3. Web accessibility principles. The WCAG

The WCAG is structured around four basic principles on what a website should be like to be accessible:

1. **Perceptible**: The information and components of the user interface must be presentable to users in a way that they can perceive them through one of their senses.
2. **Operable** – Users must be able to operate the interface, whatever device they use to do so (mouse, keyboard, etc.)
3. **Understandable** – Users must be able to understand the information and operation of the user interface.
4. **Robust** – Content must be robust enough so that it can be reliably interpreted by a wide variety of user agents, including assistive technologies.

If any of these principles are not met, disabled users will not be able to use your website. For each of the principles there are a series of guidelines and conformity criteria that help address them to improve the experience of people with disabilities.

The WCAG accessibility guidelines are structured as follows:

- Principles – First are the four principles that provide the foundation for web accessibility.
- Guidelines: Below the principles are the guidelines or guidelines. The twelve existing guidelines provide the basic goals towards which authors should aim to make content more accessible to users with different disabilities.
- Conformity criteria: For each guideline, verifiable conformity criteria are provided that allow WCAG to be used where compliance requirements and testing are necessary.
- Sufficient and recommended techniques: for each of the guidelines and each of the conformance criteria of the WCAG 2.0 document, a wide variety of techniques have also been documented. The techniques are informative and fall into two categories: those sufficient to satisfy the compliance criteria and those recommended.

3.1. Principle 1. Perceptible

To comply with the first principle of accessibility, it is necessary that both the information and the components of the user interface be presented in a way that users can perceive them.

- **Alternative text:** Text alternatives are provided to any non-text content, so it can be transformed into the formats users need, such as large text, braille, speech, symbols, or plain language.
- **Time-dependent multimedia:** Synchronized alternatives are provided for time-dependent multimedia contents; that is, multimedia only audio, only video, audio and video, or audio and/or video combined with interaction.
- **Adaptable:** Content can be presented in different ways (for example, with a simpler layout) without losing information or structure.
- **Distinguishable:** It is easier for users to see and hear the content, including the distinction between what is most important and what is least important.

Level A

1.1.1. Text alternatives are provided for non-text content.
1.2.1. An alternative to pre-recorded video-only or audio-only content is provided.
1.2.2. Subtitles are provided for videos with pre-recorded audio.
1.2.3. An audio description or verbatim transcription is provided for videos with sound.
1.3.1. The information is structured logically.
1.3.2. The content is presented in a meaningful order.
1.3.3. More than one sense is used to provide sensory instructions.
1.4.1. Information based on color alone is not presented.
1.4.2 Audio does not play automatically.

Level AA

1.2.4. Live videos have subtitles.
1.2.5. An audio description is provided for pre-recorded video content.
1.3.4. Content is not restricted to viewing in a single orientation (landscape or portrait), unless a specific orientation is essential.
1.3.5. The purpose of the form fields is identified, using the specific type of field (tel, email, password...) and autocomplete attributes.
1.4.3. The minimum contrast between text and background is at least 4.5:1
1.4.4. Text size can be increased by 200% without loss of content or functionality.

1.4.5. Text is not used in image format.
1.4.10. When content is presented at a width of 320 pixels, there is no loss of content or functionality and horizontal scrolling is avoided.
1.4.11. The contrast between non-textual elements (such as icons, graphic components, or focus indicators on a form) is at least 3:1.
1.4.12. No loss of content or functionality occurs when the user adapts the line height to 1.5 times the text size or the paragraph spacing to twice the text size, the word spacing to 0.16 times the text size and letter spacing at 0.12 times the text size.
1.4.13. When additional content is displayed by :hovering or :focus on an element, the new displayed content can be ignored with the "Esc" key without moving the keyboard pointer or focus, unless the new content fails in a form or does not interfere with any other content on the page. Additionally, the pointer can move to other content without the new content disappearing until it is moved from the point that displays the new content or is closed by the user or the new content is no longer relevant.

