W3C Working Draft

W3C Accessibility Guidelines (WCAG) 3.0

W3C Working Draft 03 March 2026

▼ **More details about this document**

This version:
https://www.w3.org/TR/2026/WD-wcag-3.0-20260303/

Latest published version:
https://www.w3.org/TR/wcag-3.0/

Latest editor's draft:
https://w3c.github.io/wcag3/guidelines/

History:
https://www.w3.org/standards/history/wcag-3.0/
Commit history

Editors:
Rachael Bradley Montgomery (Library of Congress)
Alastair Campbell (Nomensa)
Chuck Adams (Oracle)
Kevin White (W3C)
Giacomo Petri (UsableNet)
Julie Rawe (Understood.org)
Francis Storr (Intel Corporation)
Makoto Ueki (Invited Expert)
Hidde de Vries (Logius)

Former editors:
Jeanne Spellman (TetraLogical)
Michael Cooper, Staff Contact, 2016-2023 (W3C)
Shawn Lauriat, Editor, 2016-2023 (Google, Inc.)
Wilco Fiers, Project Manager, 2021-2023 (Deque Systems, Inc.)

Feedback:
GitHub w3c/wcag3 (pull requests, new issue, open issues)
public-agwg-comments@w3.org with subject line `[wcag-3.0] … message topic …` (archives)

Abstract

W3C Accessibility Guidelines (WCAG) 3.0 will provide a wide range of recommendations for making web content more accessible to users with disabilities. Following these guidelines will address many of the needs of users with blindness, low vision and other vision impairments; deafness and hearing loss; limited movement and dexterity; speech disabilities; sensory disorders; cognitive and learning disabilities; and combinations of any of these disabilities. These guidelines address the accessibility of web content on desktops, laptops, tablets, mobile devices, wearable devices, and other Web of Things devices. The guidelines apply to various types of web content, including static, dynamic, interactive, and streaming content; audiovisual media; virtual and augmented reality; and alternative access presentation and control. These guidelines also address related web tools such as user agents (browsers and assistive technologies), content management systems, authoring tools, and testing tools.

Each guideline in this standard provides information on accessibility practices that address documented user needs of people with disabilities. Guidelines are supported by multiple requirements and assertions to determine whether the need has been met. Guidelines are also supported by technology-specific methods to meet each requirement or assertion.

To keep pace with changing technology, this specification is expected to be updated regularly with updates to and new methods, requirements, and guidelines that address new needs as technologies evolve. For entities that make formal claims of conformance to these guidelines, several levels of conformance are available to address the diverse nature of digital content and the type of testing that is performed.

For an overview of WCAG 3 and links to WCAG technical and educational material, see WCAG 3 Introduction.

Status of This Document

This section describes the status of this document at the time of its publication. A list of current W3C publications and the latest revision of this technical report can be found in the W3C standards and drafts index.

This is an update to W3C Accessibility Guidelines (WCAG) 3.0. It includes all requirements that have reached the developing status.

To comment, file an issue in the wcag3 GitHub repository. Create separate GitHub issues for each comment, rather than commenting on multiple topics in a single issue. It is free to create a GitHub account to file issues. If filing issues in GitHub is not feasible, email public-agwg-comments@w3.org (comment archive).

In-progress updates to the guidelines can be viewed in the public Editor's Draft.

This document was published by the Accessibility Guidelines Working Group as a Working Draft using the Recommendation track.

Publication as a Working Draft does not imply endorsement by W3C and its Members.

This is a draft document and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to cite this document as other than a work in progress.

This document was produced by a group operating under the W3C Patent Policy. W3C maintains a public list of any patent disclosures made in connection with the deliverables of the group; that page also includes instructions for disclosing a patent. An individual who has actual knowledge of a patent that the individual believes contains Essential Claim(s) must disclose the information in accordance with section 6 of the W3C Patent Policy.

This document is governed by the 18 August 2025 W3C Process Document.

Table of Contents

§ 1. Introduction

This section (with its subsections) provides advice only and does not specify guidelines, meaning it is informative or non-normative.

▶ **Plain language summary of “Introduction”**

§ 1.1 About this draft

This draft includes an updated list of the potential guidelines, requirements, and assertions that have progressed to Developing status. While this draft has moved closer towards completion, it still has several years of work. Details will change and we encourage comments based on the questions below.

Requirements and assertions at the Exploratory status are not included in this Working Draft. If you would like to see the complete list, please review the Editor's Draft.

Please consider the following questions when reviewing this draft:

- Are there user needs which are not covered?
- The conformance model has been updated, there are several questions in the conformance section
- Is the "applies when" and "except when" format for requirements understandable and useful?
- Does the increased granularity of the requirements help understanding and application or would consolidating requirements be a better approach?

To provide feedback, please open a new issue in the WCAG 3 GitHub repository. Create a separate GitHub issue for each topic, rather than commenting on multiple topics in a single issue.

If it's not feasible for you to use GitHub, email your comments to public-agwg-comments@w3.org (comment archive). Please put your comments in the body of the message, not as an attachment.

§ 1.1.1 Draft requirements

The list of requirements is longer than the list of success criteria in WCAG 2. This is because:

- the intent at this stage is to be as inclusive as possible of potential requirements, and
- WCAG 3 requirements are more granular than WCAG 2 success criteria.

The final set of requirements in WCAG 3 will be different from what is in this draft. Requirements are likely to be added, combined, and removed. We also expect changes to the text of the requirements. Not all of the requirements will be used to meet the base level of conformance.

§ 1.1.2 Section status levels

As part of the WCAG 3 drafting process, each normative section of this document is given a status. This status is used to indicate how far along in the development this section is, how ready it is for experimental adoption, and what kind of feedback the Accessibility Guidelines Working Group is looking for.

- **Placeholder**: This content is temporary. It showcases the type of content to expect here. All of this is expected to be replaced. No feedback is needed on placeholder content.
- **Exploratory**: This content is not refined; details and definitions may be missing. The working group is exploring what direction to take with this section. Feedback should be about the proposed direction.
- **Developing**: This content has been roughly agreed on in terms of what is needed for this section, although not all high-level concerns have been settled. Details have been added, but are yet to be worked out. Feedback should be focused on ensuring the section is usable and reasonable, in a broad sense.
- **Refining**: This content is ready for wide public review and experimental adoption. The working group has reached consensus on this section. Feedback should be focused on the feasibility of implementation.
- **Mature**: This content is believed by the working group to be ready for recommendation. Feedback on this section should be focused on edge-case scenarios that the working group may not have anticipated.

§ 1.2 About WCAG 3

This specification presents a new model and guidelines to make web content and applications accessible to people with disabilities. W3C Accessibility Guidelines (WCAG) 3.0 supports a wide set of user needs, uses new approaches to testing, and allows frequent maintenance of guidelines and related content to keep pace with accelerating technology changes. WCAG 3 supports this evolution by focusing on the functional needs of users. These needs are then supported by guidelines that are written as outcome statements, requirements, assertions, and technology-specific methods to meet those needs.

WCAG 3 is a successor to Web Content Accessibility Guidelines 2.2 [*WCAG22*] and previous versions, but does not deprecate WCAG 2. It will also incorporate some content from and partially extend User Agent Accessibility Guidelines 2.0 [*UAAG20*] and Authoring Tool Accessibility

Guidelines 2.0 [*ATAG20*]. These earlier versions provided a flexible model that kept them relevant for over 15 years. However, changing technology and changing needs of people with disabilities have led to the need for a new model to address content accessibility more comprehensively and flexibly.

There are many differences between WCAG 2 and WCAG 3. The WCAG 3 guidelines address the accessibility of web content on desktops, laptops, tablets, mobile devices, wearable devices, and other Web of Things devices. The guidelines apply to various types of web content, including static, dynamic, interactive, and streaming content; visual and auditory media; virtual and augmented reality; and alternative access presentation and control methods. These guidelines also address related web tools such as user agents (browsers and assistive technologies), content management systems, authoring tools, and testing tools.

Each guideline in this standard provides information on accessibility practices that address documented user needs of people with disabilities. Guidelines are supported by multiple requirements to determine whether the need has been met. Guidelines are also supported by technology-specific methods to meet each requirement.

Content that conforms to WCAG 2.2 Level A and Level AA is expected to meet most of the minimum conformance level of this new standard but, since WCAG 3 includes additional tests and different scoring mechanics, additional work will be needed to reach full conformance. Since the new standard will use a different conformance model, the Accessibility Guidelines Working Group expects that some organizations may wish to continue using WCAG 2, while others may wish to migrate to the new standard. For those that wish to migrate to WCAG 3, the Working Group will provide transition support materials, which may use mapping and other approaches to facilitate migration.

§ 2. Guidelines DEVELOPING

This section (with its subsections) provides requirements which must be followed to conform to the specification, meaning it is normative.

▶ **Plain language summary of “Guidelines”**

EDITOR'S NOTE

The individuals and organizations that use WCAG vary widely and include web designers and developers, policy makers, purchasing agents, teachers, and students. To meet the varying needs of this audience, several layers of guidance will be provided including guidelines written as outcome statements, requirements that can be tested, assertions, a rich collection of methods, resource links, and code samples.

The following list is an initial set of potential guidelines and requirements that the Working Group will be exploring. The goal is to guide the next phase of work. They should be considered drafts and should not be considered as final content of WCAG 3.0.

Ordinarily, exploratory content includes editor's notes listing concerns and questions for each item. Because this Guidelines section is very early in the process of working on WCAG 3, this editor's note covers most of the content in this section. Unless otherwise noted, all items in the list are exploratory at this point. It is a list of all possible topics for consideration. Not all items listed will be included in the final version of WCAG 3.0.

The guidelines and requirements listed below came from analysis of user needs that the Working Group has been studying, examining, and researching. They have not been refined and do not include essential exceptions or methods. Some requirements may be best addressed by authoring tools or at the platform level. Many requirements need additional work to better define the scope and to ensure they apply correctly to multiple languages, cultures, and writing systems. We will address these questions as we further explore each requirement.

Additional Research

One goal of publishing this list is to identify gaps in current research and request assistance filling those gaps.

Editor's notes indicate the requirements within this list where the Working Group has not found enough research to fully validate the guidance and create methods to support it or additional work is needed to evaluate existing research. If you know of existing research or if you are interested in conducting research in this area, please file a GitHub issue or send email to public-agwg-comments@w3.org (comment archive).

§ 2.1 Images and media DEVELOPING

§ Guideline 2.1.1 Image alternatives

Users have equivalent alternatives for images.

§ Core requirement: Images detectable DEVELOPING

Non-decorative images are detectable.

Images detectable issues

Applies when

- content includes non-decorative images.

▶ **Tests**

§ Core requirement: Decorative images hidden DEVELOPING

Decorative images are programmatically hidden.

Decorative images hidden issues

▶ **Tests**

§ Core requirement: Image alternatives available DEVELOPING

Equivalent text alternatives are available for images that convey content.

Image alternatives available issues

▶ **Tests**

§ Guideline 2.1.2 Media alternatives

Users have equivalent alternatives for audio and video content.

§ Core requirement: Transcripts available DEVELOPING

Transcripts are available for all audio and video content.

Transcripts available issues

Except when

- The audio or video content is an alternative for text and clearly labelled as such.
- It is a background video with no spoken content.

▶ **Tests**

§ Core requirement: Media alternatives equivalent DEVELOPING

Equivalent media alternatives are available for audio and video content.

Media alternatives equivalent issues

Except when

- It is a background video with no spoken content.

▶ **Tests**

§ Core requirement: Media alternatives findable DEVELOPING

A mechanism is available within the page/view to access the media alternatives for audio and video.

Media alternatives findable issues

Except when

- It is a decorative audio or video.

▶ **Tests**

§ **Supplemental requirement: Speakers identified** DEVELOPING

Speakers are identified understandably within all media alternatives.

Speakers identified issues

Except when

- The speaker has a hidden identity as part of narrative structure.
- Identity is identifiable within the context of use.

Applies when

- There are multiple speakers in the video.

EXAMPLE
Initially using “Maya Angelou” within the context of a story regarding poetry and then using “Angelou”.

▶ **Tests**

§ **Supplemental requirement: Speaker language identified** DEVELOPING

When more than one language is spoken in audio content, the language spoken by each speaker is identified in all media alternatives.

Speaker language identified issues

Except when

- Words are used incidentally.

▶ **Tests**

§ **Core requirement: Sounds identified** DEVELOPING

Sounds needed to understand the media are identified or described in captions and transcripts.

Sounds identified issues

EDITOR'S NOTE

This includes sound effects and other non-spoken audio content.

Except when

- it is a decorative video.

Applies when

- there is sound.

▶ **Tests**

§ **Core requirement: Visual information identified** DEVELOPING

Visual information needed to understand the media is described in the transcript and audio description.

Visual information identified issues

EDITOR'S NOTE

- This includes actions, charts or informative visuals, scene changes, and on-screen text,

▶ **Tests**

§ **Supplemental requirement: Sign language available (prerecorded)** DEVELOPING

Sign language interpretation is provided for all prerecorded audio content in the primary sign language of the intended audience or region.

Sign language available (prerecorded) issues

Except when

- It is a decorative background sound.

▶ **Tests**

§ **Assertion: Sign language available (live)** DEVELOPING

[Title, role, or organization] asserts that:

Sign language available (live) issues

- We provide Sign language interpretation for all live audio content in the primary sign language of the intended audience or region.

Information that needs to be included publicly:

- Title, role or organization making the assertion (if different from the conformance claim).
- Date of assertion (if different from the date of the conformance claim).
- Example recording of a signed live event.

Recommended internal documentation (Informative):

- Procurement procedure for sign language interpreters.

§ **Assertion: Media alternatives style guide** DEVELOPING

[Title, role, or organization] asserts that:

Media alternatives style guide issues

- Our organization has a style guide that includes guidance on media alternatives and a policy and/or processes that the style guide must be followed

Information that needs to be included publicly:

- Title, role or organization making the assertion (if different from the conformance claim).
- Date of when the style guide was published.
- Date of assertion (if different from the date of the conformance claim).

Recommended internal documentation (Informative):

- Summary of the needs of users involved.
- Identified issues and details of solutions applied.
- Section labels relevant to image alternatives, or
- Copy or snapshot of the style guide

EXAMPLE

- **Name:** ABC Inc. Style Guide for Captions
- **Version:** 1.2
- **Date:** October 2024
- **Description:** The style guide include sections such as:
  - What are captions?
  - Who needs captions?
  - Guideline provided by WCAG 3
  - Recommended style of captions
  - Resources
- **Example:** The style guide has requirements such as:
  - Never use more than two lines
  - Use parentheses for sound representation
  - Identify no sound or long pause
  - …

§ **Assertion: Accessible video player selected** DEVELOPING

[Title, role, or organization] asserts that:

Accessible video player selected issues

- We provide a video player that supports appropriate media alternatives. The video player includes the following features [list all that apply]:
  - Supports closed captions in a standard caption format;
  - Turning captions on and off;
  - Turning audio descriptions on and off;
  - Adjusting caption styles, including but not limited to: font size, font weight, font style, font color, background color, background transparency, and placement;
  - Changing the location of captions; and

- Changing the language of the audio descriptions.

Applies when

- a video is used that does not play in standard browsers.

Information that needs to be included publicly:

- Title, role or organization making the assertion (if different from the conformance claim).
- Date of assertion (if different from the date of the conformance claim).
- Feature included from the list

Recommended internal documentation (Informative):

- Video player documentation detailing functional support for media alternatives.

§ **Assertion: Media alternatives usability testing** DEVELOPING

[Title, role, or organization] asserts that:

Media alternatives usability testing issues

- We have conducted usability testing with users who need media alternatives, and changes were made to fix or mitigate the issues found.

Information that needs to be included publicly:

- Title, role or organization making the assertion (if different from the conformance claim).
- Date of when the usability testing was conducted.
- Date of assertion (if different from the date of the conformance claim).

Recommended internal documentation (Informative):

- Scope
- Types of disabilities each user had

- Number of users (for each type of disability)
- Date of testing
- Identified issues and details of solutions applied.

EXAMPLE

- **Scope:** videos on the following URLs:
  - "What is accessibility?" https://xxx.xxx/yyyy1
  - "About assistive technologies" https://xxx.xxx/yyyy2
  - "How to use WCAG" https://xxx.xxx/yyyy3
- **Type and number:** 8 users in total
  - 5 users who are deaf
  - 3 users who are hard of hearing
- **Dates:** 10-11 October 2024
- **Examples of fixed issues:**
  - Identified where the speaker changed
  - Described background music
  - Indicated that a dog was barking

§ **Assertion: Reviewed by content authors** DEVELOPING

[Title, role, or organization] asserts that:

Reviewed by content authors issues

- We have reviewed the media alternatives.

Information that needs to be included publicly:

- Title, role or organization making the assertion (if different from the conformance claim).
- Date of the review
- Date of assertion (if different from the date of the conformance claim).

Recommended internal documentation (Informative):

- Role of the creator
- Number of creators (for each Role)
- Date (Period) of review
- Examples of fixed issues based on the review

EXAMPLE

- **Roles and numbers:** 3 creators
  - 1 director
  - 1 planner
  - 1 screenwriter/playwright
- **Period:** 10-21 October 2024
- **Examples of fixed issues:**
  - Provided more details based on feedback
  - Rewrote some scripts to reflect the director's intent.
  - Added the description of the background shown in the video

Guideline 2.1.3 Non-text alternatives

Users have alternatives available for non-text, non-image content that conveys context or meaning.

Core requirement: Non-text content not relied on DEVELOPING

All non-text content that is not decorative includes a programmatically determinable equivalent text alternatives.

Non-text content not relied on issues

▶ **Tests**

§ Guideline 2.1.4 Captions

Users have captions for the audio content.

§ Supplemental requirement: Captions adjustable DEVELOPING

The appearance of captions, including associated visual indicators, is adaptable including font size, font weight, font style, font color, background color, background transparency, and placement.

Captions adjustable issues

▶ **Tests**

§ Core requirement: Captions available (prerecorded) DEVELOPING

Captions are available for all prerecorded audio content.

Captions available (prerecorded) issues

Except when

- The audio content is an alternative for text and clearly labelled as such.

▶ **Tests**

§ Core requirement: Captions available (live) DEVELOPING

Captions are available for all live audio content.

Captions available (live) issues

▶ **Tests**

§ **Supplemental requirement: Captions unobstructed** DEVELOPING

Captions are placed on the screen so that they do not hide visual information needed to understand the video content.

Captions unobstructed issues

▶ **Tests**

§ **Core requirement: Captions synchronized** DEVELOPING

Captions are synchronized with the audio content of synchronized media.

