

# Manual Review Help Guide

## Table of Contents

1.3.2 Meaningful Sequence .....	3
Overview .....	3
Specific Benefits of Success Criterion 1.3.2: .....	3
Success Criterion .....	3
1.3.3 Sensory Characteristics .....	5
Overview .....	5
Specific Benefits of Success Criterion 1.3.3: .....	5
Success Criterion .....	5
1.4.4 Resize text .....	6
Overview .....	6
Success Criterion .....	6
2.1.1 Keyboard .....	7
Overview .....	7
Specific Benefits of Success Criterion 2.1.1: .....	8
Success Criterion .....	8
2.1.2 No Keyboard Trap .....	9
Overview .....	9
Success Criterion .....	10
2.2.2 Pause, Stop, Hide .....	10
Overview .....	10
Specific Benefits of Success Criterion 2.2.2: .....	11
Success Criterion .....	11
2.3.1 Three Flashes or Below Threshold .....	13
Overview .....	13
Specific Benefits of Success Criterion 2.3.1: .....	13
Success Criterion .....	14
3.2.1 On Focus.....	15
Overview .....	15
Specific Benefits of Success Criterion 3.2.1: .....	15

Success Criterion .....	16
3.2.3 Consistent Navigation .....	17
Overview .....	17
Specific Benefits of Success Criterion 3.2.3: .....	17
Success Criterion .....	17
3.2.4 Consistent Identification .....	18
Overview .....	18
Specific Benefits of Success Criterion 3.2.4: .....	18
Success Criterion .....	19
3.3.4 Error Prevention (Legal, Financial, Data) .....	19
Overview .....	19
Specific Benefits of Success Criterion 3.3.4: .....	19
Success Criterion .....	19

## 1.3.2 Meaningful Sequence

### Overview

When the sequence in which content is presented affects its meaning, a correct reading sequence can be programmatically determined. Content that does not meet this Success Criterion may confuse or disorient users when assistive technology reads the content in the wrong order, or when alternate style sheets or other formatting changes are applied.

For clarity:

1. Providing a particular linear order is only required where it affects meaning. For example, the relative order of the main section of a Web page and a navigation section does not affect their meaning.
2. There may be more than one order that is "correct" (according to the WCAG 2.0 definition).
3. Only one correct order needs to be provided.

### Specific Benefits of Success Criterion 1.3.2:

This Success Criterion may help people who rely on assistive technologies that read content aloud. The meaning evident in the sequencing of the information in the default presentation will be the same when the content is presented in spoken form.

[Learn more about 1.3.2 Meaningful Sequence](#)

### Success Criterion

All items below must be true to meet the success criterion for 1.3.2

- Check to ensure that tables are NOT created using white space characters to layout the tabular data.**

Ref #	Description	Example
<a href="#">F34</a>	The objective of this technique is to describe how using white space characters, such as space, tab, line break, or carriage return, to format tables in text content is a failure to use structure properly. When tables are created in this manner there is no way to indicate that a cell is intended to be a header cell, no way to associate the table header cells with the table data cells, or to navigate directly to a particular cell in a table.	<a href="#">Example</a>

Related techniques that will help resolve this accessibility issue:

- [Using table markup to present tabular information](#)

- Check to ensure that the columns are NOT created using white space characters to lay out the information.**

Ref #	Description	Example
<a href="#">F33</a>	The objective of this technique is to describe how using white space characters, such as space, tab, line break, or carriage return, to format columns of data in text content is a failure to use structure properly. Assistive technologies will interpret content in the reading order of the current language. Using white space characters to create multiple columns does not provide the information in a natural reading order. Thus, the assistive technology user will not be presented with the information in an understandable manner.	<a href="#">Example</a>

- Ensure that none of the words on the web page contain white space characters.**

Ref #	Description	Example
<a href="#">F32</a>	The objective of this technique is to describe how using white space characters, such as space, tab, line break, or carriage return, to format individual words visually can be a failure to present meaningful sequences properly. When blank characters are inserted to control letter spacing within a word, they may change the interpretation of the word or cause it not to be programmatically recognized as a single word.	<a href="#">Example</a>

