

The Must-Have WCAG 2.1 Checklist

Practical Resource Guide



ESSENTIAL
ACCESSIBILITY.

What is WCAG?

The Web Content Accessibility Guidelines or WCAG provides technical specifications to improve the accessibility of web content, websites and web applications on desktop computers, laptops, tablets and mobile devices for people with a wide range of disabilities, including auditory, cognitive, neurological, physical, speech and visual disabilities.

W3C, or World Wide Web Consortium, is a global community of accessibility experts who are striving to make the internet as inclusive as possible. The Web Accessibility Initiative (WAI) develops WCAG and related resources with input from individuals and organizations around the world.

The guidelines are mainly for the use of web content developers, web authoring tool developers and related professions; they aren't intended to be an introduction to accessibility. However, it is helpful for companies and organizations, especially employees who contribute to their digital properties, to have a general understanding of WCAG, its purpose and how it benefits not only people with disabilities, but all users.

This checklist is a practical resource guide for experienced accessibility professionals and for those newer to the industry. The first part is a primer of industry nomenclature and accessibility testing approaches. Fillable and printable checklists follow.

WCAG 2.1 Highlights

Mobile

- Improves support for touch interactions, keyboard and mouse
- Avoids unintended activation of device sensors

Low Vision

- Extends contrast requirements to graphics
- Improves text and layout adaptability

Cognitive and Learning Disabilities

- Enables more detailed description of page controls and elements to support personalization of user interface

EN 301 549 Coordination

- Harmonized update in progress in Europe
- Particularly for expanded mobile

What's Different About WCAG 2.1?

WCAG 2.0, released nearly 10 years ago, contains 12 guidelines for digital accessibility, divided among four principles with the acronym P.O.U.R: Perceivable, Operable, Understandable and Robust. Each guideline has a list of “success criteria,” or requirements (61 in total), for making content – including text, images, sounds, code and markup – more accessible. In addition, WCAG 2.0 has three levels of conformance: **A (minimum accessibility), AA (addresses the major, most common accessibility issues) and AAA (the highest standard).**

The success criteria found in WCAG 2.0 are included in WCAG 2.1 – the wording of those criteria has not changed. That means that WCAG 2.1 is “backwards compatible” or, as W3C puts it, “content that conforms to WCAG 2.1 also conforms to WCAG 2.0.” What’s new about WCAG 2.1 is that it includes 17 new success criteria related to mobile accessibility, as well as provisions that will benefit more people.

		SUCCESS CRITERIA		
Level		WCAG 2.0	WCAG 2.1	TOTAL WCAG 2.0 and 2.1
A	The most basic web accessibility features	25	5	30
AA	Deals with the biggest and most common barriers for users with disabilities	13	7	20
AAA	The highest (and most complex) level of web accessibility	23	5	28
Total		61	17	78

Should You Follow WCAG 2.0 or 2.1?

WCAG 2.1 doesn’t supersede or cancel out WCAG 2.0 – they are both “existing standards” – but W3C encourages organizations to use the most recent version of WCAG when developing or updating their content or digital accessibility policy.

Authorities that enforce major accessibility laws, including the Americans with Disabilities Act (ADA), Section 508 of the Rehabilitation Act, and the Accessibility for Ontarians with Disabilities Act (AODA), continue to require that organizations comply with WCAG 2.0, conformance level AA. However, this could change in the future.

If your company is in the process of making its website and other digital tools and technologies accessible and conform to WCAG 2.1 Level A and AA, it’s a good idea to implement the additional 17 success criteria now to ensure maximum accessibility.

Largest Changes in WCAG

2.1: Mobile Specific Highlights

When testing mobile platforms for accessibility, individuals had to map the old WCAG 2.0 standards to mobile design guidelines to apply WCAG to mobile.

