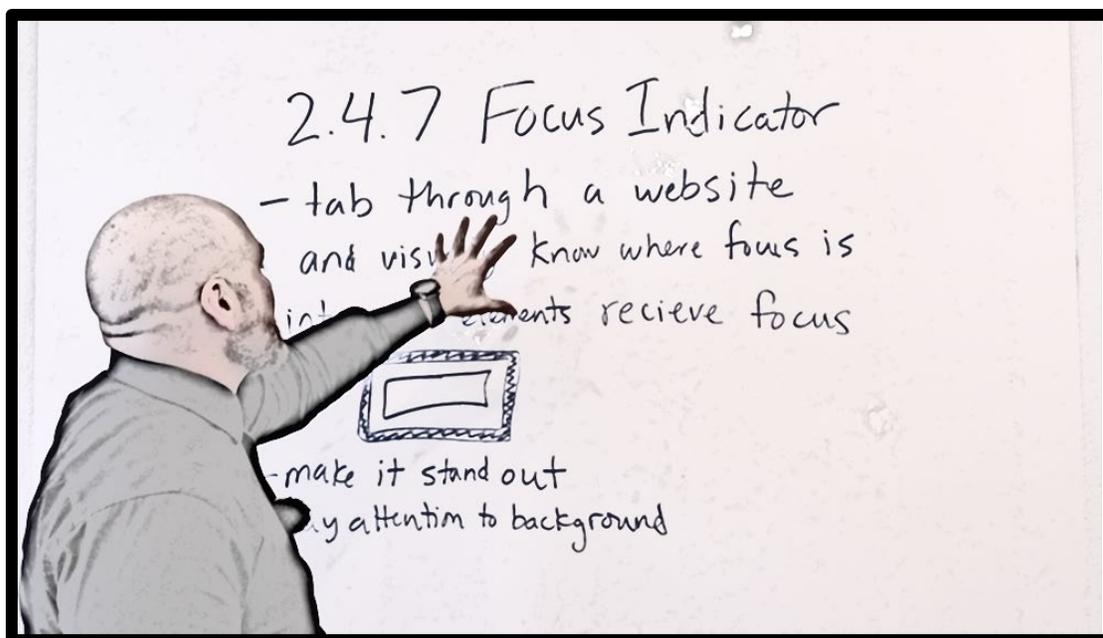# Kris's WCAG 2.1 AA Checklist

By Kris Rivenburgh

Learn WCAG in as Little As 3 Hours… Even if You're a Beginner

Want to learn WCAG faster? Why not go premium with my WCAG Course?

WCAG Course

Hire me to teach you the Web Content Accessibility Guidelines.

My WCAG Course comes with code examples, video explanations, accessibility workflows, WCAG in Chunks cheatsheet, and companion Excel spreadsheet checklist with workbooks for version 2.0 AA, 2.1 AA, and, 2.2 AA.

I keep stacking the course with more and more information and downloadable resources to make sure you save 20+ hours of time in learning and implementation.

Work smarter. Learn WCAG and seamlessly implement accessibility.

This course is for everyone – including beginners.

You can access the full course right now at WCAGCourse.com.

Table of Contents

Click a title to jump to any section

Version

This updated document is released as Version 1.2 of Kris's WCAG 2.1 AA Checklist and incorporates all of my previous writings.

The official publish date of this document is April 21, 2024.

Previous Version Information

This updated document is released as Version 1.1 of Kris's WCAG 2.1 AA Checklist and incorporates all of my previous writings.

The official publish date of this document is October 1, 2023.

Initial Version

This document is released as Version 1.0 of Kris's WCAG 2.1 AA Checklist and incorporates all of my previous writings.

The official publish date of this document is April 24, 2021.

Later versions of this document will be published (to make corrections, supplement and improve material, etc).

The latest, current version of this document can always be found at https://accessible.org and supersedes any previous versions. If a newer version is available, use of previous versions should be discontinued.