Related techniques that will help resolve this accessibility issue:

- [Using CSS letter-spacing to control spacing within a word](#)

- Check that the linear reading order matches any meaningful sequence conveyed through presentation.**

Ref #	Description	Example
<a href="#">F49</a>	Avoid using tables for page layout (otherwise known as a layout table). Instead, opt to use tables for organizing data and HTML/CSS for page content. If a layout table is used ensure there is a proper linearized reading order, content scaling, etc.	<a href="#">Example 1</a> <a href="#">Example 2</a>

Related techniques that will help resolve this accessibility issue:

- [Positioning content based on structural markup](#)

- Check that the reading order of the content is correct and the meaning of the content is preserved.**

Ref #	Description	Example
<a href="#">F1</a>	CSS, rather than HTML, is used to modify the visual layout of the content and the modified layout changes the meaning of the content.	<a href="#">Example</a>

Related techniques that will help resolve this accessibility issue:

- [Positioning content based on structural markup](#)

### 1.3.3 Sensory Characteristics

#### Overview

The intent of this Success Criterion is to ensure that all users can access instructions for using the content, even when they cannot perceive shape or size or use information about spatial location or orientation. Some content relies on knowledge of the shape or position of objects that are not available from the structure of the content (for example, "round button" or "button to the right"). Some users with disabilities are not able to perceive shape or position due to the nature of the assistive technologies they use. This Success Criterion requires that additional information be provided to clarify anything that is dependent on this kind of information.

For clarity:

People who are blind and people who have low vision may not be able to understand information if it is conveyed by shape and/or location. Providing additional information other than shape and/or location will allow them to understand the information conveyed by shape and/or alone.

#### Specific Benefits of Success Criterion 1.3.3:

People who are blind and people who have low vision may not be able to understand information if it is conveyed by shape and/or location. Providing additional information other than shape and/or location will allow them to understand the information conveyed by shape and/or alone.

[Learn more about 1.3.3 sensory characteristics](#)

#### Success Criterion

All items below must be true to meet the success criterion for 1.3.3

- Check that the references do not rely on only the visual shape or location of the content.**

Ref #	Description	Example
<a href="#">F14</a>	The objective of this technique is to show how identifying content only by its visual shape or location makes content difficult to understand and operate. When only visual identification or location is used, users with visual disabilities may find it difficult to locate content since they cannot see the screen or may perceive only a small portion of the screen at one time. Also, location of content can vary if page layout varies due to variations in font, window, or screen size.	<a href="#">Example</a>

Related techniques that will help resolve this accessibility issue:

- [Providing textual identification of items that otherwise rely only on sensory information to be understood.](#)

- Examine the page for non-text marks that convey information. Check whether there are other means to determine the information conveyed by the non-text marks.**

Ref #	Description	Example
<a href="#">F26</a>	Using a graphical symbol to convey information can make content difficult to comprehend for some readers. A graphical symbol may be an image, an image of text or a pictorial or decorative character symbol (glyph) which imparts information nonverbally.	<a href="#">Example</a>

Related techniques that will help resolve this accessibility issue:

- [Using alt attributes on img elements](#)

## 1.4.4 Resize text

### Overview

Except for captions and images of text, text can be resized without assistive technology up to 200 percent without loss of content or functionality. This Success Criterion helps people with low vision by letting them increase text size in content so that they can read it.

