

# Right to Connect: digital inclusion for persons with intellectual disabilities

Grant Agreement No. 101049125 — RTCN — ERASMUS-EDU-2021-PCOOP-ENGO

## Creating Accessible Digital Platforms for Persons with Intellectual Disabilities: Methodological Guidelines for Developers and Designers

### Deliverable D2.3



**Co-funded by the  
European Union**

Disclaimer: Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Education and Culture Executive Agency (EACEA). Neither the European Union nor EACEA can be held responsible for them.

<b>Document Identification</b>			
Deliverable Title	Creating Accessible Digital Platforms for Persons with Intellectual Disabilities: Methodological Guidelines for Developers and Designers		
Deliverable ID	D2.3		
Author(s)	Aashish Kumar Verma		
Review history			
Version	Date	Reviewer(s)	Notes
Initial draft 1.0	05/07/2023	Aashish Kumar Verma	
Initial draft 2.0	11/07/2023	All partners	
Revised draft	13/03/2024	Aashish Kumar Verma and Klaus Miesenberger	
Final version	26/03/2024	Aashish Kumar Verma, Omor Ahmed Dhali	

Dissemination Level	
Public	X
Private	

## Contents

<b>Abstract</b> .....	4
<b>Introduction</b> .....	6
<b>Coverage of Guidelines:</b> .....	6
<b>Employing Existing Standards in Practice</b> .....	8
<b>Example for Developers</b> .....	8
<b>Example for Designers</b> .....	15
<b>Overview of existing guidelines, standards, methods, and tools</b> .....	20
<b>Web Accessibility Directive</b> .....	20
<b>EN 301 549 Standard</b> .....	21
<b>W3C and WAI</b> .....	23
<b>WCAG 2.1</b> .....	24
<b>ATAG 2.0</b> .....	27
<b>ARIA WAI-ARIA</b> .....	29
<b>Cognitive Accessibility at W3C</b> .....	31
<b>Making Content Usable for People with Cognitive and Learning Disabilities</b> .....	32
<b>User Agent Accessibility Guidelines</b> .....	36
<b>ETSI EG 203 350 V1.1.1 (2016-11): Human Factors; Guidelines for the design of mobile ICT devices and their related applications for people with cognitive disabilities</b> .....	38
<b>Inclusion Europe and Easy-to-read Standard</b> .....	40
<b>ISO 24495-1:2023 Plain language , Part 1: Governing principles and guidelines</b> .....	42
<b>ISO/IEC 29138-1:2018(en) Information technology — User interface accessibility — Part 1: User accessibility needs</b> .....	43
<b>Tools for Accessibility Testing</b> .....	44
<b>Keyboard Accessibility</b> .....	44
<b>Color Contrast Analyzer (CCA)</b> .....	46
<b>NVDA Screen Reader</b> .....	47
<b>Co-Design</b> .....	48
<b>Introduction to Co-Design</b> .....	48
<b>Co-Design 5 Phases</b> .....	48
<b>Empathy Maps</b> .....	49
<b>Good Practice: Co-Design Workshop of RTCN Project in Madrid Spain</b> .....	51
<b>Conclusion</b> .....	58

## Abstract

	Questions	Answers
1.	Why this work important	<p><b>Primary Users: Developers and Designers</b></p> <ul style="list-style-type: none"> <li>• These guidelines are rules to help people who make websites and apps create ones that work well for people with intellectual disabilities.</li> <li>• People with intellectual disabilities may need special features or tools to be able to use digital platforms.</li> <li>• These guidelines help web developers and designers understand what features are needed and how to make them.</li> </ul> <p><b>Secondary Users: Content Developers, Teachers, Policy Makers</b></p> <ul style="list-style-type: none"> <li>• Teachers and content developers can use these guidelines to make sure the materials they use in their classrooms and on their websites are accessible to everyone.</li> <li>• Policymakers can use these guidelines to make sure laws and regulations require that digital platforms are accessible for people with intellectual disabilities.</li> </ul>
2.	Hypothesis	Developers and designers can create digital platforms that are easy to use and accessible for people with intellectual disabilities with the help of these guidelines.
3.	Methods Used	<ul style="list-style-type: none"> <li>• <b>Primary Research: Online Questionnaires and Interviews with PWIDs:</b> <ul style="list-style-type: none"> <li>○ We asked people with intellectual disabilities questions.</li> <li>○ We conducted interviews to understand their needs.</li> </ul> </li> <li>• <b>Secondary Research: Comprehensive Literature Review:</b> <ul style="list-style-type: none"> <li>○ We asked people with intellectual disabilities questions online.</li> <li>○ We conducted interviews to understand their website needs.</li> </ul> </li> <li>• <b>Integrate Accessibility Guidelines in RTCN e-learning platform:</b> <ul style="list-style-type: none"> <li>○ Apply accessibility guidelines in every phase of development.</li> <li>○ Ensure adherence to accessibility standards throughout the process.</li> </ul> </li> <li>• <b>Co-Design with PWID</b> <ul style="list-style-type: none"> <li>○ Use the Spiral Model for software development.</li> <li>○ Apply accessibility guidelines in every phase of development.</li> <li>○ Ensure adherence to accessibility standards throughout the process.</li> </ul> </li> <li>• <b>Accessibility Testing</b> <ul style="list-style-type: none"> <li>○ Perform comprehensive accessibility testing using guidelines.</li> <li>○ Assess platform usability for persons with intellectual disabilities.</li> <li>○ Utilize testing feedback for continuous improvement.</li> </ul> </li> </ul>

	<b>Questions</b>	<b>Answers</b>
	<b>Results</b>	We incorporated accessibility guidelines into each phase of the software development life cycle of RTCN e-learning platform, using the Spiral Model. <ul style="list-style-type: none"><li>• This approach ensured that accessibility was considered from start to finish.</li><li>• As a result, the RTCN E-learning platform accessibility was significantly improved.</li><li>• Persons with Intellectual disabilities acknowledged the enhanced accessibility resulting from these guidelines.</li><li>• The Co-Design session of the RTCN e-learning platform contributed to the accessibility improvements.</li></ul>
	<b>Conclusion</b>	The use of these guidelines underscores their effectiveness in enhancing digital platform accessibility for persons with intellectual disabilities.

## Introduction

- Digital platforms are websites, apps, and other tools people use on computers and smartphones.
- Some people have disabilities that make it hard to use these digital platforms.
- We want to make sure that everyone can use these digital platforms, including persons with intellectual disabilities.
- Persons with intellectual disabilities have a hard time understanding things that are written or spoken or using technology like computers or phones.
- We want to make sure that digital platforms, which are websites, apps, or other computer programs, can be used by all people, including those with intellectual disabilities.
- We talked to some people with intellectual disabilities and asked them what they need to be able to use digital platforms.
- They told us that they need things to be simple and easy to understand, with pictures or videos that show what to do.
- They also need things to be organized and predictable, so they know what to expect and can follow along.
- Therefore, we have developed these guidelines to help developers and designers make digital platforms accessible for everyone with a focus on persons with intellectual disabilities.
- These guidelines tell us how to design and build digital platforms that are easy to use for persons with intellectual disabilities.
- It's important to follow these guidelines to make sure everyone can use digital platforms, even if they have a disability.
- Primary users of guidelines: These guidelines are like rules for developers and designers to help them make websites and apps that are easy for people with intellectual disabilities to use. These guidelines help web developers and designers understand what features are needed and how to make them.
- Secondary users of guidelines: Teachers and content creators can also use these guidelines to make sure their classroom materials and online stuff can be accessible to persons with intellectual disabilities. And policymakers can use them to make laws saying that digital platforms must be accessible for persons with intellectual disabilities, so everyone can use them fairly.

### Coverage of Guidelines:

1. The 'Right to Connect' project operates within European Union (EU) countries. This necessitates adherence to ETSI 301 549 accessibility guidelines, mandated by the EU. We

have also covered WCAG 2.1, UAAG, ARIA and ATAG 2.0 are internally referenced by ETSI 301 549.

2. We have prioritized guidelines relevant to persons with intellectual disabilities. Specifically, incorporated 'Making Content Usable for People with Cognitive and Learning Disabilities', integral to WCAG 2.1 guidelines.
3. We have covered ETSI EG 203 350 V1.1.1 (2016-11) guidelines. These guidelines are crucial for developing mobile applications accessible to persons with intellectual disabilities.
4. Inclusion Europe and Easy-to-Read guidelines: These guidelines specifically cater to the needs of persons with intellectual disabilities, ensuring our digital platforms are accessible and comprehensible to all users.
5. Utilization of ISO Standards: By adhering to these ISO standards, we uphold international best practices in accessibility, enhancing the usability and inclusivity of our digital platforms.
  - ISO 24495-1:2023 provides governing principles and guidelines for plain language, ensuring that our content is clear, concise, and easily understandable.
  - ISO/IEC 29138-1:2018 focuses on user interface accessibility, addressing the specific needs of users with disabilities.

## Employing Existing Standards in Practice

We've compiled essential examples for developers and designers to craft accessible digital platforms. It's important to note that while these examples address significant accessibility concerns, they may not cover every aspect. To ensure a digital platform is accessible, it must adhere to the ETSI 301 549 guidelines mandated in the European Union.

### Example for Developers

Now you will know the example for developers in which we have used HTML (Hypertext Markup Language), CSS (Cascading Style Sheets), ARIA (Accessible Rich Internet Applications) and JS (Javascript) as technology to provide accessibility.

#### 1. Language and Images

	Topic	Details
1.	Primary language and changes	<ul style="list-style-type: none"> <li>○ If the <b>language</b> is not explicitly mentioned on a <b>webpage</b>, screen readers may default to the user's language, causing difficulty for blind or visually impaired users. <ul style="list-style-type: none"> <li>○ It's crucial to specify the primary language accurately using the &lt;html&gt; element. Example: <b>&lt;html lang="en"&gt;</b></li> </ul> </li> <li>○ If there's text in <b>another language within a paragraph</b>, specify the language within a &lt;span&gt; element. <ul style="list-style-type: none"> <li>○ Example: <b>&lt;p&gt; She wishes "&lt;span lang="de"&gt; Guten Morgen"&lt;/p&gt;</b></li> </ul> </li> </ul>
2.	Acronym	<ul style="list-style-type: none"> <li>○ Acronyms can be hard to understand or ambiguous, so they should be fully explained when first mentioned. <ul style="list-style-type: none"> <li>○ <b>&lt;acronym title="work in progress"&gt;WIP&lt;/acronym&gt;</b></li> </ul> </li> </ul>
3.	Alternative text	<ul style="list-style-type: none"> <li>○ When <b>label text isn't visible</b> or is replaced by graphics, use alternative text (alt text) to describe the User Interface function or feature concisely. <ul style="list-style-type: none"> <li>○ Example: <b>alt="precise description"</b></li> </ul> </li> <li>○ <b>Image Descriptions:</b> Use longdesc, CSS hidden, or link for images needing detailed descriptions.</li> <li>○ Describe <b>SVG images</b> with title, alt, aria labeledby, or desc.</li> <li>○ <b>Font Icons:</b> Alt-text is also needed for CSS-generated font icons.</li> <li>○ For <b>background images</b>, use CSS or <b>alt=""</b>.</li> </ul>

#### 2. Color

	Principle	Details
1.	Color not only means of conveying information	<ul style="list-style-type: none"> <li>○ Avoid conveying information solely through color.</li> <li>○ For example, avoid using red text to signify failure; instead, provide additional context or use other visual cues.</li> <li>○ Ensure regular text and images of text have a contrast ratio of at least 4.5 to 1 against the background.</li> </ul>
2.	Contrast ratio	<ul style="list-style-type: none"> <li>○ Ensure regular text and images of text have a contrast ratio of at</li> </ul>



	for text and images	least 4.5 to 1 against the background.
--	---------------------	--