Now, new mobile requirements in WCAG 2.1 help guide the way:

Speech Input

- Character Key Shortcuts
- Label in Name

Pointer

- Pointer Gestures
- Pointer Cancellation
- Target Size

Input Methods

- Concurrent Input Mechanisms
- Motion Actuation

Device Settings

- Orientation
- Other Standards that can be applicable to Mobile

Identify Input Purpose

Reflow

- Non-Text Contrast
- Text Spacing
- Content on Hover or Focus

Level Breakdown

Level A – 5 New Standards

Guideline 2.1 Keyboard Accessible

- [2.1.4 Character Key Shortcuts](#)
- [Guideline 2.5 Input Modalities](#)
- [2.5.1 Pointer Gestures](#)
- [2.5.2 Pointer Cancellation](#)
- [2.5.3 Label in Name](#)
- [2.5.4 Motion Actuation](#)

Level AA – 7 New Standards

Guideline 1.3 Adaptable

- [1.3.4 Orientation](#)
- [1.3.5 Identify Input Purpose](#)

Guideline 1.4 Distinguishable

- [1.4.10 Reflow](#)
- [1.4.11 Non-Text Contrast](#)
- [1.4.12 Text Spacing](#)
- [1.4.13 Content on Hover or Focus](#)

Guideline 4.1 Compatible

- [4.1.3 Status Messages](#)

Level AAA – 5 New Standards

Guideline 1.3 Adaptable

- [1.3.6 Identify Purpose](#)

Guideline 2.2 Enough Time

- [2.2.6 Timeouts](#)

Guideline 2.3 Seizures and Physical Reactions

- [2.3.3 Animation from Interactions](#)

Guideline 2.5 Input Modalities

- [2.5.5 Target Size](#)
- [2.5.6 Concurrent Input Mechanisms](#)

Each of these breakdown levels link to the W3C for additional definition depth.

Testing Against WCAG 2.1

When we use the term “digital accessibility testing” we’re referring to the step-by-step process of thoroughly and diligently checking whether or not an internal or external-facing website, mobile app, software application, or LMS is usable by people with disabilities.

Proper accessibility testing of these digital properties typically involves extensive manual scrutiny of individual web pages against the WCAG 2.1 success criteria, as well as tests of various functions such as product searches and online form submissions.

It can also mean using automated testing tools to check for accessibility of various, specific elements of the digital property. The best approach is usually a combination of both.

Automated Testing

There are many tools available that will perform an automated test of certain components of a website, mobile experience, app, or electronic document. They can be quite useful for doing preliminary inspections. Accessibility experts often use various tools in concert to effectively test a website.

Automated accessibility testing is a great way to learn more about the different reasons why persons with disabilities might encounter problems. However, this form of testing has limitations. Only about 30% of the WCAG 2.0 success criteria and precisely 0% of the WCAG 2.1 success criteria can be tested using an automated tool.

There are a number of tools to conduct preliminary automated testing. It is recommended that you determine which tools will work with firewall settings and design and developer teams use the same tools. QA teams will likely leverage even more tools to ensure compliance and usability. Here is a listing of free tools for you to peruse to get you started:

Code Validation

W3C CSS Validator software was created by the W3C to help web designers and web developers check Cascading Style Sheets (CSS). It can be used on their free service on the web, or downloaded and used either as a java program, or as a java servlet on a web server. This tool will allow comparison of style sheets to the CSS specifications, helps find errors, typos, and incorrect uses of CSS. It will also advise when the CSS poses some usability risks.

Color Contrast and Color Blindness

The Colour Contrast Analyser is a downloadable tool that helps determine the legibility of text and the contrast of visual elements, such as graphical controls and visual indicators. Currently, the tool supports WCAG 2.1 compliance indicators.

Mobile Accessibility

Two tools serve the mobile accessibility space. For Android, Accessibility Scanner checks for accessibility in Android apps. For iOS, Accessibility Inspector can be used to check for accessibility. Both apps are utilized by developer and QA audiences.

Document Accessibility

The Document Accessibility Toolbar (DAT) is a dedicated accessibility ribbon menu for Microsoft Word that makes it quicker and easier to create accessible documents. This toolbar features a range of hand-picked and custom-built functions to optimize and validate a document for accessibility. The PDF Accessibility Checker PAC 3 allows for the checking of PDFs for accessibility. It works even for people that do not have Adobe Acrobat Professional.

Web Accessibility

The WAVE by WebAIM tool is one of the favorites in the industry as it uses a simple Red, Yellow, Green icon to show errors, warnings and good areas. It also has an ARIA check and color contrast analyzer built in, and you can turn on/off style sheets. Currently, it only checks against WCAG 2.0. Other tools offer specificity of analysis for content, design, developers and QA teams.