AAA level

1.2.6. Sign language translations are provided for pre-recorded videos.
1.2.7. An extended audio description is provided for pre-recorded videos.
1.2.8. Pre-recorded videos have a complete transcript, including visual cues (for example: The fisherman holds a large fish) as well as dialogue and sounds.
1.2.9. Alternatives to live audio are provided.
1.3.6. The purpose of the user interface components, icons, regions… is identified.
1.4.6. The contrast between text and background is at least 7:1.
1.4.7. The audio is guaranteed to be clear enough for listeners to hear you, without background audio or making it very quiet.
1.4.8. A wide range of presentation options is offered.
1.4.9. Images with text are never used.

3.2. Principle 2. Operable

To comply with the second principle of accessibility, it is necessary that the user can use the user interface and navigation components. This principle can be specified in the following guidelines:

1. **Accessible via keyboard**: all functionalities are available from the keyboard.
2. **Sufficient time** – Users are provided with sufficient time to read and use the content.

3. **Epileptic seizures:** elements that can cause epileptic seizures are avoided.
4. **Navigable** – Multiple paths are provided to help users navigate, find content, and determine where they are.

A level

2.1.1. The content and functionalities are accessible using only the keyboard.

2.1.2. Keyboard users are not trapped.

2.1.4. Keyboard shortcuts are provided.

2.2.1. The time limit to complete a task can be adjusted using user controls.

2.2.2. The user is offered controls to move the content: pause, stop, hide...

2.3.1. The content does not flash more than three times per second.

2.4.1. Links are provided to jump directly to the content.

2.4.2. The page title is clear and descriptive.

2.4.3. A logical order is maintained for the focus.

2.4.4. The purpose of a link is clear from its context.

2.5.1. Functions that are operated via gestures (for example, drag and drop) can also be operated with a pointer without gestures, unless a path- or point-based gesture is essential.

2.5.2. To prevent unintentional activation of controls, non-essential activation (for example, onmousedown) is prevented when clicking, tapping, or long-pressing the screen.

2.5.3. For UI components with labels that include text or images of text, the name contains the text that is presented visually.

2.5.4. Motion Action: Functionality that is activated by moving the device (for example, shaking it) or by user movement can be disabled and equivalent functionality is provided using standard controls such as buttons.

AA level

2.4.5. Multiple ways are offered to find the pages.

2.4.6. Clear headings and labels are used.

2.4.7. The keyboard focus is sufficiently visible and clear.

AAA level

2.1.3. Everything is accessible using only the keyboard, without exception.

2.2.3. There are no time limits.

2.2.4. Users are not interrupted.

2.2.5. User data is saved when you re-authenticate.

2.2.6. Timeouts: Users are warned of the length of downtime that could result in data loss, unless data is preserved for more than twenty hours if the user takes no action.

2.3.2. The content does not flash three or more times per second.
2.3.3. Interaction-triggered motion animation can be disabled unless the animation is essential to the functionality or information being conveyed.
2.4.8. Users are let know where they are.
2.4.9. The purpose of any link is clear from its own text, that is, taken out of context.
2.4.10. The content is sectioned by headings.
2.5.5. The size of clickable elements is at least 44x44 pixels, unless an alternative target of that size is provided, the target is inline (such as a link within a sentence), or the target size small is essential for its functionality.
2.5.6. Simultaneous input mechanisms: web content does not restrict the use of the input modalities available on a platform except when the restriction is essential, necessary to guarantee the security of the content or required to respect the user's settings.

3.3. Principle 3. Understanding

To comply with the third principle of accessibility, it is necessary that both the information and the way of using the user interface are understandable. This principle can be specified in the following guidelines:

1. Readable – The textual content is readable and understandable.
2. Predictable – Web pages display and function predictably.
3. Help with data entry: Users are helped to avoid errors and correct them.

Compliance with these guidelines can be verified through the following compliance criteria:

Level A
3.1.1. The language of the page has been defined.
3.2.1. The elements do not change when receiving focus.
3.2.2. Items do not change when data is entered.
3.3.1. Errors in data entry are clearly identified.
3.3.2. Form labels are provided and instructions for data entry are given.

AA level
3.1.2. Users are notified of the language change on the pages.
3.2.3. Navigation menus are consistent.
3.2.4. Icons and buttons are used consistently.

