Web Content Accessibility Guidelines (WCAG 2.1)

This document introduces the concept of digital accessibility, with a focus on the different practices and methodologies to ensure equal access to digital technologies for all users, regardless of their physical or cognitive abilities. It includes the various types of disabilities as well as the standards and criteria that must be followed to ensure an optimal user experience for everyone.

It also includes tools and methods that can be used to conduct a technical audit to assess and improve the accessibility of websites and mobile applications.

The document also presents the criteria of the WCAG 2.1 standards, which are essential guidelines for ensuring web content accessibility. For each criterion, the testing method and expected result are explained.

Table des matières

Introduction

The ambition of HI "Humanity and Inclusion, Handicap International" is to be a leading player in the inclusion of people with disabilities and vulnerable populations. This ambition also applies to HI staff. To revolutionize HI so that it is an example of an adapted organization.

Within the framework of the Be HInclusive IT Systems sub-project, HI wants to improve its own capacity to include a greater diversity of people with disabilities among its employees. The action plan includes the accessibility of applications and websites managed by the federation.

1 Digital accessibility

The power of the Web lies in its universality. Access for all regardless of disability is an essential aspect.
Tim Berners-Lee, director and creator of the W3C "World Wide Web Consortium".
Accessibility, an advantage for all, a necessity for some.
Digital accessibility aims to make it possible to access digital information, regardless of the nature of the person's disability and the way in which they consult the information. It concerns different technologies such as the Web, videos and Word and PDF documents, but also mobile applications.
The point is not to increase the number of information media, but of respecting functional, graphic, technical and editorial rules that will allow everyone to access the information, whatever their consultation tools.
This document outlines the methods used around the world to improve the digital accessibility of web content (sites and applications). The WCAG 2.1 "Web Content Accessibility Guidelines 2.1" is the recent version of the W3C international recommendations. The French government's RGAA 4.1 "Référentiel général d'amélioration de l'accessibilité 4.1" is based on this standard. The RGAA 4.1 is fully compatible with WCAG 2.1 (level A and AA).

The WCAG 2.1 standards are also included in the ADA (American with Disability Act) and the European standard EN 301 549 V2.1.2.

2 Disability situations

The disability situations identified are not only those we see.
They are not necessarily definitive and can affect each of us at some point in our lives.

This chapter provides adapted solutions for each disability situation.

2.1 Hearing impairment

- Deaf people;
- Hearing impaired people;
- Deafblind people;
- Difficulty in perceiving speech in a noisy environment.

Adapted solutions: transcription, subtitles, videos have subtitles or textual transcription.

2.2 Visual impairment

- Visually impaired people;
- People who do not distinguish colors or do not distinguish them well (color blindness, Achromatopsia);
- People who are sensitive to high color brightness;
- Blind people;
- Deafblind people.

Adapted solutions: voice synthesis, display customization (color adjustment).

- ✓ The user can set the size of all text content in the application;
- ✓ The color contrasts are high;
- ✓ The videos have an audio description or a textual transcription;
- ✓ The images have a textual alternative.

2.3 Cognitive impairment

- People with DYS disorders;
- Other disorders affecting reading, memory, concentration, comprehension, problem solving, reasoning and language.

Adapted solutions: simplification of information, linear layout, sans-serif font (Arial, Open Dyslexic, AccessibleDfa), dictation software.

- ✓ The screens all follow the same logic of organization of content and navigation;
- ✓ Interface components (links, buttons, etc.) are easily identifiable;
- ✓ It is possible to pause or stop moving elements and sounds;
- ✓ The user is not limited in time to perform an action;

- ✓ The form field labels are clearly and explicitly marked, input help is provided if necessary, and input errors are identified and explained;
- ✓ User help is available.

2.4 Motor and grip disabilities

- Difficulty in gripping and controlling movements;
- People who have difficulty in making key combinations;
- People with shaking hands;
- People who cannot use one or both hands;

Adapted solutions: adapted peripherals (ergonomic mouse or keyboard, finger guides for keyboard, trackball, joystick, contactor…), dictation software.

- ✓ Keyboard navigation allows access to all features without keyboard traps;
- ✓ The focus is clearly visible;
- ✓ The clickable areas are large enough (width and height of at least 48 px);
- ✓ The accessibility rules serve beyond the identified disability situations and provide comfort for all, some cases below:
  - Noisy environments such as open spaces, shops, reception areas, etc.
  - People with a temporary disability (broken arm, eye infection…);
  - Inappropriate equipment or service (black and white printer, low speed, etc.);
  - Forms to be filled in keyboard navigation allows forms to be filled in without using the mouse and saves a lot of time.

2.5 Seniors

A senior citizen is an elderly person belonging to a category of people whose definition varies according to the context: work, health, etc.

From a health point of view, the World Health Organization (WHO) considers an elderly person to be someone over 60 years of age. It estimates that between 2000 and 2050, this population will double worldwide from 11% to 22%.

According to INSERM, the decline in cognitive abilities begins at the age of 45.

Impairments that may be age-related have an impact on the use of digital technologies.

The barriers faced by seniors with age-related disabilities can be manifold:

- Declining **vision,** such as reduced contrast sensitivity, color perception, ability to focus on close objects, makes it difficult to read a digital resource.
- The decline in **motor skills**, such as decreased dexterity, fine motor control, makes it difficult to use the mouse and click on small items.

- The decline in **hearing**, such as the ability to perceive higher pitched sounds, to separate sounds, makes it difficult to listen to audio material (especially in a noisy environment).
- The decline in **cognitive abilities**, such as short-term memory and concentration, makes it difficult to navigate and perform tasks.

3 W3C/WAI key player in web accessibility

3.1 World Wide Web Consortium (W3C)

The World Wide Web Consortium (W3C) is an international consortium in which member organizations, a full-time team and the public work together to develop Web standards. W3C is an NGO founded in 1994 by Tim Berners-Lee.

The W3C pursues its mission primarily through the creation of Web standards and guidelines to ensure the long-term growth of the Web. In other words, it develops international recommendations (specifications) for the Web HTML, CSS etc.

Over 400 organizations are members of the consortium. The MIT Computer Science jointly leads the W3C with the Artificial Intelligence Laboratory (MIT CSAIL) in the United States; the European Research Group in Computer Science and Mathematics (ERCIM) based in France, and Keio University in Japan, and has several offices around the world.

3.2 Web Accessibility Initiative WAI

The W3C Web Accessibility Initiative (WAI) was established in 1997 by W3C to develop specifications for the accessibility of the Web and its technologies.

The W3C in partnership with global organizations has a mission to

- ✓ Make the Web accessible to all, and therefore to ensure that Web technologies are compatible with accessibility;
- ✓ Develop accessibility guidelines for Web content, user agents and publishing tools;
- ✓ Facilitate the development of tools to evaluate and facilitate accessibility;
- ✓ Provide education and advocacy;
- ✓ Coordinate with advanced research and development.

The WAI is supported in part by the US Department of Education's National Institute for Disability and Rehabilitation Research (NIDRR), the European Commission's IST Programme, HP, IBM, Microsoft, SAP, and Wells Fargo.

3.3 W3C and WAI Guidelines

WCAG (Web Content Accessibility Guidelines)

Developed in 1999 "WCAG 1.0", in 2008 "WCAG 2.0" and in 2018 "WCAG 2.1".

ATAG (Authoring Tool Accessibility Guidelines)

Set of recommendations (V 2.0 in 2003) for software publishers.

UAAG (User Agent Accessibility Guidelines)

Recommendations to developers of software for browsing and viewing content.

XAG (XML Accessibility Guidelines)

Explain how to improve accessibility in Extensible Markup Language (XML) applications.

3.4 The WCAG 2.1 standard

These are international recommendations issued by the W3C's Web Accessibility Initiative WAI working group. In June 2018, the WCAG 2.1 becomes an official W3C recommendation.

WCAG 2.1 is structured around four main principles:

1. Perceptible: content visible even to the blind or partially sighted;
2. Usable: any user should be able to interact with the content;
3. Understandable: content and interactions that are understandable despite impairments;
4. Robust: easily read by browsers, readers and assistive technologies.

Each principle has guidelines (rules): 13 in total. WAI advocates three levels of accessibility and compliance, which are symbolized by the capital letter A: A, double AA and triple AAA. Each rule has measurable success criteria in compliance level A (the lowest level and contains 30 success criteria), AA (contains 20 success criteria) and AAA (the highest and contains 28 success criteria).

A: A web content developer must satisfy this checkpoint:

- Alternative text for images, frames and animations;
- Text equivalents.

AA: A web content developer should satisfy this checkpoint:

- Respect for semantics: appropriate use of tags;
- Separation of informational code and formatting code, now managed by style sheets;
- Ensure sufficient contrast between text and background;
- Facilitate keyboard navigation;

- Provide meta-information about the page (language, content...)
- Create a logical order for the forms.

AAA: A web content developer can meet this checkpoint:

- Implementation of keyboard shortcuts to access the different parts of the site;
- Hierarchy of menus to facilitate keyboard navigation.

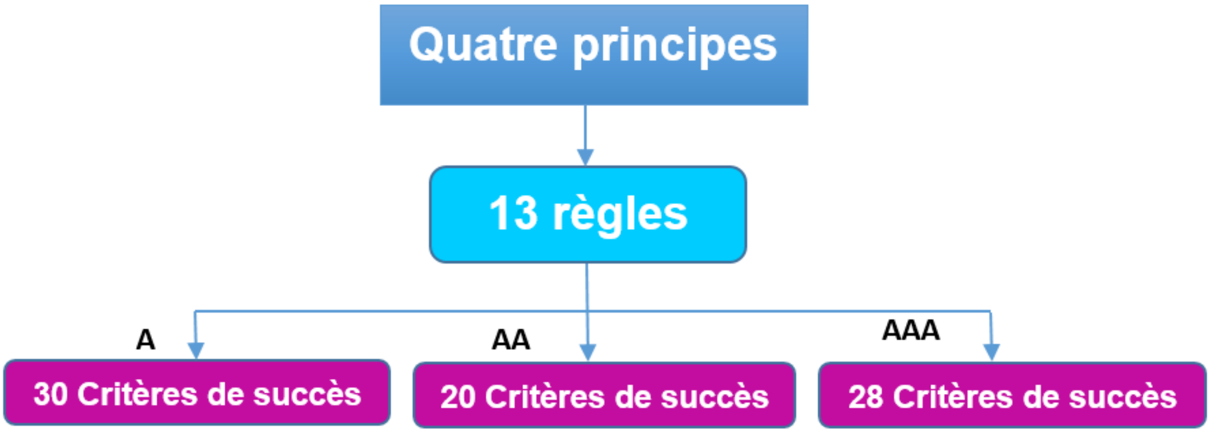


Figure 1 WCAG 2.1 organization

Perceptible

Web content is made available to the senses - sight, hearing, and/or touch.

Rule 1.1 Textual alternative: Provide a textual alternative for any non-textual content.

Success criteria	WCAG 2.1 recommendations
1.1.1 non-textual content. **Level A.**	Images, image-based form buttons, and clickable areas within an image have an equivalent alt text. Images that do not carry information, that are decorative, or that contain content also given as text, are given a null alt text (alt="") or are implemented as background images in CSS. Images that are links have descriptive alternative text. Text equivalents are provided for complex images, either in context or on a separate page to which a link is provided. Form buttons have a descriptive value attribute. Form fields are associated with labels.

Success criteria	WCAG 2.1 recommendations
	Integrated multimedia content is identified by accessible text. Frames and iframes have an appropriate title.

Rule 1.2 Temporal media: Provide alternative versions of temporal media.

Note: If the audio or video is designated as an alternative to web content (e.g.an audio or sign language version of a web page), then the web content in question serves as an alternative.

Success criteria	WGAG 2.1 recommendations
1.2.1 Audio or video content only (pre-recorded). **Level A.**	A full text transcript (containing all relevant audio information) is provided for all pre-recorded audio fragments. A text or audio description is provided for all pre-recorded video-only content (without sound), unless the video is decorative.
1.2.2 Subtitles (pre-recorded). **Level A.**	Synchronized subtitles are provided for pre-recorded videos (YouTube videos, etc.)
1.2.3 Audio description or alternative version for a time-based (prerecorded) medium. **Level A**.	A descriptive text transcript OR audio description is provided for a pre-recorded video. Note: Only necessary if visual information is not available in the audio track.
1.2.4 Subtitles (live). **Level AA.**	Synchronized subtitles are provided for any live multimedia content that contains sound (audio broadcasts, webcasts, video conferences, etc.)
1.2.5 Audio description (pre-recorded). **Level AA.**	Audio description is provided for all video content Note: Only necessary if the video conveys information visually without this information being present in the audio track.

Success criteria	WGAG 2.1 recommendations
1.2.6 Sign language (pre-recorded). **Level AAA.**	A sign language video is provided for all audio content.
1.2.7 Extended Audio Description (pre-recorded). **AAA level**.	Where it is not possible to add an audio description to a video due to timing (not enough white space in the audio track), an alternative version of the video containing pauses for audio description is provided.
1.2.8 Replacement version for time-based (prerecorded) media. **AAA level.**	A descriptive text transcript is provided for any pre-recorded time-based media that contains a video track.
1.2.9 Audio only (Live)**. AAA level.**	A descriptive text transcript (e.g. the script of the live audio) is provided for any live content that has an audio track.

Rule 1.3 Adaptable: Create content that can be presented in different ways (e.g. simplified layout) without losing information or structure.

Success criteria	WCAG 2.1 recommendations
1.3.1 Information and relationships. **Level A.**	Semantic tags are used to indicate titles (<h1>), regions/landmarks, lists (<ul>, <ol>, and <dl>), highlighted or special text (<strong>, <abbr>, <blockquote>, for example), etc. Semantic tags are used appropriately. Tables are used for tabular data and data cells are associated with their headers. Table captions, if present, are associated with their table. Labels are associated with the corresponding form fields. Linked form fields are grouped together using fieldset/legend. ARIA can be used to provide labels when standard HTML is not sufficient.
1.3.2 Logical sequential order. **Level A.**	The reading and navigation order (determined by the order of the source code) is logical and intuitive.

Success criteria	WCAG 2.1 recommendations
1.3.3 Sensory characteristics. **Level A.**	Instructions are not based on shape, size or visual positioning (e.g. "click the square to continue" or "instructions are in the right-hand column"). The instructions are not dependent on a sound (e.g. "You can continue when you hear a beep.").
1.3.4 Orientation. **Level AA.**	The orientation of web content is not limited to portrait or landscape unless a specific orientation is required.
1.3.5 Identify the purpose of the entry. **Level AA.**	Form fields that collect certain types of information from the user have an appropriate autocomplete attribute.
1.3.6 Identify the purpose. **Level AAA.**	HTML5 regions or ARIA landmark roles are used to identify page regions. ARIA is used, where appropriate, to enhance the semantics of HTML to better identify the function of interface components.

Distinguishable Rule 1.4: Facilitate the user's visual and auditory perception of the content, in particular by separating the foreground from the background.

Success criteria	WCAG 2.1 recommendations
1.4.1 Use of color. **Level A.**	Color is not used as the only way to convey information or to distinguish a visual element. Color is not used as the only way to distinguish a link from the surrounding text unless the contrast between the color of the link and the text is at least 3:1 and another way of differentiating (e.g. underlining) the link is used when the link is hovered over by the cursor or receives focus.
1.4.2 Sound control. **Level A.**	A mechanism is available to pause, stop or control the volume of the sound that automatically starts on a page for more than 3 seconds,

Success criteria	WCAG 2.1 recommendations
1.4.3 Contrast (Minimum). **Level AA.**	Text and text images have a contrast ratio of at least 4.5:1. Large text - at least 18 point (typically 24px) or 14 point bold (typically 18.66px) - has a contrast ratio of at least 3:1
1.4.4 Text Resizing. **Level AA.**	The page is readable and functional when the page is magnified to 200%. Note: 1.4.10 below introduces a more stringent criterion for content enlargement.
1.4.5 Text as Image. **Level AA.**	While it is possible to achieve the same visual presentation using text alone, this text is not presented as an image.
1.4.6 Contrast (enhanced). **Level AAA.**	Text and text in image form have a contrast ratio of at least 7:1. Large text - at least 18 point (typically 24px) or 14 point bold (typically 18.66px) - has a contrast ratio of at least 4.5:1
1.4.7 Low or no background noise. **Level AAA.**	Audio content that contains mainly speech has little or no background noise so that the speech can be easily distinguished.
1.4.8 Visual presentation. **Level AAA.**	Text blocks that are longer than one sentence: Are not wider than 80 characters. Not justified (simultaneously right and left aligned). Have sufficient space between lines (at least 1/2 the height of the text) and between paragraphs (at least 1.5 times the line spacing). Have specified text and background colors. These colors can be applied to specific elements or to the whole page using CSS (and will therefore be inherited by all other elements). Do not require horizontal scrolling when the text size is doubled.

Success criteria	WCAG 2.1 recommendations
1.4.9 Text in image form (no exceptions). **Level AAA.**	Text as an image is only decorative (the image does not convey any information) OR the image cannot be presented as text alone.
1.4.10 Redistribution. **Level AA.**	There is no loss of content or functionality and horizontal scrolling is avoided when content is presented at a width of 320 pixels. This requires a responsive design for most websites. It is best to test this option by setting the browser window to 1280 pixels wide and zooming in on the page content up to 400%. There is an exception for content that requires horizontal scrolling, such as data tables, complex images (such as maps and graphics), menu bars, etc.
1.4.11 Non-textual contrast. **Level AA**.	There is a contrast of at least 3:1 to differentiate between graphical objects (such as icons or components of graphs or diagrams) and author-adapted interface components (buttons, form fields and focus/contour indicators). The different states (focus, hover, active, etc.) of the interface components adapted by the author must have a contrast of 3:1.
1.4.12 Text Spacing. **Level AA.**	No loss of content or functionality occurs when the user adjusts text line height/spacing to 1.5 times font size, paragraph spacing to 2 times font size, word spacing to 0.16 times font size and letter spacing to 0.12 times font size. The best way to allow this is not to set the height of text elements in pixels.
1.4.13 Content with hover or focus. **Level AA**.	When additional content is presented on mouse-over or keyboard focus : The newly displayed content can be hidden (usually via the ESC key) without moving the pointer or keyboard focus, unless the

Success criteria	WCAG 2.1 recommendations
	content has an input error or does not interfere with or disturb the content of another page. The pointer can be moved to the new content without it disappearing. The new content should remain visible until the pointer, keyboard focus moves away from the element that triggered the display, the new content is closed, or the new content is no longer relevant.

Operable

The user interface components, forms and navigation are usable.

Rule 2.1 Keyboard accessibility: Make all functionality accessible by keyboard.

Success criteria	WCAG 2.1 recommendations
2.1.1 Keyboard. **Level A.**	All functionality on a page is usable using the keyboard, unless the functionality cannot be accomplished in any known way using a keyboard (e.g. freehand drawing). Keyboard shortcuts and access keys (access keys should typically be avoided) do not conflict with existing browser or screen reader shortcuts.
2.1.2 No keyboard trap. **Level A.**	The keyboard focus is never locked inside a page element. The user can move the focus in and out of any navigable element on the page using only the keyboard.
2.1.3 Keyboard (no exceptions). **Level AAA.**	All the functionality of the page is available using the keyboard.
2.1.4 Keyboard shortcuts**. Level A.**	If a keyboard shortcut uses printable letters, the user must be able to disable the shortcut, change the assigned letter to a non-printable key (Ctrl, Alt, etc.) or the shortcut can only be active when the associated interface component or button has focus.

Rule 2.2 Sufficient time: Allow the user sufficient time to read and use the content.

Success criteria	WGAG 2.1 recommendations
2.2.1 Setting the delay. **Level A.**	If a page or application has a time limit, the user can remove, adjust or extend the time limit. This is not required for real-time events (e.g. an auction), or if the time limit is more than 20 hours.
2.2.2 Pause, Stop, Hide. **Level A**.	Any content that is moving, flashing or scrolling (such as a carousel or animation) for more than 5 seconds can be paused, stopped or hidden by the user. Content that is updated automatically (e.g. a dynamically updating news feed, chat messages) can be paused, stopped or hidden by the user, or the user can manually control when updates take place.
2.2.3 No delay. **Level AAA.**	The content or functionality has no limits or time constraints.
2.2.4 Interruptions. **AAA level.**	Interruptions (alerts, page updates, etc.) can be postponed or deleted by the user.
2.2.5 New authentication. **Level AAA.**	If an authenticated session expires, the user can continue without loss of data after re-authentication.
2.2.6 Timeouts. **Level AAA.**	Users should be warned of any timeout that may result in data loss, unless the data is retained for more than 20 hours of user inactivity.

Rule 2.3 Seizures and physical reactions: Do not design content that may cause seizures or physical reactions.

Success criteria	WGAG 2.1 recommendations
2.3.1 No more than three flashes or below the critical threshold. **Level A**.	No content on the page flashes more than three times per second unless the content is small enough, has low contrast and does not contain too many reds. (See general flash and red flash thresholds)

Success criteria	WGAG 2.1 recommendations
2.3.2 Three flashes. **AAA level.**	No page content flashes more than three times per second
2.3.3 Animation following an interaction. **Level AAA**.	The user can stop animation and non-essential movements resulting from user interaction.

Rule 2.4 Navigable: Providing the user with guidance for navigating, finding content and locating themselves within the site.

Success criteria	WCAG 2.1 recommendations
2.4.1 Bypassing blocks. **Level A**.	A link is provided to avoid navigation and other elements that are repeated on all pages. A sufficient title structure and/or identification of the regions/landmarks of the page can also be considered as a sufficient technique. It is recommended that an escape link be provided for the benefit of sighted people who navigate with the keyboard, because navigation by means of headings or regions is not yet supported in most browsers.
2.4.2 Page Title. **Level A.**	The web page has an informative and descriptive page title.
2.4.3 Focus pathway. **Level A.**	The tabbing order of links, form elements etc. is logical and intuitive.
2.4.4 Function of the link (in context). **Level A.**	The function of each link (or image form button or clickable area of an image) can be determined from the link text alone, or from the link text and its context (e.g. the paragraph, list item, table cell containing the link, or table cell header). Links (or graphic buttons) that have the same text but different destinations can easily be distinguished from each other.
2.4.5 Multiple access. **Level AA**.	There are several ways to find other pages on the site - at least two of the following: a list of linked pages, a table of contents, a site map, a search function, or a list of all available pages.

Success criteria	WCAG 2.1 recommendations
2.4.6 Headers and labels. **Level AA**.	Page titles and labels of form components and interactive components are informative. Avoid duplicating titles (e.g. "More details") or labels (e.g. "First name") unless the structure allows sufficient differentiation.
2.4.7 Visibility of focus. **Level AA.**	It is possible to visually distinguish on which element the focus of the keyboard is (in other words, when you browse a page with the TAB key you see where you are).
2.4.8 Location. **Level AAA.**	If a web page is part of a sequence of pages or is in a complex site structure, an indication of the location of the current page is provided, e.g. via a breadcrumb trail or by specifying the current step (e.g. "step 2 of 5 - delivery address").
2.4.9 Link function (Link only). **Level AAA.**	The function of each link (or image form button or clickable area of an image) can be determined from the link text alone There are no links (or graphic form buttons) with the same heading that have different destinations.
2.4.10 Section headings. **Level AAA.**	In addition to indicating the overall structure of the document, individual content sections are designated using headings where appropriate.

Rule 2.5 Input methods: Facilitate the use of functionality by users with input methods other than the keyboard.

Success criteria	WCAG 2.1 recommendations
2.5.1 Pointer gestures. **Level A.**	If multi-point or path-based movements (swiping or dragging across the screen) are not essential to the functionality, the functionality can also be performed with a one-point activation (e.g. adding a button).
2.5.2 Pointer Cancellation. **Level A.**	To avoid unwanted activation of interface components, avoid using unnecessary events (e.g. on mouse down) when clicking, tapping or long pressing the screen. Use on click, on mouse up or similar instead. If on mouse up (or similar) is used, you must provide a mechanism to cancel the action.