### 3. Zoom & Reflow

	Principle	Details
1.	Zoom & Reflow	<ul style="list-style-type: none"> <li>○ Resize content <b>checking up to 200%</b> to ensure readability and usability.</li> <li>○ Use relative units like em, ex, ch, and rem for font sizes to maintain scalability and consistency across devices.</li> </ul>
2.	Line Spacing	<ul style="list-style-type: none"> <li>○ Maintain line spacing between 1.4 to 1.7 for better readability and visual comfort.</li> </ul>
3.	Line Length	<ul style="list-style-type: none"> <li>○ Ensure each line contains 40 to 70 characters to prevent strain and enhance reading speed.</li> </ul>
4.	Avoid blocks and center	<ul style="list-style-type: none"> <li>○ Avoid using block and center styles for better flexibility and responsiveness in different screen sizes.</li> </ul>
5.	Browser zoom	<ul style="list-style-type: none"> <li>○ Do not suppress browser zoom to allow users control over the magnification of content.</li> <li>○ Use <b>&lt;meta name="viewport" content="width=device-width, initial-scale=1, user-scalable=1"&gt;</b> to ensure proper zoom functionality and prevent issues.</li> </ul>

### 4. Hidden Content

	Principle	Details
1	If you would like to hide content for everyone	<ul style="list-style-type: none"> <li>○ <b>display: none;</b> or visibility: hidden; or the hidden attribute.</li> </ul>
2	Hidden visually, but present for assistive tech like screen reader	<ul style="list-style-type: none"> <li>○ <b>.hideVisually {     position: absolute !important;     clip: rect(1px, 1px, 1px, 1px); }</b></li> <li>○ <b>&lt;div class="sr-only"&gt;This text is hidden.&lt;/div&gt;</b></li> </ul>
3	Hidden for assistive tech, but not visually?	<ul style="list-style-type: none"> <li>○ <b>The aria-hidden="true" attribute/value.</b></li> </ul>
4	General Practice	<ul style="list-style-type: none"> <li>○ Developers often use display: none or visibility: hidden to hide content, but this hides it from screen readers as well, which is not recommended.</li> </ul>

### 5. Video & Audio & Timing

	Principle	Details
1.	Providing Alternative Text	<ul style="list-style-type: none"> <li>○ Provide transcripts or captions for hearing accessibility.</li> <li>○ Offer audio descriptions for vision accessibility.</li> </ul>
2.	Accessible Player Controls	<ul style="list-style-type: none"> <li>○ Ensure player controls are accessible and don't create usability issues.</li> <li>○ Avoid autoplay features; if necessary, stop after 3 seconds</li> </ul>
3.	Use of Technology	<ul style="list-style-type: none"> <li>○ Prefer HTML5 over Flash for multimedia content.</li> <li>○ Utilize standards-compliant formats and codecs like</li> </ul>

		H.264/MPEG-4.
4.	Avoiding Time Limits	<ul style="list-style-type: none"> <li>○ Avoid imposing time limits whenever possible.</li> <li>○ Allow users to turn off time limits or extend them in increments of 20 seconds, up to 10 times, with a maximum of 20 hours.</li> <li>○ Use a timer that's keyboard focusable and provides visual style changes and alerts when deadlines approach or expire.</li> </ul>
5.	Informing Users of Dynamic Updates	<ul style="list-style-type: none"> <li>○ Utilize ARIA live regions for informing users of dynamic updates, such as chats, alerts, and progress updates.</li> <li>○ Specify ARIA live attributes like <code>aria-live="polite"</code></li> <li>○ Assign appropriate roles like <code>role="alert"</code>, <code>role="status"</code>, <code>role="timer"</code>, <code>role="marquee"</code>, or <code>role="log"</code> for dynamic content.</li> </ul>

## 6. Links and navigation

	Principle	Details
1.	Links and navigation	<ul style="list-style-type: none"> <li>○ Utilize the <code>&lt;a&gt;</code> element for links and ensure they have valid href values (avoid using <b>onclick</b>).</li> <li>○ Make link text descriptive of the destination instead of generic phrases like "click here".</li> <li>○ Structure navigation or menu items as a list for better organization and accessibility.</li> <li>○ Use <b>role="navigation"</b> and ARIA labels for multiple navigational elements to enhance accessibility.</li> <li>○ Maintain consistency in navigation across pages to provide a familiar user experience.</li> <li>○ Include a "bypass to main content" link at the beginning that receives focus to allow users to skip repetitive navigation and directly access the main content.</li> </ul>

## 7. Lists

	Principle	Details
1.	Use of List Elements	<ul style="list-style-type: none"> <li>○ Utilize <code>&lt;ul&gt;</code>, <code>&lt;ol&gt;</code>, and <code>&lt;dl&gt;</code> elements for creating lists on web pages.</li> </ul>
2.	Avoid	<ul style="list-style-type: none"> <li>○ Avoid using <code>&lt;br&gt;</code>, bullet characters, or other non-semantic techniques solely for visual formatting purposes within lists.</li> </ul>
3.	Examples	<ul style="list-style-type: none"> <li>○ For example, refrain from formatting lists like this:  <code>&lt;br&gt;</code> <code>&lt;h2&gt;Drinks&lt;/h2&gt;</code><code>&lt;br&gt;</code> Coffee<code>&lt;br&gt;</code> Tea<code>&lt;br&gt;</code> Milk<code>&lt;br&gt;</code> Instead, use semantic markup like this: <code>&lt;br&gt;</code> <code>&lt;h2&gt;Drinks&lt;/h2&gt;</code><code>&lt;br&gt;</code> <code>&lt;ul style="list-style-type:none"&gt;</code><code>&lt;br&gt;</code> <code>&lt;li&gt;Coffee&lt;/li&gt;</code><code>&lt;br&gt;</code></li> </ul>

## 8. Headings

	Principle	Details
1.	Importance of Headings	<ul style="list-style-type: none"> <li>○ Headings are essential for improving readability, aiding understanding, and facilitating easier navigation on web pages.</li> <li>○ Headings range from <code>&lt;h1&gt;</code> to <code>&lt;h6&gt;</code>, providing a hierarchical structure to content, with <code>&lt;h1&gt;</code> being the most important</li> </ul>

		and <h6> the least.
2.	Proper Sequence	<ul style="list-style-type: none"> <li>○ Ensure headings are used in a proper sequence and are nested appropriately to maintain logical structure and accessibility.</li> <li>○ Follow a hierarchical order where each subsequent heading level represents a subsection of the previous one.</li> </ul>
3.	Avoid Emphasizing	<ul style="list-style-type: none"> <li>○ Do not use headings solely for emphasizing text; instead, utilize semantic elements like &lt;strong&gt; tag for emphasis.</li> </ul>

## 9. Landmarks

	Principle	Details
1.	Importance of Landmarks	<ul style="list-style-type: none"> <li>○ Landmarks are used to define different sections of the overall page design and layout, while headings designate sections within the content.</li> <li>○ They provide a structured way to navigate through a webpage, improving accessibility and user experience.</li> </ul>
2.	Usage of Landmarks	<ul style="list-style-type: none"> <li>○ Utilize landmarks to mark predefined parts of the layout, such as &lt;header&gt;, &lt;nav&gt;, &lt;main&gt;, &lt;footer&gt;, etc.</li> <li>○ Ensure each landmark has an appropriate title and role assigned to it, providing clear identification and functionality.</li> </ul>
3.	HTML 5 vs. ARIA	<ul style="list-style-type: none"> <li>○ Prefer using HTML 5 landmarks over ARIA (Accessible Rich Internet Applications) equivalents whenever possible for better semantic markup.</li> </ul>
4.	Accessibility	<ul style="list-style-type: none"> <li>○ If a landmark is not useful or meaningful for screen readers, consider hiding it using <b>aria-hidden="true"</b> to prevent confusion and improve accessibility.</li> </ul>

## 10. Keyboard Accessibility

	Principle	Details
1.	Navigation To Widget	<ul style="list-style-type: none"> <li>○ Use Tab or Shift+Tab to navigate to the widget.</li> <li>○ Within the widget, use arrow keys for navigation.</li> </ul>
2.	Activation	<ul style="list-style-type: none"> <li>○ Activate the widget using the Spacebar or Enter key.</li> </ul>
3.	Closing Widget	<ul style="list-style-type: none"> <li>○ Close the widget using the Escape key or a designated button.</li> </ul>
4.	Focus Management	<ul style="list-style-type: none"> <li>○ Ensure focus can be set back to the appropriate place after interacting with the widget.</li> <li>○ Avoid trapping users in navigation loops where they are unable to navigate away from the widget.</li> </ul>
5.	Instructions for Complex Widgets	<ul style="list-style-type: none"> <li>○ Provide clear instructions for complex widgets to assist users in understanding how to interact with them effectively.</li> </ul>
6.	Focus and Hover Indicator	<ul style="list-style-type: none"> <li>○ Include visible focus or hover indicators with sufficient contrast to indicate which element is currently selected.</li> <li>○ Avoid removing focus <b>outlines (a:focus {outline: 0; /* or outline: none; */})</b> as it hinders accessibility for keyboard users.</li> </ul>
7.	Logical Tab Order	<ul style="list-style-type: none"> <li>○ Establish a logical tab order ensuring elements are</li> </ul>

		<p>reachable and operable by keyboard users.</p> <ul style="list-style-type: none"> <li>○ Use <b>tabindex="0"</b> to make an element tabbable and <b>-1</b> to make it focusable but not tabbable. Avoid using positive numbers for tabindex.</li> </ul>
8.	Custom Widgets and ARIA	<ul style="list-style-type: none"> <li>○ Use custom widgets when necessary, ensuring they are accessible and usable via keyboard navigation.</li> <li>○ Label everything clearly to provide context and clarity for users.</li> <li>○ Utilize ARIA (Accessible Rich Internet Applications) attributes when needed to enhance the accessibility of dynamic content and widgets.</li> </ul>

## 11. Tables

	Principle	Details
1.	Meaningful Title and Summary	<ul style="list-style-type: none"> <li>○ Provide a meaningful title for the table, and if the table is complex, include a summary to explain its content and purpose.</li> <li>○ The title and summary help users understand the context and content of the table, aiding accessibility and comprehension.</li> </ul>
2.	Meaningful Table Headers	<ul style="list-style-type: none"> <li>○ Use meaningful <b>&lt;th&gt;</b> (table header) elements for rows and columns to provide clear labels for data cells.</li> <li>○ Properly labeled headers improve accessibility and assistive technology interpretation of table data.</li> </ul>
3.	Avoid Complex or Nested Tables	<ul style="list-style-type: none"> <li>○ Avoid using complex or nested tables whenever possible as they can be difficult for users to navigate and understand, especially with assistive technologies.</li> <li>○ Simplify table structures to make them more accessible and user-friendly.</li> <li>○ Use tables for tabular data presentation, not for layout purposes; CSS should be used for layout design instead.</li> </ul>

## 12. Forms

	Principle	Details
1.	Labeling Inputs	<ul style="list-style-type: none"> <li>○ Ensure all form inputs have a corresponding label to establish a clear relationship between the text and input fields.</li> <li>○ The <b>'for'</b> attribute in the <b>'&lt;label&gt;'</b> should match the <b>'id'</b> attribute of the associated <b>'&lt;input&gt;'</b> element</li> </ul>
2.	Group Labels	<ul style="list-style-type: none"> <li>○ Group related fields under their own individual labels and provide a group label or question for clarity and organization.</li> <li>○ For example, multiple choice questions with radio buttons should have a group label and individual labels for each option.</li> </ul>
3.	Use of ARIA Labels	<ul style="list-style-type: none"> <li>○ Utilize ARIA labels (hidden) for complex forms to provide additional context or instructions using attributes like <b>'aria-</b></li> </ul>

