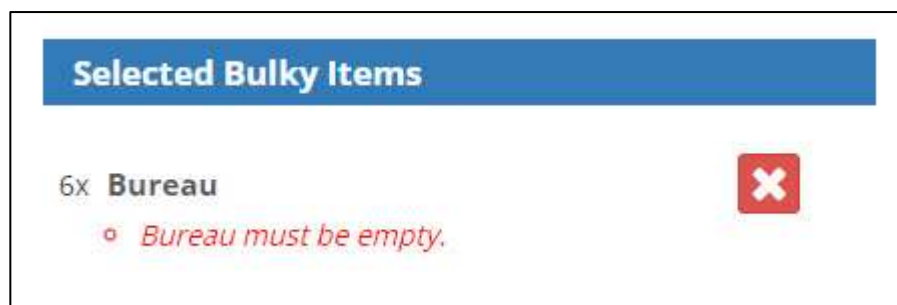


Figure 125a: - Screenshot of font icon

Required form fields are not programmatically required (CLT-126)

Components	001 - charlottenc.gov - Header/Footer
Affected community	Vision, Cognitive
WCAG Success Criterion	3.3.1 Error Identification (A), 4.1.2 Name, Role, Value (A)
WCAG Level	A
Priority	Medium

Description

On the initial live chat form fields *appear* required, per the SVG asterisks, and are validated with error messages. However, these fields are not correctly announced as required by screen readers.

In VoiceOver, when reading the chat content, all three of the SVG asterisks and instructions are announced consecutively — without the context of the fields themselves. Similarly, when tabbing through the fields, only the first field announces the asterisk and instructions.


This can cause confusion for screen-reader users, who aren't provided the necessary context of which fields are required. This confusion is compounded by the incomplete error messaging noted in [👤 CLT-124 Open](#).

Current code

```
<div class="olark-survey-form-item olark-survey-form-item-name olark-survey-form-is-required" data-reactid=".0.5.4.0.0.1.0.0:$0">
  <span class="olark-survey-form-label" data-reactid=".0.5.4.0.0.1.0.0:$0.0">Name</span>
  <span class="olark-survey-form-req-icon" title="Required field" data-reactid=".0.5.4.0.0.1.0.0:$0.1">
    <svg viewBox="0 0 20 20" x="0" y="0" fill="currentcolor" aria-label="required asterisk signifies that this field can not be left blank" role="img" data-reactid=".0.5.4.0.0.1.0.0:$0.1.0"><title data-reactid=".0.5.4.0.0.1.0.0:$0.1.0.0">Required Asterisk</title><desc data-reactid=".0.5.4.0.0.1.0.0:$0.1.0.1">signifies that this field can not be left blank</desc><path d="M10,0C4.5,0,0,4.5,0,10c0,5.5,4.5,10,10,10s10-4.5,10-10C20,4.5,15.5,0,10,0z M13.8,13.6L12,15l-2-3L8,15l-1.8-1.4l2-2.8 L5,9.6l0.8-2.2L9,8.6L8.8,5h2.4L11,8.6l3.2-1.3L15,9.6l-3.2,1.1L13.8,13.6z" data-reactid=".0.5.4.0.0.1.0.0:$0.1.0.2"></path></svg>
  </span>
  <input class="olark-survey-form-input olark-survey-form-text" data-lpignore="true" aria-label="Name" tabindex="0" type="text" placeholder="Enter your name..." data-reactid=".0.5.4.0.0.1.0.0:$0.2">
</div>
```

Solution

Adding the boolean `required` attribute to the required `input` fields will provide native, client-side validation that will ensure visitors don't submit the form without entering a value first or without entering in the correct format. This attribute will also provide screen-reader users necessary context to know the field is required. For keyboard users, sighted or not, the attribute will ensure focus returns to the field after submitted.

Also consider changing how the required instructions are provided. Instead of relying on an SVG that isn't programmatically associated with the `input` fields, add `<label>` elements ( [CLT-128](#) Open).

These `label` elements should include the field purpose, as well as any instructions. A common pattern includes:

- An asterisk, which is hidden from screen-reader users with the `aria-hidden="true"` attribute
- Instructions that the field is required, which may or may not be visibly-hidden with CSS rules that still expose the content to screen readers.

Recommended code

```
<div class="olark-survey-form-item olark-survey-form-item-name olark-survey-form-is-required" data-reactid=".0.5.4.0.0.1.0.0:$0">
  <label for="input-name" class="olark-survey-form-label" data-reactid=".0.5.4.0.0.1.0.0:$0.0">Name <span aria-hidden="true">*</span><span class="sr-only">This field is required</span></label>
  <input id="input-name" class="olark-survey-form-input olark-survey-form-text" data-lpignore="true" tabindex="0" type="text" placeholder="Enter your name..." data-reactid=".0.5.4.0.0.1.0.0:$0.2" required>
</div>
```

Linked issues

- CLT-121
- CLT-124

Screenshot(s)

The screenshot shows a chat window with a green header bar that reads "Now chatting with 311!" and a downward arrow icon. Below the header, a light gray message bubble contains the text "Hello! Let's get some quick info so we can better serve you:". Following the message are three form fields, each with a label and a text input area containing a placeholder and a required field asterisk (*):

- Name**: The input field contains the placeholder "Enter your name..."
- Phone**: The input field contains the placeholder "Enter your phone number..."
- Zip**: The input field contains the placeholder "Type text area"

At the bottom left of the chat window, the word "res" is partially visible. At the bottom right, there is a small icon consisting of three diagonal lines.

Figure 126a: - Screenshot of required form fields

Low contrast on inactive date text (CLT-127)

Components	007 - charlottenc.gov - TRASH - Schedule a Bulky Item Pickup
Affected community	Vision
WCAG Success Criterion	1.4.3 Contrast (Minimum) (AA)
WCAG Level	AA
Priority	High

Description

There is insufficient contrast for the inactive dates text that is below the minimum required ratio of 4.5:1. The foreground color is #E49E7D and the background color is #C46715. The contrast ratio is 1.79:1.

Solution

Ensure the color contrast between foreground and background colors are at least 4.5:1

Screenshot(s)

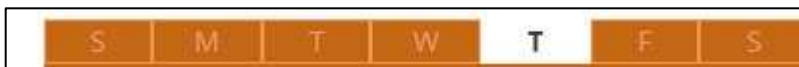


Figure 127a: - Screenshot of text with low contrast

No labels for input fields (CLT-128)

Components	001 - charlottenc.gov - Header/Footer
Affected community	Cognitive, Motor
WCAG Success Criterion	1.3.1 Info and Relationships (A), 3.3.2 Labels or Instructions (A)
WCAG Level	A
Priority	Medium

Description

In the initial live chat form, there are no `label` elements for the input fields. Instead, the fields rely on the `aria-label` attribute.