Success criteria	WCAG 2.1 recommendations
2.5.3 Label in the name. **Level A.**	If an interface component (link, button, etc.) has text (or text images), the accessible name (label, alt text, aria label, etc.) for that component must contain the visible text.
2.5.4 Activation by movement. **Level A.**	Functionality activated by moving the device (e.g. by shaking or turning the mobile device horizontally) or by user movement (such as switching to a camera) can be disabled and equivalent functionality is provided via standard interface elements such as buttons.
2.5.5 Target size. **Level AAA.**	The size of clickable areas is at least 44 by 44 pixels unless an alternative target of this size is provided, the area is embedded (e.g. a link in a sentence), the area has not been modified by the author (e.g. a standard checkbox) or the area is not essential for the functionality.
2.5.6 Simultaneous input mechanisms. **Level AAA.**	The content does not limit input to a specific modality, such as touch input or keyboard only, but must support other input methods (e.g. using a keyboard on a mobile device).

Understandable

The information and use of the interface is understandable.

Rule 3.1 Readable: Make the text readable and understandable.

Success criteria	WCAG 2.1 recommendations
3.1.1 Language of the page. **Level A.**	The language of the page is indicated by means of the lang attribute (<HTML lang="en">, for example).
3.1.2 Language of a passage. **Level AA.**	When content is in a language other than the language of the page, the language of that content is indicated using the lang attribute (e.g. <blockquote lang="es">).
3.1.3 Rare words. **Level AAA.**	Words that may be ambiguous, unfamiliar, or used in a specific way are defined in the adjacent text, in a list of definitions, a glossary or in another appropriate way.

Success criteria	WCAG 2.1 recommendations
3.1.4 Abbreviations. **Level AAA.**	The meaning of an abbreviation is provided the first time an abbreviation is used, by adding its full form or giving its definition, by using the element, or by adding a link to a definition or glossary. NOTE: WCAG 2.0 makes no exception for commonly used abbreviations (e.g. "HTML" should be given in its full form on a web design site).
3.1.5 Reading level. **Level AAA.**	A more comprehensible alternative is provided for any content that is too advanced to be read by someone with about 9 years of education (end of lower secondary school).
3.1.6 Pronunciation. **Level AAA.**	If the pronunciation of a word is vital to understanding the meaning of that word, the pronunciation is given immediately after the word or via a link or glossary.

Rule 3.2 Predictable: Ensure that pages appear and function in a predictable manner.

Success criteria	WCAG 2.1 recommendations
3.2.1 In focus. **Level A.**	When an element of the page receives focus, it does not result in a substantial change to the page, the appearance of a pop-up window, an additional change in focus or any other change that might disorient or confuse the user.
3.2.2 At entry. **Level A**.	When the user enters information or interacts with a component, this does not result in a substantial change to the page, the appearance of a pop-up window, an additional change in focus or any other change that might disorient or confuse the user, unless the user is warned.
3.2.3 Consistent navigation. **Level AA.**	Navigation links that are repeated on several pages appear in the same order on the different pages of the site.
3.2.4 Consistent identification. **Level AA.**	Elements that have the same functionality on different pages of the site are presented in a consistent manner. For example, a search box at the top of all pages should always have the same label.

Success criteria	WCAG 2.1 recommendations
3.2.5 Change on demand. **Level AAA.**	The user must initiate a substantial change to the page, the appearance of a pop-up window, an unexpected change in focus, or any other change that might disorient or confuse the user. Alternatively, the user can choose to disable such changes.

Rule 3.3 Input assistance: helping the user to avoid and correct input errors.

Success criteria	WCAG 2.1 recommendations
3.3.1 Error identification. **Level A.**	For mandatory form fields or those that require a specific format, value or length, this information is provided within the field label. If present, form validation errors are presented in an effective, intuitive and accessible manner. The error is clearly identified, quick access to the item to be corrected is provided and the user can easily correct the error and resubmit the form.
3.3.2 Labels or instructions. **Level A.**	The mandatory interactive elements have sufficient information in the form of instructions, examples, well-positioned labels and/or fieldset and legend elements.
3.3.3 Suggestion after an error. **Level AA.**	If an input error is detected (whether by client-side or server-side validation), provide a suggestion for correcting the error in a timely and accessible manner.
3.3.4 Prevention of errors (legal, financial, data). **Level AA.**	If the user can change or delete legal, financial or test data, these actions can be reversed, verified or confirmed.
3.3.5 Help. **Level AAA.**	Provide instructions and hints to help with the completion and submission of the form.
3.3.6 Error prevention (all). **Level AAA.**	If the user can submit information, the submission is reversible, verified or confirmed.

Robust

The content can be reliably used by a wide variety of user agents, including assistive technologies.

Rule 4.1 compatible: optimize compatibility with current and future user agents, including assistive technologies.

Success criteria	WCAG 2.1 recommendations
4.1.1 Syntactic analysis. **Level A.**	Significant HTML/XHTML validation/syntax errors are avoided. Check using the W3C validator.
4.1.2 Name, role, value. **Level A.**	Markup is used in a way that facilitates accessibility. This means, among other things, following the HTML/XHTML specification and using forms, form labels, frame titles etc. appropriately. ARIA is used to improve accessibility when HTML is not enough.
4.1.3 Status messages. **Level AA.**	If an important status message is presented and the focus is not on that message, the message must be announced to the screen reader users. This is usually done via an ARIA or aria-live alert.

4 RGAA 4.0 standard

4.1 General Framework

The French law n°2005-102 of 11 February 2005 for equal rights and opportunities, participation and citizenship of disabled people makes accessibility a requirement for all online public communication services of the State, local authorities and public establishments depending on them. The law recognizes the "International Web Accessibility Guidelines" WCAG 2.1 as the reference to be followed.

Confirming article 5 of the decree n° 2019-768 of 24 July 2019, concerning the accessibility of online public communication services to disabled people. The HI federation is part of the exempted organizations, nevertheless in the framework of the Be HInclusive project; the federation's web content must be accessible and consultable by disabled people. For this reason, the federation has taken the initiative to make the web content compliant with the RGAA 4 standards for web services based in France and compliant with the WCAG 2.1 for web services in other countries.

Version 4 of the RGAA is structured in two parts. The first part presents the obligations to be respected: it is aimed at lawyers, managers and all web and accessibility professionals.

The second, dedicated to the technical audit method, contains a list of control criteria for verifying the conformity of a web page: it is intended for RGAA auditors.

4.2 Communication strategy

The homepage of the online web content must display one of the following:

- "Accessibility: fully compliant" if all the RGAA/WCAG control criteria are met;
- "Accessibility: partially compliant" if at least 50% of the RGAA/WCAG control criteria are met;
- "Accessibility: not compliant" if there are no valid audit results to measure compliance with the criteria or if less than 50% of the WCAG control criteria are met.

This statement can be clickable and lead to the "Accessibility" page, which contains the accessibility statement and can be accessed directly from the home page and from any page on the site.

The accessibility statement applies to each HI website and web application. The accessibility statement is drawn up in accordance with the template set out in the RGAA. No online content is exempt from the declaration. This declaration must contain the following information:

- Compliance status;
- Test results;
- Content not accessible.

Links to the current year's multi-annual scheme and action plan are included in the accessibility statement for each website or web application.
The multi-annual plan presents **the organization's policy on digital accessibility**.

As such, it contains information on:

- Taking into account digital accessibility in the federation's digital strategy and in its policy in favor of the integration of people with disabilities, the case of the **Be HInclusive Project (mentioned above);**

- The functional position and tasks of the organization’s digital accessibility officer ;
- The human and financial resources allocated to digital accessibility ;
- Training and awareness-raising activities for staff ;
- The implementation of external resources and expertise, if any, and the technical means and tools to manage and test digital accessibility;
- The integration of digital accessibility in contractual clauses (calls for tender and quotations), in the criteria for rating and selecting service providers and in acceptance procedures and, where appropriate, in the agreements established with their operators, delegates or partners;

It also presents the work done to bring the organization’s online public communication services into compliance**,** including :

- Taking into account digital accessibility in new projects;
- Taking into account people with disabilities in user testing ;
- Compliance assessments (or audits) planned for all communication services;
- The remedial measures that will be taken to address non-accessible content, including a timetable for implementation of these measures, taking into account the priority of the most accessed content and the most used services;
- Non-mandatory accessibility measures, such as access to audio and video content in sign language, translation of some content into simplified language and any other measure that takes into account the triple AAA criteria of international standards, listed in the annex to the reference standard;
- Review of the annual action plans.

The annual action plan describes in detail the operations implemented to address digital accessibility needs at HI.

Mapping all websites, business applications and online publications: Summary table to assess the level of accessibility and establish an action plan.

5 Conformity assessment

The accessibility evaluation of an application, a website or any other digital resource is necessary to validate the accessibility of each content, we can mention two types of evaluation: technical-functional and user.

5.1 The Technical - functional assessment

To assess the compliance of the web content with the reference standard (RGAA, WCAG), the organization must conduct an accessibility audit:

- The audit may be carried out by the organization itself or by a third party;
- The evaluation is performed on a sample of pages representative of the website or web application;
- The compliance of the sample pages with the applicable criteria is checked using the WCAG 2.1 control criteria;
- The final phase of the audit is the calculation of the content compliance rate.

The audit consists of the initial technical testing, which consists of inspecting the source code, color contrast, layout and structuring of information. Some of these tests can be carried out automatically by tools, but the majority requires manual control.

Next, functional tests are needed to evaluate the behavior of the interface with the different assistive technologies. The aim is to ensure that the evaluated web content is free of blocking points. The tests are mainly carried out according to three aspects; the use of a screen reader, software zoom and keyboard navigation.

Testing should be carried out throughout the project life cycle:

- At the start of the project, when choosing the frameworks and development technologies;
- As early as the design phase on models or prototypes;
- During development and production.

The audit sample should include at least the following pages, where available:

- ✓ Home page (the main entry page);
- ✓ Contact page (page containing contact information or form(s) allowing direct contact with the organization responsible for managing the site);
- ✓ Legal notice page;
- ✓ Accessibility page (page containing the accessibility statement);
- ✓ Site map page (page summarizing the site's tree structure or, more broadly, allowing navigation within the various pages);
- ✓ Help page;
- ✓ Authentication page.

In addition to these pages, the following pages should be added, where they exist:

- ✓ At least one relevant page for each type of service provided and any other primary intended use, including search functionality;
- ✓ At least one relevant downloadable document, where appropriate, for each type of service provided and for any other primary intended use;
- ✓ The set of pages that make up a process (e.g. an input form or a multi-page transaction);
- ✓ Examples of pages with a significantly different appearance or type of content (e.g. page containing data tables, multimedia, illustrations, forms, etc.).

In particular, the number of visitors per page can be taken into account when drawing up the sample.

Randomly selected pages representing at least 10% of the pages in the sample.

The average compliance rate of the online service is calculated by averaging the compliance rates of each page.

5.2 User evaluation

To evaluate the accessibility of a site or application, the expert carries out an audit. He/she inspects, manually and with the help of tools, the code and the key pages to verify their compliance with the recommendations in force (RGAA 4.0 and or WCAG 2.1 level AA).

To complete the audit, the expert may carry out tests with users who are recognized as disabled workers.

In ergonomics, "User testing is the leading method for evaluating the user experience of a system in an iterative process. It consists of putting the user in a situation to observe his behavior, his reactions and his performance in carrying out predefined tasks. The data collected provides valuable information about the strengths and weaknesses of the system under evaluation, as well as the experience of the target users. (Lallemand & Gronier, 2016).

Carrying out tests with the application by users with disabilities. Based on tasks to be carried out, to check the level of accessibility from the user's point of view. These tests are intended for a few target users whose workstation (or mobile phone, etc.) is equipped with a particular technical aid (digital screen zoom, screen reader, keyboard). They make it possible to identify usage problems and to validate the criticality of the anomalies encountered during the audits.

The objective of the accessibility expert is not so specific. A test with users will mainly enable him to:

- Confirm and illustrate certain anomalies detected during the audit;
- Ensure that all types of anomaly have been identified (regardless of the page and technical support configuration used);
- Have tangible evidence to prioritize corrections.

To enable non-ergonomic accessibility experts to carry out user tests that are relevant to their needs and reproducible whatever the application, a specific approach has been defined. It is based on a precise protocol.

The protocol is composed of five parts, only the fourth part (Interaction with the application) needs to be contextualized according to the application or site being tested.

1. Welcome the participant (context, objective, instructions);
2. Participant profile :
    - The characteristics of the participant and his/her position;
    - The user experience of the application to be tested.
3. Use test: interaction with the application or site based on scenarios;
4. Back to use:
    - Feelings after interaction (appreciations, difficulties, suggestions);
    - The ergonomic qualities of the application.
5. Conclusion.

Participation in the test is on a voluntary basis. The sample is ideally composed of 4 to 5 users using different technical aids. The tests are carried out at the user's workstation, either face-to-face or remotely. They are not recorded and the anonymity of the participant must be respected.

6 Compliance Toolkit

There are widely used and shared techniques and tools for checking the compliance of a site. This chapter brings together and collates the various web-based tools that can be used to check the compliance of web content with the RGAA 4.1 and WCAG 2.1 standards.

Because no one should have to acquire specific and expensive software to ensure compliance with the regulations, all the tools proposed for testing were chosen for three reasons:

- The reliability of the results and the frequent use of experts in the field;
- Ease of use and exploitation of the results, saving you the installation of complex software;
- Their free, and as far as possible open source, nature does not require you to purchase a license;

Other tools exist, free to be tested and used by everyone.

6.1 W3C validator HTML and CSS**HYPERLINK "http://jigsaw.w3.org/css-validator/"**

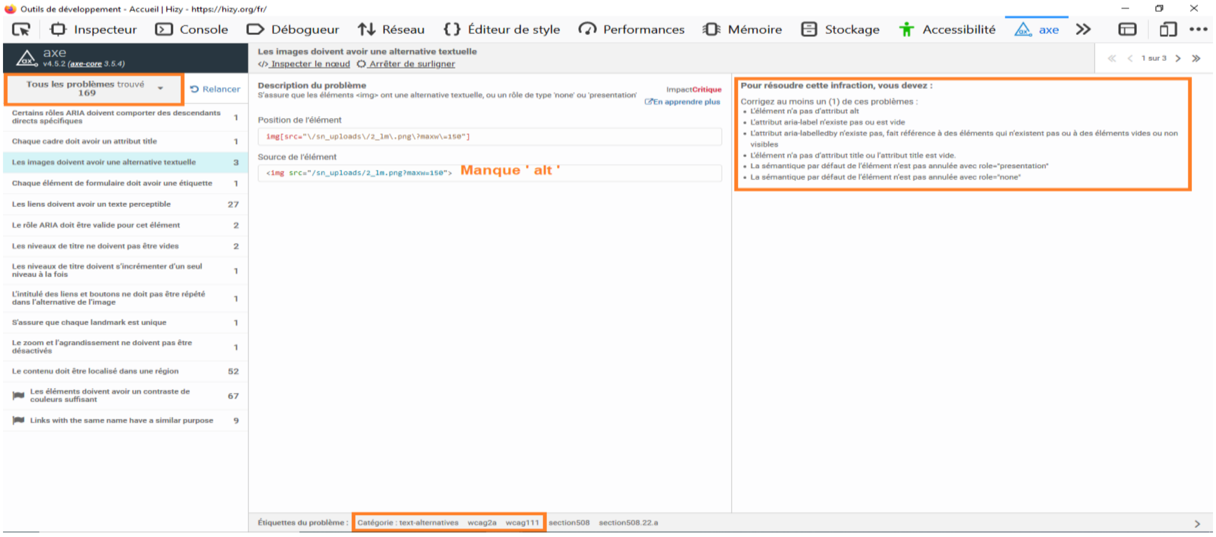
The Markup Validation Service is a World Wide Web Consortium validator that allows Internet users to check HTML and XHTML documents for well-formed markup. Markup validation is an important step in ensuring the technical quality of web pages. This service is most recommended for developers and accessibility experts.

6.2 Power mapper

SortSite - Accessibility Checker and Validator, checks sites against the W3C WCAG accessibility standards. It works on Windows or Mac, there is a 30-day trial version and it scans over 10,000 pages, interesting for accessibility experts.

6.3 Axis

Automated tool to find accessibility issues on the website. The Axis extension is available for Chrome and Firefox, performs a series of tests based on WCAG 2.1 and displays errors in a tab of the Web inspector. The tool is recommended for advanced profiles: developers and accessibility experts.



Certains rôles ARIA doivent comporter des descendants directs spécifiques	1
Chaque cadre doit avoir un attribut title	1
Les images doivent avoir une alternative textuelle	3
Chaque élément de formulaire doit avoir une étiquette	1
Les liens doivent avoir un texte perceptible	27
Le rôle ARIA doit être valide pour cet élément	2
Les niveaux de titre ne doivent pas être vides	2
Les niveaux de titre doivent s'incrémenter d'un seul niveau à la fois	1
L'intitulé des liens et boutons ne doit pas être répété dans l'alternative de l'image	1
S'assure que chaque landmark est unique	1
Le zoom et l'agrandissement ne doivent pas être désactivés	1
Le contenu doit être localisé dans une région	52
Les éléments doivent avoir un contraste de couleurs suffisant	67
Links with the same name have a similar purpose	9

Les images doivent avoir une alternative textuelle

Description du problème
S'assure que les éléments <img> ont une alternative textuelle, ou un rôle de type 'none' ou 'presentation'

Position de l'élément

```
img[src="\/sn_uploads\/2_1m\.png\?maxw\=150"]
```

Source de l'élément

```
<img src="/sn_uploads/2_1m.png?maxw=150">
```

Manque ' alt '

Pour résoudre cette infraction, vous devez :

Corrigez au moins un (1) de ces problèmes :

- L'élément n'a pas d'attribut alt
- L'attribut aria-label n'existe pas ou est vide
- L'attribut aria-labelledby n'existe pas, fait référence à des éléments qui n'existent pas ou à des éléments vides ou non visibles
- L'élément n'a pas d'attribut title ou l'attribut title est vide.
- La sémantique par défaut de l'élément n'est pas annulée avec role="presentation"
- La sémantique par défaut de l'élément n'est pas annulée avec role="none"

Étiquettes du problème : Catégorie : text-alternatives wcag2a wcag111 section508 section508.22.a

6.4 Wave toolbar

Wave is a suite of evaluation tools that helps authors make their web content more accessible to people with disabilities. Wave can identify many of the accessibility errors described in the Web Content Accessibility Guideline (WCAG) and the errors are displayed directly on the page, but it also facilitates human evaluation of web content. Wave toolbar provides a large number of automatic tests, but also a number of semi-manual tests that require human validation, called alerts. This extension also allows you to view the page title hierarchy directly (via a tab in the sidebar), display the page with and without CSS, and view contrast issues via icons located on the page.

6.5 Web Developer

The Web developer extension allows you to add a button to the browser toolbar with various web development tools. This extension is very interesting for developer profiles.

6.6 ￼￼

Many screen readers such as JAWS or NVDA have shortcuts to display a list of landmarks, headings or links. The a11y outline navigation extension provides the same without the need for a screen reader. It is recommended for simulating navigation tests.

This is mainly useful for two audiences:

- Sighted web developers who want to have an idea of how blind users may perceive their content;
- Sighted users who like this type of navigation.

Usage: Pressing Ctrl+Shift+7 opens a dialogue. You can use the up/down arrow keys to select a list. With the tab key, you can switch to the current list and navigate with the arrow keys.

6.7 Bookmarklets

Bookmarklets for accessibility testing use JavaScript to highlight the roles, states and properties of accessibility elements on the page. They are accessible to screen reader users and work on any browser, including mobile phones.
Simply drag the installation link into the favorites bar to use it in any browser.

6.8 Color Contrast Analyzer

The Color Contrast Analyzer (CCA) helps you determine the legibility of text and the contrast of visual elements, such as graphic controls and visual indicators.

6.9 Color Contrast Accessibility Validator

Color Contrast Accessibility Validator is a website that provides free color contrast analysis tools, which will display the problems of a selected web page or color pair; according to WCAG guidelines. Color contrast refers to how light or dark colors appear in relation to each other on screens, including the relative brightness of the grey scale as perceived by the human eye.

6.10 Navigation with a screen reader

A screen reader is text-to-speech software used by visually impaired and blind people. It allows the user to interact with the computer and listen to what is displayed in text form (usually) on the screen. There are several screen readers, the best known being:

- NVDA: free available for Windows.
- JAWS: available for Windows for a fee. In demo mode, you can use it for 40 minutes, after which you will have to restart your computer to use it again.
- Voiceover: available for Mac. It is free and integrated directly into the Mac OS system.

6.11 W3C Checker

W3C checker is a tool to check the compliance of an HTML page with accessibility standards to ensure that the content is accessible to all, it is recommended for developers.

6.12 HeadingsMap

The HeadingsMap extension is very useful for detecting problems with the title hierarchy in one click. It will not, however, tell you anything about missing titles and you will have to complete this analysis with a human review.

7 Technical audit method

This part of the document describes the technical method for project stakeholders, recommended for testing the compliance of Web content with the RGAA and/or WCAG 2.1 level A and AA accessibility criteria. This method of checking compliance allows:

- Checking the compliance of a website with the current standard;
- Limit the scope for interpretation of compliance with the standard;
- Verify, in the context of a label, compliance with a given level of the e-accessible label.

Absolute accessibility, which meets all the requirements and restrictions of disabled Internet users, cannot be achieved in practice. A website can still be a barrier when it meets the highest level of compliance (AAA), particularly with regard to cognitive impairment, language impairment, learning disability and multiple disabilities. W3C, however, recommends that website operators implement as many of the success criteria as possible so that the widest possible group of people can participate in the Internet.

The technical method has 106 control criteria including tests for each criterion. Some tests refer to implementation techniques (HTML, CSS, JavaScript...) to check that the criterion is respected in order to reduce the margin of interpretation as to the respect of accessibility standards. The list of control criteria is structured in 13 themes:

1. Images: images, captcha, icons, graphs;
2. Frames: <iframes> and <frame>;
3. Colors: the colors and contrasts used do not pose a problem for the user;
4. Multimedia: accessibility of audio or video tracks, automatic launch, control mechanism;
5. Tables: formatting and data tables;
6. Links: explicit and with headings;
7. Scripts: accessibility of key features with assistive technologies;
8. Mandatory elements: HTML semantics, title, language, citation, list;
9. Structuring of information: page area, ventilation of texts;
10. Presentation of information;
11. Forms: facilitate filling in, indicate errors to the user;
12. Navigation: elements of navigation and search of a page and more globally of a site, the reading order, good visibility of the focus and absence of keyboard traps;
13. Consultation: Adaptive web, screen orientation and gesture interactions, ASCII art.

For each topic, there are associated control criteria, one or more WCAG 2.1 principles corresponding to the criterion in question and tests and tools for evaluation.

7.1 Images

Give each information-bearing image a relevant textual alternative and a detailed description if necessary.

Criterion 1.1

Does every image that carries information have a textual alternative?

WCAG 2.1 reference: 1.1.1 Non-text Content (A).

Procedure

Examine the "img", role="img" elements of the content:

1. Check that each "img" element that conveys a meaning contains an "alt" attribute;
2. If the image contains words that are important for understanding the content, these words are included in the text alternative.

Result

A replacement text ("alt" attribute of images) is present.

Criterion 1.2

Is every decorative image correctly ignored by assistive technologies?

WCAG 2.1 reference: 1.1.1 Non-text Content (A), 4.1.2 Name, Role, Value (A).

Procedure

For each image that is to be ignored:

Check that the "title" attribute is either missing or empty.

Check that the "alt" attribute is present and empty.

Result

For each decorative image, the "alt" attribute is empty and the "title" attribute is missing.

Criterion 1.3

For each image that carries information and has a textual alternative, is this alternative relevant (excluding special cases)?

WCAG 2.1 reference: 1.1.1 Non-text Content (A), 4.1.2 Name, Role, Value (A).