		<p>describedby' or links.</p> <ul style="list-style-type: none"> <li>o ARIA labels help make complex forms more accessible to users with assistive technologies.</li> </ul>
4.	Indication of Mandatory Fields	<ul style="list-style-type: none"> <li>o Clearly indicate mandatory fields using the <b>'required'</b> attribute or by using an <b>asterisk (*)</b> next to the field label.</li> <li>o This helps users understand which fields are required to be filled out before submitting the form.</li> </ul>

### 13. Dynamic content & Interactivity

	Principle	Details
1.	Machine Readable	<ul style="list-style-type: none"> <li>o Establish a clear relationship between text and input elements, even if this relationship may not be obvious visually.</li> <li>o This ensures that content is machine-readable, aiding accessibility for users with assistive technologies.</li> </ul>
2.	Use of ARIA	<ul style="list-style-type: none"> <li>o Implement ARIA (Accessible Rich Internet Applications) roles and attributes to enhance accessibility where standard HTML falls short.</li> <li>o ARIA is typically used for complex interactions or when native HTML elements do not provide sufficient accessibility.</li> <li>o It allows developers to add or override HTML semantics to improve accessibility and communicate with assistive technology.</li> </ul>
3.	ARIA Roles	<ul style="list-style-type: none"> <li>o Utilize ARIA roles to define the purpose and behavior of elements such as buttons, dialogs, menus, and more.</li> <li>o ARIA roles are categorized into widget roles, document structure roles, and landmark roles, each serving specific purposes.</li> </ul>
4.	ARIA States and Properties	<ul style="list-style-type: none"> <li>o Apply ARIA states and properties to provide additional information about the state or properties of interactive elements.</li> <li>o These include attributes like aria-checked, aria-expanded, aria-haspopup, aria-label, and more.</li> <li>o ARIA states and properties aid in drag-and-drop functionality, live regions, and defining relationships between elements.</li> </ul>

### 14. Document Accessibility

	Principle	Details
1.	Apply Accessibility Rules	<ul style="list-style-type: none"> <li>o Apply above accessibility rules used in HTML to ensure document accessibility.</li> <li>o This includes providing alternative text for images, ensuring keyboard accessibility, using headings and tables, ensuring sufficient color contrast, providing meaningful titles, and choosing readable fonts.</li> </ul>
2.	Machine Readability	<ul style="list-style-type: none"> <li>o Ensure that content is machine-readable, establishing a clear relationship between text and input elements.</li> </ul>

		<ul style="list-style-type: none"><li>○ This aids accessibility for users with assistive technologies.</li></ul>
3.	Accessibility of Uploaded Content	<ul style="list-style-type: none"><li>○ Ensure all uploaded content, such as PDFs, Microsoft Word documents, and Excel files, is accessible.</li><li>○ Accessibility measures should be applied to all document formats to ensure inclusivity and usability for all users.</li></ul>

## Example for Designers

- This checklist serves as a starting point for creating accessible designs but should not be considered a final checklist.
- It highlights some of the most important accessibility concepts.
- Designers are encouraged to push themselves beyond the minimum requirements.
- Aim to create designs that are not only accessible but also enjoyable and well-suited to the needs of people with disabilities.

### 1. Title

	Aspect	Details
1.	Title	<ul style="list-style-type: none"> <li>○ Check if the page title effectively describes the topic or purpose of the page.</li> <li>○ Ensure that the page title provides users with clear information about the content they will find.</li> </ul>

### 2. Headings

	Aspect	Details
1.	Consistent Use of Headings	<ul style="list-style-type: none"> <li>○ Ensure all text that appears and behaves like a heading is marked up as a heading element.</li> <li>○ Confirm that heading levels are chosen based on their hierarchical significance in the content.</li> </ul>
2.	Proper Hierarchical Order	<ul style="list-style-type: none"> <li>○ Verify that heading levels are selected to accurately convey the hierarchical structure of the content.</li> <li>○ Avoid selecting heading levels solely for visual styling purposes.</li> </ul>

### 3. Navigation

	Navigation	Details
1.	Skip Navigation	<ul style="list-style-type: none"> <li>○ Ensure there is a method provided to skip repetitive navigation and go directly to the main content.</li> <li>○ Two main techniques for this include providing HTML/ARIA landmarks (e.g., header, navigation, main, footer) and "skip navigation" links.</li> </ul>
2.	Logical Reding Order	<ul style="list-style-type: none"> <li>○ Confirm that the content's reading and focus order, determined by the code order, is logical and intuitive.</li> </ul>

### 4. Links

	Links	Details
1.	Descriptive Link Text	<ul style="list-style-type: none"> <li>○ Ensure that the text used for links clearly describes the purpose or destination of the link.</li> </ul>

### 5. Color Contrast

	Color/Contrast	Details
1.	Information	<ul style="list-style-type: none"> <li>○ Ensure information is conveyed using methods other than color</li> </ul>

	Conveyance	alone, such as using both color and text to indicate errors in form fields or label chart categories.
2.	Minimum Color Contrast	○ Verify that all text has a minimum color contrast against its background of at least 4.5 to 1 (or 3 to 1 for large text).
3.	Distinguishable Link Text	○ Ensure link text is distinguishable from non-link text by more than just color.
4.	Clear Focus Indication	○ Actionable elements should have a clear, visible focus indicator when users navigate to them using the Tab or Arrow keys.
5.	Sufficient Color Contrast for Non-text Elements	○ Non-text elements such as user interface components and graphical objects must have sufficient color contrast of at least 3 to 1 for accessibility.

## 6. Magnification and Responsive Design

	Magnification and Responsive Design	Details
1.	Zoom Accessibility	<ul style="list-style-type: none"> <li>○ Ensure users with low vision can magnify or zoom in on content across all devices.</li> <li>○ Optimize the design for all zoom states to simplify the layout and eliminate horizontal scrolling.</li> </ul>
2.	Responsive Design	<ul style="list-style-type: none"> <li>○ Ensure the website is usable in both landscape and portrait orientations on various devices.</li> <li>○ Design the website to be responsive, adapting seamlessly to different screen sizes and orientations.</li> </ul>
3.	Tools	<ul style="list-style-type: none"> <li>○ <a href="#">Viewport Size by Device / Phone Screen Dimensions   Viewport Sizer Tool</a></li> <li>○ <a href="#">The Viewport Emulator</a></li> </ul>
4.	Mobile Emulator	<ul style="list-style-type: none"> <li>○ Firefox's Mobile Emulator: Command + Option + M (Mac) or Control + Shift + M (Windows)</li> <li>○ Chrome's Mobile Emulator: Command + Option + I (Mac) or Ctrl + Shift + I (Windows)</li> </ul>

## 7. Images

	Images	Description
1.	Alternative Text for Informative Images	<ul style="list-style-type: none"> <li>○ Ensure alternative text for informative images provides the same information as the image.</li> <li>○ Alternative text should convey the purpose or content of the image to users who cannot see it.</li> </ul>
2.	Alternative Text for Actionable Images	<ul style="list-style-type: none"> <li>○ Alternative text for actionable images (e.g., image links, buttons) should clearly identify the link destination or button purpose.</li> <li>○ Users should understand the action associated with the image based on its alternative text.</li> </ul>
3.	Explanation of Complex Images and	<ul style="list-style-type: none"> <li>○ Complex images or infographics should be fully explained in the page content with a short alternative text description.</li> <li>○ Ensure users can comprehend the content of complex images even if</li> </ul>



	Infographics	they cannot view them.
4.	Identification of Decorative Images	<ul style="list-style-type: none"> <li>○ Decorative images should be identified as not requiring alternative text.</li> <li>○ Alternative text for decorative images is unnecessary as they do not convey meaningful content</li> </ul>
5.	Use of Plain Text instead of Text in Images	<ul style="list-style-type: none"> <li>○ Whenever possible, use plain text instead of embedding text within images.</li> <li>○ Exceptions exist for text in logos and decorative text images, but plain text is preferable for accessibility.</li> </ul>

## 8. Table

	Tables	Details
1.	Data Table Accessibility	<ul style="list-style-type: none"> <li>○ Ensure data tables have a caption (name/title) to describe the purpose or content of the table.</li> <li>○ Properly identify columns and/or rows in the markup to aid users in understanding the table's structure.</li> </ul>
2.	Simplification of Complex Tables	<ul style="list-style-type: none"> <li>○ Simplify complex tables to minimize or eliminate the need for compound column or row headers.</li> <li>○ Complex tables should be made as straightforward as possible to enhance accessibility and usability.</li> </ul>

## 9. Forms

	Forms	Details
1.	Visibility and Descriptiveness of Form Labels	<ul style="list-style-type: none"> <li>○ Ensure all form fields have a visible label that clearly describes and instructs users on what to input.</li> <li>○ Form labels should provide all necessary information for users to complete the form accurately.</li> </ul>
2.	Proximity of Labels and Instructions	<ul style="list-style-type: none"> <li>○ Position form labels and instructions adjacent to their respective form elements for easy association.</li> <li>○ This helps users, including those who use screen magnification, to connect form elements with their labels.</li> </ul>
3.	Controls' Proximity to Content	<ul style="list-style-type: none"> <li>○ Place controls (e.g., Edit and Delete buttons) close to the content they modify for intuitive usage.</li> </ul>
4.	Error Message Clarity	<ul style="list-style-type: none"> <li>○ Ensure error messages provide sufficient information for users to understand and correct their errors.</li> <li>○ Clear error messages facilitate an efficient correction process for users.</li> </ul>
5.	Compatibility with Autofill	<ul style="list-style-type: none"> <li>○ Write forms so they are compatible with Autofill features, enhancing user convenience and efficiency.</li> </ul>

## 10. Dynamic Content

	Dynamic Content	Details
1.	Awareness of Dynamically Inserted Content	<ul style="list-style-type: none"> <li>○ Ensure users are made aware of dynamically inserted content on a page.</li> <li>○ Newly inserted content should follow the logical reading order/tab order of the page for user awareness.</li> </ul>

2.	Consistent Interactions	<ul style="list-style-type: none"> <li>○ Ensure all keyboard-only and touch screen interactions follow expected patterns for user familiarity.</li> <li>○ Users should know how to interact with all widgets on the page without confusion or uncertainty.</li> </ul>
3.	Feedback for Interactions	<ul style="list-style-type: none"> <li>○ Design interactions to provide success and failure feedback for all users.</li> <li>○ Feedback should include success/error messages, obvious activation of features, etc., accessible to all users.</li> </ul>

## 11. Custom Widgets

	Custom Widgets	Details
1.	Use of Standard HTML Widgets	<ul style="list-style-type: none"> <li>○ Utilize standard HTML widgets (links, buttons, form elements) whenever possible as they have built-in accessibility features.</li> <li>○ Native widgets inherently support accessibility, ensuring usability for all users.</li> </ul>
2.	Keyboard Support and WAI-ARIA Compliance	<ul style="list-style-type: none"> <li>○ If custom widgets are used, ensure they provide full keyboard support and adhere to WAI-ARIA authoring practices.</li> <li>○ Custom widgets should be accessible to users who rely on keyboard navigation and assistive technologies.</li> </ul>

## 12. Touch Devices

	Touch Devices	Details
1.	Touch Target Size and Spacing	<ul style="list-style-type: none"> <li>○ Ensure the touch target size of main links and buttons is large enough and spaced far apart for easy activation with a finger.</li> <li>○ Users should be able to interact with links and buttons comfortably without accidental activations</li> </ul>
2.	Alternative Activation for Custom Gestures	<ul style="list-style-type: none"> <li>○ Provide an alternative method to activate custom swipe actions or gestures.</li> <li>○ Note that when a screen reader is active on a touch device, it overrides all custom swipe actions and gestures.</li> </ul>

## Important Links for Developers and Designers

- **WAI guidelines for Developers:** [Developing for Web Accessibility – Tips for Getting Started | Web Accessibility Initiative \(WAI\) | W3C](#)
- **WAI guidelines for Designers:** [Designing for Web Accessibility – Tips for Getting Started | Web Accessibility Initiative \(WAI\) | W3C](#)
- [Introduction to Web Accessibility | Web Accessibility Initiative \(WAI\) | W3C](#)
- [Accessibility Principles | Web Accessibility Initiative \(WAI\) | W3C](#)
- [How People with Disabilities Use the Web | Web Accessibility Initiative \(WAI\) | W3C](#)
- [Tutorials | Web Accessibility Initiative \(WAI\) | W3C](#)
- [Before and After Demonstration: Overview \(w3.org\)](#)

- [How to Meet WCAG \(Quickref Reference\) \(w3.org\)](#)
- [Web Accessibility Evaluation Tools List | Web Accessibility Initiative \(WAI\) | W3C](#)
- [WAI-ARIA Overview | Web Accessibility Initiative \(WAI\) | W3C](#)

## Overview of existing guidelines, standards, methods, and tools

### Web Accessibility Directive

The Web Accessibility Directive (WAD) is an European Union (EU) law that focuses on making websites and mobile applications accessible to everyone. Here are the key points:

#### 1. Purpose of WAD:

- **Accessibility:** Ensures that digital services are usable by all individuals, including those with disabilities.
- **Inclusion:** Supports the European Commission's commitment to creating a '**Union of equality**'.

