Accessibility Resources for Developers and Authors

Last Updated: 10/10/2017

U.S. Federal Government

Revised 508 Standards Transition Workgroup

Contents

[Executive Summary 1](#_Toc495565371)

[Applicability – What Standards Apply 2](#_Toc495565372)

[Understanding Conformance Requirements 3](#_Toc495565373)

[Testing 3](#_Toc495565374)

[What types of content need to be tested? 3](#_Toc495565375)

[What training is needed to successfully test for 508 conformance? 3](#_Toc495565376)

[How do the new Standards affect automated scanning or evaluation tools? 4](#_Toc495565377)

[Authoring Tool Requirements 5](#_Toc495565378)

[What is an “authoring tool”? 5](#_Toc495565379)

[Requirements for authoring tools 6](#_Toc495565380)

[Accessibility Training Resources for Developers 7](#_Toc495565381)

[General Accessibility Training 8](#_Toc495565382)

[International Association of Accessibility Professionals (IAAP) Certifications 8](#_Toc495565383)

[Teach Access 8](#_Toc495565384)

[Web Accessibility Training 8](#_Toc495565385)

[Original Section 508 Standards Training 8](#_Toc495565386)

[Developer Resources 9](#_Toc495565387)

[General Guidance on Developing Accessible Web Content 9](#_Toc495565388)

[Open Source Code and Frameworks 9](#_Toc495565389)

[Mobile Accessibility 10](#_Toc495565390)

[Software 10](#_Toc495565391)

[Development Environment Accessibility 10](#_Toc495565392)

[Testing Tools and Resources 10](#_Toc495565393)

[Other References 11](#_Toc495565394)

[Section 508 Legislation and Standards 11](#_Toc495565395)

[Section 508 Refresh Working Groups: 11](#_Toc495565396)

[W3C WCAG 2.0 References 11](#_Toc495565397)

[Captioning Tools and Resources 11](#_Toc495565398)

[Adobe Captivate 9 Authoring Guidance 12](#_Toc495565399)

[Acknowledgements 12](#_Toc495565400)

# Executive Summary

Section 508 of the Rehabilitation Act (29 U.S.C. § 794d), as amended by the Workforce Investment Act of 1998 (P.L. 105-220), implemented a set of standards for electronic and information technology (EIT) referred to (in this document) as the “Original 508 Standards.” The “Revised 508 Standards” were published in the Federal register in January of 2017. The Revised 508 Standards amended terminology, using the more modern term “Information and Communication Technology” (ICT) instead of “EIT”.

The Authoring and Developer Transition Working Group was established with contributions from the Federal CIO Council’s Accessibility Community of Practice (ACoP) and the U.S. Access Board to provide technical assistance for authors and developers implementing the Revised 508 Standards.

This document provides authors and developers a starting point for understanding how the Revised 508 Standards apply to electronic documents, software applications, web content, operating systems, accessibility platforms, assistive technology, mobile environments, and modern development frameworks. Various [working groups](#_Section_508_Refresh) of the Transition Support Initiative plan to develop more in-depth guidance and resources over time. Guidance for developers of hardware devices is not included in this document but may be added in a later edition. We encourage you to check back from time to time for additional resources.

This document will help you to:

* Locate developer-oriented accessibility training;
* Understand how the Revised 508 Standards apply to electronic documents, software applications, web content, operating systems, accessibility platforms, assistive technology, mobile environments, and modern development frameworks;
* Engage with active working groups and use existing Federal Accessibility Community of Practice (ACOP) best practices resources;
* Review and prioritize highlights of W3C, WCAG, and ARIA authoring best practices;
* Use information and references for automated tools, manual testing, and assistive technology-based testing;
* Examine best practices for mobile applications and web environments; and
* Implement popular development frameworks.

The working group recognizes that there are many sources of accessibility information. In this document, we attempt to provide current best practices supported by peer review and consensus among working group participants. We listed just a single resource only when we could not identify multiple resources. Inclusion in this document does not imply endorsement by the authors or the Federal government and does not mean that other equally valid or useful resources do not exist.

