Mobile Site Accessibility Testing Methodology – Test Cases

# Table of Contents

[List of figures 9](#_Toc34063814)

[2. Alternatives 14](#_Toc34063815)

[2.1: Motion, interaction and gesture 14](#_Toc34063816)

[About this requirement 14](#_Toc34063817)

[How to test 14](#_Toc34063818)

[Examples 14](#_Toc34063819)

[2.2: Touch gestures 15](#_Toc34063820)

[About this requirement 15](#_Toc34063821)

[How to test 16](#_Toc34063822)

[Examples 16](#_Toc34063823)

[2.3: Geolocation 20](#_Toc34063824)

[About this requirement 20](#_Toc34063825)

[How to test 20](#_Toc34063826)

[Examples 20](#_Toc34063827)

[2.4: Change of state 23](#_Toc34063828)

[About this requirement 23](#_Toc34063829)

[How to test 23](#_Toc34063830)

[Examples 23](#_Toc34063831)

[2.5: Audio cues 25](#_Toc34063832)

[About this requirement 25](#_Toc34063833)

[How to test 25](#_Toc34063834)

[Examples 26](#_Toc34063835)

[2.6: Status messages 27](#_Toc34063836)

[About this requirement 27](#_Toc34063837)

[How to test 27](#_Toc34063838)

[Examples 27](#_Toc34063839)

[2.7: Abbreviations 28](#_Toc34063840)

[About this requirement 28](#_Toc34063841)

[How to test 28](#_Toc34063842)

[Examples 28](#_Toc34063843)

[2.8: Summary of content 30](#_Toc34063844)

[About this requirement 30](#_Toc34063845)

[How to test 30](#_Toc34063846)

[Examples 30](#_Toc34063847)

[2.9: Ambiguous text 32](#_Toc34063848)

[About this requirement 32](#_Toc34063849)

[How to test 33](#_Toc34063850)

[Examples 33](#_Toc34063851)

[3. Display 35](#_Toc34063852)

[3.1: Three flashes 35](#_Toc34063853)

[About this requirement 35](#_Toc34063854)

[How to test 35](#_Toc34063855)

[Examples 35](#_Toc34063856)

[3.2: Change on request 36](#_Toc34063857)

[About this requirement 36](#_Toc34063858)

[How to test 36](#_Toc34063859)

[Examples 36](#_Toc34063860)

[3.3: Target size 38](#_Toc34063861)

[About this requirement 38](#_Toc34063862)

[How to test 38](#_Toc34063863)

[Examples 38](#_Toc34063864)

[3.4: Inactive space 40](#_Toc34063865)

[About this requirement 40](#_Toc34063866)

[How to test 40](#_Toc34063867)

[Examples 40](#_Toc34063868)

[3.5: Fixed size containers 41](#_Toc34063869)

[About this requirement 41](#_Toc34063870)

[How to test 41](#_Toc34063871)

[Examples 41](#_Toc34063872)

[3.6: Justified text 42](#_Toc34063873)

[About this requirement 42](#_Toc34063874)

[How to test 42](#_Toc34063875)

[Examples 42](#_Toc34063876)

[3.7: Color contrast 44](#_Toc34063877)

[About this requirement 44](#_Toc34063878)

[How to test 44](#_Toc34063879)

[Examples 44](#_Toc34063880)

[3.8: Orientation 46](#_Toc34063881)

[About this requirement 46](#_Toc34063882)

[How to test 46](#_Toc34063883)

[Examples 47](#_Toc34063884)

[3.9: Animation 49](#_Toc34063885)

[About this requirement 49](#_Toc34063886)

[How to test 49](#_Toc34063887)

[Examples 49](#_Toc34063888)

[3.10: Pinch zoom 50](#_Toc34063889)

[About this requirement 50](#_Toc34063890)

[How to test – pinch zoom 50](#_Toc34063891)

[How to test – font resizing feature 50](#_Toc34063892)

[Examples 50](#_Toc34063893)

[3.11: Reflow 52](#_Toc34063894)

[About this requirement 52](#_Toc34063895)

[How to test 52](#_Toc34063896)

[Examples 52](#_Toc34063897)

[4. Actionable items 56](#_Toc34063898)

[4.1: Content on hover, focus or input 56](#_Toc34063899)

[About this requirement 56](#_Toc34063900)

[How to test 56](#_Toc34063901)

[Examples 56](#_Toc34063902)

[4.2: Native UI 58](#_Toc34063903)

[About this requirement 58](#_Toc34063904)

[How to test 58](#_Toc34063905)

[Examples 58](#_Toc34063906)

[4.3: Descriptive text links 60](#_Toc34063907)

[About this requirement 60](#_Toc34063908)

[How to test 60](#_Toc34063909)

[Examples 60](#_Toc34063910)

[4.4: Non-keyboard options 61](#_Toc34063911)

[About this requirement 61](#_Toc34063912)

[How to test 61](#_Toc34063913)

[Examples 61](#_Toc34063914)

[4.5: Infinite scrolling 64](#_Toc34063915)

[About this requirement 64](#_Toc34063916)

[How to test 64](#_Toc34063917)

[Examples 64](#_Toc34063918)

[4.6: Color alone 66](#_Toc34063919)

[About this requirement 66](#_Toc34063920)

[How to test 66](#_Toc34063921)

[Examples 66](#_Toc34063922)

[4.7: Removal of touch 68](#_Toc34063923)

[About this requirement 68](#_Toc34063924)

[How to test 68](#_Toc34063925)

[Examples 68](#_Toc34063926)

[5. Navigational aids 70](#_Toc34063927)

[5.1: Visual indicators 70](#_Toc34063928)

[About this requirement 70](#_Toc34063929)

[How to test 70](#_Toc34063930)

[Examples 70](#_Toc34063931)

[5.2: Character key shortcuts 75](#_Toc34063932)

[About this requirement 75](#_Toc34063933)

[How to test – turn off single character key shortcuts 75](#_Toc34063934)

[How to test – modify single character key shortcuts 75](#_Toc34063935)

[How to test – single character key shortcuts available only on focus 75](#_Toc34063936)

[Examples 75](#_Toc34063937)

[5.3: Descriptive headings 76](#_Toc34063938)

[About this requirement 76](#_Toc34063939)

[How to test 76](#_Toc34063940)

[Examples 76](#_Toc34063941)

[5.4: Inactivity timeout 77](#_Toc34063942)

[About this requirement 77](#_Toc34063943)

[How to test 77](#_Toc34063944)

[Examples 77](#_Toc34063945)

[5.5: Navigation features 78](#_Toc34063946)

[About this requirement 78](#_Toc34063947)

[How to test 78](#_Toc34063948)

[Examples 78](#_Toc34063949)

[5.6: ARIA 81](#_Toc34063950)

[About this requirement 81](#_Toc34063951)

[How to test 81](#_Toc34063952)

[Examples 81](#_Toc34063953)

[6. Audio and video 82](#_Toc34063954)

[6.1: Transcript 82](#_Toc34063955)

[About this requirement 82](#_Toc34063956)

[How to test 82](#_Toc34063957)

[Examples 82](#_Toc34063958)

[6.2: Captions 84](#_Toc34063959)

[About this requirement 84](#_Toc34063960)

[How to test 84](#_Toc34063961)

[Examples 84](#_Toc34063962)

[6.3: Live audio and video 85](#_Toc34063963)

[About this requirement 85](#_Toc34063964)

[How to test 85](#_Toc34063965)

[Examples 85](#_Toc34063966)

[7. Forms 86](#_Toc34063967)

[7.1: CAPTCHAs 86](#_Toc34063968)

[About this requirement 86](#_Toc34063969)

[How to test 86](#_Toc34063970)

[Examples 86](#_Toc34063971)

[7.2: Context-sensitive help 87](#_Toc34063972)

[About this requirement 87](#_Toc34063973)

[How to test 87](#_Toc34063974)

[Examples 87](#_Toc34063975)

[7.3: Error prevention 89](#_Toc34063976)

[About this requirement 89](#_Toc34063977)

[How to test 89](#_Toc34063978)

[Examples 89](#_Toc34063979)

[7.4: Positioned field labels 91](#_Toc34063980)

[About this requirement 91](#_Toc34063981)

[How to test 91](#_Toc34063982)

[Examples 91](#_Toc34063983)

[7.5: Visible field labels 92](#_Toc34063984)

[About this requirement 92](#_Toc34063985)

[How to test 92](#_Toc34063986)

[Examples 92](#_Toc34063987)

[7.6: Accessible name 94](#_Toc34063988)

[About this requirement 94](#_Toc34063989)

[How to test 94](#_Toc34063990)

[Examples 94](#_Toc34063991)

[7.7: Form and keyboard interaction 95](#_Toc34063992)

[About this requirement 95](#_Toc34063993)

[How to test 95](#_Toc34063994)

[Examples 95](#_Toc34063995)

[7.8: HTML5 97](#_Toc34063996)

[About this requirement 97](#_Toc34063997)

[How to test – HTML5 INPUT TYPES 97](#_Toc34063998)

[How to test – AUTOCOMPLETE 97](#_Toc34063999)

[Examples 97](#_Toc34064000)

[8. Mobile and desktop relationship 99](#_Toc34064001)

[8.1: Consistency 99](#_Toc34064002)

[About this requirement 99](#_Toc34064003)

[How to test – identifying multiple sites 99](#_Toc34064004)

[How to test 99](#_Toc34064005)

[Examples 99](#_Toc34064006)

[8.2: Linking between types of a site 101](#_Toc34064007)

[About this requirement 101](#_Toc34064008)

[How to test – identifying multiple sites 101](#_Toc34064009)

[How to test – mobile device 101](#_Toc34064010)

[How to test – desktop 101](#_Toc34064011)

[Examples 101](#_Toc34064012)

[Acknowledgements 104](#_Toc34064013)

[Relationship to existing Accessibility testing standards 104](#_Toc34064014)

[ICT Accessibility Testing Symposium Mobile Site Sub-Committee 104](#_Toc34064015)

[Contacts 104](#_Toc34064016)

[Gian Wild 104](#_Toc34064017)

[Peter McNally 104](#_Toc34064018)

# List of figures

[Figure 1: Alternative is provided on another page 14](#_Toc34064495)

[Figure 2: Alternative is provided on another page (homepage) 14](#_Toc34064496)

[Figure 7: Alternative is provided on another page (activated link) 14](#_Toc34064497)

[Figure 8: Tap alternative to drag gesture 15](#_Toc34064498)

[Figure 9: Link alternative to swipe function 16](#_Toc34064499)

[Figure 10: Link alternative to swipe function 16](#_Toc34064500)

[Figure 12: Block location and site still operates (pop-up) 19](#_Toc34064501)

[Figure 13: Block location and site still operates (after blocking) 19](#_Toc34064502)

[Figure 14: Block location and site still operates (pop-up) 19](#_Toc34064503)

[Figure 15: Block location and site still operates (after blocking) 19](#_Toc34064504)

[Figure 16: Can choose a location 20](#_Toc34064505)

[Figure 18: Slideshow (playing) 21](#_Toc34064506)

[Figure 19: Slideshow (paused) 21](#_Toc34064507)

[Figure 20: Audio and visual cue 24](#_Toc34064508)

[Figure 23: Abbreviations expanded on a glossary page 26](#_Toc34064509)

[Figure 24: Abbreviations expanded in-text 26](#_Toc34064510)

[Figure 25: Exclusion: heading (EPA) 27](#_Toc34064511)

[Figure 28: Summary on another page (post) 28](#_Toc34064512)

[Figure 29: Summary on another page (summary) 28](#_Toc34064513)

[Figure 31: Heteronym (Lead) 31](#_Toc34064514)

[Figure 32: No flashing content 33](#_Toc34064515)

[Figure 35: Form submission on activating button 34](#_Toc34064516)

[Figure 41: Touch target size is sufficient 36](#_Toc34064517)

[Figure 43: Sufficient inactive space 38](#_Toc34064518)

[Figure 46: Text is not justified 40](#_Toc34064519)

[Figure 47: Text is not justified 40](#_Toc34064520)

[Figure 53: Inactive actionable items meet color contrast requirements 42](#_Toc34064521)

[Figure 56: Site works in both orientations (landscape mode) 45](#_Toc34064522)

[Figure 57: Site works in both orientations (portrait mode) 45](#_Toc34064523)

[Figure 58: Video does not force a swap of orientation 46](#_Toc34064524)

[Figure 61: Parallax scrolling not used 47](#_Toc34064525)

[Figure 62: Parallax scrolling not used (scrolled) 47](#_Toc34064526)