Unfortunately, this attribute only supports screen-reader users, whereas native `label` elements support all visitors — including visitors with motor impairments.

Current code

```
<div class="olark-survey-form-item olark-survey-form-item-name olark-survey-form-is-required" data-reactid=".0.5.4.0.0.1.0.0:$0">
  <span class="olark-survey-form-label" data-reactid=".0.5.4.0.0.1.0.0:$0.0">Name</span>
  <span class="olark-survey-form-req-icon" title="Required field" data-reactid=".0.5.4.0.0.1.0.0:$0.1">
    <svg viewBox="0 0 20 20" x="0" y="0" fill="currentcolor" aria-label="required asterisk signifies that this field can not be left blank" role="img" data-reactid=".0.5.4.0.0.1.0.0:$0.1.0"><title data-reactid=".0.5.4.0.0.1.0.0:$0.1.0.0">Required Asterisk</title><desc data-reactid=".0.5.4.0.0.1.0.0:$0.1.0.1">signifies that this field can not be left blank</desc><path d="M10,0C4.5,0,0,4.5,0,10c0,5.5,4.5,10,10,10s10-4.5,10-10C20,4.5,15.5,0,10,0z M13.8,13.6L12,15l-2-3L8,15l-1.8-1.4l2-2.8 L5,9.6l0.8-2.2L9,8.6L8.8,5h2.4L11,8.6l3.2-1.3L15,9.6l-3.2,1.1L13.8,13.6z" data-reactid=".0.5.4.0.0.1.0.0:$0.1.0.2"></path></svg>
  </span>
  <input class="olark-survey-form-input olark-survey-form-text" data-lpignore="true" aria-label="Name" tabindex="0" type="text" placeholder="Enter your name..." data-reactid=".0.5.4.0.0.1.0.0:$0.2">
</div>
```

Solution

Add a native `<label>` element for each form field:

- Add an `id` to the `<input>` or `<textarea>`
- Add a `for` attribute to the label, the value of which is the field ID

This will provide:

- A visible label for sighted users
- An announced label for screen-reader users
- A larger clickable area for visitors with motor disabilities

Also, with `label` elements in place, the `aria-label` attributes are no longer necessary and can be removed.

Recommended code

```
<div class="olark-survey-form-item olark-survey-form-item-name olark-survey-form-is-required" data-reactid=".0.5.4.0.0.1.0.0:$0">
  <label for="input-name" class="olark-survey-form-label" data-reactid=".0.5.4.0.0.1.0.0:$0.0">Name <span aria-hidden="true">*</span><span class="sr-only">This field is required</span></label>
  <input id="input-name" class="olark-survey-form-input olark-survey-form-text" data-lpignore="true" tabindex="0" type="text" placeholder="Enter your name..." data-reactid=".0.5.4.0.0.1.0.0:$0.2">
</div>
```

Screenshot(s)

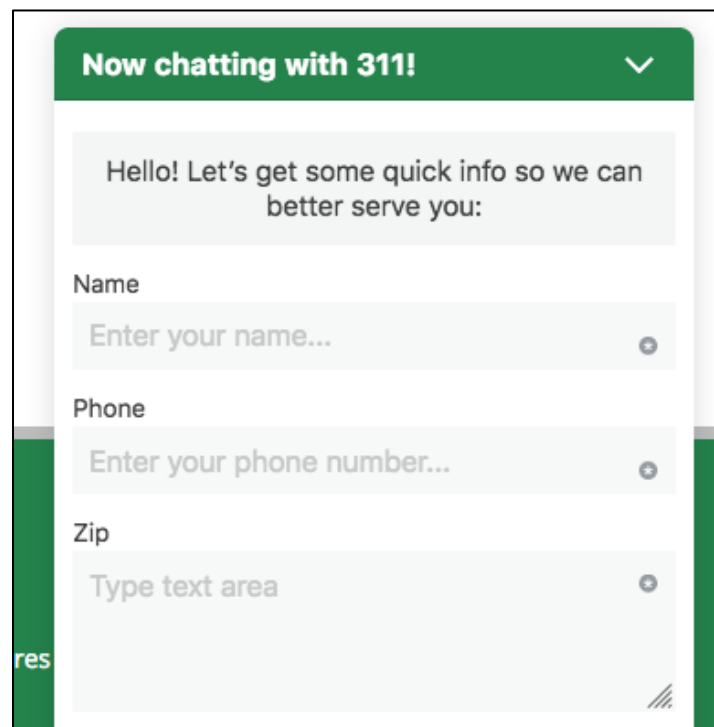
A screenshot of a chat interface. At the top, a green header bar contains the text "Now chatting with 311!" and a downward arrow icon. Below the header, a light gray message bubble contains the text "Hello! Let's get some quick info so we can better serve you:". Underneath the message bubble, there are three input fields. The first is labeled "Name" and has a placeholder "Enter your name..." with a small asterisk icon to its right. The second is labeled "Phone" and has a placeholder "Enter your phone number..." with a small asterisk icon to its right. The third is labeled "Zip" and has a placeholder "Type text area" with a small asterisk icon to its right. The input fields are light gray with rounded corners. The entire chat interface is set against a dark green background.

Figure 128a: - Screenshot of input fields without labels

Link used instead of button (CLT-129)

Components	007 - charlottenc.gov - TRASH - Schedule a Bulky Item Pickup
Affected community	Vision, Motor
WCAG Success Criterion	2.1.1 Keyboard (A)
WCAG Level	A
Priority	Medium

Description

The link is missing the `href` attribute and cannot receive keyboard focus. The link is made to function as a button since there is no destination and it does an action.

Keyboard-only users, sighted or not, will have great difficulty or might not be able to reach the link.

Current code

```
<div class="step-container clickable"><div class="step-box ">Contact</div><a  
type="button" class="btn circle btn-primary">3</a></div>
```

Solution

Change the `<a>` element to `button` element. The `button` element will receive keyboard focus by default.

Buttons do something. Link go somewhere.

Recommended code

```
<div class="step-container clickable"><div class="step-box ">Contact</div>  
<button class="btn circle btn-primary">3</button>  
</div>
```

Linked issues

- CLT-123

Screenshot(s)



Figure 129a: - Screenshot of elements marked up as links instead of buttons

Low contrast on focus style (CLT-130)

Components	007 - charlottenc.gov - TRASH - Schedule a Bulky Item Pickup
Affected community	Vision, Motor
WCAG Success Criterion	1.4.11 Non-Text Contrast (AA) (WCAG 2.1)
WCAG Level	AA
Priority	Medium

Description

The buttons for the form have low contrast for the keyboard focus style. The foreground color is #286090 and the background border color is #122B40. The contrast ratio is 2.19:1. The minimum contrast should be 4.5:1

Current code

```
<button type="submit" class="btn btn-primary wiz-btn">Next <i class="fa fa-arrow-circle-right"></i></button>
```

Solution

Ensure the color contrast between text foreground and background colors is at least 4.5:1.