# Applicability – What Standards Apply

One significant change from the Original 508 Standards in the Revised 508 Standards will affect how you do your job. The Revised 508 Standards (36 C.F.R. Part 1194) incorporate by reference the WCAG 2.0 Level AA Success Criteria. The Revised 508 Standards apply the WCAG 2.0 Level AA success criteria and conformance requirements to both Web and non-Web electronic content.

Chapter 2 of the Revised 508 Standards specifies which electronic content must be accessible – we call this scoping. In this chapter, the technical requirements should be applied based on the type of content an agency communicates. The scoping requirements specifically for electronic content (Section E205) apply to non-Web electronic documents regardless of formats (e.g., Microsoft Office, Portable Document Format, and HTML Web pages). Other scoping provisions apply to non-Web software, e.g., spreadsheet or video conferencing applications, and native mobile “apps”. The technical requirements eliminate the distinction between traditional “Web pages” and “non-Web” content because the same accessibility needs exist for all electronic content. See the [Instructions - Revised 508 Standards Applicability Checklist](https://section508.gov/revised-508-standards-applicability-checklist) for a fully detailed applicability worksheet.

In addition to applying WCAG 2.0 to Web and non-Web content, the Revised 508 Standards now include specific requirements for authoring tools (see Section 504). While these requirements may not initially affect authors and developers, we expect over time that the capability of authoring tools such as word processors, spreadsheets, and presentation creation products, will decrease the level of effort needed to create fully accessible electronic content. Authoring tools also include those used to develop web pages or applications and the Integrated Development Environments (IDE) used by computer programmers for software development.

Agencies that have not previously used the [harmonized test methods](https://section508.gov/refresh-toolkit/test) published by the ACOP may have to significantly revise their policies and procedures to implement and verify conformance to the Revised 508 Standards. There will be less change for agencies already using these or similarly comprehensive test methods. The test processes published by the ACOP already incorporate most of the Revised 508 Standards because they incorporated WCAG 2.0 Level AA early in their development. The most important change for authors is that the revised standards have clarified which electronic documents must conform to the technical requirements in the Revised 508 Standards. Refer to Planning Organizational Support for Accessible Electronic Content for more information.

* Electronic documents must meet the W3C WCAG 2.0, Level A and AA guidelines (with four exceptions); see 36 CFR 1194 [E205 electronic Content, and E205.4 Accessibility Standards](https://www.access-board.gov/guidelines-and-standards/communications-and-it/about-the-ict-refresh/final-rule/single-file-version#E205-content)
* For Web content, see [E205, E205.2 and E205.3](https://www.access-board.gov/guidelines-and-standards/communications-and-it/about-the-ict-refresh/final-rule/single-file-version#E205-content)
* For hardware, see [E206](https://www.access-board.gov/guidelines-and-standards/communications-and-it/about-the-ict-refresh/final-rule/single-file-version#E206-hardware) and Chapter 5, [501](https://www.access-board.gov/guidelines-and-standards/communications-and-it/about-the-ict-refresh/final-rule/single-file-version#501-general)
* For software, see [E207, E207.2 and E207.3](https://www.access-board.gov/guidelines-and-standards/communications-and-it/about-the-ict-refresh/final-rule/single-file-version#E207-software)
* For authoring tools, see Chapter 5, [504](https://www.access-board.gov/guidelines-and-standards/communications-and-it/about-the-ict-refresh/final-rule/single-file-version#504-authoring-tools)
* For functional performance criteria, see [E204](https://www.access-board.gov/guidelines-and-standards/communications-and-it/about-the-ict-refresh/final-rule/single-file-version#E204-functional-performance-criteria) and Chapter 3, [301](https://www.access-board.gov/guidelines-and-standards/communications-and-it/about-the-ict-refresh/final-rule/single-file-version#301-general) and [302](https://www.access-board.gov/guidelines-and-standards/communications-and-it/about-the-ict-refresh/final-rule/single-file-version#302-functional-performance-criteria)

## Understanding Conformance Requirements

As an author or developer, you should know that electronic documents, software and websites covered by the Revised 508 Standards must meet the WCAG 2.0 Level A and AA success criteria”. However, it is important to assess the extent to which your work conforms to WCAG 2.0 Level AA. A page that fails to meet even one of the 38 applicable WCAG success criteria does not conform to the standards. A set of pages in a sequence, e.g., identifying, selecting, and paying for a ticket to a public event, does not conform if any of those steps fails to conform fully.

