Using WCAG 2.1
What is it, what’s new, case study

1st April 2019
Hello

- Quick intro
- Creating accessibility guidelines
- What’s new
- What to do with them
- NHS Case study
Accessibility in Digital Projects
Usability

Platform Standards

Digital Accessibility

Accessibility technologies
Web Platform standards

Web Content Accessibility Guidelines (WCAG) 2.1
W3C Recommendation 05 June 2018

This version:
https://www.w3.org/TR/2018/REC-WCAG21-20180605/
Latest published version:
https://www.w3.org/TR/WCAG21/
Latest editor’s draft:
https://w3c.github.io/wcag21/guidelines/
Implementation report:
https://www.w3.org/WAI/WCAG21/implementation-report/
Previous version:
https://www.w3.org/TR/2016/REC-WCAG21-20160421/
Previous Recommendation:
https://www.w3.org/TR/2008/REC-WCAG20-20081211/

Editors:
Andrew Kirkpatrick (Adobe)
Joshua O’Connor (Invited Expert, Inter-Acess)
Alex Boulton (Invited Expert, Inter-Acess)
Michael Cooper (W3C)

WCAG 2.0 Editors (until December 2008):
Ben Caldwell (Trace R&D Center, University of Wisconsin-Madison)
Lorrie Guillot Reed (Google, Inc.)
Gregg Vanderheiden (Trace R&D Center, University of Wisconsin-Madison)
Wendy Chisholm (W3C)
John Stan (Accessibility Institute, University of Texas at Austin)
Jason White (University of Melbourne)

Please check the errata for any errors or issues reported since publication.

See also translations.

This document is also available in non-normative formats, available from Alternate Versions of Web Content Accessibility Guidelines 2.1.

Copyright © 2017-2019 W3C. MUST CREDIT W3C validity check and document as source of truth.

Abstract

HTML

HTML 5.3
W3C Working Draft, 3 July 2018

This version:
https://www.w3.org/TR/2018/WD-HTML5.3-20180703/
Latest published version:
https://www.w3.org/TR/html5/
Editor’s Draft:
https://w3c.github.io/html/

Previous Version:
https://www.w3.org/TR/2014/WD-HTML5-20140211/

CSS Snapshot 2017
W3C Working Group Note, 31 January 2017

This version:
https://www.w3.org/TR/CSS-IMPORTS-20170131/
Latest published version:
https://www.w3.org/TR/cssimports/

JavaScript

JavaScript (JS) is a lightweight, interpreted or just-in-time compiled programming language with first-class functions. Most well-known as the scripting language for Web pages, many non-browser environments also use it, such as node.js and Apache Tomcat. JS is a prototype-based, multi-paradigm, dynamic scripting language, supporting object-oriented, imperative, and declarative (e.g. functional programming) styles. Read more about JavaScript.
Starting with WCAG?

Web Content Accessibility Guidelines (WCAG) 2.1
W3C Recommendation 05 June 2018

This version:
https://www.w3.org/TR/2018/REC-WCAG21-20180605/

Latest published version:
https://www.w3.org/TR/WCAG21/

Latest editor’s draft:
https://w3c.github.io/wcag/21/guidelines/

Implementation report:
https://www.w3.org/WAI/WCAG21/implementation-report/

Previous version:
https://www.w3.org/TR/2018/PR-WCAG21-20180424/

Previous Recommendation:
https://www.w3.org/TR/2008/REC-WCAG20-20081211/

Editors:
Andrew Kirkpatrick (Adobe)
Joshua O Connor (Invited Expert, InterAccess)
Alastair Campbell (Nomensa)
Michael Cooper (W3C)

WCAG 2.0 Editors (until December 2008):
Ben Caldwell (Trace R&D Center, University of Wisconsin-Madison)
Loretta Guarino Reid (Google, Inc.)
Gregg Vanderheiden (Trace R&D Center, University of Wisconsin-Madison)
Avoid a checklist mentality
Define your accessibility approach early
Goal: Continuous improvement

Policy

Organisation's Accessibility Policy & Guidelines

Assets

Brand Guidelines
Design Patterns
Code Libraries

Projects

Discovery ➔ Low-Fi ➔ Design ➔ Development ➔ QA & Launch
How do we evaluate accessibility?