Captions synchronized issues

▶ **Tests**

§ **Core requirement: Captions centered (immersive)** DEVELOPING

In 360-degree digital environments, captions remain directly in front of the user.

Captions centered (immersive) issues

Applies when

- the position of the captions is controlled by the user.

▶ **Tests**

§ **Supplemental requirement: Captions controllable** DEVELOPING

A mechanism is available to turn captions on and off.

Captions controllable issues

Except when

- Captions are hard coded into the video content.

▶ **Tests**

§ **Core requirement: Direction indicated (immersive)** DEVELOPING

In 360-degree digital environments, the direction of a sound or speech is indicated when audio is heard from outside the current view.

Direction indicated (immersive) issues

EXAMPLE

- **Visual indicators:** An arrow with an outline pointing to the speaker or the direction of a sound
- **Customizable styles:** Users can select various arrow sizes, colors, and outline colors

▶ **Tests**

§ Guideline 2.1.5 Audio descriptions

Users have audio descriptions for video content.

§ **Core requirement: Audio descriptions available (prerecorded)** DEVELOPING

Audio descriptions are available in prerecorded video for visual content needed to understand the media.

Audio descriptions available (prerecorded) issues

EDITOR'S NOTE

- WCAG 3 needs to specify how to handle video content with audio that does not include gaps to insert audio descriptions. Two possible solutions are providing an exception that allows the content author(s) to use descriptive transcripts instead or requiring content authors to provide an extended audio description.

Except when

- the video content is an alternative for text and clearly labelled as such.

▶ **Tests**

§ **Core requirement: Audio descriptions synchronized** DEVELOPING

Audio descriptions are synchronized with video content without overlapping dialogue and meaningful audio content.

Audio descriptions synchronized issues

Except when

- There are no audio descriptions.

▶ **Tests**

§ **Supplemental requirement: Audio descriptions available (live)** DEVELOPING

Audio descriptions are available in live video for visual content needed to understand the media.

Audio descriptions available (live) issues

▶ **Tests**

§ **Core requirement: Extended audio descriptions available** DEVELOPING

The video pauses to extend the audio track and provides an extended audio description to describe visual information needed to understand the media.

Extended audio descriptions available issues

Applies when

- the existing pauses in a soundtrack are not long enough.

▶ **Tests**

§ **Supplemental requirement: Audio description language adjustable** DEVELOPING

A mechanism is available that allows users to change the audio description language if multiple languages are available.

Audio description language adjustable issues

▶ **Tests**

§ Guideline 2.1.6 Figure captions

Users can view figure captions even if not focused at figure.

> EDITOR'S NOTE
> Requirements and assertions for this guideline do not appear here because they have not yet progressed beyond exploratory. See the Editor's Draft for the complete list of potential requirements and assertions.

§ Guideline 2.1.7 Single sense

Users have content that does not rely on a single sense or perception.

§ **Core requirement: Hue not relied on** DEVELOPING

Information is not conveyed by hue alone.

Hue not relied on issues

NOTE

Information conveyed includes presenting data or meaning, indicating an action, prompting a response, distinguishing between items, conveying boundaries, etc. Artistic expression is not part of information conveyed.

Except when

- Content is artistic or expressive.
- Content is designed only for a device that is limited to presenting hues.

▶ **Tests**

§ **Core requirement: Graphical object contrast sufficient** DEVELOPING

Parts of graphical object required to understand the content meet a minimum contrast ratio test

Graphical object contrast sufficient issues

Except when

- a particular presentation of graphical objects is essential to the information being conveyed.

▶ **Tests**

§ **Core requirement: Visual depth not relied on** DEVELOPING

Information is not conveyed through visual depth perception alone.

Visual depth not relied on issues

▶ **Tests**

§ **Core requirement: Sound not relied on** DEVELOPING

Information is not conveyed by sound alone.

Sound not relied on issues

Except when

- Content is audio-based media.

NOTE
Information conveyed includes presenting data or meaning, indicating an action, prompting a response, distinguishing between items, conveying boundaries, etc. Artistic expression is not part of information conveyed.

▶ **Tests**

§ **Core requirement: Spatial audio not relied on** DEVELOPING

Information is not conveyed by spatial audio alone.

Spatial audio not relied on issues

NOTE
Information conveyed includes presenting data or meaning, indicating an action, prompting a response, distinguishing between items, conveying boundaries, etc. Artistic expression is not part of information conveyed.

▶ **Tests**

§ 2.2 Text and wording

§ Guideline 2.2.1 Text appearance

Users can read visually rendered text.

> EDITOR'S NOTE
> AGWG remains committed to providing Text Appearance guidance for multiple scripts and/or languages. To support our internationalization goals, AGWG is exploring ways to:
>
> - not require WCAG 3 to be republished when metrics are added for more scripts/languages,
> - make script/language-specific metrics normative,
> - meet the needs of different scripts and languages by allowing flexibility in which metrics are required and in their values, and
> - preserve backward compatibility.

§ Core requirement: Text size adjustable DEVELOPING

Text can be increased in size to at least 200% of the platform's default body-text size.

Text size adjustable issues

Except when

- the same text is available elsewhere in the page/view which can be increased to at least 200% of the platform's default body-text size.

▶ **Tests**

§ Core requirement: Text color adjustable DEVELOPING

The foreground and background color of text can be adjusted without losing content or functionality.

Text color adjustable issues

NOTE
The requirement is that the text is manipulable and the colors can be overridden. That could be achieved by the user-agent (including operating system, browser, and assistive technology), or provided by the content author.

Applies when

- text is presented, including text embedded in an image format.

Except when

- the text is:
  - also present elsewhere in the page/view which meets the requirement,
  - part of an inactive Interactive element,
  - pure decoration,
  - not visible to anyone,
  - part of a picture that includes significant other visual content,
  - part of a logo.

▶ **Tests**

§ **Supplemental requirement: Text customizations retained** DEVELOPING

Content that is exported, saved, or printed retains user-applied text-appearance customizations.

Text customizations retained issues

Applies when

- the page/view can be exported, saved, or printed.

EXAMPLE 1

Examples of interoperable formats

- PDF
- HTML
- SVG
- OpenDocument

EXAMPLE 2

Examples of non-interoperable formats

- Scriptable Network Graphics
- DOCX, unless you know your audience has the required software (for example, you're writing internal documents for your company which provides employees with the necessary software)
- Java object code

▶ **Tests**

§ Guideline 2.2.2 Text-to-speech

Users can access text content and its meaning with text-to-speech tools.

§ Core requirement: Text detectable DEVELOPING

All visible text has a programmatically determinable equivalent.

Text detectable issues

Except when

- making visible text programmatically determinable would lead to duplication within the view.

▶ **Tests**

§ **Core requirement: Human language detectable** DEVELOPING

The human language of all content within the view is programmatically determinable.

Human language detectable issues

Except when

- a language tag is not available in ISO 639 language codes, or
- the technology used to create the view does not support indicating languages.

▶ **Tests**

§ **Core requirement: Numerical metadata available** DEVELOPING

Numerical information includes sufficient context in written text and a programmatic equivalent to avoid confusion when presenting dates, temperatures, time, and Roman numerals.

Numerical metadata available issues

NOTE

Numerical metadata is information that provides context about the numbers presented. This context helps users understand what the numbers represent and how they should be read. Without these cues, numbers can be ambiguous or misleading, making it harder for users to understand the intended meaning—especially across different regions, disciplines, or assistive technologies.

EXAMPLE

- Dates include markers for day and month such as “3 May 2025” instead of the ambiguous “03/05/2025.”
- Temperatures specify degrees Celsius or Fahrenheit such as “0° C” or “32° F.”
- Times specify am, pm, or 24-hour clock such as “1 pm” or “13:00.”
- Roman numerals include a programmatic equivalent, such as an aria-label of “Henry the Fourth” for Henry IV, or an inline alternative, such as Super Bowl LIX (59).

▶ **Tests**

§ Guideline 2.2.3 Clear language DEVELOPING

Users can understand the content without having to process complex or unclear language.

EDITOR'S NOTE
This guideline will include exceptions for poetic, scriptural, artistic, and other content whose main goal is expressive rather than informative.

EDITOR'S NOTE
See also: Structure as these guidelines are closely related.

EDITOR'S NOTE

To ensure this guideline works well across different languages, members of AGWG, Cognitive and Learning Disabilities Accessibility Task Force (COGA), and internationalization (i18n) agreed on an initial set of languages to pressure-test the guidance.

The five “guardrail” languages are:

- Arabic
- English
- Hindi
- Mandarin
- Russian

We started with the six official languages of the United Nations (UN). Then we removed French and Spanish because they are similar to English. We added Hindi because it is the most commonly spoken language that is not on the UN list.

The group of five languages includes a wide variety of language features, such as:

- Right-to-left text layout
- Vertical text layout
- Tonal sounds that affect meaning

This list doesn’t include every language, but it helps keep the work manageable while making the guidance more useful for a wide audience.

We will work with W3C’s Global Inclusion community group, the Internationalization (i18n) task force, and others to review and refine the testing and techniques for these requirements. We also plan to create guidance for translating the guidelines into more languages in the future.

§ **Core requirement: Abbreviations explained** DEVELOPING

Explanations of abbreviations are available when first used.

Abbreviations explained issues

Except when

- the abbreviation is:

- used so often it has become a word with its own dictionary entry, such as “scuba,” “info,” and “HTML”
- used in a logo
- included in a longer phrase, such as “brand DNA,” whose meaning needs to be defined to meet the non-literal language requirement.

EXAMPLE
(Pass) ‘He has Avoidant/Restrictive Food Intake Disorder (ARFID).’

(Pass) ‘He has [ARFID].’ — (link to a glossary or tooltip.)

(Fail) ‘He has ARFID.’ — (first reference, no explanation is available.)

▶ **Tests**

§ **Supplemental requirement: Non-literal language explained** DEVELOPING

Explanations or unambiguous alternatives are available in text content for non-literal language, such as idioms and metaphors.

Non-literal language explained issues

Except when

text content is:

- poetic,
- scriptural,
- artistic, or
- expressive rather than informational.

EDITOR'S NOTE
Translation software and other tools can aid content authors in identifying non-literal language.

▶ **Tests**

§ **Core requirement: Diacritics available** DEVELOPING

Diacritics required to identify the correct meaning of each word are available.

Diacritics available issues

Applies when

- a human language has a version that removes diacritics for proficient readers.

NOTE 1

A diacritic is a small mark that is added to a letter or character that changes how it is pronounced or what it means. Diacritics may appear above, below, within, or between letters or characters.

NOTE 2

Hebrew and Arabic are examples of human languages that omit diacritics for proficient readers.

EXAMPLE 1

The Hebrew word אמר, with three consonants and no diacritics

- Without diacritics to represent the vowels, אמר can represent different verb and noun forms of “to say.”
- Readers must rely on context to determine the intended pronunciation and meaning.

EXAMPLE 2

The Hebrew word אָמַר, with diacritics under two of the three consonants

- A diacritic (whose unicode looks similar to an uppercase T) under the א makes an “ah” sound.
- A horizontal mark under the מ makes a short “a” sound.
- Pronunciation: ah-mar
- Meaning: “He said”

EXAMPLE 3

The Hebrew word יֹאמַר, with diacritics in front of the first consonant and under another

- A mark (that looks similar to a floating comma) in front of the word indicates future tense, masculine.
- A dot above the prefix makes an “o” sound.
- A horizontal mark under the מ makes a short “a” sound.
- Pronunciation: yo-mar
- Meaning: “He will say”

EXAMPLE 4

The Hebrew word אֹמֶר, with a different set of diacritics above one consonant and under another

- A dot above the first letter makes an “o” sound.
- Three dots forming a small triangle under the מ make a short “e” sound.
- Pronunciation: oh-mer
- Meaning: “utterance; saying” (noun)

▶ **Tests**

§ **Supplemental requirement: No nested clauses** DEVELOPING

Sentences do not include nested clauses.

No nested clauses issues

Except when

- text content:
  - is poetic, scriptural, artistic, or expressive rather than informational, or
  - provides legal information.

EXAMPLE
(Fail) ‘The teacher was surprised that the student who did well on yesterday’s test was absent today.’

(Pass) ‘The teacher was surprised that the student was absent today. This was unexpected because he did well on yesterday’s test.’

▶ **Tests**

§ **Supplemental requirement: No unnecessary words** DEVELOPING

Sentences do not include unnecessary words.

No unnecessary words issues

Except when

- text content is:
  - poetic,
  - scriptural,
  - artistic, or
  - expressive rather than informational.

EXAMPLE

- redundant modifiers such as ‘free gift’ and ‘past history’
- double negatives to express a positive such as ‘She didn’t see nothing’ (which can be shortened to ‘She saw nothing’)
- indirect phrasing such as ‘at this point in time’ (which can be shortened to ‘now’)
- noun versions of verbs, such as ‘take into consideration’ (which can be shortened to ‘consider’)

NOTE
Automated tools can help content authors identify unnecessary words in many languages, including Arabic, English, Hindi, Mandarin, and Russian.

▶ **Tests**

§ **Assertion: Clear language review** DEVELOPING

[Title, role, or organization] asserts that:

Clear language review issues

- Our organization has a process and policy to review text content for clear language before publication. The process includes confirming:
  - All of the core requirements in the ‘Clear Language’ guideline are met.
  - Verb tense is chosen for ease of understanding.
  - Content uses short paragraphs.
  - Paragraphs that convey information begin with a sentence stating the main point or purpose (often called a topic sentence).
  - If a style guide is used by content authors, it must provide guidance on these aspects of clear language.

    - If author training is provided, it must provide guidance on these aspects of clear language.

Information that needs to be included publicly:

- Title, role, or organization making the assertion (if different from the conformance claim).
- Date of when the policy was implemented.
- Date of assertion (if different from the date of the conformance claim).

Recommended internal documentation (Informative):

- Copy of the policy implementing the clear language review.
- Date author training was provided (if any).
- Number or proportion of authors who completed the training.
- Copy of the style guide (if any) where clear language review has been defined.

§ **Assertion: Visual aids review** DEVELOPING

[Title, role, or organization] asserts that:

Visual aids review issues

- Our organization has a process and policy for reviewing text content to identify complex ideas such as processes, workflows, relationships, or chronological information, and adding visual aids to help readers understand them.

Information that needs to be included publicly:

- Title, role or organization making the assertion (if different from the conformance claim)
- Date of when the policy was implemented
- Date of assertion (if different from the date of the conformance claim)

Recommended internal documentation (Informative):

- Copy of the policy implementing the use of visual aids
- Date author training was provided (if any)
- Number or proportion of authors who completed the training
- Copy of the style guide (if any) where visual aids have been defined

§ 2.3 Interactive components

§ Guideline 2.3.1 Keyboard focus appearance DEVELOPING

Users can see which element has keyboard focus.

How to meet Keyboard focus appearance

§ Supplemental requirement: Default focus indicator used DEVELOPING

The focusable item uses the user agent default focus indicator.

Default focus indicator used issues

▶ Tests

§ Core requirement: Focus indicator contrast sufficient DEVELOPING

If a custom focus indicator is used, it has sufficient adjacent contrast and change of contrast.

Focus indicator contrast sufficient issues

Applies when

- the user agent's default focus indicator is replaced by a custom focus indicator.

▶ Tests

§ **Supplemental requirement: Focus indicator size sufficient** DEVELOPING

If a custom focus indicator is used, it has sufficient size and adjacency.

Focus indicator size sufficient issues

Applies when

- the user agent's default focus indicator is replaced by a custom focus indicator.

▶ **Tests**

§ **Assertion: Focus indicator style guide** DEVELOPING

[Title, role, or organization] asserts that:

Focus indicator style guide issues

- Our organization has a style guide that includes guidance on focus indicators and a policy and/or processes that the style guide must be followed.

Information that needs to be included publicly:

- Title, role or organization making the assertion (if different from the conformance claim).
- Date of when the design style guide was published.
- Date of assertion (if different from the date of the conformance claim).

Recommended internal documentation (Informative):

- Copy of the policy implementing the use of design style guide on focus style.
- Whether training was provided for authors
  - Date training was provided.
  - Number or proportion of authors who completed the training.
- A copy of the design style guide (if any) where focus style has been defined.

§ Guideline 2.3.2 Pointer focus appearance

Users can see the location of the pointer focus.

§ Core requirement: Pointer activation indicated (minimum) DEVELOPING

There is a visible indication of the activation of an interactive element when selected by the pointer.

Pointer activation indicated (minimum) issues

Applies when

- The platform does not use a visible pointer indicator.

NOTE
This is primarily aimed at touch-interfaces and VR where you don't have a pointer indicator, but do need to know when something has been selected.

▶ **Tests**

§ Supplemental requirement: Pointer activation indicated (enhanced) DEVELOPING

The user can choose to always have a visible pointer indicator.

Pointer activation indicated (enhanced) issues

Applies when

- The platform does not use a visible pointer indicator.

NOTE
This is primarily aimed at eye-tracking and touch-screens, where it is useful for the user to be able to have a visible indicator, but it wouldn't be universal (i.e., it might get in the way for some users).

▶ **Tests**

§ **Core requirement:** **Pointer contrast sufficient** DEVELOPING

The default pointer meets the @@[non-text-contrast] requirement, and is at least as large as the platform default.

Pointer contrast sufficient issues

NOTE

There can be multiple types of pointer indicator (e.g. arrow, hand, caret). The size requirement applies to whichever type of indicator would be the default for that scenario.

Applies when

- the pointer indicator appearance can be adjusted from the platform default.

▶ **Tests**

§ **Supplemental requirement:** **Default pointer used** DEVELOPING

The user can ensure that the appearance of the pointer is not overridden by the authored interface.

Default pointer used issues

Except when

- Changing the pointer appearance is essential.

EDITOR'S NOTE

Methods & best practices:

- No scripting or styling overrides the pointer indicator appearance.
- A setting is provided so that the pointer indicator appearance is not overridden.

▶ **Tests**

§ **Core requirement: Pointer focus indicated** DEVELOPING

There is a visible pointer indicator.

Pointer focus indicated issues

Except when

- The pointer is on a video - it can be hidden if there is a pointer mechanism to un-hide the pointer indicator.
- The pointer controls the direction of view in a virtual or real world environment.

NOTE 1

Examples of pointers which do not always show the pointer indicator:

- A touch-screen interface does not need to have an indicator as the pointer is not on an element before activating it.
- An eye-tracking interface highlights the element under the gaze of the user, but otherwise does not have a pointer indicator.
- A game where you use the pointer to move the entire view around a virtual environment.

NOTE 2

Methods & best practices:

- Method: Interactive elements are highlighted when the pointer is on the element. For example, a set of image-links are shown, and the one under the pointer is highlighted with an outline or size-change.