[Learn more about 1.4.4 resize text](#)

### Success Criterion

All items below must be true to meet the success criterion for 1.4.4

- Increase the text size of the content by 200%. Check that no text is clipped, truncated, or obscured.**

Ref #	Description	Example
<a href="#">F69</a>	The objective of this failure condition is to describe a problem that occurs when changing the size of text causes text to be clipped, truncated, or obscured, so that it is no longer available to the user.	<a href="#">Example</a>

Related techniques that will help resolve this accessibility issue:

- [Providing textual identification of items that otherwise rely only on sensory information to be understood.](#)

- Check that the text in text-based form controls can be increased by 200%**

Ref #	Description	Example
<a href="#">F80</a>	Text-based form controls include input boxes (text and textarea) as well as buttons. The objective of this failure condition is to describe a problem that occurs when changing the size of text does not cause the text-based form controls to resize accordingly.	<a href="#">Example</a>

Related techniques that will help resolve this accessibility issue:

- [Scaling form elements which contain text](#)

## 2.1.1 Keyboard

### Overview

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

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

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

For clarity:

The intent of this Success Criterion is to ensure that, wherever possible, content can be operated through a keyboard or keyboard interface (so an alternate keyboard can be used). When content can be

operated through a keyboard or alternate keyboard, it is operable by people with no vision (who cannot use devices such as mice that require eye-hand coordination) as well as by people who must use alternate keyboards or input devices that act as keyboard emulators. Keyboard emulators include speech input software, sip-and-puff software, on-screen keyboards, scanning software and a variety of assistive technologies and alternate keyboards. Individuals with low vision also may have trouble tracking a pointer and find the use of software much easier (or only possible) if they can control it from the keyboard.

#### Specific Benefits of Success Criterion 2.1.1:

- People who are blind (who cannot use devices such as mice that require eye-hand coordination)
- People with low vision (who may have trouble finding or tracking a pointer indicator on screen)
- Some people with hand tremors find using a mouse very difficult and therefore usually use a keyboard

[Learn more about 2.1.1 Keyboard](#)

#### Success Criterion

All items below must be true to meet the success criterion for 2.1.1

- Ensure that pointing-device-specific event handlers are not the only means to invoke scripting functions.**

Ref #	Description	Example
<a href="#">F54</a>	When pointing device-specific event handlers are the only mechanism available to invoke a function of the content, users with no vision (who cannot use devices such as mice that require eye-hand coordination) as well as users who must use alternate keyboards or input devices that act as keyboard emulators will be unable to access the function of the content.	<a href="#">Example</a>

Related techniques that will help resolve this accessibility issue:

- [Using both keyboard and other device-specific functions](#)



- **Use the keyboard to verify that you can get to all interactive elements using the keyboard. Check that when focus is placed on each element, focus remains there until user moves it.**

Ref #	Description	Example
<a href="#">F55</a>	Content that normally receives focus when the content is accessed by keyboard may have this focus removed by scripting. This is sometimes done when designer considers the system focus indicator to be unsightly. However, the system focus indicator is an important part of accessibility for keyboard users. In addition, this practice removes focus from the content entirely, which means that the content can only be operated by a pointing device such as a mouse.	<a href="#">Example</a>

- **For all elements presented as links which use JavaScript event handlers to make the element emulate a link, check if the emulated link can be activated using the keyboard.**

Ref #	Description	Example
<a href="#">F42</a>	This failure occurs when JavaScript event handlers are attached to elements to emulate links. A link created in this manner cannot be tabbed to from the keyboard and does not gain keyboard focus like other controls and/or links. If scripting events are used to emulate links, user agents including assistive technology may not be able to identify the links in the content as links. They may be recognized as interactive controls but still not recognized as links. Such elements do not appear in the links list generated by user agents or assistive technology.	<a href="#">Example</a>

Related techniques that will help resolve this accessibility issue:

- [Using semantic elements to mark up structure](#)

## 2.1.2 No Keyboard Trap

### Overview

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

The intent of this Success Criterion is to ensure that that content does not "trap" keyboard focus within subsections of content on a Web page. There may be times when the functionality of the Web page restricts the focus to a subsection of the content, as long as the user knows how to leave that state and "untrap" the focus.

For clarity:

An example of a keyboard trap is when a web application brings up a dialog box. At the bottom of the dialog are two buttons, Cancel and OK. When the dialog has been opened, focus is trapped within the dialog; tabbing from the last control in the dialog takes focus to the first control in the dialog. The dialog is dismissed by activating the Cancel button or the OK button.