Manual and Functional Testing

Manual and functional testing are an essential component of accessibility testing. These testing types involve using human expertise to check the automated tests and then having trained teams and persons with disabilities actually engage with the digital experiences directly. There simply isn't any technology that can replace this portion of accessibility testing. When conducting a manual review, a sample test plan should include the following components:

Ensure you're testing the most accurate environments for all users:

- Check Google Analytics to determine high trafficked pages.
- Cross reference this list against WebAIM's Screen Reader Survey and Low Vision Survey.

Cross check the accessibility features with HTML 5 Browser Accessibility to ensure all features are supported

Keyboard accessibility check

Code validation check

Automated accessibility tool check

User stories for manual and functional testing:

- Develop accessibility-related user stories, such as:
 - As a keyboard only user, I want the ability to reach all links (text or image), form controls and page functions, so that I can perform an action or navigate to the place I choose.
 - As a user who is hearing-impaired, I want closed captioning functionality so that I can have access to all information provided in video clips.
- Ask yourself the questions:
 - Why is the screen reader reading the sidebar before the main article?
 - Do I have to tab through every page and every navigation before getting to the content? (Why isn't there a skip to content link?)
 - What does image IMG_238429.jpg mean?
 - What did I miss on the page?

There are a number of ways organizations can ramp up their manual testing capabilities:



Option 1

Build an in-house team of accessibility testers to perform QA on digital properties in development.



Option 2

Hire an outside consultant to systematically test the website, apps and/or electronic documents. They will provide you with a one time report outlining the issues and barriers encountered.



Option 3

Work with an accessibility partner over time with access to a team of testers who manually check digital properties in multiple environments using different assistive technologies. These partners also work with you to develop a prioritization report outlining the critical, high, medium and low-level issues, monitor your digital properties on an ongoing basis and integrate seamlessly into your backend systems to better collaborate with your team.

WCAG 2.1 Level A Checklist

Project:
Digital Asset:

Date:

Success Criteria	Description	Notes	Pass/Fail	Complete
<u>1.1.1 – Non-text Content</u>	Provide text alternatives for non-text content		<input type="checkbox"/>	
<u>1.2.1 – Audio-only and Video-only (Pre-recorded)</u>	Provide an alternative to video-only and audio-only content		<input type="checkbox"/>	
<u>1.2.2 – Captions (Pre-recorded)</u>	Provide captions for videos with audio		<input type="checkbox"/>	
<u>1.2.3 – Audio description or Media Alternative (Pre-recorded)</u>	Video with an audio has a second alternative		<input type="checkbox"/>	
<u>1.3.1 – Info and Relationships</u>	Logical structures		<input type="checkbox"/>	
<u>1.3.2 – Meaningful Sequence</u>	Present content in a meaningful order		<input type="checkbox"/>	
<u>1.3.3 – Sensory Characteristics</u>	Use more than one sense for instructions		<input type="checkbox"/>	
<u>1.4.1 – Use of Colour</u>	Don't use presentation that relies solely on colour		<input type="checkbox"/>	
<u>1.4.2 – Audio Control</u>	Don't play audio automatically		<input type="checkbox"/>	
<u>2.1.1 – Keyboard</u>	Accessible by keyboard only		<input type="checkbox"/>	
<u>2.1.2 – No Keyboard Trap</u>	Don't trap keyboard users		<input type="checkbox"/>	
<u>2.1.4 – Character Key Shortcuts</u>	Do not use single key shortcuts or provide a way to turn them off or change them		<input type="checkbox"/>	
<u>2.2.1 – Timing Adjustable</u>	Time limits have user controls		<input type="checkbox"/>	
<u>2.2.2 – Pause, Stop, Hide</u>	Provide user controls for moving content		<input type="checkbox"/>	
<u>2.3.1 – Three Flashes or Below</u>	No content flashes more than three times per second		<input type="checkbox"/>	

WCAG 2.1 Level A Checklist (continued)