Screenshot(s)

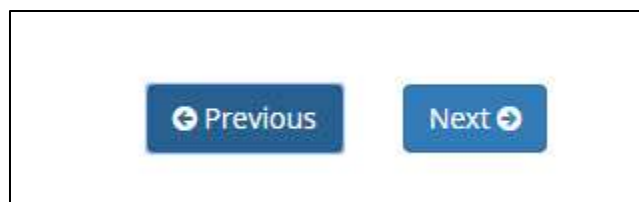


Figure 130a: - Screenshot of focus styles with low contrast

Live chat interactions not announced to screen readers (CLT-131)

Components	001 - charlottenc.gov - Header/Footer
Affected community	Vision
WCAG Success Criterion	3.2.2 On Input (A)
WCAG Level	A
Priority	High

Description

When engaged in the live chat, the interactions are not announced by screen readers. For example, customer support agent responses aren't announced.

While these interactions are obvious *visually*, screen-reader users aren't provided any information about what is happening on the screen. As such, these users can't interact effectively with the live chat.

Current code

```
<div class="olark-message-container" data-  
reactid=".0.5.1.0.1:$269=159294899790433.1">  
  <p class="olark-operator-name" aria-hidden="true" data-  
reactid=".0.5.1.0.1:$269=159294899790433.1.0">Serina</p>  
  <div class="olark-message-group" role="group" aria-label="messages from  
Serina" data-reactid=".0.5.1.0.1:$269=159294899790433.1.1">  
    <p class="olark-operator-message" data-  
reactid=".0.5.1.0.1:$269=159294899790433.1.1.0:$0">Ok</p>  
  </div>  
</div>
```

Solution

Add an `aria-live="polite"` attribute to the block of content that changes dynamically. This will notify screen readers that they need to announce a change in content.

Recommended code

```
<div class="olark-message-container" data-  
reactid=".0.5.1.0.1:$269=159294899790433.1">  
  <p class="olark-operator-name" aria-hidden="true" data-  
reactid=".0.5.1.0.1:$269=159294899790433.1.0">Serina</p>  
  <div aria-live="polite" class="olark-message-group" role="group" aria-  
label="messages from Serina" data-reactid=".0.5.1.0.1:$269=159294899790433.1.1">
```

```
<p class="olark-operator-message" data-  
reactid=".0.5.1.0.1:$269=159294899790433.1.1.0:$0">Ok</p>  
</div>  
</div>
```

Screenshot(s)

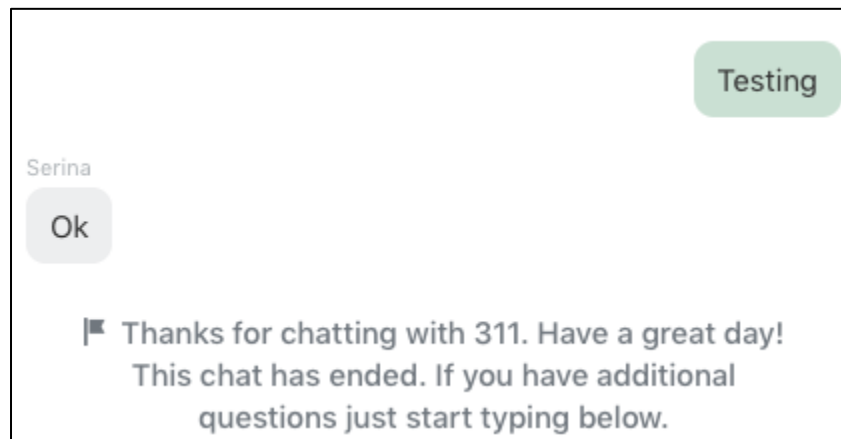


Figure 131a: - Screenshot of live chat interaction

Removing item is not announced by the screen reader (CLT-132)

Components	007 - charlottenc.gov - TRASH - Schedule a Bulky Item Pickup
Affected community	Vision
WCAG Success Criterion	3.2.2 On Input (A)
WCAG Level	A
Priority	Medium

Description

When an item is removed from the "Selected Bulk Item" list, screen reader does not announce the change to the user. Screen reader users will be confused and not know if anything happened.

Solution

Apply the attribute `aria-live="polite"` to the "bulky item" block of content. This will inform the screen reader when there is a change - they can then announce to the user what the change was.

Screenshot(s)



Figure 132a: - Screenshot of interaction not announced by screen reader

SVG decorative images announced by screen reader (CLT-133)

Components	001 - charlottenc.gov - Header/Footer
Affected community	Vision
WCAG Success Criterion	1.1.1 Non-text content (A)
WCAG Level	A
Priority	High

Description

There are several SVG images that are decorative, but are announced to screen readers. For example, the "end chat" option has an SVG that is announced as "Crossed out chat bubble a chat bubble with a slash through it to indicate ending the chat".

This can create a cumbersome experience for screen-reader users, who don't need information about decorative images and can rely on the surrounding text for context (e.g. "End chat" that appears in the <button>).

Current code

```
<svg viewBox="531 3471 15 16" class="olark-end-chat-endicon" aria-label="a chat bubble with a slash through it to indicate ending the chat" role="img" data-reactid=".0.5.0.0.0"><title data-reactid=".0.5.0.0.0">Crossed out chat bubble</title><desc data-reactid=".0.5.0.0.0.1">a chat bubble with a slash through it to indicate ending the chat</desc><g stroke="none" fill="none" stroke-width="1" transform="translate(531.000000, 3472.000000)" data-reactid=".0.5.0.0.0.2"><path d="M5.8266005,11.2595836 L11.1836089,2.55444494 C13.0574735,3.22324548 14.3800899,4.84688391 14.3800899,6.74354628 C14.3800899,8.39504344 13.377672,9.83941205 11.8817967,10.6269148 C11.9889945,12.6897487 13.6068524,13.191371 13.6068524,13.191371 C10.8363324,13.830996 8.94945868,12.0486558 8.27082176,11.2595836 L5.8266005,11.2595836 Z M2.647,10.7023078 C1.06913675,9.93398966 8.98097187e-05,8.44924119 8.98097187e-05,6.74354628 C8.98097187e-05,4.24975985 2.28607423,2.22797037 5.10625491,2.22797037 L7.86197688,2.22797037 L2.647,10.7023078 Z" fill="#ffffff" transform="translate(7.190090, 7.777628) scale(-1, 1) translate(-7.190090, -7.777628)" data-reactid=".0.5.0.0.0.2.0"></path><path d="M3.6,0.479980469 L11.6,13.4799805" stroke="#ffffff" stroke-linecap="square" data-reactid=".0.5.0.0.0.2.1"></path></g></svg>
```

Solution

Implement the `aria-hidden` attribute on the SVG images that are purely decorative. Further, since these are decorative images, remove the `aria-label` attribute.

