

Automated Testing

Overview

This page will discuss two popular automated web accessibility checkers. Before diving into the information about these checkers, be sure to pay close attention to the discussion of their limitations.

Limitations of Automated Testing

Automated accessibility checkers are a must in your Web Accessibility Auditing Toolkit, though it is important to understand their limitations. Think of an automated accessibility checker like a spell checker in a word processor. Though a good start for identifying misspelled words, a person must still read through the text to ensure words have been used correctly (e.g. where "there" is used in place of "their"). For now, **human judgement must also be involved** for any potential barriers that involve assessing meaning. For example, automated checkers can identify ambiguous phrases like "click here" or "this link" used as link text, but a person needs to determine whether this text accurately describes the link's destination or function. Similarly, a person must decide whether alt text or a long description for an image accurately describes the meaningful information in the image, something automated checker cannot currently do.

Can you think of other instances where human judgement is required to assess the full extent of a barrier in Web content?

Another limitation worth noting is that automated checkers are unlikely to identify with certainty whether accessible equivalents are available for Web content that has been flagged as a potential barrier. A human perspective is required to make the association between equivalent elements. For example, an automated checker may identify a flash object (see some **Flash examples**), such as a **matching game** embedded in a page as a barrier.

However, the site provider may offer an accessible HTML version of the game as well. An automated checker would not recognize the connection between the barrier and its accessible alternative, but a person would.

Web Accessibility Checkers

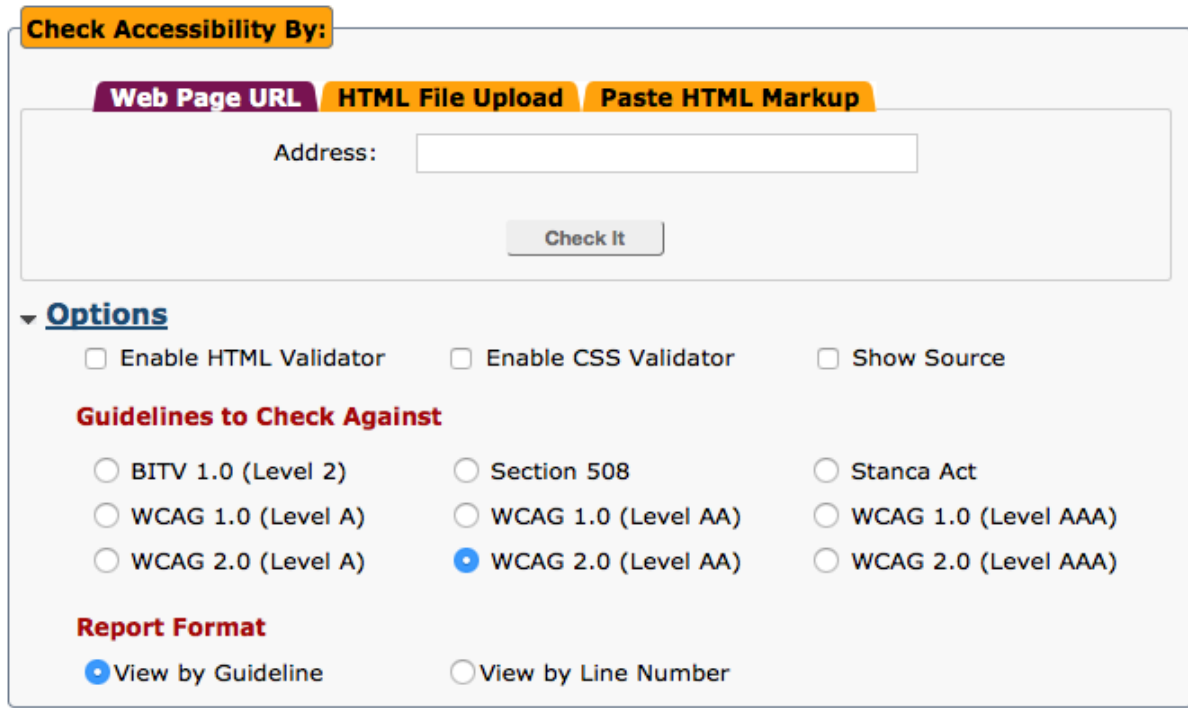
AChecker Web Accessibility Checker

First released in 2005, AChecker was created with the goal of providing an accessibility checking tool that was 100% transparent, interactive, customizable, and free. AChecker makes use of Open Accessibility Checks (OAC), which is a collection of checks based on all web accessibility guidelines available globally. Currently, there are a total of 310 OAC checks employed by AChecker.