Procedure

Manually check that the value of the "alt" attributes is relevant:

1. Image link**:** check that the content of the "alt" attribute of each image-link is relevant to its function as a link.
2. Information-bearing image: check that the "alt" attribute of each image is relevant to the image's role in the page.

Result

1. Image link: the target of the link is specified in the "alt" attribute.
2. Information-bearing image: the "alt" attribute contains the information conveyed by the image.

Criterion 1.4

For each image used as a captcha or as a test image, with a text alternative, does the alternative identify the nature and function of the image?

WCAG 2.1 reference: 1.1.1 Non-text Content (A).

Captchas are often the source of difficulties for users in general.

Procedure

Examine the images used as CAPTCHA or as test images:

1. Remove, hide or conceal the CAPTCHA;
2. Replace it with the alternative text.

Result

The text alternative describes the purpose of the CAPTCHA.

Criterion 1.5

For each image used as a CAPTCHA, is there an alternative access to the content or function of the CAPTCHA?

WCAG 2.1 reference: 1.1.1 Non-text Content (A).

Captchas based on a single resolution system should offer an alternative to allow the largest number of users to access them.

Procedure

For each CAPTCHA in a web page, check that the web page contains another CAPTCHA with the same purpose but using a different modality.

To solve the captcha, users can choose between an audio or visual system.
If the captcha is only audio or only visual, an alternative is present in the form of a logical test (a question with an obvious answer, a simple mathematical test). Thus, users without visual information should be able to solve the captcha by an audio system; similarly, those without auditory information should be able to solve it visually.

Result

The page offers a combination below:

1. Audio + visual captcha.
2. Audio or visual captcha + logical tests (question with an obvious answer, simple mathematical test).

Criterion 1.6

Is there a detailed description of each information-bearing image, if necessary?

WCAG 2.1 reference: 1.1.1 Non-text Content (A).

Procedure

Check that the "img" element has a "longdesc" attribute:

1. Check that the value of the "longdesc" attribute is a valid Uniform Resource Identifier (URI) of an existing resource.
2. Check that the target content of this URI contains a long description describing the original non-textual content associated with it.

Ex: <p><img src="chart.gif" alt="a complex chart" **longdesc="chartdesc.html"**/></p>

Result

The URI link is valid, links directly to the long description of the non-textual content and the long description contains the same information as the non-textual content.

Criterion 1.7

For each information-bearing image with a detailed description, is the description relevant?

WCAG 2.1 reference: 1.1.1 Non-text Content (A).

Procedure

For any non-textual content requiring a long description:

1. View long description;
2. Check that the long description serves the same purpose or presents the same information as the non-textual content.

Result

If step 2 is correct, then the success criterion is validated.

Criterion 1.8

If there is no replacement mechanism, each text image carrying information should be replaced by styled text if possible. Is this rule respected (except in special cases)?

Reference WCAG 2.1: 1.4.5 Images of Text (AA).

Procedure

Check that the alt attribute includes at least the text of the image (diagram, graph).

Check that the web page has a control that allows users to select an alternative layout, and when the control is activated, the resulting page includes text (programmatically determined text) where text images had been used.

Disable page styles to check that there is no loss of information due to the absence of background images (CSS).

Result

Presence of an alternative text image that provides information.

Lack of information-carrying CSS images.

Criterion 1.9

Is each image caption, if necessary, correctly linked to the corresponding image?

WCAG 2.1 reference: 1.1.1 Non-text Content (A), 4.1.2 Name, Role, Value (A).

Procedure

Examine the target page to determine if it has an <figure> tag and corresponding <figcaption> caption.

```
Ex:   <figure>
          <img src=" image.jpg" alt=""/>
          <figcaption>Related caption</figcaption>
      </figure>
```

Result

If a <figure> tag is present and relevant, then the checkpoint is validated.

7.2 Frames

Give each frame and each online frame a relevant title.

Criterion 2.1

Does every manager have a title of manager?

Reference WCAG 2.1: 4.1.2 Name, Role, Value (A).

Procedure

Launch aXe , Web developer or the "Iframes" bookmarklet:

Check that all iframes and frames contain a non-empty "title" attribute.

Result

No errors related to the lack of a "title" attribute on frame and iframe.

Criterion 2.2

For each manager with a managerial title, is the managerial title relevant?

Reference WCAG 2.1: 4.1.2 Name, Role, Value (A).

Procedure

Launch one of the code inspectors; axe, Web developer or the "Iframes" bookmarklets: examine the content of the "title" attribute of the frames.

Result

The value of the "title" attribute correctly describes the content or function of the iframe or frame.

7.3 Colors

Do not give information by color alone and use sufficiently high color contrast.

Criterion 3.1

In every web page, information should not be given only by color. Is this rule respected?

Reference WCAG 2.1: 1.3.1 Info and Relationships (A), 1.4.1 Use of color (A).

Procedure

1) Examine the target page to identify where color is used to communicate a message, for example:

- Communicate the status of a task or process.
- To indicate the status of a UI component (as selected or targeted).
- Suggest an answer.
- Identification of an error.

For each case, check that at least one of these visual alternatives is also provided:

- On-screen text that identifies the color itself and/or describes the meaning conveyed by the color.
- Visual differentiation (e.g. shape, position, size, underlining) and a clear indication of its meaning.

2) Examine the target page to identify where instructions refer to the sensory characteristics of an item, e.g.:

- Color;
- Shape;
- Size;
- Visual location;
- Orientation and sound.

For each case, check that the instructions also include sufficient additional information to locate and identify the item without knowing its sensory characteristics. (For example, "Press the green button").

3) Interact with the target page to identify any instances where the system provides auditory cues, such as:

- A tone that indicates the successful completion of a process.
- A tone that indicates the arrival of a message.

For each case, check that the system also provides a visible marker.

Result

In all cases, check whether the information conveyed visually is also explicitly mentioned in the text, to validate this criterion.

Criterion 3.2

In each web page, is the contrast between the color of the text and the color of its background sufficiently high (excluding special cases)?

Reference WCAG 2.1: 1.4.3 Contrast (Minimum) (AA).

Procedure

Run one of the Wave, axe, WCAG - Color contrast checker extensions or use the Color Contrast Analyzer application.

The Wave extension can analyze each page to identify an area where the text and background are most likely to have a low contrast ratio (for example, white text on a light grey background).

Check that each page meets these contrast thresholds:

- Ordinary text should have a ratio ≥ 4.5.
- Large text (18pt or 14pt+bold) should have a ratio ≥ 3.0.

Result

The contrast ratio is in accordance with the RGAA recommendations.

Exception

This recommendation does not apply in the following cases:

- If the image is a logo;
- If a text, such as a label or a data table, provides the same information as the icon or graphic;

- If the lack of contrast in the image does not prevent understanding of the content or its function;
- If the components are inactive.

Criterion 3.3

In each web page, are the colors used in the interface components or graphic elements carrying information sufficiently contrasting (except in special cases)?

WCAG 2.1 Reference: 1.4.11 Non-text Contrast (AA).

Procedure

On the target page, examine each highlighted item in its normal state (not disabled or selected, no mouse over).

Use the color contrast analyzer to check that visual presentations have a contrast ratio of at least 3:1 against the background:

- Any visual boundary that indicates the clickable area of the component.
- Any visual effect that indicates that the component is in its normal state.

Result

The contrast ratio is in accordance with the RGAA recommendations.

Exception

A lower contrast ratio is allowed if one of the following conditions is met: the component is inactive/deactivated; solely the browser determines the appearance of the component.

7.4 Multimedia

Provide each temporal media with a textual transcript, synchronized subtitles and a relevant synchronized audio description if necessary. Provide each non-temporal media with a relevant textual alternative.

Make it possible to control the viewing of each temporal and non-temporal media with the keyboard and ensure their compatibility with assistive technologies.

Criterion 4.1

Does each pre-recorded time-lapse medium have a textual transcription or audio description, if necessary (except in special cases)?

Reference WCAG 2.1: 1.2.1 Audio-only and Video-only (Prerecorded) (A), 1.2.3 Audio Description and Media Alternative (Prerecorded) (A).

Procedure

Examine the target page for any pre-recorded audio or video content:

1. The audio content alone must be accompanied by :
   - Simple static visuals, such as the title of a speech and the name of the speaker;
   - From a textual transcription ;
   - In addition, the transcription must be compliant and provide an accurate and complete description of the audio content.
2. Video-only content includes video content accompanied by sound (such as background music) that does not contribute to comprehension, determine whether video-only content is accompanied by at least one of these alternatives:
   - A textual transcript, subtitles are provided in text format and are not embedded directly into the video.
   - Audio description is available for videos with information in visual form.

An audio description is used to describe information that can only be accessed through the image, thus complementing the audio tape for, for example, the visually impaired. The subtitles allow all the information in the soundtrack to be rendered. They must be provided in text format so that they can be adapted to the needs of users (color, size).

Results

For all audio and video content, a full transcript (text containing all visual and auditory information) is present on the page or accessible via a link nearby.
Subtitles in text format are present for all videos and reproduce all the information in the soundtrack.
An audio description is available to complement the audio tape.

Criterion 4.2

For each pre-recorded time-lapse media with a synchronized text transcript or audio description, are these relevant (excluding special cases)?

Reference WCAG 2.1: 1.2.1 Audio-only and Video-only (Prerecorded) (A), 1.2.3 Audio Description and Media Alternative (Prerecorded) (A).

Procedure

1. Audio content only:
   - Activate the text transcript;
   - Compare the transcript with the audio track;
   - Listen to the synchronized media;
   - Check for gaps in the dialogue that prevent important information about the visual content from being conveyed.
2. Video content only:
   - Activate audio description from the content/player;
   - Check that the audio description provides an accurate and complete description of the video content, including information about actions, characters, scene changes and on-screen text;
   - Check that the audio description is broadcast and relevant to the content of the video.

Result

If all conditions are valid for each case, then criterion 4.2 is valid.

Criterion 4.3

Does each pre-recorded time-synchronized media have synchronized subtitles, if required (except in special cases)?

Reference WCAG 2.1: 1.2.2 Captions (Prerecorded) (A).

Procedure

Examine the target page to identify any pre-recorded multimedia content (synchronized video and audio), to check that the <track> tag is present, and that it has a kind="captions" attribute.

Result

Presence of the <track> tag and that it has a kind="captions" attribute in the time-based media code.

Criterion 4.4

For each pre-recorded time-synchronized media with synchronized subtitles, are these subtitles relevant?

Reference WCAG 2.1: 1.2.2 Captions (Prerecorded) (A).

Procedure

Examine the target page to identify any pre-recorded multimedia content (synchronized video and audio):

- Activate the subtitles and then play the multimedia content;
- Check that subtitles provide an accurate and complete description of the audio content, including the identity of the speaker, the speech and the relevant sounds;
- Check that the subtitles do not obscure or hide relevant information in the video.

Results

Subtitles (of all dialogue and important sounds) are visible and relevant.

Criterion 4.5

Does each pre-recorded time-lapse media have, if necessary, a synchronized audio description (excluding special cases)?

WCAG 2.1 Reference: 1.2.5 Audio Description (Prerecorded) (AA).

Procedure

Examine the target page to identify any pre-recorded multimedia content with audio description:

- Activate the audio descriptions and play the multimedia content;
- Check that the audio description is synchronized with the video content;
- Check that the audio description does not conflict with the relevant information on the soundtrack (the descriptive narration should be added during existing pauses in the dialogue).

Result

Presence of an audio description for each pre-recorded time-lapse video media only.

Criterion 4.6

For each pre-recorded time-lapse media with a synchronized audio description, is it relevant?

WCAG 2.1 Reference: 1.2.5 Audio Description (Prerecorded) (AA).

Procedure

Verify that each pre-recorded time-lapse media only video, synchronized audio description is present and relevant.

Result

Presence of a relevant audio description for each pre-recorded video-only time-lapse media.

Criterion 4.7

Is each time medium clearly identifiable (except in special cases)?

Reference WCAG 2.1: 1.1.1 Non-text Content (A).

Procedure

Check that for each time-based media, whether sound only, video-only or synchronized, the adjacent textual content clearly identifies the time-based media in question.

Result

The environment introduces the content of the temporal medium.

Criterion 4.8

Does each non-temporal medium have an alternative, if necessary (except in special cases)?

WCAG 2.1 reference: 1.1.1 Non-text Content (A).

Procedure

For each non-temporal medium:

- A clearly identifiable adjacent link or button contains the address (URL) of a page containing an alternative.
- A clearly identifiable adjacent link or button provides access to an alternative on the page.

- The page referenced by the adjacent link or button is accessible.
- The alternative in the page, referenced by the adjacent link or button, is accessible.

Result

Presence of a link or button to access the alternative version of the time-based media content.

Criterion 4.9

For each non-temporal medium that has an alternative, is that alternative relevant?

WCAG 2.1 reference: 1.1.1 Non-text Content (A).

Procedure

Check for each non-temporal medium that has an alternative, that alternative provides access to the same content and similar functionality as the medium in question.

Result

Presence of a relevant alternative.

Criterion 4.10

Is each automatically triggered sound controllable by the user?

WCAG 2.1 reference: 1.4.2 Audio Control (A).

Procedure

For each sound sequence automatically triggered via a <object>, <video>, <audio>, <embed>, <bgsound> tag or a JavaScript code check the conditions:

- The sound sequence has a duration of 3 seconds or less;
- The sound sequence can be stopped by the use;
- The user independently of the system volume control can control the volume of the sound sequence.

Result

No audio or video tracks are automatically launched when a page is loaded.
Pause, play and volume controls are associated with all audio and video tracks.

Criterion 4.11

Is the consultation of each time medium, if necessary, controllable by the keyboard and any pointing device?

WCAG 2.1 reference: 2.1.1 Keyboard (A), 2.1.2 No Keyboard Trap (A).

Procedure

Manually check that it is possible to stop/resume the playback of an audio or video track from a navigation with the mouse.
Manually check that it is possible to stop/resume playback of an audio or video track from a keyboard navigation.
A mechanism for controlling the audio or video track is present. This mechanism allows the media to be played or paused, regardless of the technical aid used.

Result

Each feature is accessible via keyboard and other pointing devices.

Criterion 4.12

Is the viewing of each non-temporal media controllable by the keyboard and any pointing device?

WCAG 2.1 reference: 2.1.1 Keyboard (A), 2.1.2 No Keyboard Trap (A).

Procedure

Check that all functionality is accessible using only the keyboard, or other pointing device.

Result

The keyboard and other assistive devices can activate each feature.

Criterion 4.13

Is every temporal and non-temporal media compatible with assistive technologies (except in special cases)?

WCAG 2.1 reference: 4.1.2 Name, Role, Value (A).

Procedure

Each temporal and non-temporal medium satisfies one of these conditions:

- The name, role, value, setting and state changes of interface components are accessible to assistive technologies via an accessibility API.
- An alternative that is compatible with an accessibility API provides access to the same functionality.

Each temporal and non-temporal media that has an alternative that is compatible with assistive technologies satisfies one of these conditions:

- The alternative is adjacent to the temporal or non-temporal media.
- The alternative is accessible via an adjacent link or button.
- A mechanism is provided to replace the temporal or non-temporal media with its alternative.

Result

All time-based media work with assistive technologies.

7.5 Tables

Give each complex data table a relevant summary and title, clearly identify the header cells, use a relevant mechanism to link data cells to header cells. For each formatting table, ensure that it is properly linearized.

Criterion 5.1

Does each complex data table have a summary?

Reference WCAG 2.1: 1.3.1 Info and Relationships (A).

Procedure

Use the Web Developer extension (information > Display Table Information) to delimit display the attributes of the <table> tag.

For each data table:

- If a "summary" attribute is present, check that the "summary" attribute describes the organization of the table or explains how to use the table;
- If a "summary" attribute and a <caption> element are both present for the data table, check that the "summary" does not duplicate the <caption>.

Result

Presence of a "summary" attribute for each complex data table.

Criterion 5.2

For each complex data table with a summary, is the summary relevant?

WCAG 2.1 reference: 1.3.1 Info and Relationships (A).

Procedure

Use the Web Developer extension (Information > Display Table Information) to delimit the "summary" attribute of each table.

Examine each underlined 'summary' attribute to determine the relevance of the summary.

Result

Presence of a relevant "summary" attribute for each complex data table.

Criterion 5.3

For each formatting table, is the linearized content still comprehensible (except in special cases)?

Reference WCAG 2.1: 1.3.2 Meaningful Sequence (A), 4.1.2 Name, Role, Value (A).

Procedure

Use the Web Developer extension (Outline > Outline Tables) to identify any <table> elements in the target page.

If you find a table, determine whether it is a layout table: a layout table uses rows and columns to visually position content without involving relationships.

If you find a presentation table, use the Web Developer browser extension (Miscellaneous > Linearize page) to display the page in DOM order, the <table> tag has a role="presentation" attribute.

Check that the contents of the layout tables are always in the correct reading order when the page is linearized.

Result

The presentation table has a correct reading order when the page is linearized and a role="presentation" attribute.

Criterion 5.4

For each data table with a title, is the title correctly associated with the data table?

WCAG 2.1 reference: 1.3.1 Info and Relationships (A).

Procedure

Use the Web Developer extension (Outline > Outline tables captions) to delimit <caption> of each table.

Result

Presence of the <caption> tag for each data table.

Criterion 5.5

For each data table with a title, is it relevant?

WCAG 2.1 reference: 1.3.1 Info and Relationships (A).

Procedure

Use the Web Developer extension (Outline > Outline tables captions) to delimit <caption> of each table.

Examine each underlined <caption> to determine the relevance of the caption.

Result

Presence of the relevant <caption> tag for each data table.

Criterion 5.6

For each data table, are each column header and each row header correctly reported?

WCAG 2.1 reference: 1.3.1 Info and Relationships (A).

Procedure

For all data tables, check whether the table headers can be correctly determined using one of the following mechanisms:

- Headers marked with array header elements <th>;
- Scope attributes for tables with more than one row or column of headers;
- Scope attributes for tables with more than one row or column of table headers;
- Headers and associated data cells using headers and id attributes;
- Headers marked as <td> elements with the "scope" attribute;
- Headers marked with the ARIA role attributes "rowheader" or "columnheader";
- Check that all id's are unique (i.e. no two elements on the page have the same id);
- Check that all "scope" attributes have the value "row", "col", "rowgroup" or "colgroup".

Result

Each complex data table is correctly coded for accessibility.

Criterion 5.7

For each data table, is the appropriate technique used to associate each cell with its headers (excluding special cases)?

WCAG 2.1 reference: 1.3.1 Info and Relationships (A).

Procedure

For each data table:

- Check that all elements have a "scope" attribute;
- Check that all td elements that serve as headers for other elements have a "scope" attribute;
- Check that all "scope" attributes have the value "row", "col", "rowgroup" or "colgroup".

➢ **One entry array:** check for the presence of <th> headers for simple arrays;

➢ **Table with two entries**: for each <th> check for the presence of a scope='col' attribute for column headers, and the presence of a scope='row' for row headers;

➢ **Complex tables:** check for the presence of <th> headers or "id" and "headers" attributes.

A different background color for every second row makes it easier to consult a row in a multiline table.

Result

Each complex data table is correctly coded to be accessible:

- Simple tables have <th> headers;
- Double entry arrays have <th> headers as well as scope='col' and scope='row' attributes;
- Complex arrays have headers (<th> and "scope" attributes) or complex headers based on the use of "id" and "headers" attributes;
- Complex tables have a relevant summary.

Criterion 5.8

Each formatting table should not use elements specific to data tables. Is this rule respected?

WCAG 2.1 reference: 1.3.1 Info and Relationships (A).

Procedure

Examine the document source code for the table element.

If the table is only used to visually format elements of the content:

- Check that the table does not contain any "th" elements;
- Check that the table element does not contain a "summary" attribute;
- Check that the table element does not contain a <caption> caption element;

Result

A formatting table is correctly coded.

7.6 Links

Provide explicit link headings, including contextual information, and use the link title. Duplicate with links or a navigation form the responsive areas of a server-side clickable image.

Criterion 6.1

Is each link explicit (except in special cases)?

WCAG 2.1 reference: 1.1.1 Non-text Content (A), 2.4.4 Link Purpose (In Contact) (A), 2.5.3 Label in Name (A).

Procedure

For each link in the content:

1. Check that the text or a text alternative for the non-textual content is included in the <a> element:
   - If an <img> element is the only content of the <a> element, check that its text alternative describes the purpose of the link;
   - If the <a> element contains one or more <img> elements and the text alternative of the <img> element(s) is empty, check that the link text describes the purpose of the link;
   - If the <a> element contains only text, check that the text describes the purpose of the link.
2. For each link that has an "aria-labelledby" attribute:
   - Check that each "id" of the "aria-labelledby" attribute value corresponds to an "id" of a text element used as part of the link's purpose;
   - Check that the combined value of the text referenced by one or more "id" in the "aria-labelledby" attribute correctly describes the purpose of the link element.

Example:

```
<h2 id="headline">Storms hit east coast</h2>

<p>Torrential rain and gale force winds have struck the east coast, causing flooding in many coastal towns
    <a id="p123" href="news.html" aria-labelledby="p123 headline">Read more…</a>
</p>
```

3. For link elements that use the "aria-label": check that the value of the "aria-label" attribute correctly describes the purpose of the link element.

Result

If all the conditions for each case are met then the criterion is valid.

Criterion 6.2

In each web page, does every link, except for anchors, have a title?

WCAG 2.1 reference: 1.1.1 Non-text Content (A), 2.4.4 Link Purpose (In Contact) (A), 2.5.3 Label in Name (A).

Procedure

Inspect the text label of the links; inspect the "aria-label" and "title" attributes.

- Check that there are no identical labels pointing to different destinations;
- Check that it is possible to determine the action or destination of a link;
- The links are understandable out of context.

Result

Presence of a relevant title for each link.

7.7 Scripts

Give each script a relevant alternative if necessary. Make each script code controllable at least by keyboard and mouse and ensure compatibility with assistive technologies.

Criterion 7.1

Is each script, if necessary, compatible with assistive technologies?

WCAG 2.1 reference: 2.5.3 Label in Name (A), 4.1.2 Name, Role, Value (A).

Procedure

Parse the source code for elements that have been assigned event handlers in the markup or by scripting (indicating that the element is a user interface control):

- Check if the role of the control is already natively defined in the markup language;
- Check whether another valid method, such as assigning an appropriate WAI-ARIA role, has been used to define the control's role.

For a user interface component using the WAI-ARIA role, attribute:

- Check that the required states and properties for the role are present;
- Check that no WAI-ARIA states or properties that are not required, supported or inherited are present;
- Check that state and property values are updated to reflect the current state when the UI component changes state.

Result

Each script is compatible with assistive technologies.

Criterion 7.2

For each script with an alternative, is the alternative relevant?

WCAG 2.1 reference: 1.1.1 Non-text Content (A), 4.1.2 Name, Role, Value (A).

Procedure

Every script starting with the <script> tag and having an alternative checks one of these conditions:

- The alternative between <noscript> and </noscript> provides access to similar content and functionality;
- The page displayed, when JavaScript is disabled, provides access to similar content and functionality;
- The alternative page provides access to similar content and functionality;
- The server-side scripting language provides access to similar content and functionality;
- The alternative on the page provides access to similar content and functionality.

Each non-textual element updated by a script (in the page, or in a frame) and having an alternative, which checks these conditions:

- The alternative of the non-textual element is updated;
- The updated alternative is relevant.

Result

The alternative proposed for each script is relevant.

Criterion 7.3

Is each script controllable by the keyboard and by any pointing device (except in special cases)?

WCAG 2.1 reference: 1.3.1 Info and Relationships (A), 2.1.1 Keyboard (A), 2.4.7 Focus Visible (AA).

Procedure

Each element with an event handler controlled by a script must check one of these conditions:

- The element is accessible via the keyboard and any pointing device;
- An element accessible by the keyboard and any pointing device allowing the same action is present in the page.

A script must not remove the focus of an element that receives it.

Result

All functionality is accessible using only the keyboard and any pointing device.

Criterion 7.4

For each script that initiates a context switch, is the user notified or have control?

WCAG 2.1 reference: 3.2.1 On Focus (A), 3.2.2 On Input (A).