Approach | Designs | Development
Accessibility testing methods

Testing with people with disabilities
- Usability testing
- (Testing by experts as proxies)

Testing with rules
- Review pages with the guidelines
- Semi-automated review
- Automated review
Usability testing (with PwD)

Great for:
• Testing a live site / app
• Prioritising issues
• Learning
• Getting buy in

Not so good for:
• Site-wide, even in-page coverage
• Budgets
Accessibility testing methods

People

Guidelines

Semi-automated tools

With automated tools
Testing with guidelines

Great for...

- Per-page checks at later stages of development
- Coverage (with a good sample)
- Saying you’ve done your bit

Not so good for:

- In-experienced testers
Automated testing

Great for...

• Wide coverage of a site
• Integrating with a build process

Not great for:

• Depth of coverage
• Saying it meets the guidelines
Usability

Accessibility

Reliably Testable Accessibility

Automatically Testable Accessibility
What is Wuh-CAG?

Web CONTENT accessibility guidelines
World Wide Web Consortium (W3C)
Browser / User-Agent

Website/App
Creation process

- Applies across technologies
- Applies across types of site
- Doesn’t overlap
- Has (feasible) techniques to achieve
Guidelines are a baseline

Many user-requirements did not become criteria because they:
• could not be applied across all types of website;
• could not be tested reliably;
• would dictate the visual display of the website;
• needed solutions (across technologies) that were not reliable or feasible.

Therefore: There are probably requirements not covered by WCAG that may be relevant for your website.
What is a content guideline?

Unworkable:

“Users can navigate by headings and lists when appropriate.”

Difficult to understand, but works:

“Information, structure, and relationships conveyed through presentation can be programmatically determined or are available in text.”
WCAG 2.1: Principles, Guidelines, Success Criteria

**Success criteria:** A testable statement about any web content that is known to improve access for people with disabilities.
Levels: A, AA, AAA

Not meeting a SC means:

Level A (30): Critical, people will not be able to access the content.

Level AA (20): Essential, significant barriers would remain for some people.

Level AAA (28): Useful, possibly essential to some.

Not officially defined, in practice:

↑ Barrier that cannot be overcome by user’s technology

↓ Difficult or infeasible in practice

↓ Does not apply to all content

↓ Requires a change to look & feel for everyone
Structure of supporting info

Success Criteria

Understanding this Success Criteria

Technique

Sufficient

Technique

Technique

Technique

Advisory
Success Criterion 1.3.1 Info and Relationships

(Level A)

Information, structure, and relationships conveyed through presentation can be programmatically determined or are available in text.

Guideline 1.3 – Adaptable
Create content that can be presented in different ways (e.g., simpler layout) without losing information or structure.

1.3.1 Info and Relationships — Level A
Information, structure, and relationships conveyed through presentation can be programmatically determined or are available in text.

Sufficient Techniques
Note: Other techniques may also be sufficient if they meet the success criterion. See Understanding Techniques.

Situation A: The technology provides semantic structure to make information and relationships conveyed through presentation programmatically determinable:

- ARIA11: Using ARIA landmarks to identify regions of a page
- ARIA12: Using role-label to identify headings
- ARIA13: Using aria-labelledby to name regions and landmarks
- ARIA16: Using aria-labelledby to provide a name for user interface controls
- ARIA17: Using grouping roles to identify related form controls
- ARIA20: Using the region role to identify a region of the page
- G115: Using semantic elements to mark up structure AND H49: Using semantic markup to mark up non-essential text

On this page:
- Intent
- Benefits
- Examples
- Related Resources
- Techniques
WCAG 2.1

What’s new?
Why update WCAG?
espeakup connector

espeakup is a program which makes it possible for spekup to use the espeak software synthesizer. It does this by reading spekup’s softsynth device and passing the text to espeak which actually speaks it.