Add the public version of **AChecker Web Accessibility Checker** to your browser. In the public version, you can set up an account and do your automated accessibility testing. See below for details on downloading and installing your own version.

AChecker Features

AChecker has specific features for public users, registered users, administrators, and developers. To take advantage of these features, you should first create an account on the public AChecker site, if you are not planning to install a version of your own (see below). Follow the link you added to your Toolkit above, and click "*Register*" to create an account. Creating an account will allow you to save your accessibility reviews, and generate an AChecker seal for sites that pass its review.



The screenshot shows the AChecker web interface. At the top, there's a section titled "Check Accessibility By:" with three tabs: "Web Page URL" (selected), "HTML File Upload", and "Paste HTML Markup". Below the tabs is a text input field labeled "Address:" and a "Check It" button. Underneath is a section titled "Options" with three checkboxes: "Enable HTML Validator", "Enable CSS Validator", and "Show Source". Below that is a section titled "Guidelines to Check Against" with nine radio button options: "BITV 1.0 (Level 2)", "Section 508", "Stanca Act", "WCAG 1.0 (Level A)", "WCAG 1.0 (Level AA)", "WCAG 1.0 (Level AAA)", "WCAG 2.0 (Level A)", "WCAG 2.0 (Level AA)" (which is selected), and "WCAG 2.0 (Level AAA)". At the bottom is a section titled "Report Format" with two radio button options: "View by Guideline" (selected) and "View by Line Number".

Figure: AChecker screenshot showing the main interface for conducting accessibility reviews

For more about using AChecker, watch the following video:

[Using AChecker to Test Web Accessibility \(8:52\)](#) [↗](#)



Set Up Your Own AChecker

[Download AChecker](#) [↗](#)

Installation Instructions

1. Unzip the achecker.tar.gz file into a php enabled, web accessible directory.
2. Open the installer in your browsers at
`http://[yourserver.com]/AChecker/install`
3. Follow the instructions provided by the installer.

Installing from GitHub or If You Plan to Contribute

If you are familiar with using GitHub, you can clone the most current source code from there. This version often has new features not available yet in the public site, or in the downloadable version of the software, though it may be less stable than the publicly distributed version. If you would like to participate in AChecker's development, or you would like to add your own accessibility checks, or perhaps fix a bug you've found, working from GitHub is the way to have your work added to the public source code.

AChecker Source Code

WAVE Accessibility Evaluation Tool

Another popular free accessibility checker is WAVE, developed by WebAIM at the Center for Persons with Disabilities at Utah State University. It is a Web service, similar to AChecker, though without much of the interactivity and customizability. Those who prefer a visual presentation of the issues, as opposed to the list presentation of AChecker, may find WAVE easier to use.

WAVE Features

WAVE is similar to AChecker in the following respects:

- WAVE can take a URL and assess the page it leads to
- It evaluates one page at a time

WAVE produces a report by reproducing the page that was reviewed, inserting a variety of icons into the content to identify errors (known problems) and alerts (potential problems), as well as the accessibility features that are present. Clicking on any of the icons will provide a brief description and a link to additional information. See the video below for details on how WAVE works.

Add the **WAVE Chrome Extension** to your browser.

WAVE Toolbar

WAVE Toolbar (4:44)

Resources

- **Accessibility Bookmarklets** [↗](#) - Adding these bookmarklets to your browser can help visually identify the presence of accessibility features. They allow you to check not only if they are present, but if they are used correctly by allowing you to analyze the HTML markup.
- **WAVE Help** [↗](#)
- Other Accessibility Checkers:
 - **AllInspector Sidebar for Firefox** [↗](#)
 - **Cynthia Says**
 - **FAE (Functional Accessibility Evaluator)**
 - **Chrome Accessibility Developer Tools**
 - **aXe (Chrome plugin)**
 - **aXe (Firefox plugin)**

Content from "Limitations of Automated Web Accessibility Checkers" and "WAVE Accessibility Evaluation Tool" and "AChecker Web Accessibility Checker" by [Greg Gay of Ryerson University, Change School of Continuing Education](#) [↗](#), is licensed under [CC BY-NC-SA 4.0](#) [↗](#)