Procedure

Each script that initiates a context switch must check one of these conditions:

- The user is notified by text of the script action and the type of change before it is triggered;
- An explicit button initiates the context switch (<input> of type "submit", "Button" or "img" or <button> tag);
- The context switch is initiated by an explicit link.

Result

Each change of context is reported to the user.

Criterion 7.5

In each web page, are status messages correctly rendered by assistive technologies?

WCAG 2.1 reference: 4.1.3 Status Messages (SA).

Procedure

Examine the target page to determine if it generates status messages. A status message provides information to the user about one of the following:

- The success or results of an action;

- The pending status of an application;
- The progress of a process;
- The existence of errors.

Inspect the HTML of the page to identify an empty container with one of the following attributes:

- role=" alert";
- role=" log";
- role="progressbar";
- aria-live="assertive".

Trigger the action that generates the status message:

- Check that the status message is entered in the container;
- Determine that an empty "role=alert" or "aria-live=assertive" error container is present in the DOM when the page loads;
- Trigger the error that causes content to appear or update in the live region;
- Determine that the error message has been injected into the error container already present.

For each status message:

- Check that the container intended to hold the status message has a role attribute with a status value before the status message appears;
- Check that when the status message is triggered, it is inside the container;
- Check that elements or attributes that provide information equivalent to the visual experience of the status message (such as a shopping cart image with appropriate alternative text) also reside in the container.

On a page, that contains sequentially updated information:

- Check that the information container has a "log" role;
- Use dialogue boxes to interact with the user, for example, to alert them or ask for confirmation. Keep messages short and clear;
- Check that a mechanism based on WAI-ARIA roles (status, alert, alertdialog...) or aria-live dynamic regions is implemented.

Result

The nature of the status message implicitly corresponds to the action. Messages are automatically vocalized.

7.8 Mandatory elements

Check that each web page contains the following mandatory elements:

- A valid source code according to the type of document;
- A relevant title and a default language indication;
- Tags are not only used for presentation;
- Changes in language and direction of reading are indicated.

Criterion 8.1

Is each web page defined by a document type?

WCAG 2.1 reference: 4.1.1 Parsing (A).

Procedure

Inspect the source code of the target page:

- For each web page, the document type (<doctype> tag) is present;
- For each web page, the document type (doctype tag) is valid;
- For every web page with a document type declaration, is it located before the <html> tag in the source code.

Result

Presence of the <DOCTYPE> tag located before the <html> tag.

Criterion 8.2

For each web page, is the generated source code valid for the specified document type (excluding special cases)?

WCAG 2.1 reference: 4.1.1 Parsing (A), 4.1.2 Name, Role, Value (A).

Procedure

Inspect the source code of the target page.

This test uses the Nu HTML checker and the corresponding bookmarklets.

Open the About Nu HTML Checker page in a new browser window.

Add these two bookmarklets to your browser's favorites:

1. Check the serialized DOM of the current page;
2. Checking the conformity of the WCAG analysis.

Mouse users can simply drag the relevant links from the page to the browser's bookmarks bar.

Run the first bookmarklets in the browser tab containing your target page. It will send the DOM of the page to the checker and display the results in a new browser tab.

Launch the second bookmarklets in the browser tab containing the checker results. It will filter the results to show only WCAG analysis errors.

Examine the filtered results to check for errors:

- Missing start or end tags;
- Duplicate "id" attributes;
- In addition, incorrect nesting of elements;
- No attribute appears more than once on the same element.

Result

The source code is free of syntax errors.

Criterion 8.3

Is the default language present on each web page?

Reference WCAG 2.1: 3.1.1 Language of Page (A).

Procedure

Use bookmarklets Lang.

Examine the target page to determine its primary language.

Inspect the <html> tag on the page to ensure that it has the correct language attribute.

Result

The language of each page is correctly implemented in the <html> tag.

Criterion 8.4

For each web page with a default language, is the language code relevant?

Reference WCAG 2.1: 3.1.1 Language of Page (A).

Procedure

Look at the html element of the document:

- Check that the <html> element has a "lang" attribute;
- Check that the value of the "lang" attribute is correct and reflects the main language used by the web page.

Result

The language code is relevant to the content of each page. The value of the "lang" attribute corresponds to the main language of the document

Criterion 8.5

Does each web page have a page title?

Reference WCAG 2.1: 2.4.2 Page Titled (A).

Procedure

Inspect the source code and check the <title> tag:

- The title of the page is the first element read by the speech synthesis;
- Page titles allow the user to know where they are in the site tree;
- Examine the source code of the document and check that a non-empty <title> element appears in the <head> section;
- Check that the <title> element describes the document.

Result

Presence of a title for each page.

Criterion 8.6

For each web page with a page title, is the title relevant?

Reference WCAG 2.1: 2.4.2 Page Titled (A).

Procedure

Use the Page Title bookmarklets to check the relevance of the title:

- Each title is specific and unique to the content of the page. It is written from the most precise information to the most general;
  Example: [name of current page - name of site];
- Check that the title of the page is unique; it should allow the user to understand the context of the page;
- To be effective, the page title must be unique and represent the content of the page;
- Each page has a unique and descriptive title, globally from the most precise to the most general.

Example: [summary of page content - site name].

Result

Presence of a relevant title for each page.

Criterion 8.7

In each web page, is each language change indicated in the source code (except in special cases)?

Reference WCAG 2.1: 3.1.2 Language of Parts (AA).

Procedure

Examine the target page to identify any passages in a language other than the default language of the page. Use bookmarklet Lang.

If you find such a passage, examine the HTML of the containing element to check that it has the correct language attribute.

Result

The change of language is mentioned in the source code of each passage. The "lang" attribute present in the element encapsulating the text.
Example: <q lang='en'> [...] </q>.

Exception

For foreign language words or expressions that have passed into common usage (Google, newsletter) or proper nouns, do not indicate a change of language.

Criterion 8.8

In each web page, is the language code for each language change valid and relevant?

Reference WCAG 2.1: 3.1.2 Language of Parts (AA).

Procedure

Examine the target page to identify any passages in a language other than the default language of the page. Use bookmarklet Lang.

If you find such a passage, examine the HTML of the containing element to check that it has the correct language attribute.

Result

The language change is valid and relevant in the source code of each passage. The "lang" attribute present in the element encapsulating the text. Example: <q lang='en'> [...] </q>.

Criterion 8.9

In every web page, tags should not be used for presentation purposes only. Is this rule respected?

WCAG 2.1 reference: 1.3.1 Info and Relationships (A).

Procedure

In every web page, the tags (with the exception of <div>, <span> and <table>) should not be used solely for presentation purposes.

Result

Absence of tags used for presentation purposes only.

Criterion 8.10

Are changes in the reading direction on each web page indicated?

WCAG 2.1 reference: 1.3.2 Meaningful Sequence (A).

Procedure

Examine the target page to determine whether it uses a right-to-left script (Arabic, Hebrew, Persian or Urdu).

If the page or passage uses right-to-left scripting, examine the HTML of the containing element to ensure that it has the correct direction attribute (dir="rtl" or "ltr") and the value of the "dir" attribute is relevant.

Result

Each change of reading direction is correctly indicated attribute "dir".

7.9 Structuring information

Use headings, lists, abbreviations and quotations to structure information.

Criterion 9.1

In each web page, is the information structured by the appropriate use of headings?

Reference WCAG 2.1 :1.3.1 Info and Relationships (A), 2.4.1 Bypass Block (A), 2.4.6 Headings and Labels (AA), 4.1.2 Name, Role, Value (A).

Procedure

In each target page, examine the hierarchy of titles with the HeadingsMapp extension:

- The content of the page is structured by means of the < hn> tag and tag level titles with a WAI-ARIA role="heading" attribute associated with a WAI-ARIA aria-level attribute;
- The titles follow one another in a logical way;
- Each title is relevant to the content that follows it;
- The titles are relevant and not empty;
- The absence of jumps in the title hierarchy.

Structuring a page well makes it easier to understand and optimizes reading.

The number of headings depends on the density of information on the page: the absence of headings can be annoying, but, conversely, an abundance of headings can be counterproductive.

Level jumps, e.g. from an "n" level title to an "n+2" level title, make no sense and can disrupt the reading order.

Headings guide and facilitate reading, provided they are consistent with the content they precede.

Result

All content, treated visually as headings, has title semantics (<hn> tags and tag with a WAI-ARIA role="heading" attribute associated with a WAI-ARIA aria-level attribute).

The level headings are structured in a logical way.

Criterion 9.2

In each web page, is the structure of the document coherent (excluding special cases)?

WCAG 2.1 reference: 1.3.1 Info and Relationships (A).

Procedure

This test uses the Web Developer browser extension and the Landmarks bookmarklets to check that the document structure meets these conditions:

- The header area of the page is structured via a <header> tag;
- The main and secondary navigation areas are structured via a <nav> tag;
- The <nav> tag is reserved for structuring the main and secondary navigation areas;
- The main content area is structured via a <main> tag;
- The document structure uses a single visible <main> tag;
- The footer area is structured via a <footer> tag.

Result

For each region with a semantic function, a semantic tag corresponds to it.

Exception

When the doctype declared in the page is not the HTML5 doctype, this criterion is not applicable.

Criterion 9.3

Is every list on every web page properly structured?

WCAG 2.1 reference: 1.3.1 Info and Relationships (A).

Procedure

Examine the target page to identify any content that appears to be a list and run the Wave extension.

A list is a set of related items displayed consecutively.

The elements of a list are usually, but not always, stacked vertically.

Examine the HTML code of each list to ensure that it is semantically correct:

- An unordered list (simple or bulleted) must be contained within an <ul> element;
- An ordered (numbered) list must be contained within an <ol> element;

- A descriptive list (a set of terms and definitions) must be contained in an <dl> element;
- Check for the absence of a single item list;
- Check for an empty list.

Result

Existence of relevant lists.

Criterion 9.4

Is every quotation on every web page correctly indicated?

WCAG 2.1 reference: 1.3.1 Info and Relationships (A).

Procedure

Inspect the source code for <blockquote>, <em> :

- Examine the target page to identify visually accented words or phrases;
- Inspect the HTML for each accented word or phrase to check that it is contained in a <em> or <strong> element;
- Search the HTML of the page to determine if the page includes <blockquote> elements;
- Examine each <blockquote> element to check that it contains a quote.

Result

Citations are correctly coded.

7.10 Presentation of information

Use style sheets to control the presentation of information. Check the effect of larger font sizes on readability. Ensure that links are correctly identifiable, that focus is indicated, that line spacing is sufficient and that the user can control the justification of texts. Ensure that hidden text is correctly rendered and that information is not given solely by the shape or position of an element.

Criterion 10.1

In the website, are CSS style sheets used to control the presentation of information?

Reference WCAG 2.1: 1.3.1 Info and Relationships (A), 1.3.2 Meaningful Sequence (A).

Procedure

For content, that uses CSS for positioning, remove the style information from the document or disable the use of style sheets in the Web developer extension:

- Check that the structural relationships and the meaning of the content are preserved;
- Check that structural information and functionality are explicitly provided and that they are logically separated from presentation information;
- For each word that appears to have non-standard spacing between characters, check whether any words in the content text contain spacing characters;
- Check whether columns are created using white space characters to format the information;
- Check whether the 'pre' element is used: for each occurrence of the 'pre' element, check whether the attached information is presented in table form.

Result

Structural and functional information is explicitly provided and is logically separated from presentation information.

Criterion 10.2

In each web page, is visible content still present when style sheets are disabled?

WCAG 2.1 reference: 1.1.1 Non-text Content (A), 1.3.1 Info and Relationships (A).

Procedure

With the extension Web developer. Examine all images added to content via CSS or HTML style attributes, or dynamically in the script as background images:

- Check that the images do not convey important information;
- If an image conveys important information, this information is provided to assistive technologies and is available when the CSS image is not displayed.

Result

No loss of information when CSS is disabled.

Criterion 10.3

In each web page, is the information still understandable when style sheets are disabled?

WCAG 2.1 reference: 1.3.2 Meaningful Sequence (A), 2.4.3 Focus Order (A).

Procedure

For each target page, with the Web developer extension deactivate the CCS:

- Determine the order of the interactive elements in the content;
- Determine the logical order of the interactive elements;
- Check that the order of the interactive elements in the content is the same as the logical order.

Result

The content of the target page remains understandable when CSS is disabled.

Criterion 10.4

In each web page, is the text still readable when the font size is increased to at least 200% (excluding special cases)?

WCAG 2.1 reference: 1.4.4 Resize Text (AA).

Procedure

With the Firefox browser: zoom option, check text zoom, and use the browser settings to set the target page zoom to 200%. Examine the target page to verify this:

- All text is fully resized, including text in form fields;
- The text is not cut, truncated or obscured;
- All content remains available;
- All features remain available.

The page does not contain in the <meta name= 'viewport'> tag:

- The parameter user-scalable='no';
- In addition, the maximum-scale parameter is not less than 2.

Check that the value of the CSS property that defines the font size is a percentage or expressed in em units, and non-relative units (pt, pc, mm, cm, in) should not be used for the media types screen, tv, handheld, projection.

Result

The content of the target page is readable.

Exception

Text images and video captions are exempt from this requirement.

Criterion 10.5

In each web page, are the CSS declarations for element background and font colors used correctly?

WCAG 2.1 reference: 1.4.3 Contrast (Minimum) (AA).

Procedure

Examine the web page code and CSS statements:

- Check whether a specified "foreground" color is present;
- Check whether a specified "background" color is present (background, background-color);
- Check whether an image is used to create a background color for an element that may contain text, via CSS (background, background-image), is accompanied by a background color declaration (background, background-color).

Result

The background color is declared in the CSS.

Criterion 10.6

On every web page, is every link that is not obvious in nature visible in relation to the surrounding text?

WCAG 2.1 reference: 1.4.1 Use of Color (A).

Procedure

In every web page, every text link marked only by color, and whose nature is not obvious, checks these conditions:

- The color of the link has a contrast ratio greater than or equal to 3:1 with respect to the surrounding text;
- The link has a visual indication on hovering other than a color change;
- The link has a visual indication of the focus other than a color change.

Result

Each link on the page that is identifiable by color is visually identifiable by another means (e.g. underlined, bold, italic, with sufficient difference in brightness, etc.).

Criterion 10.7

In each web page, for each element receiving focus, is the focus capture visible?

WCAG 2.1 reference: 1.4.1 Use of Color (A), 2.4.7 Focus Visible (AA).

Procedure

For each element receiving focus, the focus capture checks one of these conditions:

- The browser's native focus style is not removed or degraded;
- A focus style defined by the author is visible.

Place the focus on each user interface element on the page with the mouse:

- Check that there is a clearly visible focus indicator;
- Emphasize each user interface element on the page using the keyboard;
- Check that the area of the focus indicator is at least the size of a 2 px CSS line along the longest edge of the element;
- Check that the change in indicator contrast between in-focus and out-of-focus states has a ratio of 3:1 for the minimum area of the focus indicator;
- If the focus indicator does not have a 3:1 contrast ratio with its adjacent colors, check that it is at least 2px thick.

For each focusable element:

- Hover over the item with a mouse;
- Check that the background or border changes color;
- Move the mouse away from the object before attempting to focus the keyboard;
- Using a keyboard, tab over the item;
- Check that the background or border changes color;
- Check that the background or border changes color when the item goes out of focus.

Result

The focus remains visible and logical.

Criterion 10.8

For each web page, is hidden content intended to be ignored by assistive technologies?

WCAG 2.1 reference: 1.3.2 Meaningful Sequence (A), 4.1.2 Name, Role, Value (A).

Procedure

Linearize the content using the Web developer extension and check whether the sequential order of the content gives the same meaning as the original.

Result

Hidden content is ignored by assistive technologies.

Criterion 10.9

In every web page, information should not be given only by shape, size or position. Is this rule respected?

WCAG 2.1 reference: 1.3.3 Sensory Characteristics (A), 1.4.1 Use of Color (A).

Procedure

Find all references in the web page that mention the shape, size or position of an object. For each of these objects:

- Check that the reference contains additional information that allows the object to be located and identified without knowing its shape, size or relative position;
- For each instruction that refers to non-textual markers that convey information:
    - Ensure that there are other ways of determining the information conveyed by non-text marks.

The same applies to sensory indications such as shape, position in space or sound, all of which must be accompanied by an alternative in the form of text.

Result

The information conveyed by sensory data can also be obtained by explicit text.

Criterion 10.10

In every web page, information should not be given by shape, size or position alone. Is this rule implemented in a relevant way?

WCAG 2.1 reference: 1.3.3 Sensory Characteristics (A), 1.4.1 Use of Color (A).

Procedure

Find all references in the web page that mention the shape, size or position of an object. For each of these objects:

- Check that the reference contains additional information that allows the object to be located and identified without knowing its shape, size or relative position;
- Examine the web page to check for textual references to the content of the web page;
- Check that references are not based solely on visual form or location of content.

Result

Presence of an alternative for each sensory reference. Text, or an explicit message, is present to convey the information.

Criterion 10.11

For each web page, can the contents be presented without using both vertical scrolling for a window with a height of 256px or a width of 320px (excluding special cases)?

WCAG 2.1 reference: 1.4.10 Reflow (AA).

Procedure

Use the Web developer extension to adjust the browser display settings to set the screen resolution to 1280 x 1024.

Use the browser settings to set the target page zoom to 400%.

Examine the target page to ensure that all text content is available without horizontal scrolling. The content can be displayed directly on the page.

Result

No horizontal scroll.

Criterion 10.12

In each web page, can the text spacing properties be redefined by the user without loss of content or functionality (except in special cases)?

WCAG 2.1 reference: 1.4.12 Text Spacing (AA).

Procedure

This test uses the Text spacing bookmarklet to adjust the spacing of the text on the target page.

Open Text Spacing in a new browser window, add the "Bookmarklet: Text Spacing" of this page to your browser bookmarks.

- Run the bookmarklet in the browser tab containing your target page. The text style will be automatically adjusted as follows;
- Letter spacing (tracking) at 0.12 times the font size;
- Word spacing at 0.16 times the font size;
- Line height (line spacing) at 1.5 times the font size;
- Spacing after paragraphs at two times the font size.

Check that all the following points are true:

- The whole text responds to each change of style;
- All text remains visible (no cuts);
- There is no overlapping text;
- For each word that appears to have non-standard spacing between characters:
  - Check whether the CSS property "letter-spacing" has been used to control the spacing.
  - Check that the spacing between lines in the text blocks is between 1.5 and 2.

Result

The spacing has been adjusted according to the criteria. The texts are not truncated or overlapped.

Criterion 10.13

In each web page, is the additional content that appears when the focus is on or when a component is hovered over controllable by the user (except in special cases)?

WCAG 2.1 reference: 1.4.13 Content on Hover or Focus (AA).

Procedure

For each additional content area, that appears when the pointer is hovered over:

- The pointer can be moved to the new content without the content in question disappearing;
- The user agent controls the appearance of additional content.

Result

Access to additional content is user-controllable.

Criterion 10.14

In each web page, can additional content appearing via CSS styles only be made visible by keyboard and any pointing device?

WCAG 2.1 reference: 2.1.1 Keyboard (A).

Procedure

This test uses the Force focus bookmarklet to track the keyboard pointer in the target page:

- Run the bookmarklet in the browser tab containing the target page;
- Identify all features on the content;
- Check that all features are accessible only through the keyboard or keyboard interface.

Result

All functions are accessible from the keyboard.

7.11 Forms

For each form, associate each of its fields with its heading, group the fields in blocks of information of the same nature, structure the choice lists in a relevant manner, and give each button an explicit heading. Check the presence of input help; ensure that the input control is accessible and that the user can control financial, legal or personal data.

Criterion 11.1

Does each form field have a label?

WCAG 2.1 reference: 1.3.1 Info and Relationships (A), 2.4.6 Headings and Labels (AA), 3.3.2 Labels or Instructions (AA), 4.1.2 Name, Role, Value (A).

Procedure

For each interface component with a:

- Identify the function of the interface component;
- Check that each label clearly indicates the purpose of the component.

For all <input> elements of "type = text", file or password, for all <textarea> and for all <select> elements on the web page:

- Check that there is a <label> element that identifies the purpose of the control before the <input>, <textarea> or <select> element;
- Check that the "for" attribute of the <label> element matches the "id" attribute of the <input>, <textarea> or <select> element;
- Check that the <label> element is visible.

For all <input> elements of "type= checkbox" or "radio" in the web page:

- Check that there is a <label> element that identifies the purpose of the control after the <input> element;
- Check that the "for" attribute of the <label> element matches the "id" of the input element;
- Check that the <label> element is visible;
- For all forms that are not associated with a <label> element:
    - Check that the control has a "title" attribute.
    - Check that the purpose of the form control is clear to users who can see the control.
    - Check that the 'title' attribute identifies the control object and that it matches the apparent visual object.

For all <input>, <textarea>, and <select> elements on the web page (except <input> of "type= hidden", "submit", "reset", or "button", check that each element has a specified name using one of the following methods:

- The <label> text is associated with the control element via the "aria-labelledby" attribute (each identifier given as a value in the "aria-labelledby" attribute corresponds to "id" of the <label> text element);
- The control is determined by the value of its attribute "aria-label";
- The <label> text is contained in a <label> element which is correctly associated with the respective <input> element via the "for" attribute of the <label> (the "id" value given in the "for" attribute corresponds to the "id" of the "input" element);
- The control is contained in a <label> element which also contains the label text;
- The control is an <input> of "type= image" and the "alt" attribute provides text for the label.

- The control is determined by the value of the "title" attribute.

Result

Presence of a <label> tag or equivalent for any form field.

Criterion 11.2

Is each label associated with a form field relevant (excluding special cases)?

WCAG 2.1 reference: 2.4.6 Headings and Labels (AA), 2.5.3 Label in Name (A), 3.3.2 Labels or Instructions (AA).

Procedure

For each interface component with a:

- The text in the <label> tag indicates the function of the field;
- The "title" attribute is used to identify the exact function of the form field to which it is associated;
- The label implemented via the WAI-ARIA attribute "aria-label" allows to know the exact function;
- Each associated text passage via the WAI-ARIA attribute "aria-labelledby" provides the exact function of the form field.

Each form field with a visible heading checks these conditions:

- If present, the content of the WAI-ARIA attribute "aria-label" of the form field contains at least the visible label;
- If present, the text passage linked to the form field via a WAI-ARIA attribute "aria-labelledby" contains at least the visible heading;
- If present, the content of the "title" attribute of the form field contains at least the visible heading;
- If present the content of the <label> tag associated with the form field contains at least the visible heading.

Each button adjacent to the form field that provides a visible label indicates the exact function of the form field with which it is associated.

For each element where an "aria-label" attribute is present: consider whether the textual description correctly labels the object or provides a description of its purpose or equivalent information.

Result

Each label clearly indicates the purpose of the component and the form field.

Criterion 11.3

In each form, is each label associated with a form field having the same function and repeated several times in the same page or in a set of pages consistent?

WCAG 2.1 reference: 3.2.4 Consistent Identification (AA).

Procedure

Examine all web pages that have the same repeated forms:

Check the consistency of each label associated with a form field with other labels of the same function.

Result

The labels are identical for the same forms on different pages.

Criterion 11.4

In each form, are each field label and its associated field joined together (except in special cases)?

WCAG 2.1 reference: 3.3.2 Labels or Instructions (AA).

Procedure

For each form field on the web page:

- Check that the form field has a visible label;
- If the form field is a "check box" or "radio button", check that the label is immediately after the field;
- If the form field is not a "checkbox" or "radio button", check that the "label" is immediately before the field.

Result

The labels are in close proximity to each corresponding form field.

Criterion 11.5

In each form, are fields of the same nature grouped together, if necessary?

WCAG 2.1 reference: 1.3.1 Info and Relationships (A), 3.3.2 Labels or Instructions (AA).

Procedure

For groups of related controls, where the individual labels for each control do not provide a sufficient description and an additional description at the group level is required:

- Check that the logically related <input> or <select> element group is contained within the <fieldset> elements;
- Check that each <fieldset> has an element of <legend> that includes a description for that group.