[Figure 68: Pinch zoom operable (not zoomed in) 48](#_Toc34064527)

[Figure 69: Pinch zoom operable (zoomed in) 48](#_Toc34064528)

[Figure 70: 3.11: Reflow exclusion 50](#_Toc34064529)

[Figure 80: Image reflow 51](#_Toc34064530)

[Figure 81: Video reflow 51](#_Toc34064531)

[Figure 82: Table reflow 52](#_Toc34064532)

[Figure 83: Table reflow 52](#_Toc34064533)

[Figure 84: Table reflow 53](#_Toc34064534)

[Figure 85: Hoverable and persistent (desktop) 54](#_Toc34064535)

[Figure 86: Hoverable and persistent (mobile) 54](#_Toc34064536)

[Figure 89: Native SELECT element on iOS 56](#_Toc34064537)

[Figure 90: Native SELECT element on Android 56](#_Toc34064538)

[Figure 91: Native dropdown menu 57](#_Toc34064539)

[Figure 94: Descriptive text links 58](#_Toc34064540)

[Figure 96: SELECT box used instead of text input field 59](#_Toc34064541)

[Figure 97: Options provided on entering a few characters 60](#_Toc34064542)

[Figure 98: Options provided on entering a few characters 61](#_Toc34064543)

[Figure 100: No infinite scrolling 62](#_Toc34064544)

[Figure 103: Links are bolded 64](#_Toc34064545)

[Figure 110: Action on up event (original page) 66](#_Toc34064546)

[Figure 111: Action on up event (upon up event) 66](#_Toc34064547)

[Figure 113: Right to left swipe 68](#_Toc34064548)

[Figure 114: Indicating additional information on touch (indicator) 69](#_Toc34064549)

[Figure 115: Indicating additional information on touch (pop-up) 69](#_Toc34064550)

[Figure 116: indicating additional information on swipe 69](#_Toc34064551)

[Figure 117: Indicating swipe from right to left to show more content 70](#_Toc34064552)

[Figure 118: Indicating swiping left and right 71](#_Toc34064553)

[Figure 119: Indicating link to expand contents (links collapsed) 71](#_Toc34064554)

[Figure 120: Indicating link to expand contents (link expanded) 71](#_Toc34064555)

[Figure 121: Arrow and dots indicating swipe area (on load) 72](#_Toc34064556)

[Figure 122: Arrow and dots indicating swipe area (on swipe) 72](#_Toc34064557)

[Figure 126: Descriptive headings 74](#_Toc34064558)

[Figure 131: Back button 76](#_Toc34064559)

[Figure 132: Multiple methods of navigation 77](#_Toc34064560)

[Figure 133: Multiple methods of navigation 78](#_Toc34064561)

[Figure 135: Transcript provided 80](#_Toc34064562)

[Figure 136: Transcript provided 81](#_Toc34064563)

[Figure 137: Closed captions provided 82](#_Toc34064564)

[Figure 140: No CAPTCHA 84](#_Toc34064565)

[Figure 144: Context-sensitive help provided next to relevant field 85](#_Toc34064566)

[Figure 145: Context-sensitive help provided at the beginning of the form 86](#_Toc34064567)

[Figure 146: Checked 87](#_Toc34064568)

[Figure 147: Checked (form filled out incorrectly) 88](#_Toc34064569)

[Figure 148: Checked (on submission of the form) 88](#_Toc34064570)

[Figure 150: Field labels are positioned with their fields 89](#_Toc34064571)

[Figure 154: Text field labels 90](#_Toc34064572)

[Figure 157: Visible text field label and accessible name match 92](#_Toc34064573)

[Figure 162: Form fields accessible by the keyboard 93](#_Toc34064574)

[Figure 164: Correct HTML5 INPUT TYPE 95](#_Toc34064575)

[Figure 165: Correct HTML5 INPUT TYPE 96](#_Toc34064576)

[Figure 168: Mobile and desktop are consistent (mobile homepage) 97](#_Toc34064577)

[Figure 169: Mobile and desktop are consistent (desktop homepage) 97](#_Toc34064578)

[Figure 170: Mobile and desktop are consistent (mobile subpage) 98](#_Toc34064579)

[Figure 171: Mobile and desktop are consistent (desktop subpage) 98](#_Toc34064580)

[Figure 174: Can move between sites on both mobile and desktop (mobile site) 99](#_Toc34064581)

[Figure 175: Can move between sites on both mobile and desktop (desktop site) 99](#_Toc34064582)

[Figure 176: Can move between sites on both mobile and desktop (mobile site) 100](#_Toc34064583)

[Figure 177: Can move between sites on both mobile and desktop (desktop site) 100](#_Toc34064584)

# 2. Alternatives

## 2.1: Motion, interaction and gesture

Functionality that can be operated by device motion or user motion, interaction and/or gesture can also be operated by user interface components, and responding to the motion, interaction and/or gesture can be disabled to prevent accidental actuation, except for certain situations (for more information see [SC 2.5.4: Motion Actuation](https://www.w3.org/WAI/WCAG21/Understanding/motion-actuation.html)).

### About this requirement

Please note that this requirement differs from the Touch Gestures requirement as this requires a non-touch interaction or user motion (for example, recording a specific user motion by the camera, selecting the home button, using a fingerprint, etc.).

We are also aware that a number of these motions, interactions and gestures are available via iOS Assistive Touch and will be reviewing this further in the coming year.

#### Exclusions

This requirement does not need to be met if the user motion, interaction or gesture is integral to the site, for example, some games require specific user interaction to operate (swinging the phone to operate a light sabre app, for example).

### How to test

1. Identify requirements in the site that require either:
   1. Device motion (for example, moving the device around in a circle to determine which way the device is pointing)
   2. User motion (for example, walking a set distance)
   3. User interaction (for example, squeezing the sides of the phone)
   4. User gesture (for example, capturing a specific user gesture, such as a thumbs up sign, by the camera)
2. Ensure either that:
   1. The device motion, user motion, user interaction or user gesture is not required; or
   2. There is an alternative accessible user interactive component that triggers functionality accessed by the device motion, user motion, user interaction or user gesture.
3. Ensure that the device motion, user motion, user interaction or user gesture can be disabled to prevent accidental actuation.

### Examples

There are currently no example passes documented for this requirement. Similar Pass examples are available in the Native App Test Cases document.

## 2.2: Touch gestures

Any touch gesture must have an alternative, accessible, actionable item (for more information see [SC 2.5.1: Pointer Gestures](https://www.w3.org/WAI/WCAG21/Understanding/pointer-gestures.html)).

### About this requirement

Touch gestures tend to fall into the following categories:

* Swiping up and down or left and right
* Dragging up and down or left and right
* Double-tapping
* Tap and hold
* Tap and swipe
* Two pinch zoom
* Press and long hold

Alternative accessible actionable items must meet WCAG2 and the following requirements:

* Change of State
* Touch Targets
* Inactive Space
* Color Contrast
* Native UI
* Removal of Touch

In this requirement, sufficient alternative actionable items to a touch gesture could include:

* A link
* A button
* A dropdown
* A separate page with the same functionality (in this case this would need to be indicated to the user)

This requirement is particularly important for screen reader users. For example, if you require your user to swipe right to complete a purchase, when the screen reader is on, the swipe right gesture moves you to the next focusable item and doesn’t complete the purchase. You must be able to perform the same action, by using a link, an up or down swipe, or some other gesture.

Please note that this requirement is similar to the Exit Trap requirement. A failure of the Exit Trap requirement is that a user cannot escape from content or a page. A failure of the Touch Gestures requirement is that the user cannot choose content or a page (i.e. they are not trapped).

#### Exclusions

Please note that standard touch gestures such as tapping to activate actionable items and swiping up and down to scroll the page are not included in this requirement as they are standard user agent components and therefore have built-in alternatives.

Please note that touch gestures used to manipulate assistive technologies, such as tap and swipe to operate the VoiceOver rotor, are not included in this requirement.

### How to test

1. Identify any site controls. If they require any of the following gestures, is there an accessible actionable item provided as an alternative?
   1. Swiping up and down or left and right
   2. Dragging up and down or left and right
   3. Double-tapping
   4. Tap and hold
   5. Tap and swipe
   6. Two pinch zoom
   7. Press and long hold

### Examples

#### Pass 1 – Alternative is provided on another page

Two rows of content under the “Top Stories” heading indicate that more content is visible on swipe as part of the content is cut off. There is a link labelled “See more” and the same content is shown when this link is activated as the additional content on swipe.

|  |  |
| --- | --- |
| Figure 1: Alternative is provided on another page  A weather and news mobile site. |  |

#### Pass 2 – Alternative is provided on another page

In this example you can swipe from right to left to show additional furniture, however you can also activate the “See All 4” link to show all items on a single page.

|  |  |
| --- | --- |
| Homepage | On selecting the “See all 4” link |
| Figure 2: Alternative is provided on another page (homepage)  A furniture shopping mobile site displaying a series of chairs in a horizontal list. | Figure 7: Alternative is provided on another page (activated link)  A furniture shopping mobile site displaying a series of chairs in a vertical list. |

#### Pass 3 – Tap alternative to drag gesture

When viewing the weather on Google you can select and drag the slider to determine the weather at certain times during the day. You can also tap on one of the times to move the slider to that specific time.

|  |  |
| --- | --- |
| Figure 8: Tap alternative to drag gesture  Portland, Oregon weather results on Google. |  |

#### Pass 4 – Link alternative to swipe function

In this example you can swipe from right to left or from left to right to see more podcasts. You can also tap the left arrow to mimic the right to left swipe and the right arrow to mimic the left to right swipe.

|  |  |
| --- | --- |
| Figure 9: Link alternative to swipe function  Museum mobile site showing several webcasts available in a horizontal list. |  |

#### Pass 5 – Link alternative to swipe function

In this example you can swipe from right to left to see more images. You can also tap the right-pointing arrow to mimic the right to left swipe.

|  |  |
| --- | --- |
| Figure 10: Link alternative to swipe function  Portland Audubon mobile website showing an image of members and an eagle in flight. |  |

## 2.3: Geolocation

Alternatives are provided for geolocation functionality that is mandatory (for example, requiring a specific geolocation before functionality appears), unless the geolocation is essential for legal reasons, or doing so would invalidate the activity. This applies to geolocation via GPS, VPN, user statement, IP address or other methods.

### About this requirement

Many users with disabilities also have physical limitations. They may be unable to get to a location, or they may find it very difficult to get to particular location. In some cases they may want to access information (such as maps) about a location before they enter the location in order to properly orient themselves and minimize any problems they could have. For example, a shopping center which limits access to their maps to people currently located in the shopping center, will cause an accessibility problem for someone who is vision impaired and needs to determine which entrance they should be dropped at when visiting a particular store.

There are also occasions where a person with a disability has assistance from a carer to complete some tasks. If these tasks are geolocked then a carer not physically located with the person with a disability will not be able to assist. For example, a grocery store that only allows purchases within a certain geolocked area will cause an accessibility problem for the person with a disability whose carer is outside the location area.

Examples of failures of this requirement include:

* The site not working if the user blocks the location
* The site not working if the user uses a VPN
* Different content being served at different locations
* The site not working if the user visits from a certain IP address
* The site not working if the user self-identifies a certain location

#### Exclusions

If the geolocation is essential for legal reasons then it does not need to meet this requirement. For example, streaming services often only have rights to a particular production in a particular location. For instance, in the United States, the TV show Supernatural is on Netflix. In Australia it is not on Netflix and is available only on the streaming service Stan.

If the geolocation would invalidate the activity then it does not need to meet this requirement. For example, an official race that relies on a person’s GPS to determine their location, cannot have an alternative as the person’s movement is integral to the activity.

### How to test

1. If the site asks for your location, does it still work if you block the location?
2. If you use a VPN to pretend to be in another location, does the site still work?
3. If the site requires your exact location, are you able to choose a different location?

### Examples

#### Pass 1 – Block location and site still operates

In the Cirque Du Soleil website you can block the location and the site still operates.

|  |  |
| --- | --- |
| On loading page | On blocking the location |
| Figure 12: Block location and site still operates (pop-up)  Required pop up reads "www,cirquedusoleil.com wants to use your device's location." Options are Block and Allow. | Figure 13: Block location and site still operates (after blocking)  Website homepage with no pop up. |

#### Pass 2 – Block location and site still operates

In the Woolworths website you can block the location and the site still operates.

|  |  |
| --- | --- |
| On loading the site | On blocking the location |
| Figure 14: Block location and site still operates (pop-up)  Required pop up reads "https//www.woolworths.com.au" would like to use your current location." Options are Don't Allow and OK. | Figure 15: Block location and site still operates (after blocking)  Woolworths delivery schedule page. |

#### Pass 3 – Can choose a location

In the Bureau of Meteorology website, the user can choose a location or select the “Find me” button, which uses the GPS in the device.

|  |  |
| --- | --- |
| Figure 16: Can choose a location  Screenshot of forecast from the Australian Bureau of Meteorology mobile site. The "Change location" field reads, "Start typing, then select from list" |  |

## 2.4: Change of state

Changes of state of non-standard controls (e.g. hamburger menu, star ratings) are clearly indicated.