3.3.3. When users make mistakes, they are suggested how to correct them.

3.3.4. The risk of error in data entry is minimal, especially for sensitive data (financial, legal...)

AAA level

3.1.3. Unusual terms are explained.

3.1.4. Any abbreviations are explained.

3.1.5. Users with nine years of schooling can read and understand its content.

3.1.6. Words that are difficult to pronounce are explained.

3.2.5. The elements of the web page are not changed unless it is at the request of the user.

3.3.5. Detailed help and instructions are provided.

3.3.6. The risk of all kinds of errors in data entry is minimal.

3.4. Principle 4. Robust

According to the latest principle of accessibility, content must be robust enough so that it can be faithfully interpreted by a wide variety of user agents, including assistive devices.

Compatible – Maximizes compatibility with current and future user agents, including assistive devices.

A level

4.1.1. The HTML code validates correctly, it does not contain errors.

4.1.2. All elements have been built with accessibility in mind and all elements, even those provided by third parties, are properly validated.

AA level

4.1.3. State messages: In content implemented using markup languages, state messages can be determined programmatically using functions or properties so that they can be presented to the user through assistive technologies without receiving focus.

3.5. Web accessibility techniques

To be accessible, a web page must meet the four principles of accessibility: it must be operable, perceptible, understandable and robust.

3.5.1 Alternative texts for images

According to compliance criterion 1.1.1 Level A corresponding to the perceptibility principle, it is necessary to provide text alternatives for non-textual content. Avoid text in image format.

- Distinguish between merely ornamental images and images that provide some type of meaning. For example, we sometimes use a pencil icon to indicate a link to edit. If, in addition to the icon, we show the text “Edit”, including alternative text for this image would be redundant and ineffective for a user browsing with a screen reader. Thus, it is better to leave the alt attribute empty:

```
<a href="edit.php"><img src="ico-lápiz.png" alt="">
Editar</a>
```

- Use clear and descriptive alternative texts. For example, for a landscape photo, alternative text such as Pyrenean Mountains, with the snow-capped Aneto in the background, may be appropriate. Alternative texts that do not provide any meaning should be avoided, such as alt=“_DSC2345.jpg” or that begin with “Image of…”, since screen readers already tell the user that it is an image.
- Do not use images to represent text or images with embedded text. This is a common obsolete practice when the formatting and typographical variety possibilities that CSS allowed were very limited.

3.5.2 Header structure for web accessibility

Using headings (<h1>, <h2>, <h3>, <h4>, <h5>, <h6> tags) is much more than making a bold title, as they provide a solid structure to the web page. Think of headers as an outline of your web page.

To correctly structure headers keep in mind that:

- When a user without vision problems accesses the website, they decide, in a matter of seconds, whether the content seems relevant to them taking into account the keywords in the headings.

- The structure must be represented both visually and technically so that users can see the structure and screen readers can identify the structure to read it. The code structure must be aligned with the visual presentation.
- Headers need to be used to introduce content, as they are not statements or labels.

- The <h1> heading is the most important (both for accessibility and search engine positioning) and normally corresponds to the title of the page. It gives an indication of what the page is about, and should be unique for each page.
- Avoid skipping heading levels (for example, from H1 to H3) for accessibility reasons: people who use screen readers often rely on navigating through headings, so if the structure is not hierarchical it is possible who do not understand the structure of the content.

3.5.3 Accessible tables

The characteristics that a table must have to guarantee good accessibility are:

- Very often data tables have a short descriptive text before or after it indicating the contents of the table. This text is indicated by the <caption> tag and must be the first element after the opening <table> tag.
- It is important to clearly distinguish the header of the table with <thead>, and the body of the table with <tbody>
- Content cells are denoted by <td>, while header cells are denoted by <th>, and can be in either the <thead> or the <tbody>, for example, at the head of a row.
- In a simple table you can define the scope of a header cell through the scope attribute, which can take the values col(column header), row(row header) or, in headers that correspond to multiple columns or rows, colgroup, or rowgroup, respectively.
- In a more complex table, id attributes should be used to precisely identify each of the header cells and headers to name the series of headers that refer to a content cell.