For groups of related controls, where the individual labels for each control do not provide a sufficient description and an additional description at the group level is required:

- Check that the group of logically related <input> or <select> elements is contained in an element with "role=group";
- Check that this group has an accessible name defined using "aria-label" or "aria-labelledby";
- Presence of a role='group' attribute (or radiogroup for radio buttons) on the parent element encapsulating the fields.

Result

Fields of the same nature are grouped together.

Criterion 11.6

In each form, does each grouping of form fields have a legend?

WCAG 2.1 reference: 1.3.1 Info and Relationships (A), 3.3.2 Labels or Instructions (AA).

Procedure

Each grouping of fields of the same nature has a legend <legend> tag.

Result

Presence of a <legend> tag for grouped form fields.

Criterion 11.7

In each form, is each legend associated with a grouping of fields of the same nature relevant?

WCAG 2.1 reference: 3.3.2 Labels or Instructions (AA).

Procedure

Examine the content of the <legend> tag and check that it conveys reliable information for the user.

Result

The caption is correctly associated with the function of the form.

Criterion 11.8

In each form, are items of the same nature in a choice list grouped in a relevant way?

WCAG 2.1 reference: 1.3.1 Info and Relationships (A).

Procedure

Check the set of options in a pick list to see if there are any related groups of options.

- If there are related option groups, they should be grouped with "optgroup", for the <select> tag, items of the same nature in a choice list are grouped with an <optgroup> tag;
- Each <optgroup> tag has a label attribute;
- As each <optgroup> tag has a label attribute, the content of the label attribute is relevant.

Result

A relevant <optgroup> tag groups selection lists.

Criterion 11.9

In each form, is the title of each button relevant (excluding special cases)?

WCAG 2.1 reference: 2.5.3 Label in Name (A), 4.1.2 Name, Role, Value (A).

Procedure

For all <button> elements, the content of the <button> tag contains at least the visible heading.

For all <input> elements that have a "type= image" attribute value:

- Check for the presence of an "alt" attribute;
- Check that the "alt" attribute indicates the function of the button.

For <input> that use "aria-labelledby":

- Check that the id's referenced in "aria-labelledby" are unique and correspond to the id's of the text nodes that together provide the label;
- Check that the concatenated content of elements referenced by 'aria-labelledby' is descriptive of the purpose or function of the labelled element.

Result

The title of each button is relevant and indicates its function.

Criterion 11.10

In each form, is the input control used in a relevant way (except in special cases)?

WCAG 2.1 reference: 3.3.1 Error Identification (A), 3.3.2 Labels or Instructions (AA).

Procedure

Indications that fields are mandatory check one of these conditions:

- A mandatory field indication is visible and allows the field concerned to be identified by name before the form is validated, e.g.: * (star);
- The required field has the attribute aria-required="true" or required before the form is validated.

Error messages indicating the absence of a mandatory field check one of these conditions:

- The error message indicating that a mandatory field has not been entered is visible and allows the field to be identified by name;
- The mandatory field has the attribute "aria-invalid=true".

The mandatory data type and/or format instructions and indications check one of these conditions:

- An instruction or indication of the type of data and/or mandatory format is visible and allows the field concerned to be identified by name before the form is validated;
- An instruction or indication of the type of data and/or mandatory format is visible in the label or text passage associated with the field prior to form validation.

Error messages providing an instruction or indication of the type of data and/or mandatory format of the fields:

- Checks that the error message provides an instruction or indication of the mandatory data type and/or format is visible and identifies the field concerned;
- Check that the field has the attribute "aria-invalid=true".

Fields with the attribute "aria-invalid=true" whose input requires a mandatory data type and/or format, check that a statement or indication of the mandatory data type and/or format is visible and located in the <label> tag associated with the field.

Result

Presence of text specifies the type of input and format expected for each form and the mandatory fields are visually identifiable.

Criterion 11.11

In each form, is the input check accompanied, if necessary, by suggestions to help correct input errors?

WCAG 2.1 reference: 3.3.3 Error Suggestion (AA).

Procedure

Identify fields in the form where correct text could be deduced from incorrect text:

- Fill in the form, deliberately filling in the identified form fields with incorrect text;
- Check that the user receives suggestions for the incorrect text;
- Check that suggestions are provided next to the form field or that a link to the suggestions is provided near the form field.

Check that one of the following solutions is implemented:

1. The error message is present before the form. The message lists the fields in error. The focus is placed on the error message. Anchors allow the user to position themselves on the fields in error;
2. Error messages are added to the field labels;
3. Error messages are not embedded in the labels, but a mechanism (script) is present to link fields and errors, via "aria-describedby" for example.

Result

Errors are detected and messages are relevant.

Criterion 11.12

For each form that changes or deletes data, or transmits answers to a test or exam, or whose validation has financial or legal consequences, does the data entry check one of these conditions?

WCAG 2.1 reference: 3.3.4 Error Prevention (Legal, Financial, Data) (AA).

Procedure

In a test application or an application, that causes financial or legal transactions and collects user data in several steps:

- Check that the user is prompted to review and confirm the data;
- If the user's data is collected in several stages, the user is allowed to return to the previous stages to review and modify the data;
- Determine whether a summary of all data entered by the user is provided before the transaction is initiated and whether a method is provided to correct errors if necessary.

For user input pages that cause irreversible transactions:

- Check that a checkbox indicating confirmation of the input or action by the user is provided in addition to the "Send" button.
- Check if the box is not ticked when the form is submitted, the entry is rejected and the user is prompted to review their entry, tick the box and resubmit.

Result

An accessible mechanism is present to identify errors and warn the user.

Criterion 11.13

Can the purpose of an input field be inferred to facilitate the automatic filling of fields with user data?

WCAG 2.1 reference: 1.3.5 Identify Input Purpose (AA).

Procedure

For each field on the form that collects user information and is an autocomplete field, check the following:

- The form field has an 'autocomplete' attribute and a pair of valid and well-formed values;

- The form field object indicated by the <label> corresponds to the autocomplete code of the <input> tag.

Result

Presence of the "autocomplete" attribute for each form field if required.

7.12 Navigation

Facilitate navigation through a set of pages by at least two different navigation systems (navigation menu, sitemap or search engine), a breadcrumb trail and indication of the active page in the navigation menu. Identify important link groups and the content area and give the possibility to avoid them by internal navigation links. Ensure that the tabbing order is consistent and that the page does not contain any keyboard traps.

Criterion 12.1

Does each set of pages have at least two different navigation systems (excluding special cases)?

WCAG 2.1 reference: 2.4.5 Multiple Ways (AA).

Procedure

Each set of pages checks one of these conditions:

- A navigation menu and a site map are present;
- A navigation menu and a search engine are present;
- A search engine and a site map are present;
- A list of links to navigate through the site;
- A table of contents;
- Provide a breadcrumb trail.

Result

A navigation mechanism is present on every page of the site.

Criterion 12.2

In each set of pages, are the menu and navigation bars always in the same place (except in special cases)?

WCAG 2.1 reference: 3.2.3 Consistent Navigation (AA).

Procedure

List the elements that are repeated on each web page in a set of web pages (for example, on each page of a website).

- For each component, check that it appears in the same relative order to the other components repeated on each web page where it appears;
- For each navigation element, check that the links or references are always in the same relative order;
- Check whether a navigation mechanism is used on more than one web page, and check the default layout of the navigation mechanism on each page to see if the list of links is in the same relative order on each web page.

Result

The navigation mechanism is identical for all the pages of the site.

Criterion 12.3

Is the site map page relevant?

WCAG 2.1 reference: 2.4.5 Multiple Ways (AA).

Procedure

In each target page:

- Check for the presence of the site map;
- The site map page is representative of the general architecture of the site;
- Check that the links in the sitemap lead to the relevant sections of the site and are functional;
- For each link in the sitemap, check that the target page contains a link to the sitemap;
- Check that the page can be reached by following a set of links that start at the sitemap.

Result

A relevant and identical sitemap for each page of the site.

Criterion 12.4

In each set of pages, is the "site map" page reachable in the same way?

WCAG 2.1 reference: 2.4.5 Multiple Ways (AA), 3.2.3 Consistent Navigation (AA).

Procedure

In each set of pages:

- The "site map" page can be accessed using the same functionality;
- The functionality to the site map page is located in the same place in the layout;
- The functionality to the sitemap page is always in the same relative order in the source code.

Result

The "site map" page is identical and functional in a set of pages.

Criterion 12.5

In each set of pages, is the search engine reachable in the same way?

WCAG 2.1 reference: 3.2.3 Consistent Navigation (AA).

Procedure

In each set of pages:

- The search engine is accessible from an identical functionality;
- The functionality to the search engine is located at the same place in the presentation;
- The functionality to the search engine is always in the same relative order in the source code.

Result

The search engine is present and accessible.

Criterion 12.6

Can content grouping areas on several web pages (header, main navigation, main content, footer and search engine areas) be reached or avoided?

WCAG 2.1 reference: 1.3.1 Info and Relationships (A), 2.4.1 Bypass Blocks (A), 4.1.2 Name, Role, Value (A).

Procedure

In each set of pages and using the bookmarklet Landmarks, inspect the parts of the content, which have a semantic function:

1. For each part that has a semantic function, if the technology has a corresponding semantic tag, check that the content has been tagged using that semantic tag.
2. For a user interface component using the "role" attribute:
   - Check that the value of the "role" attribute is one of the non-abstract roles among the values defined in the WAI-ARIA specification;
   - Check that the characteristics of the user interface component are described by the role;
   - Examine each element with a landmark role:
     - Consider whether the landmark role attribute is applied to the section of the page that corresponds to that role. (i.e. the 'navigation' role is applied to a navigation section; the 'main' role is applied to where the main content begins).

Result

Key navigation areas are mentioned and correctly implemented. The different regions are visually clear and identifiable.

Criterion 12.7

In each web page, is there an avoidance or quick access link to the main content area (excluding special cases)?

WCAG 2.1 reference: 2.4.1 Bypass Blocks (A), 2.4.3 Focus Order (A), 3.2.3 Consistent Navigation (AA).

Procedure

Avoidance links are present as the first interactive element on the page. These links redirect to the main areas of the page (navigation, content, footer and top of page).
In each target page, a link is the first focusable element:

- Check that the description of the link indicates that it refers to the main content;
- Check that the link is either always visible or visible when highlighted by the keyboard;
- Check that activating the link shifts the focus to the main content;
- Check that after the link has been activated, the keyboard focus has moved to the main content.

Check that a link is the last focusable element before the repeated content block or the first link in the block:

- Check that the description of the link indicates that it skips the block;
- Check that the link is either always visible or visible when highlighted by the keyboard;
- Check that after the link is activated, the keyboard focus has moved to the content immediately after the block.

For each link in the set of avoidance links:

- Check that the description of each link indicates that it refers to a section of the content;
- Check that the link is either always visible or visible when highlighted by the keyboard;
- Check that activating the link shifts the focus to that section of the content.

Result

Presence of skip links to access the main parts of the page.

Criterion 12.8

Is the order of tabs on each web page consistent?

WCAG 2.1 reference: 2.4.3 Focus Order (A).

Procedure

In each target page:

- Determine the order of the interactive elements in each content;
- Determine the logical order of the interactive elements;
- Check that the order of the interactive elements in the content is the same as the logical order.

Check if the "tabindex" attribute is used:

- If the "tabindex" attribute is used, check that the tabbing order specified by the "tabindex" attributes follows the relationships in the content;
- The values of the attribute "tabindex" :
  - A negative value: the item cannot be reached via keyboard navigation
  - Zero: the item can be reached via keyboard navigation;

   - A positive value: the element can be reached via keyboard navigation and the elements will be browsed in ascending order of the attribute's value.

For each menu or dialog embedded in a web, page that opens via a trigger command:

- Activate the trigger control via the keyboard;
- Check whether the focus is on the menu or the dialogue;
- Check whether moving the focus forward in the sequential navigation order puts the focus in the menu or dialogue.

Exit the menu or dialog box:

- Check if the focus is on the trigger control;
- Check whether moving the focus backwards in the sequential navigation order puts the focus on the trigger control.

Result

The tabbing order is sequential and logical.

Criterion 12.9

In every web page, the navigation should not contain any keyboard traps. Is this rule respected?

WCAG 2.1 reference: 2.1.1 Keyboard (A), 2.1.2 No Keyboard Trap (A).

Procedure

Use the standard keyboard commands (tab keys and unmodified arrows) to navigate all elements of the interactive interface of the target page.

If you cannot exit a component using the standard keyboard commands:

- Examine the accessible name and accessible description of the component to determine if they describe an alternative keyboard command;
- If an alternative keyboard command is provided, check if it work;.
- Check that you can exit all components using either:
  - Standard keyboard commands;
  - An alternative keyboard command that is described to users.

Browse the content from start to finish:

- Check that the keyboard focus is not blocked in the content;

- If the keyboard focus seems to be trapped in the content, check that help information is available explaining how to leave the content and is accessible via the keyboard.

Moving the focus on the screen allows for logical and consistent navigation (from most important to least important, from top to bottom of the screen).

The reading order allows for a good understanding of the content (from the most important to the least important, from top to bottom of the screen).

Check that the order of focus of elements (buttons, links, form fields) is consistent and logical (often equivalent to the visual order).

Result

The keyboard path through the page is logical and there are no traps when navigating the keyboard.

Criterion 12.10

In each web page, are keyboard shortcuts using only one key (lower or upper case letter, punctuation, number or symbol) controllable by the user?

WCAG 2.1 reference: 2.1.4 Character Key Shortcuts (A).

Procedure

Examine the target page to identify keyboard shortcuts that can be activated using only printable keys, such as letters, numbers, symbols or punctuation. (Alt, Shift and Ctrl are non-printable keys).

Check that at least one of the following is true:

- A mechanism is available to disable the shortcut;
- A mechanism is available to configure the hotkey using the modifier keys (Ctrl, Alt, Shift, etc.)
- In the case of a user interface component, the keyboard shortcut associated with it can only be activated if the keyboard focus is on that component.

Result

The keyboard shortcuts are user controllable.

Criterion 12.11

In each web page, is additional content that appears when hovering, focusing or activating an interface component, if necessary, reachable by keyboard?

WCAG 2.1 reference: 2.1.1 Keyboard (A).

Procedure

Use the keyboard to navigate through all the elements of the interactive interface of the target page, and activate the Force Focus bookmarklet:

- Use the Tab and Shift + Tab keys to navigate back and forth between widgets;
- Use the arrow keys to navigate between the items that receive focus.

If a "keyboard trap" prevents the focus from leaving a widget:

- Use your mouse to move the focus to the next widget;
- Repeat the tests.

If you encounter a trigger that reveals hidden content:

- Activate the trigger;
- Browse through the revealed content;
- Close the revealed content;
- Resume navigation on the page.

Check that you can navigate to all interactive components using the keyboard:

- Check that the user can move the focus from the trigger to the additional content without it closing;
- Check that there is a mechanism to remove the additional content without having to move the focus;
- Check that the content that appears remains visible as long as the user does not interact with the trigger and that the content remains valid.

Any opening of windows or pop-up boxes creates a change of context that can be confusing to the user. The use of dialogue boxes should therefore be reserved for the presentation of information or actions that require users' immediate attention.

Result

All interactive elements are accessible from a keyboard navigation and usable from keyboard interactions.

7.13 Consultation

Check that the user has control over refreshment processes, sudden changes in brightness, opening of new windows and moving or flashing content. Indicate when content opens in a new window and provide information about viewing downloaded files. Do not make the completion of a task dependent on a time limit unless it is essential and ensure that input data is recovered after an authenticated session break. Ensure that unusual expressions and jargon are explained. Provide accessible versions or make documents available for download.

Criterion 13.1

For each web page, does the user have control over each time limit modifying the content (except in special cases)?

WCAG 2.1 reference: 2.2.1 Timing adjustable (A), 2.2.2 Pause, Stop, Hide (A).

Procedure

Examine the target page to determine if it has time limits on content.

Ignore any time limit, which is part of a real-time event (such as an offer), and there is no alternative to the time limit; or essential to the activity (such as an online test), and allowing users to extend it would invalidate the activity; or over 20 hours.

If the page has a time limit, check that there is a way to:

- Disable the time limit;
- Adjust the time limit to at least 10 times the default limit;
- Be warned of the time limit, given at least 20 seconds to extend the time limit by a simple action (e.g. Press the space bar"), and given the option to extend the time limit at least 10 times.

Measure or estimate the time the average user spends on the page:

- Go to page;
- Wait for 10 times the time the average user stays on the page;
- To detect if there is a change of context during this period;
- If there is no change of context, STOP.

If there is a context switch, then check if there is a mechanism on the page that would have prevented the context switch.

If there is a mechanism to prevent this context switch, use that mechanism to prevent the context switch and retest.

If there is a change of context and there is no mechanism to prevent that change of context, then you have a failure.

Find pages whose content updates automatically:

- For each automatic update, look for a mechanism to adjust the update schedule;
- Check that automatic updating is disabled by default or that the user is warned before an automatic update occurs and is allowed to remove it;
- Find each link or reference to another page or web page:
    - For each link or reference, check whether the referenced web page contains any code (e.g. a meta-element or script) that causes a client-side redirect;
    - For each link or reference that causes a client-side redirect, check that, the redirect is implemented without time limits or delays and that the page contains only information related to the redirect.

Find all the <meta> elements in the document:

- For each <meta> element, check whether it contains the "http-equiv" attribute with the value "refresh" (case insensitive) and the "content" attribute with a number greater than 0 followed by; "URL=anyURL" (where anyURL represents the URI that should replace the current page).

If the web page contains the first part of a multi-part form:

- Check that there is a checkbox on the web page to request additional time to complete the form;
- Check that if the box is ticked, there is extra time to complete the form.

Detect the mechanisms for setting the delay to 10 times the default delay:

- Change the delay to a new value of 10 times the default delay;
- Perform an action that has a time limit;
- Wait until the default time has elapsed;
- Check that the time limit does not expire before the limit specified in step 2 has passed.

For each web page, each refresh process (<object> tag, <embed> tag, <svg> tag, <canvas> tag, <meta> tag) checks one of these conditions:

- The user can stop or restart the refresh;
- The user can increase the time limit between two refreshes by at least ten times;
- The user is warned of the impending refresh and has at least twenty seconds to increase the time limit before the next refresh;
- The time limit between refreshments is twenty hours, at least

For each web page, each redirection process performed via a <meta> tag is immediate.

For each web page, each redirection process performed via a script checks one of these conditions:

- The user can stop or restart the redirection;
- The user can increase the time limit before redirection by at least ten times;
- The user is warned of the impending redirection and has at least twenty seconds to increase the time limit before the next redirection;
- The time limit before redirection is at least twenty hours.

For each web page, each time-limiting process checks one of these conditions:

- The user can remove the time limit;
- The user can increase the time limit;
- The time limit before the end of the session is at least twenty hours.

Result

The user can deactivate, or increase the duration of the consultation time of a content.

Criterion 13.2

In every web page, the opening of a new window must not be triggered without action from the user. Is this rule respected?

WCAG 2.1 reference: 3.2.1 on focus (A).

Procedure

Load the web page:

- Check if new (additional) windows open.
- Find all actionable elements on the web page, such as links and buttons.

- Activate each item.
- Check whether activating the item opens a new window.
- Check whether elements that open new windows have associated text indicating that this will happen. The text may be displayed in the link, or available through a hidden association such as an HTML title attribute.

In order not to surprise the user, the opening of a new window must be announced beforehand. The user can then choose not to trigger the event. In a specification document or on the mock-ups, the wording of the triggers is described in the form: [link target] + (new window), for example: 'Hi.com (new window)'.

Result

The user, explicitly signaled, triggers the opening of a new window.

Criterion 13.3

In each web page, does each office document available for download have, if necessary, an accessible version (excluding special cases)?

WCAG Reference 2.1: 1.1.1 Non-text Content (A), 1.3.1 Info and Relationships (A), 1.3.2 Meaningful Sequence (AA), 2.4.1 Bypass Blocks (AA), 2.4.3 Focus Order (A), 3.1.1 Language of Page (A), 4.1.2 Name, Role, Value (A)

Procedure

In each web page, each office document download feature checks one of these conditions:

- The downloadable document is accessibility friendly;
- There is an alternative version of the document available for download that is accessibility friendly;
- An alternative version of the document is available for download in HTML format.

Result

Office content is accessible and compatible with assistive technologies.

Criterion 13.4

For each office document that has an accessible version, does this version offer the same information?

WCAG Reference 2.1: 1.1.1 Non-text Content (A), 1.3.1 Info and Relationships (A), 1.3.2 Meaningful Sequence (AA), 2.4.1 Bypass Blocks (AA), 2.4.3 Focus Order (A), 3.1.1 Language of Page (A), 4.1.2 Name, Role, Value (A)

Procedure

Every office document with an accessible version meets one of these conditions:

- The accessibility-friendly version offers the same information;
- The alternative version in HTML format is relevant and offers the same information.

Result

The compatible version transmits the same information.

Criterion 13.5

In every web page, does every cryptic content (ASCII art, emoticon, cryptic syntax) have an alternative?

WCAG 2.1 reference: 1.1.1 Non-text Content (A).

Procedure

Open the page in a common browser:

- Check that the content contains ASCII art , emoticons and/or leetspeak;
- Check that there is an alternative text immediately before or after any ASCII art, emoticons and/or Leetspeak;
- A "title" attribute is available for ASCII art content;
- A definition is given by the adjacent context.

Result

Content available as ASCII art has an accessible alternative.

Criterion 13.6

In each web page, for each cryptic content (ASCII art, emoticon, cryptic syntax) that has an alternative, is that alternative relevant?

WCAG 2.1 reference: 1.1.1 Non-text Content (A).

Procedure

In each web page, each cryptic content (ASCII art, emoticon, cryptic syntax) checks one of these conditions:

- The content of the "title" attribute is relevant;
- The definition given by the adjacent context is relevant.

Result

Content available as ASCII art has a relevant and accessible alternative.

Criterion 13.7

In each web page, are sudden changes in brightness or flash effects used correctly?

WCAG 2.1 reference: 2.3.1 Three Flashes or Below Threshold (A).

Procedure

In every web page, every image (<img> tag, <svg> tag, <canvas> tag, <embed> tag, or <object> tag) that causes an abrupt change in brightness or a flash effect checks one of these conditions:

- The frequency of the effect is less than 3 per second;
- The total cumulative area of the effects is less than or equal to 21824 pixels.

In every web page, every script that causes an abrupt change in brightness or flash effects checks one of these conditions:

- The frequency of the effect is less than 3 per second;
- The total cumulative area of the effects is less than or equal to 21824 pixels.

In every web page, every CSS layout that causes an abrupt change in brightness or flash effects checks one of these conditions:

- The frequency of the effect is less than 3 per second;
- The total cumulative area of the effects is less than or equal to 21824 pixels.

Result

No content that flashes for less than 3 seconds. No element on the page is flashing rapidly or generating flashes of light of more than 3 repetitions per second.

Criterion 13.8

In every web page, is every moving or flashing content controllable by the user?

WCAG 2.1 reference: 2.2.1 Timing Adjustable (A), 2.2.2 Pause, Stop, Hide (A).

Procedure

Review the target page to identify any information that:

- Moves, flashes or scrolls;
- Starts automatically;
- Lasts more than 5 seconds;
- Is presented in parallel with other content;
- Not part of an activity where it is essential.

If you find such content, check that you can pause, stop or hide it.

In each web page, each automatically triggered moving content checks one of these conditions:

- The duration of the movement is less than or equal to 5 seconds;
- The user can stop and restart the movement;
- The user can show and hide the content in motion;
- The user can view the entire information without the movement.

In each web page, each automatically triggered flashing content checks one of these conditions:

- The flashing time is less than or equal to 5 seconds;
- The user can stop and restart the flashing;
- The user can show and hide the flashing content;
- The user can display the full information without the flashing.