### About this requirement

In most cases this requirement applies to screen reader users. When a non-standard control changes state it is often visually different. For example, in some video players, when captions are on the “CC” button is green and when captions are off the “CC” button is gray. This is a violation of WCAG2. However, under WCAG2 it is possible for a non-standard control to stay the same although the state may change. For example, the icon indicating a hamburger menu could be used as a link to both open and close the hamburger menu. The visual indicator is the same and the alternative (such as “Menu”) could also be the same.

This requirement intends to ensure that changes of state of non-standard components are clearly indicated to the user, through both display and an accessible name. This requirement does not apply to native user interface components as changes of state of these components are built into the feature.

Some examples of non-standard controls are:

* Hamburger menus (the open hamburger menu icon and the close hamburger menu icon must be visually different and also have different accessible names, for example, “Open menu” and “Close menu” respectively).
* Star ratings (for example a selected and a non-selected star must be visually different and also have different accessible names, for example “Not rated”, “1 out of 5 stars”, “4 out of 5 stars”). For information on creating accessible star ratings see [WCAG2 Technique G196: “Using a text alternative on one item within a group of images that describes all items in the group.”](https://www.w3.org/TR/2016/NOTE-WCAG20-TECHS-20161007/G196)
* Play / Pause buttons on video players or slideshows (for an example of an [accessible slideshow see the AccessibilityOz website](https://www.accessibilityoz.com/). The pause and play button are visually different and have an accessible name of “Pause” and “Play” respectively. This [slideshow is available for re-use through GPLv2](https://www.accessibilityoz.com/products/accessibilityoz-slideshow/).)

### How to test

1. Review each actionable item. Does it change state?
2. If so, is the alternative for each state appropriately descriptive?
3. If so, does each state differ from one another using more than color alone?

### Examples

#### Pass 1 – Slideshow

In the AccessibilityOz website there are two non-standard controls: the navigation menu and the slideshow controls. When the slideshow is playing the control is a Pause button. When the slideshow is paused the control is a Play button.

|  |  |
| --- | --- |
| Figure 18: Slideshow (playing)  The AccessibilityOz mobile site with the slideshow pause button indicated. | Figure 19: Slideshow (paused)  The AccessibilityOz mobile site with the slideshow play button indicated. |

## 2.5: Audio cues

Audio cues have an equivalent, accessible, visual cue.

### About this requirement

Audio cues are sometimes known as non-verbal sounds, or “earcons” or auditory icons. They are distinctive noises generated by the system and usually associated with specific actions or states. Audio cues are used to give an indication for those who cannot see visually what action is taking place in the device. By their nature, audio cues are inaccessible to people who are Deaf or hard of hearing.

Examples of use of audio cues include:

* Voice interaction
  + Audio cue: Siri emits a 2-tone beep after detecting its activation phrase, to signal that it is now ‘listening’ for a command.
  + Visual cue: An animated icon, or bar, to show Siri is “listening” and ready to take a voice command. In Android, there is text that instructs the user to say something.
* Photo-taking
  + Audio cue: A site may use audio cues to assist someone taking a photo with their device. Through audio, the person frames a shot. In some cases, the site enables the mobile or tablet to say aloud the number of faces that were detected, and they can take the photo with confidence.
  + Visual cue: The camera on the mobile device usually shows what it is viewing, and sometimes a frame around the view to indicate what will be captured in the frame when the picture is taken. There are other visual controls (buttons, etc.)
* Sending an email or a text
  + Audio cue: Most sites play a sound when an email or text is sent successfully. This can usually be turned off in settings.
  + Visual cue: Some sites show the message in the conversation, include the word “Delivered” or show an icon.

### How to test

1. Open the site.
2. Navigate to the page(s) that contains the element or elements that have been designed with an **audio cue** to indicate a state or action.
3. Activate the audio cue. Look for any visual indicators that accompany the audio cue at the time it is activated. Is there something, or is **sound** the only indicator of the state, status or action to be triggered?
4. Continue testing all audio cues.
5. Turn on the **screen reader** feature. Ensure the audio cue functions when the screen reader feature is on.

### Examples

#### Pass 1 – Audio and visual cue

In this game when each block is removed there is a sound as well as a little icon that appears and disappears. However, the blocks disappear – which is an adequate accessible visual indicator.

|  |  |
| --- | --- |
| Figure 20: Audio and visual cue  Screenshot of a game with cubes stuck together. The cubes are covered in various symbols. |  |

## 2.6: Status messages

Status messages are available to all users without receiving focus (for more information see [SC 4.1.3: Status Messages](https://www.w3.org/WAI/WCAG21/Understanding/status-messages.html)).

### About this requirement

Status messages are important as they announce whether a particular action is successful or not. Often they appear at the top or the bottom of the screen and disappear quickly and without user interaction. On mobile devices status messages are mostly known by the term “notifications.” Because of these features it is almost impossible for assistive technology users to access the status message before it disappears. However, it is also important that the status message does not take focus, because this interrupts the user unexpectedly.

This requirement consists of two parts:

* The status message must be read aloud by the screen reader
* The screen reader must not move focus to the status message.

This requirement matches the WCAG2.1 Status Messages requirement. It is the opinion of this committee that the Status Messages SC focuses exclusively on-screen reader accessibility and should be broader to encompass the needs of other assistive technology users who may also have difficulty with status messages, such as magnifier users. This will be reviewed in the coming year.

It is absolutely essential when providing status messages that you use the native UI elements inherent in the device. Status messages created with native UI components have built-in accessibility features which are likely to be more accessible to users than non-standard status messages.

### How to test

1. Identify all actionable items that generate a status message.
2. Turn on the screen reader.
3. Activate the applicable actionable item.
4. Does the status message get read by the screen reader?
5. Does the focus remain on the actionable item?

### Examples

#### Pass

There are currently no example passes documented for this requirement. Similar Pass examples are available in the Native App Test Cases document.

## 2.7: Abbreviations

All abbreviations are expanded the first time they are used on the page or a glossary of abbreviations and their expansions is provided (for more information see [SC 3.1.4: Abbreviations](https://www.w3.org/TR/UNDERSTANDING-WCAG20/meaning-located.html)).

### About this requirement

Abbreviated text is often mispronounced by screen readers. In addition, abbreviated text can be unclear to some users.

#### Exclusions

Exclusions are allowed for abbreviations that have become part of the natural language, for example, the abbreviation “TV” does not need to be expanded to “television.” Exclusions are also allowed for abbreviations that are well-known in a particular industry, if the site caters exclusively to that industry. For example, an accessibility site would not need to expand the abbreviation “W3C” as it is a known abbreviation in the accessibility industry. However, if the site was about other standards bodies, the abbreviation would need to be expanded.

If the first use of an abbreviation is part of a control, field, button or other actionable item, it does not need to be expanded on first use, but must be expanded elsewhere on the page. For example, if the first use of the abbreviation “VIT” was part of a field, for example “VIT number” it would not need to be expanded in the field label, but the expansion (Victorian Institute of Teaching) would need to be provided elsewhere on the page.

Creative content is also excluded from this requirement.

### How to test

1. Identify all abbreviations.
2. Is the abbreviation expanded when it is first used on the page?   
   Or  
   Is a glossary of abbreviations provided in the site?

### Examples

#### Pass 1 – Abbreviations expanded in-text and on a glossary page

In this site there is a glossary called “Acronyms and Abbreviations.”

|  |  |
| --- | --- |
| Glossary | Page with an abbreviation: |
| Figure 23: Abbreviations expanded on a glossary page  Screenshot of the UC San Diego Faculty and Staff glossary of Acronyms and Abbreviations | Figure 24: Abbreviations expanded in-text  Screenshot of page listing project details, reading "Designated Vice Chancellor area representative approves Project Charter and provides pre-design funding to Capital Program Management (CPM)." |

#### Pass 2 – Exclusion: heading (EPA)

The abbreviation “EPA” does not need to be expanded on first use as its first use is part of a heading.

|  |  |
| --- | --- |
| Figure 25: Exclusion: heading (EPA)  Screenshot of an article with the heading "EPA warns that smoke over Melbourne will turn air quality 'hazardous'" |  |

## 2.8: Summary of content

Where the text requires reading ability more advanced than Flesch Kincaid level 8, a summary or description of content is provided (for more information see [SC 3.1.5: Reading Level](https://www.w3.org/TR/UNDERSTANDING-WCAG20/meaning-supplements.html)).

### About this requirement

Text at a high reading level can be difficult for a number of users to understand, including people with reading disabilities. Providing a summary at a lower reading level provides the information that is required and also assists users in deciding whether they should attempt to interpret the main text.

#### Exclusions

Creative content is excluded from this requirement.

### How to test

1. Identify all blocks of text.
2. Where there is a block of text is there a summary at the beginning of the page, or referenced near the beginning of the page?
3. Select the summary text and paste it into Word.
4. In Word, select the **File** menu. Select **Options**.
5. In the dialog box, select the **Proofing** option.
6. Under the section **When correcting spelling and grammar in Word**, ensure the **Check grammar with spelling** check box is selected.

### Examples

#### Pass 1 – Summary on another page

In the LF Legal website, each article has a “simplified summary.”

|  |  |
| --- | --- |
| Original post | Simplified Summary |
| Figure 28: Summary on another page (post)  LF Legal mobile site displaying a list of links to articles. | Figure 29: Summary on another page (summary)  LF Legal mobile site showing one article's Simplified Summary, reading "This is a post responding to some negative comments about an announcement by the United States Supreme Court. The Court said that a lawsuit against Domino's Pizza about web access can stay in the court system. The lawsuit is also about the Domino's mobile application. This is good news for people who care about making sure websites and apps work for people with disabilities. Some people think the Domino's website does not need to be accessible because blind people can call on the phone. Domino's will be able to argue this if the case does not settle." |

## 2.9: Ambiguous text

Controls, primary headings, links, field labels and page titles are not ambiguous when read aloud.

### About this requirement

Screen reader users hear words spoken aloud. Therefore, there are some words which will be ambiguous to them, such as write / right, aloud / allowed, etc.

There are five types of ambiguous text.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Term | Meaning | Spelling | Pronunciation | Accessibility | Example words | Example issue |
| Homonym | Different | Same ~~or Different~~ | Same | Screen reader  Cognitive | Address (location) Address (presentation)  Ruler (system of measurement) Ruler (royalty) | Amazon Address  (the address for Amazon or a presentation by Amazon?) |
| Homophone | Different | Different | Same | Screen reader | Allowed (permitted) Aloud (spoken)  Pair (couple) Pear (fruit)  Deer (animal) Dear (form of address)  Bough (tree) Bow (ribbon) | Amazon Aloud / Amazon Allowed  (Amazon podcast or Amazon only?) |
| Heteronym / Heterophone | Different | Same | Different | Screen reader  Cognitive | Minute (60 in an hour) Minute (teeny)  Bow (ribbon) Bow (curtsey) | Close - Amazon Stores  (close the Amazon stores page or Amazon stores that are located nearby?) |
| Capitonym 1 | Different | Capital vs not | Same | Screen reader  Cognitive | March (month) march (walk)  Turkey (country) turkey (animal)  Cancer (astrological sign) cancer (disease) | Turkey – Christmas  (Christmas in Turkey, or serving turkey for Christmas?) |
| Capitonym 2 | Different | Capital vs not | Different | Screen reader  Cognitive | Polish (country person) polish (the silver)  August (month) august (majestic) | Nice – Vacations  (pleasant vacations or vacations in the city Nice?) |

There are multiple lists of homonyms, homophones, heteronyms and capitonyms available on the web. Wikipedia has a list of these:

* [Homonyms](https://en.wikipedia.org/wiki/List_of_true_homonyms)
* [Homophones](https://en.wikipedia.org/wiki/Wikipedia:Lists_of_common_misspellings/Homophones)
* [Heteronyms / Heterophones](https://en.wiktionary.org/wiki/Category:English_heteronyms)
* [Capitonyms](https://en.wikipedia.org/wiki/Capitonym)

#### About homographs and heterographs

Please note there are two other terms that are used in these situations: homographs and heterographs.