▶ **Tests**

Core requirement: Pointer visible DEVELOPING

§

The pointer indicator is always visible.

Pointer visible issues

Except when

- The pointer is on a video and is not moving.
- The pointer controls the direction of view in a virtual or real world environment.

NOTE
Methods & best practices

- Method: No styling or scripting hides the pointer indicator.

▶ **Tests**

§ **Supplemental requirement: Enhanced pointer available** DEVELOPING

Provide a more visible pointer indicator than the platform default. The enhanced pointer indicator can be enabled in a setting, and be visible temporarily (for a few seconds) or permanently.

Enhanced pointer available issues

NOTE
Methods & best practices:

- Provide a setting to adjust the pointer indicator.

▶ **Tests**

Guideline 2.3.3 Navigating content

Users can determine where they are and move through content (including interactive elements) in a systematic and meaningful way regardless of input or movement method.

Core requirement: Focus relevant DEVELOPING

The focus order does not include hidden, static, or groups of repeated interactive elements.

Focus relevant issues

▶ Tests

Supplemental requirement: Focus retained DEVELOPING

A user can focus on a content “area”, such as a modal or popup, then resume their view of all content using a limited number of steps.

Focus retained issues

▶ Tests

Core requirement: Focus order meaningful DEVELOPING

The keyboard focus moves sequentially through content in an order and way that preserves meaning and operability.

Focus order meaningful issues

NOTE

- The Common Keyboard Navigation Techniques are considered logical by default
- Having the navigation follow a consistent pattern on the page would be an indication of logic (if it is not consistently random).
- A strictly start-to-end order is not required.
- Linear means in a single direction (forward/backward) - and is not required as long the non-linear (x-y) technique is in the Common Keyboard Navigation Techniques or is described on the page or where the user will encounter it prior.

▶ **Tests**

§ Guideline 2.3.4 Expected behavior

Users can interact with interactive elements that behave as expected.

§ Assertion: Consistent interactions DEVELOPING

[Title, role, or organization] asserts that:

Consistent interactions issues

- A review has been conducted and changes made (when necessary) to ensure that Components with the same functionality behave consistently:
  - Components that perform the same function behave in the same way and use the same visual indicators.
  - Within the component, interactive elements with the same function have consistent labels.
  - Standard user interface designs for the platform are used when applicable.

Information that needs to be included publicly:

- Title, role or organization making the assertion (if different from the conformance claim).

- Date of when the review was completed.
- Date of assertion (if different from the date of the conformance claim).

Recommended internal documentation (Informative):

- Results from the review.
- Process or policy for maintaining the review.

§ **Assertion: Conventional pattern used** DEVELOPING

[Title, role, or organization] asserts that:

Conventional pattern used issues

- A review has been conducted and changes made (when necessary) to ensure that Components follow established conventions:
  - Each component follows applicable platform conventions for how users interact with that type of component.
  - If a component library is used, then each component in the library:
    - is tested for accessibility before being used
    - follows applicable platform conventions for how users interact with that type of component

Information that needs to be included publicly:

- Title, role or organization making the assertion (if different from the conformance claim).
- Date of when the review was completed.
- Date of assertion (if different from the date of the conformance claim).

Recommended internal documentation (Informative):

- Results from the review.
- Process or policy for maintaining the review.

NOTE

A component library specifies a set of standardized components used in a product or products. It can include code to use, but at a minimum would define how the component is used and define pointer, keyboard, and assistive technology interactions.

Using a component library can help larger teams and organizations provide a consistent experience.

Examples of component libraries include GDS, Carbon Design System, and Lion.

Examples of platform patterns are ARIA Platform Authoring Guide (web), Apple human interface guidelines, and Android Material Design.

§ Guideline 2.3.5 Control information

Users have information about interactive elements that is identifiable and usable visually and using assistive technology.

§ Core requirement: Interactive element contrast sufficient DEVELOPING

Visual information required to identify interactive elements and states meet a minimum contrast ratio test

Interactive element contrast sufficient issues

Except when

- the interactive element is inactive, or
- when the appearance of the component is determined by the user agent and not modified by the content author.

▶ **Tests**

§ Core requirement: Interactive element names available DEVELOPING

Persistent names (including labels) that identify the purpose of the interactive element are visually and programmatically available.

Interactive element names available issues

NOTE
Visible labels can be text or non-text, for instance icons.

EDITOR'S NOTE
Methods & best practices

- Method: Provide unique, descriptive text or image (icon) for each interactive element.
- Best Practice: Ensure clarity by pairing an icon with a persistent visible text label or only use a persistent visible text label.
- Best Practice: When an icon is the only visual label, provide a tooltip (hover label) with the text description.

▶ **Tests**

§ **Core requirement: Changes to elements notified** DEVELOPING

Changes to interactive elements' names, roles, values or states are visually and programmatically indicated.

Changes to elements notified issues

EDITOR'S NOTE
Methods & best practices:

- Method: Add code that clearly defines the name, role, value and state.
- Method: visually indicate the names, roles and values of the interactive element.
- Method: HTML - html tags or aria

▶ **Tests**

§ **Core requirement: Input constraints used** DEVELOPING

Field constraints and conditions (required line length, date format, password format, etc.) are available.

Input constraints used issues

EDITOR'S NOTE

Methods & best practices

- Method (HTML): to programmatically indicate, use the pattern attribute or write the information in a label; to visually indicate, add the requirements to the page near the input.

Best practice: the constraints remain persistent

▶ **Tests**

§ **Core requirement: Label included in programmatic name** DEVELOPING

The programmatic name includes the visual label.

Label included in programmatic name issues

EDITOR'S NOTE

Methods & best practices

- Method: Code the same name as you display.
- Best Practice: Use unique names and keep the programmatic and visual names the same.

▶ **Tests**

§ **Core requirement: Roles, values, states, properties available** DEVELOPING

Accurate names, roles, values, and states are available for interactive elements.

Roles, values, states, properties available issues

EDITOR'S NOTE

Methods & best practices

- Method (HTML): use HTML elements according to specification.
- Method (ARIA): add roles, values, states, and properties according to specification.

▶ **Tests**

2.4 Input / operation

Guideline 2.4.1 Keyboard interface input

Users can navigate and operate content using only the keyboard.

Core requirement: Keyboard operable DEVELOPING

All components on the page/view that can be operated by pointer, audio (voice or other), gesture, camera, or other means can be operated using keyboard interface only.

Keyboard operable issues

▶ **Tests**

Core requirement: Keyboard accessible DEVELOPING

All content that can be accessed by other input modalities can be accessed using keyboard interface only.

Keyboard accessible issues

NOTE 1
All content includes content made available via hovers, right clicks, etc.

NOTE 2
Other input modalities include pointing devices, voice and speech recognition, gesture, camera, and any other means of input or control.

NOTE 3
The "Keyboard operable" requirement allows you to navigate to all actionable elements, but if the next element is 5 screens down, you also need to be able to access all the content. Also, if the content is in expanding sections, you need to not only open them but also access all of the content, not just its actionable elements.

▶ **Tests**

§ **Core requirement: Bidirectional navigation** DEVELOPING

The keyboard interface can always move forward to the next interactive element and back to the previous interactive element.

Bidirectional navigation issues

NOTE 1
Although keyboard navigation is required to be bidirectional, it is not required that it be symmetrical, even though this is usually best practice.

NOTE 2
Methods & best practices:

- Method: Use standard HTML to create interactive elements.
- Avoid modifying the tab order to be in only one direction.

EXAMPLE

Tabbing through menus, hitting `space` to open a menu, tabbing off the end of the menu lands you on the next menu. A `shift+tab` may go back to the last item in the previous menu (symmetrical) or to the last menu (not symmetrical but also does not fail this requirement).

▶ **Tests**

§ **Supplemental requirement: Custom keys documented** DEVELOPING

Non-standard keyboard commands provided by content authors are documented and that documentation is programmatically and visually available from any page/view to which they apply.

Custom keys documented issues

▶ **Tests**

§ **Core requirement: No keyboard conflicts** DEVELOPING

Keyboard commands provided by content authors do not conflict with standard platform keyboard commands or they can be remapped.

No keyboard conflicts issues

▶ **Tests**

§ **Core requirement: Keyboard navigable if responsive** DEVELOPING

If the page/view uses responsive design, the page / view remains fully keyboard navigable.

Keyboard navigable if responsive issues

▶ **Tests**

§ **Supplemental requirement: Focus placed** DEVELOPING

When keyboard focus moves from one context to another within a page/view, whether automatically or by user request, the keyboard focus is preserved so that, when the user returns to the previous context, the keyboard focus is restored to its previous location unless that location no longer exists.

Focus placed issues

EXAMPLE 1

When a user closes a modal dialog or other popup, keyboard focus is returned to the element that caused the dialog or popup to open.

EXAMPLE 2

When there are tags, with a delete button on each tag, the keyboard focus needs to be moved to a meaningful location after a tag is deleted.

EDITOR'S NOTE

Method: When removing interactive elements such as filters, dialogs, or popups that currently contain focus, actively place the focus back on the element that led to that element, the previous element within the focus order, or another meaningful location.

Best Practice: Conduct usability testing with screen reader users to evaluate the focus movement.

▶ **Tests**

§ **Core requirement: No keyboard traps** DEVELOPING

Components that can be activated or entered using the keyboard interface, can be deactivated or exited using a standard keyboard navigation-operation technique, standard platform

keyboard commands.

No keyboard traps issues

Except when

- The non-standard keyboard navigation technique is described on the page/view or earlier in the process.

EXAMPLE
Examples include:

1. Ensuring that users can exit a modal dialog, menu, or calendar picker after entering it
2. Ensuring that users can deactivate an animation, video, drop down menu, or carousel after activating it.

▶ **Tests**

§ **Core requirement: Focus user-controlled** DEVELOPING

The keyboard focus only moves as a result of user interaction.

Focus user-controlled issues

Except when

- The keyboard focus automatically moves to the next interactive elements in keyboard navigation order on completion of some user action, such when the focus moves between the fields for a time-based one-time password (TOTP) code.
- The keyboard focus results from security or emergency situations, such as warning about an imminent session timeout that would cause the user to lose their work.
- The user is informed of the potential keyboard focus move before it happens and given the chance to avoid the move.

▶ **Tests**

Supplemental requirement: Focus movement relevant DEVELOPING

Except for skip links and other elements that are hidden but specifically added to aid keyboard navigation, tabbing does not move the keyboard focus onto content that was not visible before the tab action.

Focus movement relevant issues

NOTE

Accordions, dropdown menus, and ARIA tab panels are examples of expandable content. According to this requirement, these would not expand simply because they include an element in the tab-order contained in them. They would either not expand or would not have any tab-order elements in them.

EXAMPLE

A menu that expands when you tab to it, but then uses arrow keys to navigate in it would pass. But a menu that expands and then requires you to tab through all the newly-visible elements to navigate past it would fail.

▶ **Tests**

Guideline 2.4.2 Physical or cognitive effort when using keyboard

Users can use keyboard without unnecessary physical or cognitive effort.

Supplemental requirement: Navigation keys described DEVELOPING

If any keyboard action needed to navigate, perceive, and operate the full content of the page/view is not a common keyboard navigation technique, then it is described in the page/view where it is required or on a page/view earlier in the process where it is used.

Navigation keys described issues

NOTE

Any platform-related functions are not the responsibility of the author as long as they are not overridden by the content. Examples:

- `Tab` and `Shift + Tab` to move through elements
- Sticky Keys functionality that allows single key activation of multi-key commands

▶ **Tests**

§ **Supplemental requirement: No repetitive links** DEVELOPING

Repetitive adjacent links that have the same destination are avoided.

No repetitive links issues

EDITOR'S NOTE

Supplemental if applicable to all content, else best practice.

NOTE 1

A common pattern is having a component that includes a linked image and some linked text, where both links go to the same content. Someone using screen reading software can be disoriented from the unnecessary chatter, and a keyboard user has to navigate through more tab stops than should be necessary. Combining adjacent links that go to the same content improves the user experience.

NOTE 2

Methods & best practices

- Method: When repetitive links are used, remove them from the focus and reading order.
- Method: Use a single link instead of multiple links to the same destination.
- Best practice: Combine repetitive links into a single interactive element.

▶ **Tests**

§ **Assertion: Keyboard effort comparable** DEVELOPING

[Title, role, or organization] asserts that:

Keyboard effort comparable issues

- A review has been conducted and changes made (when necessary) to ensure that there is minimal difference between the number of input commands required when using the keyboard interface only and the number of commands when using other input modalities.

NOTE
Other input modalities include pointing devices, voice and speech recognition, gesture, camera, and any other means of input or control.

Information that needs to be included publicly:

- Title, role or organization making the assertion (if different from the conformance claim).
- Date of when the review was conducted.
- Date of assertion (if different from the date of the conformance claim).
- Scope of assertion (if different from the conformance claim).

Recommended internal documentation (Informative):

- Copy of the review implementing the use of user interface design principles that include minimizing the difference between the number of input commands required when using the keyboard interface only and the number of commands when using other input modalities.
- Whether training was provided for authors.
- Date training was provided.

- Number or proportion of authors who completed the training.
- A copy of the user interface design principles document.

§ Guideline 2.4.3 Pointer input

Pointer input is consistent and all functionality can be done with simple pointer input in a time and pressure insensitive way.

§ Core requirement: Pointer activation controllable DEVELOPING

At least one of the following is true for functionality that can be activated using a simple pointer input:

Pointer activation controllable issues

No Down Event

The down event of the pointer is not used to execute any part of the function.

Cancel or Undo

Completion of the function is on the up event and a mechanism is available to cancel the function before completion, or there is a mechanism to undo the function after completion.

Up Reversal

The up event reverses any outcome of the preceding down event.

Except when

- Completing the function on the down event is essential.

> EXAMPLE 1
> An example of Cancel would be dragging where there is a pickup action on button down, but it can be canceled by dropping anywhere other than the drop area.

> EXAMPLE 2
> Examples of places where action on down event may be essential include a descending-price auction or a game trigger.

▶ **Tests**

§ **Core requirement: Simple pointer input available** DEVELOPING

All functionality and content available using complex pointer inputs is also available using a simple pointer input or a sequence of simple pointer inputs that do not require timing.

Simple pointer input available issues

EXAMPLE

Examples of complex pointer inputs:

- Double clicks
- Dragging movements
- Swipe gestures
- Multipoint gestures such as pinching, split tap, or two-finger rotor
- Variable pressure or timing

NOTE 1

Complex pointer inputs are not banned, but they cannot be the only way to accomplish an action.

NOTE 2

Simple pointer input is different than single pointer input and is more restrictive than simply using a single pointer.

▶ **Tests**

§ **Supplemental requirement: Consistent pointer cancellation** DEVELOPING

The method of pointer cancellation is consistent for each type of interaction within a set of pages/views.

Consistent pointer cancellation issues

Except when

- It is essential to be different.

NOTE
Where it is essential to be different, it can be helpful to alert the user.

▶ **Tests**

§ **Core requirement: Pointer pressure not relied on** DEVELOPING

Specific pointer pressure is not the only way of achieving any functionality.

Pointer pressure not relied on issues

Except when

- Specific pressure is essential to the functionality.

EXAMPLE
Specific pressure would be essential to a paintbrush feature or advanced signature verification.

NOTE
Methods & best practices:

Method: When building in functionality that relies on pointer pressure, add a slider or other control that can complete the same functionality.

▶ **Tests**

§ **Core requirement: Pointer speed not relied on** DEVELOPING

Functionality does not rely solely on specific pointer speed.

Pointer speed not relied on issues

Except when

- Specific pointer speed is essential to the functionality.

EXAMPLE
Specific pointer speed would be essential to a paintbrush feature, advanced signature verification, or time-constrained gaming.

▶ **Tests**

§ Guideline 2.4.4 Speech and voice input

Provide alternatives to speech input and facilitate speech control.

§ Core requirement: Speech not relied on DEVELOPING

Content or functionality does not rely on speech alone.

Speech not relied on issues

Except when

- Speech input is essential to the functionality.

EDITOR'S NOTE
Methods & best practices

Method: When speech input is supported, an additional way of providing input is also supported.

▶ **Tests**

§ **Core requirement: Real-time text available** DEVELOPING

A real-time text option is available for real-time bidirectional voice communication.

Real-time text available issues

EDITOR'S NOTE

Methods & best practices

Method: Provide a chat option for any voice communication

▶ **Tests**

§ **Assertion: Generated speech testing** DEVELOPING

[Title, role, or organization] asserts that:

Generated speech testing issues

- We have tested voice input and communication systems with generated speech to ensure the systems work with artificial speech as well as human speech.

Information that needs to be included publicly:

- Title, role or organization making the assertion (if different from the conformance claim).
- Date of when the policy was implemented.
- Date of assertion (if different from the date of the conformance claim).

Recommended internal documentation (Informative):

- Systems tested.
- Accuracy rates.

Guideline 2.4.5 Input operation

Users have the option to use different input techniques and combinations and switch between them.

Core requirement: Hover or focus content dismissible DEVELOPING

A mechanism is available to dismiss content that appears on pointer hover or keyboard focus without moving pointer hover or keyboard focus, unless the additional content does not obscure or replace other content

Hover or focus content dismissible issues

EXAMPLE 1
Examples of additional content controlled by the user agent include browser tooltips created through use of the HTML `title` attribute.

Applies when

- Receiving and then removing pointer hover or keyboard focus triggers additional content to become visible and then hidden, and the visual presentation of the additional content is controlled by the author and not by the user agent.

NOTE
This applies to content that appears in addition to the triggering of the interactive element itself. Since hidden interactive elements that are made visible on keyboard focus (such as links used to skip to another part of a page/view) do not present additional content, they are not covered by this requirement.

EXAMPLE 2
Custom tooltips, sub-menus, and other non-modal popups that display on hover and keyboard focus.

▶ **Tests**

§ **Core requirement: Hover content persistent** DEVELOPING

If pointer hover can trigger content, then the pointer can be moved over the additional content without the additional content disappearing.

Hover content persistent issues

EXAMPLE 1
Examples of additional content controlled by the user agent include browser tooltips created through use of the HTML `title` attribute.

Applies when

- Receiving and then removing pointer hover or keyboard focus triggers additional content to become visible and then hidden, and the visual presentation of the additional content is controlled by the author and not by the user agent.

NOTE
This applies to content that appears in addition to the triggering of the interactive element itself. Since hidden interactive elements that are made visible on keyboard focus (such as links used to skip to another part of a page/view) do not present additional content, they are not covered by this requirement.

EXAMPLE 2
Custom tooltips, sub-menus, and other non-modal popups that display on hover and keyboard focus.

▶ **Tests**

§ **Supplemental requirement: Hover or focus content persistent** DEVELOPING

Content that appears on pointer hover or keyboard focus remains visible until the hover or keyboard focus trigger is removed, the user dismisses it, or its information is no longer valid.