In some situations, you can comply with the Revised 508 Standards by providing a conforming alternate version of content. However, you should be aware that Sections [E101.2](https://www.access-board.gov/guidelines-and-standards/communications-and-it/about-the-ict-refresh/final-rule/single-file-version#E101-general), [E205](https://www.access-board.gov/guidelines-and-standards/communications-and-it/about-the-ict-refresh/final-rule/single-file-version#E205-content) and [E207](https://www.access-board.gov/guidelines-and-standards/communications-and-it/about-the-ict-refresh/final-rule/single-file-version#E207-software) of the Revised 508 Standards impose significant constraints on this option. For additional information on understanding conformance, including the requirements for conforming versions, see [Understanding WCAG 2.0 Conformance](http://www.w3.org/TR/UNDERSTANDING-WCAG20/conformance.html).

If you use interagency authoring or test processes, most of what you need for WCAG 2.0 Level AA conformance is satisfied by those processes. The WCAG 2.0 standards and accompanying guidance are a rich resource for web developers and testers. Understanding them requires some study before they are useful to the casual reader. Conformance determination rule sets will likely be more understandable and useful to you.

The ACOP publishes test guidelines for Web & Applications, Microsoft Word, Microsoft Excel, Microsoft PowerPoint, and PDF Documents, located in the [Testing Resources Section](https://section508.gov/refresh-toolkit/test) of the [Revised 508 Standards Toolkit](https://section508.gov/refresh-toolkit). These resources are designed to produce repeatable results and harmonize federal test practices across Federal agencies. The best way to meet the Revised 508 Standards is to use these resources. These rule sets establish a minimum set of requirements and agencies are encouraged to improve upon them.

## Testing

### What types of content need to be tested?

All public facing electronic content, including application and interactive content, must meet the Revised 508 Standards. All internally facing electronic content in at least one of the nine categories of “Agency Official Communications” found in Section [E205](https://www.access-board.gov/guidelines-and-standards/communications-and-it/about-the-ict-refresh/final-rule/single-file-version#E205-content) must be accessible.

### What training is needed to successfully test for 508 conformance?

To determine the type and the extent of training you will need, first assess how accessibility is addressed in the rule sets currently in use by your Agency. If your development and authoring best practices and test rules do not currently address accessibility, it will take you longer to learn to test content for accessibility than it would if you were merely updating current practices aimed at meeting the Original 508 Standards. This additional time can be minimized by adopting the harmonized test processes published by the ACOP. These test processes already incorporate most of the Revised 508 Standards.

### How do the new Standards affect automated scanning or evaluation tools?

#### General considerations for automated testing

Automated scanning tools for Section 508 or WCAG conformance have existed for a long time; automated testing for WCAG 2.0 conformance is not new. There are several types of automated tools, including those that are workstation-based, server-based, and hybrids. Workstation-based tools are generally lower cost and useful for small volumes of electronic content, or for small development group use. Server-based tools are generally higher cost but good for larger organizations with large amounts of electronic content. They can also help dispersed development groups to more effectively share conformance test results. Hybrid automated tools can be integrated into existing development or authoring workflows using application programming interfaces (APIs). Such integration of automated evaluation has great potential but requires significant upfront planning and implementation work to be successful.

Automated evaluation tools reduce but rarely eliminate the need for manual accessibility testing, because human judgment is often required to determine conformance. While automated tools are helpful when used properly, improper use can result in false assurances of conformance or, at a minimum, confusion among authors and developers.

If you plan to use automated evaluation tools, validate them against the Revised 508 Standards to ensure they evaluate content correctly and provide consistent results. Address results which are incorrect, inconsistent, or incomplete during implementation. Ask for validation data from prospective automated tool sources. Decide how automated test rules can be modified if you find inaccuracies, and determine how you will maintain your test rules.

Understand whether automated tools test pure HTML, or test the HTML as it appears on the rendered page in the browser document object model (DOM). Inspect the DOM so you know that the tool is evaluating against what end-users will experience. Understand also how automated test results will be used (with associated manual test results, as periodic bulk rough measures of conformance, to track progress towards a goal, or for some other purpose)? If using automated tools to locate trouble spots, you need a consistent way to identify and rank the severity of failures and a reasonable way to aggregate such information for reporting.