Homographs have the same spelling and different meaning and the pronunciation is irrelevant. An example homograph is:

* minute (60 in an hour)
* minute (take notes)
* minute (tiny).

Therefore, homographs are always either homonyms or heteronyms.

Heterographs have different spelling, different meaning and the same pronunciation. An example heterography is:

* to (travel)
* too (also)
* two (one more than one)

Therefore, heterographs are also homophones.

#### About testing this requirement

This requirement is probably best tested by automation.

#### Exclusions

Creative content is excluded from this requirement.

### How to test

1. Search the text for use of homonyms, homophones, heteronyms / heterophones and capitonyms.
2. Where these are used is the meaning clear?

### Examples

#### Pass 1 – Heteronym (Lead)

All page titles are not ambiguous when using the term “lead” (as in a compound).

|  |  |
| --- | --- |
| Figure 31: Heteronym (Lead)  List of links to pages: "Facts About Lead," "Air Quality - Lead," "Lead (Pb) - Chemical properties, Health and Environmental effects" and "Lead poisoning and health" |  |

# Display

## 3.1: Three flashes

Web pages do not contain more than three flashes in a one-second period (for more information see [SC 2.3.2: Three Flashes](https://www.w3.org/TR/UNDERSTANDING-WCAG20/seizure-three-times.html)).

### About this requirement

This is exactly the same requirement as [SC 2.3.2: Three Flashes](https://www.w3.org/TR/UNDERSTANDING-WCAG20/seizure-three-times.html)

The committee believes that this requirement should be met on all mobile sites. Due to the small size of the mobile device, even a small amount of flashing can cause problems.

### How to test

1. Identify any moving or changing content.
2. Does it contain flashing or strobe-like effects?
3. If so, capture the movement using a product like [Camtasia](https://www.techsmith.com/video-editor.html).
4. Test the capture with the [Photo-sensitive Epilepsy Analysis Tool](https://trace.umd.edu/peat).

### Examples

#### Pass 1 – No flashing content

The NASA Seasons Greetings video does not contain any flashing content.

|  |  |
| --- | --- |
| Figure 32: No flashing content  Landscape screenshot of a video on YouTube. |  |

## 3.2: Change on request

Changes of context must always be user-initiated unless it is time-sensitive or an emergency (for more information see [SC 3.2.5: Change on Request](https://www.w3.org/TR/UNDERSTANDING-WCAG20/consistent-behavior-no-extreme-changes-context.html)).

### About this requirement

This requirement differs from WCAG2.1 [SC 3.2.5: Change on Request](https://www.w3.org/TR/UNDERSTANDING-WCAG20/consistent-behavior-no-extreme-changes-context.html) as it does not allow for meeting this requirement by providing a method to turn off such changes. It is the committee’s opinion that an unexpected change of context is so disrupting that it should not occur unless it is time-sensitive or an emergency.

#### Ads

Where a user must proceed through an ad to access content (for example, watching an ad before a YouTube video plays), this is not defined as a change of context.

### How to test

1. Identify all changes of context, such as:
   1. Window or popup launching
   2. Windows or popups launching on entering text in an input field
   3. Automatic submission of form on entering text or selecting an option
   4. Automatic submission of form on removing focus from a field
   5. Opening a new window without warning when a user activates a link
   6. Automatic update of content
   7. Meta refresh
2. Are all changes of context user-initiated? (unless it is time-sensitive or an emergency)

### Examples

#### Pass 1 – Form submission on activating button

The location does not change until the user activates the “Locate” button.

|  |  |
| --- | --- |
| Figure 35: Form submission on activating button  Screenshot of a weather site with a "Change location" field, activated by a "Locate" button |  |

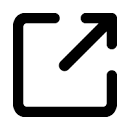
## 3.3: Target size

Touch targets should be at least 44 by 44 CSS pixels (approximately 7 to 10 millimeters). For more information see [SC 2.5.5: Target Size](https://www.w3.org/TR/WCAG21/#target-size).

### About this requirement

Most people use touch as the form of interaction on mobile and tablet devices. Touch is not as granular as mouse interaction and can depend on the size of a person’s fingers. People with certain physical disabilities may find it difficult to activate very small touch targets.

This requirement is the same as [SC 2.5.5: Target Size](https://www.w3.org/TR/WCAG21/#target-size), however the following exclusions are **not allowed:**

* **Equivalent**: the target is available through an equivalent link or control on the same page that is at least 44 by 44 CSS pixels; and
* **Inline**: the target is **not text** within a sentence or block of text  
  We allow for words not to meet Touch Target Size requirements, but non-word content must meet this requirement. For example, a link “learn more about us” **does not need** to meet the requirement, however a footnote or icon, such as “learn more about us” **does** need to meet this requirement. This is because text on mobile devices can be easily increased in size, whereas icons are more difficult to change.  
  The committee believes the term “block of text” means text within a sentence of sentence fragment. A block of text is not a set of links, such as a navigation or footer. Although this text will respond to increase text size features within the mobile device, they are often restricted to fixed size containers and may not resize appropriately.

#### Exclusions

This requirement allows for two exceptions:

* **Inline:** the target **is text** within a sentence or block of text
* **User agent control**: the size of the target is determined by the user agent and is not modified by the author; or
* **Essential**: a particular presentation of the target is essential to the information being conveyed

### How to test

1. Identify all actionable touch targets. Turn on screen reader.
2. Swipe to each actionable touch target.
3. Measure the size of the actionable touch target.

### Examples

#### Pass 1 – Touch target size is sufficient

In the Fandom website all actionable items have adequately sized touch targets.

|  |  |
| --- | --- |
| Figure 41: Touch target size is sufficient  Screengrab of Fandom mobile site |  |

## 3.4: Inactive space

Touch targets have sufficient inactive space between them (inactive space of at least 22 by 22 CSS pixels, or 4 to 5 mm).

### About this requirement

Most people use touch as the form of interaction on mobile and tablet devices. Touch is not as granular as mouse interaction and can depend on the size of a person’s fingers. People with certain physical disabilities may find it difficult to activate the correct item if multiple options are available without a minimum of inactive space.

#### Exclusions

This requirement allows for two exceptions:

* **User agent control**: the size of the target is determined by the user agent and is not modified by the author; or
* **Essential**: a particular presentation of the target is essential to the information being conveyed

### How to test

1. Identify all adjacent touch targets. Turn on screen reader.
2. Swipe to each actionable touch target.
3. Measure the distance between them.

### Examples

#### Pass 1 – Sufficient inactive space

In this example there is sufficient inactive space between consecutive links.

|  |  |
| --- | --- |
| Figure 43: Sufficient inactive space  Screengrab of mobile Wikipedia site. The article's Personal Life unexpanded dropdown link is indicated. |  |

## 3.5: Fixed size containers

Do not use fixed size containers for blocks of text unless the display is essential.

### About this requirement

In most cases where fixed containers are used, they inhibit the ability for assistive technology to repurpose the text. For example, 10pt text in a fixed size container will respond to increase text size features in the mobile device, but as the container does not increase in size when the text does, the text is likely to be cut off.

### How to test

The committee believes this is adequately tested by the Reader / Simplified view requirement. Additional testing instructions may be provided in the coming year.

### Examples

There are currently no example passes documented for this requirement. Similar Pass examples are available in this document under the requirement Reader / Simplified view on page 29.

## 3.6: Justified text

Justified text has not been used.

### About this requirement

Justified text changes the spacing between letters to ensure that blocks of text align on the right and left side of the device. As a result, text is more difficult to read. Some people with reading disabilities cannot read justified text at all due to the “rivers of white” – as they read they come to a larger area of white space than usual and this pulls their eyes to the next area of large white space.

### How to test

1. Find all blocks of text.
2. Ensure the text is not justified.

### Examples

#### Pass 1 – Text is not justified

In this site the content is left-aligned.

|  |  |
| --- | --- |
| Figure 46: Text is not justified  Screenshot of an article with an image of a bat perched on a tree |  |

#### Pass 2 – Text is not justified

In this site the content is left-aligned.

|  |  |
| --- | --- |
| Figure 47: Text is not justified  Screenshot of an article, "Wanting to live a more sustainable life is one thing, knowing where on earth to start is another" |  |

## 3.7: Color contrast

Actionable items (including non-actionable items that will become actionable), or items that convey meaning should have a minimum color contrast ratio of 4.5:1 when compared with the surrounding background. For more information see [SC 1.4.11: Non-text Contrast](https://www.w3.org/WAI/WCAG21/Understanding/non-text-contrast.html).

### About this requirement

Please note that this differs from WCAG2.1 as SC 1.4.11 allows an exception for non-actionable items that will become actionable, but this exception does not exist in this methodology. It is the opinion of the committee that actionable items that will become actionable (also referred to as “inactive items”) convey information to the user and therefore should meet color contrast requirements. This differs from WCAG2.1. It is the opinion of this committee that if an inactive item truly conveys no information then it should be removed from the screen for all users.

Please note that although an item must always contrast sufficiently with the background color, it is not necessary for states to contrast with each other. However if a change of state is indicated with color alone (i.e. the on state is black on white and the off state is green on white) then this would fail [SC 1.4.1: Use of Color](https://www.w3.org/TR/UNDERSTANDING-WCAG20/visual-audio-contrast-without-color.html). Therefore, when an item changes state it can change in color however there must be some additional indication, such as underline, bold or border (or a different icon).

#### Exclusions

The Essential exceptions detailed in [SC 1.4.11: Non-text Contrast](https://www.w3.org/WAI/WCAG21/Understanding/non-text-contrast.html) are applicable to this requirement, but please note that [SC 1.1.1: Non-text Alternatives](http://www.w3.org/TR/UNDERSTANDING-WCAG20/text-equiv-all.html) would still apply.

### How to test

1. Identify all actionable items and important content.
2. Verify that the color contrast of these items compared to the background color is a minimum of 4.5:1.
3. Activate all actionable items to see if it contains a change of state.
4. If so, ensure the change of state also meet color contrast requirements compare to the background color.

You also may find the WCAG2.1 Testing principles in the detailed in [SC 1.4.11: Non-text Contrast](https://www.w3.org/WAI/WCAG21/Understanding/non-text-contrast.html) useful in testing.

### Examples

#### Pass 1 – Inactive actionable items meet color contrast requirements

In this website the inactive options (23, 24, 25, 26, 27, 29 and 1) meet color contrast requirements and are indicated as inactive through a pattern.

|  |  |
| --- | --- |
| Figure 53: Inactive actionable items meet color contrast requirements  Screenshot of a calendar in list format. Inactive days are indicates with a gray striped pattern |  |

## 3.8: Orientation

The site can be used in portrait mode (for more information see [WCAG2.1 SC 1.3.4: Orientation](https://www.w3.org/WAI/WCAG21/Understanding/orientation.html)).

The site can be used in landscape mode (for more information see [WCAG2.1 SC 1.3.4: Orientation](https://www.w3.org/WAI/WCAG21/Understanding/orientation.html)).

The system does not swap orientation unexpectedly.

### About this requirement

Some users with disabilities are restricted to one orientation, for example, when a device is attached to a wheelchair. This orientation can be portrait or landscape. Therefore, it is important that all mobile sites operate in both portrait and landscape mode.

Users who are restricted to a particular orientation often lock their device in their respective orientation. If the orientation still changes this can render the content inaccessible.

Other users lock the orientation because moving between portrait and landscape orientation is very distracting. This could be the case when a user has a tremor and the device interprets the movement and changes orientation accordingly. If the orientation still changes this can render the content inaccessible.

Orientation can swap unexpectedly when using mobile sites that display in only one orientation, such as the Mahjong app, or change display orientation when showing particular content, such as in Netflix when starting a movie (search and other pages can display in portrait orientation).

#### Exclusions

This committee agrees with the exceptions stated in WCAG2.1:

“Examples where a particular display orientation may be essential are a bank check, a piano application, slides for a projector or television, or virtual reality content where content is not necessarily restricted to landscape or portrait display orientation.”

#### Locking orientations restrictions by devices

The following devices only allowing locking the orientation in **Portrait mode**:

* Google Pixel
* iPhone X

#### How to lock orientation by device

##### iPhone X / iPad

1. Move to the preferred orientation.
2. Swipe down from the top of the screen to show Quick Settings
3. Select the icon with the lock and the rotating circle (called “Lock rotation”)
4. Please note you can only lock the iPhone in the portrait orientation.

##### Google Pixel

1. Swipe down from the top of the screen to show Quick Settings
2. Select the icon called “Auto-rotate”
3. Please note you can only lock the Google Pixel in the portrait orientation.