Hover or focus content persistent issues

EXAMPLE 1

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

Applies when

- Receiving and then removing pointer hover or keyboard focus triggers additional content to become visible and then hidden, and the visual presentation of the additional content is controlled by the author and not by the user agent.

NOTE

This applies to content that appears in addition to the triggering of the interactive element itself. Since hidden interactive elements that are made visible on keyboard focus (such as links used to skip to another part of a page/view) do not present additional content, they are not covered by this requirement.

EXAMPLE 2

Custom tooltips, sub-menus, and other non-modal popups that display on hover and keyboard focus.

▶ **Tests**

§ **Core requirement: Path-based gesture not relied on** DEVELOPING

Path-based gestures are not the only way of achieving any functionality.

Path-based gesture not relied on issues

Except when

- A path-based gesture is essential to the functionality.

▶ **Tests**

§ **Supplemental requirement: Input method flexible** DEVELOPING

The ability to switch between input methods is available at any time.

Input method flexible issues

NOTE

This does not mean that all input technologies (pointer, keyboard, voice, gesture) need to be supported, but if an input modality is supported, it is supported everywhere in the content except where a particular input method is essential to the functionality.

▶ **Tests**

§ **Core requirement: Body movements not relied on** DEVELOPING

Functionality does not rely solely on full or gross body movement.

Body movements not relied on issues

Except where

- Full or gross body movement is essential to the functionality.

NOTE

This includes both detection of body movement and actions to the device, such as shaking, that require body movement.

▶ **Tests**

§ **Core requirement: Eye tracking not relied on** DEVELOPING

Content and functionality does not rely solely on eye tracking.

Eye tracking not relied on issues

Except when

- Eye tracking is essential.

NOTE 1
This is primarily aimed at ensuring there is an alternative for people who cannot use eye-tracking (but do have sight) due to eye conditions.

NOTE 2
Some platforms may only allow eye tracking. Ideally the platforms allow additional mechanisms for control.

▶ **Tests**

§ **Core requirement: Pointer accessible** DEVELOPING

Pointer selection of elements moves the keyboard focus to that element, even if the user selects an interactive element and drags away from the element without activation.

Pointer accessible issues

Applies when

- Content can interfere with pointer or keyboard focus behavior.

EXAMPLE

A user scrolls a document down six screens, then clicks on a paragraph with their pointer. The user then presses the `Tab` key, which moves the focus to the first interactive component after the position on the screen that was clicked, rather than from the previous position, six screens up the document.

▶ **Tests**

§ Guideline 2.4.6 Authentication

Users have alternative authentication methods available to them.

§ Core requirement: Biometrics not relied on DEVELOPING

Biometric identification is not the only way to identify or authenticate.

Biometrics not relied on issues

NOTE 1

Biometrics includes facial recognition software, fingerprinting, vocal patterns and other voice characteristics.

NOTE 2

Methods & best practices

- Method: When requiring biometric information for authentication, provide an additional way to authenticate that is not biometric. For example, if a finger print is required for authentication, then a password must also be supported.

▶ **Tests**

Core requirement: Voice identification not relied on DEVELOPING

Voice identification is not the only way to identify or authenticate.

Voice identification not relied on issues

▶ Tests

2.5 Error handling

Guideline 2.5.1 Correct errors

Users know about and can correct errors.

Core requirement: Error notifications available DEVELOPING

Errors that are programmatically determined are identified and the problem is described to the user in text.

Error notifications available issues

EXAMPLE

Examples of errors:

- Invalid form input
- Server errors
- Application errors

▶ Tests

Supplemental requirement: Error suggestions provided DEVELOPING

Error messages include suggestions for corrections.

Error suggestions provided issues

Applies when

- Errors require corrections by the user.

Except when

- including suggestions would jeopardize the security or purpose of the content.

▶ **Tests**

§ **Supplemental requirement: Errors indicated in multiple ways** DEVELOPING

Error messages are visually indicated using at least two of the following:

Errors indicated in multiple ways issues

- A symbol that is consistent throughout the conformance scope.
- Color that differentiates the error message from surrounding content.
- Text that clearly indicates the error.

▶ **Tests**

§ **Supplemental requirement: Error messages persistent** DEVELOPING

Error messages persist at least until the error is resolved or the user dismisses them.

Error messages persistent issues

EDITOR'S NOTE

Methods & best practices

- Method: Keep track of the state of the error and make visibility of the error message depending on this state.
- Method: In a form, revalidate all fields when the form is submitted and remove all error messages that are no longer relevant.
- Method: Add a “Dismiss” button to the error that makes the error message disappear.
- Best Practice: [1-2 sentence description or a link to an example]

▶ **Tests**

§ **Core requirement:** **Errors associated** DEVELOPING

When input validation fails, the errors are visually and programmatically associated with the element that caused the error or that can resolve it.

Errors associated issues

EXAMPLE

Examples of failing validation:

- When input is required and the field is left empty
- When the user input does not meet the requested format

▶ **Tests**

§ **Supplemental requirement:** **Error messages collocated** DEVELOPING

Error messages are visually collocated with the error source or the focus is moved to the error message and a mechanism is available to move to the input that is in error.

Error messages collocated issues

Applies when

- Error messages relate to user input.

▶ **Tests**

§ Guideline 2.5.2 Prevent errors

Users can review, confirm and fix information they submit in order to prevent errors.

§ Core requirement: Errors preventable DEVELOPING

Data entry interfaces allow for users to do at least one of the following before submission:

Errors preventable issues

- Review, confirm, and correct all information; or
- Review and correct input errors found during validation.

Except when

- entered data is auto-saved and/or reversible.

EDITOR'S NOTE
Editors are looking at removing the grey area that may exist in this requirement due to interpretations of the word “submission“ (pressing “Submit” in the UI or receiving information server-side).

▶ **Tests**

§ Supplemental requirement: Submission status notified DEVELOPING

Data entry interfaces notify users of submission status at the time of submission.

Submission status notified issues

Applies when

- data submission has succeeded or failed.

▶ **Tests**

§ **Supplemental requirement: Data entry validated** DEVELOPING

Data entered is validated after the user enters data, either:

Data entry validated issues

- when the user leaves the field, or
- when the form is submitted.

▶ **Tests**

§ **Assertion: Error prevention review** DEVELOPING

[Title, role or organization] asserts that:

Error prevention review issues

- For each form in the conformance claim, we have reviewed the form designs to reduce the possibility of users making mistakes.

This review includes checking to make sure that we:

- make the user enter as little information as possible,
- clearly indicate required fields, for long numbers, divide input fields into chunks (supporting autocomplete across fields),
- use an interface where only valid input can be selected,
- use autocomplete and personalization of form controls,
- use common words and metrics or units that users are likely to be familiar with,

- automatically correct input errors when possible and reliable, and
- provide the user with known suggestions and corrections.

Information that needs to be included publicly:

- Title, role or organization making the assertion
- Date of assertion (if different from the date of the conformance claim)

Recommended internal documentation (Informative):

- Documentation of which forms were reviewed
- Documentation of any changes made as a result of the review
- Date of usability testing, if applicable

2.6 Animation and movement

Guideline 2.6.1 Avoid physical harm

Users do not experience physical harm from content.

Core requirement: No flashing DEVELOPING

Content does not include flashing.

No flashing issues

Except when

- The flashing is essential.

EDITOR'S NOTE

Method(s)

- Consider if flashing is essential and, if it is not, refrain from including it.

▶ **Tests**

§ **Supplemental requirement: No flashing (no exceptions)** DEVELOPING

Content does not include flashing.

No flashing (no exceptions) issues

EDITOR'S NOTE

Method(s)

- Design content without using flashing.

▶ **Tests**

§ **Core requirement: No visual motion** DEVELOPING

Content does not include pseudo-motion or visual motion lasting longer than 5 seconds.

No visual motion issues

Except when

- The motion or pseudo-motion is essential.

EDITOR'S NOTE

Method(s)

- Consider if motion or pseudo-motion is essential, and if it is not, refrain from including it.

▶ **Tests**

§ **Supplemental requirement: No visual motion (no exceptions)** DEVELOPING

Content does not include pseudo-motion or visual motion lasting longer than 5 seconds.

No visual motion (no exceptions) issues

EDITOR'S NOTE

Method(s)

- Design content without using visual motion or pseudo-motion.

▶ **Tests**

§ **Core requirement: Trigger warning available** DEVELOPING

A warning is provided before users encounter triggers and a mechanism is available to access the same information without the triggering content.

Trigger warning available issues

Applies when

- triggers are present

NOTE

Note: Triggers are flashing, motion lasting more than 5 seconds, and pseudo-motion.

EDITOR'S NOTE

Method(s)

- Provide a warning for triggers and provide an option without the triggers.

▶ **Tests**

§ **Core requirement: Haptic stimulation adjustable** DEVELOPING

Haptic feedback can be reduced or turned off.

Haptic stimulation adjustable issues

Applies when

- content triggers haptic feedback.

Except when

- the operating system or user agent converts non-haptic feedback to haptics at user request.

EDITOR'S NOTE

Method(s)

- Add a setting to reduce haptic feedback or turn it off.

▶ **Tests**

§ **Core requirement: Audio shifting adjustable** DEVELOPING

Audio shifting designed to create a perception of motion can be paused or turned off.

Audio shifting adjustable issues

Applies when

- content includes audio shifting.

Except when

- operating system or user agent triggers audio shifting.

EDITOR'S NOTE

Method(s)

- Add a setting to pause audio-shifting or turn it off.

▶ **Tests**

§ **Assertion: Safe content review** DEVELOPING

[Title, role, or organization] asserts that:

Safe content review issues

- We have reviewed content for violent, explicit, or troubling content and provided appropriate warnings before accessing the content.

Information that needs to be included publicly:

- Title, role or organization making the assertion (if different from the conformance claim).
- Date of when the review was conducted.
- Date of assertion (if different from the date of the conformance claim).

NOTE

Troubling content will vary based on culture or individual situations, and reviews should take target audiences into consideration.

§ 2.7 Layout

§ Guideline 2.7.1 Recognizable layouts

Users have consistent and recognizable layouts available.

§ **Assertion:** **Conventional layout review** DEVELOPING

[Title, role, or organization] asserts that:

Conventional layout review issues

- We have reviewed layout conventions for similar products or processes.
- The layout used follows a conventional pattern or a tested non-conventional pattern was used.

Information that needs to be included publicly:

- Title, role or organization making the assertion (if different from the conformance claim).
- Date of assertion (if different from the date of the conformance claim).

Recommended internal documentation (Informative):

- Report covering review of layouts for similar products or processes.
- Design log capturing decision to use specific layouts.
- If a non-convention layout is used, usability testing results that demonstrate the utility of the approach taken.

§ **Guideline 2.7.2 User orientation**

Users can determine their location in content both visually and using assistive technologies.

§ **Core requirement:** **Page/view title available** DEVELOPING

Pages/views have a title that describes the name, topic or purpose.

Page/view title available issues

Except when

- The presented content has no way to include a title.

▶ **Tests**

§ Assertion: Location within product review DEVELOPING

[Title, role, or organization] asserts that:

Location within product review issues

- We have reviewed conventions for presenting current location within the conformance scope.
- The presentation of current location within the conformance scope uses appropriate visually and programmatically patterns.

> NOTE
> It is often helpful for users to understand where within a digital product they are. There are many ways to achieve this, for example, a breadcrumb. Ideally this is consistently presented throughout the conformance scope but for some pages/views it may make less sense to include. For example, including a breadcrumb trail on the homepage or on pages that sit outside the hierarchy, for example a shopping cart.

Information that needs to be included publicly:

- Title, role or organization making the assertion (if different from the conformance claim).
- Date of assertion (if different from the date of the conformance claim).

Recommended internal documentation (Informative):

- Design log capturing decision to use specific approaches to current location patterns used within the conformance scope.
- If a non-convention design pattern is used, usability testing results that demonstrate the utility of the design approach taken.

§ Core requirement: All steps listed DEVELOPING

A list of all steps in a multi-step process is visually and programmatically available at each step.

All steps listed issues

Except when

- The total number of steps is unknown, or the sequence of steps depends on user actions.

▶ **Tests**

§ **Core requirement: Current step indicated** DEVELOPING

The current step within a multi-step process is visually and programmatically indicated.

Current step indicated issues

▶ **Tests**

§ **Core requirement: Page/view change notified** DEVELOPING

When content triggers a change of page/view there is a visual change within the view and programmatic notification of the change.

Page/view change notified issues

▶ **Tests**

§ **Supplemental requirement: Return to start supported** DEVELOPING

A visual and programmatically available mechanism exists that allows users to return to the starting point of the conformance scope.

Return to start supported issues

Except when

- The page/view is the starting point of the conformance scope.
- It is essential to the functionality not to provide this mechanism.

NOTE
Where the conformance scope is a sub-part of larger digital product, then the starting point should be the conformance scope starting point. For example, an organization's careers website that is separate from the main website.

▶ **Tests**

§ **Best practice: Return to start prominent** DEVELOPING

Mechanisms that return the user to the starting point of the conformance scope are available in prominent positions both programmatically and visually.

Return to start prominent issues

NOTE
For HTML, a good programmatic positioning of such a mechanism would be early in the DOM.

EXAMPLE

- Common design practices place the homepage link for websites in the top-left corner on the site logo.
- Common design patterns for mobile applications is to include a sticky navigation at the bottom of the viewport which often includes a control to return to the starting point.

§ **Guideline 2.7.3 Structure**

Users can understand and navigate through the content using structure.

EDITOR'S NOTE
See also: Clear Language as these guidelines are closely related.

§ **Supplemental requirement: Relationships detectable** DEVELOPING

Relationships of meaning between elements are conveyed programmatically.

Relationships detectable issues

▶ **Tests**

§ **Core requirement: Blocks of content available (minimum)** DEVELOPING

Meaningful blocks of content are programmatically determinable and visually presented with sufficient surrounding space.

Blocks of content available (minimum) issues

▶ **Tests**

§ **Core requirement: Sections labeled** DEVELOPING

Meaningful blocks of content have a semantically appropriate label that defines their purpose.

Sections labeled issues

Except when

- A label is not needed to understand the purpose of the content within the context of use.

EDITOR'S NOTE

- Add example(s) of labels and heading usage

▶ **Tests**

§ **Supplemental requirement: Heading structure available** DEVELOPING

Meaningful blocks of content are organized with a logical hierarchy of headings.

Heading structure available issues

Except when

- The technology does not support heading levels.

▶ **Tests**

§ **Core requirement: Order detectable** DEVELOPING

Ordered content includes programmatically determinable markers that indicate the position of each item.

Order detectable issues

Except when

- The nature of the ordering of the content is presented immediately prior.

NOTE
This includes lists and processes

▶ **Tests**

§ **Supplemental requirement: Blocks of content available (enhanced)** DEVELOPING

Styling is used to enhance the visual separation between meaningful blocks of content.

Blocks of content available (enhanced) issues

▶ **Tests**

Assertion: Clear structure review DEVELOPING

[Title, role, or organization] asserts that:

Clear structure review issues

- Our organization has a process and policy for reviewing written content for clear structure before publication. The process includes confirming:
  - all of the core requirements in the Structure guideline are met
  - content sections are as concise as possible
  - icons are considered as possible ways to help users understand the content structure and identify key parts, and
  - authors consider when to turn sentences into lists to make the information easier to understand and remember
- If a style guide is used by authors, it must provide guidance on these aspects of clear structure.
- If author training is provided, it must provide guidance on these aspects of clear structure.

Information that needs to be included publicly:

- Title, role or organization making the assertion (if different from the conformance claim).
- Date of when the policy was implemented.
- Date of assertion (if different from the date of the conformance claim).

Recommended internal documentation (Informative):

- Copy of the policy implementing the clear structure review.
- Whether training was provided for authors
  - Date training was provided.
  - Number or proportion of authors who completed the training.
- A copy of the style guide (if any) where clear structure review has been defined.

§ **Assertion: Key information usability testing** DEVELOPING

[Title, role or organization] asserts that:

Key information usability testing issues

- We conducted usability testing to ensure that a diverse group of users, including people with cognitive and mental health challenges, understand the site's information hierarchy and menu organization and are able to find key information.

Information that needs to be included publicly:

- Title, role or organization making the assertion
- Date of assertion
- Date of the testing

Recommended internal documentation (Informative):

- Documentation of usability testing and results
- Scope of testing
- Number of participants and disabilities represented
- Documentation of changes made as a result of usability findings

§ Guideline 2.7.4 No obstruction

Users can perceive and operate user interface components and navigation without obstruction.

§ **Core requirement: Overlay content dismissible** DEVELOPING

When new content becomes visible and covers the main content, a mechanism is available to dismiss the new content.

Overlay content dismissible issues

EDITOR'S NOTE
Methods

- The ‘Escape’ (`Esc`) key closes new content.
- An accessible ‘close’ button is provided to close the new content.

▶ **Tests**

2.8 Consistency across views

Guideline 2.8.1 Consistency

Users have consistent and alternative methods for navigation.

Supplemental requirement: Consistent structural order DEVELOPING

The relative order of structural components remains consistent throughout each variation of pages/views in the conformance scope.

Consistent structural order issues

Applies when

- In a set of pages/views

NOTE
Relative order means that content can be added or removed, but repeated items are in the same order relative to each other.

▶ **Tests**

§ **Supplemental requirement: Consistent navigation order** DEVELOPING

The relative order of navigation items is consistent within blocks of navigation that are repeated on multiple pages/views of the conformance scope or process.

Consistent navigation order issues

Applies when

- In a set of pages/views

EDITOR'S NOTE
This relates to consistency and terminology within blocks of navigation. The consistent ordering of blocks of navigation within a page/view is covered by ‘Consistent relative order’.

▶ **Tests**

§ **Supplemental requirement: Consistent navigation labels** DEVELOPING

The labelling of navigation items within blocks of navigation that are repeated on multiple pages/views in the conformance scope or process is consistent.

Consistent navigation labels issues

Except when

- Labels for navigation items that are marked as ‘current’ within the conformance scope or process.

EDITOR'S NOTE
This relates to consistency and terminology within blocks of navigation. The consistent ordering of blocks of navigation within a page/view is covered by ‘Consistent relative order’.

▶ **Tests**

2.9 Process and task completion

Guideline 2.9.1 Avoid exclusionary cognitive tasks

Users can complete tasks without needing to memorize nor complete advanced cognitive tasks.

Core requirement: Automated entry allowed DEVELOPING

Automated input of personal information from user agents, third-party tools, or paste is not prevented.

Automated entry allowed issues

NOTE

Personal information includes names, passwords, et cetera.