Recommended code

```
<svg aria-hidden="true" viewBox="531 3471 15 16" class="olark-end-chat-endicon"
role="img" data-reactid=".0.5.0.0.0"><title data-reactid=".0.5.0.0.0.0">Crossed out
chat bubble</title><desc data-reactid=".0.5.0.0.0.1">a chat bubble with a slash
through it to indicate ending the chat</desc><g stroke="none" fill="none" stroke-
width="1" transform="translate(531.000000, 3472.000000)" data-
reactid=".0.5.0.0.0.2"><path d="M5.8266005,11.2595836 L11.1836089,2.55444494
C13.0574735,3.22324548 14.3800899,4.84688391 14.3800899,6.74354628
C14.3800899,8.39504344 13.377672,9.83941205 11.8817967,10.6269148
C11.9889945,12.6897487 13.6068524,13.191371 13.6068524,13.191371
C10.8363324,13.830996 8.94945868,12.0486558 8.27082176,11.2595836
L5.8266005,11.2595836 Z M2.647,10.7023078 C1.06913675,9.93398966 8.98097187e-
05,8.44924119 8.98097187e-05,6.74354628 C8.98097187e-05,4.24975985
2.28607423,2.22797037 5.10625491,2.22797037 L7.86197688,2.22797037 L2.647,10.7023078
Z" fill="ffffff" transform="translate(7.190090, 7.777628) scale(-1, 1) translate(-
7.190090, -7.777628)" data-reactid=".0.5.0.0.0.2.0"></path><path d="M3.6,0.479980469
L11.6,13.4799805" stroke="ffffff" stroke-linecap="square" data-
reactid=".0.5.0.0.0.2.1"></path></g></svg>
```

Screenshot(s)



Figure 133a: - Screenshot of SVG decorative image

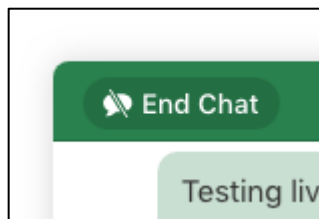


Figure 133b: - Screenshot of SVG decorative image

Live chat session ends without warning (CLT-134)

Components	001 - charlottenc.gov - Header/Footer
Affected community	Vision, Cognitive
WCAG Success Criterion	2.2.1 Timing Adjustable (A)
WCAG Level	A
Priority	Medium

Description

If an active live chat conversation remains idle for a period of time, the chat session ends without warning.

This can cause confusion for *all* visitors, regardless of ability or device, who need more time to engage.

Solution

Don't set time limits on interactions. If that isn't possible:

- Advise the visitor about the time limit in advance
- Give the the visitor the option to turn off or adjust the time limit
- Warn the visitor before time expires and give them an option to extend the time limit

Screenshot(s)

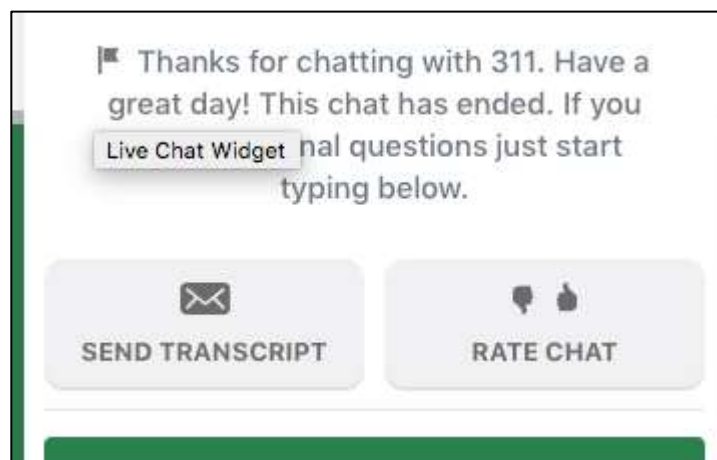


Figure 134a: - Screenshot of end of live chat session

Bugs

Non required field gives error message that it is required (CLT-36)

Components	006 - charlottenc.gov - Report Broken Parking Meter
Affected community	
WCAG Success Criterion	
WCAG Level	
Priority	Medium

Description

There are fields that aren't required. But if focus is set to the field, then moved away, the error message that the field is required is shown.

Screenshot(s)



Figure 36a: - Screenshot of error message

Extraneous markup (CLT-66)

Components	003 - charlottenc.gov - Services - CharMeck 311 (drop downs), 009 - charlottenc.gov - HR - Jobs - See Available Jobs
Affected community	
WCAG Success Criterion	
WCAG Level	
Priority	Low

Description

There is extraneous markup at the bottom of the page. These HTML elements are announced to screen reader users even though there is no content available.

Screenshot(s)

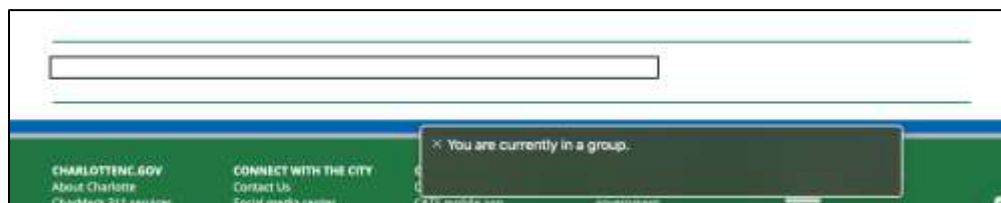


Figure 66a: - Screenshot of elements with extra markup



Figure 66b: - Screenshot of element with extra markup

Interactive map isn't loading (CLT-67)

Components	010 - charlottenc.gov - Charlotte Future (carousel and interactive map)
Affected community	
WCAG Success Criterion	
WCAG Level	
Priority	High

Description

The map that should be displayed in the lower portion of this page isn't rendering. The issue appears to be related to the fact the map is non-secure content, but the site is secure (HTTPS). However, there are also several JavaScript issues on the page, that could be contributing, as well.

Linked issues

- CLT-100

Screenshot(s)



Figure 67a: - Screenshot of code inspector with errors for the interactive map

Conclusion

The City of Charlotte pages reviewed in this assessment contain a large number of barriers that could block access to information by some users.