Requirements

This program works with the spekup screen reader, which can be obtained from http://linux-spekup.org, and the espeak software speech synthesizer which can be obtained from http://espeak.sourceforge.net. You must have both of these installed and operational. Setup is beyond the scope of this document.

Installation

The preferred way to install espeakup is using your distribution’s packaging system, but if your distribution does not have a
Buttons to touch screen
Solid to virtual
WCAG 2.1 timescale

NB: WCAG 2.0 is unchanged, and some policies will likely reference it for some time

Three task forces formed:
• Mobile
• Cognitive
• Low vision

~2013

User requirements

Feb 2017

1st Public working draft

Monthly Iterations

Jan 2018

Candidate Recommendation

June 2018

Recommendation (“Standard”)

Over 60 proposals for new criteria (A or AA), 17 made it
New criteria - Mobile

- **Orientation (AA):** Functionality is not locked to one orientation.
- **Character Key Shortcuts (A):** Single-key shortcuts can be turned-off or remapped.
- **Pointer Gestures (A):** Everything can be achieved without gestures.
- **Pointer cancelation (AA):** Don’t use the ‘down’ event.
- **Label in Name (A):** the programmatic name includes the visible name.
- **Motion Activation (A):** Only activate on the ‘up’ event.
- **Target Size (AAA):** controls are at least 44 x 44 pixels (with exceptions, AAA without exceptions).
- **Concurrent Input Mechanisms (AAA):** are not restricted.
Orientation

“Content does not restrict its view and operation to a single display orientation, such as portrait or landscape, unless a specific display orientation is essential.”
Character Key Shortcuts

Paraphrasing:

If the site implements keyboard short-cuts then restrict them to particular components; or let people turn them off or remap them.

Bob (a Dragon user) sits in front of gmail.
Their friend Kim walks in, Bob looks up.
“Hey Kim”, Dragon picks up:
Y = archive current message
K = move down one
M = mute message/thread

Bob looks at the screen, “Doh!”
Pointer Gestures

All functionality that uses multipoint or path-based gestures for operation can be operated with a single pointer without a path-based gesture, unless a multipoint or path-based gesture is essential.
Paraphrasing:

For functionality that is used with a click or tap, don’t use the ‘down’ event unless there is no other way.

Alice scrolled through an article on her phone, an advert in the page activates as soon as she touches it, even though she is trying to scroll.
Label in Name

“For user interface components with labels that include text or images of text, the name contains the text that is presented visually.”

Pass:

Fail:
Motion Actuation

Paraphrasing:

Features activated by device-motion also have buttons to use, and motion-based features can be disabled.
New criteria – Low vision

- **Zoom content**: Pages can be zoomed to 400%.
- **Adapting Text**: Allow some buffer for small text-adaptations.
- **Graphics contrast**: Is 3:1 for graphics that are ‘required for understanding’ and controls / inputs.
- **Content on Hover or Focus**: Does not obscure the trigger, and the pointer/focus can move into the new content.
Reflow - 400% zoom

“Content can be presented without loss of information or functionality, and without requiring scrolling in two dimensions for:

• Vertical scrolling content at a width equivalent to 320 CSS pixels;
• Horizontal scrolling content at a height equivalent to 256 CSS pixels.

Except for parts of the content which require two-dimensional layout for usage or meaning.”
Key questions remain unanswered in Kraft’s case

A Globe review of law-enforcement documents found a number of lines of inquiry that have received little public attention but almost certainly are being examined by Robert Kraft’s lawyers.

A new series talking with city business leaders about their careers and visions for Boston.