In Memory of Graham

Graham was an unexpected hospice adoption. Originally the plan was for him to be a foster but it was discovered that he was terminally ill with an expectation that he only had one to two months to live. Graham amazingly lived 10 months despite having stage 4 kidney disease, prostate cancer, severe arthritis, and other ailments.

Graham had lived most of his life in a cage but he was determined as I've ever seen any dog to have fun. He loved to rumble and wreak havoc. In the picture above, he was caught reveling in a turned over box of clothes that were to be donated.

Another time I caught him chewing on the dishwasher soap compartment for absolutely no reason. There was also a towel that hung off a kitchen island rack that he made of a tradition of yanking down to the ground, again with no reason other than he felt like it.

And one more Graham anecdote - one time he came across the remnants of a large discarded cardboard box and flopped over and rolled back and forth in the cardboard rubble and the assorted Styrofoam and plastic coverings.

He destroyed some stuff but only in the cutest of ways and never my shoes or clothes. I would have adopted Graham 10/10 times. He was one of the greatest dogs with the best personalities; he really and truly just loved to live life and it was written all over his face. I really do believe that's how he ended up living so long despite terminal illness and his age (he was a senior when adopted).

There are too many amazing senior dogs in shelters and rescues. One big plus to adopting a senior – or a hospice senior – is you're not signing up for a lifetime of commitment but you know you'll make that dog extremely happy and provide the best last part of their life for them.

Introduction

I highly recommend you read my brief [explainer](#) before you read my checklist.

After my explainer is a very quick and concise checklist to give you a solid idea of what each WCAG success criterion is asking for.

Update: I have added potential roles (content editor, designer, and developer) in parenthesis after each success criterion. This will help you in assigning each success criteria to different persons and/or departments. In the event I designated multiple roles to a success criterion, I have listed them in order of who potentially has the primary or most significant responsibilities. Note that these designations are approximate and may very based on your digital asset and roles and responsibilities within your organization.

My WCAG checklist is by no means comprehensive of each success criterion's requirements, nor is it intended to be. Additional explanation is needed to fully understand the essence of (and details and exceptions to) each success criterion.

Here's a loose illustration for context: For a given success criterion, I may have condensed 1,000 words into 25.

Think of my checklist as a cheatsheet you can study while cramming the night before the exam. The cheatsheet will help you score points on the most important concepts tested but you will be missing key information on the specifics.

For example, my cheatsheet tells you that closed captions are necessary but do you know the difference between closed captions and subtitles?

To get a full and complete understanding of each success criterion, I recommend you read the [W3C source documentation](#).

Further, my WCAG guide, Kris's WCAG 2.1 AA Guide, provides more detail and explanation for each success criterion and is available for free at https://accessible.org.

Also, you can purchase my [ADA Compliance Course](#) which includes a full WCAG section that acts as a simple, easy to understand supplement to the W3C official documentation.

Copyright

Kris's WCAG 2.1 AA Checklist is Copyright © 2021 - 2024 Accessible.org and Kris Rivenburgh. All rights reserved. You may not copy, edit, change, update, rename, sell / resell, supplement, add to, bundle, white label, private label, password protect, and/or take credit for this checklist in any way.

You may share this document so long as it remains completely free, does not require a membership or subscription to access, and remains unedited, unaltered, unbundled, and in its original form.

Acceptable means of sharing include:

- Posting a link to social media
- Posting a link on your website
- Uploading and linking to the original, unedited PDF on your website (e.g., yourwebsite.com/kris-checklist.pdf)
- Emailing a copy to your friend

No more than 24 words of this checklist may be copied, printed, republished, etc. in a magazine, newspaper, blog, or website – without permission in writing from the Author/Creator, Kris Rivenburgh.

Any quote or cite to content in this guide must be accompanied with an active hypertext link attribution to https://accessible.org.

If you would like to tell friends, family, followers, reads, and well-wishers about this guide, you can also share a link to https://accessible.org where they can download the latest version.