▶ **Tests**

Core requirement: Cognitive test alternatives available DEVELOPING

Processes, including authentication, can be completed without a cognitive function test.

Cognitive test alternatives available issues

▶ **Tests**

Guideline 2.9.2 Adequate time

Users have enough time to read and use content.

Supplemental requirement: Timeout adjustable DEVELOPING

A mechanism exists to extend the time limit before timeout, or the time limit can be disabled.

Timeout adjustable issues

Applies when

- Time limit(s) exist.

Except when

- The time limit is essential.

▶ **Tests**

§ **Supplemental requirement: No time limits** DEVELOPING

The completion of a process does not include time limits.

No time limits issues

Except when

- the time limit is essential, such as in an auction or timed exam.

NOTE
Implying to a user that they will lose a benefit if they don't act immediately is not an essential time limit.

▶ **Tests**

§ **Assertion: No unnecessary time limits** DEVELOPING

[Title, role, or organization] asserts that:

No unnecessary time limits issues

- Products and processes in scope have no non-essential time limits for engagement or completion.

Information that needs to be included publicly:

- Title, role or organization making the assertion (if different from the conformance claim).
- Date of assertion.

§ **Supplemental requirement: Time limits conveyed** DEVELOPING

Users are informed at the start of the process or session that a time limit exists, its length, and that it can be adjusted.

Time limits conveyed issues

Applies when

- pages/views within the conformance scope have a time limit

Except when

- hiding the existence of the time limit is essential

▶ **Tests**

§ **Guideline 2.9.3 Avoid deception**

Users do not encounter deception when completing tasks.

§ **Core requirement: Preselections visible** DEVELOPING

During the completion of a process, preselected options that impact finance, privacy or safety are visibly and programmatically available to the user, by default.

Preselections visible issues

▶ **Tests**

§ **Assertion: Deceptive practices usability testing** DEVELOPING

[Title, role, or organization] asserts that:

Deceptive practices usability testing issues

- We conducted a usability test that included participants with cognitive disabilities and/or mental health based disabilities to evaluate content for misleading wording, artificial pressure, misdirection, and other deceptive practices
- We removed any deceptive practices found.

Information that needs to be included publicly:

- Title, role or organization making the assertion (if different from the conformance claim).
- Date of assertion.

Recommended internal documentation (Informative):

- Number of participants.
- The deceptive practices evaluated.
- Maintain records of usability testing protocol, and results,
- Disabilities represented within the group of participants.

§ **Assertion: Deceptive messaging expert review** DEVELOPING

[Title, role, or organization] asserts that:

Deceptive messaging expert review issues

- We conducted an expert review to evaluate content for misleading wording, artificial pressure, misdirection, and other deceptive practices
- We removed any deceptive practices found.

Information that needs to be included publicly:

- Title, role or organization making the assertion.

- Date of assertion.

Recommended internal documentation (Informative):

- The deceptive practices evaluated.
- Maintain records of deceptive practices found, and resolutions.

§ Guideline 2.9.4 Retain information

Users do not have to reenter information or redo work.

§ Supplemental requirement: Going back supported DEVELOPING

In a multi-step process, the interface supports stepping backwards in a process and returning to the current point without data loss.

Going back supported issues

Except when

- It is essential that the user cannot step back in a process.

> EXAMPLE
> Certain tests in education may require that the student cannot go back through previously-submitted responses and change them.

▶ **Tests**

§ Supplemental requirement: No redundant entry DEVELOPING

Information previously entered by or provided to the user that is required to be entered again in the same process is either auto-populated, or available for the user to select.

No redundant entry issues

Except when

- re-entering the information is essential,
- the information is required to ensure the security of the content, or
- previously entered information is no longer valid.

▶ **Tests**

§ **Supplemental requirement: Progress saved** DEVELOPING

Data entry and other task completion processes allow saving and resuming from the current step in the task.

Progress saved issues

Except when

- The task completion is part of a real-time event (for example, an auction or concert ticket purchase), and no alternative to the time limit is possible.

▶ **Tests**

§ Guideline 2.9.5 Complete tasks

Users understand how to complete tasks.

§ **Supplemental requirement: Required action available** DEVELOPING

The interface indicates when user input or action is required in order to proceed to the next step.

Required action available issues

Applies when

- The user needs to complete an action in order to proceed to the next step. For example: the user needs to agree to the terms and conditions before they can submit the form.

▶ **Tests**

§ **Supplemental requirement: Information requirements available at start** DEVELOPING

Information and resources that are needed to complete a multi-step process are provided at the start of the process, including the:

Information requirements available at start issues

- number of steps it might take (if known in advance),
- details of any resources needed to perform the task, and
- overview of the process and next step.

Applies when

- the user needs to complete a multi-step process.

▶ **Tests**

§ **Supplemental requirement: Process instructions available** DEVELOPING

The instructions needed to complete a multi-step process are available.

Process instructions available issues

Applies when

- The user needs to complete a multi-step process.

▶ **Tests**

§ **Guideline 2.9.6 Unnecessary steps**

Users can complete tasks without unnecessary steps.

§ **Assertion:** **Usability testing for unnecessary steps** DEVELOPING

[Title, role, or organization] asserts that:

Usability testing for unnecessary steps issues

- Usability testing has been conducted to review for unnecessary steps in the process or unnecessary information being requested.
- The sample of test participants included people with cognitive disabilities and/or mental health based disabilities.

Information that needs to be included publicly:

- Title, role or organization making the assertion.
- Date of assertion.

Recommended internal documentation (Informative):

- Maintain records of usability testing protocol, scope of the testing, and results.
- Number of participants and disabilities represented within the group of participants.
- Record of actions taken to address identified issues.

§ 2.10 Policy and protection

§ Guideline 2.10.1 Risk

Users understand the benefits, risks, and consequences of options they select.

§ **Core requirement:** **Consequences of choices explained** DEVELOPING

Choices with legal, financial, privacy, or security consequences are accompanied by a description of the benefits, risks, and potential consequences when users make the choice.

Consequences of choices explained issues

▶ **Tests**

§ **Supplemental requirement: Consequences explained before agreement** DEVELOPING

Legal, financial, privacy, and security consequences are provided before finalizing an agreement.

Consequences explained before agreement issues

Applies when

- Entering an agreement is required.

▶ **Tests**

§ **Assertion: Diverse disabilities considered** DEVELOPING

[Title, role, or organization] asserts that:

Diverse disabilities considered issues

- When considering the safety and security of our users, we considered use cases of people with diverse disabilities. For example, people with cognitive or learning disabilities are sometimes targeted on social media for sexual exploitation and other bad intent.
- Research has been conducted on risks to safety, wellbeing, and mental health for users with diverse disabilities and, when risks are found, all reasonable practical steps have been identified and taken to mitigate the risk.

Information that needs to be included publicly:

- Title, role or organization making the assertion
- Date of assertion

Recommended internal documentation (Informative):

- List of steps that have been taken

- List of use cases used

§ **Assertion: Algorithm inclusivity review** DEVELOPING

[Title, role, or organization] asserts that:

Algorithm inclusivity review issues

- We have a policy and process to review — and have reviewed — the data set, results, and/or algorithm in order to minimize the possibility that algorithms are disadvantageous for people with disabilities.

Information that needs to be included publicly:

- Title, role or organization making the assertion
- Date of assertion
- Disabilities considered
- What was reviewed (data set, results, or algorithm itself)
- Review scope

Recommended internal documentation (Informative):

- Results
- Copy of any process and policy
- Copy of usability testing, if conducted
- Steps taken to resolve the issues found

§ **Guideline 2.10.2 Algorithms**

Users are not disadvantaged or harmed by algorithms.

§ **Assertion: Inclusive data set** DEVELOPING

[Title, role, or organization] asserts that:

Inclusive data set issues

- Content author(s) train AI models using representative and unbiased disability-related information that is proportional to the general population.

§ **Assertion:** **No harm from algorithms** DEVELOPING

[Title, role, or organization] asserts that:

No harm from algorithms issues

- Content author(s) conduct usability testing and ethics reviews to minimize the possibility that algorithms disadvantage people with disabilities.

§ 2.11 Help and feedback

§ Guideline 2.11.1 Help available

Users have help available.

§ **Supplemental requirement:** **Consistent help available** DEVELOPING

Help is labeled consistently and is available in a consistent location relative to other content.

Consistent help available issues

Applies when

- human contact information, a human contact mechanism, a self-help option, or a fully automated contact mechanism is available.

▶ **Tests**

§ **Supplemental requirement:** **Contextual help available** DEVELOPING

Context-sensitive help is available.

Contextual help available issues

▶ **Tests**

§ **Supplemental requirement: Disabled controls explained** DEVELOPING

Information explaining why a visible interactive element is disabled is available and, if the user can take action(s) to enable the element, those action(s) are described.

Disabled controls explained issues

EXAMPLE

When a submit button is disabled until all required fields are filled in, explain that this is the case.

▶ **Tests**

§ **Core requirement: Sensory characteristics not relied on** DEVELOPING

Instructions and help do not rely on sensory characteristics such as shape, color, size, visual location, orientation, or sound.

Sensory characteristics not relied on issues

EXAMPLE

Sensory characteristics include shape, color, size, visual location, orientation, or sound.

▶ **Tests**

§ **Assertion: Supported decision-making review** DEVELOPING

[Title, role, or organization] asserts that:

Supported decision-making review issues

- We have conducted a review to identify when users need to make substantial decisions about money, privacy, or well-being. In these situations, additional support was provided such as:
  - a clear layout of options advantages and disadvantages
  - aids for comprehension such as icons and graphics
  - reduced distractions

Information that needs to be included publicly:

- Title, role or organization making the assertion (if different from the conformance claim).
- Date of when the policy was implemented.
- Date of assertion (if different from the date of the conformance claim).

Recommended internal documentation (Informative):

- Results of review
- Documentation of decisions and changes made as a result.

§ **Assertion: Help usability testing** DEVELOPING

[Title, role, or organization] asserts that:

Help usability testing issues

- Help and training was added based on usability testing with people with cognitive and mental health disabilities to identify gaps.

Information that needs to be included publicly:

- Title, role or organization making the assertion (if different from the conformance claim).
- Date of when the usability testing was conducted.
- Date of assertion (if different from the date of the conformance claim).

Recommended internal documentation (Informative):

- Usability findings
- Solutions added

> EXAMPLE
> Solutions include help with completing functionality such as data entry, task completion, search, and understanding complex ideas and visualizations.

Guideline 2.11.2 Feedback

Users can provide feedback to content author(s).

Supplemental requirement: Feedback mechanism available DEVELOPING

A mechanism is available to provide feedback to authors.

Feedback mechanism available issues

▶ **Tests**

2.12 User control

Guideline 2.12.1 Assistive technology control

Users can control content settings from their user agents including assistive technology.

Core requirement: Assistive technology supported DEVELOPING

Content can be controlled using assistive and adaptive technology.

Assistive technology supported issues

▶ **Tests**

§ **Core requirement: User settings supported** DEVELOPING

Content responds to users' platform and user agent accessibility-related settings.

User settings supported issues

NOTE
Accessibility-related user settings include font size, icon size, color scheme, magnification, and motion.

▶ **Tests**

§ **Core requirement: Virtual cursor supported** DEVELOPING

Assistive technologies can access content and interactions when using mechanisms that convey alternative points of regard or focus (i.e. virtual cursor).

Virtual cursor supported issues

▶ **Tests**

§ **Supplemental requirement: Notifications adjustable** DEVELOPING

The timing or positioning of notifications can be changed, suppressed or saved.

Notifications adjustable issues

Applies when

- Notifications or other interruptions are present.

Except when

- The notification involves an emergency or is essential.

▶ **Tests**

§ Guideline 2.12.2 Control text

Users can control text presentation.

> EDITOR'S NOTE
> Requirements and assertions for this guideline do not appear here because they have not yet progressed beyond exploratory. See the Editor's Draft for the complete list of potential requirements and assertions.

§ Guideline 2.12.3 Adjustable viewport

Users can adjust the size and orientation of the viewport, adjust text size of content, or zoom content without loss of legibility or functionality.

§ Core requirement: Orientation supported (minimum) DEVELOPING

If the platform has a default orientation, content supports that orientation. If the platform does not have a default orientation, content supports both portrait and landscape orientations.

Orientation supported (minimum) issues

Except when

- Content is aligned with the physical world.
- The orientation is essential.

> NOTE 1
> For extended reality, the platform default orientation aligns with the real world orientation.

> NOTE 2
> Content does not have to re-layout or change aspect ratio in a different orientation, it just needs to display in the device orientation.

▶ **Tests**

§ **Supplemental requirement: Orientation supported (enhanced)** DEVELOPING

Content supports both portrait and landscape orientations.

Orientation supported (enhanced) issues

Except when

- Content is aligned with the physical world.
- The orientation is essential.

▶ **Tests**

§ **Supplemental requirement: Text reflow supported** DEVELOPING

Blocks of text are legible at 320 CSS pixels in the orientation of text, without the need to scroll in the orientation of text.

Text reflow supported issues

Except when

- The meaning of text relies on a two dimensional structure. For example, preformatted text such as code, poems, maps, or comics.

EDITOR'S NOTE

Other languages may have other rules around line breaking: https://r12a.github.io/scripts/script-features/index.html

▶ **Tests**

§ **Supplemental requirement:** **Layout reflow supported** DEVELOPING

All content fits within 320 CSS pixels in the default orientation of text without requiring scrolling in more than one direction. Sections of content within the page/view that scroll in a different direction to the page/view fit within 320 CSS pixels of the page-scrolling direction.

Layout reflow supported issues

Except when

- 2D relationships. For example, tables, electronic program guides.
- Canvases of presentational content. For example, slides.
- Multiple palettes or panels that act on content. For example, Photoshop, IDE.

EDITOR'S NOTE
All block-level elements fit within a 320px inline-size without requiring scrolling in more than one direction.

▶ **Tests**

§ Guideline 2.12.4 Media control

Users can control media and media alternatives.

§ **Core requirement:** **Page/view audio adjustable** DEVELOPING

There are mechanisms to pause, stop, and adjust the volume independently of the overall system volume level, of any automatically playing audio in a page / view.

Page/view audio adjustable issues

Applies when

- Notifications or other interruptions are present.

> NOTE
> Mechanisms include controls on each instance of content, or a single app-wide control that disables audio, for example: app-wide earcons.

▶ **Tests**

§ **Supplemental requirement: Media alternatives searchable** DEVELOPING

Alternatives for media can be searched and queried.

Media alternatives searchable issues

▶ **Tests**

§ **Supplemental requirement: Media alternatives controllable** DEVELOPING

Closed captions and audio descriptions can be turned on and off.

Media alternatives controllable issues

▶ **Tests**

§ Guideline 2.12.5 Content changes

Users know when content changes.

§ **Core requirement: Change of content notified** DEVELOPING

Meaningful changes in visual content are conveyed programmatically.

Change of content notified issues

Applies when

- The changes appear before the user’s current position in reader order.
- The changes appear earlier in the process.
- Notifications, status or error messages appear.
- The amount of content changes.
- The change affects the content’s meaning.
- The audience changes.

Except when

- changes are continuous, without pause.

> EXAMPLE
>
> - A new message is added in a conversation above the current location of focus. A message is provided that programmatically conveys that there is new content above.
> - Filters are adjusted which changes the number of items shown. As the number of items changes, that is provided programmatically as a status update.

▶ **Tests**

§ **Core requirement: Change of focus notified** DEVELOPING

When the focus changes on-focus or automatically, the user is notified visually and programmatically.

Change of focus notified issues

▶ **Tests**

§ **Core requirement: Change of user agent notified** DEVELOPING

When content triggers a device change or an automatic user agent change, the user is notified before the change occurs.

Change of user agent notified issues

▶ **Tests**

§ 3. Conformance EXPLORATORY

This section (with its subsections) provides requirements which must be followed to conform to the specification, meaning it is normative.

As well as sections marked as non-normative, all authoring guidelines, diagrams, examples, and notes in this specification are non-normative. Everything else in this specification is normative.

The key words *MAY*, *MUST*, *MUST NOT*, *NOT RECOMMENDED*, *RECOMMENDED*, *SHOULD*, and *SHOULD NOT* in this document are to be interpreted as described in BCP 14 [*RFC2119*] [*RFC8174*] when, and only when, they appear in all capitals, as shown here.

▶ **Plain language summary of “Interpreting normative provisions”**

This section lists requirements for conformance to WCAG 3.0. It also gives information about how to make conformance claims, which are optional. Finally, it describes what it means to be accessibility supported, since only accessibility-supported ways of using technologies can be relied upon for conformance.

EDITOR'S NOTE: Goals for a new conformance model

In this draft, AG is exploring a model based on Core Requirements, Supplemental Requirements, and Assertions. We ask reviewers to consider the following questions:

- Should the core set of requirements provide approximately the same coverage as WCAG 2.2 A & AA? If not, should it be smaller or larger? If you think it should be a different size, what rules would you use to select the set?
- Should we include assertions within the core set? Would organizations be willing to make public, attributable assertions about processes they completed?
- Should there be additional levels within the core set of requirements? If yes, what and why?
- In WCAG 2, every provision within the minimum conformance level has to be met to conform at the lowest level of conformance. To conform at the lowest level in WCAG 3, every provision within the core set must be met AND then additional effort must be done to better support people with disabilities but WCAG 3 allows some flexibility on what and how that effort is provided. Does this approach work or would a stricter set of requirements with more flexibility within the methods be a preferred approach?
- The proposed conformance model ties conformance to functional performance statements. This list is expanded from the one currently used in legislation. What do you see as the positive and negative aspects of this approach? What statements should be added or changed?

§ 3.1 Interpreting normative provisions

The main content of WCAG 2.2 is normative and defines requirements that impact conformance claims. Introductory material, appendices, sections marked as "non-normative", diagrams, examples, and notes are informative (non-normative). Non-normative material provides advisory information to help interpret the guidelines but does not create requirements that impact a conformance claim.

The key words *MAY*, *MUST*, *MUST NOT*, *NOT RECOMMENDED*, *RECOMMENDED*, *SHOULD*, and *SHOULD NOT* are to be interpreted as described in [*RFC2119*].

§ 3.1.1 Types of provision

WCAG 3 consists of three types of provisions.

Core Requirements — Requirements that must be met in order to conform. These include requirements that ensure:

- content is detectable by user agents and AT,
- content is conveyed to multiple senses and can be used by multiple inputs,
- content does not cause immediate, physical harm, and
- necessary support is available such as captions, text alternatives, and expanded forms of abbreviations.

Supplemental Requirements — Requirements that build on the core set. These include:

- quality checks,
- stricter or higher levels than those in a comparable core requirement,
- requirements that are currently more difficult or expensive to implement, and
- requirements that may apply in certain situations.