Presented by Koch

Joanne Chang is always on the move

The Globe sits down with leaders in the city's business community to talk about their career paths, work, and accomplishments, as well as their vision for Boston's future. First up is Joanne Chang of Flour...
In HTML/CSS users can override styles. Check that doing this doesn’t make things unreadable when you set:

- Line height to at least 1.5;
- Spacing following paragraphs to at least 2;
- Letter spacing (tracking) to at least 0.12;
- Word spacing to 0.16.
Testing layouts and text spacing

Combining three criteria in one test:

• Open Chrome / Edge / Firefox at 1280px wide, landscape
• Zoom in to 400% (i.e. 320px wide)
• **Check** the same/equivalent functionality is available
• **Check** that text is at least 200% bigger at some point
• Apply the text-spacing styles
• Gradually zoom out
• **Check** for overlapping / unreadable text
HOW ARE YOU?

Take our free quiz to get your health score, along with personalised advice and easy tips for healthier living.

TAKE THE QUIZ

THERE’S ONLY ONE YOU
Non-text contrast

The visual presentation of the following have a contrast ratio of at least 3:1 against adjacent color(s):

• **User Interface Components**: Visual information required to identify user interface components and states, except for inactive components or where the appearance of the component is determined by the user agent and not modified by the author;

• **Graphical Objects**: Parts of graphics required to understand the content, except when a particular presentation of graphics is essential to the information being conveyed.
User Interface Components (Interactive controls)

The aspects that define a control need reasonable (3:1) contrast.

Focus styles

Input styles
Graphical objects

WCAG 2.1 and support materials

- 1,398 commits
- 84 branches
- 68 releases
- 18 contributors

Branch: master
New pull request

<table>
<thead>
<tr>
<th>Folder</th>
<th>Description</th>
<th>Latest commit</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>css</td>
<td>CSS for all files</td>
<td>775484e</td>
<td>10 hours ago</td>
</tr>
<tr>
<td>guidelines</td>
<td>Editorial change</td>
<td></td>
<td></td>
</tr>
<tr>
<td>requirements</td>
<td>https a crap load of links to reduce link report noise</td>
<td></td>
<td></td>
</tr>
<tr>
<td>script</td>
<td>Merge pull request #737 from w3c/intro-edits</td>
<td></td>
<td></td>
</tr>
<tr>
<td>understanding</td>
<td>add lang and charset to understanding</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Essential – representing either a brand-logo or a real-life picture.

These can be ignored.
Not required for understanding – has text.

These can also be ignored.
<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>awkawk</td>
<td>Editorial change</td>
</tr>
<tr>
<td>css</td>
<td>CS to CSS</td>
</tr>
<tr>
<td>guidelines</td>
<td>Ed to guidelines</td>
</tr>
<tr>
<td>requirements</td>
<td>https a crap load of links to reduce link report noise</td>
</tr>
<tr>
<td>script</td>
<td>Merge pull request #737 from w3c/intro-edits</td>
</tr>
<tr>
<td>understanding</td>
<td>add lang and charset to understanding</td>
</tr>
</tbody>
</table>
- Light bell shape – 8.9:1
- Blue circle – 5.4:1

- Plus Shape – 8.9:1
- Down arrow – 8.9:1

- Down arrow – 8.9:1

- Down arrow – 13.5:1

- Down arrow – 2.4:1 (Fail)