#### 2. What It Requires:

- **Public Sector Bodies:** All public sector websites and mobile apps within the EU must be made **accessible**.
- **Common Understanding:** To define what "accessible" means, the WAD is supported by a **harmonized technical standard**.

#### 3. Technical Standard:

- The **EN 301 549 standard** provides a **legal 'presumption of conformity'**.
- **Directive 2016/2102** refers directly to the European standard EN 301 549.
- If a website or app meets all applicable technical requirements in this standard, it's assumed to be **accessible** under the WAD.
- Member States can impose stricter requirements, but following this standard ensures conformity with the WAD.

#### 4. Important Links:

- [Web Accessibility Directive — Standards and harmonization | Shaping Europe's digital future \(europa.eu\)](#)
- EU Law applies to Public sector: [EU Directive 2016/2102](#)
- EU Law applies to Private sector: [EU Directive 2019/882](#)
- European Accessibility Act (EAA): [European Accessibility Act](#)
- [Blog post: EU Web Accessibility Compliance and Legislation](#) 

## EN 301 549 Standard

### 1. What is EN 301 549?

- It's a **harmonized European standard** that focuses on **accessibility requirements for ICT products and services**.
- Specifically, it deals with making **websites and mobile applications** accessible.

### 2. Why Is It Important?

- **Legal Compliance:** EN 301 549 aligns with the **Web Accessibility Directive (WAD)**, which mandates that **public sector websites and apps** be accessible.
- **Inclusivity:** Ensures that digital content can be used by **everyone**, regardless of abilities or disabilities.
- **WCAG Connection:** EN 301 549 draws heavily from the **Web Content Accessibility Guidelines (WCAG)**, which are globally recognized standards for web content.

### 3. Key Points:

- **Presumption of Conformity:** If your website or app meets all applicable technical requirements in EN 301 549, it's assumed to be **accessible** under the WAD.
- **European Standard:** EN 301 549 is a **harmonized norm** with legal force across the EU.
- **Mapping to WCAG:** It maps relevant provisions from the standard to the **accessibility requirements** in the WAD.

### 4. Version Details:

- The latest version is **EN 301 549 V3.2.1 (2021-03)**.
- It's in line with **WCAG 2.1**.
- Future updates may happen, but this version is the authoritative reference for accessibility.

### 5. Areas Covered:

- **Functional Performance (Chapter 4):** This chapter describes the needs of persons with disabilities, written as functional performance statements. It explains the functionality needed to enable users with different abilities to locate, identify, and operate functions in technology.
- **Generic Requirements (Chapter 5):** These are the basic requirements that all ICT products and services should meet to be accessible to persons with disabilities.
- **ICT with Two-Way Voice Communication (Chapter 6):** This section covers the accessibility requirements for ICT products and services that involve two-way voice communication.
- **ICT with Video Capabilities (Chapter 7):** This section outlines the accessibility requirements for ICT products and services that have video capabilities.

- **Hardware (Chapter 8):** This chapter discusses the accessibility requirements for hardware products like smartphones, personal computers, information kiosks, etc.
- **Web (Chapter 9):** This section covers the accessibility requirements for web-based software like web pages and mobile applications.
- **Non-Web Documents (Chapter 10):** This section outlines the accessibility requirements for non-web documents.
- **Software (Chapter 11):** This chapter discusses the accessibility requirements for software products.
- **Documentation and Services (Chapter 12):** This section covers the accessibility requirements for documentation and services related to ICT products and services.
- **ICT Providing Relay or Emergency Service Access (Chapter 13):** This chapter outlines the accessibility requirements for ICT products and services that provide relay or emergency service access.

#### 6. Important Links:

- [ETSI EN 301 549 - V3.2.1 - Accessibility requirements for ICT products and services](#)

## W3C and WAI

### 1. What is W3C and WAI?

- The **World Wide Web Consortium (W3C)** develops international standards for the Web: HTML, CSS, and many more.
- The W3C **Web Accessibility Initiative (WAI)** develops standards and support materials to help you understand and implement accessibility.
- You can use W3C WAI resources to make your websites, applications, and other digital creations more accessible and usable to everyone.

### 2. W3C and WAI Standards

The World Wide Web Consortium (W3C) Web Accessibility Initiative (WAI) makes rules to help different parts of the web be accessible:

- **Authoring Tool Accessibility Guidelines (ATAG)** Overview helps people who make authoring tools.
- **Web Content Accessibility Guidelines (WCAG 2)** Overview helps with web content. It's used by developers, authoring tools, and tools that check accessibility.
- **User Agent Accessibility Guidelines (UAAG)** helps with web browsers and media players, including some parts of assistive technologies.
- These rules are based on the basic technical specifications of the web and are made in partnership with all W3C technical specifications like HTML, CSS, SVG, SMIL, etc. W3C also makes rules specifically for accessibility, such as ARIA.
- **Accessible Rich Internet Applications (ARIA)** helps to make web applications easier to use for people with disabilities. It's especially useful for things like dynamic content and advanced user interface controls on websites made with Ajax, HTML, JavaScript, and similar tools.

### 3. Important Links:

- [Authoring Tool Accessibility Guidelines \(ATAG\) Overview](#)
- [WCAG 2 Overview](#)
- [User Agent Accessibility Guidelines \(UAAG\)](#)
- [W3C technical specifications](#)
- [ARIA, the Accessible Rich Internet Applications](#)

## WCAG 2.1

### 1. What is WCAG 2.1?

- The **Web Content Accessibility Guidelines (WCAG) 2.1**, published by the World Wide Web Consortium (W3C) in 2018, are considered the leading global standards for **web accessibility**.
- These guidelines are designed to be technology-neutral, meaning they apply to various web technologies like HTML, PDF, Java, Flash, Silverlight, video players, etc.

### 2. What is Success Criterion WCAG 2.1?

- Success Criteria are categorized into three levels: **Level A**, **Level AA**, and **Level AAA**.
- **Level A** signifies the **minimum level** of compliance, while **Level AAA** represents the **highest level**.
- Many organizations adopt **Level AA** as their standard because it's **achievable and meaningful** without overwhelming developers.
- Despite Level AA being commonly targeted, it's crucial to recognize that all Success Criteria, including those at **Level AAA**, are essential for some people with disabilities.
- **Level AA** is often chosen as the **baseline accessibility target** by organizations, either due to legal obligations or because it's viewed as a reasonable balance between accessibility and implementation effort.

### 3. The 4 Main Principles of Accessibility in WCAG are:

- **Perceivable:**
  - Ensure information and interface elements are presented in ways that users can perceive.
  - Aimed at making content accessible to people who are blind and/or deaf and/or other disabilities.
- **Operable:**
  - Make user interface components and navigation operable.
  - All features should be accessible via keyboard, not just mouse.
- **Understandable:**
  - Ensure information and user interface operations are understandable.
- **Robust:**
  - Content must be robust enough to be reliably interpreted by various user agents, including assistive technologies.
  - These principles are summarized using the acronym POUR. While the acronym itself may not mean much, it serves as a mnemonic to remember the four main accessibility principles.



4. There are **13 guidelines** grouped under the **four main principles**:

- **Perceivable:**
  1. **Provide text alternatives:** Ensure non-text content can be changed into other forms like large print or braille.
  2. **Offer alternatives for time-based media:** Provide options for users to access content like audio or video.
  3. **Create adaptable content:** Ensure content can be presented in different ways without losing its meaning.
  4. **Enhance content visibility:** Improve the ability for users to see and hear content by separating foreground from background.
- **Operable:**
  5. **Keyboard accessibility:** Make all functions accessible via keyboard.
  6. **Provide sufficient time:** Allow users enough time to read and interact with content.
  7. **Avoid seizure triggers:** Design content to avoid causing seizures.
  8. **Facilitate navigation:** Help users navigate and find content easily.
  9. **Support multiple inputs:** Enable users to interact with functionality using various input methods beyond the keyboard.
- **Understandable:**
  - **Readable text:** Ensure text content is readable and understandable.
  - **Consistent presentation:** Make web pages appear and behave predictably.
  - **Error prevention and correction:** Assist users in avoiding and rectifying mistakes.
- **Robust:**
  13. **Maximize compatibility:** Ensure compatibility with current and future user agents and assistive technologies.


5. **"Sufficient Techniques" and "Advisory Techniques"**

- In WCAG, there are two levels of techniques to help achieve success criteria: "Sufficient Techniques" and "Advisory Techniques."
- "Sufficient Techniques" are essential methods; meeting them ensures success with the criterion.
- "Advisory Techniques" are additional suggestions, not critical for accessibility but good practices.
- Techniques cover various technologies like HTML, CSS, SMIL, scripting, Flash, PDF, Silverlight, and ARIA.
- Choose techniques based on the technologies you're using.

## 6. Cognitive Accessibility in WCAG Standard

- This WCAG standard includes cognitive accessibility requirements as mentioned by W3C document on cognitive accessibility: [Cognitive Accessibility at W3C | Web Accessibility Initiative \(WAI\) | W3C](#).
- [Guideline 1.3 Adaptable](#): Design content that can be displayed in various ways (like a simpler layout) without losing any information or structure.
- [Guideline 1.4 - Distinguishable](#): Make it simpler for users to see and hear content, including separating the main content from the background.
- [Guideline 2.2 - Enough Time](#): Ensure users have sufficient time to read and interact with the content.
- [Guideline 2.4 - Navigable](#): Offer methods to assist users in navigating, locating content, and understanding their current location.
- [Guideline 3.1 - Readable](#): Ensure that text content is easy to read and comprehend.
- [Guideline 3.2 - Predictable](#): Ensure that web pages function and appear in consistent ways.
- [Guideline 3.3 - Input Assistance](#): Provide help to users to prevent and rectify errors.