### How to test

1. In one orientation, open the browser and type in the URL. Does the site honor the orientation?
2. Close the browser.
3. Switch to the other orientation.
4. Re-open the browser and type in the URL. Does the site honor the new orientation?
5. Open Quick Settings and turn off "Auto rotate screen."
6. Open the browser and see if the screen orientation has not changed.
7. If the site has a displayable content (e.g. video or text of an eBook), play the content and see if the screen orientation has not changed.
8. Open Quick Settings and turn on "Auto rotate screen."
9. Close Settings and change the orientation.
10. Open Quick Settings and turn off "Auto rotate screen."
11. Open the browser and see if the screen orientation has not changed.
12. If the site has a displayable content (e.g. video or text of an eBook), play the content and see if the screen orientation has not changed.

### Examples

#### Pass 1 – Site works in both orientations

The site works and displays correctly in both portrait and landscape orientation.

|  |  |
| --- | --- |
| Landscape mode |  |
| Figure 56: Site works in both orientations (landscape mode)  Screengrab of an Audubon site. |  |
| Portrait mode |  |
| Figure 57: Site works in both orientations (portrait mode)  Screengrab of Audubon site with content narrowed but properly displayed. |  |

#### Pass 2 – Video does not force a swap of orientation

Playing the video does **not** force the orientation to landscape.

|  |  |
| --- | --- |
| Figure 58: Video does not force a swap of orientation  Portrait screengrab of Fandom site showing a video. |  |

## 3.9: Animation

Animation triggered by interaction can be disabled (for more information see [SC 2.3.3: Animation from Interactions](https://www.w3.org/WAI/WCAG21/Understanding/animation-from-interactions.html)).

### About this requirement

Movement is problematic for a number of different users, and this is the case whether the movement is user initiated or not.

Please note that this differs from User motion, interaction and gesture as that requirement includes motion that is triggered by movement of the device or user, not normal interactions such as swiping.

### How to test

1. Open the site.
2. Scroll to the bottom of the page. If movement is triggered, other than scrolling, can it be turned off?
3. Scroll to the top of the page. If movement is triggered, other than scrolling, can it be turned off?
4. Find any actionable items that load new content.
5. Load the new content. If movement is triggered, other than scrolling, can it be turned off?

### Examples

#### Pass 1 – Parallax scrolling not used

The MoMA website does not use parallax scrolling.

|  |  |
| --- | --- |
| Figure 61: Parallax scrolling not used  Screenshot of the MoMA mobile site homepage | Figure 62: Parallax scrolling not used (scrolled)  Screenshot of the MoMA mobile site homepage scrolled down the page |

## 3.10: Pinch zoom

Pinch zoom is operable, unless an accessible font resizing feature has been included in the website that allows the user to increase the size of content at least two times the size of the standard font size. For more information see [WCAG2.1 SC 1.4.4: Resize text](https://www.w3.org/TR/WCAG21/#resize-text).

### About this requirement

The 400% increase in text size suggested by [WCAG 2.1 SC 1.4.10: Reflow](https://www.w3.org/TR/WCAG21/#reflow) is not applicable to mobile devices due to the limited screen sizes. However, it is still mandatory to be able to resize text to at least 200% original size.

### How to test – pinch zoom

1. Within a browser window, open the website.
2. Place two fingers together on the screen.
3. Keeping the fingers on the screen move fingers apart.
4. The area of the page should enlarge.
5. Place two fingers on the screen, spaced apart.
6. Keeping the fingers on the screen move fingers together.
7. The page should decrease in size, showing more of the page. If the page is at full size it will not decrease.

### How to test – font resizing feature

1. Find a page with a block of content.
2. Has a font resizing feature been provided?

### Examples

#### Pass 1 – Pinch zoom operable

In the Creation Entertainment website, the pinch zoom is functional.

|  |  |
| --- | --- |
| Figure 68: Pinch zoom operable (not zoomed in)  Screengrab of site with "Gold Weekend Admission Package" link and accompanying image displayed. | Figure 69: Pinch zoom operable (zoomed in)  Screengrab of site. "Gold Weekend Admission Package" link and image are taking up most of the screen. |

## 3.11: Reflow

Horizontal scrolling is not required to view the content on the page. For more information, see [WCAG2.1 SC 1.4.10: Reflow](https://www.w3.org/TR/WCAG21/#reflow).

### About this requirement

Please note that this differs from WCAG2.1 as SC 1.4.10 allows exceptions for tables, videos and images, which are not valid exceptions in this methodology.

Please note that this requirement specifically requires that a site be developed with responsive design. This committee believes that a responsive website is more accessible than a desktop site, however there may be occasions where site owners are restricted to a desktop site. In this case, we believe that these sites can meet the other accessibility requirements of this methodology, but not this Reflow requirement.

#### Exclusions

An exception is allowed when the site has been specifically designed to provide additional items on horizontal scroll, for example in the following site:

|  |  |
| --- | --- |
| Figure 70: 3.11: Reflow exclusion  Weather.com displaying video link previews horizontally |  |

### How to test

1. Open each page on the mobile device.
2. Does the content reflow so that it is all visible on the page without requiring the use of horizontal scrolling?

### Examples

#### Pass 1 – Image

In this example, the image reflows correctly and fits on the screen without horizontal scrolling.

|  |  |
| --- | --- |
| Figure 80: Image reflow  Portland Audubon featured image has resized to fit onto the mobile portrait display. |  |

#### Pass 2 –Video

In this example, the video reflows correctly and fits on the screen without horizontal scrolling.

|  |  |
| --- | --- |
| Figure 81: Video reflow  Fandom site where a video has resized to fit onto the mobile portrait display. |  |

#### Pass 3 – Table

The sun protection table meets the reflow requirements and does not require horizontal scrolling.

|  |  |
| --- | --- |
| Figure 82: Table reflow  The three columns have resized to fit onto the mobile portrait display. |  |

#### Pass 4 – Table

In the SF MOMA website, the table of dates fits in the screen without horizontal scrolling.

|  |  |
| --- | --- |
| Figure 83: Table reflow  November 2019 calendar is fully displayed on the mobile portrait screen. |  |

#### Pass 5 – Table

In the Rotten Tomatoes website, the table of dates fits in the screen without horizontal scrolling.

|  |  |
| --- | --- |
| Figure 84: Table reflow  The five  columns have resized to fit onto the mobile portrait display. |  |

# 4. Actionable items

## 4.1: Content on hover, focus or input

When additional content appears on hover, focus or input it is dismissable, hoverable and persistent (for more information see [SC 1.4.13: Content on Hover or Focus](https://www.w3.org/WAI/WCAG21/Understanding/content-on-hover-or-focus.html)).

### About this requirement

This requirement is similar to [SC 1.4.13: Content on Hover or Focus](https://www.w3.org/WAI/WCAG21/Understanding/content-on-hover-or-focus.html), however extends it to content displayed on input as well. Please note that some failures of this requirement are also defined as Exit traps (for example when a user is trapped in this new content and cannot close it). As per [SC 1.4.13: Content on Hover or Focus](https://www.w3.org/WAI/WCAG21/Understanding/content-on-hover-or-focus.html), this requirement has three conditions, that the additional content be:

* **Dismissable** – can the additional content be dismissed easily  
  As this version of the methodology does not include mouse behavior, we have generalised this requirement. Please note that we do not allow for the exception that the content is an input error.
* **Hoverable** – can the additional content be actioned  
  As this version of the methodology does not include mouse behavior, we have generalised this requirement.
* **Persistent** – the additional content remains visible until dismissed  
  As this version of the methodology does not include mouse behaviour and the dominant method of interaction with a mobile is touch, we have generalised this requirement.

#### Exclusions

The only exception to the dismissable portion of this requirement is if the additional content does not obscure or replace other content.

It is the opinion of this committee that the use of the TITLE attribute violates this requirement and therefore should not be used, unless additional JavaScript is included which allows this attribute to meet the dismissable, hoverable and persistent conditions.

### How to test

1. Identify all actionable items that display additional content on hover, focus and / or input.
2. Is the content persistent - does it remain onscreen until it is dismissed or another item receives hover, focus or input?
3. If the additional content contains actionable items, can they be actioned by touch?
4. If the additional content contains actionable items, can they be actioned by the keyboard?
5. Can the additional content be dismissed easily?

### Examples

#### Pass 1 – Hoverable and persistent

In the Filoli website on desktop, the sub-items for the top-level menu item for “Get Involved” appear on hover. On mobile, these appear when the “Get Involved” top-level menu item is actioned, and they are hoverable and persistent.

|  |  |
| --- | --- |
| Filoli website on desktop | Filoli website on mobile |
| Figure 85: Hoverable and persistent (desktop)  Dropdown menu from the top navigation bar's "Get Involved" option is open. | Figure 86: Hoverable and persistent (mobile)  Dropdown menu for the right-hand navigation bar's "Get Involved" option is open. The menu is not obscuring any other content on the page. |

## 4.2: Native UI

Native UI controls, objects, alerts and elements have been used.

### About this requirement

Content should be presented using semantic HTML tags. Tags should be used to represent appropriate content. For example, a heading tag should not be used to make text large; heading tags should be reserved for headings only. Lists should be presented in an <ol> or a <ul> and paragraphs of text should be presented in a <p> tag. Inversely, role=“presentation” should not be used on semantic HTML tags to remove a tag’s role if the role is essential to understanding the content. For example, a table used for formatting purposes should have role=“presentation” because the table is not a data table. A visual list, coded as a <ol>, but has role=“presentation” would not be announced as a list. The assistive technology user would not know a list was present, but the visual user would. For more information see [WCAG2.1 SC 1.3.1: Info and Relationships](https://www.w3.org/WAI/WCAG21/Understanding/info-and-relationships).

### How to test

1. Open the site and turn on the native **screen reader** feature (VoiceOver in iOS or TalkBack in Android).
2. Explore the site, using the object, alerts and elements that have been coded in the site: headings, buttons, links, dropdowns, menu lists, modal windows, input fields, checkboxes, date pickers, etc. All elements should be tested to make sure their purpose and function is coded semantically and they are announced by the screen reader. i.e. when a button is in focus, the screen reader announces “button.”   
   In **VoiceOver**, use the rotor to select Headings/Links/Form Controls, then swipe down.   
   In **TalkBack**, swipe up/down to select Headings/Links/Controls/Default, then swipe right.
3. If you are aware of any elements that are not semantic but have been custom-built instead, ensure these are also tested and work with VoiceOver/TalkBack and comply with the rest of this methodology.
4. If there is a **dropdown menu**, or **input fields that trigger a dropdown menu**, ensure that on activation, they launch the native UI selector, and do not open a dropdown on the screen.
5. If there is a **date picker**, ensure that on activation, it launches the native UI date picker, and this does not appear in the screen.
6. Ensure that when an **input field** is activated, it launches the native UI keyboard, number keyboard or external keyboard where relevant.
7. Test the focus in the site:

* When a new page loads, focus should land at the top.
* When a modal window opens, focus should land inside the modal (at the top), and focus should be constrained inside the modal.

1. In the event that the screen reader does not announce the correct tag or does not announce a tag at all (i.e. <p>), inspect the code associated with the element and ensure that the correct tag has been used

### Examples

#### Pass 1 – Native SELECT element

In this example, the native SELECT element has been used and so the options are presented in the standard mobile view.

|  |  |
| --- | --- |
| Figure 89: Native SELECT element on iOS  Lane Bryant site utilizing the iOS scroll list function. | Figure 90: Native SELECT element on Android  Lane Bryant site utilizing the Android popup radio button list function. |

#### Pass 2 – Native dropdown menu

The US states are all listed in an HTML5 SELECT box, which triggers the native dropdown functionality on both iOS and Android devices.

|  |  |
| --- | --- |
| Figure 91: Native dropdown menu  State abbreviations are displayed on a scrolling pop-up list with radio buttons. |  |

## 4.3: Descriptive text links

Where text links are used the text is visually descriptive (for more information see [SC 2.4.9: Link Purpose (Link Only)](https://www.w3.org/TR/UNDERSTANDING-WCAG20/navigation-mechanisms-link.html)).

### About this requirement

Assistive technologies such as screen readers can jump from link to link. It is therefore important that the text of the link is adequately descriptive.

### How to test

1. Identify all the links in the site.
2. Is each link descriptive of the target?

### Examples

#### Pass 1 – Descriptive text links

All text links are descriptive.

|  |  |
| --- | --- |
| Figure 94: Descriptive text links  The Age Site reads "Our coverage of the bushfire crisis is free for all readers. Please consider supporting our journalism with a subscription. "Bushfire crisis" and "with a subscription" are links. |  |

## 4.4: Non-keyboard options

When direct input via the keyboard is not required provide options for the user to achieve the same result (i.e. use dropdown, radio buttons & checkboxes, etc.).