- Three dots – 6.6:1

- Folder – 3:1
Responsive page variations

• BBC Homepage full width

• BBC Homepage smaller, more graphics relied on for understanding
Content on Hover or Focus
Content on Hover or Focus
<table>
<thead>
<tr>
<th>Furniture by Room</th>
<th>Furniture by Type</th>
<th>Lighting</th>
<th>Lighting By Room</th>
<th>Don't Miss</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bedroom</td>
<td>Beds</td>
<td>View all Lighting</td>
<td>Garden &amp; Outdoor Lighting</td>
<td>20% off John Lewis &amp; Partners floor lamps</td>
</tr>
<tr>
<td>Living Room</td>
<td>Mattresses</td>
<td>Ceiling Lighting</td>
<td>Bathroom Lighting</td>
<td>15% off selected designer lighting</td>
</tr>
<tr>
<td>Dining Room</td>
<td>Wardrobes</td>
<td>Wall Lighting</td>
<td>Kitchen Lighting</td>
<td>New in: Palazzo Collection</td>
</tr>
<tr>
<td>Kitchen</td>
<td>Bedside Tables</td>
<td>Desk &amp; Table Lamps</td>
<td>Children’s Lighting</td>
<td>New in: House by John Lewis</td>
</tr>
<tr>
<td>Bathroom</td>
<td>Chests of Drawers</td>
<td>Floor Lamps</td>
<td></td>
<td>New in: Sofas &amp; Armchairs</td>
</tr>
<tr>
<td>Children’s</td>
<td>Dressing Tables</td>
<td>Ceiling &amp; Lamp Shades</td>
<td></td>
<td>John Lewis &amp; Partners Natural Collection</td>
</tr>
<tr>
<td>Nursery</td>
<td>Headboards</td>
<td>Line, Fairy &amp; Novelty</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Home Office</td>
<td>Bookcases, Shelving</td>
<td>Lights</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hallway</td>
<td>Units &amp; Shelves</td>
<td>SAD &amp; Wake Up Lighting</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Garden</td>
<td>Office Desks</td>
<td>Cabinet Lighting</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Office Chairs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Furniture by Type</td>
<td>Lighting</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------------------------</td>
<td>------------------------------------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Beds</td>
<td>View all Lighting</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mattresses</td>
<td>Ceiling Lighting</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wardrobes</td>
<td>Wall Lighting</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bedside Tables</td>
<td>Desk &amp; Table Lamps</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chests of Drawers</td>
<td>Floor Lamps</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dressing Tables</td>
<td>Ceiling &amp; Lamp Shades</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Headboards</td>
<td>Line, Fairy &amp; Novelty</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
What’s new - Cognitive

- **Identify Input Purpose (AA):** can be identified.
- **Identify Purpose (AAA):** is added to controls, symbols & regions.
- **Timeouts (AAA):** If data might be lost due to timeouts, warn the user.
Identify Input Purpose

Started as ‘personalisation’, but changed to:

“The purpose of each input field collecting information about the user can be
programmatically determined when:

• The input field serves a purpose identified in the Input Purposes for User
  Interface Components section; and

• The content is implemented using technologies with support for identifying the
  expected meaning for form input data.”

<table>
<thead>
<tr>
<th>Token</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Full name</td>
</tr>
<tr>
<td>honorific-prefix</td>
<td>Prefix or title (Mr., Mrs. Dr., etc.)</td>
</tr>
<tr>
<td>given-name</td>
<td>Given or first name</td>
</tr>
<tr>
<td>additional-name</td>
<td>Additional or middle name</td>
</tr>
</tbody>
</table>
Autocomplete

Send us a message

Leave us a message and we will get back to you.

Your full name (required)

Alastair Campbell

Your email (required)

acampbell@nomensa.com

Email example: name@example.com

Your telephone number (required)

0117 929 7333

Phone number example: 01119999999

Your company

Nomensa
Other criteria

- **Animation from Interactions**: Allow users to turn off animations from scrolling (e.g. parallax)

- **Status Messages**: Notify users when something changes on the page (e.g. ARIA live)
Animations from interactions – Vestibular trigger!

Rayons ft. Predawn - Waxing Moon

Directed and produced the music video for Rayons ft. Predawn’s song, “Waxing Moon”.

Inspired by the meanings of "Rayons", which is "ray of light" in French and "Waxing Moon", which is the title of the song, we depicted a world at dusk.
Status updates

Where content is updated away from the focus, make sure it is ‘programmatically’ available.

Gov.uk Character count demo uses `aria-live` to enable this.
What to do?
0-60mph in accessibility

Guidelines are a **baseline** for creating accessible interfaces.