Assertions — Documented statements you can make about specific accessibility practices your organization follows to improve the accessibility of your content or product.

§ 3.1.2 Best practices

In addition to requirements and assertions, WCAG 3 includes best practices. These are listed at the provision and method level. They provide important guidance that often saves time and improve accessibility but may not always apply in every situation. These are integrated with requirements, assertions and methods and are clearly marked as best practice.

Best practices do not need to be satisfied to conform.

§ 3.2 Conformance requirements

In order for a web page to conform to WCAG 3.0, all of the following conformance requirements must be satisfied:

§ 3.2.1 Conformance level

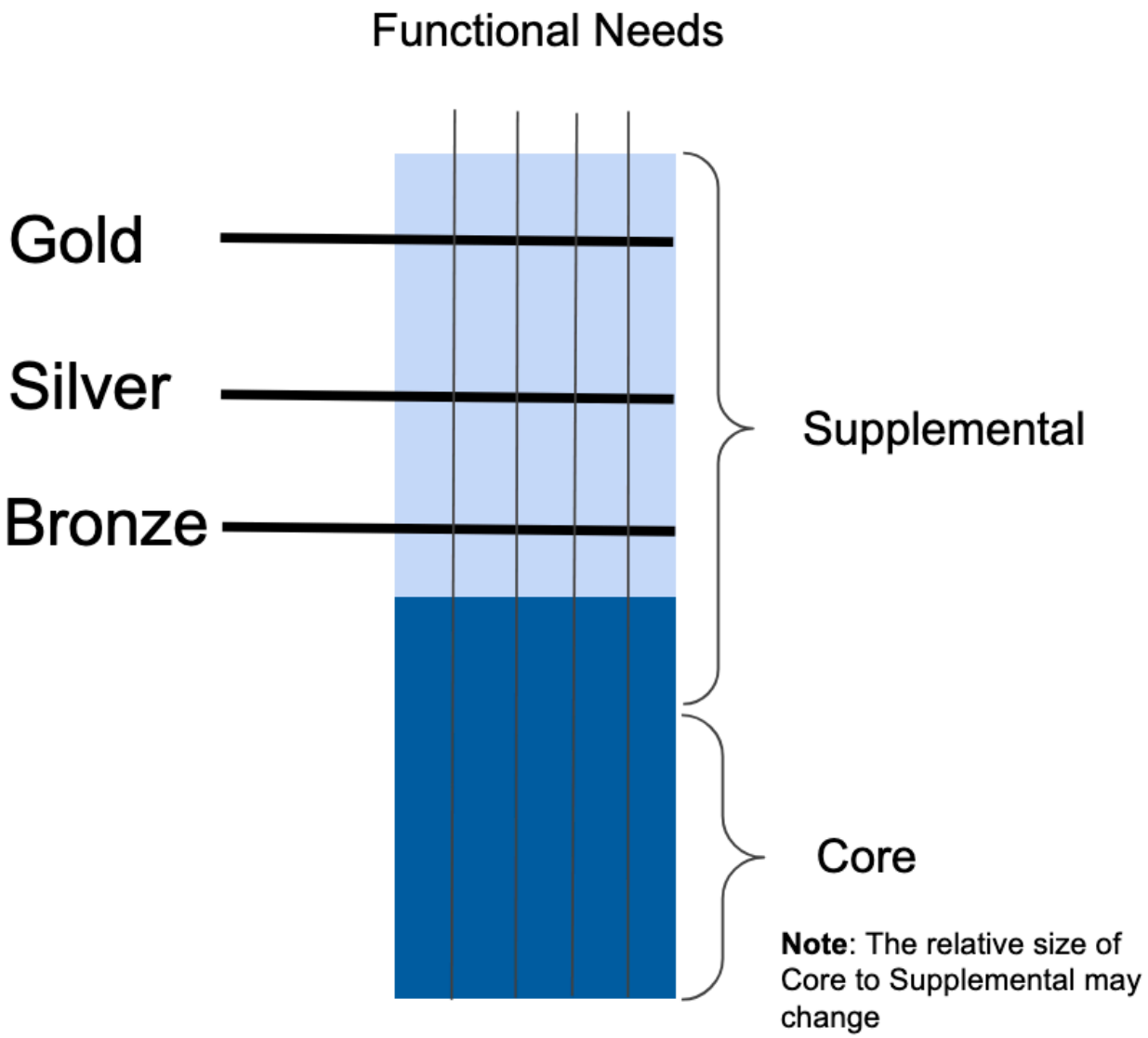
One of the following levels of conformance is met in full.

- For Bronze conformance (the minimum level of conformance), all of the pages/views and processes within the conformance scope *MUST* satisfy all the core requirements and some portion (to be determined) of the supplemental requirements and assertions within each functional performance statement.
- For Silver conformance, all of the pages/views and processes within the conformance scope *MUST* satisfy all the core requirements and a larger portion of the supplemental requirements and assertions than in Bronze within each functional performance statement.
- For Gold conformance, all of the pages/views and processes within the conformance scope satisfy all the core requirements and a larger portion of the supplemental requirements and assertions than in Silver within each functional performance statement.

§ 3.2.2 Accessibility supported

Many methods, such as those using ARIA or CSS, are only effective when they are supported by the user agents (UA) and assistive technologies (AT) of real users.

The UA and AT used vary by language and region, and may be restricted in closed environments such as behind firewalls or on kiosks. As a result, it is important to understand which UA and AT support which methods and clearly state what UA and AT are assumed on conformance statements. The concept that methods must be supported by users' assistive technologies as well as the accessibility features in browsers and other user agents is referred to as "accessibility supported".



§ 3.2.2.1 Accessibility support set

WCAG 3 includes the concept of "accessibility-support sets". This is the set of UAs, ATs, and accessibility settings that AGWG will test against when determining methods that support a requirement.

WCAG 3 will use two accessibility support sets:

1. **Default accessibility-support set** — A set of common browsers and assistive technology and accessibility supported settings support English content. The default accessibility-support set will be for HTML methods. Additional sets may be created for technologies such as mobile or VR.

2. **Alternative accessibility support set** — People, organizations or regional regulators can also specify an alternative accessibility support set. This is a different set of browsers, assistive technology, and accessibility settings that support the intended languages, regions and technologies when the default accessibility-set does not provide that support.

While the AGWG will strive to ensure that requirements have as broad support as possible, it may not be possible for every requirement to be met on every combination of UA and AT. AGWG will only include core requirements if they include at least one HTML method that works with the default accessibility-support set at the time of publication.

The accessibility support set must be included in conformance claims. If you are working in a closed environment, with non-HTML technologies, or are in a region that relies on UA and AT outside the AG default accessibility-support set defined in WCAG 3, we also recommend including your accessibility support set in public accessibility statements. For example, if you claim conformance against a kiosk that uses a non-standard way to provide text to speech, then the accessibility-support set you are using should be included in your conformance claim and we recommend including it in public statements about the kiosk accessibility.

EDITOR'S NOTE

- The default accessibility support set has not yet been defined.
- Informative documentation will be needed to specify how people can create their own alternative accessibility support set.
- Maintenance of the accessibility support set over time needs additional discussion.
- An exception for long-present bugs in assistive technology is still under discussion.

§ 3.2.3 Defined conformance scope

When evaluating the accessibility of content, WCAG 3 requires the guidelines apply to a specific scope.

WCAG 3 defines two ways to scope content: pages/views and processes. Evaluation is done on one or more complete pages/views or processes, and conformance is determined on the basis of one or more complete pages/views or processes.

Conformance is defined only for clearly defined pages/views and processes. However, a conformance claim may be made to cover one pages/views and processes, a series of pages/views and processes, or multiple related pages/views and processes. All unique steps in a process *MUST* be represented in the set of views. Views outside of the process *MAY* also be included in the scope.

> EDITOR'S NOTE
>
> We recognize that representative sampling is an important strategy that large and complex sites use to assess accessibility. While it is not addressed within this document at this time, our intent is to later address it within this document or in a separate document before the guidelines reach the Candidate Recommendation stage. We welcome your suggestions and feedback about the best way to incorporate representative sampling in WCAG 3.

§ 3.3 Functional performance statements (expanded)

Each core requirement, supplemental requirement, and assertion will be tagged with one or more of the expanded functional performance statements below. The statements are intended to describe the functional performance of ICT enabling people to locate, identify, and operate ICT functions, and to access the information provided, regardless of physical, cognitive or sensory abilities.

This is not a list of disabilities. Different disabilities may lead to the need for use with one or more of the limitations listed below. For example, a visual impairment or an inability to process visual information may require usage with limited vision.

- Usage without vision
- Usage with limited vision
- Usage without perception of color
- Usage without hearing
- Usage with limited hearing
- Usage without vocal capability
- Usage with limited vocal capability
- Usage with limited manipulation or strength
- Usage with limited reach
- Usage with sensitivity to sensory stimuli

- Usage with limited executive function (includes self-regulation, task management, planning, organization and social skills)
- Usage with limited reasoning (includes problem solving, abstraction, mathematical intelligence, recognition, and crystallized intelligence)
- Usage with limited attention (includes the ability to concentrate, multitask, and focus over time)
- Use with sensitivity to certain content (includes anxiety and content-based triggers)
- Usage with limited language processing (includes the ability to process spoken language, sign language, written language, and image based language.)
- Usage with limited memory
- Usage with atypical physical attributes

NOTE

The difference between this list and the one currently used in legislation is:

- Limited cognition has been broken apart to better reflect the diversity of needs within cognition-related disabilities.
- An alternative breakdown to usage with limited cognition based on FAST would be:
  - Usage with limited attention
  - Usage with limited language and communication
  - Usage with limited learning
  - Usage with limited memory
  - Usage with limited executive function
- The following functional performance statements were added:
  - Use with limited vocal capability
  - Use with atypical physical attributes
  - Use with sensitivity to certain content

§ 4. Glossary DEVELOPING

This section (with its subsections) provides requirements which must be followed to conform to the specification, meaning it is normative.

> NOTE
>
> Many of the terms defined here have common meanings. When terms appear with a link to the definition, the meaning is as formally defined here. When terms appear without a link to the definition, their meaning is not explicitly related to the formal definition here. These definitions are in progress and may evolve as the document evolves.

> EDITOR'S NOTE
>
> This glossary includes terms used by content that has reached a maturity level of Developing or higher. The definitions themselves include a maturity level and may mature at a different pace than the content that refers to them. The AGWG will work with other task forces and groups to harmonize terminology across documents as much as is possible.

abbreviation DEVELOPING

shortened form of a word, phrase, or name where the abbreviation has not become part of the language

> NOTE 1
>
> This includes initialisms, acronyms, and numeronyms.
>
> 1. **initialisms** are shortened forms of a name or phrase made from the initial letters of words or syllables contained in that name or phrase. These are not defined in all languages.
> 2. **acronyms** are abbreviated forms made from the initial letters or parts of other words (in a name or phrase) which may be pronounced as a word.
> 3. **numeronyms** are shortened forms of a word that use the first and last letters, with a number in between showing the number of letters left out.

EXAMPLE 1

SNCF is a French initialism that includes the initial letters of the Société Nationale des Chemins de Fer, the French national railroad.

EXAMPLE 2

ESP is an initialism for extrasensory perception.

EXAMPLE 3

NOAA is an acronym made from the initial letters of the National Oceanic and Atmospheric Administration in the United States.

EXAMPLE 4

a11y is a numeronym that includes the first and last letter of the word ‘accessibility’ and the number 11 to indicate the 11 letters in between.

EXAMPLE 5

In non-alphabetic writing systems, abbreviated forms are created using different linguistic units, and the alphabetic model does not apply directly. For example, in Japanese, 全国学力・学習状況調査 (National Academic Achievement and Learning Status Survey) is abbreviated as 全国学調 (National Academic Survey).

NOTE 2

Some companies have adopted what used to be an initialism as their company name. In these cases, the new name of the company is the letters (for example, Ecma) and the word is no longer considered an abbreviation.

accessibility support set DEVELOPING

group of user agents and assistive technologies you test with

> EDITOR'S NOTE
>
> The AGWG is considering defining a default set of user agents and assistive technologies that they use when validating guidelines.
>
> Accessibility support sets may vary based on language, region, or situation.
>
> If you are not using the default accessibility set, the conformance report should indicate what set is being used.

accessibility supported DEVELOPING

available and working in the user agents and assistive technology in the accessibility support set

> EDITOR'S NOTE
>
> The working group intended to include a default accessibility support set. See Default accessibility support set #277.

actively available DEVELOPING

available for the user to perceive and use

ASCII art EXPLORATORY

picture created by a spatial arrangement of characters or glyphs (typically from the 95 printable characters defined by ASCII)

assertion DEVELOPING

formal claim of fact, attributed to a person or organization, regarding procedures practiced in the development and maintenance of the content or conformance scope to improve accessibility

assistive technology DEVELOPING

hardware and/or software that acts as a user agent, or along with a mainstream user agent, to provide functionality to meet the requirements of users with disabilities that go beyond those offered by mainstream user agents

> NOTE 1
>
> Functionality provided by assistive technology includes alternative presentations (e.g., as synthesized speech or magnified content), alternative input methods (e.g., voice), additional navigation or orientation mechanisms, and content transformations (e.g., to make tables more accessible).

NOTE 2

Assistive technologies often communicate data and messages with mainstream user agents by using and monitoring APIs.

NOTE 3

The distinction between mainstream user agents and assistive technologies is not absolute. Many mainstream user agents provide some features to assist individuals with disabilities. The basic difference is that mainstream user agents target broad and diverse audiences that usually include people with and without disabilities. Assistive technologies target narrowly defined populations of users with specific disabilities. The assistance provided by an assistive technology is more specific and appropriate to the needs of its target users. The mainstream user agent may provide important functionality to assistive technologies like retrieving web content from program objects or parsing markup into identifiable bundles.

EXAMPLE

Assistive technologies that are important in the context of this document include the following:

- screen magnifiers, and other visual reading assistants, which are used by people with visual, perceptual and physical print disabilities to change text font, size, spacing, color, synchronization with speech, etc. in order to improve the visual readability of rendered text and images;
- screen readers, which are used by people who are blind to read textual information through synthesized speech or braille;
- text-to-speech software, which is used by some people with cognitive, language, and learning disabilities to convert text into synthetic speech;
- speech recognition software, which may be used by people who have some physical disabilities;
- alternative keyboards, which are used by people with certain physical disabilities to simulate the keyboard (including alternate keyboards that use head pointers, single switches, sip/puff and other special input devices.);
- alternative pointing devices, which are used by people with certain physical disabilities to simulate mouse pointing and button activations.

audio DEVELOPING

live or recorded sound signal

audio description DEVELOPING

narration added to the soundtrack to describe important visual details that cannot be understood from the main soundtrack alone

> NOTE
>
> For audiovisual media, audio description provides information about actions, characters, scene changes, on-screen text, and other visual content.
>
> Audio description is also sometimes called “video description”, “described video”, “visual description”, or “descriptive narration”.
>
> In standard audio description, narration is added during existing pauses in dialogue. See also extended audio description.
>
> If all important visual information is already provided in the main audio track, no additional audio description track is necessary.

automated evaluation DEVELOPING

evaluation conducted using software tools, typically evaluating code-level features and applying heuristics for other tests

> NOTE
>
> Automated testing is contrasted with other types of testing that involve human judgement or experience. Semi-automated evaluation allows machines to guide humans to areas that need inspection. The emerging field of testing conducted via machine learning is not included in this definition.

available EXPLORATORY

present and, in the context of alternatives, sufficient to understand the content

biometric DEVELOPING

behavioral or biological characteristics

> NOTE
>
> examples include but are not limited to voice, iris, appearance, fingerprint, face shape, or body movement

blinking DEVELOPING

switching back and forth between two visual states in a way that is meant to draw attention

> NOTE
>
> See also flash. It is possible for something to be large enough and blink brightly enough at the right frequency to be also classified as a flash.

block of text DEVELOPING

more than one sentence of text

captions DEVELOPING

synchronized visual and/or text alternative for both the speech and non-speech audio portion of a work of audiovisual content

> NOTE 1
>
> Closed captions are equivalents that can be turned on and off with some players and can often be read using assistive technology.

> NOTE 2
>
> Open captions are any captions that cannot be turned off in the player. For example, if the captions are visual equivalent images of text embedded in video.

> NOTE 3
>
> Audio descriptions can be, but do not need to be, captioned since they are descriptions of information that is already presented visually.

> NOTE 4
>
> In some countries, captions are called subtitles. The term ‘subtitles’ is often also used to refer to captions that present a translated version of the audio content.

clear language EXPLORATORY

writing in a simple, direct way

> NOTE
>
> Clear language avoids unnecessary complexity and helps make content accessible to everyone, including people with cognitive disabilities, limited reading skills, and limited language proficiency.

cognitive function test DEVELOPING

task that requires the user to remember, manipulate, or transcribe information

> NOTE
>
> Examples include, but are not limited to:
>
> - memorization, such as remembering a username, password, set of characters, images, or patterns. The common identifiers name, email, and phone number are not considered cognitive function tests as they are personal to the user and consistent across websites;
> - transcription, such as typing in characters;
> - use of correct spelling;
> - performance of calculations;
> - solving of puzzles.

color contrast DEVELOPING

relationship of hue, saturation, and lightness values between two colors

common keyboard navigation technique DEVELOPING

keyboard navigation technique that is the same across most or all applications and platforms and can therefore be relied upon by users who need to navigate by keyboard alone

> NOTE
>
> A sufficient listing of common keyboard navigation techniques for use by authors can be found in the Standard Keyboard Navigation & Operation Keys and Techniques

complex numerical information EXPLORATORY

numbers, statistics, or quantitative data that require interpretation, comparison, or multiple steps to understand.

NOTE

This includes mental calculation, proportional reasoning, or comparison against an unstated whole.

complex pointer input DEVELOPING

any pointer input other than a single pointer input

EXAMPLE

- Double clicking
- Multipoint clicking
- Dragging (single or multipoint)
- Gestures (single or multipoint)
- Pinching
- Path-based gestures
- Timing-based gestures

component DEVELOPING

grouping of elements for a distinct function

conformance DEVELOPING

satisfying all the requirements of the guidelines

NOTE

Conformance is an important part of following the guidelines even when not making a formal Conformance Claim

See the Conformance section for more information.

conformance scope DEVELOPING

A set of Views and/or Pages selected to be part of a conformance claim. Where a View or Page is part of a Process, all the Views or Pages in the process must be included.