Particular attention should be paid to color contrast issues which were common throughout the site. One way to improve this and reduce color contrast risk in future updates is to establish an approved color palette and requiring all page content meet the requirements of that palette. This would eliminate consistency issues observed on the subset of pages in our assessment.

ARIA has been used on many elements throughout the site, apparently in an effort to increase accessibility. ARIA should not be used unless there is no way to accomplish the same thing using native HTML. When used, ARIA should respect the standard, use only accepted roles, and be applied only on allowed elements. The extensive and poor use of ARIA not only complicates the code but makes it more likely that assistive technology users will encounter barriers.

Keyboard-only users, sighted or not, will encounter multiple barriers. These include lack of visible focus indicator for sighted users, or interactive elements that can only be triggered with a mouse. Tab order has also been forced in some cases with a series of elements with positive tabindexes. In other cases, elements have been completely removed from the tab order with negative tabindexes.

Most forms encountered, from the simple search form to more complex board application forms are problematic. Input elements are lacking labels, or share the same label between two elements. Additional information for input field isn't programmatically associated with the input, meaning screen reader users won't have access to the information, or only with difficulty. Error messages are also not linked to the erroneous input fields.

Multiple areas expose content on user action. These dynamic changes of content aren't marked up in a way that allows screen readers to announce the change to their users.

The date picker and live chat widgets are filled with errors and it might be easier to find alternative solutions for these than to try and fix this 3rd party content.

A number of various other issues were identified and reported on, but can't easily be categorized in this narrative conclusion.

Once your team has reviewed these findings, we welcome the opportunity to provide more detailed, in person feedback that will help you make this useful resource more broadly accessible. Thanks for the opportunity to work with you.

Contact Information



Knowbility, Inc.
1033 La Posada, Suite 372

Austin, Texas 78754
512 527-3138

Appendix 1: Web Content Accessibility Guidelines

Table 1: Success Criteria, Level A and AA

Criteria	Level	URL
1.1.1 Non-text Content	A	https://www.w3.org/TR/WCAG20/#text-equiv-all
1.2.1 Audio-only and Video-only (Prerecorded)	A	https://www.w3.org/TR/WCAG20/#media-equiv-av-only-alt
1.2.2 Captions (Prerecorded)	A	https://www.w3.org/TR/WCAG20/#media-equiv-captions
1.2.3 Audio Description or Media Alternative (Prerecorded)	A	https://www.w3.org/TR/WCAG20/#media-equiv-audio-desc
1.2.4 Captions (Live)	AA	http://www.w3.org/TR/WCAG20/#media-equiv-real-time-captions
1.2.5 Audio Description (Prerecorded)	AA	http://www.w3.org/TR/WCAG20/#media-equiv-audio-desc-only
1.3.1 Info and Relationships	A	https://www.w3.org/TR/WCAG20/#content-structure-separation-programmatic
1.3.2 Meaningful Sequence	A	https://www.w3.org/TR/WCAG20/#content-structure-separation-sequence
1.3.3 Sensory Characteristics	A	https://www.w3.org/TR/WCAG20/#content-structure-separation-understanding
1.4.1 Use of Color	A	https://www.w3.org/TR/WCAG20/#visual-audio-contrast-without-color
1.4.2 Audio Control	A	https://www.w3.org/TR/WCAG20/#visual-audio-contrast-dis-audio
1.4.3 Contrast (Minimum)	AA	http://www.w3.org/TR/WCAG20/#visual-audio-contrast-contrast
1.4.4 Resize Text	AA	http://www.w3.org/TR/WCAG20/#visual-audio-contrast-scale

1.4.5 Images of Text	AA	http://www.w3.org/TR/WCAG20/#visual-audio-contrast-text-presentation
2.1.1 Keyboard	A	https://www.w3.org/TR/WCAG20/#keyboard-operation-keyboard-operable
2.1.2 No Keyboard Trap	A	https://www.w3.org/TR/WCAG20/#keyboard-operation-trapping
2.2.1 Timing Adjustable	A	https://www.w3.org/TR/WCAG20/#time-limits-required-behaviors
2.2.2 Pause, Stop, Hide	A	https://www.w3.org/TR/WCAG20/#time-limits-pause
2.3.1 Three Flashes or Below Threshold	A	https://www.w3.org/TR/WCAG20/#seizure-does-not-violate
2.4.1 Bypass Blocks	A	https://www.w3.org/TR/WCAG20/#navigation-mechanisms-skip
2.4.2 Page Titled	A	https://www.w3.org/TR/WCAG20/#navigation-mechanisms-title
2.4.3 Focus Order	A	http://www.w3.org/TR/WCAG20/#navigation-mechanisms-focus-order
2.4.4 Link Purpose (In Context)	A	http://www.w3.org/TR/WCAG20/#navigation-mechanisms-refs
2.4.5 Multiple Ways	AA	http://www.w3.org/TR/WCAG20/#navigation-mechanisms-mult-loc
2.4.6 Headings and Labels	AA	http://www.w3.org/TR/WCAG20/#navigation-mechanisms-descriptive
2.4.7 Focus Visible	AA	http://www.w3.org/TR/WCAG20/#navigation-mechanisms-focus-visible
3.1.1 Language of Page	A	http://www.w3.org/TR/WCAG20/#meaning-doc-lang-id
3.1.2 Language of Parts	AA	http://www.w3.org/TR/WCAG20/#meaning-other-lang-id

3.2.1 On Focus	A	http://www.w3.org/TR/WCAG20/#consistent-behavior-receive-focus
3.2.2 On Input	A	http://www.w3.org/TR/WCAG20/#consistent-behavior-unpredictable-change
3.2.3 Consistent Navigation	AA	http://www.w3.org/TR/WCAG20/#consistent-behavior-consistent-locations
3.2.4 Consistent Identification	AA	http://www.w3.org/TR/WCAG20/#consistent-behavior-consistent-functionality
3.3.1 Error Identification	A	http://www.w3.org/TR/WCAG20/#minimize-error-identified
3.3.2 Labels or Instructions	A	http://www.w3.org/TR/WCAG20/#minimize-error-cues
3.3.3 Error Suggestion	AA	http://www.w3.org/TR/WCAG20/#minimize-error-suggestions
3.3.4 Error Prevention (Legal, Financial, Data)	AA	http://www.w3.org/TR/WCAG20/#minimize-error-reversible
4.1.1 Parsing	A	http://www.w3.org/TR/WCAG20/#ensure-compat-parses
4.1.2 Name, Role, Value	A	http://www.w3.org/TR/WCAG20/#ensure-compat-rsv

PDF Accessibility Report

File Name: certificateofdisabilityform 1

Document Title: Charlotte Area Transit system Certificate of Disability

Knowbility reviewed the PDF document(s) submitted by **Customer** to determine if the document(s) meet accessibility compliance standards for persons with disabilities who use assistive technology.