[Learn more about 2.1.2 No Keyboard Trap](#)

### Success Criterion

All items below must be true to meet the success criterion for 2.1.2

- Check to see that the keyboard focus is not "trapped" and it is possible to move keyboard focus out of the plug-in content without closing the user agent or restarting the system.**

Ref #	Description	Example
<a href="#">F10</a>	When content includes multiple formats, one or more user agents or plug-ins are often needed in order to successfully present the content to users. For example, a page that includes XHTML, SVG, SMIL and XForms may require a browser to load as many as three different plug-ins in order for a user to successfully interact with the content. Some plug-ins create a common situation in which the keyboard focus can become "stuck" in a plug-in, leaving a keyboard-only user with no way to return to the other content.	<a href="#">Example</a>

Related techniques that will help resolve this accessibility issue:

- [Ensuring that users are not trapped in content](#)

## 2.2.2 Pause, Stop, Hide

### Overview

The intent of this Success Criterion is to avoid distracting users during their interaction with a Web page. For moving, blinking, scrolling, or auto-updating information, all of the following are true:

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

hide it or to control the frequency of the update unless the auto-updating is part of an activity where it is essential.

"Moving, blinking and scrolling" refers to content in which the visible content conveys a sense of motion. Common examples include motion pictures, synchronized media presentations, animations, real-time games, and scrolling stock tickers. "Auto-updating" refers to content that updates or disappears based on a preset time interval. Common time-based content includes audio, automatically updated weather information, news, stock price updates, and auto-advancing presentations and messages.

### Specific Benefits of Success Criterion 2.2.2:

Providing content that stops blinking after five seconds or providing a mechanism for users to stop blinking content allows people with certain disabilities to interact with the Web page.

One use of content that blinks is to draw the visitor's attention to that content. Although this is an effective technique for all users with vision, it can be a problem for some users if it persists. For certain groups, including people with low literacy, reading and intellectual disabilities, and people with attention deficit disorders, content that blinks may make it difficult or even impossible to interact with the rest of the Web page.

[Learn more about 2.2.2 Pause, Stop, Hide](#)

### Success Criterion

All items below must be true to meet the success criterion for 2.2.2

- On a page with moving or scrolling content, check that a mechanism is provided in the Web page or user agent to pause moving or scrolling content.**
- Check that a mechanism is provided in the Web page or user agent to restart the paused content.**

Ref #	Description	Example
<a href="#">F16</a>	In this failure technique, there is moving or scrolling content that cannot be paused and resumed by users. In this case, some users with low vision or cognitive disabilities will not be able to perceive the content.	<a href="#">Example</a>

Related techniques that will help resolve this accessibility issue:

- [Allowing the content to be paused and restarted from where it was paused](#)

- Ensure the blink element is not being used.**

Ref #	Description	Example
<a href="#">F47</a>	The blink element, while not part of the official HTML or XHTML specification, is supported by many user agents. It causes any text inside the element to blink at a predetermined rate. This cannot be interrupted by the user, nor can it be disabled as a preference. The blinking continues as long as the page is displayed. Therefore, content that uses blink fails the Success Criterion because blinking can continue for more than three seconds.	<a href="#">Example</a>

- Ensure that the inline styles, internal stylesheets, and external stylesheets do not contain the text-decoration property with a value of "blink". If the property is used, determine if the ID, class, or element identified by selectors on which this property is defined are not used within the document.**

Ref #	Description	Example
<a href="#">F4</a>	CSS defines the blink value for the text-decoration property. When used, it causes any text in elements with this property to blink at a predetermined rate. This cannot be interrupted by the user, nor can it be disabled as a user agent preference. The blinking continues as long as the page is displayed. Therefore, content that uses text-decoration:blink fails the Success Criterion because blinking can continue for more than three seconds.	<a href="#">Example</a>

Related techniques that will help resolve this accessibility issue:

- [Using scripts to control blinking and stop it in five seconds or less](#)

- For each instance of blinking content, determine if the blinking stops in 5 seconds or less.**