#### When should you use automated tools, and when should you manually test for conformance?

Automated tools cannot compare various options for remediation against one another, or even determine if the accessibility error must be remediated.

When purely yes/no answers are required, even if they are a complicated set of true/false logical conditions, automated tools can do well. For example, automated tools can easily identify if an image element has an alt attribute, and possibly if the text content in the alt attribute is insufficient (e.g., a filename). However, is not possible today to use automated tools to determine if an image is decorative, or if the text in the alt attribute appropriately describes the information in the image. Such an evaluation requires an understanding of the content, and how best to describe the content in context. Automated testing tools are no substitute for human judgment.

The following resources explain the testing algorithms used by two automated scanning tools (Tenion.io and FAE) and an integrated solution (aXe):

* [What Tenon Tests](https://tenon.io/documentation/what-tenon-tests.php)
* [Functional Accessibility Evaluator 2.0](https://fae.disability.illinois.edu/rulesets/wcag/)
* Deque’s [aXe-core](https://github.com/dequelabs/axe-core)

#### Automated testing

You may need to revalidate your testing results, since the WCAG 2.0 Level AA algorithms used by the automated testing tools differ somewhat from those used for the Original 508 Standards (which were largely based on WCAG 1.0). If using automated Section 508 evaluation tools for Microsoft SharePoint sites, carefully consider how the automated rules work with SharePoint, to avoid false negative or positive results.

## Authoring Tool Requirements

### What is an “authoring tool”?

The Revised 508 Standards include new requirements for authoring tools. From the Definitions section (E103.4) of the Revised 508 Standards:

***Authoring Tool:*** *any software, or collection of software components, that can be used by authors, alone or collaboratively, to create or modify content for use by others, including other authors.*

Examples include electronic document authoring tools such as word processors, presentation creation tools, and video editing tools. Software that is used to develop software is an authoring tool. A content management system (CMS) used for an internal or public facing website is an authoring tool. Collaboration software and wikis are authoring tools. Conferencing and meeting software might include an authoring tool in its feature set.

Why is accessibility important for authoring tools? When the author of an electronic document, web page, or software application is unable to include accessibility from the start of the creative process, special (and often-unsuccessful) workarounds must be employed to make the output accessible (i.e., meet the standards). Such workarounds are usually found by trial and error, rarely are documented, and usually not shared within the authoring or developer community. Without accessibility support in the authoring or development environment, the responsibility for ensuring accessibility often falls to specialized subject matter experts. In the worst case, accessibility simply may not be addressed because authors and developers are unaware of their responsibilities and the resources available to meet those responsibilities.

The Revised 508 Standards include an exception when working in a raw text format.

***EXCEPTION:*** *Authoring tools shall not be required to conform to 504.2 when used to directly edit plain text source code.*

An application such as Notepad, which can be used as an authoring tool, meets the Revised 508 Standards, because it only functions as an editor of plain text files.

### Requirements for authoring tools

In general, authoring tools produce electronic content which include a very wide variety of information types. However, electronic content is specifically covered by the W3C WCAG Level A and Level AA Success Criteria and Conformance Requirements. Therefore, the basic requirement is that authoring tools allow the author to produce output conforming to WCAG Level A and Level AA Success Criteria and Conformance Requirements.

From the Section 504 of the Revised 508 Standards:

* ***504.2 Content Creation or Editing.*** *Authoring tools shall provide a mode of operation to create or edit content that conforms to Level A and Level AA Success Criteria and Conformance Requirements in WCAG 2.0 (incorporated by reference, see 702.10.1) for all supported features and, as applicable, to file formats supported by the authoring tool. Authoring tools shall permit authors the option of overriding information required for accessibility.*
* ***504.2.1 Preservation of Information Provided for Accessibility in Format Conversion.*** *Authoring tools shall, when converting content from one format to another or saving content in multiple formats, preserve the information required for accessibility to the extent that the information is supported by the destination format.*
* ***504.2.2 PDF Export.*** *Authoring tools capable of exporting PDF files that conform to ISO 32000-1:2008 (PDF 1.7) shall also be capable of exporting PDF files that conform to ANSI/AIIM/ISO 14289-1:2016 (PDF/UA-1) (incorporated by reference, see 702.3.1).*

The requirement above ensures that you don’t lose accessibility information when converting from one format to another. Migrating from one format to another when the source is fully accessible does not mean it will be accessible when converted. When tools do provide a “save to PDF” feature they should be capable of saving specifically to accessible PDF.