### About this requirement

Using the onscreen or an external keyboard can be very difficult for some users. Therefore the use of the keyboard should be kept to a minimum and, where possible, replaced with an alternative method of accessing content.

Recommended alternatives to the keyboard include:

* Sitemap in addition to a Search field
* List of states in a dropdown instead of a free text field
* Autocomplete in addition to completely typing an option

### How to test

1. Identify all text input fields used.
2. Does the text input field allow for more than 100 options? (if not then the text input field should be a dropdown).
3. Is there an alternative way of accessing the content, without the keyboard?
4. If there isn’t an alternative way of accessing content does the text input field provide options on entering a few characters?

### Examples

#### Pass 1 – SELECT box used instead of text input field

Instead of having a free text field to enter the state in the US, the fifty states are listed as an HTML5 SELECT element so they can be selected without entering text in the field.

|  |  |
| --- | --- |
| Figure 96: SELECT box used instead of text input field  State abbreviations are displayed on a scrolling popup list with radio buttons. |  |

#### Pass 2 – Options provided on entering a few characters

In the weather.com site, suggestions for autofill are provided once text is entered in the search field. This means that a person wanting to search for Portland only needs to type the first few letters and then select the Portland option.

|  |  |
| --- | --- |
| Figure 97: Options provided on entering a few characters  "Portla" is entered into the search bar, and a dropdown listing various Portlands in the U.S. is shown. |  |

#### Pass 3 – Options provided on entering a few characters

In the Bureau of Meteorology site, suggestions for autofill are provided once text is entered in the search field. This means that a person wanting to search for Mallacoota only needs to type the first few letters and then select the Mallacoota option.

|  |  |
| --- | --- |
| Figure 98: Options provided on entering a few characters  "Malla" is entered into the search bar, and a dropdown lists cities such as Mallabula, Mallacoota and Mallala. |  |

## 4.5: Infinite scrolling

Infinite scrolling has not been used.

### About this requirement

Infinite scrolling can be problematic for a number of users. For example, screen reader users often scan a page when it first loads to determine what content is included. This is often done by pulling out links or headings. If infinite scrolling is used then screen reader users will not get a proper indication of the content of the page. Of course, if more content is added through a link then this issue still applies, however, in that case, the screen reader user will be made aware that there are additional items and that they may need to scan the page again.

In addition, often assistive technologies, such as screen readers, switches and keyboards do not trigger the infinite scrolling and therefore the rest of the content is inaccessible to these users.

Infinite scrolling should never be used to load additional content; it can cause an unreasonable number of swipes or keyboard tabbing to get through all the content. Often the user has no way of skipping or leaving the infinite scrolling until they are at the end of it. For more information see [WCAG2.1 SC 2.1.2: No Keyboard Trap.](https://www.w3.org/WAI/WCAG21/Understanding/no-keyboard-trap) Scrolling should be initiated by the user, and a reasonable amount of content should be displayed as a result. For more information see [WCAG2.1 SC 3.2.1: On Focus.](https://www.w3.org/WAI/WCAG21/Understanding/on-focus) Also see [WCAG 2.1 SC 3.2.5 Change on Request](https://www.w3.org/WAI/WCAG21/Understanding/change-on-request); while this is Level AAA, it is a best practice.

### How to test

1. Identify parts of application that shows a large list of items.
2. Ensure the list shows additional items only when the user requests additional items.

### Examples

#### Pass 1 – No infinite scrolling

In this site, once you have reached the end of the page no more stories load but there is a button called “More stories.” Activating this button will show more stories.

|  |  |
| --- | --- |
| Figure 100: No infinite scrolling  Screengrab of the bottom of a page on The Onion site. |  |

## 4.6: Color alone

Color alone should not be used to indicate actionable items (if not underlined). A secondary method, such as underline or bold should be used, in addition to color.

### About this requirement

This technique is aimed at visual users only. Although this requirement relies heavily on WCAG2.1 SC 1.4.1: Use of Color, this success criterion allows exceptions for actionable items that differ from text in color alone if that difference meets color contrast requirements of a 3:1 ratio. Mobile devices are, by nature, mobile and used in a variety of environments, including full sun and full darkness. This means that color differences that may be obvious to all users in an office environment could be unclear in other locations.

In addition to this, actionable items on desktops provide feedback to the user when the user mouses over (at the minimum the cursor changes icon) them or tabs to them (at the minimum the address of the actionable item is displayed at the bottom of the screen). In WCAG2 there are requirements to ensure that these focus indicators are clear to all users. Mobile devices do not provide this kind of feedback by default, so an important aspect of indicating actionable items is lost. Thus this committee believes color alone is not sufficient to indicate actionable items.

#### Exclusions

This committee believes that exceptions exist for links that are not inline text. This includes items in a navigation bar, button text (although these do need to meet color contrast requirements they do not need to meet this requirement) as there is enough visual information to separate these actionable items from text.

### How to test

1. Identify all actionable items.
2. Verify that color alone is not used to differentiate actionable items from non-actionable items.

### Examples

#### Pass 1 – Links are bolded

In the Supernatural Wiki, the links are green bold and the text is black.

|  |  |
| --- | --- |
| Figure 103: Links are bolded  Screengrab of the Fandom site. |  |

## 4.7: Removal of touch

Actionable elements are triggered only on removal of touch (for more information see [SC 2.5.2: Pointer Cancellation](https://www.w3.org/WAI/WCAG21/Understanding/pointer-cancellation)).

### About this requirement

#### Relationship to WCAG2.1

For more information see [SC 2.5.2: Pointer Cancellation](https://www.w3.org/WAI/WCAG21/Understanding/pointer-cancellation). This requirement is written as if “the down event of the pointer is not used to execute any part of the function”, however the other parts of [SC 2.5.2: Pointer Cancellation](https://www.w3.org/WAI/WCAG21/Understanding/pointer-cancellation) are included as exclusions.

#### Exclusions

As per [SC 2.5.2: Pointer Cancellation](https://www.w3.org/WAI/WCAG21/Understanding/pointer-cancellation) the following exceptions apply:

* **Abort or Undo**: Completion of the function is on the up-event, and a mechanism is available to abort the function before completion or undo the function after completion;
* **Up Reversal**: The up-event reverses any outcome of the preceding down event;
* **Essential**: Completing the function on the down-event is essential, for example a keyboard, an on-screen piano keyboard

### How to test

1. Identify all actionable items.
2. Press down on each element. While pressing down, move off of the active element and unpress.
3. The action will not take place.

### Examples

#### Pass 1 – Action on up event

In the Yahoo site, activating the ellipsis is on the up event

|  |  |
| --- | --- |
| Figure 110: Action on up event (original page)  Screengrab of au.yahoo.com homepage. | Figure 111: Action on up event (upon up event)  Screengrab of au.yahoo.com with pop up list displaying Sydney train lines. |

# 5. Navigational aids

## 5.1: Visual indicators

Visual indicators, (such as arrows, next and previous buttons) have been used to indicate swipe or scroll areas or additional functionality (for more information see [WCAG2.1 SC 2.5.1: Pointer Gestures](https://www.w3.org/WAI/WCAG21/Understanding/pointer-gestures.html)).

### About this requirement

When using a mobile device, there is less feedback presented to the user as to the functionality available. For example, on a desktop or laptop when the mouse hovers over an actionable item the cursor changes. This does not occur on mobile devices as mobiles are reliant on touch and not mouse movement. As a result, it is important to appropriately indicate additional content or functionality so users are aware of them.

### How to test

1. Identify all areas of the site where additional content or functionality is added.
2. Is there a visual indicator that indicates that swipe will show additional content?
3. Does this visual indicator meet color contrast, touch target size and inactive space requirements?

### Examples

#### Pass 1 – Right to left swipe

In this page, you can swipe from right to left to show additional ratings, as only ratings from 0 to 7 are visible. This is indicated as rating 7 is cut off by the right edge of the mobile screen.

|  |  |
| --- | --- |
| Figure 113: Right to left swipe  Mobile site for viewing and rating television shows. |  |

#### Pass 2 – Indicating additional information on touch

In this example, dual arrows indicate that there is additional functionality if you select the word. This pops up an overlay with annotations.

|  |  |
| --- | --- |
| Figure 114: Indicating additional information on touch (indicator)  Screenshot of site showing song lyrics. One lyric has a double arrow indicator in the middle of it. | Figure 115: Indicating additional information on touch (pop-up)  A popup showing that an annotation was made. |

#### Pass 3 – Indicating additional information on swipe

Two rows of content under the “Top Stories” heading indicate that more content is visible on swipe as part of the content is cut off.

|  |  |
| --- | --- |
| Figure 116: indicating additional information on swipe  A weather and news mobile site. |  |

#### Pass 4 – Indicating swipe from right to left to show more content

In this page, you can swipe from right to left to show more furniture. This is indicated as the third option is cut off by the right edge of the mobile screen.

|  |  |
| --- | --- |
| Figure 117: Indicating swipe from right to left to show more content  A furniture shopping mobile site displaying chairs. |  |

#### Pass 5 – Indicating swiping left and right

In this page you can swipe from right to left to show more podcasts, but you can also swipe from left to right to show more as well. This is indicated as the first and fourth options are cut off by the edge of the mobile screen.

|  |  |
| --- | --- |
| Figure 118: Indicating swiping left and right  Museum mobile site showing several webcasts available in a horizontal list. |  |

#### Pass 6 – Indicating link to expand contents

In this example, the text is acting as a link which expands a section. This is indicated via a plus sign.

|  |  |
| --- | --- |
| Links collapsed (this is the view on loading the page) | First link expanded |
| Figure 119: Indicating link to expand contents (links collapsed)  Frontier mobile site showing list of unexpanded options | Figure 120: Indicating link to expand contents (link expanded)  Frontier mobile site with Tour Dates and Ticketing dropdown expanded. |

#### Pass 7 – Arrow and dots indicating swipe area

There are two visual indicators – a right pointing arrow at the bottom of the image indicates that the image can be swiped and there are four dots under the image and the second dot is highlighted (please note that this is using color alone).

|  |  |
| --- | --- |
| Page on load | Page on swiping from right to left |
| Figure 121: Arrow and dots indicating swipe area (on load)  Portland Audubon site slideshow on first image. | Figure 122: Arrow and dots indicating swipe area (on swipe)  Portland Audubon site slideshow on second image. |

## 5.2: Character key shortcuts

Single character key shortcuts can be turned off, modified by the user or are active only on focus (for more information see [SC 2.1.4: Character Key Shortcuts](https://www.w3.org/WAI/WCAG21/Understanding/character-key-shortcuts.html)).

### About this requirement

Single character key shortcuts are incredibly important to users with physical disabilities, however when used with other assistive technologies these single character key shortcuts can unintentionally disrupt the user.

This requirement accords with see [SC 2.1.4: Character Key Shortcuts](https://www.w3.org/WAI/WCAG21/Understanding/character-key-shortcuts.html).

### How to test – turn off single character key shortcuts

1. Open the site.
2. Look for a Settings or Account section.
3. Does this section contain the ability to turn off the single character key shortcuts?

### How to test – modify single character key shortcuts

1. Open the site.
2. Look for a Settings or Account section.
3. Does this section contain the ability to modify the single character key shortcuts?

### How to test – single character key shortcuts available only on focus

1. Open the site. Ensure no active items are in focus.
2. Press all printing characters (i.e., all number, letter, sign and punctuation keys).
3. Hold down the Shift key and press the same keys again.
4. Ensure that nothing is triggered.

### Examples

#### Pass 1

There are currently no example passes documented for this requirement. Similar Pass examples are available in the Native App Test Cases document.

## 5.3: Descriptive headings

Blocks of content have descriptive headings (for more information see [SC 2.4.10: Section Headings](https://www.w3.org/TR/UNDERSTANDING-WCAG20/navigation-mechanisms-headings.html)).

### About this requirement

For more information see [SC 2.4.10: Section Headings](https://www.w3.org/TR/UNDERSTANDING-WCAG20/navigation-mechanisms-headings.html).

#### Exclusions

Creative content is excluded from this requirement.

### How to test

1. Pull out all the headings in the site.
2. Are the headings descriptive of their content?

### Examples

#### Pass 1 – Descriptive headings

The headings “Falls Festival in Lorne cancelled due to forecast extreme weather” and “Key points” are descriptive headings.

|  |  |
| --- | --- |
| Figure 126: Descriptive headings  Screengrab of abc.net.au mobile site. |  |

## 5.4: Inactivity timeout

Prior to starting a process, users are warned if there is a timeout and the length of time of inactivity that will trigger the timeout. Please note you will also need to comply with WCAG2 SC 2.2.1 Timing Adjustable (for more information see [SC 2.2.6: Timeouts](https://www.w3.org/WAI/WCAG21/Understanding/timeouts.html)).