Ref #	Description	Example
<a href="#">F50</a>	Scripts can be used to blink content by toggling the content's visibility on and off at regular intervals. It is a failure for the script not to include a mechanism to stop the blinking at 5 seconds or earlier.	<a href="#">Example</a>

Related techniques that will help resolve this accessibility issue:

- [Using scripts to control blinking and stop it in five seconds or less](#)

- **For each page that has blinking content in a plugin or applet, ensure that the content does not continue to blink for longer than 5 seconds and that there is a means to pause the blinking content.**

Ref #	Description	Example
<a href="#">F7</a>	When content that is rendered by a plug-in or contained in an applet blinks, there may be no way for the user agent to pause the blinking. If neither the plug-in, applet, nor the content itself provides a mechanism to pause the content, the user may not have sufficient time to read the content between blinks or it may be so distracting that the user will not be able to read other content on the page.	<a href="#">Example</a>

Related techniques that will help resolve this accessibility issue:

- [Using scripts to control blinking and stop it in five seconds or less](#)

## 2.3.1 Three Flashes or Below Threshold

### Overview

The intent of this Success Criterion is to allow users to access the full content of a site without inducing seizures due to photosensitivity. Web pages should not contain anything that flashes more than three times in any one second period, or the flash is below the [general flash and red flash thresholds](#).

Individuals who have photosensitive seizure disorders can have a seizure triggered by content that flashes at certain frequencies for more than a few flashes. People are even more sensitive to red flashing than to other colors, so a special test is provided for saturated red flashing. These guidelines are based on guidelines for the broadcasting industry as adapted for computer screens, where content is viewed from a closer distance (using a larger angle of vision).

Flashing can be caused by the display, the computer rendering the image or by the content being rendered. The author has no control of the first two. They can be addressed by the design and speed of the display and computer. The intent of this criterion is to ensure that flicker that violates the flash thresholds is not caused by the content itself. For example, the content could contain a video clip or animated image of a series of strobe flashes, or close-ups of rapid-fire explosions.

### Specific Benefits of Success Criterion 2.3.1:

Individuals who have seizures when viewing flashing material will be able to view all of the material on a site without having a seizure and without having to miss the full experience of the content by being limited to text alternatives. This includes people with photosensitive epilepsy as well as other photosensitive seizure disorders.

[Learn more about 2.3.1 Three Flashes or Below Threshold](#)

## Success Criterion

All items below must be true to meet the success criterion for 2.3.1

- (A) Check that there are no more than three flashes during any 1-second period. If there are three flashes, check that the Light/Dark status at the end of the 1-second period is the same as at the start.**

-OR-

- (B) Check that only one area of the screen is flashing at any time. Check that the flashing content would fit into a contiguous container whose area is less than the small safe area (explanation provided below).**

-AND-

- (C) Ensure the red flash threshold was not exceeded. A red flash is defined as any pair of opposing changes involving a saturated red.**

For A:

Ref #	Description	Example
<a href="#">G19</a>	The objective of this technique is to avoid flashing at rates that are known to cause seizures if the flashes are bright and large enough. Since some users may be using screen enlargers, this technique limits the flashing of any size content to no more than three flashes in any 1-second period.	<a href="#">Example</a>

Related techniques that will help resolve this accessibility issue:

- [Using a tool to ensure that content does not violate the general flash threshold or red flash threshold](#)

For B:

Ref #	Description	Example
<a href="#">G176</a>	<p>If you have something that flashes more than 3 times in a one second period (so G19 can't be used), but the area that is flashing is less than 25% of 10 degrees of visual field (which represents the central area of vision in the eye), then it would automatically pass.</p> <p>The 10 degree of visual field represents the central area of vision in the eye. This area is highly packed with visual sensors. Flashes in this area are transmitted to the visual cortex. For those with photosensitivity, this flashing of activity on the visual cortex can cause seizures. Flashing on</p>	<a href="#">Example</a>