* **504.3 Prompts.** Authoring tools shall provide a mode of operation that prompts authors to create content that conforms to Level A and Level AA Success Criteria and Conformance Requirements in WCAG 2.0 (incorporated by reference, see 702.10.1) for supported features and, as applicable, to file formats supported by the authoring tool.

The requirement above ensures that authoring tools help authors produce accessible content. For example, when an authoring tool accepts images, the tool should prompt authors (at some point) to add alternative text. Similarly, authoring tools that allow drag and drop form fields could ask the author what the name of the field is, whether the field is required, and if any extra instructions are needed in the form field for people using assistive technology.

* **504.4 Templates.** Where templates are provided, templates allowing content creation that conforms to Level A and Level AA Success Criteria and Conformance Requirements in WCAG 2.0 (incorporated by reference, see 702.10.1) shall be provided for a range of template uses for supported features and, as applicable, to file formats supported by the authoring tool.

The requirement above ensures that those authoring tools that come with templates come with some that are accessible. Authoring tools (as well as development tools) frequently are bundled with “starter documents” for everything from business letters to helpdesk applications. Traditionally, such templates have not included accessibility information, even when the authoring tools could be used to create accessible documents. Consequently, anyone using those by default would create documents which were not accessible.

When purchasing or selecting software, ask the vendor if the software lets users create or edit content, or if the software saves into PDF format. If so, the software is probably an authoring tool and you must determine whether the software meets the requirements specified in Section 504 of the Standards. You can ask the vendor, test the tools, or both. You can also request the vendor to provide reference output with at least one example of every type of interface element. The reference output might be a simple PDF report, or a complicated application or website. With the reference output, you can then test the interface elements the tool will create, to see if the software is conformant (or not). Reference outputs are a great way to evaluate development platforms, especially where authors are constrained to only the available authoring options.

Clear authoring guidelines for tools can be a good indication that those tools can produce accessible outputs. See [Accessibility Testing for Electronic Content](https://section508.gov/refresh-toolkit/test) for examples.

# Accessibility Training Resources for Developers

The sections below contain both general accessibility training, and resources specifically for authors and developers. General training will help to establish a solid understanding of accessibility, which the additional resources can improve understanding of how to achieve accessibility in certain specialized environments. Some are available online for free, and others are commercial offerings. Unfortunately, there is no single set of accessibility-related raining resources available to date for developers. This document gives you several places to start, but your development environment will drive your specific training needs.

## General Accessibility Training

### International Association of Accessibility Professionals (IAAP) Certifications

The [International Association of Accessibility Professionals (IAAP)](http://www.accessibilityassociation.org/), a division of the [Global Initiative for Inclusive ICT (G3ICT)](http://www.g3ict.org/), offers two accessibility certifications:

* [Certified Professional in Accessibility Core Competency (CPACC)](http://www.accessibilityassociation.org/cpacccertification)
* [Web Accessibility Specialist (WAS)](http://www.accessibilityassociation.org/wascertification)

Any individual who completes both certifications also receives the designation of Certified Professional in Web Accessibility (CPWA).

The [WAS content outline](http://www.accessibilityassociation.org/content.asp?admin=Y&contentid=459), [CPACC Body of Knowledge](http://www.accessibilityassociation.org/content.asp?contentid=356), and [WAS Body of Knowledge](http://iaap.membershipsoftware.org/files/IAAP%20WAS%20BOK%202017_083017.doc) provide great guides for learning about accessibility.

* [Approved training providers for IAAP](http://www.accessibilityassociation.org/certificationproviders)
* [IAAP certifications](http://www.accessibilityassociation.org/certificationproviders)

### Teach Access

[Teach Access](http://teachaccess.org/) is an initiative by industry, academia, and accessibility advocates to expand the quality and quantity of undergraduate programs that teach the accessibility fundamentals in fields such as design, computer science, and human computer interaction. The initiative has established a core set of Accessibility Fundamental Concepts and Skills on web accessibility, federal accessibility laws and industry best practices, along with an industry guest speakers program and online accessibility tutorials, to prepare designers, engineers, and researchers to build products and services inclusively.