3.5.4 Text scalability. The 'viewport' meta

According to level AA compliance criterion 1.4.4, corresponding to the basic principle of perceptibility, the text size must be able to be increased by up to 200% without loss of content or functionality. To meet this criterion, one of the issues to keep in mind is that using the <meta name="viewport">with the content="user-scalable=no" attribute disables the ability to zoom in browsers (especially mobile devices, forcing users to view text at the specified size). It is easy for many people with visual impairments, low vision, or limited

color perception to find text illegible when it is too small. That is why you have to avoid that attribute that prevents zooming.

See an example of an accessible setting for this meta tag: Set the initial zoom level to 1, but allow the user to zoom the text to a readable size:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

3.5.5 Accessible links

There are several compliance criteria related to ensuring efficient navigation for all users. The accessible links technique focuses on:

2.4.4. The purpose of a link is clear from its context (level A)
2.4.9. The purpose of any link is clear from its own text, i.e. taken out of context (level AAA)

To understand the reason for these criteria, we must be clear about how users who use screen readers navigate. These users have tools that allow them to display a list of all the links on a page. These tools help make your browsing more efficient. However, if the links use anchor text (so-called anchor text) such as “read more” or “click here”, the advantages provided by these tools are lost, since a user with vision problems has no clue (you don't have a visual context that gives you more information) that allows you to figure out where each link goes. That is why it is very important that each link has adequate anchor text that is sufficiently descriptive of the destination of the link. Additionally, providing clear anchor texts has an additional benefit for the positioning of your web pages.

You must avoid:

Links where the anchor text is not understandable out of context.

NO

```
<h3>Título de la noticia</h3>
<p>Haga clic <a href="productes.html">aquí</a> para conocer nuestro catálogo de productos.</p>
<p><a href="noticia.html">Leer más</a></p>
```

YES

```
<h3><a href="noticia.html">Título de la noticia</a></h3>
<p><a href="productes.html">Catálogo de productos</a></p>
```

Links where the link text is the URL. These types of anchor text are not easy to understand when read by a screen reader.

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

Empty links or links containing images that have empty alt text. If the link consists of an image, it must have alternative text that adequately describes the functionality of the link.

NO

```
<a href="https://www.facebook.com"></a>
<a href="https://www.facebook.com"><img src="img/facebook.svg"
alt=""></a>
```

YES

```
<a href="https://www.facebook.com"><span class="sr-only">Síguenos en
Facebook</span></a>
<a href="https://www.facebook.com"><img src="img/facebook.svg"
alt="Facebook"></a>
```

Links that open downloadable documents or that open in a new window. Opening a new window via the target="_blank" attribute is a widespread practice, especially when linking to PDF documents or a different website. This can be useful in some cases, but for users browsing with a screen reader this context change can cause a serious disorientation problem, especially since in the new window they will not retain their browsing history and will not be able to go back.

NO

```
<a href="fichero.pdf" target="_blank">Más información sobre la
preinscripción</a>
<a href="https://ioc.xtec.cat/educacio/" target="_blank">Web de la
XTEC</a>
```

If, despite this problem, we consider it necessary to open the page in a new window, it is important to warn the user through, for example, the title attribute.

SI

```
<a href="fichero.pdf" target="_blank" title="Descargue información sobre la preinscripción en PDF">Más información sobre la preinscripción</a>
<a href="https://ioc.xtec.cat/educacio/" target="_blank" title="Abre en ventana nueva la web de la XTEC">Web de la XTEC</a>
```

Links made through Javascript. This is also a very bad practice that affects both accessibility and positioning, because search engines cannot track all possible links through Javascript.

NEVER

```
<a href="javascript:document.location='index.html'">Enlace Javascript</a>
<span onclick="window.open('index.html')">Otro enlace Javascript</a>
```

Therefore, when developers create links, they should always use the <a> tag, since this tag has a series of accessibility features built in by default: focus can be made through the keyboard, the screen reader announces that it is a link and by default, hovering the mouse over the cursor turns it into a pointer instead of showing the arrow by default.