> NOTE
>
> How a person or organization selects the set is not defined in WCAG3. There maybe informative guidance on selecting a suitable set in future (similar to WCAG-EM), but regional laws or regulations may provide a methodology.

content DEVELOPING
: information, sensory experience and interactions conveyed

content author EXPLORATORY
: person or persons responsible for the content presented, including word choice, formatting, images, video, audio, and other elements

context-sensitive help DEVELOPING
: help text that provides information related to the function currently being performed

contrast ratio test EXPLORATORY
: meeting a sufficient level of contrast between two colors using the relationship of hue, saturation, and lightness values

> EDITOR'S NOTE
>
> The contrast algorithm used in WCAG 3 is yet to be determined.

conversational support EXPLORATORY
: support that is provided in the form of interactive, bi-directional, context-sensitive conversations

css pixel DEVELOPING
: visual angle of about 0.0213 degrees

> NOTE
>
> A CSS pixel is the canonical unit of measure for all lengths and measurements in CSS. This unit is density-independent, and distinct from actual hardware pixels present in a display. User agents and operating systems should ensure that a CSS pixel is set as closely as possible to the CSS Values and Units Module Level 3 reference pixel [css3-values], which takes into account the physical dimensions of the display and the assumed viewing distance (factors that cannot be determined by content authors).

decorative DEVELOPING

serving only an aesthetic purpose, providing no information, and having no functionality

> NOTE
>
> Text is only purely decorative if the words can be rearranged or substituted without changing their purpose.

> EXAMPLE
>
> The cover page of a dictionary has random words in very light text in the background.

default orientation EXPLORATORY

the orientation of view that is preferred or enforced by the platform

> EXAMPLE
>
> Some mobile devices may default to portrait orientation.

default visual presentation EXPLORATORY

formatting of content defined by the author without any modifications or customizations by the end user

deprecate DEVELOPING

declare something outdated and in the process of being phased out, usually in favor of a specified replacement

> NOTE
>
> Deprecated documents are no longer recommended for use and may cease to exist in the future.

descriptive transcript DEVELOPING

a text version of the speech and non-speech audio information and visual information needed to understand the content

down event DEVELOPING

platform event that occurs when the trigger stimulus of a pointer is depressed

> NOTE
>
> The down event may have different names on different platforms, such as “touchstart” or “mousedown”.

equivalent EXPLORATORY

equal, and, in the context of alternatives, includes or conveys the same information as the original.

essential exception DEVELOPING

exception because there is no way to carry out the function without doing it this way or fundamentally changing the functionality

> EXAMPLE 1
>
> Content that must fail since it is an example of failure or a test sample to be used testing a tool.

> EXAMPLE 2
>
> Real-time based exceptions such as:
>
> - purchasing items in limited supply;
> - an auction where the price keeps dropping until the first person hits the “Buy” button;
> - controlling a drone in flight;
> - submitting a proposal by deadline.

> EXAMPLE 3
>
> Actions that cannot be undone, such as purchasing things in an auction.

essential to outcome DEVELOPING

always necessary to achieve the same result

NOTE

If something is essential to the outcome then: If it were removed, the information or functionality of the content would be fundamentally changed, and the information and functionality cannot be achieved in another way that would conform

EXAMPLE

- Content and/or markup that fails because it is an example of failure or a test sample to be used testing a tool.
- Real-time based exceptions such as:
  - purchasing items in limited supply;
  - a dutch auction where the price keeps dropping until the first person hits the BUY button;
  - controlling a drone in flight;
  - submitting a proposal by deadline.
- There can be no ability to undo, such as purchasing things in an auction.

evaluation DEVELOPING

process of examining content for conformance to these guidelines

NOTE

Different approaches to evaluation include automated evaluation, semi-automated evaluation, human evaluation, and usability testing.

extended audio description DEVELOPING

audio description that is added to audiovisual media by pausing the video to allow for additional time to fit in the audio description

NOTE

This technique is only used when the sense of the video would be lost without the additional audio description and the pauses between dialogue or narration are too short.

figure caption DEVELOPING

title, brief explanation, or comment that accompanies a work of visual media and is always visible on the page

flash DEVELOPING

a pair of opposing changes in relative luminance that can cause seizures in some people if it is large enough and in the right frequency range

> NOTE 1
>
> See general flash and red flash thresholds for information about types of flash that are not allowed.

> NOTE 2
>
> See also blinking.

focus indicator DEVELOPING

pixels that are changed to visually indicate when a user interface component is in a focused state

functional need DEVELOPING

statement that describes a specific gap in one's ability, or a specific mismatch between ability and the designed environment or context

general flash and red flash thresholds DEVELOPING

a flash or rapidly-changing image sequence is below the threshold (i.e., content **passes**) if any of the following are true:

- there are no more than three **general flashes** and / or no more than three **red flashes** within any one-second period; or
- the combined area of flashes occurring concurrently occupies no more than a total of .006 steradians within any 10 degree visual field on the screen (25% of any 10 degree visual field on the screen) at typical viewing distance

where:

- A **general flash** is defined as a pair of opposing changes in relative luminance of 10% or more of the maximum relative luminance (1.0) where the relative luminance of the darker image is below 0.80; and where "a pair of opposing changes" is an increase followed by a decrease, or a decrease followed by an increase, and

- A **red flash** is defined as any pair of opposing transitions involving a saturated red

Exception: Flashing that is a fine, balanced, pattern such as white noise or an alternating checkerboard pattern with “squares” smaller than 0.1 degree (of visual field at typical viewing distance) on a side does not violate the thresholds.

NOTE 1

For general software or web content, using a 341 x 256 pixel rectangle anywhere on the displayed screen area when the content is viewed at 1024 x 768 pixels will provide a good estimate of a 10 degree visual field for standard screen sizes and viewing distances (e.g., 15-17 inch screen at 22-26 inches). This resolution of 75 - 85 ppi is known to be lower, and thus more conservative than the nominal CSS pixel resolution of 96 ppi in CSS specifications. Higher resolutions displays showing the same rendering of the content yield smaller and safer images so it is lower resolutions that are used to define the thresholds.

NOTE 2

A transition is the change in relative luminance (or relative luminance/color for red flashing) between adjacent peaks and valleys in a plot of relative luminance (or relative luminance/color for red flashing) measurement against time. A flash consists of two opposing transitions.

NOTE 3

The new working definition in the field for **“pair of opposing transitions involving a saturated red”** (from WCAG 2.2) is a pair of opposing transitions where, one transition is either to or from a state with a value R/(R + G + B) that is greater than or equal to 0.8, and the difference between states is more than 0.2 (unitless) in the CIE 1976 UCS chromaticity diagram. [*ISO_9241-391*]

NOTE 4

Tools are available that will carry out analysis from video screen capture. However, no tool is necessary to evaluate for this condition if flashing is less than or equal to 3 flashes in any one second. Content automatically passes (see #1 and #2 above).

gesture DEVELOPING

motion made by the body or a body part used to communicate to technology

guideline DEVELOPING

high-level, plain-language outcome statements used to organize requirements

> NOTE
>
> Guidelines provide a high-level, plain-language outcome statements for managers, policy makers, individuals who are new to accessibility, and other individuals who need to understand the concepts but not dive into the technical details. They provide an easy-to-understand way of organizing and presenting the requirements so that non-experts can learn about and understand the concepts.
>
> Each guideline includes a unique, descriptive name along with a high-level plain-language summary. Guidelines address functional needs on specific topics, such as contrast, forms, readability, and more.
>
> Guidelines group related requirements and are technology-independent.

high-frequency corpus DEVELOPING

large collections of text (corpora) used in linguistics to identify and analyze words and phrases that appear most often in a language

> EXAMPLE
>
> - English: British National Corpus, Corpus of Canadian English, Corpus of Contemporary American English
> - Mandarin: A Frequency Dictionary of Mandarin Chinese
> - Russian: Russian National Corpus

human evaluation DEVELOPING

evaluation conducted by a human, typically to apply human judgement to tests that cannot be fully automatically evaluated

> NOTE
>
> Human evaluation is contrasted with automated evaluation which is done entirely by machine, though it includes semi-automated evaluation which allows machines to guide humans to areas that need inspection. Human evaluation involves inspection of content features, in contrast with usability testing which directly tests the experience of users with content.

human language EXPLORATORY

language that is spoken, written, or signed (through visual or tactile means) to communicate with humans

> NOTE
>
> See also sign language.

image PLACEHOLDER

> EDITOR'S NOTE
>
> To be defined.

image role PLACEHOLDER

> EDITOR'S NOTE
>
> To be defined.

image type PLACEHOLDER

> EDITOR'S NOTE
>
> To be defined.

infinite scrolling PLACEHOLDER

> EDITOR'S NOTE
>
> To be defined.

informative DEVELOPING

content provided for information purposes and not required for conformance. Also referred to as non-normative

interactive element DEVELOPING

element that responds to user input and has a distinct programmatically determinable name

> EXAMPLE
>
> Buttons, controls, input fields.

> NOTE
>
> In contrast to non-interactive elements. For example, headings or paragraphs.

items DEVELOPING

smallest testable unit for testing scope

> NOTE
>
> Items could be an interactive component such as a drop down menu, a link, or a media player.
>
> They could also be units of content such as a phrase, a paragraph, a label or error message, an icon, or an image.

keyboard focus DEVELOPING

point in the content where any keyboard actions would take effect

EXAMPLE

1. the keyboard focus on a control where an activate action like a space bar would take effect, or
2. the place in an input field where a typed character would be inserted.

keyboard interface DEVELOPING

API (Application Programming Interface) where software gets “keystrokes” from

NOTE

“Keystrokes” that are passed to the software from the “keyboard interface” may come from a wide variety of sources including but not limited to a scanning program, sip-and-puff morse code software, speech recognition software, AI of all sorts, as well as other keyboard substitutes or special keyboards.

label DEVELOPING

text or other component with a text alternative that is presented to a user to identify a component within web content

NOTE 1

A label is presented to all users whereas the name may be hidden and only exposed by assistive technology. In many (but not all) cases the name and the label are the same.

NOTE 2

The term label is not limited to the label element in HTML.

legal information EXPLORATORY

content that must be provided to meet statutory or regulatory requirements, or that provides legal protection such as terms and conditions

live DEVELOPING

information captured from a real-world event and transmitted to the receiver with no more than a broadcast delay

NOTE 1

A broadcast delay is a short (usually automated) delay, for example used in order to give the broadcaster time to cue or censor the audio (or video) feed, but not sufficient to allow significant editing.

NOTE 2

If information is completely computer generated, it is not live.

long-form text content EXPLORATORY

written material made up of multiple blocks of text, along with structural elements such as headings and lists, that work together to explain, inform, or convey a narrative

NOTE

Long-form text content is meant to be read as a continuous piece of text, such as an article, report, essay, or guide, rather than as separate or standalone items like product listings on a shopping page.

meaningful blocks of content EXPLORATORY

Group of related content that represents a distinct topic or function, intended to be perceived as a single unit and capable of being programmatically identified and labelled.

EXAMPLE

Examples include: a single sentence, a group of related user interface components, or a combination of text and media that serves a single purpose.

NOTE

These blocks are required to be programmatically determinable so that assistive technologies can identify the boundaries and purpose of the information.

mechanism DEVELOPING

process or technique for achieving a result

> NOTE 1
>
> The mechanism may be explicitly provided in the content, or may be relied upon to be provided by either the platform or by user agents, including assistive technologies.

> NOTE 2
>
> The mechanism needs to meet all requirements for the conformance level claimed.

media alternative DEVELOPING

alternative formats, usually text, for audio, video, and audio-video content including captions, audio descriptions, and descriptive transcripts

method DEVELOPING

detailed information, either technology-specific or technology-agnostic, on ways to satisfy the requirement

minimum bounding box DEVELOPING

the smallest enclosing rectangle aligned to the horizontal axis within which all the points of a shape lie

> NOTE
>
> For components which wrap onto multiple lines as part of a sentence or block of text, such as hypertext links, the bounding box is based on how the component would appear on a single line.

nested clause DEVELOPING

clause that is embedded inside another clause and neither clause can stand alone as a sentence

> NOTE
>
> Nested clauses use groups of words that are sometimes called dependent clauses or subordinate clauses.

EXAMPLE

“I know the teacher who said the book that you lent me was missing.” The main or independent clause is “I know the teacher”. It includes the dependent or subordinate clause “who said the book … was missing”, which includes the nested clause “that you lent me”.

non-interactive element DEVELOPING

element that does not respond to user input and does not include sub-parts

EXAMPLE

Text, an image, a video, and if they have no sub-elements: a paragraph, list item, and table cell.

NOTE 1

If a paragraph included a link, the text either side of the link would be considered a static element, but not the paragraph as a whole.

NOTE 2

Letters within text do not constitute a “smaller part”.

non-literal language DEVELOPING

words or phrases used in a way that are beyond their standard or dictionary meaning to express deeper, more complex ideas

NOTE

This is also called figurative language.

To understand the content, users have to interpret the implied meaning behind the words, rather than just their literal or direct meaning.

EXAMPLE

Examples include:

- allusions
- hyperbole
- idioms
- irony
- jokes
- litotes
- metaphors
- metonymies
- onomatopoeia
- oxymorons
- personification
- puns
- sarcasm
- similes

non-text content EXPLORATORY

any content that is not a sequence of characters that can be programmatically determinable or where the sequence is not expressing something in human language

NOTE

This includes ASCII art (which is a pattern of characters), emoticons, leetspeak (which uses character substitution), and images representing text

normative DEVELOPING

content whose instructions are required for conformance

page DEVELOPING

non-embedded resource obtained from a single URI using HTTP plus any other resources that are used in the rendering or intended to be rendered together

> NOTE
>
> Where a URI is available and represents a unique set of content, that would be the preferred conformance unit.

Path-based gesture DEVELOPING

gesture that depends on the path of the pointer input and not just its endpoints

> NOTE
>
> Path-based gesture includes both time dependent and non-time dependent path-based gestures.

platform DEVELOPING

software, or collection of layers of software, that lies below the subject software and provides services to the subject software and that allows the subject software to be isolated from the hardware, drivers, and other software below

> NOTE 1
>
> Platform software both makes it easier for subject software to run on different hardware, and provides the subject software with many services (e.g. functions, utilities, libraries) that make the subject software easier to write, keep updated, and work more uniformly with other subject software.

> NOTE 2
>
> A particular software component might play the role of a platform in some situations and a client in others. For example a browser is a platform for the content of the page but it also relies on the operating system below it.

> NOTE 3
>
> The platform is the context in which the conformance scope exists.

EXAMPLE

- A web browser is a platform as it provides software services to HTML, CSS, and JavaScript based software.
- An operating system provides software services to applications designed to run on that platform.
- An e-reader is a platform that provides software services to e-books.

point of regard DEVELOPING

position in rendered content that the user is presumed to be viewing, of which the dimensions can vary

EXAMPLE

It can be a two-dimensional area (e.g. content rendered through a two-dimensional graphical viewport), or a point (e.g. a moment during an audio rendering or a cursor position in a graphical rendering), or a range of text (e.g. focused text).

NOTE

The point of regard is almost always within the viewport, but it can exceed the spatial or temporal dimensions of the viewport. See rendered content for more information about viewport dimensions.

The point of regard can also refer to a particular moment in time for content that changes over time. For example, an audio-only presentation.

User agents can determine the point of regard in a number of ways, including based on viewport position in content, keyboard focus, and selection.

pointer DEVELOPING

a hardware-agnostic representation of input devices that can target a specific coordinate (or set of coordinates) on a screen, such as a mouse, pen, or touch contact

prerecorded EXPLORATORY

information that is not live

presentation DEVELOPING

rendering of the content in a form to be perceived by users

private and sensitive information EXPLORATORY

private and sensitive information

> EXAMPLE
>
> Such as, but not limited to:
>
> - Racial or ethnic origin
> - Personally identifiable information
> - Biometrics
> - Medical and health information
> - Gender identification
> - Financial information

process DEVELOPING

series of views or pages associated with user actions, where actions required to complete an activity are performed, often in a certain order, regardless of the technologies used or whether it spans different sites or domains

programmatically determinable DEVELOPING

meaning of the content and all its important attributes can be determined by software functionality that is accessibility supported

pseudo-motion EXPLORATORY

static content on the page that gives the user the perception or feeling of motion

relationship of meaning EXPLORATORY

Logical associations between elements that convey meaning and are necessary to understand the content

relative luminance DEVELOPING

the relative brightness of any point in a colorspace, normalized to 0 for darkest black and 1 for lightest white

NOTE 1

For the sRGB colorspace, the relative luminance of a color is defined as L = 0.2126 * **R** + 0.7152 * **G** + 0.0722 * **B** where **R**, **G** and **B** are defined as:

- if RsRGB <= 0.04045 then **R** = RsRGB/12.92 else **R** = ((RsRGB+0.055)/1.055) ^ 2.4
- if GsRGB <= 0.04045 then **G** = GsRGB/12.92 else **G** = ((GsRGB+0.055)/1.055) ^ 2.4
- if BsRGB <= 0.04045 then **B** = BsRGB/12.92 else **B** = ((BsRGB+0.055)/1.055) ^ 2.4

and RsRGB, GsRGB, and BsRGB are defined as:

- RsRGB = R8bit/255
- GsRGB = G8bit/255
- BsRGB = B8bit/255

The ”^” character is the exponentiation operator. (Formula taken from [*SRGB*].)

NOTE 2

Before May 2021 the value of 0.04045 in the definition was different (0.03928). It was taken from an older version of the specification and has been updated. It has no practical effect on the calculations in the context of these guidelines.

NOTE 3

Almost all systems used today to view web content assume sRGB encoding. Unless it is known that another color space will be used to process and display the content, authors should evaluate using sRGB colorspace.

NOTE 4

If dithering occurs after delivery, then the source color value is used. For colors that are dithered at the source, the average values of the colors that are dithered should be used (average R, average G, and average B).

> NOTE 5
>
> Tools are available that automatically do the calculations when testing contrast and flash.