## 7. Important Links:

- [WCAG 2.1 Specification](#) 
- [What's New in WCAG 2.1](#) 
- [WCAG 2.1 Quick Reference](#) 

## ATAG 2.0

### 1. Introduction

- The Authoring Tool Accessibility Guidelines (ATAG) 2.0 by W3C provide principles for web authoring tools to ensure accessible web content creation.
- These guidelines emphasize tasks like prompting for alt text for images and adding labels to forms.
- **ATAG 2.0 focuses on:**
  - Part A: Making the authoring tool user interface accessible.**
  - Part B: Supporting the production of accessible content.**
- ATAG 2.0 applies to many websites and web applications that function as authoring tools.
- Often, web developers are unaware of ATAG, but any part of a web app enabling content creation must adhere to these principles.
- Web-based authoring tools like Google Drive and WYSIWYG editors in CMS platforms such as Drupal and Joomla should comply with ATAG.

### 2. Part A: Make the authoring tool user interface accessible





- The authoring tool should be usable by people with disabilities, following the Web Content Accessibility Guidelines 2.0.
- Specific clarifications for authoring tools include:
  - **A.1. Authoring tool user interfaces follow applicable accessibility guidelines**
    - Ensure both web-based and non-web-based functionality is accessible.
  - **A.2. Editing-views are perceivable**
    - Provide alternative content.
    - Ensure editing-view presentation can be determined programmatically.
  - **A.3. Editing-views are operable**
    - Offer keyboard access to authoring features.
    - Provide sufficient time for authors.
    - Avoid flashing that could cause seizures.
    - Enhance navigation and editing through content structure.
    - Enable text search of the content.
    - Manage preference settings.
    - Ensure previews are as accessible as in-market user agents.
  - **A.4. Editing-views are understandable**

- Help authors avoid and correct mistakes.
- Document the user interface, including accessibility features.

### 3. Part B: Support the production of accessible content

- Authoring tools should facilitate the creation of accessible content effortlessly.
- The tool's standard output should be accessible, and it should prompt users to fix accessibility issues, offering guidance on how to do so.
- **B.1. Fully automatic processes produce accessible content**
  - Ensure automatically-generated content is accessible.
  - Preserve accessibility information.
- **B.2. Authors are supported in producing accessible content**
  - Enable the production of accessible content.
  - Guide authors in creating accessible content.
  - Assist authors in managing alternative content for non-text elements.
  - Provide accessible templates for authors.
  - Offer accessible pre-authored content.
- **B.3. Authors are supported in improving the accessibility of existing content**
  - Aid authors in identifying accessibility issues.
  - Assist authors in fixing accessibility problems.
- **B.4. Authoring tools promote and integrate their accessibility features**
  - Ensure features supporting accessible content production are available.
  - Document how to produce accessible content effectively.

### 4. Important Links:

- [ATAG 2.0 Overview](#) 
- [ATAG 2.0 Specification](#) 
- [Implementing ATAG 2.0](#) 
- [Essential Components of Web Accessibility](#)  (how WCAG and ATAG are inter-related)

## ARIA WAI-ARIA

### 1. WAI-ARIA: Accessible Rich Internet Applications Suite

- WAI-ARIA aims to enhance web content and applications' accessibility for individuals with disabilities.
- It particularly benefits dynamic content and advanced user interface controls created with HTML, JavaScript, and similar technologies.
- Without WAI-ARIA, certain web functionalities are inaccessible to some disabled users, especially those reliant on screen readers or unable to use a mouse.
- WAI-ARIA tackles these accessibility hurdles by defining methods for assistive technology to access functionality.
- Developers can utilize WAI-ARIA to ensure advanced web applications are accessible and functional for people with disabilities.

### 2. Technical Solutions with WAI-ARIA

- WAI-ARIA offers a framework for adding attributes that identify user interaction features, their relationships, and current states.
- It describes navigation techniques to label web regions and common structures like menus, primary and secondary content, banners, etc.
- For instance, WAI-ARIA allows developers to mark page regions, enabling keyboard users to navigate among them easily, avoiding excessive tab pressing.
- WAI-ARIA also provides technologies to link controls, live regions, and events with accessibility APIs, including custom controls used in rich internet applications.
- Its techniques are applicable to various widgets like buttons, drop-down lists, calendars, expandable menus, and more.
- WAI-ARIA offers web authors:
  - Roles to define widget types (e.g., "menu", "treeitem", "slider", "progressbar").
  - Roles to describe web page structures such as headings and regions.
  - Properties to indicate widget states (e.g., "checked" for checkboxes, "readonly" for form controls).
  - Properties to define live regions likely to receive updates (e.g., stock quotes).
  - A method for keyboard navigation through web objects and events.

### 3. WAI-ARIA 1.2 Version Overview:

- Extends WAI-ARIA 1.1 to enhance HTML + ARIA accessibility model with a few additional features.
- Provides guidance for user agent implementation through a suite of accessibility API mapping specifications.

### 4. Important Links: Published WAI-ARIA Specifications:

- **WAI-ARIA 1.2 Technical Specification:**
  - Offers features to define accessible user interface elements, enhancing web content and application accessibility and interoperability.
  - Target audience includes developers of web browsers, assistive technologies, web technologies, and accessibility evaluation tools.
- **Core Accessibility API Mappings 1.2:**
  - Describes how user agents should expose web content semantics to accessibility APIs, providing support across multiple content technologies.
- **Accessible Name and Description Computation 1.2:**
  - Explains how user agents determine names and descriptions of accessible objects from web content, facilitating their exposure in accessibility APIs.
- **HTML Accessibility API Mappings 1.0:**
  - Defines mapping of HTML elements and attributes to platform accessibility APIs, ensuring consistent exposure of roles, states, properties, and events.
- **SVG Accessibility API Mappings 1.0:**
  - Extends Core Accessibility API Mappings 1.2 to map SVG markup to platform accessibility APIs, enabling creation of accessible SVG content like charts and graphs.

## Cognitive Accessibility at W3C

### 1. Cognitive and Learning Disabilities:

- Cognitive and Learning disability types of disabilities can influence how individuals process information.
- They can alter a person's perception, memory, language skills, attention span, problem-solving abilities, and understanding.
- Various terms are used to describe these conditions, including intellectual disabilities, developmental disabilities, ADHD, autism, dementia, dyslexia, and others.

### 2. Technology and Accessibility:

- Technology offers ways for individuals with cognitive and learning disabilities to engage with content and process information that are more suitable for them.
- For instance, these individuals can navigate web content in various ways, access information in text, audio, or other formats, and alter the content's presentation based on their personal needs or preferences.

### 3. Responsibilities of Designers and Developers:

- Designers and developers can optimize the user experience to make their products.
- They can avoid creating accessibility barriers that prevent people from using their products and enhance the user experience for individuals with cognitive and learning disabilities.

### 4. Examples of Cognitive and Learning Disabilities and Accessibility Barriers:

- Specific examples of these disabilities and accessibility barriers can be found in the "[Cognitive and learning](#)" section of the resource "How People with Disabilities Use the Web: Diverse Abilities and Barriers".

## Making Content Usable for People with Cognitive and Learning Disabilities

### 1. Introduction to Cognitive Accessibility at W3C

- **Cognitive and Learning Disabilities:**
  - Cognitive and Learning disability types of disabilities can influence how individuals process information.
  - They can alter a person's perception, memory, language skills, attention span, problem-solving abilities, and understanding.
  - Various terms are used to describe these conditions, including intellectual disabilities, developmental disabilities, ADHD, autism, dementia, dyslexia, and others.
- **Technology and Accessibility:**
  - Technology offers ways for individuals with cognitive and learning disabilities to engage with content and process information that are more suitable for them.
  - For instance, these individuals can navigate web content in various ways, access information in text, audio, or other formats, and alter the content's presentation based on their personal needs or preferences.
- **Responsibilities of Designers and Developers:**
  - Designers and developers can optimize the user experience to make their products.
  - They can avoid creating accessibility barriers that prevent people from using their products and enhance the user experience for individuals with cognitive and learning disabilities.
  - Examples of Cognitive and Learning Disabilities and Accessibility Barriers:
  - Specific examples of these disabilities and accessibility barriers can be found in the "Cognitive and learning" section of the resource "How People with Disabilities Use the Web: Diverse Abilities and Barriers".
- **Examples of Cognitive and Learning Disabilities and Accessibility Barriers:**
  - Specific examples of these disabilities and accessibility barriers can be found in the "[Cognitive and learning](#)" section of the resource "How People with Disabilities Use the Web: Diverse Abilities and Barriers".
- This section provides a straightforward overview of the document's main points.
- Refer to "[How to Use this Document](#)" for additional guidance.
- To assist web content providers in catering to individuals with cognitive and learning disabilities, we have highlighted the following key objectives:

### 2. Objective 1: Help users understand what things are and how to use them



- Use icons, symbols, terms, and design patterns that users are already familiar with to avoid the need for learning new ones.
- Individuals with [cognitive and learning disabilities](#) benefit from familiar behavior and design patterns.
- For instance, adhere to standard hyperlink conventions like underlining and using blue for unvisited links and purple for visited ones.
- See also: [user needs](#), [design patterns](#), [mappings to scenarios](#), and [user testing](#) for objective 1.

### 3. Objective 2: Help users find what they need

- Ensure easy navigation through the system.
- Employ a clear and straightforward layout with visual cues like icons.
- Use clear headings, boundaries, and regions to enhance understanding of the page design.
- See also: [user needs](#), [design patterns](#), [mappings to scenarios](#), and [user testing](#) for objective 2.

### 4. Objective 3: Use clear content (text, images and media)

- Utilize simple words.
- Craft short sentences.
- Use clear images.
- Ensure videos are easy to understand.
- See also: [user needs](#), [design patterns](#), [mappings to scenarios](#), and [user testing](#) for objective 3.

### 5. Objective 4: Help users avoid mistakes

- Aim for a design that reduces the likelihood of errors.
- Only request necessary information from the user.
- When errors happen, ensure it's simple for the user to fix them.
- See also: [user needs](#), [design patterns](#), [mappings to scenarios](#), and [user testing](#) for objective 4.

### 6. Objective 5: Help users focus

- Prevent distractions for the user during tasks.
- Use headings and breadcrumbs to guide and reorient the user if they get distracted.
- Linked breadcrumbs can aid the user in undoing mistakes.
- See also: [user needs](#), [design patterns](#), [mappings to scenarios](#), and [user testing](#) for objective 5.

## 7. Objective 6: Ensure processes do not rely on memory

- Memory barriers hinder individuals with cognitive disabilities from accessing content.
- Examples include long passwords for logging in and voice menus requiring specific number or term recall.
- Ensure there's an easier alternative available for those who require it.
- See also: [user needs](#), [design patterns](#), [mappings to scenarios](#), and [user testing](#) for objective 6.

## 8. Objective 7: Provide help and support

- Ensure users can easily access human assistance.
- Difficulty in providing feedback may hinder your awareness of users' content usage and problems.
- Offer various methods to comprehend content.
- Examples include using graphics, summarizing long documents, adding icons to headings and links, and providing alternatives for numbers.
- See also: [user needs](#), [design patterns](#), [mappings to scenarios](#), and [user testing](#) for objective 7.

## 9. Objective 8: Support adaptation and personalization

- Individuals with cognitive and learning disabilities frequently rely on add-ons or extensions as assistive tools.
- Personalization options, where users can choose preferred settings from a selection, can provide additional support with minimal effort.
- Whenever possible, support personalization to accommodate user preferences.
- Avoid disabling add-ons and extensions as they can provide valuable support.
- Personalization can offer users extra assistance in some cases.
- See also: [user needs](#), [design patterns](#), [mappings to scenarios](#), and [user testing](#) for objective 8.

## 10. Test with real users

- Engage individuals with cognitive and learning disabilities in the research, design, and development phases.
- They possess expertise in what is effective for them.
- Involvement can occur through:
  - Focus groups.
  - Usability tests.
  - Participation in the design and research team.

- See also: [working with users with cognitive and learning disabilities](#) and [section 5](#).