Although it is possible to emulate links with other elements, such as or , using Javascript listeners, you should emulate all these features to have an accessible emulated link, adding:

tabindex="0" so that the emulated link can receive keyboard focus.
role="link" so that assistive devices, such as screen readers, identify the element as a link.
style cursor:pointer so that mouse users recognize the element as a link.
Something like this and additionally the Javascript code that picks up the click event (or another event depending on the input device) and executes the link action:

```
<span role="link" tabindex="0" style="cursor:pointer;texto-decoration:underline;color:blue;">Esto es un enlace emulado</span>
<a href="link.html">Esto es un enlace de verdad</a>
```

3.5.6 Links to skip navigation

According to compliance criterion 2.4.1 level A, corresponding to the principle of operable accessibility, it is necessary to provide links to jump directly to content (skip to content).

The main content is usually not the first content displayed on a web page. This means that users who navigate with the keyboard or using a screen reader have to navigate through a long list of navigation links, link sublists, corporate icons, search engines and other elements before finally reaching the main content. This makes navigation particularly difficult for users with motor or visual disabilities.

There are several alternatives to create links that allow you to skip navigation. It involves providing a link at the top of the page that allows the user to download to an anchor located at the beginning of the main content. These links can always be visible or you can create "invisible" links that become visible when they receive focus from the keyboard.

```
<a href="#maincontent">Vaya al contenido principal</a>
<header>
  <nav>
    <ul>
      <li><a href="index.html">Inicio</a></li>
      <li><a href="productos.html">Productos</a></li>
      <li><a href="empresa.html">Empresa</a></li>
      <li><a href="contacta.html">Contacta</a></li>
    </ul>
  </nav>
</header>
<main id="maincontent">
  <h1>Título principal</h1>
  <p>Este es el primer párrafo del texto.</p>
...
</main>
```

3.5.7 Hiding elements in an accessible way

Sometimes we may be interested in hiding certain interface elements. If, to do so, we use the properties display:none;or visibility:hidden;we will be hiding these elements in a way that will make them inaccessible to users with some type of disability.

Some cases in which we may be interested in hiding elements visually are:

Elements that are only aimed at users who use a screen reader.
Elements that we want to show later, after some interaction, such as displaying an accordion-type component, distributing the content in several tabs...

```
.sr-only{
  posición: absoluta;
  width: 1px;
  height: 1px;
  padding: 0;
  margin: -1px;
  overflow: hidden;
  clip: recto(0, 0, 0, 0);
  white-space: nowrap;
  border-width: 0;
}
```

3.5.8 Writing accessible texts

Usually the task of making the text perceptible falls to the people responsible for the design and CSS coding, who must guarantee the following:

The contrast of the text with respect to the background is sufficient (4.5:1 to meet level AA and 7:1 for level AAA).
The font size can be enlarged by 200% without loss of content or functionality. In this case, we must especially ensure that when enlarging the font size through the tools provided by the browser, the text containing elements adapt to the content, preventing it from protruding from its natural space, either because it overlaps with others. elements

and causes legibility problems or obstacles to functionality or because it remains hidden. That's why you need to carefully use CSS properties, such as overflow:hidden o height. The user can adapt the text styles to their needs (in addition to colors, typography, line spacing, paragraph spacing, word spacing or letter spacing) without losing content or functionality.

The possibility of adapting styles to one's own needs and preferences is very important, not only for users with low vision but also for users who suffer from dyslexia, who can benefit from adapting typography, line spacing and letter spacing, for example. example. It is required to have advanced knowledge of browser tools to be able to configure your own styles.

- Avoid justified alignment of the text, since on the web it causes “rivers” between the words that make readability difficult.
- Serve larger fonts, between 18 and 24 points.
- Use sans serif fonts.
- Avoid entire fragments in italics, as they decrease the reading performance of people with dyslexia.
- Limit the width of the texts to about 44 characters per column.
- Apply a character spacing higher than average (between +7% and +14%).

For a text to be understandable, it is necessary to adhere to the following AAA level guidelines:

3.1.3. Unusual terms are explained.
3.1.4. Any abbreviations are explained.
3.1.5. Users with nine years of schooling can read and understand its content.

3.5.9 Page language and language changes

Level A success criterion 3.1.1 indicates that the language of the page needs to be set, while Level AA success criterion 3.1.2 specifies that users need to be notified of language changes within the page.