Standards: For accessibility compliance, Knowbility tested against the Web Content Accessibility Guidelines, version 2, Level AA guidelines for PDF accessibility - [W3C - PDF Techniques for WCAG 2.0](#). WCAG is the global standard, developed by member organizations of the World Wide Web Consortium (W3C), the standards makers for the web. WCAG 2.0 standards are consistent and compatible with US federal standards for accessibility as described [Section 508](#), [Section 504](#) and [IDEA](#) as well as individual states' accessibility statutes.

Method: Testing consisted of manual inspection using Adobe Acrobat Pro DC. Testing was conducted by members of Knowbility's document accessibility team.

Report: **Pass/Fail** status (compliant/non-compliant) for 15 essential document accessibility issues are listed along with accompanying notes and standards references for each issue. Links to the WCAG 2.0 and/or the US Section 508 document accessibility standard with descriptions and examples of how to achieve the desired conformance outcome.

Resources for Understanding Accessible PDF and MS Office Docs

- [W3C - PDF Techniques for WCAG 2.0](#)
- [W3C - PDF Techniques for Web Content Accessibility Guidelines 1.0 and 2.0](#)
- [HHS - PDF File 508 Checklist](#)
- [Over-reliance on automated PDF accessibility checkers](#)
- [Webaim – PDF Accessibility](#)
- [HHS - HHS Section 508 Accessibility checklists for PDF and MS Office Docs](#)
- [International Language Codes](#)

Summary of Findings

We found multiple accessibility barriers, listed below, that will prevent blind screen reader users, low-vision users and others who use assistive technology from accessing the content of this document.

- Document is not tagged – multiple omissions and failures
- There are no Heading tags
- Reading Order is incorrect; will have to be properly set after tag structure has been created. When a document is properly tagged, including Alt Text for all non-text elements, has a proper heading structure, then the Reading Order can be configured for screen reader software. This makes the document is “readable”, i.e. accessible to blind users and other users of assistive technology.

1. Charlotte Area Transit System Certificate of Disability

Issue 1 – Title **Pass** (Note: “Microsoft Word – Disability.doc” is not a very descriptive title.)

WCAG 2.0 Guideline

Document titles identify the current location without requiring users to read or interpret page content. User agents make the title of the page easily available to the user for identifying the page. For instance, a user agent may display the page title in the window title bar or as the name of the tab containing the page.

[2.4.2: PDF18: Specifying the document title using the Title entry in the document information dictionary of a PDF document](#)

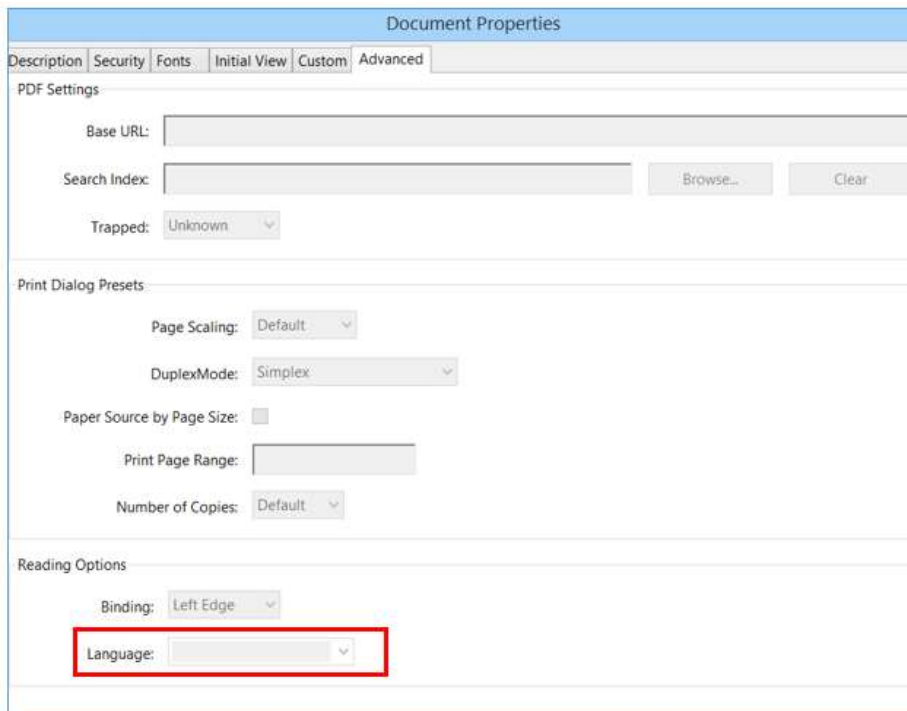
Issue 2 – Language Specified **Fail**

WCAG 2.0 Guideline

Both assistive technologies and conventional user agents can render text more accurately when the language of the document is identified. Screen readers can load the correct pronunciation rules. Visual browsers can display characters and scripts correctly. Media players can show captions correctly. As a result, users with disabilities are better able to understand the content.

[3.1.1 PDF16: Setting the default language using the /Lang entry in the document catalog of a PDF document](#)

[3.1.2 PDF19: Specifying the language for a passage or phrase with the Lang entry in PDF documents](#)



The screenshot shows the 'Document Properties' dialog box with the 'Advanced' tab selected. The 'Language' field in the 'Reading Options' section is highlighted with a red rectangle. The 'Language' field is a dropdown menu currently showing 'Unknown'.

Document Properties					
Description	Security	Fonts	Initial View	Custom	Advanced
PDF Settings					
Base URL:	<input type="text"/>				
Search Index:	<input type="text"/>	Browse...	Clear		
Trapped:	Unknown				
Print Dialog Presets					
Page Scaling:	Default				
DuplexMode:	Simplex				
Paper Source by Page Size:	<input type="checkbox"/>				
Print Page Range:	<input type="text"/>				
Number of Copies:	Default				
Reading Options					
Binding:	Left Edge				
Language:	<input type="text"/>				

Issue 3 – OCR N/A

WCAG 2.0 Guideline

A document that consists of scanned images of text is inherently inaccessible because the content of the document is images, not searchable text. Assistive technologies cannot read or extract the words; users cannot select, edit, resize, or reflow text nor can they change text and background colors; and authors cannot manipulate the PDF for accessibility. For these reasons, authors should use actual text rather than images of text

[1.4.5 PDF7: Performing OCR on a scanned PDF document to provide actual text](#)

[1.4.9 Images of text](#) are only used for [pure decoration](#)