### About this requirement

For more information see [SC 2.2.6: Timeouts](https://www.w3.org/WAI/WCAG21/Understanding/timeouts.html). Please note that this requirement differs from [SC 2.2.6: Timeouts](https://www.w3.org/WAI/WCAG21/Understanding/timeouts.html) as it does not allow for an exception if data is kept for longer than 20 hours. The committee’s opinion is that this causes a significant privacy issue and should not be used as a method of meeting this requirement.

### How to test

1. Identify any areas in the site that have a timeout.
2. Is the user warned of the length of inactivity that triggers the timeout prior to starting the process?

### Examples

There are currently no examples for this requirement.

|  |  |
| --- | --- |
|  |  |

## 5.5: Navigation features

Navigation features such as breadcrumbs, next and previous buttons are provided (for more information see [SC 2.4.8: Location](https://www.w3.org/TR/UNDERSTANDING-WCAG20/navigation-mechanisms-location.html)).

### About this requirement

For more information see [SC 2.4.8: Location](https://www.w3.org/TR/UNDERSTANDING-WCAG20/navigation-mechanisms-location.html).

Navigation features that are recommended on all mobile sites are:

* Breadcrumbs
* Menu link on all pages
* Site map
* Indicating the current section in navigation (without using color alone)
* Next and previous buttons in features such as image galleries, slideshows and multi-page forms.

### How to test

1. Open the website.
2. With the screen reader off, starting from the landing screen, explore each page of the website.
3. Look to see whether navigational aids – i.e. elements such as back buttons, breadcrumbs, next and previous buttons are provided as part of the design.
4. Ensure that these elements, if used repeatedly, are placed consistently on the pages on which they appear. Ensure these elements are also visually consistent. (See [SC 3.2.3 Consistent Navigation](https://www.w3.org/WAI/WCAG21/quickref/#consistent-navigation))
5. Ensure that each element receives visible focus on touch.
6. Ensure that each element is operable through swipe gesture and touch activation (double-tapping or other).
7. Ensure that each element has the same functionality on each page on which it appears.
8. Ensure that the page, screen or element is the expected destination as defined by the navigational aid.
9. Ensure that each element on the page that is clearly labelled to indicate the purpose (buttons) and/or is descriptiveof its purpose and/or destination (breadcrumb links).

### Examples

#### Pass 1 – Back button

In this example, there is an arrow to go back to the previous page and a close icon to close the pop-up that overlays the page.

|  |  |
| --- | --- |
| Figure 131: Back button  Screengrab of carsales.com.au with pop up list of car dealers. |  |

#### Pass 2 – Multiple methods of navigation

In the Coles website, there is a hamburger menu, search and a back button, that reads “Christmas.”

|  |  |
| --- | --- |
| Figure 132: Multiple methods of navigation  Screengrab of Coles.com.au mobile site. |  |

#### Pass 3 – Multiple methods of navigation

The weather.com site provides a menu that appears on all pages, as well as buttons at the bottom of the screen.

|  |  |
| --- | --- |
| Figure 133: Multiple methods of navigation  Screengrab of weather.com mobile site. |  |

## 5.6: ARIA

ARIA document landmarks have been used to appropriately describe document structure.

### About this requirement

It is the committee’s opinion that ARIA is essential in making a mobile site accessible to screen reader users, especially on iOS devices. The VoiceOver rotor allows users to navigate via ARIA elements and therefore utilizing these elements can significantly improve the accessibility of a site for these users.

### How to test

1. Open the website.
2. For each page, visually scan for what appears to be document landmarks.
3. Either use a screen reader for each page to verify that each item on the landmark list aligns with what it visually appears to be, or, use an automated accessibility evaluation tool to verify document structure aligns with a visual scan.

### Examples

There are currently no examples for this requirement.

|  |  |
| --- | --- |
|  |  |

# 6. Audio and video

## 6.1: Transcript

All audio and video have an accessible transcript.

### About this requirement

Although captions are essential to people who are Deaf or hard of hearing and audio descriptions are essential for people with vision impairments, there are users that either cannot interpret these features or are Deaf-Blind. In this case, the only accessible method is to provide a transcript. Please note that the transcript must include both the audio and visual content in the video. It is not sufficient to have a transcript of only speech.

### How to test

1. Open the site.
2. Navigate to the page(s) that contains – or launches – the video or audio player or file link.
3. Ensure that there is a textual transcript alternative available on the same page as the video or audio player.
   1. Ideally it is adjacent to the video or audio player, with easy access.
   2. The transcript may be in the form of text on a separate page in the application, or available as a download, or is available on a web page that is launched from the native app.
   3. Access and read the transcript to ensure the textual transcript accurately reflects the content of the video or audio file.
   4. Ensure the transcript is perceivable and operable using the built-in screen reader feature on mobile, whether on a page, webpage or PDF.
   5. Ensure that any links to the transcript, or a title for the transcript document, is descriptive and clearly identifies the transcript as the content.

### Examples

#### Pass 1 – Transcript provided

The 3PlayMedia website provides transcripts for all their videos.

|  |  |
| --- | --- |
| Figure 135: Transcript provided  Screenshot of a video playing on the 3PlayMedia mobile site with a transcript underneath. |  |

#### Pass 2 – Transcript provided

The AccessibilityOz website provides transcripts for all their videos.

|  |  |
| --- | --- |
| Figure 136: Transcript provided  Screenshot of a video playing on the AccessibilityOz mobile site with a transcript underneath. |  |

## 6.2: Captions

Captions must be closed captions.

### About this requirement

Open captions are burnt into the video and cannot be accessed by assistive technologies. Closed captions can be turned on and off by the user and often can be manipulated by the user, say by increasing the text size of the captions or font or color.

It is the opinion of this committee that all captions should be closed captions for the following reasons:

* They can be switched off, which means reducing distractions for some users
* They can be manipulated to meet a user’s requirements, such as large text and high color contrast
* They can be accessed by other assistive technologies, such as screen readers.

### How to test

1. Open the site.
2. Navigate to the page(s) that contains – or launches – the video or audio player or file link.
3. Is there a closed caption button on the player?
4. Does activating it open and close the captions?

### Examples

#### Pass 1 – Closed captions provided

All 3PlayMedia videos have closed captions.

|  |  |
| --- | --- |
| Figure 137: Closed captions provided  Screenshot of a video playing on the 3PlayMedia mobile site. Closed captioning is enabled. |  |

## 6.3: Live audio and video

Live captions and audio descriptions are provided for any live audio or video.

### About this requirement

It is the opinion of this committee that people with disabilities have a right to access live audio and video. Therefore, all live audio or video should have associated live captions and live audio descriptions.

### How to test

1. Open the site.
2. Navigate to the page(s) that contains the live audio and / or video.
3. Are captions provided?
4. Are they Closed Captions?
5. Are audio descriptions provided (either as a separate audio file or within the audio or video)?

### Examples

There are currently no examples for this requirement.

# 7. Forms

## 7.1: CAPTCHAs

Visual and audio CAPTCHAs are not used.

### About this requirement

It is the opinion of this committee that CAPTCHAs, by their nature, are inaccessible. Google’s reCAPTCHA (where the user ticks a box) may work for most users but users of assistive technologies often trip the CAPTCHA, which is inaccessible. Some alternatives to CAPTCHAs are:

* Human test question (for example, Is fire hot or cold?)
* Honeypot traps (where a field must be left blank or the form is not submitted)
* Server-side spam filters
* Server-side validation
* Time-stamped forms and not allow submissions of the same forms within a ten second timeframe

### How to test

1. Identify all forms in the site.
2. Ensure there are no CAPTCHAs.

### Examples

#### Pass 1 – No CAPTCHA

The Apple Store does not have a CAPTCHA.

|  |  |
| --- | --- |
| Figure 140: No CAPTCHA  Apple store log in screen. |  |

## 7.2: Context-sensitive help

All complex forms contain context-sensitive help as instructions at the beginning of the form and/or specific instructions at each field (for more information see [SC 3.3.5: Help](https://www.w3.org/TR/UNDERSTANDING-WCAG20/minimize-error-context-help.html)).

### About this requirement

The Level A and AA requirements regarding forms are mostly regarding the provision of accessible errors (for example [SC 3.3.1: Error Identification](http://www.w3.org/TR/UNDERSTANDING-WCAG20/minimize-error-identified.html), [SC 3.3.3: Error Suggestion](http://www.w3.org/TR/UNDERSTANDING-WCAG20/minimize-error-suggestions.html) and [SC 3.3.4: Error Prevention (Legal, Financial and Data](http://www.w3.org/TR/UNDERSTANDING-WCAG20/minimize-error-reversible.html)). The committee believes that providing instructions prior to form input is an important part of accessibility. This is particularly important on mobile devices as inputting content is already difficult, especially for some users.

There are two main ways to provide context-sensitive help:

* Instructions at the beginning of the form
* Instructions at each field on the requirements for that field

Context-sensitive help includes information such as:

* Requirements for passwords (for example one upper-case letter, one number, etc.)
* Characters that are not allowed in certain fields (for example, letters are not allowed in ID fields, as the ID consists only of numbers)
* Requirements for phone numbers (for example, area codes, etc.)

### How to test

1. Identify all forms.
2. Do they contain more than two fields?
3. Are detailed instructions provided at the beginning of the form? or
4. Are detailed instructions provided at each field.?

### Examples

#### Pass 1 – Context-sensitive help provided next to relevant field

The Create account process for IMDB provides context-sensitive help about creating a password – that “Passwords must be at least 8 characters” (please note that this fails the Visible Field Labels requirement).

|  |  |
| --- | --- |
| Figure 144: Context-sensitive help provided next to relevant field  IMDb create account form. |  |

#### Pass 2 – Context-sensitive help provided at the beginning of the form

When resetting a password on 3PlayMedia, the requirements of the password are detailed.

|  |  |
| --- | --- |
| Figure 145: Context-sensitive help provided at the beginning of the form  Screenshot of the 3PlayMedia Reset Password form. Password requirements are listed above the form. |  |

## 7.3: Error prevention

All submitted forms are reversible by the user, checked for errors by the application or confirmed prior to submission (for more information see [SC 3.3.6: Error Prevention (All)](https://www.w3.org/TR/UNDERSTANDING-WCAG20/minimize-error-reversible-all.html)).

### About this requirement

The Level A and AA requirements regarding forms are mostly regarding the provision of accessible errors (for example [SC 3.3.1: Error Identification](http://www.w3.org/TR/UNDERSTANDING-WCAG20/minimize-error-identified.html), [SC 3.3.3: Error Suggestion](http://www.w3.org/TR/UNDERSTANDING-WCAG20/minimize-error-suggestions.html) and [SC 3.3.4: Error Prevention (Legal, Financial and Data](http://www.w3.org/TR/UNDERSTANDING-WCAG20/minimize-error-reversible.html)), however error prevention is only required for legal, financial and data input. The committee believes that providing error prevention for all forms is an important part of accessibility. This is particularly important on mobile devices as inputting content is already difficult, especially for some users.

For more information see [SC 3.3.6: Error Prevention (All)](https://www.w3.org/TR/UNDERSTANDING-WCAG20/minimize-error-reversible-all.html).

#### Exceptions

An exception to this requirement is if the user opts out of error prevention. For example, Amazon has an option to “Buy with one click.” This does not meet the Error Prevention requirement, but it can be turned off in the Settings.

### How to test

1. Identify all forms.
2. Prior to submission is at least one of the following true:
   1. The submission is **reversible** (for example, a form that logs a user into an application is reversible if there is a log out feature); or
   2. The submission is **checked** for errors (for example, a login form identifies that a certain username does not exist when the user enters it and prior to the form being submitted); or
   3. The submission is **confirmed** by the user (for example, when filling out an account registration, a page is presented with all the input details so the user can review it prior to creating the account).

### Examples

#### Pass 1 – Checked

In this site, each field is checked when focus is removed from the field and errors identified.

|  |  |
| --- | --- |
| Figure 146: Checked  Screenshot of a create password form. The Password field has three characters typed in, and an error message below the field reads "Please enter at least 8 characters" |  |

#### Pass 2 – Checked

When incorrect information is submitted the form returns error messages explaining the errors.

|  |  |
| --- | --- |
| Form filled out incorrectly: | On submission of the form: |
| Figure 147: Checked (form filled out incorrectly)  A Contact Details form filled out but not yet submitted. | Figure 148: Checked (on submission of the form)  The Contact Details form has highlighted in red the incorrect fields and listed the specific errors below the relevant fields |

## 7.4: Positioned field labels

Field labels are positioned adjacent to their input field and appear closest to their respective input field in relation to other field labels and other input fields.