For information, please contact the author by email at kris@accessible.org or by mail at PO Box 791691, San Antonio, Texas 78279.

If you are aware of any copyright infringement, I would greatly appreciate you informing me of the person or organization illegally infringing upon my copyrighted work.

Attribution

I want to make three things very clear:

1) When written, this checklist, myself, my company (Kris's WCAG 2.1 AA Checklist, Kris Rivenburgh, and Accessible.org) are not affiliated or associated with W3C, WAI, WCAG, or any W3C initiatives in any way. I may participate in W3C activities at some point but this checklist is completely independent and separate of the W3C.
2) WCAG is the source of any material web accessibility and technical information that I present in this checklist. The W3C and WAI gets all credit for any material ideas, recommendations, guidelines, specifications, etc.
3) All I have done is condensed, rewritten, interpreted, and presented WCAG's material information in an informal, less technical manner.

Here is full attribution to W3C:

Copyright © 2017-2018 World Wide Web Consortium W3C® (MIT, ERCIM, Keio, Beihang). This software or document includes material copied from or derived from WCAG 2.1 as well as the informal Understanding WCAG 2.1 and Techniques for WCAG 2.0 documents. Source W3.org.

Accessibility

PDF/UA and WCAG 2.0 AA standards were followed in the creation of this document.

If you have any difficulty accessing the contents of this document, contact me at kris@accessible.org and I will ensure that you have full access to this checklist.

Disclaimer

By reading this document, you agree to release Author/Creator/Publisher, Kris Rivenburgh and Accessible.org, from any and all liability resulting from your use of this document.

Explainer

WCAG stands for the Web Content Accessibility Guidelines which is a set of technical standards to make digital assets (e.g., websites, apps, etc.) more accessible to persons with disabilities. WCAG is authored by the Web Accessibility Initiative (WAI) under the W3C.

There are currently three versions of WCAG (1.0, 2.0, 2.1) and three conformance levels (A, AA, AAA). Each conformance level includes the previous (i.e., AA includes all A success criteria). Another version, 2.2, is expected to be released in mid-2021.

Level AA conformance is all that is required for legal compliance with laws across the world.

Version 2.0 is best viewed as the classic version. It was published in 2008 and provides a strong baseline for accessibility.

Version 2.1 is the current version. It was published in 2018 and includes key mobile considerations. WCAG 2.1 AA conformance is highly recommended. Although 2.0 AA conformance is a very strong start, 2.1 AA conformance is even better.

Each version and conformance level is comprised of success criteria. Each success criterion is a requirement under WCAG. To be fully WCAG conformant, your digital asset must meet all success criteria under a given version and conformance level.

WCAG 2.1 AA includes all of the success criteria found in WCAG 2.0 AA with 12 new success criteria added in. Thus, conformance with WCAG 2.1 AA will result in conformance with WCAG 2.0 AA.

There are 38 success criteria in WCAG 2.0 AA.

There are 50 success criteria in WCAG 2.1 AA.

This checklist begins with the 38 WCAG 2.0 AA success criteria. These 38 success criteria are essential to creating a good foundation of accessibility for your digital asset.

The remaining 12 success criteria in WCAG 2.1 AA build on that foundation to make your asset's accessibility foundation even stronger and more robust, particularly on mobile devices.