## 11. Important Links

- [Making Content Usable for People with Cognitive and Learning Disabilities \(w3.org\)](#)
- [Cognitive Accessibility at W3C | Web Accessibility Initiative \(WAI\) | W3C](#)
- [WAI-Adapt Overview | Web Accessibility Initiative \(WAI\) | W3C](#)
- [Coga Task Force \(w3.org\)](#)

## User Agent Accessibility Guidelines

### 1. Introduction

- The User Agent Accessibility Guidelines are part of the W3C's efforts to enhance web accessibility.
- They assist developers in creating user agents that are more disability friendly.
- The current version, UAAG 2.0, came out in 2015.
- These guidelines will undergo ongoing revisions and updates in the future.

### 2. What are User Agents?

- "User agents" encompass various tools like browsers, plug-ins, extensions, media players, and more.
- Following UAAG 2.0 enhances accessibility in both user interface and communication with other technologies, including assistive technologies.

### 3. Goals of the UAAG:

- Focus on enhancing accessibility for people with various disabilities.
- Aim to support individuals with visual, auditory, physical, speech, cognitive, language, learning, neurological impairments, and aging-related disabilities.
- UAAG 2.0 strives for equal control over web access for all users, regardless of abilities.
- Address situations where the needs of different disabilities may conflict.
- Define configuration preferences to prevent features designed for one user's accessibility from hindering another user's needs.
- Include requirements for clear documentation and easy configuration to prevent overwhelming users with excessive options.

### 4. Benefitting a Wide Audience:

- UAAG and similar resources offer advantages to accessibility experts, policymakers, and other stakeholders.
- They serve as effective references for enhancing user agent accessibility.
- The guidelines assist developers of assistive technologies by clarifying the information and control they can expect from user agents compliant with UAAG 2.0.

### 5. Five key principles

- The User Agent Accessibility Guidelines (UAAG) 2.0 outline five key principles to ensure user agents, such as web browsers and media players, are accessible to all users, including those with disabilities. Here are the five principles:

- **Principle 1: Perceivable:** User agents must present information in ways that users can perceive. Examples:

- Provide access to alternative content.
- Repair missing content.
- **Principle 2: Operable:** User agents must interface in ways that users can operate.  
Example:
  - Ensure full keyboard access.
- **Principle 3: Understandable:** User agents must be understandable to users.  
Example:
  - Help users avoid and correct mistakes.
- **Principle 4: Programmatic access:** User agents must be robust enough to work with current and future user technologies. Example:
  - Facilitate programmatic access to assistive technology.
- **Principle 5: Specifications and Conventions:** User agents must follow standards and conventions.

These principles are designed to make the web more accessible and inclusive, benefiting not only users with disabilities but all users.

#### 6. Structure of UAAG 2.0:

- Each of the five principles contains specific guidelines, forming a framework to aid developers and authors in understanding success criteria objectives.
- Beneath each guideline are testable success criteria applicable for conformance testing in various contexts such as design specifications, procurement, regulations, and contracts.
- Success criteria are categorized into three levels: A (minimum conformance), AA (recommended conformance), and AAA (advanced conformance), catering to different needs and situations.

#### 7. Overall Purpose of UAAG 2.0:

- UAAG 2.0 is part of a suite of guidelines beneficial for advancing accessibility.
- It benefits individuals with disabilities and developers of accessibility software and technology, providing clear frameworks.

#### 8. Important links

- [UAAG 2.0 Reference: Explanations, Examples, and Resources for User Agent Accessibility Guidelines 2.0 \(w3.org\)](#)

## ETSI EG 203 350 V1.1.1 (2016-11): Human Factors; Guidelines for the design of mobile ICT devices and their related applications for people with cognitive disabilities

### 1. What is ETSI EG 203 350?

- It's a guide by the European Telecommunications Standards Institute (ETSI).
- ETSI is an organization that creates standards for the telecommunications industry.
- These guidelines are for designers and developers to create mobile devices and applications that are easy to use for people with cognitive disabilities.

### 2. Why Is It Important?

- **Inclusivity:** Ensures that mobile devices and apps are usable by everyone, including those with cognitive challenges.
- **Emerging Services:** Helps people with learning disabilities (including the elderly) benefit from new services in rapidly evolving mobile technology.

### 3. Key Areas Covered:

The ETSI EG 203 350 V1.1.1 (2016-11) provides comprehensive guidelines for designing mobile ICT devices and applications, particularly for people with cognitive disabilities. Here's a summary of the key areas covered:

- **Guidelines applicable to any mobile ICT:** These include universal design principles that ensure accessibility and usability for all users, regardless of their abilities or disabilities.
- **Guidelines for the use of language:** Recommendations focus on clear and simple language to improve understanding and ease of use for individuals with cognitive impairments.
- **Guidelines specific to mobile ICT with two-way voice communication:** These guidelines aim to enhance clarity and comprehension during voice communication for users with cognitive disabilities.
- **Guidelines specific to mobile ICT with media playing and recording capabilities:** This section provides guidance on making media playback and recording features more accessible, such as using simple controls and providing clear feedback.
- **Guidelines specific to hardware of mobile ICT:** Hardware design recommendations include considerations for physical interaction with devices, such as button size and feedback, to accommodate users with various cognitive challenges.
- **Guidelines specific to the mobile Web:** This part addresses the design of web content and interfaces to be more intuitive and less cognitively demanding for users.
- **Guidelines specific to mobile software:** Software design guidelines suggest ways to make interfaces simpler and more predictable to support users with cognitive disabilities.
- **Guidelines specific to documentation for mobile ICT:** The focus here is on creating user manuals and help systems that are easy to understand and navigate, aiding users who may struggle with complex instructions.

4. These guidelines are part of ETSI's commitment to inclusivity and ensuring that mobile ICT can be used by the widest possible audience, including those with cognitive disabilities. They emphasize the importance of considering cognitive aspects in the design process to create products that are more accessible and user-friendly.

**5. For Developers and Designers:**

- Sensitivity: Understand the unique requirements of users with cognitive disabilities.
- Testing: Validate designs with real users to ensure usability.
- Legal Compliance: Follow these guidelines to meet accessibility standards.

**6. Important Link:**

- [ETSI EG 203 350 V1.1.1 \(2016-11\): Human Factors; Guidelines for the design of mobile ICT devices and their related applications for people with cognitive disabilities](#)

## Inclusion Europe and Easy-to-read Standard

### 1. Introduction

- Inclusion Europe is a European organization that advocates for the rights of people with intellectual disabilities. They have created an Easy-to-read Standard to make information more accessible to everyone.

### 2. Why is the Easy-to-read Standard important?

- The Easy-to-read Standard is important because it helps to make information accessible to everyone, including people with intellectual disabilities. When information is presented in an easy-to-read format, it is easier to understand and can help people to make informed decisions.

### 3. Key Areas Covered

- These guidelines cover various aspects such as design, language, formatting, accessibility features, and more to ensure that digital platforms are accessible and understandable to a diverse audience.
- **Part 1: General Standards for Easy-to-Understand Information**
  - Understand who will read your information.
  - Choose the best way to share information.
  - Use language that fits your audience.
  - Explain things clearly, avoiding complex words.
  - Involve people with persons with cognitive disabilities in the process.
  - Use simple words, examples, and be consistent.
- **Part 2: Standards for Written Information**
  - Make written material easy to read and copy.
  - Choose clear fonts and avoid all caps.
  - Keep sentences and paragraphs short.
  - Use clear headings and images.
  - Highlight important information and use an easy-to-read symbol.
- **Part 3: Standards for Electronic Information**
  - Test websites with persons with cognitive disabilities.
  - Make homepage and navigation clear for websites.
  - Keep content and design of websites simple.
  - Use text-based links and avoid pop-ups on the website.
  - Ensure compatibility and accessibility of the website.



- **Part 4: Standards for Video Information**
  - Keep videos simple and paced.
  - Design DVD cases and subtitles clearly.
  - Use background voices and images effectively.
  - Ensure good screen visibility and sound quality.
  - Make subtitles easy to read and offer audio description.
- **Part 5: Standards for Audio Information**
  - Review general standards for accessibility.
  - Speak clearly and at a moderate pace.
  - Maintain balanced volume and clear sound.
  - Use pauses and repetition for clarity.
  - Be polite and avoid interruptions in messages.

#### 4. Conclusion:

- The Easy-to-read Standard is an important tool for making information accessible to everyone. By following the guidelines, anyone can create easy-to-read texts that are clear, concise, and easy to understand.

#### 5. Important Link:

- [Inclusion Europe and Easy-to-read Standard](#)

## ISO 24495-1:2023 Plain language , Part 1: Governing principles and guidelines

### 1. Introduction

- It's an international standard for developing documents in **plain language**.
- Provides principles and guidelines to make information clear and understandable.

### 2. Why it is important

- **Universal Design:** Intended for anyone creating documents, especially for public use.
- **Enhances Clarity:** Aims to make documents clear and understandable for a wider audience.
- **Language Inclusivity:** Applicable to most written languages, with examples in English.
- **Scope of Communication:** Focuses on text-based printed or digital information.
- **Useful for Various Creators:** Beneficial for creators of other communication forms, like podcasts and videos.
- **Complements Accessibility:** Supports accessible communication practices without replacing technical accessibility guidelines like Web Content Accessibility Guidelines and EN 301 549.

### 3. Key points from the standard include:

- **Plain Language Principles:** Outlines core principles for clear communication.
- **Guidelines Interpretation:** Details how to apply these principles in practice.
- **Broad Relevance:** Useful for technical writing, legislative drafting, and more.
- **Intentional Limits:** Does not cover all communication types or provide technical accessibility guidance.
- This standard is crucial for ensuring that information is accessible and easily understood, which is particularly important in our diverse and information-rich society.

### 4. Important Link:

- [ISO 24495-1:2023 - Plain language — Part 1: Governing principles and guidelines](#)

## ISO/IEC 29138-1:2018(en) Information technology — User interface accessibility — Part 1: User accessibility needs

### 1. Introduction

- ISO/IEC 29138-1:2018 is a standard that describes the needs and requirements of people with disabilities when it comes to using technology. This standard is part of a larger effort to make sure that technology is accessible to everyone, including people with disabilities.
- ISO/IEC 29138-1:2018 is a standard that describes the needs and requirements of people with disabilities when it comes to using technology.
- The standard covers things like how technology should be designed to be accessible, and how it should work to make sure that people with disabilities can use it.

### 2. Why is this standard important?

- ISO/IEC 29138-1:2018 is important because it helps ensure that technology is accessible to everyone, including people with disabilities.
- Technology is used in many areas of life, and it is important that everyone can use it to access information, communicate with others, and carry out daily tasks.

### 3. Who is ISO/IEC 29138-1:2018 for?

- ISO/IEC 29138-1:2018 is for anyone involved in designing or developing technology, such as software developers, user interface designers, and others.
- It is also important for policymakers and advocates who work to ensure that technology is accessible to all people, including those with disabilities.

### 4. Conclusion:

- The ISO/IEC 29138-1:2018 standard provides guidelines for making digital technology accessible for people with disabilities.
- The standard focuses on the needs of users with different types of disabilities, including visual, hearing, physical, and cognitive disabilities.
- By following these guidelines, technology developers can create products and services that are accessible to everyone, regardless of their abilities.
- Making technology accessible is important because it helps ensure that people with disabilities can participate fully in society and have equal access to information and services.
- These guidelines can help create a more inclusive world where everyone has an equal opportunity to use technology and benefit from its many advantages.