> EDITOR'S NOTE
>
> WCAG 2.2 includes a separate page giving the relative luminance definition using MathML to display the formulas. This will need to be addressed for inclusion in WCAG 3.

rendered content PLACEHOLDER

> EDITOR'S NOTE
>
> To be defined.

requirement DEVELOPING

result of practices that reduce or eliminate barriers that people with disabilities experience

section DEVELOPING

self-contained portion of content that deals with one or more related topics or thoughts

> NOTE
>
> A section may consist of one or more paragraphs and include graphics, tables, lists and sub-sections.

semi-automated evaluation DEVELOPING

evaluation conducted using machines to guide humans to areas that need inspection

> NOTE
>
> Semi-automated evaluation involves components of automated evaluation and human evaluation.

sign language EXPLORATORY

a language using combinations of movements of the hands and arms, facial expressions, or body positions to convey meaning

sign language interpretation EXPLORATORY

translation of one language, generally a spoken language, into a sign language

> NOTE
>
> True sign languages are independent languages that are unrelated to the spoken language(s) of the same country or region.

simple pointer input DEVELOPING

input event that involves only a single ‘click’ event or a ‘button down’ and ‘button up’ pair of events with no movement between

> EXAMPLE
>
> Examples of things that would *not* be considered simple pointer inputs:
>
> - Double clicks
> - Dragging movements or gestures
> - Any use of multipoint input or gestures
> - The simultaneous use of a mouse and keyboard

single pointer input DEVELOPING

input modality that only targets a single point on the page/view at a time – such as a mouse, single finger on a touch screen, or stylus

> NOTE 1
>
> Single pointer interactions include clicks, double clicks, taps, dragging motions, and single-finger swipe gestures. In contrast, multipoint interactions involve the use of two or more pointers at the same time, such as two-finger interactions on a touchscreen, or the simultaneous use of a mouse and stylus.

> NOTE 2
>
> Single pointer input is in contrast to multipoint input such as two, three or more fingers or pointers touching the surface, or gesturing in the air, at the same time.

> NOTE 3
>
> Activation is usually by click or tap but can also be by programmatic simulation of a click or tap or other similar simple activation.

spatial audio DEVELOPING
: sounds that vary in volume and source location to create the illusion of different origin points

standard platform keyboard commands DEVELOPING
: keyboard commands that are the same across most or all platforms and are relied upon by users who need to navigate by keyboard alone

> NOTE
>
> A sufficient listing of common keyboard navigation techniques for use by authors can be found in Standard Keyboard Navigation & Operation Keys and Techniques.

starting point DEVELOPING
: the home page of a website or a set of web pages in a website

state DEVELOPING
: dynamic property expressing characteristics of a user interface component that may change in response to user action or automated processes

> NOTE
>
> States represent data associated with the component or user interaction possibilities. They do not affect the nature of the component. Examples include focus, hover, select, press, check, visited/unvisited, and expand/collapse.

structural component EXPLORATORY
: a component that provides the same organizational or navigational purpose across multiple pages/views pages/views

> EXAMPLE
>
> Examples include headers, footers, primary navigation menus, sidebars, and other repeated regions that help users understand and navigate the content consistently across a website or other digital product.

support PLACEHOLDER

> EDITOR'S NOTE
>
> To be defined.

synchronized media DEVELOPING

audio or video synchronized with another format for presenting information and/or with time-based interactive components, unless the media is a media alternative for text that is clearly labeled as such

task flow DEVELOPING

testing scope that includes a series views that support a specified user activity

> NOTE
>
> A task flow may include a subset of items in a view or a group of views. Only the part of the views that support the user activity are included in a test of the task flow.

test DEVELOPING

mechanism to evaluate implementation of a method

text DEVELOPING

sequence of characters that can be programmatically determined, where the sequence is expressing something in human language

text alternative DEVELOPING

text that is programmatically associated with non-text content or referred to from text that is programmatically associated with non-text content

> EXAMPLE
>
> An image of a chart is described in text in the paragraph after the chart. The short text alternative for the chart indicates that a description follows.

text content DEVELOPING

text and formatting that, together with the semantic or hierarchical structure provided by markup, communicate the content and meaning

transcript EXPLORATORY

a text equivalent of the source material

unnecessary words EXPLORATORY

extra words or complicated phrases that can be removed or replaced with simpler ones without changing the meaning of the text content

> EXAMPLE
>
> - due to the fact that → because
> - in order to complete → to complete
> - at this point in time → now

up event DEVELOPING

platform event that occurs when the trigger stimulus of a pointer is released

> NOTE
>
> The up event may have different names on different platforms, such as “touchend” or “mouseup”.

usability testing DEVELOPING

evaluation of the experience of users using a conformance scope or process by observation and feedback

user agent DEVELOPING

any software that retrieves and presents web content for users

> EXAMPLE
>
> Web browsers, media players, plug-ins, and other programs — including assistive technologies — that help in retrieving, rendering, and interacting with external content.

user interface component PLACEHOLDER

> EDITOR'S NOTE
>
> To be defined.

user need DEVELOPING
end goal a user has when starting a process through digital means

user-manipulable text DEVELOPING
text which the user can adjust

> NOTE
>
> This could include, but is not limited to, changing:
>
> - Line, word or letter spacing
> - Color
> - Line length — being able to control width of block of text
> - Typographic alignment — justified, flushed right/left, centered
> - Wrapping
> - Columns — number of columns in one-dimensional content
> - Margins
> - Underlining, italics, bold
> - Font face, size, width
> - Capitalization — all caps, small caps, alternating case
> - End of line hyphenation
> - Links

variation of product or process PLACEHOLDER

> EDITOR'S NOTE
>
> To be defined.

video DEVELOPING

the technology of moving or sequenced pictures or images

> NOTE
>
> Video can be made up of animated or photographic images, or both.

view DEVELOPING

content that is actively available in a viewport including that which can be scrolled or panned to, and any additional content that is included by expansion while leaving the rest of the content in the viewport actively available

> NOTE
>
> A modal dialog box would constitute a new view because the other content in the viewport is no longer actively available.

> EXAMPLE
>
> Examples of “included by expansion” include, but are not limited to: expanding paragraphs, non-modal dialogs, non-modal popups, error messages that appear embedded in content (for example, an “invalid password” error message).

viewport DEVELOPING

object in which the platform presents content

> NOTE 1
>
> The author has no control of the viewport and almost always has no idea what is presented in a viewport (e.g. what is on screen) because it is provided by the platform. On browsers the hardware platform is isolated from the content.

NOTE 2

Content can be presented through one or more viewports. Viewports include windows, frames, loudspeakers, and virtual magnifying glasses. A viewport may contain another viewport. For example, nested frames. Interface components created by the user agent such as prompts, menus, and alerts are not viewports.

EXAMPLE 1

For a web page author, the viewport is the HTML viewport.

EXAMPLE 2

For an application developer, the viewport is the window.

visual aids DEVELOPING

diagrams, photos, illustrations, or a simplified step process to supplement complex ideas, such as processes, workflows, relationships, or chronological information presented in the text content

visually collocated with DEVELOPING

in reasonable proximity within the viewport, regardless of magnification or zoom level, to sufficiently associate the error message to the field in error

§ A. Privacy Considerations

EDITOR'S NOTE

The content of this document has not matured enough to identify privacy considerations. Reviewers of this draft should consider whether requirements of the conformance model could impact privacy.

B. Security Considerations

> EDITOR'S NOTE
>
> The content of this document has not matured enough to identify security considerations. Reviewers of this draft should consider whether requirements of the conformance model could impact security.

C. Change log

This section shows substantive changes made in WCAG 3 since the First Public Working Draft was published in 21 January 2021.

The full commit history to WCAG 3 and commit history to Silver is available.

- 2025-09-04: Published requirements and assertions that have reached developing; removed exploratory content from draft; and updated assertions section of Explainer for WCAG 3.0.
- 2024-12-12: Reorganized exploratory guidelines; added 3 developing guidelines and accessibility supported; added user agent support, updated conformance section, and moved explanatory content to the Explainer for WCAG 3.0
- 2024-03-15: Updated placeholder guidelines with exploratory guidelines.
- 2023-07-24: Changed approach to WCAG 3 based feedback and removed old material that was not consistent with the new approach. Added WCAG 3 Guideline placeholders to indicate maturity level.
- 2021-12-07: Add Project Manager
- 2021-06-08: Moved explanatory information to Explainer for W3C Accessibility Guidelines (WCAG) 3.0

D. Acknowledgements

Additional information about participation in the Accessibility Guidelines Working Group (AG WG) can be found on the Working Group home page.

D.1 Contributors to the development of this document

- Adam Page (Invited Expert)
- Alastair Campbell (Gain)
- Alina Vayntrub (Understood.org)
- Andrew Kirkpatrick (Evinced Inc.)
- Ashley Firth (Invited Expert)
- Atya Ratcliff (Google LLC)
- Azlan Cuttilan (Invited Expert)
- Ben Tillyer (University of Oxford)
- Brian Elton (TPGi)
- Bruce Bailey (Invited Expert)
- Bryan Trogdon (Google LLC)
- Carrie Hall (SAP SE)
- Charu Pandhi (TPGi)
- Chuck Adams (Oracle Corporation)
- Claire Chang (Invited Expert)
- Detlev Fischer (Invited Expert)
- Eloisa Guerrero (Invited Expert)
- Eric Hind (Google LLC)
- Filippo Zorzi (UsableNet)
- Francis Storr (Intel Corporation)
- Frankie Wolf (Invited Expert)

- Gez Lemon (TetraLogical Services Ltd)
- Giacomo Petri (UsableNet)
- Glenda Sims (Deque Systems, Inc.)
- Graham Ritchie (Invited Expert)
- Gregg Vanderheiden (Invited Expert)
- Gundula Niemann (SAP SE)
- Heather Bellis (Oracle Corporation)
- Hidde de Vries (Logius)
- Illai Zeevi (Evinced Inc.)
- JaEun Jemma Ku (University of Illinois)
- Jan Jaap de Groot (Abra)
- Jan McSorley (Invited Expert)
- Jaunita Flessas (Invited Expert)
- Jen Goulden (Crawford Technologies)
- Jennifer Delisi (Invited Expert)
- Jeremy Katherman (Deque Systems, Inc.)
- Jeroen Hulscher (Logius)
- John Kirkwood (Invited Expert)
- John Rochford (Invited Expert)
- John Toles (Rhonda Weiss Center for Accessible IDEA Data)
- Jon Avila (Level Access)
- Jory Cunningham (Amazon)
- Julie Rawe (Understood.org)
- Kathy Eng (Invited Expert)
- Ken Franqueiro (W3C)
- Kevin White (W3C)
- Kimberly McGee (SAP SE)
- Laura Carlson (Invited Expert)
- Len Beasley (CVS Pharmacy, Inc.)

- Lisa Seeman-Kestenbaum (Invited Expert)
- Lori Oakley (Oracle Corporation)
- Makoto Murata (Information Accessibility Institute)
- Makoto Ueki (Invited Expert)
- Mary Jo Mueller (IBM Corporation)
- Mike Beganyi (Invited Expert)
- Mike Gower (IBM Corporation)
- Patrick H. Lauke (TetraLogical Services Ltd)
- Poornima Badhan Subramanian (Invited Expert)
- Priti Rohra (BarrierBreak Solutions Private Limited)
- Rachael Bradley Montgomery (Library of Congress)
- Radostina (Ina) Tsvetkova (Invited Expert)
- Rain Breaw Michaels (Google LLC)
- Rashmi Katakwar (Invited Expert)
- Rayianna Daniels-Wynn (Rhonda Weiss Center for Accessible IDEA Data)
- Reinaldo Ferraz (NIC.br - Brazilian Network Information Center)
- Roland Buss (SAP SE)
- Ryan Romaine (Google LLC)
- Sailesh Panchang (Deque Systems, Inc.)
- Sam Hobson (Intopia)
- Sarah Horton (Invited Expert)
- Sazzad Mahamud (Google LLC)
- Scott O’Hara (Microsoft Corporation)
- Shadi Abou-Zahra (Amazon)
- Shawn Thompson (Shared Services Canada)
- Steve Faulkner (TetraLogical Services Ltd)
- Steve Kerr (Google LLC)
- Steven Hoober (Invited Expert)
- Sydney Coleman (Google LLC)

- Tamsin Ewing (W3C)
- Tayef Farrar (CVS Pharmacy, Inc.)
- Tiffany Burtin (Invited Expert)
- Todd Libby (Invited Expert)
- Venkata Vemuri (Oracle Corporation)
- Wendy Reid (Invited Expert)
- Wilco Fiers (Deque Systems, Inc.)

§ D.2 Previous contributors to the development of this document

Abi James, Abi Roper, Adam Page, Alastair Campbell, Alexandra Yaneva, Alice Boxhall, Alina Vayntrub, Alistair Garrison, Amani Ali, Andrew Kirkpatrick, Andrew Somers, Andy Heath, Angela Hooker, Aparna Pasi, Ashley Firth, Atya Ratcliff, Avneesh Singh, Avon Kuo, Azlan Cuttilan, Ben Tillyer, Betsy Furler, Brooks Newton, Bruce Bailey, Bryan Trogdon, Carrie Hall, Caryn Pagel, Charles Hall, Charles Nevile, Chris Loiselle, Chris McMeeking, Christian Perera, Christy Owens, Chuck Adams, Cybele Sack, Daniel Bjorge, Daniel Henderson-Ede, Darryl Lehmann, David Fazio, David MacDonald, David Sloan, David Swallow, Dean Hamack, Detlev Fischer, DJ Chase, E.A. Draffan, Eleanor Loiacono, Eric Hind, Filippo Zorzi, Francis Storr, Frankie Wolf, Frederick Boland, Garenne Bigby, Gez Lemon, Giacomo Petri, Glenda Sims, Graham Ritchie, Greg Lowney, Gregg Vanderheiden, Gundula Niemann, Hidde de Vries, Imelda Llanos, Irfan Mukhtar, Jaeil Song, JaEun Jemma Ku, Jake Abma, Jamie Herrera, Jan Jaap de Groot, Jan McSorley, Janina Sajka, Jaunita Flessas, Jaunita George, Jeanne Spellman, Jedi Lin, Jeff Kline, Jen Goulden, Jennifer Chadwick, Jennifer Delisi, Jennifer Strickland, Jennison Asuncion, Jeroen Hulscher, Jill Power, Jim Allan, Joe Cronin, John Foliot, John Kirkwood, John McNabb, John Northup, John Rochford, John Toles, Jon Avila, Jory Cunningham, Joshue O’Connor, Judy Brewer, Julie Rawe, Justine Pascalides, Karen Schriver, Katharina Herzog, Kathleen Wahlbin, Katie Haritos-Shea, Katy Brickley, Kelsey Collister, Ken Franqueiro, Kevin White, Kim Dirks, Kimberly McGee, Kimberly Patch, Laura Carlson, Laura Miller, Len Beasley, Léonie Watson, Lisa Seeman-Kestenbaum, Lori Oakley, Lori Samuels, Lucy Greco, Luis Garcia, Lyn Muldrow, Makoto Ueki, Marc Johlic, Marie Bergeron, Mark Tanner, Mary Ann Jawili, Mary Jo Mueller, Matt Garrish, Matthew King, Melanie Philipp, Melina Maria Möhnle, Michael Cooper, Michael Crabb, Michael Elledge, Michael Fairchild, Michael Weiss, Michellanne Li, Michelle Lana, Mike Beganyi, Mike Crabb, Mike Gower, Nat Tarnoff, Nicaise Dogbo, Nicholas Trefonides, Nina Krauß, Omar Bonilla, Patrick H. Lauke, Paul Adam, Peter Korn, Peter McNally, Pietro Cirrincione, Poornima Badhan Subramanian, Rachael Bradley Montgomery, Rain Breaw Michaels, Ralph de Rooij, Rashmi Katakwar, Rebecca Monteleone, Rick Boardman, Roberto Scano,

Roldon Brown, Ruoxi Ran, Ruth Spina, Ryan Hemphill, Sailesh Panchang, Sarah Horton, Sarah Pulis, Scott Hollier, Scott O'Hara, Shadi Abou-Zahra, Shannon Urban, Shari Butler, Shawn Henry, Shawn Lauriat, Shawn Thompson, Sheri Byrne-Haber, Shrirang Sahasrabudhe, Shwetank Dixit, Stacey Lumley, Stein Erik Skotkjerra, Stephen Repsher, Steve Faulkner, Steve Lee, Sukriti Chadha, Susi Pallero, Suzanne Taylor, sweta wakodkar, Takayuki Watanabe, Tananda Darling, Theo Hale, Thomas Logan, Thomas Westin, Tiffany Burtin, Tim Boland, Todd Libby, Todd Marquis Boutin, Victoria Clark, Wayne Dick, Wendy Chisholm, Wendy Reid, Wilco Fiers.

§ D.3 Research Partners

These researchers selected a Silver research question, did the research, and graciously allowed us to use the results.

- David Sloan and Sarah Horton, The Paciello Group, "WCAG Success Criteria Usability Study"
- Scott Hollier et al, Curtin University, "Internet of Things (IoT) Education: Implications for Students with Disabilities"
- Peter McNally, Bentley University, "WCAG Use by UX Professionals"
- Dr. Michael Crabb, University of Dundee, Student research papers on Silver topics
- Eleanor Loiacono, Worcester Polytechnic Institute "Web Accessibility Perceptions" (Student project from Worcester Polytechnic Institute)

§ D.4 Enabling funders

This publication has been funded in part with U.S. Federal funds from the Health and Human Services, National Institute on Disability, Independent Living, and Rehabilitation Research (NIDILRR), initially under contract number ED-OSE-10-C-0067, then under contract number HHSP23301500054C, and now under HHS75P00120P00168. The content of this publication does not necessarily reflect the views or policies of the U.S. Department of Health and Human Services or the U.S. Department of Education, nor does mention of trade names, commercial products, or organizations imply endorsement by the U.S. Government.

E. References

E.1 Normative references

[RFC2119]

Key words for use in RFCs to Indicate Requirement Levels. S. Bradner. IETF. March 1997. Best Current Practice. URL: https://www.rfc-editor.org/rfc/rfc2119

[RFC8174]

Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words. B. Leiba. IETF. May 2017. Best Current Practice. URL: https://www.rfc-editor.org/rfc/rfc8174

E.2 Informative references

[ATAG20]

Authoring Tool Accessibility Guidelines (ATAG) 2.0. Jan Richards; Jeanne F Spellman; Jutta Treviranus. W3C. 24 September 2015. W3C Recommendation. URL: https://www.w3.org/TR/ATAG20/

[ISO_9241-391]

Ergonomics of human-system interaction—Part 391: Requirements, analysis and compliance test methods for the reduction of photosensitive seizures. International Standards Organization. URL: https://www.iso.org/standard/56350.html

[SRGB]

Multimedia systems and equipment - Colour measurement and management - Part 2-1: Colour management - Default RGB colour space - sRGB. IEC. URL: https://webstore.iec.ch/publication/6169

[UAAG20]

User Agent Accessibility Guidelines (UAAG) 2.0. James Allan; Greg Lowney; Kimberly Patch; Jeanne F Spellman. W3C. 15 December 2015. W3C Working Group Note. URL: https://www.w3.org/TR/UAAG20/

[WCAG22]

Web Content Accessibility Guidelines (WCAG) 2.2. Michael Cooper; Andrew Kirkpatrick; Alastair Campbell; Rachael Bradley Montgomery; Charles Adams. W3C. 12 December 2024. W3C Recommendation. URL: https://www.w3.org/TR/WCAG22/