WCAG 2.0 AA Checklist

Section 1: Alternatives

Alt text (1.1.1): All images and non-text content that conveys necessary meaning or information needs a text alternative. (Developer, Content Editor)

Video & Audio alternatives (1.2.1): All video-only and audio-only content has a text transcript. Transcripts are clearly labeled and available near the media. (Content Editor)

Closed captioning (1.2.2): All video with sound has closed captioning. (Content Editor)

Audio description (1.2.3): For any video where meaningful information is not available audibly, add a transcript or an audio description describing that information. (Content Editor)

Live captions (1.2.4): More formal, live presentations must have closed captions. (Content Editor)

Audio description (1.2.5): An audio description is optional under 1.2.3 level A but not in 1.2.5 AA. The best route is to provide both a transcript (optional) and an audio description but the audio description is required. (Content Editor)

Section 2: Presentation

Website structure (1.3.1): Use proper HTML markup techniques to structure your website's content so that meaning and information presented visually and audibly is available programmatically. (Developer, Content Editor)

Meaningful order (1.3.2): Make sure that content is ordered in a way that preserves meaning programmatically. (Developer)

Sensory characteristics (1.3.3): When providing detailed instructions, make it so they aren't reliant on a single sensory ability. (Content Editor)

Use of color (1.4.1): Do not rely on color alone to convey information. (Designer, Content Editor)

Audio control (1.4.2): Any audio must be able to be paused, stopped, or muted. (Developer)

Color contrast (1.4.3): There must be a color contrast ratio of at least 4.5:1 between regular text and background and 3:1 for large text. (Designer)

Text resize (1.4.4): Text must be able to be resized up to 200% without negatively affecting the ability to read content or use functions. (Designer)

Images of text (1.4.5): Do not use images of text unless necessary (e.g., logo). (Designer, Content Editor)

Section 3: User Control

Keyboard only (2.1.1): All functionality on a website must be accessible by keyboard only (i.e., no mouse). (Developer)

No keyboard trap (2.1.2): Keyboard-only users must never get stuck on any part of the website; they must be able to navigate forwards and backwards. (Developer)

Adjustable time (2.2.1): If there any time limits on a website, users have the ability to turn it off, adjust it, extend it. (Developer)

Pause, stop, hide (2.2.2): If there is content that blinks, scrolls, moves, users must have the ability to pause, stop, or hide it. (Designer, Developer, Content Editor)

Three flashes or below (2.3.1): Web pages do not contain anything that flashes more than three times in any one second period. (Designer, Content Editor)

Bypass Blocks (2.4.1): A "Skip to Content" link or other mechanism allows users to bypass repeated blocks of content (i.e., the header with menu) and go straight to the main content. Other mechanisms to bypass blocks include headings and landmarks. (Developer, Designer)

Section 4: Understandable

Page titles (2.4.2): Each page of a website needs to have a unique and descriptive page title. (Content Editor)

Focus order (2.4.3): Users must be able to navigate through focusable (interactive) elements in a logical, sequential order that preserves meaning. (Developer)

Link anchor text (2.4.4): The purpose of each link should be clear based on its anchor text (e.g., don't use "click here", etc.) and/or within programmatic context. (Content Editor)

Multiple ways (2.4.5): There are multiple ways to access content/pages on a website (e.g., search bar, nav menus, sitemap, breadcrumbs, etc.). (Designer)

Descriptive headings and labels (2.4.6): Headings and labels must be clear and descriptive. They do not need to be lengthy. (Content Editor)

Focus indicator (2.4.7): Any "user interface control" that receives focus from a keyboard user should have a focus indicator that indicates that focus visually. (Designer)

Website language (3.1.1): Set the language for your website. (Developer)

Language changes (3.1.2): Indicate any language changes for an entire page or within the content. (Content Editor)

Section 5: Predictability

No focus change (3.2.1): No context changes and nothing is activated merely because an item receives focus. (Developer, Designer)

No input change (3.2.2): User interface changes or actions shouldn't automatically occur due to input (e.g., form doesn't auto submit once all fields are filled out). (Developer)

Consistent navigation (3.2.3): Keep navigation links / layout consistent throughout all pages of the website (e.g., links remain in same relative order). (Designer)

Consistent identification (3.2.4): Components that have the same function within a website are identified consistently (but not necessarily identically). (Developer, Designer)

Error identification (3.3.1): Make any form errors easy to identify, understand, and correct. (Developer, Designer)