Issue 4 – Tagging Fail – The document is not tagged

WCAG 2.0 Guideline

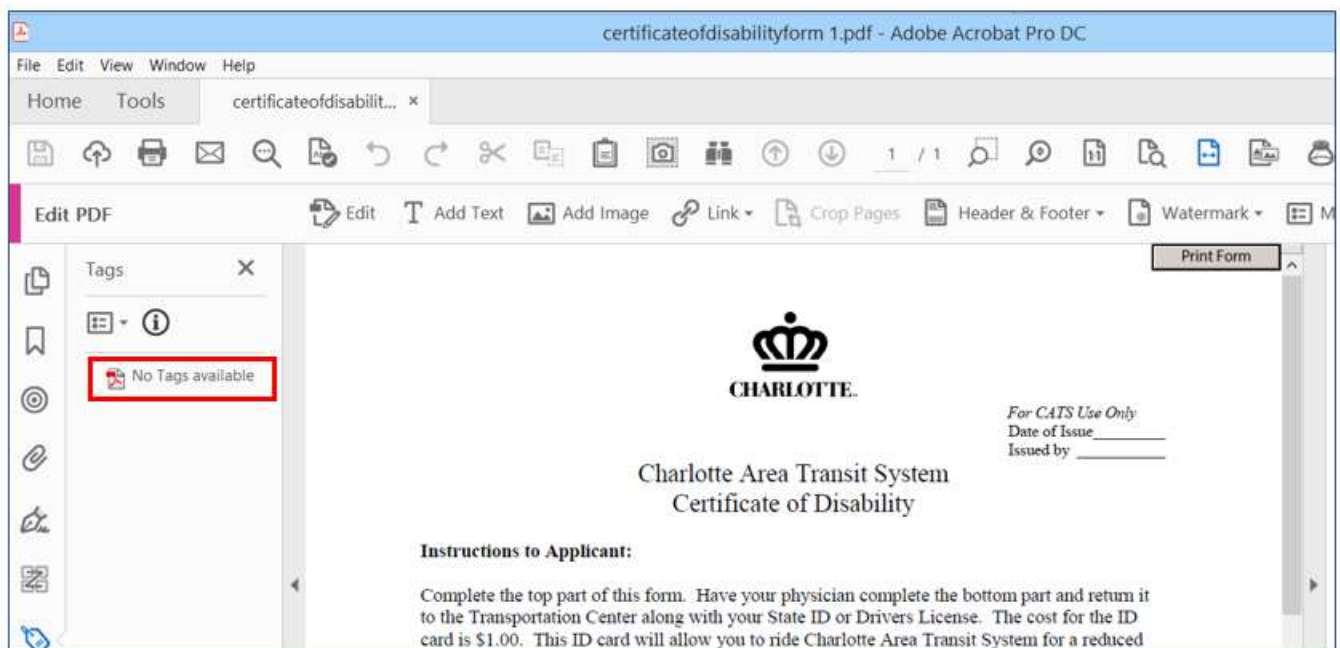
Tagged PDF builds on PDF's logical structure framework. It defines a set of standard structure types and attributes that allow page content (text, graphics, and images) to be extracted and reused for other purposes. Tagged PDFs make content accessible to people who rely on assistive technology.

[1.1.1 PDF4: Hiding decorative images with the Artifact tag in PDF documents](#)

[1.3.1 PDF9: Providing headings by marking content with heading tags in PDF documents](#)

[2.4.1 PDF20: Using Adobe Acrobat Pro's Table Editor to repair mis-tagged tables](#)

[PDF21: Using List tags for lists in PDF documents](#)



Issue 5 – Reading Order Markup Fail – Reading Order is not set

WCAG 2.0 Guideline

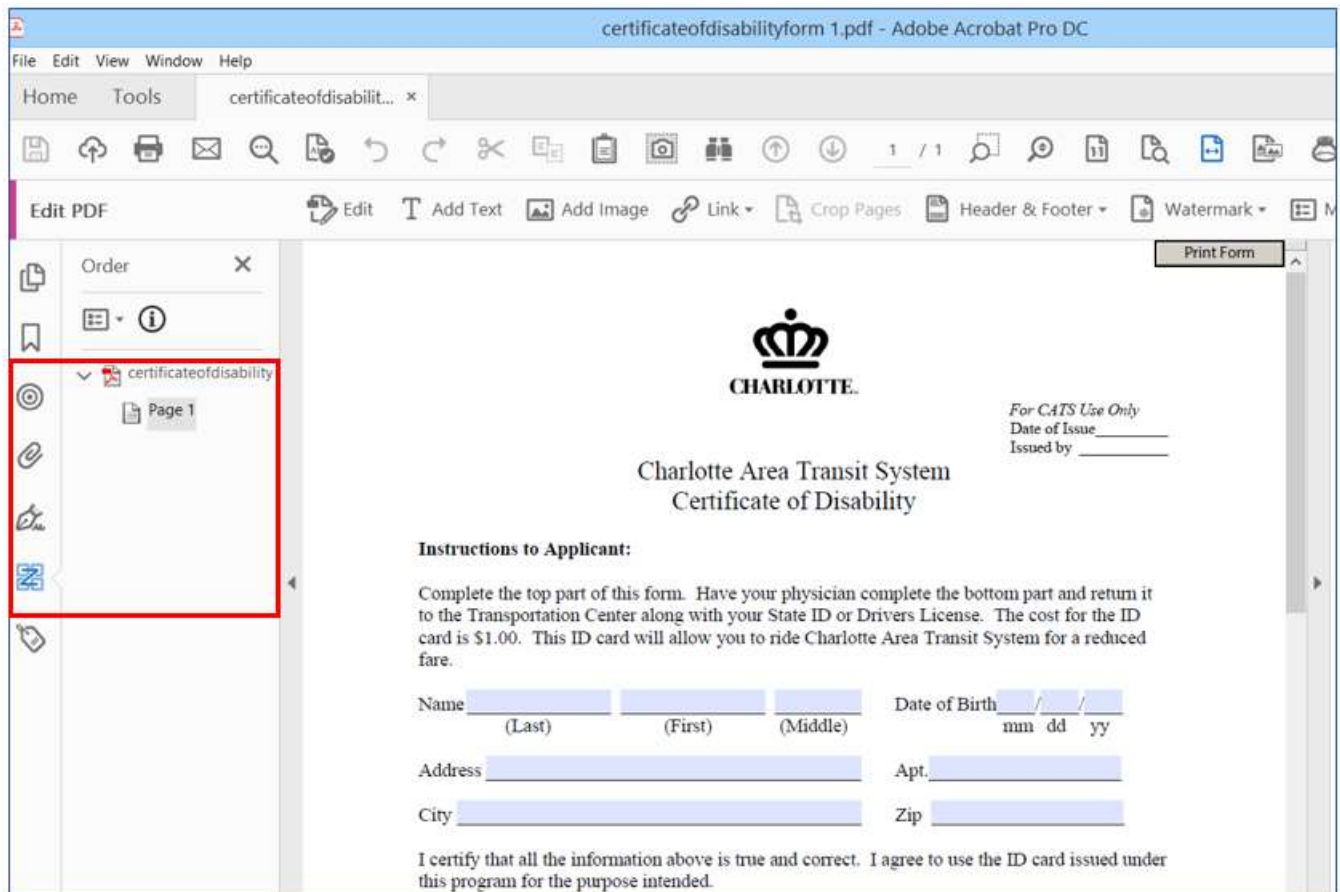
Logical structure is created when a document is saved as tagged PDF. The reading order of a PDF document is determined primarily by the tag order of document elements, including interactive elements, but the order of content within individual tags is determined by the PDF document's content tree structure. If the reading order is not correct, keyboard and assistive technology users may not be able to understand the content. For example, some documents use multiple columns, and the reading order is clear visually to sighted users as flowing from the top to the bottom of the first column, then to the top of the next column. But if the document is not properly tagged, a screen reader may read the document from top to bottom, across both columns, interpreting them as one column.