- Screen readers can identify the language of the document and therefore select the appropriate language to read the text to the user. Can you imagine the voice synthesizer reading a text written in Spanish in English?
- The necessary information is provided to some applications, such as automatic translators or spell checkers, to interpret the text according to the rules of the language in which it is written.

Meeting these compliance criteria is very easy. With the lang attribute applied to the root tag <html lang="es"> the general language of the HTML document is established. If a part of the page uses a word or a piece of text in another language, simply apply the langa attribute to the HTML tag that wraps the text.

```
<html lang="es">
  <head>
    <title>Título del documento</title>
    ...
  </head>
  <body>
    <main>
      <h1>¿Qué es un <span lang="en">brainstorming</span>?</h1>
      ...
    </main>
  </body>
</html>
```

3.5.10 Accessible forms

Several compliance criteria, both the perceptibility principle and the understanding principle, refer to data entry through forms.

1.3.5. The purpose of the form fields is identified, using the specific type of field (phone, email, password...) and autocomplete attributes.
3.3.1. Errors in data entry are clearly identified.
3.3.2. Form labels are provided and instructions for data entry are provided.
3.3.3. When users make mistakes, they are suggested how to correct them.

3.3.4. The risk of error in data entry is minimal, especially for sensitive data (financial, legal...)

- A set of related fields needs to be grouped together using the <fieldset> tag, which must have a legend field with the <legend> tag to describe the grouping.
- The ideal display shows at most 1/2 fields per line, as users with limited vision may have trouble scanning the screen horizontally.
- All fields (regardless of type) must have a related <label>tag. To optimally relate labels to their fields, it is necessary to match the value of the for attribute of the label with the value of the id attribute of the field.
- The button to submit the form must be type="submit" and have clear and actionable text. For example: "Send message".
- It is necessary to give all fields a precise type attribute. You can use type="email" and type="tel" respectively for emails and phone numbers. With this specific type, browsers will adapt the interface to improve the user experience. For example, a browser on mobile will display the numeric keypad when it encounters a field with type="tel".
- Clearly indicate required fields.
- Use the required attribute, which allows validation prior to sending without the need for Javascript. Patterns can even be included with regular expressions to validate more specific input formats.
- Do not use the placeholder attribute. This attribute raises several accessibility issues (including lack of text contrast by default, but can also confuse users). Never use it as a substitute for <label> tags.
- Provide all the information about what needs to be filled out through the <label> and not the placeholder.
- If captcha is used, it should be noted that although they are implemented to prevent spammers and robots from submitting forms, by doing so they also deny access to some human users with accessibility issues, who will not be able to complete the form: blind people or people with learning disorders such as dyslexia.
- checkboxManually validate whether radio can fill out and send the form using only the keyboard.

4. Evaluation of web accessibility. Practical case

There are tools that help with evaluation. However, no single tool can determine whether a website meets accessibility standards. Good knowledge is required to determine if a website is accessible and manual review is an important component to consider.

4.1. Automatic evaluation

Web accessibility assessment tools are online programs or services that help identify accessibility barriers. They save a lot of time and effort in the evaluation. However, tools cannot do everything: keep in mind that in some cases tools may provide inaccurate results and that some accessibility checks cannot be automated and require manual intervention.

4.1.1 Automatic evaluation

The WAVE Evaluation Tool extension for Chrome is a web accessibility evaluation tool developed by WebAIM.org. Provides visual feedback on the accessibility of web content by injecting icons and indicators into the page. No automated tool can tell if the page is accessible, but WAVE facilitates human evaluation and reports accessibility issues. All analyzes are performed entirely in the Chrome browser, allowing you to securely evaluate web pages, as well as those that are located on an intranet, in a local environment, or that are password protected.

4.1.2 Evaluation through Lighthouse

Lighthouse is an open source automated tool whose goal is to improve the quality of web pages. It can be run on any web page, whether it is public or requires authentication. It has audits of performance, accessibility, web positioning…

Lighthouse can be run from Chrome Developer Tools. After selecting the type of audit, it begins to run the analysis and generates a report with the indicators that have failed and tips to improve the page. Each indicator has to do with a reference document that explains why it is important and the techniques to solve the problem. At web.dev/lighthouse-

accessibility you can consult all the documents related to the accessibility indicators that Lighthouse evaluates.