Form labels and instructions (3.3.2): Provide clear visual and programmatic labels or instructions for user input fields, ensuring users understand what information is required and how to input it. (Developer)

Error suggestions (3.3.3): If an input error is automatically detected, then suggestions for correcting the error should be provided. (Developer)

Error prevention on important forms (3.3.4): For pages that create legal commitments or financial transactions or any other important data submissions, one of the following is true: 1) submissions are reversible, 2) the user has an opportunity to correct errors, and 3) confirmation is available that allows an opportunity to review and correct before submission. (Developer)

Parsing (4.1.1): Make sure HTML code is clean and free of errors, particularly missing bracket closes. Also, make sure all HTML elements are properly nested. (Developer)

Name, role, value (4.1.2): For all user interface components (including forms, links, components generated by scripts), the name, role, state, and/or value should all be able to be programmatically determined and set by the user. (Developer)

WCAG 2.1 AA Checklist

Note: 2.1 AA includes all of the preceding 2.0 AA success criteria plus the success criteria below.

Orientation (1.3.4): Web pages and content does not lock on portrait or landscape mode, unless necessary. (Developer)

Input (1.3.5): The purpose of an input element can be determined so browsers and assistive technology can help guide and facilitate inputting information (e.g., provide autocomplete option). (Developer)

Reflow (1.4.10): Ensure someone can zoom in on your website without requiring scrolling or causing poor experience. (Designer)

Non-text contrast (1.4.11): All meaningful non-text content (e.g., buttons, form fields, icons, etc.) on your website should have a minimum 3:1 color contrast ratio to ensure they stand out. (Designer)

Text spacing (1.4.12): Make sure your text spacing can be adjusted without causing a poor experience. (Designer)

Content on hover or focus (1.4.13): Make it so any additional content (e.g., pop-ups, submenus) can be dismissed or remain visible if the user desires. (Designer, Developer)

Keyboard shortcuts (2.1.4): If you have a keyboard shortcut, make sure a user can either 1) turn it off, 2) there's a way to add another key in the shortcut, and/or 3) have the shortcut only active while focusing on a specific component. (Developer)

Pointer gestures (2.5.1): Provide simple alternatives (e.g., single tap vs. swipe) to potentially complex finger motions on touch screens. (Developer)

Pointer cancellation (2.5.2): Provide a way to cancel the trigger action when you activate a function with a "down-event" using a mouse or press /touch with your finger. (Developer)

Label in Name (2.5.3): Make sure any programmatic labels you make are aligned with the corresponding visual text. (Developer)

Motion Actuation (2.5.4): For any functions that are activated by motion, provide a simpler, alternative means of action. Also, give users the option to turn off motion activation. (Developer)

Status Messages (4.1.3): When a status message appears, it should be coded with role or properties so that people using assistive technologies (e.g., screen readers) are alerted without losing focus. (Developer)

WCAG Course

Would you like to save 20+ hours with WCAG and implementation?

My WCAG Course contains video explanations of not just WCAG 2.2 AA success criteria, but 2.0 AA and 2.1 AA.

The course also includes:

- Code examples
- Excel spreadsheet checklist
- Accessibility workflows based on roles
- WCAG in Chunks cheatsheet

Access everything you need right now at WCAGCourse.com.

ADA Compliance Course

Are you worried about an ADA website lawsuit?

Make it much, much less likely you're sued over website accessibility.

My ADA Compliance Course tells you what the most urgent accessibility issues are and how to find and fix them right now.

Access the ADA Compliance Course right now at ADACompliance.net.

Kris's Resources

Here are my channels and websites:

https://www.youtube.com/@adabook

https://www.linkedin.com/in/krisrivenburgh/

https://medium.com/@krisrivenburgh

https://krisrivenburgh.com

https://adabook.com

https://accessible.org

https://adacompliance.net

https://wcagcourse.com

https://audits101.com

If you have any questions, you are welcome to contact me at kris@accessible.org.

Thank you very much.