### 5. Important Link:

- [ISO/IEC 29138-1:2018 - Information technology — User interface accessibility — Part 1: User accessibility needs](#)

## Tools for Accessibility Testing

### Keyboard Accessibility

#### 1. Why it is important?

- Keyboard accessibility is crucial for users with motor disabilities who rely solely on keyboards for navigation.
- Some users have conditions like tremors, limited hand mobility, or no hands at all, making keyboard navigation essential.
- Various types of keyboards, including modified ones or alternative hardware, are used by different users to access content.
- Blind users also rely on keyboards for navigation on websites.
- Even users without disabilities may prefer using keyboards for navigation due to efficiency or personal preference.

#### 2. Main Issues

- **Focus indicators:**
  - Keyboard users rely on the **Tab key** to navigate interactive elements like links and buttons.
  - Web browsers automatically provide focus indicators, typically shown as borders or highlights around focused elements.
  - Avoid hiding focus indicators using CSS styles like **outline:0** or **outline: none**.
  - Customize CSS to enhance focus indicator visibility and ensure it matches your site design.
- **Navigation order:**
  - Default keyboard navigation order should follow the visual flow of the page, typically left to right, top to bottom.
  - Ensure underlying source code structures the navigation order correctly.
  - Use CSS for visual presentation without altering default keyboard navigation order.
  - Avoid using **tabindex** values to change navigation order.
- **Items that should not receive keyboard focus:**
  - Links, buttons, and form controls are naturally accessible to keyboard users and should be used for interactivity.
  - Non-interactive elements should not be made keyboard focusable to prevent confusion.

- Ensure <a> elements have non-empty **href** attributes to be accessible to screen reader users.
- **Inaccessible custom widgets:**
  - Custom controls must be accessible to keyboard users and may require `tabindex="0"` for keyboard focus.
  - Utilize ARIA to ensure proper presentation to screen reader users and define keyboard interactions.
  - Ensure custom widgets have intuitive interactions, work with both keyboard and mouse, and use standardized keystrokes.
- **Lengthy navigation:**
  - Tabbing through lengthy navigation can be demanding for keyboard-only users, particularly those with motor disabilities.
  - Implement best practices to facilitate efficient keyboard navigation:
    - Provide a **"skip to main content"** link.
    - Use a clear **heading structure**.
    - Utilize regions or **ARIA landmarks** to aid navigation.

## 5. Important Link:

- [WebAIM: Keyboard Accessibility](#)

## Color Contrast Analyzer (CCA)

### 1. What is the CCA?

- The Colour Contrast Analyzer (CCA) is a free tool by TPGi.
- It helps you check the **contrast ratio** between two colors.

### 2. Why Is It Important?

- **Readability:** Ensures that text and visual elements are easy to read.
- **Accessibility:** Benefits individuals with vision impairments (like color blindness or low vision).
- **Compliance:** Helps meet **WCAG (Web Content Accessibility Guidelines)** standards.
- **Versatility:** Use it for websites, app development, social media graphics, presentations, and more.

### 3. Key Features:

- **WCAG Compliance Indicators:** Shows if your content meets accessibility standards.
- **Dark Mode Support:** Works well even in dark-themed interfaces.
- **Color Selection Options:** Manually enter colors, use sliders, or pick colors from your design.
- **Color Blindness Simulation:** Visualize how different color-blind users perceive your content.

### 4. WCAG Contrast Levels:

- **Level AA:** Requires a contrast ratio of at least 4.5:1 for normal text and 3:1 for large text.
- **Level AAA:** Demands even higher contrast ratios (7:1 for normal text, 4.5:1 for large text).

### 5. Important Links:

- [Colour Contrast Analyzer - TPGi](#)
- [How to use Colour Contrast Analyser Tool - YouTube](#)

## NVDA Screen Reader

### 2. What is Screen Reader?

- A Screen Reader is a type of assistive technology. It converts text and image content into speech or braille output. It's essential for people who are blind, and useful for those who are visually impaired, illiterate, or have a learning disability.
- Some examples of screen readers are Windows Narrator, JAWS, and NVDA.

### 3. What is NVDA Screen Reader?

- **NVDA** is a **free, open-source** screen reader designed for **Microsoft Windows**.
- Developed by **NV Access**, it's portable and can output visual content as **audio or braille**.

### 4. Why Is It Important?

- **Accessibility:** Ensures websites are usable by everyone, including people with disabilities.
- **Legal Compliance:** Helps meet standards (e.g., **WCAG**).
- **User Experience:** Improves usability and inclusivity.
- **Testing Tool:** Widely used for **digital accessibility testing**.

### 5. Key Features of NVDA:

- **Free and Robust:** Remains 100% free and provides reliable control over web browsers and Windows applications.
- **Portability:** Install it on a **USB flash drive** for use on multiple machines.
- **Customizable:** Extend with **plugins** for specific applications or magnification.
- **Global Reach:** Supports **50+ languages** and has an international user community.

### 6. Important Links:

- [NV Access | About NVDA](#)
- [WebAIM: Keyboard Shortcuts for NVDA](#)
- [List of screen readers - Wikipedia](#)
- [Screen Reader Basics: NVDA -- A11ycasts #09 - YouTube](#)

## Co-Design

Co-design is a teamwork approach where people with different backgrounds and expertise work together to give advice and decide on projects, policies, programs, or initiatives. It's inclusive and collaborative, ensuring everyone's input is valued.

### Introduction to Co-Design

- Co-design is when people with different skills and experiences work together to give advice and make decisions on projects, policies, or programs.
- This process starts from planning and goes all the way to reviewing the project's progress.
- Co-design is used in many areas, not just specific to disabilities.
- When used for disabilities, it's important to have a diverse group, including people with different disabilities, ages, genders, and backgrounds.
- For projects involving persons with intellectual disabilities, it's crucial to include them in the co-design process. Co-Design is how we bring people with disabilities and other together to design and build accessible solutions.
- This group may also include other people like those who benefit from the project, representatives from funding organizations, and experts in the subject matter.

### Co-Design 5 Phases

Co-Design has 5 phases:

#### 1. Discovery:

- **Challenge Identification:** Identify and define the problem or challenge you're facing.
- **Research and Exploration:** Conduct research and gather relevant information about the problem, stakeholders, and context.
- **User Engagement:** Engage with stakeholders and users to gain insights into their needs, preferences, and experiences.

#### 2. Interpret:

- **Data Analysis:** Analyze the gathered information and data to identify patterns, trends, and key insights.
- **Understanding Stakeholder Perspectives:** Gain a deeper understanding of stakeholders' perspectives, concerns, and expectations.
- **Identifying Opportunities:** Identify opportunities for innovation and improvement based on the interpreted data and insights.

#### 3. Ideation:

- **Brainstorming:** Generate creative ideas and potential solutions to address the identified challenges and opportunities.



- **Co-Creation Workshops:** Facilitate collaborative workshops where stakeholders can contribute ideas and concepts.
- **Concept Development:** Develop and refine concepts and prototypes that address the identified needs and goals.

#### 4. Experiment:

- **Prototyping:** Create prototypes or mock-ups of the proposed solutions to test and iterate upon.
- **Pilot Testing:** Conduct pilot tests or trials to gather feedback and insights from users and stakeholders.
- **Iterative Development:** Continuously refine and improve the prototypes based on user feedback and testing results.

#### 5. Evaluation:

- **Feedback Collection:** Gather feedback from users, stakeholders, and other relevant parties on the implemented solutions.
- **Performance Assessment:** Evaluate the effectiveness, efficiency, and usability of the solutions against predefined criteria and objectives.
- **Iterative Improvement:** Use the evaluation findings to inform further iterations and improvements to the solutions.

#### Additional Information:

- These phases are iterative and may overlap or repeat throughout the co-design process.
- Collaboration and communication among stakeholders are essential in each phase to ensure alignment and shared understanding.
- Flexibility and adaptability are key as the process may evolve based on new insights, challenges, or opportunities encountered along the way.
- The goal of co-design is to create innovative and impactful solutions that address the needs and aspirations of all stakeholders involved.

## Empathy Maps

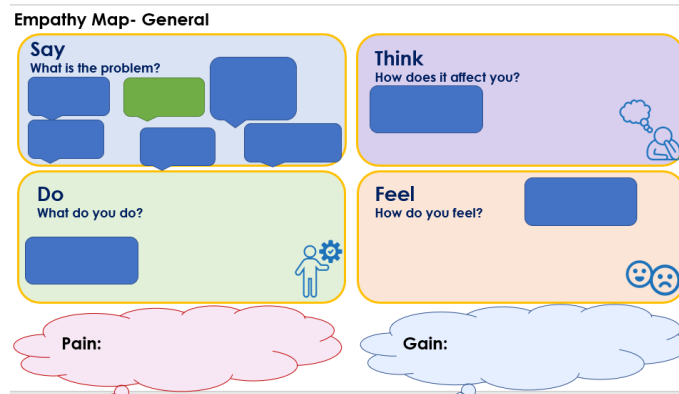
### 1. Introduction

- An Empathy Map is a tool used in design research to summarize insights gathered from interactions with people.
- It consists of four quadrants, each focusing on different aspects of a person's experience:
  - **Said:** What the user verbalized or communicated during the observation.
  - **Did:** The actions or behaviors exhibited by the user.
  - **Thought:** The thoughts or internal reflections of the user.
  - **Felt:** The emotions or feelings experienced by the user.

- While it's relatively straightforward to capture what the user said and did, understanding their thoughts and feelings requires careful observation and analysis.
- Empathy maps serve as a valuable background for creating personas, which are detailed representations of user archetypes often used in design processes.

## 2. How to create an Empty Map

- **Step 1: Fill out the Empathy Map:**
  - Lay out the four quadrants (Say, Do, Think, Feel) on paper or a whiteboard.
  - Review your research notes, pictures, audio, and video.
  - Fill in each quadrant
    1. **What did the user SAY?:** Capture significant quotes and keywords.
    2. **What did the user DO?:** Describe observed actions or include visual evidence.
    3. **What did the user THINK?:** Dig deeper into their motivations and beliefs.
    4. **How did the user FEEL?:** Consider emotions based on cues.



- **Step 2: Synthesize NEEDS:**
  - Based on the empathy map, synthesize the user's needs. Understand what drives them and what they truly require.
- **Step 3: Synthesize INSIGHTS:**
  - Extract unexpected insights from the empathy map. These insights inform design decisions and solutions.

## 3. Benefits of Empty Maps

- **User-Centered:** Empathy maps keep the focus on users' needs and experiences.
- **Deeper Understanding:** They help designers gain insights beyond surface-level observations.
- **Enhanced Communication:** Teams can align better when they share a common understanding of users.

- **Identifying Opportunities:** Empathy maps reveal areas for improvement and innovation.

## Good Practice: Co-Design Workshop of RTCN Project in Madrid Spain

### 1. RTCN - Right to Connect Project

- Right to Connect: Digital inclusion for persons with intellectual disabilities" is an innovative co-designed e-learning project involving persons with intellectual disabilities (PwID). The project builds on the experience of the ENTELIS and ENTELIS+ projects. It involves 10 partners in 7 countries forming a multidisciplinary consortium of service providers, universities, a company and European umbrella organizations. The project aims at fostering the digital skills development of PwID to enhance their participation in society and to express active citizenship.