4.1.3 Assessment through TAW

Another online tool for validating web accessibility is TAW. The operation is somewhat different than in the two previous tools, since it runs on the server and, therefore, does not allow the validation of the accessibility of pages in a local environment or protected with a password.

4.2. Manual evaluation

Many WCAG compliance criteria require manual validation. Therefore, for a website to be considered accessible, it is not only necessary to pass an automatic test, but a manual review is also necessary. There are some elements that can be easily validated manually and, to do so, we must look at various characteristics of each one.

4.2.1 Page Titles

Web page titles are displayed in search engine search results and read by screen readers, as well as displayed in browser tabs when multiple pages are open. To check that the title of a page is accessible, you must check that:

Indeed, there is a title (<title>) that adequately and briefly describes the content of the page.
The title of the page is different from the title of other pages on the same website and, therefore, allows the current page to be distinguished from the rest of the pages.

4.2.2 Alternative texts for images

Image alt texts indicate the purpose of an image, being very important for people who cannot see the image. The alt text must be functional and provide a user experience equivalent to that of the image (and should not necessarily have a descriptive function). For

example, an appropriate alt text for a search button with a picture of a magnifying glass is more appropriate “search” than “magnifying glass.”

Therefore, although automated tests can indicate whether the alt attribute, which is used to include alt text in images, is missing, determining whether or not the alt text is appropriate can only be done by viewing the image and evaluating the alt text itself. depending on its context.

4.2.3 Resizable text

Some people need to increase the size of the text to be able to read it, or change other properties. This change can be easily made with the tools provided by most browsers. However, if a page is not properly assembled, when the text size changes, unwanted effects can occur that affect accessibility. This can only be validated manually. To consider that a page meets this conformity criterion, it is necessary to verify that:

When you use the tools to increase the size of the text, the text, in effect, increases in size.
The text does not disappear or get cut off.
Text, images or other content do not overlap.
All buttons, form fields, and other controls remain visible and can be used correctly.
It is not necessary to scroll horizontally to read the text.

4.2.4 Access via keyboard and visible focus

To consider that a page meets this conformity criterion, it is necessary to verify that:

The keyboard focus indicator is visible.
Any site can be reached using the keyboard (usually by pressing the tab key to advance through links, form fields, media playback controls...)
You can leave any place using the keyboard without falling into any “trap”.
The tab command follows the read command.
Any functionality can be reproduced through the keyboard.
In drop-down lists (form selectors), you can use the cursor keys to move through the options without triggering any action.
If images have links, they clearly show focus when the link is activated and the link can be accessed via the keyboard (usually by pressing the Enter key).

4.2.5 Motion, flashes and flickering content

Therefore, it is necessary to manually verify the following points:

If there is content that automatically moves, blinks, or scrolls, the user has a way to stop or hide the movement.
No flashes or content flickers occur at a frequency of more than three times per second.

4.2.6 Alternatives for multimedia content (audio and video)

You must check that:

Audio or video playback controls can be accessed through the keyboard.
Optimally, the video or audio does not play automatically or, if it does, it may stop.
The videos have subtitles, and these are well synchronized with the spoken content.
If there are several people speaking, it is clearly identified when one is speaking and when the other is speaking.
Important sounds (and not just dialogue) are included in the subtitles.
Transcripts include all information provided by the audio, including speaker identification and relevant sounds.
The information provided visually has an alternative in audio description format.

4.3 Evaluation methodology according to the WAI

This type of evaluation is often performed in these cases:

Perform the final check before launching a product.
Provide information to potential buyers of the product.
Regularly monitor the accessibility of the website.
Before purchasing a product.
in implementing accessibility, for a list of accessibility issues.

To carry out an effective conformity assessment, it is necessary to have experience in:

Accessibility standards

Design and development of accessible websites

Technical aids

Knowledge about the use of the web by people with disabilities

The WCAG-EM, or Web Accessibility Conformity Assessment Methodology, provides a structure for the assessment process:

Define the scope of the evaluation.

Explore website resources.

Select a representative sample of website pages.

Evaluate the selected sample.

Report the results of the evaluation.