<p>other areas of the eye (which have far fewer sensors) has much less effect on the cortex. Hence, the focus on just the 10 degrees of central vision.</p> <p>At this point in time, the most prevalent display is 1024 x 768 and about 15-17 inches diagonally. When viewed at a typical viewing distance (22-26 inches) a 10 degree visual field will capture an area approximately 341 x 256 pixels. This is not circular, but neither is the central vision of most users, and the difference is so small (and at the edge of the central vision where sensors are fewer) that it is not important.</p> <p>Since the criterion is 25% of any 10 degree visual field, any single flashing event on a screen (there is no other flashing on screen) that is smaller than a contiguous area of 21,824 sq pixels (any shape), would pass the General and Red Flash Thresholds.</p>	
---	--

#### Resources:

- [Harding FPA Web Site](#)
- [Trace Center Photosensitive Epilepsy Analysis Tool \(PEAT\)](#)

#### Related techniques that will help resolve this accessibility issue:

- [Using a tool to ensure that content does not violate the general flash threshold or red flash threshold](#)

## 3.2.1 On Focus

### Overview

When any component receives focus, it does not initiate a change of context. The intent of this Success Criterion is to ensure that functionality is predictable as visitors navigate their way through a document. Any component that is able to trigger an event when it receives focus must not change the context. Examples of changing context when a component receives focus include, but are not limited to:

- forms submitted automatically when a component receives focus;
- new windows launched when a component receives focus;
- focus is changed to another component when that component receives focus;

### Specific Benefits of Success Criterion 3.2.1:

This Success Criterion helps people with visual disabilities, cognitive limitations, and motor impairments by reducing the chance that a change of context will occur unexpectedly.

[Learn more about 3.2.1 On Focus](#)

### Success Criterion

All items below must be true to meet the success criterion for 3.2.1

- Load the web page, ensure that a new window has not been opened as a result of loading the new page. Ensure that when a new window is opened it is not automatically given focus.**

Ref #	Description	Example
<a href="#">F52</a>	Some Web sites open a new window when a page is loaded, to advertise a product or service. The objective of this technique is to ensure that pages do not disorient users by opening up one or more new windows that automatically attain focus as soon as a page is loaded.	<a href="#">Example</a>

Related techniques that will help resolve this accessibility issue:

- [Using “activate” rather than “focus” as a trigger for changes of context](#)

- For each link that opens automatically in a new window or tab when a change of context is initiated by a user request, check that there is a warning spoken in assistive technology that this link opens to a new window. Also, check that there is a visual warning in text that this link opens to a new window.**

Ref #	Description	Example
<a href="#">G201</a>	The objective of this technique is to provide a warning before automatically opening a new window or tab. Opening new windows automatically when a link is activated can be disorienting for people who have difficulty perceiving visual content, and for some people with cognitive disabilities, if they are not warned in advance. Providing a warning allows the user to decide if they want to leave the current window, and the warning will help them find their way back, if they do decide they would like to go to the new window. It will help them understand that the "back" button will not work and that they have to return to the last window they had open, in order to find their previous location.	<a href="#">Example</a>

Related techniques that will help resolve this accessibility issue:

- [Using the target attribute to open a new window on user request and indicating this in link text](#)
- [Using progressive enhancement to open new windows on user request](#)
- [Opening new windows and tabs from a link only when necessary](#)



- **Use the keyboard to verify that you can get to all interactive elements using the keyboard. Check that when focus is placed on each element, focus remains there until user moves it.**

Ref #	Description	Example
<a href="#">F55</a>	Content that normally receives focus when the content is accessed by keyboard may have this focus removed by scripting. This is sometimes done when designer considers the system focus indicator to be unsightly. However, the system focus indicator is an important part of accessibility for keyboard users. In addition, this practice removes focus from the content entirely, which means that the content can only be operated by a pointing device such as a mouse.	<a href="#">Example</a>

### 3.2.3 Consistent Navigation

#### Overview

Navigational mechanisms that are repeated on multiple Web pages within a set of Web pages occur in the same relative order each time they are repeated, unless a change is initiated by the user. The intent of this Success Criterion is to encourage the use of consistent presentation and layout for users who interact with repeated content within a set of Web pages and need to locate specific information or functionality more than once. Individuals with low vision who use screen magnification to display a small portion of the screen at a time often use visual cues and page boundaries to quickly locate repeated content. Presenting repeated content in the same order is also important for visual users who use spatial memory or visual cues within the design to locate repeated content.