* [Teach Access Tutorial](http://teachaccess.org/initiatives/tutorial/) - learn about mobile app and website accessibility

## Web Accessibility Training

Here are some great overviews and introductions:

* Rob Dodson from Google has a [great introductory talk](https://www.youtube.com/watch?v=A5XzoDT37iM)
* Google’s [Introduction to Web Accessibility](https://webaccessibility.withgoogle.com/) – introduces tools and techniques for web developers to easily ensure that websites are more accessible
* [Web Accessibility | Udacity](https://www.udacity.com/course/web-accessibility--ud891) – get hands-on experience making web applications accessible
* [The Code4Lib Journal](http://journal.code4lib.org/articles/12697) – A Practical Guide on Developing Accessible Websites
* [Making Microsoft Office Documents Accessible](https://support.office.com/en-us/article/Make-your-Word-documents-accessible-d9bf3683-87ac-47ea-b91a-78dcacb3c66d)

### Original Section 508 Standards Training

A number of organizations developed training on the Original 508 Standards, including:

* [Department of Homeland Security (DHS), Trusted Tester Certification](https://www.dhs.gov/publication/trusted-tester-resources)
* [Department of Veterans Affairs (VA) - Creating Accessible Flash](https://www.section508.va.gov/support/flash/)
* [The Accessibility Community of Practice (ACOP) Best Practices Library](https://section508.gov/best-practices)
* [The Section 508 Universe](https://www.gsa.gov/portal/content/101182)

# Developer Resources

## General Guidance on Developing Accessible Web Content

The Sufficient Techniques outlined in the W3C’s [How to Meet WCAG 2.0](https://www.w3.org/WAI/WCAG20/quickref/) quick reference guide provide numerous examples of how to meet Success Criteria using specific technologies (relevant to HTML, CSS, server-side scripting, Flash, ARIA, etc.). [See the full list of sufficient techniques and common failures](https://www.w3.org/TR/WCAG20-TECHS/). Some other useful resources provided by W3C include:

* [Guidance on getting started with web accessibility](http://www.w3.org/WAI/gettingstarted/)
* [Guidance on designing for inclusion](http://www.w3.org/WAI/users/)
* [Guidance on selecting web accessibility evaluation tools](http://www.w3.org/WAI/eval/selectingtools)
* [Guidance on applying WCAG 2.0 to non-web content](http://www.w3.org/TR/wcag2ict/)
* [WCAG tutorials](http://www.w3.org/WAI/train)

Other sites also provide authoring guidelines and techniques for meeting WCAG Success Criteria, including:

* The University of Washington’s [IT Accessibility Checklist](http://www.washington.edu/accessibility/checklist/)
* [The Code4Lib Journal](http://journal.code4lib.org/articles/12697) – A Practical Guide on Developing Accessible Websites
* [Web Accessibility In Mind, (WebAIM)](http://webaim.org/)

### Open Source Code and Frameworks

Using open source is attractive, but comes with additional requirements to make the content accessible. The website Medium provides a great article about [*Making Open-Source Accessible for All*](https://medium.com/@kaelig/making-open-source-accessible-for-all-8131429913b1). An article on opensourse.com, [*Why Open Source Needs Accessibility Standards*](https://opensource.com/life/15/2/why-open-source-needs-accessibility-standards), also highlights the responsibility for developers to make open-source code accessible.

The [Open Source Accessibility initiative (OSAi)](https://www.ow2.org/bin/view/OSAi/) advocates for open and compliant digital solutions for people with physical or cognitive difficulties among the larger open-source community, and promotes accessibility in open-source projects.