Display an animated gif and specify the duration of the animation:

- You can also use an image editor to determine the number of images, the frame rate and the number of repeats. Calculate the product of the number of frames multiplied by the frame rate and the number of repeats. If the frame rates are not uniform, calculate the product of the sum of the frame rates multiplied by the number of repeats.

Result

Whichever method is used, the duration of the animation must be less than or equal to 5 seconds.

Criterion 13.9

In each web page, is the proposed content viewable, regardless of the screen orientation (portrait or landscape) (except in special cases)?

WCAG 2.1 reference: 1.3.4 Orientation (AA).

Procedure

Open the target page on a device that automatically redirects web content when the orientation of the device changes (e.g. a mobile device):

- The consultation is possible, whatever the orientation of the screen, vertically and then horizontally;
- Check that the page content reorients itself when the orientation of the device changes;
- The content offered remains the same regardless of the screen orientation used, although its presentation and means of access may differ.

Result

The content of the site can be consulted regardless of the orientation of the screen.

Exception

Orientation locking is allowed if a specific orientation is essential to the underlying functionality (e.g. a banking application that requires horizontal orientation when photographing a check for deposit).

Criterion 13.10

In each web page, can the functionalities that are usable or available by means of a complex gesture also be available by means of a simple gesture (excluding special cases)?

WCAG 2.1 reference: 2.5.1 Pointer Gestures (A).

Procedure

Examine the target page to identify features that can be operated using any of the following:

- Multi-touch gestures, such as a two-finger zoom or three-finger tap;
- Path-based gestures, such as dragging or dropping;
- The function can also be used using a single, non-path based gesture, such as a double-click or a long press;
- Is every feature that is usable or available as a result of a multi-point touch also usable or available as a result of a single-point touch on the screen;
- Every feature that is usable or available because of a gesture based on tracking a path on the screen is also usable or available because of touching a single point on the screen.

Example: for a zoom action on a map at the pinch of a finger, an alternative in the form of <button> + or - is proposed.

Result

An alternative mechanism is present for each complex gesture.

Exception

Multi-point and path-based gestures are allowed when they are essential to the basic function.

Criterion 13.11

In each web page, can actions triggered by a pointing device on a single point on the screen be undone (except in special cases)?

WCAG 2.1 reference: 2.5.2 Pointer Cancellation (A).

Procedure

Examine the target page to identify all functions that can be operated using a single pointer.

Check that at least one of the following is true:

- No down-event: the pointer stop event is not used to execute any part of the function;

- Abort or cancel: the completion of the function is done on the "up" event, and a mechanism is available to cancel the function before its completion or to cancel the function after its completion;
- Upstream cancellation: the upstream event reverses any result of the previous downstream event.

Result

Presence of mechanisms that allow for interactions with different devices.

Exception

This requirement does not apply if the completion of the function in the downward phase is essential to the basic function. For example, for a keyboard emulator, entering a key at the event is considered essential.

Criterion 13.12

In each web page, can functionality that involves movement to or from the device be satisfied in an alternative way (excluding special cases)?

WCAG 2.1 reference: 2.5.4 Motion Actuation (A).

Procedure

Examine the target page to identify any functions that can be activated by the movement of the device (such as shaking) or the user (such as walking).

Check that the following are true:

- The function can also be used via user interface components (such as a toggle button);
- The functionality available by gesturing towards the device can be achieved with user interface components;
- The user can disable the motion operation.

Result

The user can pause/restart the movements and access the alternatives.

Exception

This requirement does not apply if motion activation is essential to the basic function, such as tracking the user's steps.

8 Conclusion

This document is only a start; it allows knowing the standards, the diagnostic tools and the guidelines for the implementation of an accessible web content. In addition, to raise awareness and keep the involvement of the teams by creating reflexes and good practices.

Accessibility is an essential approach for organizations and project actors who want to create high quality websites and web tools, with the aim of including many people.

9 Glossary

9.1 A

a11y

Acronym "accessibility" in English; the first letter "a", the number 11 the number of letters between "a" and "y" and the last letter "y".

Accessible and activatable via keyboard and any pointing device

An interface component (link, button ...) is accessible by the keyboard and by any pointing device when the user can take the focus by either a pointer or the tab key.

An interface component (link, button...) can be activated by the keyboard and by any pointing device when the user can initiate, indifferently, the action provided by the interface component by a pressure of the pointer or the keyboard enter key.

Please note: For some interface components such as sliders (sliding or rotating knobs...), it is not possible to control the component with the input key alone. In these situations, other keys (such as direction keys) can be used. In particular, for elements with a WAI-ARIA role corresponding to a design pattern it is recommended to consider the WAI-ARIA 1.1 Authoring Practices document when implementing them.

In the standard, the term "controllable by keyboard and any pointing device" also refers to this definition.

Important note: the use of some technologies may make focus management too complex or unstable to rely solely on tab, arrow and enter keys. In this case, providing keyboard shortcuts may be the only way to make the component usable.

The criterion can only be considered compliant if the keyboard shortcuts used are properly documented, functional and comply with criterion 12.10.

Joined (label and field joined)

The label and its field must be visually close so that the relationship between the two cannot be confused.

Alert

An alert message that interrupts navigation or use of the page, including a request to click a button or link to continue navigation or use of the content. For example, a dialogue box generated by JavaScript *via* the alert function. By extension, a modal window (content presented as a "window" inserted or displayed in the DOM) that needs to be closed to continue browsing or using the content is considered an alert.

Note: the deactivation of the alerts concerned can be proposed before the alert is triggered, for example, *via* a user parameter or when the first alert is displayed, for example *via* a "do not display this alert again" checkbox.

Alternative (to script)

A text or process associated with the script *via* an appropriate technique and allowing the provision of a function or content like that offered by the script.

Note: Where an alternative to a JavaScript process or functionality is offered, the means of accessing it must be provided by the site itself. This could be a link or button to an alternative page that does not require JavaScript or to replace the component(s) with an alternative component that does not require JavaScript, for example.

Alternative to an SVG image

The following are considered possible alternatives to an SVG image:

- A replacement mechanism
- An adjacent link or button that provides access to an alternative with relevant content, and identical to the WAI-ARIA attribute aria-label and the title attribute of the <svg> tag, if present.

Short and concise alternative

The conditions for rendering a textual alternative *via assistive* technologies (e.g. a screen magnifier) require that it be as short as possible. A maximum length of 80 characters is strongly recommended; this will limit the number of manipulations required to read the alternative by users of Braille displays or screen magnifiers in particular.

Textual alternative (image)

"Accessible name" rendered by assistive technologies for graphic elements such as:

- Image (<img> tag or tag with a WAI-ARIA role="img" attribute);
- Reactive image area (<area> tag);
- Image type button (<input> tag with type="image" attribute);
- Image object (tag <object type="image/…">);
- Vector image (tag <svg>);
- Bitmap image (<canvas> tag);
- Embedded image (<embed> tag).

In the case of a graphic element, the "accessible name" is obtained in the following order:

- Passing associated text via the WAI-ARIA attribute aria-labelledby for tags:
  - <img>;
  - <I'm not sure what to do about it;
  - <I'm not sure what to do;
  - <This is the only way to get a better understanding of the situation;
  - <I'm not sure what to do about it;
  - <I'm not sure what to do;
  - Tags with a WAI-ARIA role="img" attribute.
- Otherwise, content of the WAI-ARIA attribute aria-label for :
  - <img>;
  - <area>;
  - <I'm not sure what to do about it;
  - <I'm not sure what to do;
  - <This is the only way to get a better understanding of the situation;
  - <I'm not sure what to do about it;
  - <I'm not sure what to do;
  - tags with a WAI-ARIA role="img" attribute.
- Otherwise, content of the alt attribute for :
  - <img>;
  - <area>;
  - <I'm not sure what to do about it.
- Otherwise, content of the title attribute for :
  - <img>;
  - <I'm not sure what to do about it;

- <This is the only way to get a better understanding of the situation;
- <I'm not sure what to do about it.

This order should be used to determine what constitutes the textual alternative.

In the case of partial support for the "accessible name" algorithm, the value actually rendered by the assistive technologies used in the test environment (or "baseline") should be considered the textual alternative.

For example:

- ➔ In the case of the joint presence of a WAI-ARIA aria-label attribute and a WAI-ARIA aria-labelledby attribute on an <img> tag, it is the text passage referenced by the WAI-ARIA aria-labelledby attribute that shall be considered as the textual alternative if the content of the text passage is actually rendered by the assistive technologies used in the test environment.
- ➔ In the case of the joint presence of a WAI-ARIA aria-label attribute and an alt attribute on an <img> tag: the content of the WAI-ARIA aria-label attribute should be considered as the textual alternative if the content of the WAI-ARIA aria-label attribute is rendered by the assistive technologies used in the test environment.

The RGAA considers three types of textual alternatives related to the nature of the image:

- ➔ For an image that carries information, the textual alternative provides the information needed to understand the content it conveys;
- ➔ For a decorative image, no textual alternative should be rendered;
- ➔ For a CAPTCHA image or a test image, the textual alternative only describes the nature and function of the image. Indeed, the textual alternative cannot provide the information conveyed by the image without making the associated function inoperative.

Note 1: For a CAPTCHA image, the alternative may be, for example: "Anti-spam security code" or "code to verify that you are a human" or any other alternative that allows the user to understand the nature and function of the image.

Note 2: For a group of images, e.g. a voting system consisting of several star images, it is strongly advised to use either the first image in the group to give a consistent alternative to the group of images (see WCAG2.1 technique G196), or a container tag with a WAI-ARIA img role and a textual alternative. In the first case, the other images in the group are considered as decorative images. In the second case, all the images in the group are considered as decoration images.

Note 3: for link-images, the alternative must make it possible to understand the function and destination of the link; this case is dealt with in the links theme.

Note 4: for vector images <svg> tag, the text alternative could also be present in a <text> tag whether this tag is visible, even if this is not the role of this element in SVG.

Note 5: As the use of the alt attribute is the only technique fully supported by technical aids it is recommended that this solution be preferred when implementing an alternative to an <img>, <area> and <input type="image"> tag.

Ambiguous for everyone

Intent cannot be determined from the link and all the information on the web page presented to the user along with that link. (I.e. a reader without functional limitations would not know the function of a link before activating it). **Example**: The word "guava" in the following sentence used as a link: "One of the important exports is guava". This link could lead to a definition of guava, a graph listing the quantities of guava exported, or a picture of people harvesting guava. Until the link is activated, any user is in the dark, so a disabled person is not disadvantaged.

Anchor

In HTML, an anchor (also called a bookmark) consists of a <a> tag with the id attribute and no href, <a id="content"></a> for example.

An anchor serves as a target for a link of the form <a href="#id"> Title </a>:

<a href="#content"> Content </a> for example.

ASCII art

ASCII art consists of making images using only the letters and special characters contained in the ASCII code.

Synchronised audio description (time-based media)

Narration added (via a sound file) to an audio track to describe important visual details that cannot be understood from the main audio track alone. The audio description must be synchronized with the time-based media by an application device linked to the player itself or provided by the developer, for example with JavaScript.

Note 1: Audio description of a video provides information about actions, characters, scene changes, on-screen text and other visual content.

Note 2: In standard audio description, narration is added during pauses in the dialogue (see also extended audio description).

Note 3: When all the information in the video is already given in the audio track, no additional audio description is required.

9.2 B

Menu and navigation bar

A list of links allowing specific navigation within the site, within a section or within a collection of pages.

The main navigation bars (criterion 12.2) are:

- A navigation menu;
- A breadcrumb trail;
- A navigation list of a list of results ;
- Avoidance links.

There are different types of navigation menu (criteria 12.1 and 12.2):

- Main navigation menu ;
- Sub-heading menu;
- Context menu ;
- Table of contents for a set of pages.

Note: Footer links to the legal notice, site map and other information about the site are not considered a main navigation menu.

Button (form)

The element of a form that allows a predefined action to be performed. For example, the submit button on a form allows the information collected to be sent to the server for processing. The title of a button must describe the action that results from its activation (for example: "Start your search", "Send your message").

In HTML, there are three types of form buttons:

- <input> tag of type submit, reset or button;
- Image <input> tag;
- Tag <button>.

It is also possible to render a button using the WAI-ARIA button role.

The button label can be of six types:

- The content of the text passage associated with the button via the WAI-ARIA attribute aria-labelledby when present;
- The content of the aria-label attribute when present;
- The content of the alt attribute of an image button ;
- The content of the value attribute of submit, reset or button type buttons;
- The content of the <button> tag;
- The content of the title attribute when it is present.

Important note: when several of these techniques are present on the same button, the calculation of the "accessible name", i.e. what will be returned, follows a strict order:

- aria-labelledby ;
- Otherwise aria-label ;
- Otherwise alt for the case of image input ;
- Otherwise value for the case of input submit, reset or button ;
- Otherwise content of the <button> tag;
- Otherwise title.

This order should be used when assessing the relevance of the "accessible name" of the button.

For example, even in the case of the presence of a title and a text passage referenced by aria-labelledby on the same button: the text passage referenced by aria-labelledby should be evaluated.

Reference: Accessible name and description calculation.

Furthermore, an "accessible name" will be considered irrelevant if it does not include the visible text of the button. For example:

<button aria-label="confirm entry"> Validate entry </button> will be considered as not complying with criterion 11.2.

9.3 C

Frame

Frame: HTML element (<frame> tag) used to display content in the web page in which it is implemented.

Inline frame: an HTML element (<iframe> tag) used to display content on the web page in which it is implemented.

In-line frame

HTML elements (iframe tag) that allow content to be displayed on the web page in which it is implemented.

CAPTCHA

A CAPTCHA "Completely Automated Public Turing Test To Tell Computers and Humans Apart" is a test used to distinguish a human user from a computer. The test often uses images containing distorted text, mixed with other shapes or using altered color schemes, which the user is asked to retype. Other forms of CAPTCHA may be based on logical questions or sound bites.

Fields of the same nature

In a form, a set of fields that can be grouped by the nature of the information expected. The purpose of grouping is to identify the fields that should be treated as a set.

Some examples:

- Three successive fields for entering a date (day/month/year);
- Successive fields for a telephone number;
- A block for entering the user's identity and address, when the form contains several contact blocks;

- A set of radio buttons or checkboxes that relate to a question.

These fields should be grouped when the label headings are not sufficient to inform the user that the fields are part of a grouping.

Form input field

The object of a form allowing the user to perform several actions:

- Enter textual or pre-formatted data:
  - input type="text" ;
  - input type="password" ;
  - input type="search" ;
  - input type="email" ;
  - input type="number" ;
  - input type="tel" ;
  - input type="url" ;
  - textarea ;
- Select predefined values:
  - input type="checkbox" ;
  - input type="radio" ;
  - input type="date" ;
  - input type="range" ;
  - input type="color" ;
  - input type="time" ;
  - input type="month" ;
  - input type="week" ;
  - input type="datetime-local" ;
  - select ;
  - datalist ;
  - optgroup ;
  - option ;
- To download files :
  - input type="file" ;
  - Or to display results:
  - output ;

        - progress ;
        - meter.
    - Tags with a WAI-ARIA role to render a form field are also covered by this definition:
        - progressbar ;
        - slider ;
        - spinbutton ;
        - textbox ;
        - listbox ;
        - searchbox ;
        - combobox ;
        - option ;
        - checkbox ;
        - radio ;
        - switch ;
    - The following WAI-ARIA form objects and roles are not considered form fields:
        - input type="submit" ;
        - input type="reset" ;
        - input type="hidden" ;
        - input type="image" ;
        - input type="button" ;
        - button ;
        - WAI-ARIA attribute role="button".

Sudden change in brightness or flash effect

Alternating relative brightness that can cause seizures in some people if they are large, enough in a specific frequency range.

Change of context

Major changes in the content of a web page that, if made without the user's awareness, can disorient the user from seeing the entire page at once. Contextual changes include:

1. User agent ;
2. A space for restitution ;

3. From focus ;
4. Content that changes the meaning of the web page.

Note: A change in content is not always a change in context. A change in content such as a tree spread, a dynamic menu or a tab move does not necessarily change the context unless it also changes one of the elements listed above (e.g. the focus).

For example, opening a new window, shifting the focus to a different component, moving to a new page (including anything that to the user would look like moving to another page), or significantly reorganizing the content of a page are all illustrations of a context switch.

Change of language

The indication of language changes is necessary to tell assistive technologies to modify the speech output of an element. Language changes affect all content, including the values of certain attributes such as title.

Note: it is not possible to indicate language changes in an attribute value itself, in which case the language change is indicated on the element that contains the attribute. For example, a link with an English title should have a Lang="en" attribute. When the attribute contains several passages of text in different languages, the criterion is not applicable.

Language code

A 2-character (ISO 639-1) or 3-character (ISO 639-2 onwards) code to indicate the language of a document or a text passage. The language code indication consists of two parts separated by a hyphen on the model Lang="[code]-[option]".

- [code] represents a valid language code of 2 or 3 characters;
- [Option] is an indication left to the discretion of the author.

When a country code is used as an option, it can be used to indicate a regionalization of the language, e.g. 'en-us' indicates the American language. The indication of the language code only concerns the part [code] before the dash.

Compatible with assistive technologies

Content or functionality must be compatible with assistive technologies for users as well as with accessibility features of browsers and other user agents *via* an accessibility API.

This concerns both the technology, its functionality and its uses:

- The way the web technology is used must be compatible with users' assistive technologies. This means that the way the technology is used has been tested for interoperability with users of assistive technologies in the language(s) of the content;
- The technology runs natively in widely distributed user agents that are themselves accessibility compliant (such as HTML and CSS) or with a widely distributed, plug-in that is itself accessibility compliant.

Checking compatibility with assistive technologies requires a number of tests specific to the technology used, for example

- Check the name, role, settings and status changes of interface components;
- Check that the rendering of an interface component is correct for the assistive technology (ies) used.

Interface component

An interface component is an element with which the user can interact, for example a button, a link, an input field. Some components can be more complex, such as a menu, a dialogue window, a tab system. Finally, an interface component can be based on native HTML elements or developed from scratch in JavaScript and WAI-ARIA attributes. In particular for elements with WAI-ARIA attributes corresponding to a design pattern it is recommended to consider the WAI-ARIA 1.1 Authoring Practices document when implementing them.

Understandable (reading order)

Understandable content is readable (the order of the elements is logical) and coherent (the sequence of reading is consistent).

Alternative content

Content that replaces another providing the same information but may be presented in a different way. This content may be textual in nature but may also be structured using tags. Alternative content must comply with all the applicable criteria of the RGAA in order to be considered an accessible alternative to the element it replaces. Example: a data table can be the alternative content of a bitmap image (<canvas> tag) displaying a statistical graph.

Hidden content

Assistive technologies (such as screen readers) do not render content hidden via the:

- display with the value none (display: none);
- visibility with the value hidden (visibility: hidden);
- Font-size with value 0 (font-size: 0);
- HTML5 hidden attribute;
- WAI-ARIA attribute aria-hidden="true".

All content using one or more of these properties and attributes is applicable for criterion 10.13.

Visible content

For test 10.2.1: 'Content present' means that visible content remains present when CSS is disabled. For example, an image carrying information as a CSS background property invalidates this test because the information is no longer "present" when CSS is disabled. On the other hand, an image carrying information as a CSS background property but accompanied by hidden text validates this test because the information is indeed "present" when CSS is disabled.

Note: the practice of managing images as background properties of elements *via* CSS is not recommended, even if accompanied by hidden text.

Context of the link

The context of the link represents the additional information (called context information) that can be related by a computer program to the link name. The context information that can be used to complete the link name is as follows:

- The content of the sentence in which the text link is present;
- The content of the paragraph (<p> tag) in which the text link is present;
- The contents of the list item (<li> tag) or the contents of a parent list item (<li> tag) in which the text link is present;
- The content of the title (<hx> tag) preceding the text link ;

- The content of the table header cell(s) (<th> tag(s)) associated with the data cell (<td> tag) in which the text link is present;
- The content of the data cell (<td> tag) in which the text link is present;

Note 1: One of the 6 link contexts combined with the link title should make it possible to understand the function and destination of the link

Note 2: The RGAA considers particular links such as mailto links (which generate a link in the form of a clickable email address) to be sufficiently explicit and do not require the action of sending an email to be indicated.

The authors' attention is drawn to the fact that this general rule can be adapted to the context. For example if the page contains several clickable email addresses with different behaviors (sending an email via the email client for one, accessing a form for the other), it may be necessary to give additional information on the action of the link in order to differentiate their behavior.

Contrast

A sharp contrast between the brightness of a foreground color and a background color. The contrast ratio is based on the difference in relative brightness between the background and foreground according to the rule: (L1 + 0.05) / (L2 + 0.05) where L1 is the lightest relative brightness and L2 the darkest relative brightness. The brightness is calculated according to the following formula: L = 0.2126 * R + 0.7152 * G + 0.0722 * B. Where R, G and B are defined by:

- If RsRGB <= 0.03928 then R = RsRGB/12.92 otherwise R = ((RsRGB+0.055)/1.055) ^ 2.4 ;
- If GsRGB <= 0.03928 then G = GsRGB/12.92 otherwise G = ((GsRGB+0.055)/1.055) ^ 2.4 ;
- If BsRGB <= 0.03928 then B = BsRGB/12.92 otherwise B = ((BsRGB+0.055)/1.055) ^ 2.4 ;

and RsRGB, GsRGB and BsRGB are defined by :

- RsRGB = R8bit/255 ;
- GsRGB = G8bit/255 ;

- BsRGB = B8bit/255.

The character "^" is the exponential operator.

Note: The contrast measure applies to text, image text, text and image text in animations, captioned text and embedded text in videos. For text and image text in animations, captioned text, and embedded text in videos, the font size should be measured against the default display size (as displayed). Text within the elements of an image or video (e.g. a sign, poster etc.) is not affected.

Control (moving or flashing content)

The ability for the user to control the display or playback of moving or flashing content by the keyboard and mouse, at least.

All moving content, with the exception of time-based media supported by the multimedia theme, is covered: moving images (e.g. an animated gif), moving content offered via an <object> tag, JavaScript code or CSS effects for example.

Note 1: Where appropriate, the control method should be available as the first item on the page.

Note 2: The method of controlling moving or flashing content must allow the user to interact with the rest of the page. Therefore, stopping or pausing via an event triggered solely on focus does not meet the criterion.

Note 3: In some cases, movement is an integral part of the component and it is not possible to give control to the user, for example, a progress bar whose function is to indicate by movement the progress of an event such as a download. In this case, the criterion is Not Applicable.

Control (automatically triggered sound)

Possibility for the user to stop or restart an automatically triggered sound.

Note: The sound control method should be available as the first item on the page.

Control of the consultation (of a temporal medium)

The ability of the user to control the viewing of a time-based medium by the keyboard and any pointing device, at least. The following points must be respected:

- List of mandatory consultation control features :
- The multimedia object must always have the following functionality, as a minimum: play, pause or stop;
- If the media object has sound, it must have sound on/off functionality.
- If the media object has subtitles, it must have functionality to control the appearance/disappearance of subtitles;
- If the multimedia object has an audio description, it must have a functionality to control the appearance/disappearance of the audio description.

Each feature must be accessible via the keyboard, via the tab key, and via any pointing device at least.

Each feature must be activatable by the keyboard and by any pointing device, at least.

Note: If there is no sound to a time-based media, there is no need to put a sound on/off feature. If, however, the feature is present and requires a text alternative to be understood by some users, then it should be given one, as the user is likely to access it and want to activate it.

Input control (form)

A set of processes that warn the user of required fields, expected type or format indications and input errors in a form. These input controls can be implemented by the author of the content or be based on attributes (such as required or pattern), WAI-ARIA attributes (such as aria-required) or field types that automatically produce input or error indications (such as URL, email, date, time types for example).

Contiguous background color and contiguous color

Contiguous background color: The background color that is in direct contact with the outer edge of the interface component or graphic element.

Examples:

- For a white button with a blue border on a white background, the white background outside the blue border is the adjoining background color;
- For a red button on a white background, the white background outside the red corresponds to the adjacent background color;

- For a white button with a green border, that turns black on focus and hover, the white background outside the green border of the default state and the black border of the hover and focus state is the contiguous background color.