#### Specific Benefits of Success Criterion 3.2.3:

Ensuring that repeated components occur in the same order on each page of a site helps users become comfortable that they will be able to predict where they can find things on each page. This helps users with cognitive limitations, users with low vision, users with intellectual disabilities, and also those who are blind.

[Learn more about 3.2.3 Consistent Navigation](#)

#### Success Criterion

All items below must be true to meet the success criterion for 3.2.3

- Check to see if a navigation mechanism is being used on more than one Web page.**
- Check the default presentation of the navigation mechanism on each page to see if the list of links are in the same relative order on each Web page.**

Ref #	Description	Example
<a href="#">F66</a>	This describes a failure condition for all techniques involving navigation mechanisms that are repeated on multiple Web pages within a set of Web pages (Success Criterion 3.2.3). If the mechanism presents the order of links in a different order on two or more pages, then the failure is triggered.	<a href="#">Example</a>

- List components that are repeated on each Web page in a set of Web pages (for example, on each page in a Web site). For each component, check that it appears in the same relative order with regard to other repeated components on each Web page where it appears.**

Ref #	Description	Example
<a href="#">G61</a>	The objective of this technique is to make content easier to use by making the placement of repeated components more predictable. This technique helps maintain consistent layout or presentation between Web pages by presenting components that are repeated in these Web units in the same relative order each time they appear. Other components can be inserted between them, but their relative order is not changed.	<a href="#">Example</a>

## 3.2.4 Consistent Identification

### Overview

Components that have the same functionality within a set of Web pages are identified consistently. The intent of this Success Criterion is to ensure consistent identification of functional components that appear repeatedly within a set of Web pages. A strategy that people who use screen readers use when operating a Web site is to rely heavily on their familiarity with functions that may appear on different Web pages. If identical functions have different labels on different Web pages, the site will be considerably more difficult to use. It may also be confusing and increase the cognitive load for people with cognitive limitations. Therefore, consistent labeling will help.

### Specific Benefits of Success Criterion 3.2.4:

- People who learn functionality on one page on a site can find the desired functions on other pages if they are present.
- When non-text content is used in a consistent way to identify components with the same functionality, people with difficulty reading text or detecting text alternatives can interact with the Web without depending on text alternatives.
- People who depend on text alternatives can have a more predictable experience. They can also search for the component if it has a consistent label on different pages.

[Learn more about 3.2.4 Consistent Identification](#)

### Success Criterion

All items below must be true to meet the success criterion for 3.2.4

- In a set of Web pages, find components with the same function that are repeated in multiple Web pages. For each component with the same function found, check that the naming is consistent.**

Ref #	Description	Example
<a href="#">F31</a>	Components that have the same function in different Web pages are more easily recognized if they are labeled consistently. If the naming is not consistent, some users may get confused.	<a href="#">Example</a>

## 3.3.4 Error Prevention (Legal, Financial, Data)

### Overview

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

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

### Specific Benefits of Success Criterion 3.3.4:

Providing safeguards to avoid serious consequences resulting from mistakes helps users with all disabilities who may be more likely to make mistakes.

[Learn more about 3.3.4 Error Prevention \(Legal, Financial, Data\)](#)

### Success Criterion

All items below must be true to meet the success criterion for 3.3.4

Situation A: If an application causes a legal transaction to occur, such as making a purchase or submitting an income tax return:

- Check that the Web page describes the time period to cancel or change an order. Additionally, check that the Web page describes the process for canceling or changing an order.**

Ref #	Description	Example
<a href="#">G164</a>	The objective of this technique is to allow users to recover from errors made when placing an order by providing them with a period of time during which they can cancel or change the order. In general, a contract or an order is a legal commitment and cannot be canceled. However, a Web site may choose to offer this capability, and it provides a way for users to recover from errors.	<a href="#">Example</a>