#### JavaScript Frameworks

Contributors to these JavaScript frameworks actively consider accessibility in development:

* [JQuery](https://www.nomensa.com/blog/2010/accessible-forms-using-the-jquery-validation-plug-in/) - the “grandfather” of JavaScript frameworks; this article specifically addresses how to create accessible forms
* [React.js](https://facebook.github.io/react/docs/accessibility.html) - web application development framework from Facebook
* [Angular.js](https://docs.angularjs.org/guide/accessibility) - web development framework from Google
* [Bootstrap.js](https://paypal.github.io/bootstrap-accessibility-plugin/demo.html) - JavaScript development framework from Twitter
* [ext.js](https://docs.sencha.com/extjs/6.0.2/guides/core_concepts/accessibility.html) - JavaScript development framework
* [ally.js](https://allyjs.io/) - JavaScript library to make accessibility simpler for modern web applications

## Mobile Accessibility

* [BBC Mobile Accessibility Guidelines](http://www.bbc.co.uk/guidelines/futuremedia/accessibility/mobile_access.shtml)
* [Apple Accessibility Programing Guide for IOS](https://developer.apple.com/library/content/documentation/UserExperience/Conceptual/iPhoneAccessibility/Introduction/Introduction.html)
* [Google Accessibility for Android Developers](https://developer.android.com/guide/topics/ui/accessibility/index.html)

## Software

* [Apple Accessibility programming Guide for OSX](https://developer.apple.com/library/content/documentation/Accessibility/Conceptual/AccessibilityMacOSX/index.html)
* [Introduction to Microsoft UI Automation](https://www.youtube.com/watch?v=6b0K2883rXA)
* [Microsoft UI Automation Overview](https://docs.microsoft.com/en-us/dotnet/framework/ui-automation/ui-automation-overview)
* [Oracle Product Accessibility Guidance](http://www.oracle.com/us/corporate/accessibility/products/index.html)

## Development Environment Accessibility

* [Microsoft Visual Studio Accessibility Information](https://msdn.microsoft.com/en-us/library/bkxh4340.aspx)
* [Eclipse.org Eclipse Accessibility Information](https://help.eclipse.org/mars/index.jsp?topic=%2Forg.eclipse.platform.doc.user%2Fconcepts%2Faccessibility%2Faccessmain.htm)

## Testing Tools and Resources

Neither the Federal government nor the authors of this document endorse any of the products referenced here, or the companies that produce these products. These are merely examples of various kinds of tools for accessibility testing. For more options, refer to a [list published by the W3C](https://www.w3.org/WAI/ER/tools/) (note, the ACoP is not responsible for the content or accuracy of that list).

* [University of Illinois Functional Accessibility Evaluator 2.0](https://fae.disability.illinois.edu/)
* [React A11y Testing](https://github.com/reactjs/react-a11y) on GitHub
* [pa11y.org](http://pa11y.org/)
* Deque Systems – [aXe Core](https://www.deque.com/products/axe/)
  + [Accessibility Testing with aXe](https://marcysutton.github.io/a11y-testing-with-axe/) - a presentation on web accessibility testing with aXe, including browser extensions and automated testing tools.
  + [Deque Systems WorldSpace ATest](https://www.deque.com/products/worldspace-attest/)
* [Tenon.io](https://tenon.io/)
* [Level Access AMP For Mobile](https://www.levelaccess.com/amp-for-mobile-is-here/)
* [IBM Ability Lab Mobile Accessibility Checker](http://www-03.ibm.com/able/mobile-accessibility-checker.html)
* For Apple OSX and IOS:
  + [Accessibility Inspector](https://developer.apple.com/library/content/documentation/Accessibility/Conceptual/AccessibilityMacOSX/OSXAXTestingApps.html) - Apple Developer
  + [Testing and Debugging in Simulator](https://developer.apple.com/library/content/documentation/IDEs/Conceptual/iOS_Simulator_Guide/TestingontheiOSSimulator/TestingontheiOSSimulator.html) - Apple Developer
* [The CommonLook set of tools](http://commonlook.com/accessibility-software/) – provide additional accessibility evaluation and remediation capabilities for PDF documents

# Other References

## Section 508 Legislation and Standards

* [Section 508 of the Rehabilitation Act](https://www.gpo.gov/fdsys/pkg/USCODE-2011-title29/html/USCODE-2011-title29-chap16-subchapV-sec794d.htm) (29 U.S.C. § 794d), as amended by the Workforce Investment Act of 1998 (P.L. 105-220)
* [Original Section 508 Standards](https://www.access-board.gov/guidelines-and-standards/communications-and-it/about-the-section-508-standards/section-508-standards)
* [Revised 508 Standards](https://www.access-board.gov/guidelines-and-standards/communications-and-it/about-the-ict-refresh/final-rule/single-file-version) (revised January 2017, effective January 2018)
* [Revised 508 Standards Functional Performance Criteria](https://www.access-board.gov/guidelines-and-standards/communications-and-it/about-the-ict-refresh/final-rule/text-of-the-standards-and-guidelines#E204-functional-performance-criteria)