Contiguous color: color directly in contact with another color. For example, in a "one-way" sign, the red of the sign is the color adjacent to the white line in the center of the sign

Note 1: In the case of a gradient, the least contrasting contiguous color in the gradient is to be considered the contiguous color or contiguous background color.

Note 2: In the case of the presence of several colors, all the colors are to be considered as contiguous color or contiguous background color.

9.4 D

Detailed description (image)

Content associated with an image in addition to its textual alternative in order to fully describe the information conveyed by the image. The detailed description can be associated with the image via:

- A longdesc attribute that contains the address of a page or location within the page containing the detailed description;
- A reference to a detailed description adjacent to the image in the text alternative;
- A link or an adjacent button to access the detailed description on the page or on another page;
- One or more text passages identified by an id and linked by a WAI-ARIA attribute aria-describedby="ID1 ID2 ID3…"

Note 1: If support for the aria-describedby attribute is lacking, it is possible to use one or more text passages identified by an id and linked by a WAI-ARIA attribute aria-labelledby following the textual alternative.

Note 2: To ensure maximum compatibility with user agents, particularly Internet Explorer11, it is recommended that a tabindex="-1 "be implemented on tags that contain a text passage and are not interactive elements (buttons, links, form elements, etc.).

DOM

Document Object Model DOM, is a programming interface standardized by the W3C, which allows scripts to examine and modify web browser content. Through the DOM, the composition of an HTML or XML document is represented as a set of objects - which may represent a window, a sentence or a style, for example - linked in a tree structure. Using the DOM, a script can modify the document in the browser by adding or removing nodes from the tree.

DYS

A shorthand term for some or all of the learning disabilities with the prefix 'dys'.

9.5 E

Graphic element

An element using visual representation such as images, pictograms or graphics.

This element may be composed of one or more parts whose visibility is necessary for its understanding (for example each point on each line of a statistics graph).

Set of pages

Web pages that are linked to each other and form a coherent whole within a website. For example, the pages of a specific section, the pages of a blog, and the administration pages of a customer account are all sets of pages.

Note: the home page of a website may constitute a "set of pages" on its own because of its uniqueness.

Column or row header

The contents of a cell in a data table (usually the first cell in a column or row) that serves as a heading for all or part of the cells in the column or row. A column or row may contain several headers (intermediate header). When the headers apply to an entire row or column they can be structured with a <th> tag or a tag with a WAI-ARIA role="rowheader" or "columnheader" attribute. Otherwise, only a <th> tag can be used.

Note: as only the <th> tag is fully supported by all assistive technologies, it is strongly recommended that this solution is preferred during implementation in order to avoid numerous checks in the different combinations provided in the test environment (or "baseline").

Controlled environment

Any environment in which information access, technologies, conditions of use and user profiles can be known and controlled. The main elements whose control is essential are

- Browser type and version;
- Supported technologies, their version and activation (JavaScript, WAI-ARIA, Flash, Silverlight...);
- Assistive technologies and any devices used specifically by disabled users;
- Operating systems and accessibility APIs supported;
- Training of users of assistive technologies in the use of any particular device (interface, online application...).

Authors and administrators must ensure compatibility of the technologies used and their use by users and their technologies (including assistive technologies). Information services or websites, regardless of their status, that offer public access cannot be considered controlled environments.

Form field label

Text near the form field that indicates the nature, type or format of the expected information. The label can be associated with the form field in several ways:

- By using a <label> tag;
- By using the WAI-ARIA attribute aria-label ;
- By using a link between the text and the field through the WAI-ARIA attribute aria-labelledby ;
- By using the title, attribute.

Important note: when several of these techniques are present in the same field, the calculation of the "accessible name", i.e. what will be returned, follows a strict order:

1. aria-labelledby ;
2. Otherwise aria-label ;
3. Otherwise <label>;
4. Otherwise title.

This order should be used for the assessment of the relevance of the label (criterion 11.2). For example, even in the case of the presence of a <label>, it is the text passage referenced by aria-labelledby that should be taken into account.

An important note about the use of placeholders: when the placeholder attribute is present, it is likely to be rendered instead of the title attribute. Therefore, when both the title and placeholder attributes are present, they must be identical.

Note about labels linked by aria-labelledby: this is one or more text passages identified by ids and linked by aria-labelledby on the following model: aria-labelledby="ID1 ID2 ID3..."

To ensure maximum compatibility with user agents, particularly Internet Explorer 11, it is recommended that a tabindex="-1" be implemented on text passages that are not interactive elements (buttons, links, form elements, etc.).

Note: the aria-label attribute cannot be used to indicate the mandatory nature of a field.

Consistent labels

Labels for form fields on the same page or set of pages that require the same information to be entered must be unambiguously worded so that the user knows that the information they are required to provide is the same.

9.6 F

Style sheet

The CSS language (Cascading Stylesheets) for formatting content elements (e.g. page background color, font size/color, positioning of information in the web page, etc.). CSS styles can be external (CSS file), embedded (declared in the page header) or inline (declared via the style attribute of a tag).

Form

Tag <form> or having a WAI-ARIA role="form" attribute.

9.7 G

Complex and simple gestures

A single gesture involves contact at a single point on the screen. It may be a single press or click, a double press or click, or a long press.

A complex gesture can be

- A gesture involving several points of contact on the screen (e.g. a gesture with two fingers on the screen to zoom in or out on a map);
- A gesture based on the tracking of a trajectory on the screen (e.g. a JavaScript function allowing the detection of the movement of a finger to the left or right on a touch surface to trigger the passage to the previous/next item in a carousel).

9.8 I

Mandatory format specification

A textual or graphical indication (icon) that lets the user know that the input of a field is mandatory prior to input.

Note: If this indication is not made explicitly in text (icon, *, !,...) the explanation of the meaning of this indication must be visible and in the order of the source code before the first use of the indication.

Indication of the type of data and/or format

Textual indication allowing the user to know what type of data and/or input format is required by a mandatory field, prior to its completion.

Examples:

- Email (format: vous@domaine.com);
- Postal code (format: 00000);
- Date (format: DD/MM/YYYY).

Decoration image

An image that has no function and conveys no particular information in relation to the content with which it is associated. Examples:

- An image preceding each item in a list;
- An image to set the layout;
- A rounded corner image to dress up an information block;

- An illustrative image that does not provide any information necessary for the understanding of the text with which it is associated.

Note: tags with a WAI-ARIA role="img" attribute can only act as a decorative image if they have a WAI-ARIA aria-hidden="true" attribute.

Image object

Image embedded in or generated by an <object> tag.

Image as a carrier of information

An image that conveys information necessary for the understanding of the content with which it is associated.

Note 1: When the image is the only content of a link, its alternative is the link title. In this case, the image alternative should be evaluated with the "Links" theme.

Note 2: When a form button, inserted with the <button> element, contains only an image (<img>, <object>, <embed>, <canvas> or <svg> tag), the alternative to the image is the button label. There are two possible cases:

- The button is controlled by its type, for example, submit or reset, and is part of a form. In this case, the image button should be evaluated with the "Forms" theme.
- A JavaScript device controls the button. In this case, the image button must be evaluated with the "Scripts" theme.

Reactive image

1. Client-side **responsive image** (usemap attribute): image divided into clickable or neutral areas (nohref attribute);
2. **Server-side responsive image** (ismap attribute): An image for which the browser sends the pointer coordinates to the server, each set of coordinates corresponding to a resource (web page). The server-side responsive image is extremely rare.

Note: in HTML5, the ismap attribute is obsolete for image buttons (input type="image").

Test image

Image used in a test, Captcha or an image used as a test in a quiz or game. Example: a series of images shows a detail from famous paintings; you have to recognize the title and the painter of each painting. In this situation, it is not possible to give a relevant alternative (e.g. the name of the painting and/or the painter) without making the test unusable. The alternative must then simply give the possibility of identifying the picture, e.g. "picture 1 of the test".

Image text

Image displaying text.

Note: the use of text images is not recommended. Where it is possible to reproduce the same effects in CSS, criterion 1.8 [AA] requires that the text be reproduced in CSS text, or that a replacement mechanism allows the user to replace these images with CSS styled text.

Image text object

Image generated by the <object> tag and displaying text.

Image conveying information (given by color)

An image in which all or part of the content visually conveys information by means of color only.

Indication by shape, size or position

This may include, for example:

- The presence of a visual marker, to indicate the active page in a navigation menu (indicated by the position);
- A foreground to indicate an active tab (indicated by the shape);
- A change of font size in a tag cloud (indicated by the size).
- Or any other similar graphic effect.

Information (given by color)

Information conveyed visually through color. The indication that red fields are mandatory in a form, the use of a blue background to indicate the page currently being consulted in a menu with a green background, the change of color of an item name to indicate its unavailability in a list of items are all examples of indications given by color.

Where color information is accompanied by another method for users who do not see or perceive colors or their associations well, the criterion shall be considered not applicable.

Ways of conveying information other than by color can be:

- A visible textual indication;
- A means involving graphics (pictogram, background image, shape, different border style, etc.) and by means of a code complement (aria-label, title, hidden text, aria-current, etc.)
- Another typographic style (bold, italic, text size, other font, etc.) and by means of a complement in the code (aria-label, title, hidden text, aria-current, etc.).

Title (or accessible name) of link

"Accessible name" rendered by assistive technologies.

In the case of an HTML link, the "accessible name" is obtained in the following order:

- Text passage associated with the WAI-ARIA attribute aria-labelledby;
- Otherwise, content of the WAI-ARIA attribute aria-label;
- Otherwise, content of the link ;
- otherwise, content of the title attribute;

This order should be used to determine what constitutes the title of the link. For example:

- In the case of a WAI-ARIA aria-label attribute and a WAI-ARIA aria-labelledby attribute, the text passage referenced by the WAI-ARIA aria-labelledby attribute shall be considered as the label;
- If a WAI-ARIA aria-label attribute and content are present in the link, the content of the WAI-ARIA aria-label attribute shall be considered as the title.

In the case where the "accessible name" is obtained from the content of the link, this will vary according to the following cases:

Text link: In HTML, the "accessible name" corresponds to the text made up of:

- the text contained in the link;
- of the text contained in the child elements of the link.

Image link: In HTML, the "accessible name" corresponds to the text made up from the textual alternative of one or more images in the link of the type:

- image (<img> tag or tag with WAI-ARIA role="img" attribute) ;
- image object (tag <object>) ;
- bitmap image (tag <canvas>) ;
- vector image (tag <svg>).

Composite link: In HTML, the "accessible name" corresponds to the text made up of the whole:

- the text contained in the link;
- the text contained in the child elements of the link;
- the content of the textual alternative of the image(s) included in the link.

In the case of an SVG link (version 1.1), the 'accessible name' is obtained as follows:

- Text passage associated with the WAI-ARIA attribute aria-labelledby;
- Otherwise, content of the WAI-ARIA attribute aria-label;
- Otherwise, content of the <title> element direct child of the link;
- Otherwise, content of the x-link: title attribute ;
- Otherwise, text content of one or more <text> elements.

However, care should be taken, as this calculation algorithm is not yet supported and effective within the various screen readers. To date, support is available with Voiceover, but incomplete or lacking with JAWS and NVDA. Therefore, the lowest common denominator that can be relied upon to provide a title to the link is the <text> element.

Note 1: A link title will be considered non-explicit if the "accessible name" does not include the visible title of the link.

Note 2: Due to the possible configuration of technical aids allowing forcing the restitution of «accessible name» from the content of the title, attribute at the expense of the "accessible name" from the content of the link. A link title will be considered as non-explicit if the link has a title attribute whose value does not include at least the "accessible name" from the link content.

Note 3: In the case of the presence of several links with a different destination whose "accessible name" is identical. The link name alone will be considered as non-explicit if the link context does not allow differentiating them.

Note 4: Where a link has no content (except for anchors), it will be non-compliant with respect to criteria 10.2 and 6.4.

Note 5: Although the calculation of the accessible name of a link takes into account text content generated in CSS via the ::before and ::after pseudo-elements, this practice must not be used as it constitutes non-compliance with WCAG 2.1 criterion 1.3.1 (see F87).

Visible title

The text displayed as a visible heading on the screen within a button or link.

Text displayed as a label for a form field.

This text can be text or an image containing text.

Items of the same nature in the choice lists

In a drop-down list (<select> tag), a set of items (<option> tags) that can be grouped by their nature. The grouping aims to identify items that should be treated as a whole (for example, a list of departments grouped by regions).

9.9 L

Landmark s

WAI-ARIA provides roles to indicate the main areas (regions) of the document. These roles are very useful for screen reader users in particular, but also for keyboard navigation users who can benefit from quick navigation features in the document structure.

The roles should be defined in the document according to the nature of the area:

- The header area must have a WAI-ARIA role="banner" attribute.
- The main navigation menu must have a WAI-ARIA role="navigation" attribute.
- The main content area must have a WAI-ARIA role="main" attribute.
- The footer area must have a WAI-ARIA role="contentinfo" attribute.
- The search engine area on the site must have a WAI-ARIA role="search" attribute.

Note 1: While most screen readers provide these features, browsers have not yet provided dedicated navigation functionality for users who cannot use the mouse. The use of avoidance links is therefore still preferred over landmarks.

Note 2: The WAI-ARIA banner, main and contentinfo roles must be unique within the page. The WAI-ARIA navigation role is reserved for the main and secondary navigation areas. When there are several WAI-ARIA navigation roles, it may be useful to differentiate them by specifying a name for each of the areas via the WAI-ARIA aria-label or aria-labelledby attribute.

Default language

Indication of the main processing language of the document, which can be present on the html root element or on each element of the page concerned via the Lang and/or xml: Lang attributes according to the following scheme:

- For HTML up to version 4.01: lang attribute mandatory, xml:lang attribute not supported;
- For XHTML 1.0 served in "text/html": lang attribute and xml:lang mandatory;
- For XHTML 1.0 served in "application/xhtml+xml": xml:lang attribute mandatory, lang attribute recommended;
- For XHTML 1.1: xml:lang attribute mandatory, lang attribute not supported;
- For HTML5: Lang attribute required.

Name, role, value, settings and status changes

A component must have an appropriate role and name. Its values, states and parameters, if any, must also be accessible and correctly passed to accessibility APIs and others.

A component may rely on an interactive HTML element or on a non-interactive element overloaded by WAI-ARIA via an ad hoc role. Important: buttons (<button> or <input type="button"> tags) when controlled via JavaScript are to be evaluated with criterion 7.1.

The name can be the name of the component (the name of a button, for example).

The value is, for example, the selected item in a drop-down list or the current value of a slider.

The role corresponds to the type of element defined by the HTML or WAI-ARIA specification (such as the <button> tag or the WAI-ARIA role="button" attribute).

The parameterization corresponds to the particular information of a component, generally made available by WAI-ARIA. For example, aria-controls is a parameter that passes on to the APIs the information that the component controls such and such content (referenced by its identifier - id attribute).

State changes are also made available by WAI-ARIA. For example, aria-expanded is a state that signals to APIs that the component is "open" or "closed". Note that a state can also be passed via the name, where the label is dynamically changed to match the state of the controlled area.

These parameters are not mandatory but may be required if they are essential to make the component accessible. It is up to the auditor to consider when these parameters are required depending on the context in which the component is used.

The auditor should also check that, where present, these parameters are used correctly.

To do this (if he considers it relevant given the context of the components' implementation and the ergonomic choices implemented) he can rely on the WAI-ARIA usage recommendations for components with WAI-ARIA attributes corresponding to a design pattern as described in the WAI-ARIA 1.1 Authoring Practices document.

Note: WAI-ARIA roles, properties and states are implemented via attributes, for example role="banner", aria-hidden="true".

Legend

HTML provides a facility for titling groups of like fields via <fieldset> and <legend> elements.

It is also possible to create groupings with the WAI-ARIA group role and a text passage, acting as a legend, linked by the WAI-ARIA aria-labelledby attribute or implemented via a WAI-ARIA aria-label attribute.

Note 1: Field groupings can use other methods that associate the grouping information directly into the field label. For example, via a title attribute, a WAI-ARIA aria-label attribute, an aria-labelledby link acting as a label, or the WAI-ARIA aria-describedby attribute associating additional text. In this case, the grouping of fields becomes unnecessary since the labels are sufficiently relevant.

Note 2: Where the form consists of a single block of information of the same nature (e.g. the user's identity and address) or a single field (e.g. a search engine), grouping of fields is not mandatory.

Image caption

When a text, adjacent to an image, contains information about the image (e.g. copyright, date, author...) or is intended to complete the information provided by the image (e.g. a text associated with an image in an image gallery), it is called an image caption.

When an image is captioned, it is necessary to associate the image caption with the image by a structural relationship, so that assistive technologies can treat the image and its caption as a single whole.

HTML5 offers the possibility of associating a caption with an image via the elements figure (the whole image and the caption) and figcaption (the caption).

An image without a caption can define:

- An image that is not inserted in a figure element ;
- An image inserted in a figure element without a figcaption element.

Note: Where text adjacent to the image can be used as replacement text, it is not necessary to use the figure, figcaption set, as the image can simply be treated as a decorative image.

See the W3C's Requirements for providing text to act as an alternative for images.

Link

In HTML:

- A <a> tag with an href attribute;
- A tag with a WAI-ARIA role="link" attribute whose navigation action is handled by a script.

In SVG:

The <a> tag with an xlink:href attribute in SVG 1.1.

Link or adjacent button

Link or button presented adjacent to an element in the page. The link or button must be visually adjacent in the graphical representation (CSS enabled) and in the HTML code. In the HTML code, the link or button must be located just before or just after the element to which it is adjacent.

Composite link

In HTML, a link containing both text and one or more image children:

- image (<img> tag or tag with WAI-ARIA role="img" attribute) ;
- clickable area (<area> tag) with an href attribute;
- image object (tag <object>) ;
- bitmap image (tag <canvas>) ;
- Vector image (tag <svg>).

Important note: It is recalled that the use of two adjacent (image link and text link) and identical links constitutes a significant annoyance to the user. Even if this does not constitute non-compliance, this usage should be avoided. One way of dealing with this type of link is to include the image in the text link to form a composite link, thus avoiding redundancy.

See H2: Combining adjacent image and text links for the same resource.

Link of which the nature is not clear

A link that can be mistaken for normal text when it is signaled by color only by certain types of users who do not perceive color or do not perceive it well. For example, in this text "New strike at SNCF", if the word "strike" is a color-only link, non-color-perceiving users accessing the activated CSS content may ignore its nature. On the other hand, in this text "New SNCF strike, read more" if "read more" is a link, a non-color-perceptive user will have no problem understanding its nature.

Note: "indicated only by color" means that any visual marker (icon, underline, border does not accompany the link...). Consequently, a link of the same color as the surrounding text is concerned by this criterion.

Image link

In HTML, a link containing only one or more image children:

- image (<img> tag or tag with WAI-ARIA role="img" attribute) ;
- clickable area (<area> tag) with an href attribute;
- image object (tag <object>) ;
- bitmap image (tag <canvas>) ;

- vector image (tag <svg>).

Text link

In HTML, a link containing no child elements of type:

- image (<img> tag or tag with WAI-ARIA role="img" attribute) ;
- image object (tag <object>) ;
- bitmap image (tag <canvas>) ;
- vector image (tag <svg>).

Avoidance or quick access links

Links whose function is to allow navigation within the page (skip link, link to the search form or menu, etc.). These links can either allow access to an area of the page (quick access link) or skip an area in the page (skip link).

Note 1: An avoidance or quick access link that, when activated, would not allow resumption of reading and keyboard navigation from the target of the link when using the browsers and technical aids selected in the test environment (or 'baseline') of the audit would be considered non-compliant.

Note 2: Avoidance or quick access links must be located in the same place in the presentation and in the same relative order in the source code in order to meet criterion 12.2.

Identical links

Two links are said to be identical when link x (link title alone, content of the title attribute or context of the link) is equal to link y. This definition applies to all types of links: text link, image link (the links have the same image) and composite link.

Please note: links with identical headings but different link titles or different link contexts are not identical (example: <a href="link_bar.html" title="click here to download the toolbar">click here</a> and <a href="link_doc.html" title="click here to download the document">click here</a>).

SVG link

Link contained in an <svg> element

In the case of an SVG link (version 1.1), the 'accessible name' is obtained as follows:

- Text passage associated with the WAI-ARIA attribute aria-labelledby;
- Otherwise, content of the WAI-ARIA attribute aria-label;
- Otherwise, content of the <title> element direct child of the link;
- Otherwise, content of the x-link:title attribute ;
- Otherwise, text content of one or more <text> elements.

However, care should be taken as this calculation algorithm is not yet supported and effective within the various screen readers. To date, support is available with Voiceover, but incomplete or lacking with JAWS and NVDA. Therefore, the lowest common denominator that can be relied upon to provide a title to the link is the <text> element.

Lists

A sequence of items that can be grouped together in the form of an ordered, unordered or definitional structured list. For example, the sequence of links in a navigation menu is an unordered list of links, the different steps in a procedure is an ordered list of items, the term/description pair in a glossary is a description list. In HTML, lists use the following tags:

- Ordered list: <ol> and <li> tags (each element of the list is assigned an indexed marker) ;
- Unordered list: <ul> and <li> tags (each element of the list is assigned an unindexed marker ;
- Description list: tags <dl>, <dt> (term to be described) and <dd> (description).

Note 1: In HTML5, the <dl> tag no longer represents just a list of definitions, but generically any list of descriptions that may include as a group of term-descriptions names and definitions, questions and answers, categories and topics, etc.

Note 2: It is also possible to structure lists using the WAI-ARIA attributes role="list" and role="listitem" for ordered and unordered lists.

Note 3: the notion of "visually grouped in a list" is characterized by:

- The presence of a visual marker to indicate that it is an unordered list, e.g. -, -, *, etc.

- The presence of a visual marker to make it clear that this is an ordered list, e.g. a number, a Greek letter, etc.
- the presence of a series of elements visually following each other, with identical visual form, nature and function, but without having a direct visual marker of a list (unordered or ordered), such as a navigation menu.

However, not all lists require a list structure, for example a series of terms separated by a comma.

List of possible values for the autocomplete attribute

The list of available values is provided by the WCAG 2.1 specification:

- name - Full name;
- honorific-prefix - Abbreviation, title or title;
- given-name - First name;
- additional-name - Additional names;
- family-name - Family name;
- honorific-suffix - Honorific suffix;
- nickname - nickname, diminutive;
- organization-title - Function, job title;
- username - User name;
- new-password - New password (for example, when creating an account or changing a password);
- current-password - Current password for the account identified by the username field (e.g. when logging in);
- organization - Name of the organization corresponding to the person, address or contact information in the other fields associated with this field;
- street-address - Postal address (multiline, new lines kept);
- address-line1 - Postal address (one line per field, line 1);
- address-line2 - Postal address (one line per field, line 2);
- address-line3 - Postal address (one line per field, line 3);
- address-level4 - The most detailed administrative level, for addresses with four administrative levels;
- address-level3 - The third administrative level, for addresses with three or more administrative levels;

- address-level2 - The second administrative level, for addresses with two or more administrative levels;
- address-level1 - The broadest administrative level of an address, i.e. the province in which the locality is located;
- country - Country code;
- country-name - Country name;
- postal-code - Postal code, CEDEX code (if CEDEX is present, add 'CEDEX', and what follows it should be added in the address-level2 field);
- cc-name - Full name on the means of payment;
- cc-given-name - First name on the means of payment;
- cc-additional-name - Additional first names on the means of payment cc-family-name - Family name on the means of payment;
- cc-number - Code identifying the means of payment (e.g. a credit card number)
- cc-exp - Expiry date of the means of payment;
- cc-exp-month - The month of the expiry date of the means of payment;
- cc-exp-year - The year of the expiry date of the means of payment;
- cc-csc - Payment Security Code (also known as the card security code (CSC), card validation code (CVC), card verification value (CVV), signature panel code (SPC), credit card ID (CCID), etc);
- cc-type - Type of payment method (e.g. Visa);
- transaction-currency - The currency that the user prefers in a transaction;
- transaction-amount - The amount that the user prefers in a transaction (e.g., in response to an auction or clearance price);
- language - Preferred language;
- bday – Birthday;;
- bday-day - The day of the birthday;
- bday-month - The month of the birthday;
- bday-year - The year of the birthday;
- sex - Sexual identity;
- url - Home page or other web page corresponding to the organization, person, address or contact information in the other fields associated with this field;
- photo - A photograph, icon or other image that matches the organization, person, address or contact information in the other fields associated with this field;
- tel - Full telephone number, including country code;

- tel-country-code - Country code of the telephone number;
- tel-national - Telephone number without the country code part, with an internal country prefix, if applicable;
- tel-area-code - Area code of the telephone number, with a country prefix, if applicable;
- tel-local - Telephone number without the country code part and the area code.
- tel-local-prefix - The first part of the telephone number component following the area code, when this component is split into two parts;
- tel-local-suffix - The second part of the telephone number component following the area code, when this component is split into two parts;
- tel-extension - Telephone number of an internal extension;
- email - Email address;
- impp - corresponding URL of an instant messaging protocol (e.g. "aim:goim?screenname=example" or "xmpp:fred@example.net").