- Check that the user is prompted to review and confirm data. If user data are collected in multiple steps, the user is allowed to return to previous steps to review and change data.**

-OR-

- Determine if a summary of all data input by the user is provided before the transaction is committed and a method is provided to correct errors if necessary.**

Ref #	Description	Example
<a href="#">G98</a>	The objective of this technique is to provide users with a way to ensure their input is correct before completing an irreversible transaction. Testing, financial, and legal applications permit transactions to occur which cannot be "undone". It is therefore important that there be no errors in the data submission, as the user will not have the opportunity to correct the error once the transaction has been committed.	<a href="#">Example</a>

- For user input pages that cause irreversible transactions to occur: check that a checkbox indicating user confirmation of the input or action is provided in addition to the submit button.**

- Check that if the checkbox is not selected when the form is submitted, the input is rejected and the user is prompted to review their entry, select the checkbox, and resubmit.**

Ref #	Description	Example
<a href="#">G155</a>	The objective of this technique is to provide a checkbox that users must select to indicate they have reviewed their input and are ready for it to be committed. This is important when the nature of the transaction is such that it may not be reversible if input errors are subsequently discovered or when the result of an action is that data is deleted.	<a href="#">Example</a>

Situation B: If an action causes information to be deleted:

- Identify functionality that allows deleting content. Delete content and attempt to recover it. Check if deleted information can be recovered.**

Ref #	Description	Example
<a href="#">G99</a>	When a Web application provides the capability of deleting information, the server can provide a means to recover information that was deleted in error by a user. One approach is to delay deleting the data by merely marking it for deletion or moving it to a holding area (such as a trash can) and waiting some period of time before actually deleting it. During this time period, the user can request that the data be restored or can retrieve it from the holding area. Another approach is to record all delete transactions in such a way that data can be restored if requested by the user, such as in the edit history stored by wikis and source control applications. The retrievable information that is stored should be that which would be needed to correct the transaction.	<a href="#">Example</a>

- Initiate the action that cannot be reversed or changed. Check that a request to confirm the selected action is presented. Check that the action can be confirmed and canceled.**

Ref #	Description	Example
<a href="#">G168</a>	This technique is to seek confirmation from the user that the selected action is his or her intended action. Use this technique in situations where the action cannot be undone after it has been followed through. This will help users avoid submitting a form or deleting data by mistake.	<a href="#">Example</a>

- For user input pages that cause irreversible transactions to occur: check that a checkbox indicating user confirmation of the input or action is provided in addition to the submit button.**

- Check that if the checkbox is not selected when the form is submitted, the input is rejected and the user is prompted to review their entry, select the checkbox, and resubmit.**

Ref #	Description	Example
<a href="#">G155</a>	The objective of this technique is to provide a checkbox that users must select to indicate they have reviewed their input and are ready for it to be committed. This is important when the nature of the transaction is such that it may not be reversible if input errors are subsequently discovered or when the result of an action is that data is deleted.	<a href="#">Example</a>

Situation C: If the Web page includes a testing application:

- Check that the user is prompted to review and confirm data. If user data are collected in multiple steps, the user is allowed to return to previous steps to review and change data.**

-OR-

- Determine if a summary of all data input by the user is provided before the transaction is committed and a method is provided to correct errors if necessary.**

Ref #	Description	Example
<a href="#">G98</a>	The objective of this technique is to provide users with a way to ensure their input is correct before completing an irreversible transaction. Testing, financial, and legal applications permit transactions to occur which cannot be "undone". It is therefore important that there be no errors in the data submission, as the user will not have the opportunity to correct the error once the transaction has been committed.	<a href="#">Example</a>

- Initiate the action that cannot be reversed or changed. Check that a request to confirm the selected action is presented. Check that the action can be confirmed and canceled.**

Ref #	Description	Example
<a href="#">G168</a>	This technique is to seek confirmation from the user that the selected action is his or her intended action. Use this technique in situations where the action cannot be undone after it has been followed through. This will help users avoid submitting a form or deleting data by mistake.	<a href="#">Example</a>