- Establish your current issues / barriers
- Training to ensure the team understand the core requirements
- Work out when and who should apply each
- Integrate with:
  - Brand guidelines, Pattern libraries (Design system), Documentation
  - Design, Development, Content
- Test early, little and often
- Test with people when possible
- Feed the results back into your artefacts and process
NHS Case study

Automated sweep
Accessibility Audit
Testing with people

Draft Recommendations
Stakeholder workshops / interviews
Team training / workshops

Proposals
Draft Materials
Support
Automated test:
• 22,500 pages
• 14,000 pages had at least one issue.
• 17 types of issue found

Audit:
• 30 pages
• 26 types of issue found

Testing with people:
• Mostly “usability” issues
• No new interface issues
Automated vs Manual

Focus based on results would be:

**Automated testing**
- Alt text
- Onclick handlers
- Incorrect labels
- Read more links
- Nesting of HTML markup
- Incorrect ARIA

**Manual testing**
- Heading structure & list markup
- Non-text contrast & focus styles
- Incorrect labels & fieldsets
- Missing ARIA
- Alt text
- Multimedia alternatives (audio desc)
- Read more links
Analysis of process & results, working out what activity should define each accessibility requirement.

Of 50 WCAG (AA) criteria:

Policy: 14
Graphic design: 17
Front-end code: 28
Content design: 13
Testers: 50
Guidelines vs Patterns

- Guidelines: Translating to ‘how we do things here’ is more efficient
- Patterns: Clarity & efficiency

Flexibility
Accessibility working group

Distributed function, central responsibility

• At least 1 person from each team
• Regular meetings (monthly?)
• Empathy lab
• Tackle issues, process, resources
• Report to a senior stakeholder
• Ensure new research feeds into resources
• Sponsor is point of escalation
Graphic design

Defining focus styles
(WCAG 2.4.7)

Users should be able to access all interactive components with a keyboard alone and it should be clear which element or link has focus.

Define focus styles to guide the user, to clearly identify their current position on the page. The browser default indicator is generally not good enough.

Good examples:

How to test:
Use a tool (link TBC) which calculates the ratio between the colour of the focus indicator and background colour it is replacing. The ratio should meet minimum of 3:1.

Make your service accessible

The NHS is for everyone, so everything we create must be accessible to everyone, considering different needs and abilities.

Expectations

Here’s what we expect all project teams to commit to at NHS Digital.

Consider accessibility at every stage

Think about how you are going to address accessibility at the beginning and at every stage of your project.

It’s much harder to make a service accessible if it’s only addressed late on.
Make guidelines your own – medical parallel

“When it was introduced without any programme or support, it was just impossible for teams to buy into it”
Psychologist & researcher Stephanie Russ

The creator of the checklists said that each hospital should customise it:
“They were 95% the same, but that 5% made it work for them. Every one of these hospitals thought that theirs was the best.”
Peter Pronovost, Johns Hopkins University
Inclusion of accessibility guidelines
Flyout Notice

Introduction

The flyout notice pattern is a system-activated flyout. It draws the user's attention to a specific element or region on the page.

A flyout notice is typically classified as either high or low priority.

An error that prevents an action or task is high-priority, anything else can be considered low priority.

Working Examples
Interaction Design

This section provides guidance for keyboard, screen reader and pointing devices.

Keyboard

Please refer to the flyout pattern for all inherited keyboard interactions.

Screen Reader

If high-priority, notice must be announced.

If high-priority, notice must be listed as a region/landmark.

Please refer to the flyout pattern for all inherited screen reader interactions.

Pointer

Please refer to the flyout pattern for all inherited pointer interactions.

Developer Guide

Developer guide coming soon.
Next steps

• Review the guidelines in the light of your people & processes
• Create your internal standard, with WCAG as core, and build on it
• Run usability testing, and feed that back in to your internal standard
• Join up and help with creating techniques for WCAG 2.1!
Key links

• WCAG 2.1: https://www.w3.org/TR/WCAG21/ (Don’t forget the ‘understanding’ links)
• Issues & comments: https://github.com/w3c/wcag/

• How to use guidelines: https://www.nomensa.com/blog/2017/web-accessibility-guidelines-and-how-use-them
• Overview of new guidelines: https://www.nomensa.com/blog/2018/wcag-21-accessibility-recommended
“The results of inclusive design for accessibility always leads to a better product for everyone.”

Head of Xbox, Phil Spencer
Questions?

@alastc