### 2. Why Good Practice?

- **Diversity:**
  - Include a variety of participants, representing different ages, cultures, different languages, locations, sexual and gender identities, and disability experiences.
  - Different countries and organizations participated: Atempo (Pilot partner) & JKU (Digital Accessibility & Research support) from Austria; AIAS Bologna, Italy (Pilot partner); Maragrita, Greece(Pilot partner); SJOG, Ireland (Pilot partner); FJC (Pilot partner) & Hermes (Technical Development/ Developer team) from Spain.
  - Different stakeholders participated as Co-Designers including: Persons with intellectual disabilities, Accessibility testers, Researchers, and Developers.
  - This diversity ensured a rich pool of perspectives and expertise, enhancing the quality and inclusivity of the Co-Design workshop.
- **Accessibility:**
  - The venue provided by FJC was perfect for this event with great accessibility and space for everyone to move around the workshop floor freely and confidently.
  - Accessibility researchers are participated to make sure that development of e-learning platform as per European standards like ETSI 301549 etc. Participants would have become more informed about the significance of accessibility in digital platforms.
- **Respect:**
  - Ensure all participants feel welcomed and respected.
  - Treat everyone with dignity and value their contributions.
- **Equality:**
  - Give all participants an equal voice and consider their contributions equally.
- **Safety:**

- Create a supportive environment where participants feel safe to share their ideas.
- Ensure no individual faces retribution for their contributions.
- **Commitment:**
  - Demonstrate genuine dedication to co-design from planning to review stages of the RTCN project.
- **Authenticity:**
  - Ensure the process is meaningful and incorporates participants' input into the final product.
- **Confidentiality:**
  - Treat all shared materials confidentially and refrain from discussing them outside the group.
- **Acknowledgement:**
  - Recognize and appreciate the skills and experiences of all participants, including considering payments and certificates for those participating personally.

### 3. What impact/changes did the participants experience by joining this workshop?

There were several impacts and changes that all the entire Irish team noted during the two-day co-design workshop, and they included.

- **Increased Confidence:**
  - Workshop provided an inclusive space where all opinions mattered.
  - Participants gained confidence by actively participating and sharing thoughts.
  - Families reported happiness and enjoyment from participants, sharing their experiences afterward.
- **Enhanced Technical Skills:**
  - Engaging in digital mock-ups and exploring platform functionalities improved tech skills.
  - Training from Spanish co-designers on using Quizzes enhanced peer learning and admiration.
- **Improved Communication:**
  - Interaction with different countries and languages honed communication skills.
  - Participants learned to express ideas clearly and encountered various communication methods.
- **Better Understanding of Inclusive Design:**

- Focus on co-design and user feedback deepened understanding of inclusive design.
- Importance of designing with end-users was emphasized.
- **Increased Awareness of Accessibility:**
  - Guidance on accessibility elements raised awareness about their significance in digital platforms.
- **Collaboration Skills:**
  - Working in multicultural and multilingual teams fostered teamwork and appreciation of different perspectives.
  - Participants learned to find standard solutions through collaboration.
- **Enhanced Problem-solving Skills:**
  - Activities like mapping feedback and co-designing required real-time problem-solving.
  - Participants identified issues and devised solutions collaboratively.
- **Empathy Development:**
  - Exercises like "empathy map" helped participants step into others' shoes, fostering empathy.
  - Understanding others' feelings, needs, and perspectives was emphasized.
- **Pride and Ownership**
  - Participants felt pride and ownership seeing their ideas potentially implemented in the final product.
- **Learning from Peers:**
  - Observing and interacting with other teams provided valuable insights and knowledge.
  - Specialized knowledge from teams like Atempo and the Spanish team was particularly beneficial.

In conclusion, the workshop served the purpose of refining the platform and providing growth opportunities for all participants.

#### 4. Co-Design Workshop steps

- Every Co-Design workshop has 5 phases: 1. Discovery 2. Interpret 3. Ideation 4. Experiment 5. Evaluation.
- Explain importance of **Empathy** in Design:
  - Empathy in design means understanding and sharing the feelings, thoughts, and experiences of users. Here users are persons with intellectual disabilities.
  - It's essential for creating products and experiences that meet users' needs.
- **Use of Empathy Maps:** Empathy maps are most relevant during the Interpret phase of the Co-Design process. In this phase, you've gathered insights and need to understand

users' feelings, thoughts, and actions. Empathy maps provide a deeper understanding, guiding you toward more user-centered solutions. It contains: What did user Say, Do, Think and Feel.

- Say- What are the problems identified?
- Think - How does it affect you?
- Do - What do you do?
- How do you feel?

○ **Prototyping:** It is part of Experimentation phase.

- Just start building: Creating a prototype will help you visualize your idea more clearly.
- Don't spend too much time: Remember, this is still a part of the creative process, not a final product.
- Remember what you're testing for: Stay focused on the problem you're trying to solve.
- Build with the user in mind: Incorporate insights from your co-design team and prioritize user needs and preferences in your prototype development.

○ **Prototype Guessing Game:**

- In the context of co-design, a "Guessing Game" prototype could involve creating an early version of a product or feature and inviting stakeholders to interact with it.
- Co-designers can explore the prototype, provide feedback, and make informed guesses about how it might function or meet their needs.
- For example: You can ask participants that look at the apps you are using- contrast and compare with the current apps- youtube , how does youtube do it?, spotify, whatsapp, facebook, figma.

○ **What questions to ask:** The questions "What do you like?", "What do you wish?", and "What if?" are particularly relevant during the evaluation phase of Co-Design. In this phase, Co-designers brainstorm creative ideas and explore potential solutions. Let's break it down:

➤ **What Do You Like?:**

1. During evaluation, understanding what participants like helps identify existing positive elements or features that can be built upon.
2. Co-designers can express preferences related to design, functionality, or user experience.

➤ **What Do You Wish?:**

1. This question encourages participants to think about improvements or enhancements.
2. Participants can share their wishes, desires, and aspirations related to the project. It sparks innovative thinking.

➤ **What If?**

1. “What If” prompts participants to think beyond the obvious. It’s about exploring alternative scenarios.
2. “What If” prompts participants to think beyond the obvious. It’s about exploring alternative scenarios.

➤ **For more simplicity**

1. If co-designers respond “I like it” then Ask “Can you tell me why you like it”.
2. If co-designers say “I agree with [another person/participant]” then Ask “Why do you agree/disagree”? Or “Can you tell me about why you agree/disagree?”
3. A participant makes a comment, and you are not entirely sure of the meaning then repeat back their comment to clarify and phrase as a question to get more meaning/clarification.
4. If you ask a question to the group and do not get any response, then adjust the question to make it more comprehensible. Go around the table/ virtual meeting by calling out names, ask people to contribute.

- **Further Evaluation Phase questions:** During the evaluation phase of co-design, asking questions like “Are you happy with the Co-Design Workshop?” and “What have we discovered?” helps assess participant satisfaction, reflect on the process, gather feedback for improvement, and validate outcomes. These questions guide continuous learning and refinement.

**5. Exercises performed to develop accessible e-learning platform for persons with intellectual disabilities.**

- Every pilot partner (Atempo, Austria; AIAS Bologna, Italy; Maragrita, Greece; SJOG, Ireland; FJC Spain) has done co-design session in their own country before final Co-Design session in Spain.
- **Exercise 1 Course Set up**
  - Partners explored current course creation functionality and proposed alternatives.
  - Completed co-design tools like empathy map, "I like; I wish; What if" maps in their own language.
  - Feedback from each country was mapped across the room.
  - Break was taken after each session.
- **Exercise 2 Course Edit**
  - Partners tested course edit functionality and alternatives.

- Co-designed with facilitator and technical partner on desired portal functionality.
- Discussions were informative, allowing everyone to share thoughts and ideas.
- Visual prototypes aided understanding, with technical team providing insight on feasibility.
- **Exercise 3 Multimedia**
  - Explored current multimedia edit functionality and alternatives.
  - Discussed simplification by removing confusing elements.
  - AIAS and Spanish teams contributed ideas for making multimedia editing easier.
- **Exercise 4 Text Input**
  - Partners tested current text input process and alternatives.
  - Atempo team shared insights, suggesting text on right, image on left, specific font and size preferences.
  - Feedback included reducing font, color, and size choices to improve usability.
- **Exercise 5 Forum**
  - Uncertainty about forum usage and end users led to skipping this exercise.
- **Exercise 6 Quiz**
  - Spanish team led the quiz exercise, sharing experience with educational games.
  - Spanish co-designers trained other teams in creating quizzes on the platform.
  - Impressed by the skills, patience, and problem-solving demonstrated by the Spanish team.

## 6. Evaluation

- **Demographic Trends:**
  - Gender:
    1. Males were dominant in Ireland, Spain, Italy, and Austria.
    2. Greece had exclusively female participants.
  - Age groups:
    1. Younger generation (18-25 years) dominated in Spain, Austria, and Greece.
    2. Middle-aged group (26-40 years) was more prevalent in Ireland and Italy.
    3. Representation from older age groups, notably in Ireland and Italy.
- **Technology & Accessibility Insights:**
  - Communication support: Significant portion of participants across regions expressed a need, with Ireland and Greece recording the highest percentages.



- Built-in accessibility features: High utilization in Ireland and Greece, lesser in Spain and Italy.
- Preferences: Recurrent preferences for zoom functions, text resizing, and reading text aloud.
- **Social Media & Technology Usage:**
  - Communication methods: Typing using app's keyboard was common, alongside voice messages, video calls, and emojis.
  - Technical challenges: Included lack of awareness about technology options, login issues, and difficulty in understanding and usage.
- **Workshop Experience & Engagement:**
  - Motivations for participation: Desire to try something new, meet new people, and work with technology.
  - Benefits: Collaborative learning was highlighted, emphasizing sharing ideas, understanding different perspectives, and teamwork.
  - Importance of co-design: Active engagement emphasized collaboration, idea sharing, and iterative nature of the co-design process.
  - Value perception: Almost all participants found value in workshops for skill acquisition and networking.
- **Feedback & Comments:**
  - Overall sentiment: Positive feedback expressed gratitude and enjoyment.
  - Areas for improvement: Better organization, translation services, and refining platform for inclusivity.
  - Desire for future workshops: Recurrent theme across all feedback indicating high satisfaction and perceived value.
- In essence, the co-design workshops serve as invaluable platforms for diverse participants to collaborate, learn, and share, despite challenges in technology and accessibility, fostering personal and collective growth.

## Conclusion

### 1. Guidelines Used:

1. The 'Right to Connect' project operates within European Union (EU) countries, necessitating adherence to ETSI 301 549 accessibility guidelines mandated by the EU.
  - WCAG 2.1, UAAG, ARIA, and ATAG 2.0 are internally referenced by ETSI 301 549.
2. We have prioritized guidelines relevant to persons with intellectual disabilities.
  - Specifically, we incorporated 'Making Content Usable for People with Cognitive and Learning Disabilities', integral to WCAG 2.1 guidelines.
3. Covered ETSI EG 203 350 V1.1.1 (2016-11) guidelines crucial for developing mobile applications accessible to persons with intellectual disabilities.
4. Inclusion Europe and Easy-to-Read guidelines: Specifically cater to the needs of persons with intellectual disabilities, ensuring accessibility and comprehensibility for all users.
5. Utilization of ISO Standards:
  1. Adhering to ISO 24495-1:2023 ensures clear, concise, and easily understandable content.
  2. ISO/IEC 29138-1:2018 focuses on user interface accessibility, addressing the specific needs of users with disabilities.
6. By incorporating these guidelines, we uphold international best practices in accessibility, enhancing the usability and inclusivity of our digital platforms.

### 2. Accessibility Examples for Developers and Designers:

1. We've provided practical examples for creating accessible digital platforms.
2. While comprehensive, these examples may not cover all aspects of accessibility.
3. Adherence to ETSI 301 549 guidelines is crucial for ensuring platform accessibility.

### 3. Accessibility and SDLC:

1. Emphasized starting accessibility considerations from the requirement phase in the software development life cycle (SDLC).
2. Implementation of accessibility from the initial phase reduces costs and time compared to retrofitting in later stages.
3. Highlighted the importance of incorporating accessibility throughout all phases of the SDLC, including requirement, design, implementation, testing, and maintenance.

### 4. Co-Design

1. Co-design workshops serve as invaluable platforms for diverse participants to collaborate, learn, and share, despite challenges in technology and accessibility, fostering personal and collective growth.