Success Criteria	Description	Notes	Pass/Fail	Complete
2.4.1 – Bypass Blocks	Provide a “Skip to Content” link		<input type="checkbox"/>	
2.4.2 – Page Titled	Helpful and clear page title		<input type="checkbox"/>	
2.4.3 – Focus Order	Logical order		<input type="checkbox"/>	
2.4.4 – Link Purpose (In Context)	Every link’s purpose is clear from its context		<input type="checkbox"/>	
2.5.1 – Pointer Gestures	Users can perform touch functions with assistive technology or one finger		<input type="checkbox"/>	
2.5.2 – Pointer Cancellation	This requirement applies to web content that interprets pointer actions		<input type="checkbox"/>	
2.5.3 – Label in Name	The name contains the text that is presented visually		<input type="checkbox"/>	
2.5.4 – Motion Actuation	Functions that are triggered by moving a device or by gesturing towards a device can also be operated by more conventional user interface components		<input type="checkbox"/>	
3.1.1 – Language of Page	Page has a language assigned		<input type="checkbox"/>	
3.2.1 – On Focus	Elements do not change when they receive focus		<input type="checkbox"/>	
3.2.2 – On Input	Elements do not change when they receive input		<input type="checkbox"/>	
3.3.1 – Error Identification	Clearly identify input errors		<input type="checkbox"/>	
3.3.2 – Labels or Instructions	Label elements and give instructions		<input type="checkbox"/>	
4.1.1 – Parsing	No major code errors		<input type="checkbox"/>	
4.1.2 – Name, Role, Value	Build all elements for accessibility		<input type="checkbox"/>	
Result				<input type="checkbox"/>

WCAG 2.1 Level AA Checklist

Project:
Digital Asset:

Date:

Success Criteria	Description	Notes	Pass/Fail	Complete
<u>1.2.4 – Captions (Live)</u>	Live videos have captions		<input type="checkbox"/>	
<u>1.2.5 – Audio Description (Pre-recorded)</u>	Users have access to audio description for video content		<input type="checkbox"/>	
<u>1.3.4 – Orientation</u>	Requires authors not to rely on a screen orientation		<input type="checkbox"/>	
<u>1.3.5 – Identify Input Purpose</u>	Ensure common names are provided using the HTML autocomplete list		<input type="checkbox"/>	
<u>1.4.3 – Contrast (Minimum)</u>	Contrast ratio between text and background is at least 4.5:1		<input type="checkbox"/>	
<u>1.4.4 – Resize Text</u>	Text can be resized to 200% without loss of content or function		<input type="checkbox"/>	
<u>1.4.5 – Images of Text</u>	Don't use images of text		<input type="checkbox"/>	
<u>1.4.10 – Reflow</u>	Your website must be responsive		<input type="checkbox"/>	
<u>1.4.11 – Non-Text Contrast</u>	High contrast between pieces of text and their backgrounds		<input type="checkbox"/>	
<u>1.4.12 – Text Spacing</u>	Text spacing can be overridden to improve the reading experience		<input type="checkbox"/>	
<u>1.4.13 – Content on Hover Focus</u>	Ensuring content visible on hover or keyboard focus does not lead to accessibility issues		<input type="checkbox"/>	
<u>2.4.5 – Multiple Ways</u>	Offer several ways to find pages		<input type="checkbox"/>	

WCAG 2.1 Level AA Checklist (continued)

Success Criteria	Description	Notes	Pass/Fail	Complete
<u>2.4.6 – Headings and Labels</u>	Use clear headings and labels		<input type="checkbox"/>	
<u>2.4.7 – Focus Visible</u>	Keyboard focus is visible and clear		<input type="checkbox"/>	
<u>3.1.2 – Language of Parts</u>	Tell users when the language on a page changes		<input type="checkbox"/>	
<u>3.2.3 – Consistent Navigation</u>	Use menus consistently		<input type="checkbox"/>	
<u>3.2.4 – Consistent Identification</u>	Use icons and buttons consistently		<input type="checkbox"/>	
<u>3.3.3 – Error Suggestion</u>	Suggest fixes when users make errors		<input type="checkbox"/>	
<u>3.3.4 – Error Prevention (Legal, Financial, Data)</u>	Reduce the risk of input errors for sensitive data		<input type="checkbox"/>	
<u>4.1.3 – Status Changes</u>	Distances between paragraphs, rows, words and characters must be able to be increased to a certain value		<input type="checkbox"/>	
Result			<input type="checkbox"/>	<input type="checkbox"/>