### About this requirement

As a mobile device has a small screen, if field labels are positioned far away from their respective input fields it is possible that they will not appear on the screen at the same time as the field that they describe. In addition, where a field label is positioned close to other input fields it may be unclear to some users which field the field label describes.

### How to test

1. Review the website for form fields
2. Ensure that all form fields have adjacent field labels
3. When two or more form fields appear on the same webpage, ensure that field labels appear adjacent to their respective form fields and are placed closest to their form field than other form fields
4. Ensure that field label spacing and layout is consistent between like form fields on the same page.
5. Ensure that all form fields and field labels maintain accessible color contrast ratios in all states (e.g. selected/not selected)

### Examples

#### Pass 1 – Field labels are positioned with their fields

The field labels are positioned closer to their respective fields than to any other fields.

|  |  |
| --- | --- |
| Figure 150: Field labels are positioned with their fields  Flowplayer.com log in screen. |  |

## 7.5: Visible field labels

Fields must have an associated visible label which is also programmatically associated with the field (please note that placeholding characters do not meet this requirement) (for more information see [SC 2.5.3: Label in Name](https://www.w3.org/WAI/WCAG21/Understanding/label-in-name.html)).

### About this requirement

WCAG2 allows for labelling of input fields by adjacent buttons or other items. As a mobile device has a small screen, if adjacent buttons or items are positioned far away from their respective input fields it is possible that they will not appear on the screen at the same time as the field that they describe. Therefore, this committee believes that all fields should have a visible field label that is also programmatically associated with the field (preferably with LABEL FOR and ID).

#### What do we mean by “visible field label”?

Although a text label for a field is preferred, this requirement does allow for a visible field label that is an icon or an image. However please note that [SC 1.1.1: Non-text Alternatives](http://www.w3.org/TR/UNDERSTANDING-WCAG20/text-equiv-all.html) would still apply.

#### Are placeholding characters visible field labels?

This committee does not believe that placeholding characters are an appropriate visible label for three reasons:

* By default, placeholding characters do not meet color contrast requirements and therefore will not be visible to some users.
* Placeholding characters that disappear when a user focuses on a field means the describing label is lost at this point.
* Placeholding characters that do not disappear when a user focuses on a field cause multiple problems for users of assistive technologies such as screen readers and often mean they cannot submit the form correctly.

### How to test

1. Review the fields in the form.
2. Does each field have a visible field label – text or icon?
3. Is the visible field label accessible (placeholding characters do not meet accessibility requirements)?

### Examples

#### Pass 1 – Text field labels

This website provides field labels positioned inside the field, but they are not placeholding characters.

|  |  |
| --- | --- |
| Figure 154: Text field labels  "Enter your details" form. |  |

## 7.6: Accessible name

If there is visible text label for a input field or component, the accessible name matches the visual name (for more information see [SC 2.5.3: Label in Name](https://www.w3.org/WAI/WCAG21/Understanding/label-in-name.html)).

### About this requirement

It is essential that when a visible text label is used for an input field or component that its accessible name matches the visual name. This is to ensure that users of assistive technologies, such as Dragon Naturally Speaking, know how to refer to the input field or component so they can appropriately interact with it.

The committee intends to provide additional guidance on how to meet this requirement in the coming year.

### How to test

1. Identify all input fields with visible text labels.
2. Does the text label match the accessible name? (You can do this by viewing the source)
3. Identify all user interface components with visible text labels.
4. Does the text label match the accessible name?

### Examples

#### Pass 1 – Visible text field label and accessible name match

In the WebAIM website, the visible text label “Search” matches the accessible name “Search.”

|  |
| --- |
| Figure 157: Visible text field label and accessible name match  WebAIM homepage showing the search bar. |
| <span><label **for="q"**>**Search:**</label> <input type="text" name="q" **id="q"**><input type="image" src="/media/template/search.png" alt="Submit Search"></span> |

## 7.7: Form and keyboard interaction

Forms interact appropriately with the keyboard, for example, providing submission on Enter and moving between fields.

### About this requirement

When the user is using an onscreen keyboard, this keyboard takes up almost half the space on the screen. Therefore, it is common that the Submit button is not visible when entering content into an input field. Where a form consists of a large number of input fields, the ability to move between fields using the keyboard is very important.

When the user is using an external keyboard, allowing submission on Enter means that the user can interact and submit the form using only the keyboard and not relying on touch.

#### Exclusions

An exception is allowed if the Enter key is required to act as an Enter key – in the event that the field allows for multiple lines of text input.

### How to test

1. For each form, enter each field.
2. If there is only one field, is the ability to move to the next field disabled?
3. If there is more than one field, does the keyboard include an option to move to the next field?
4. At the last field, does the keyboard include an option to Submit the form?

### Examples

#### Pass 1 – Form fields accessible by the keyboard

In this site, you can move between fields using the keyboard and also submit the form.

|  |  |
| --- | --- |
| Figure 162: Form fields accessible by the keyboard  Card number form on a banking site. |  |

## 7.8: HTML5

All HTML5 INPUT TYPES and AUTOCOMPLETE values are used according to the specification and where they are applicable (for more information see [SC 1.3.5: Identify Input Purpose](https://www.w3.org/WAI/WCAG21/Understanding/identify-input-purpose.html) and particularly the list of [input types that should be used](https://www.w3.org/TR/WCAG21/#input-purposes)).

### About this requirement

Using the onscreen or an external keyboard can be very difficult for some users. Therefore the use of the keyboard should be kept to a minimum. This can be done by meeting two requirements:

* By using standard HTML5 input types, which will display the correct keyboard on the mobile device (the numeric keyboard for a telephone field, for example)
* By using AUTOCOMPLETE attributes so that the device can automatically fill in commonly used fields, such as First name, Last name, Email, Address, etc.

#### NUMBER vs TEL types

In most cases it is preferable to use the TYPE=“TEL” attribute instead of TYPE=“NUMBER”.

### How to test – HTML5 INPUT TYPES

1. Identify each field.
2. For each field, verify that either:
   1. The correct keyboard appears for each form field (e.g. a number pad for phone numbers, a full keyboard with @ sign for email, etc.)
   2. The appropriate native selection mechanism appears for date and time-based fields.

### How to test – AUTOCOMPLETE

1. Identify each field.
2. Determine if it is a field that has an associated [AUTOCOMPLETE attribute](https://www.w3.org/TR/html52/sec-forms.html#autofill-field).
3. Check the code to ensure that the correct AUTOCOMPLETE attribute is used.

### Examples

#### Pass 1 – Correct HTML5 INPUT TYPE

In this website, the correct HTML5 format has been used for the email field, and the @ symbol is visible in the onscreen keyboard.

|  |  |
| --- | --- |
| Figure 164: Correct HTML5 INPUT TYPE  Focus is active on the "Confirm Email" field, and the alphabet keyboard is displayed. |  |

#### Pass 2 – Correct HTML5 INPUT TYPE

In this website, the correct HTML5 format has been used for the Phone field, and the number keyboard is displayed.

|  |  |
| --- | --- |
| Figure 165: Correct HTML5 INPUT TYPE  Focus is active on the "Phone" field, and the number keyboard is displayed. |  |

# Mobile and desktop relationship

## 8.1: Consistency

Item labelling between different types of a site (desktop, m.dot and/or responsive), and different variations of a responsive site, is consistent.

### About this requirement

WCAG2 requires consistency within a website. This requirement expands this requirement to include different variations of a site, for example:

* The desktop and the m.dot site
* Variations of a responsive site

Users are likely to interact with multiple types of a single site and it is important that iconography and labels are used consistently across all versions.

### How to test – identifying multiple sites

1. Open the site in a desktop.
2. Open the site on a mobile device.
3. If the site is the same on both devices then this requirement does not apply and no further testing is required.

### How to test

1. Open the site on the mobile device.
2. Identify all UI components, icons and other formatting (for example, link formatting).
3. Open the site on the desktop.
4. Are all UI components, icons and formatting the same?

### Examples

#### Pass 1 – Mobile and desktop are consistent

In the Insightly site, there is consistency between the mobile version and the desktop version (please note that this site fails the Reflow requirement).

|  |  |
| --- | --- |
| Mobile homepage | Desktop site homepage |
| Figure 168: Mobile and desktop are consistent (mobile homepage)  Insightly mobile homepage with vertical navigation menu displayed on the left-hand side. | Figure 169: Mobile and desktop are consistent (desktop homepage)  Insightly desktop homepage with vertical navigation menu displayed on the left-hand side. |

|  |  |
| --- | --- |
| Mobile subpage | Desktop subpage |
| Figure 170: Mobile and desktop are consistent (mobile subpage)  Insightly mobile subpage with vertical navigation menu displayed on the left-hand side. | Figure 171: Mobile and desktop are consistent (desktop subpage)  Insightly desktop subpage with vertical navigation menu displayed on the left-hand side. |

## 8.2: Linking between types of a site

Links between different types of a site (desktop, m.dot and/or responsive) have been provided, where the site is not solely a responsive site.

### About this requirement

WCAG2 clearly states that all functionality be available on all different variations of a responsive site. However, in some cases an m.dot site is created specifically to provide limited functionality (for example a map of the campus). In these cases, it is adequate to provide limited functionality on the m.dot site, as long as the user can easily move to the desktop (or responsive) site to access all content and functionality.

It is important that the user can move between both types of sites, for example:

* Move from the m.dot site to the desktop / responsive site
* Move from the desktop / responsive site to the m.dot site

### How to test – identifying multiple sites

1. Open the site in a desktop.
2. Open the site on a mobile device.
3. If the site is the same on both devices then this requirement does not apply and no further testing is required.

### How to test – mobile device

1. Open the site on the mobile device.
2. Is there a link to the desktop / responsive site?
3. Activate the link. Does the site change?
4. Is there a link back to the mobile version of the site?

### How to test – desktop

1. Open the site on the desktop.
2. Is there a link to the mobile site?
3. Activate the link. Does the site change?
4. Is there a link back to the desktop version of the site?

### Examples

#### Pass 1 – Can move between sites on both mobile and desktop

In the IMDB site, in the mobile view in the footer there is a link to go to the desktop view and when on the desktop site there is a link to go back to the mobile view.

|  |  |
| --- | --- |
| Mobile site: | Desktop site: |
| Figure 174: Can move between sites on both mobile and desktop (mobile site)  IMBd mobile site homepage. | Figure 175: Can move between sites on both mobile and desktop (desktop site)  IMBd desktop site homepage on mobile. |

#### Pass 2 – Can move between sites on both mobile and desktop

In a Wikipedia page, in the mobile view in the footer there is a link to go to the desktop view and when on the desktop site there is a link to go back to the mobile view.

|  |  |
| --- | --- |
| Mobile site | Desktop site |
| Figure 176: Can move between sites on both mobile and desktop (mobile site)  Wikipedia mobile site page. | Figure 177: Can move between sites on both mobile and desktop (desktop site)  Wikipedia desktop site page on mobile. |

# Acknowledgements

## Relationship to existing Accessibility testing standards

This document is based on:

* W3C Web Content Accessibility Guidelines, Version 2.0
* W3C Web Content Accessibility Guidelines, Version 2.1
* BBC Mobile Accessibility Guidelines
* AccessibilityOz Mobile Testing Methodology
* TPG Mobile Testing Guide

## ICT Accessibility Testing Symposium Mobile Site Sub-Committee

This document was developed by the ICT Accessibility Testing Symposium Mobile Sub-Committee. Members include: Gian Wild (Co-Chair), Peter McNally (Co-Chair), Brent Davis, Corbb O'Connor, Karen Herr, Kathryn Weber-Hottleman, Kathy Eng, Laura Renfro, Megha Rajopadhye, Mona Rekhi, Morgan Lee Kestner, Rafal Charlampowicz, Ryan Pugh, Steve Sawczyn, Sunish Gupta, Tom Lawton and Chris Law.

# Contacts

## Gian Wild

**Web:** [www.accessibilityoz.com](http://www.accessibilityoz.com)

**Phone:** 415 621 9366

**Email:** [gian@accessibilityoz.com](mailto:gian@accessibilityoz.com)

## Peter McNally

**Web:** [www.bentley.edu](http://www.bentley.edu)

**Phone:** 781 891 2893

**Email:** [pmcnally@bentley.edu](mailto:pmcnally@bentley.edu)