[1.3.2 PDF3: Ensuring correct tab and reading order in PDF documents](#)

[2.1.1](#)

[2.1.3](#)

[2.4.3](#)



certificateofdisabilityform 1.pdf - Adobe Acrobat Pro DC

File Edit View Window Help

Home Tools certificateofdisabilit... x

Edit PDF

Order

certificateofdisability

Page 1

CHARLOTTE.

Charlotte Area Transit System
Certificate of Disability

For CATS Use Only
Date of Issue _____
Issued by _____

Instructions to Applicant:

Complete the top part of this form. Have your physician complete the bottom part and return it to the Transportation Center along with your State ID or Drivers License. The cost for the ID card is \$1.00. This ID card will allow you to ride Charlotte Area Transit System for a reduced fare.

Name _____ Date of Birth _____
(Last) (First) (Middle) mm dd yy

Address _____ Apt. _____

City _____ Zip _____

I certify that all the information above is true and correct. I agree to use the ID card issued under this program for the purpose intended.

Issue 6 – Reading Order Headings **Fail**

WCAG 2.0 Guideline

The intent of this Success Criterion is to ensure that information and relationships that are implied by visual or auditory formatting are preserved when the presentation format changes. For example, the presentation format changes when the content is read by a screen reader or when a user style sheet is substituted for the style sheet provided by the author. Having content structures and relationships programmatically determined or available in text ensures that information important for comprehension will be perceivable to all.

[1.3.2 PDF3: Ensuring correct tab and reading order in PDF documents](#)

[2.1.1](#)

[2.1.3](#)

[2.4.3](#)

Issue 7 – Alt Text **Fail**

WCAG 2.0 Guideline

Providing text alternatives allows non-text content to be rendered in a variety of ways by a variety of user agents. For example, a person who cannot see a picture can have the text alternative read aloud using synthesized speech. A person who cannot hear an audio file can have the text alternative displayed so that he or she can read it. In the future, text alternatives will also allow information to be more easily translated into sign language or into a simpler form of the same language.

[1.1.1 PDF1: Applying text alternatives to images with the Alt entry in PDF documents](#)

Ex. 1 Charlotte logo is not tagged as a <Figure> and, therefore, has no Alt Text in the “Properties”



Ex. 2 CATs logo is not tagged as a <Figure> and, therefore, has no Alt Text in the “Properties”



Issue 8 – Form Fields **Fail** – Form Fields are not tagged with properties set
WCAG 2.0 Guideline

The intent of this success criterion is to have content authors place instructions or labels that identify the controls in a form so that users know what input data is expected.

[3.3.1 PDF5: Indicating required form controls in PDF forms](#)

[3.3.2 PDF10: Providing labels for interactive form controls in PDF documents](#)

[3.3.3 PDF12: Providing name, role, value information for form fields in PDF documents](#)

[1.3.1 PDF22: Indicating when user input falls outside the required format or values in PDF forms](#)

[4.1.2 PDF23: Providing interactive form controls in PDF documents](#)

[3.2.2](#)

[2.1.1](#)

[2.1.3](#)

Issue 9 – Tables **N/A**

WCAG 2.0 Guideline

Tabular information must be presented in a way that preserves relationships within the information even when users cannot see the table or the presentation format is changed. Information is considered tabular when logical relationships among text, numbers, images, or other data exist in two dimensions (vertical and horizontal). These relationships are represented in columns and rows, and the columns and rows must be recognizable in order for the logical relationships to be perceived.

[1.3.1 PDF6: Using table elements for table markup in PDF Documents](#)

[PDF20: Using Adobe Acrobat Pro's Table Editor to repair mistagged tables](#)

Issue 10 – Links not working **N/A**

Section 508 Guideline

Are all URLs linked to correct Web destinations?

[Section 508 – HHS Checklist: 1.8, 1.9 & 1.10](#)

Issue 11 - Hyperlinks not qualified or inactive **N/A**

Section 508 Guideline

Do all URLs contain descriptive hyperlinks (i.e., avoid generic phrases like “Click here” and, instead, use phrases that let users know about the content of the linked page prior to selecting it)?

[Section 508 – HHS Checklist: 1.8, 1.9 & 1.10](#)

WCAG 2.0 Guideline

Link text in PDF documents can be marked up to be recognizable by keyboard and assistive technology users. That is, the link information is programmatically available to user agents so that links are recognizable when presented in a different format. This is typically accomplished by using a tool for authoring PDF.

[1.3.1 PDF11: Providing links and link text using the Link annotation and the /Link structure element in PDF documents](#)

[2.1.1](#)

[2.1.3](#)

[2.4.4](#)

[2.4.9](#)

Issue 12 – Bookmarks N/A

WCAG 2.0 Guideline

The intent of this Success Criterion is to make it possible for users to locate content in a manner that best meets their needs.

[2.4.5 PDF2: Creating bookmarks in PDF documents](#)

Issue 13 - Table of Contents Unlinked N/A

Section 508 Guideline

Are all internal links/TOC entries functioning correctly (if linked)?

[Section 508 – HHS Checklist: 1.10](#)

Issue 14 – Table of Contents Linked N/A

Section 508 Guideline

Are all internal links/TOC entries functioning correctly (if linked)?

[Section 508 – HHS Checklist: 1.10](#)

Issue 15 – Color Contrast N/A

Section 508 Guideline

Is the color contrast ratio between foreground text and background color at least 4.5:1?

[Section 508 – HHS checklist: 1.13](#)

WCAG 2.0 Guideline

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

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

[1.4.3 G18: Ensuring that a contrast ratio of at least 4.5:1 exists between text \(and images of text\) and background behind the text](#)