WCAG 2.1 Level AAA Checklist

Project:
Digital Asset:

Date:

Success Criteria	Description	Notes	Pass/Fail	Complete
1.2.6 – Sign Language (Pre-recorded)	Provide sign language translations for videos		<input type="checkbox"/>	
1.2.7 – Extend Audio Description (Pre-recorded)	Provide extended audio description for videos		<input type="checkbox"/>	
1.2.8 – Media Alternative (Pre-recorded)	Provide a text alternative to videos		<input type="checkbox"/>	
1.2.9 – Audio only (Live)	Provide alternatives for live audio		<input type="checkbox"/>	
1.3.6 – Identify Purpose	Anticipates the release of cognitive metadata to be used with assistive technology to simplify interfaces		<input type="checkbox"/>	
1.4.6 – Contrast (Enhanced)	Contrast ratio between text and background is at least 7:1		<input type="checkbox"/>	
1.4.7 – Low or No Background Audio	Audio is clear for listeners to hear		<input type="checkbox"/>	
1.4.8 – Visual Presentation	Offer users a range of presentation options		<input type="checkbox"/>	
1.4.9 – Images of Text (No Exception)	Don't use images of text		<input type="checkbox"/>	
2.1.3 – Keyboard (No Exception)	Accessible by keyboard only, without exception		<input type="checkbox"/>	
2.2.3 – No Timing	No time limits		<input type="checkbox"/>	
2.2.4 – Interruptions	Don't interrupt users		<input type="checkbox"/>	
2.2.5 – Re-authenticating	Save user data when re-authenticating		<input type="checkbox"/>	
2.2.6 – Timeouts	Users need to be warned of the duration of any inactivity that could cause data loss		<input type="checkbox"/>	

WCAG 2.1 Level AAA Checklist (continued)

Success Criteria	Description	Notes	Pass/Fail	Complete
2.3.2 – Three Flashes	No content flashes more than three times per second		<input type="checkbox"/>	
2.3.3 – Animation from Interaction	Motion animation triggered by interaction can be disabled		<input type="checkbox"/>	
2.4.8 – Location	Let users know where they are		<input type="checkbox"/>	
2.4.9 – Link Purpose (Link Only)	Every link's purpose is clear from its text		<input type="checkbox"/>	
2.4.10 – Section Headings	Break up content with headings		<input type="checkbox"/>	
2.5.5 – Target Size	The size of the target for pointer inputs is at least 44 x 44 CSS pixels		<input type="checkbox"/>	
2.5.6 – Concurrent Input Mechanisms	Web content does not restrict use of input modalities available on a platform		<input type="checkbox"/>	
3.1.3 – Unusual Words	Explain any strange words		<input type="checkbox"/>	
3.1.4 – Abbreviations	Explain any abbreviations		<input type="checkbox"/>	
3.1.5 – Reading Level	Users with nine years of school can read your content		<input type="checkbox"/>	
3.1.6 – Pronunciation	Explain any words that are hard to pronounce		<input type="checkbox"/>	
3.2.5 – Change on Request	Don't change elements until users ask		<input type="checkbox"/>	
3.3.5 – Help	Provide detailed help and instructions		<input type="checkbox"/>	
3.3.6 – Error Prevention (All)	Reduce the risk of all input errors		<input type="checkbox"/>	
Result			<input type="checkbox"/>	<input type="checkbox"/>

eSSENTIAL Accessibility is a comprehensive digital accessibility platform. We help organizations create inclusive web, mobile, and product experiences through digital accessibility testing, evaluation and remediation. Organizations can then enhance the digital experience for people with disabilities, comply with regulatory and statutory accessibility standards, and project an inclusive and disability-friendly presence.



To learn more about how you can make your web, mobile, and product experiences accessible, visit www.essentialaccessibility.com

Phone: 416-364-2501

Toll Free: 1-866-333-3909

Fax: 416-466-4452

Email: info@essentialaccessibility.com