## Section 508 Refresh Working Groups:

* [Section508.gov Refresh Toolkit](https://section508.gov/refresh-toolkit)
* [Section 508 Transition Support Initiative](https://section508.gov/content/section-508-transition-support-initiative)

## W3C WCAG 2.0 References

The W3C is an internationally recognized web standards body that identifies its approved technical specification standards as “W3C Recommendations” (such as HTML, CSS, etc.). The consortium has several Accessibility specifications that have achieved W3C Recommendation status, including WCAG, ATAG, and WAI-ARIA. Other accessibility-related W3C recommendations – such as the User Agent Accessibility Guidelines are also relevant.

* [An Introduction to Understanding WCAG 2.0](https://www.w3.org/TR/UNDERSTANDING-WCAG20/intro.html), (Mar. 17, 2016)
* [Understanding WCAG 2.0](http://www.w3.org/TR/UNDERSTANDING-WCAG20/conformance.html)
* [WCAG Quick Reference](https://www.w3.org/WAI/WCAG20/quickref/)
* [WAI-ARIA Authoring Best Practices](http://w3c.github.io/aria-practices/)
* [W3C Authoring Tool Accessibility Guidelines](http://www.w3.org/TR/ATAG20/)
* [W3C User Agent Accessibility Guidelines](http://www.w3.org/TR/UAAG20/)
* [SVG Accessibility Mappings](http://www.w3.org/TR/svg-aam-1.0/)

## Captioning Tools and Resources

Courtesy of <https://www.section508.va.gov/support/resources_508.html>

* [Adding Captions, Annotations and/or Subtitles to Presentations](http://www.pptmagic.com/articles/captions.htm)
* [Automatic Sync Technologies (AST), CaptionSync](http://www.automaticsync.com/captionsync/about_us/static/captioning-flash-video-with-captionsync/)
* [Caption Colorado](http://www.captioncolorado.com/)
* [Captionate](http://www.buraks.com/captionate/)
* [Captioning Styles and Conventions](http://main.wgbh.org/wgbh/pages/mag/services/captioning/faq/sugg-styles-conv-faq.html)
* [CPC Closed Captioning and Subtitling Software and Services](http://www.cpcweb.com/)
* [MAGpie](http://ncam.wgbh.org/invent_build/web_multimedia/tools-guidelines/magpie/)
* [Media Access Group at WGBH](http://main.wgbh.org/wgbh/access/access.html)
* [National Captioning Institute](http://www.ncicap.org/)
* [National Center for Accessible Media (NCAM) tools](http://ncam.wgbh.org/invent_build/web_multimedia/tools-guidelines/)
* [Subtitle-horse](http://www.subtitle-horse.com/)
* [WEBAIM captioning guide](http://www.webaim.org/techniques/captions/)

## Adobe Captivate 9 Authoring Guidance

[Adobe Captivate 9 Accessibility Authoring Guidance](https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=0ahUKEwjV4oSA7qLWAhUMTbwKHUOrDbMQFggoMAA&url=https%3A%2F%2Fapps.kennesaw.edu%2Ffiles%2Fpr_app_uni_cdoc%2Fdoc%2FAdobe_Captivate_9_Accessibility.pdf&usg=AFQjCNGRJdgUYgBiXOoE9d_K0sgP7WmraA)

# Acknowledgements

The authors and developers Section 508 transition members are:

* Allen Hoffman, Department of Homeland Security
* Pierce Crowell, Social Security Administration
* Kathy Eng, Department of Justice
* Rebecca Lovely, Federal Communications Commission
* Ken Bennett, Federal Bureau of Investigation
* David Fanning, Veterans Administration
* John Beatty, Nuclear Regulatory Commission
* Nick Bristo, General Services Administration
* David Stenger, General Services Administration
* Bruce Bailey, U.S. Access Board