9.10 M

Replacement mechanism

A mechanism generally based on CSS, allowing the user to replace text with image text and vice versa on the principle of style switcher. The mechanism can use a server-side scripting language or a client-side scripting language.

Mechanism for displaying a consistent contrast ratio

An interface component whose activation allows the appearance of the site or page to be modified so that content is displayed with a sufficient contrast ratio. The design of this interface component shall comply with criteria 3.2 and 3.3 without itself employing a mechanism to display a compliant contrast ratio.

Non-temporal media

Content that does not unfold in time, which can be viewed *via* a plugin (Flash, Java, Silverlight, etc.) or *via* svg and canvas elements; for example, an interactive map in Flash, a Flash or Java application, or a slideshow are non-temporal media. Non-temporal media can contain temporal media (e.g. a Flash player that offers a list of videos to view).

Note: Using the wmode parameter for a Flash object with the values "transparent" and "opaque" effectively invalidates criterion 4.12. Using these values results in the Flash animation being invisible to screen reader users.

Time-based media (sound, video and synchronised type)

Time-based media only audio: sound content (Wav, Mp3...) ;

Time-based media only video: moving or sequential images or photos;

Synchronised time-based media: audio or video streams synchronised with another format to present information and/or with interactive time-based components. Time-based media can be accessed in 2 different ways:

Downloadable file that can be viewed with software external to the web page;

Content embedded in the web page and viewable in the web page *via:*

- A plugin (e.g. a video played by a Flash player) ;
- The <video> element (e.g. a video);
- The <audio> element (e.g. a podcast);
- The <svg> element (e.g. a vector cartoon);
- The <canvas> element (for example a bitmap cartoon).
- The <bgsound> element to play a background sound to the web page.

Time-based media can be broadcast in real time or offered for playback asynchronously (pre-recorded media).

Note 1: Using the wmode parameter for a Flash object with the values "transparent" and "opaque" effectively invalidates criterion 4.11. Using these values means that the Flash animation seen from the screen reader users' side is invisible.

Note 2: animated gifs, image animations made with JavaScript or CSS are not considered temporal media.

Note 3: The <bgsound> element is Internet Explorer specific and should not be used.

Status message

A status message informs the user of a change in content on the page without interrupting the user's main activity (there is no change in context, e.g. a shift in focus to the message).

A status message can inform about:

- The success or outcome of an action;
- The busy state of an application;
- The state of progress of a process;
- The existence of errors.

Modify or cancel data and actions taken

Procedures, by which a user can change the data he or she has entered, have his or her entry reversed, or have actions resulting from his or her entry reversed, e.g. cancellation of an order or a bank transfer.

Note: The page containing a form that modifies or deletes data, or transmits answers to a test or exam, or whose validation has financial or legal consequences, must explicitly indicate the period during which the user can request the cancellation of his or her entry. It should also contain the procedure to be followed to cancel the input. This procedure does not have to be available online, although it is recommended.

Search engine (internal to a website)

Area containing the search engine for searching the contents of the entire site.

Note: Be careful not to confuse this search zone, which is unique in the site, with any other search engine that allows you to search a restricted part of the site: a catalogue, the offers in a public procurement section, etc.

See the technical definition of header field provided by WAI-ARIA search (role).

Design pattern

A Design Pattern is a template defined in the document: WAI-ARIA 1.1 Authoring Practices. which describes the structure, roles and properties, and keyboard behavior that a JavaScript component (widget) must follow.

It is recommended that components developed in JavaScript using WAI-ARIA attributes corresponding to a design pattern respect the design pattern. However, not all design patterns are suitable for non-application use, especially for sites with a mobile display.

Note 1: Given the lack of support for some WAI-ARIA properties and roles, and the wide variability of situations in which a JavaScript component may be offered, it is possible to adapt design patterns to particular contexts or uses. In this case, the adapted design pattern should:

- Respect the general structure: for example a set of panels (WAI-ARIA tabpanel role) of a tab system is necessarily linked to a set of tabs (WAI-ARIA tablist role);
- Use an equivalent WAI-ARIA role or property with similar behavior and output to replace a poorly supported WAI-ARIA role or property.

Note 2: Enriching a design pattern with additional WAI-ARIA roles or properties whose accessibility compatibility is checked by the rendition test on the test environment (or "baseline") does not constitute an adaptation of a design pattern. For example, adding the WAI-ARIA attribute aria-hidden to the panels (WAI-ARIA role tabpanel) of a tab system does not define an adapted design pattern.

9.11 O

Tabulation order

The order in which the focus moves (to a next element or to a previous element). The natural order is that which is implemented *via* the source code. When it is modified by the use of the tabindex attribute or by the use of a JavaScript command, it is the modified order that is referred to.

Warning: when an element initiates a change in the page (change of context, management of hidden zones, addition of content, management of form fields, etc.) it is necessary to activate the element which initiates the change in order to test the consistency of the tab order.

9.12 P

Site map page

A dedicated page presenting the tree structure of a website, generally in the form of lists of links organized into headings and sub-headings giving access to all the pages of the site.

Note 1: Sitemap links can be made up of <a> tags or <area> tags.

Note 2: It is not necessary for the sitemap to contain links to all the pages of the site, but it is necessary for the user to be able to reach all the pages of the site from the sitemap.

Passage of text associated with the data table

Text in a <caption> tag or in a tag associated with the table given via the WAI-ARIA attribute aria-labelledby on the table.

Passage of text linked by aria-labelledby or aria-describedby

This is one or more text passages identified by ids whose value is unique in the page and associated with an element (form field, button, etc.) by the WAI-ARIA attributes aria-labelledby or aria-describedby on the following model: aria-labelledby="ID1 ID2 ID3..." where the value of the attribute used is equal to the list of id attribute values of the text passages to be associated in the page.

Note 1: To ensure maximum compatibility with user agents, in particular Internet Explorer 11, it is recommended to implement a tabindex="-1" on text passages that are not interactive elements (buttons, links, form elements, etc.).

Note 2: The value of the WAI-ARIA attributes aria-labelledby or aria-describedby must not create a recursive reference (A refers to B which refers to A) or a traversal reference (A which refers to B which refers to C);

Relevance (information other than color)

The means of retrieving information other than by color must be accessible to all. For example, in the case of a list of items where the yellow items are on special offer, the use of hidden text *via* CSS is a way of retrieving the "on special offer" information, but it is not relevant because this information will remain hidden to the user who views the activated CSS page.

Note: using an emphasis tag (<strong> or <em>) as an alternative means of retrieving information given by the color allows the criterion to be validated even though these elements are generally not supported by assistive technologies, particularly screen readers.

Presentation of information

Visual rendering of content via a browser in graphic mode. The presentation concerns the style, position and dimensions of HTML elements and their content. The presentation of the information must be done using CSS. The elements (basefont, blink, centre, font, marquee, s, strike, tt and big) and attributes (align, alink, background, bgcolor, border, cellpadding, cellspacing, char, charoff, clear, compact, color, frameborder,hspace, link, marginheight, marginwidth, text, valign, vlink, vspace, size) are forbidden

Note: Width and height attributes used on elements other than <img>, <object>, <embed>, <canvas> and <svg> tags are prohibited.

Pressed or posed

Corresponds to JavaScript event handlers considered as top-down events (e.g. mousedown, touchstart).

Taking the focus

Focusing is the state returned by an element that receives attention as a result of a user action. There are three ways in HTML to give focus to an element:

- By activating the element with a pointing device (e.g. mouse);
- By reaching the item with the tab key or shift + tab key;
- By activating the item with a keyboard shortcut (accesskey).

Some elements naturally receive focus, for example: <a href>, <area href>, <button>, <input>, <object>, <select>, <label>, <legend>, <optgroup>, <option> and <textarea>. The behaviour of the element, when the focus is taken, depends on its nature; a link, for example, will have to be activated after the focus is taken (unless a script is used). On the other hand, a form element, such as <textarea>, should allow input after the focus is taken. The <label> and <legend> elements only receive focus capture via the mouse pointer. For the <label> element, the expected behaviour is to transfer the focus capture to its associated element.

Note 1: The WAI-ARIA specification extends the role assigned to the tabindex attribute by defining that any HTML element can acquire the ability to receive focus by assigning it the value tabindex="0". On the other hand, no behavior is assigned by the mere presence of tabindex. Similarly, the value tabindex="-1", when used on an element that naturally receives focus, removes the affected element from the tab plane by inhibiting its ability to signal, "focus". The use of tabindex, in accordance with the WAI-ARIA specification, can validate certain tests relating to the management of tab focus, for example.

Note 2: The visual indication of the browser's focus must not be removed or degraded unless a focus style defined by the author is visible and sufficiently contrasted in accordance with criterion 3.3.

Cooling method

A technique for changing the content of one or more elements of the web page. The refreshment process can be done by automatic page reload or dynamically without reloading the page (e.g. *via* AJAX). The user must be able to control each refresh process independently.

CSS property determining a color

This concerns the following properties: color, background-color, background, border-color, border, outline-color, outline.

Note: the use of a background image to insert a color (background:url... property) is also concerned.

9.13 R

Shortcut keyboard

A means of triggering an action associated with a user interface component by pressing one or more keys.

Note: "Access keys" (HTML accesskey attribute) are indeed keyboard shortcuts, but they are not concerned by criterion 12.15 insofar as their activation already requires the use of modifier keys (which vary according to the browser).

Redirection

A process that consists of redirecting the user to another page on the same or a different domain when a page is displayed on the client computer.

Writing rules

The source code must respect the following rules in accordance with the document type declaration used in the page:

- no opening or closing tag without < or > (example of error: li>toto);
- no closing tag with missing / (error example: <li>toto<li>);
- no attribute value with missing ' or ' (error example: alt="toto);
- no multiple attribute values separated by a space without ' or ' (error example: alt=hello toto);
- no missing spaces between attributes (example: alt="toto "title="toto");
- no missing closing tag for elements that require one (error example: <object> without </object>).

Released or raised

Corresponds to JavaScript event handlers considered as bottom-up events (mouseup, touchend for example).

Correctly rendered (by assistive technologies)

Where a criterion, test or test condition requires verification of the rendition of a device, it must be ensured that the rendition is compatible with accessibility.

The test consists of verifying that the rendering is relevant for at least one of the combinations of the test environment (or "baseline") used to declare that an element, device or alternative is "accessibility compatible".

For example: Test 1.3.7 asks to verify that the alternative of an image carrying vector information is correctly rendered.

A test is then carried out with NVDA (latest version) and Firefox, JAWS (previous version) and IE9+, and Voice Over (latest version) and Safari.

If the alternative is found to be correctly rendered, the test is validated.

Summary (of table)

A summary is a passage of text associated with a complex data table. It provides information about the nature and structure of the table to facilitate its use by users of assistive technologies for example.

Note: in HTML5, the only technique that can currently be used is to insert the summary directly into the title (<caption> tag), masking the summary via CSS if necessary.

In previous versions of HTML, the summary can be inserted via a summary attribute on the <table> tag

In the case of a tag with the WAI-ARIA role="table" attribute, the summary must be provided by means of an aria-describedby attribute and be correctly rendered by assistive technologies.

9.14 S

Script

Code usually written as a list of commands (e.g. JavaScript). Client-side interpreted languages require a compatible browser on which the execution of the language is active. The commands of a client-side scripting language can be embedded or contained in an external file. In both cases, insertion is done *via* the <script> tag.

Reading direction

Indicates the reading direction of the document or of a text passage *via* the dir attribute, dir="ltr", for example. The two recognised values are:

- ltr (left to right) indicates a left to right reading direction;
- rtl (right to left) indicates a right to left reading direction.

Note: if there is no indication of the reading direction *via* the dir attribute on the html element, body, or one of the parents of the text being analyzed, the default reading direction is from left to right (value ltr).

If necessary (visible text in addition to the aria-label or aria-labelledby attribute)

The WAI-ARIA attributes aria-label and aria-labelledby can be used to create a form field label. The created label will remain invisible, which solves many cases where a visible label is not desired.

However, this creates a potential risk for some users and it may be necessary for a visible text attached to the field to complement the feature provided by the WAI-ARIA attribute aria-label or aria-labelledby.

Website: all pages Web

Linked by web links ;

Belonging to the same domain name (e.g. references.modernisation.gouv.fr) ;

Which form a coherent whole from the user's point of view.

Special case of web pages in a sub-domain; a sub-domain can :

- Or belong to the website attached to the domain name, if the user has a coherent perception of it with the other pages of the website (for example: same structure, same navigation...);
- Either not belonging to the website attached to the domain name (e.g. different blogs in a sub-domain of a domain name and unrelated to each other).

Synchronised subtitles (multimedia object)

Text of the audio information (words of a character, noise important to understand the action...) present in a temporal media and displayed synchronously with the flow of the multimedia object.

Note 1: to differentiate between sound sources (different characters, voice-overs, etc.), it is recommended to use an appropriate mechanism (bracketing, italics, explicit announcement such as "voice-over: ...").

Note 2: subtitling for translation (kind="subtitles" attribute of the <track> tag in HTML5, for example) should not be confused with subtitling for the deaf and hard of hearing (kind="captions" attribute of the <track> tag in HTML5, for example). These two types of subtitling have different purposes. Only the presence and relevance of subtitling for the deaf and hard of hearing allows compliance.

Structure of the document

A set of elements used to define the main areas of an HTML page, such as the page header area, the main and secondary navigation areas, the main content area and the footer area.

Navigation system

Any process allowing navigation within the site or within a page, the navigation systems retained are :

- Main navigation menu ;
- Table of contents;
- Site map ;
- Search engine.

9.15 T

Tabindex attribute

The tabindex attribute can be used to indicate whether an item can be reached via keyboard navigation (using the Tab key) and, if so, in what order the items should be browsed. Its value is an integer; it can take the following values:

- a negative value: the item cannot be reached via keyboard navigation;
- zero: the item can be reached via keyboard navigation;
- a positive value: the element can be reached via keyboard navigation and the elements will be browsed in ascending order of the attribute values.

If several elements share the same attribute value, the tabbing order will be that of the elements in the document.

Since HTML5, tabindex can be used on any element. In HTML 4.01, tabindex could only be used on elements that naturally receive focus (a, area, button, input, object, select, textarea).

The use of tabindexes is not recommended; it is better to arrange the elements in a logical order in the HTML code.

Data table

A data table is a structure introduced by a <table> tag or when correctly rendered by assistive technologies by a tag with a WAI-ARIA role="table" attribute. This tag allows information to be structured in rows and columns via data cells and header cells.

Data table with a title

An array of data that has an attribute or contains a tag whose content acts as a title.

A table of data preceded or followed by a passage of text associated with the table as a title.

If it is correctly rendered and associated by assistive technologies with the data table, the associated title can be:

- In a <caption> tag;
- In a title attribute ;
- In a WAI-ARIA attribute aria-label ;
- In a tag associated with the data table via a WAI-ARIA attribute aria-labelledby on the table.

Note: As only the <caption> tag is fully supported by all assistive technologies, it is strongly recommended that this solution be preferred during implementation in order to avoid numerous checks in the different combinations provided by the test environment (or "baseline").

Complex data table

A data table is a structure introduced by a <table> tag or when correctly rendered by assistive technologies by a tag with a WAI-ARIA role="table" attribute.

When a data table contains headings that are not distributed only in the first row and/or column of the grid or whose scope is not valid for the whole column or row, it is called a complex data table. It is then necessary to provide a "summary" to explain its nature and structure in order to make it easier to consult for users of assistive technologies for example.

Formatting table

A technique that uses an HTML element (<table> tag) to control the display of information *via* cells (<td> tag).

Character size

The value assigned to the fonts on a web page.

Styled text

Text whose formatting is controlled by a style sheet.

Title

HTML element (h tag) with 6 levels of hierarchy (from h1 for the most important title to h6 for the least important) or HTML element with WAI-ARIA attributes role="heading" and aria-level allowing to structure the information of a Web content.

The hierarchy between headings must be respected in a web page, and headings may not be skipped (an h3 heading may not come directly after an h1 heading, for example). On the other hand, the hierarchy does not have to start with an h1. Even if this usage is not encouraged, it is considered compliant to start the title hierarchy of a page with a level other than level 1.

Note: titles hidden via CSS are considered to be present and validate criterion 9.1.

Executive title

Content of the title attribute of the <iframe> or <frame> tag allowing to know the nature of the content delivered *via* the frame when the user navigates from frame to frame or displays the list of frames on the page for example.

Note 1: Some frames are only used for technical operations such as application processing to prepare or control content displayed on the page, such as the frames used by some social networks such as Facebook.

If these frames do not have a frame title provided by the remote service, or if the frame titles are deemed irrelevant, generic mentions can be used, e.g. title="Facebook technical content".

Note 2: If this does not interfere with the operation of this type of framework, it is possible to make them unavailable to assistive technologies by using the WAI-ARIA attribute aria-hidden="true". In this case criterion 2.2 will not apply.

Page title

Content of the <title> tag of a web page that clearly, concisely and uniquely identifies the contents/nature of the page ("Site map www.nomdusite.fr" for a page presenting the website map, for example).

Textual transcript (time-based media)

Textual content associated with a time-based media object by the appropriate technique (text encoded in HTML or in a text file that is on the same page or searchable via a link). This content gives the user access (independently of the consultation of the multimedia object) to:

- All that is expressed orally;
- All descriptive information necessary for an equivalent understanding of the action.

This textual information should be presented in the chronological order of its appearance in the time medium.

Note: the textual transcription must be outside the <object> tag.

Type of document

A set of reference data that allows user agents to know the technical characteristics of the languages used on the Web page (doctype tag).

Data type and format

An indication of the type and format of data expected when entering a form field. For example:

- Date (dd/mm/yyyy) ;
- Amount in euros ;
- Postcode (5 digits: e.g. 75001).

Important note: When the form field type offers an input mask, e.g. date or time fields, the format indication is not necessary.

9.16 U

For presentation purposes only

For presentation purposes only: use of HTML tags for a purpose other than that specified in the specifications (with respect to the type of document declared). Examples: use of h tags for the sole purpose of creating a typographic effect; use of the <blockquote> tag for the sole purpose of indenting a paragraph, etc.

Note 1: Using <div> or <span> elements or multiple <br>to visually create a paragraph is considered non-compliant and invalidates the criterion. Example:

<div>paragraphs of a block of text simulated<br><br>using multiple <br></div> tags.

Note 2: WAI-ARIA provides a presentation role to remove the semantics of an element, for example <h1 role="presentation">Title</h1>. In this case, the text will be correctly rendered but its title role will no longer be rendered. The use of the presentation role may be required when using a WAI-ARIA design pattern.

The use of the WAI-ARIA presentation role can also be used to remove semantics from an element when it is used for presentation purposes only, e.g. <blockquote role="presentation"> will have the same effect as no <blockquote> element.

Even if this use is strongly discouraged (in the case of assistive technology that does not implement ARIA, for example) it can be considered WCAG compliant. On the other hand, the use of a WAI-ARIA role presented on an element whose nature (e.g. semantics) is essential to the understanding of the content is a violation of the WCAG rules (particularly failure F92) and invalidates the criterion.

URL

Address allowing access to World Wide Web resources: HTML document, Web pages, image, sound...

Note: in the RGAA, the notion of URL is used instead of URI (compact character string identifying a resource).

URI

Uniform Resource Identifier is a short string of characters that identifies a resource on a physical or abstract network, and whose syntax follows an Internet standard set up for the World Wide Web.

9.17 V

Accessible version (for a downloadable document)

Downloaded documents whose format types are recognized as compatible with accessibility must be made accessible either directly or through an accessible version or HTML version. The document formats that are recognized as compatible are :

- Microsoft Office (Word 2003, OOXML) ;
- Open Office Org (ODF) ;
- Adobe PDF ;
- Epub/Daisy.

The contents must comply with the criteria list Office Documents for Download (ODT format, 74 kilobytes).

Note: the txt format cannot be used to produce an accessible version of a document for download.

Alternative "audio only" version

An "audio only" version is an audio version, in the form of a simple MP3 file for example, used as an alternative to a video only (video without audio information). As the only users impacted by accessibility are blind people, who cannot see the video, WCAG considers it acceptable to offer an audio version as an alternative.

The "audio only" version should contain all the important visual information from the video.

Generally it is easier to produce an audio version than a text version when the video is very descriptive (textual transcription often requires significant editorial work). However, it should be remembered that only the textual transcription ensures universal access to the information provided by the video, in the event that a user is unable to launch the video, for example.

9.18 W

WAI-ARIA

WAI-ARIA (Web Accessibility Initiative - Accessible Rich Internet Applications)1 is a W3C technical specification.

The objective is to increase the accessibility of dynamic content and dynamic interface components developed using Ajax, HTML, JavaScript and related technologies.

9.19 Z

Area (of a reactive image)

Clickable or non-clickable area of a client-side responsive image or clickable area of a server-side responsive image.

Clickable area

A region of a responsive image to which an action has been associated; for example, triggering an event by clicking on a link (for a client-side clickable area: <area> tag with the href attribute). The <area> tags are contained within the <map> tag.

For server-side responsive images, the coordinates are held on the server.

Header area

The area at the top of the document, usually containing the title of the document, a logo, a slogan, etc.

Note: Be careful not to confuse this header area, which is unique in the site, with any content that can be tagged in HTML5 with the <header> element.

See the technical definition provided by WAI-ARIA: Banner (role).

Main content area

Area containing the main contents of the page, where the background information and functionalities are located (i.e. outside the menus, search or secondary areas of advertisements, related news, etc.).

Note: This area is unique within the page. It may be difficult to determine on some particular pages, such as the home page.

See the technical definition provided by WAI-ARIA: Main (role).

Footer area

This is information about the functioning of the site or legal information. It includes, for example, the legal notice, credits, terms of use, site map and possibly the accessibility page.

Note: Be careful not to confuse this footer area, which is unique in the site, with any content that can be tagged in HTML5 with the <footer> element.

See the technical definition provided by WAI-ARIA: Contentinfo (role).

Non-clickable area

A region of a responsive image that has no action associated with it. A client-side non-clickable area is contained within a <area> tag:

With the nohref attribute when the HTML code of the page is not HTML5 ;

No href attribute in HTML5.

The <area> tags are contained within the <map> tag.

10 References

This document has been prepared using several references and documentary sources. This chapter lists the references that were used.

General accessibility guidelines - GAAR Version 4.0 - reference as of 20 September 2019.

Web Content Accessibility Guidelines (WCAG) 2.1 - reference as of June 2018.

Understanding WCAG 2.1 - reference to April 2019 date.

Techniques for WCAG 2.1 - reference as of April 2019.

HTML 4.01 Specification.

HTML 5.2 - reference as of December 2017.

Accessible Rich Internet Applications (WAI-ARIA) 1.1 - reference as of December 2017.

WAI-ARIA Authoring Practices 1.1 - reference as of February 2019.

Using ARIA - reference as of September 2018.

ARIA in HTML - reference as of April 2019.

Accessible name and description calculation.

WCAG contrast calculation procedure.