

# Vergleich automatischer Web-Checker

Zum Untersuchen der Barrierefreiheit einer Website gibt es die Möglichkeit mit Tools zu arbeiten, die die Website automatisch auf Verstöße gegen die Barrierefreiheit untersuchen. Solche Tools können erste Ansätze liefern, in welchen Bereichen es zu Problemen kommt, aber man muss immer auch manuell überprüfen. Dies ist ein Vergleich von vier unterschiedlichen automatischen Web-Checkern: **WAVE (Chrome Extension/Online Version), AChecker, Functional Accessibility Evaluator 2.1 und Siteimprove (Chrome Extension)**

1) Vorstellung der 4 Tools.....	2
1.1. WAVE.....	2
1.2. AChecker.....	2
1.3. Functional Accessibility Evaluator 2.1 .....	3
1.4. Siteimprove Accessibility Checker Chrome Extension .....	3
2) Welche Regeln zum Prüfen werden von den Tools jeweils angeboten?.....	4
3) Wie viele Probleme fand das Tool jeweils, und wie steht das in Relation zu den Ergebnissen der manuellen Prüfung (vorherige Aufgaben)?.....	4
4) Was sind Ihre jeweiligen Erfahrungen mit den Tools? Was fanden Sie hilfreich, was nicht? War es schwierig, das Tool zu benutzen, oder hatten Sie Schwierigkeiten? Wenn ja, welche? .....	4
5) Was sind jeweils die Stärken und Schwächen der Tools? Für welches Anwendungsszenario ist das Tool jeweils besonders gut geeignet?.....	6
6) Relation zur manuellen Prüfung und Schlussfolgerung.....	7

# 1) Vorstellung der 4 Tools

## 1.1. WAVE

Evaluations Tool zur Erkennung von Problemen bei der Barrierefreiheit von einzelnen Webseiten, entweder als Online Version oder als Browser-Extension für Chrome und Firefox.

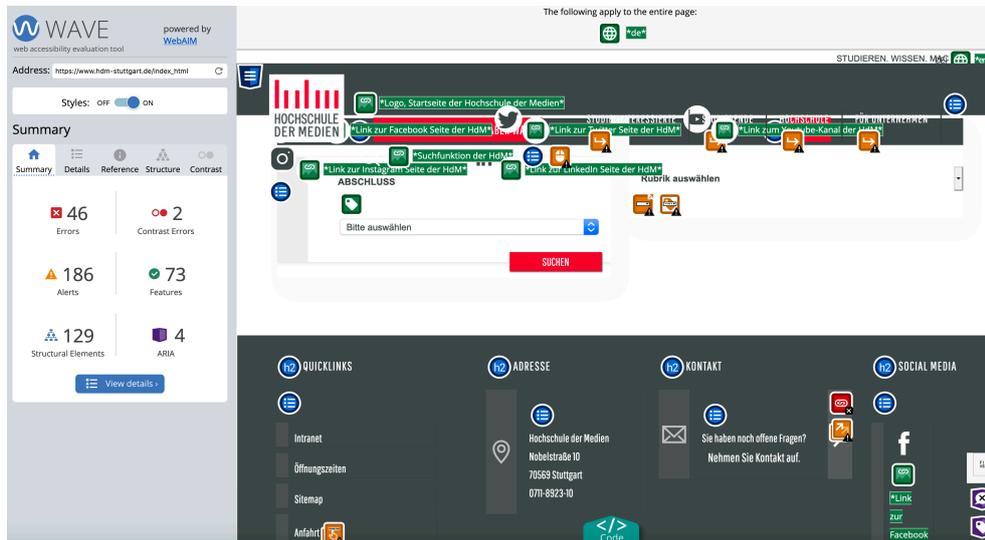


Abb. 1: Wave Web Accessibility Evaluation Tool (Online version)

## 1.2. AChecker

Dieses Tool überprüft einzelne HTML-Seiten, um Konformität mit Standards zur Barrierefreiheit zu gewährleisten. Man kann eine URL eingeben, html-Dateien uploaden oder html direkt eingeben, und sogar nur einen Teil HTML-Code checken lassen.

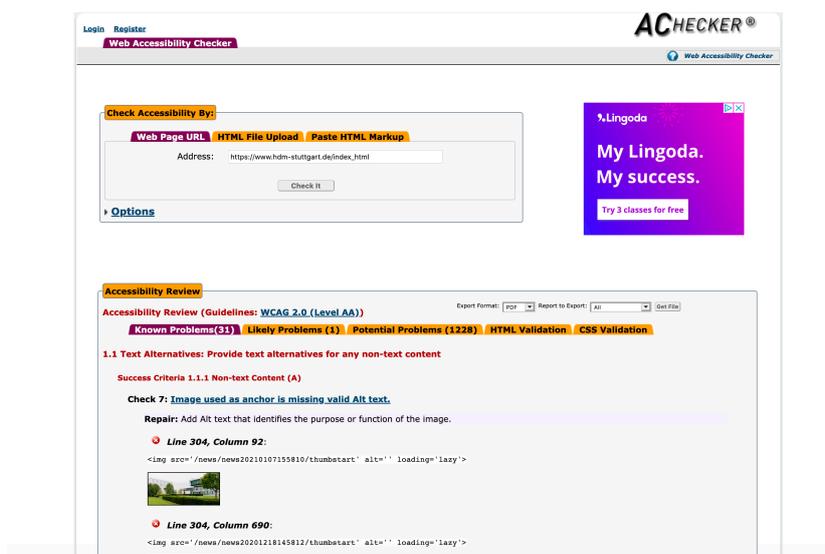


Abb. 2: AChecker – Web Accessibility Checker

### 1.3. Functional Accessibility Evaluator 2.1

Der Functional Accessibility Evaluator (FAE) evaluiert eine Website oder eine einzelne Webseiten basierend auf WCAG 2.0 Level A und AA. Man muss einen kostenlosen Nutzer-Account erstellen, um das Tool zu nutzen.

Functional Accessibility Evaluator 2.1

Page Title: Hochschule der Medien - Startseite - Studieren. Wissen. Machen. URL: https://www.hdm-stuttgart.de/index.html

### Page Summary

	Violations	Warnings	Manual Checks	Passed
Number of Rules	4	3	36	16

Rule Group	Number of Rules					Implementation Level	
	V	W	MC	P	N/A	Score	Status
Landmarks	1	-	4	5	9	58	PI-MC
Headings	-	2	-	3	3	74	PI
Styles/Content	1	-	8	1	14	24	NI-MC
Images	-	-	5	2	-	39	NI-MC
Links	1	1	1	-	1	19	NI-MC
Tables	-	-	-	-	9	-	U
Forms	1	-	4	5	9	60	PI-MC
Widgets/Scripts	-	-	2	-	12	47	NI-MC
Audio/Video	-	-	1	-	12	0	MC
Keyboard	-	-	4	-	2	2	NI-MC

Abb. 3: Functional Accessibility Evaluator 2.1

### 1.4. Siteimprove Accessibility Checker Chrome Extension

Mit der Chrome Extension von Siteimprove kann man einzelne Webseiten auf Barrierefreiheit checken. Mit der Vollversion von Siteimprove, die jedoch sehr viel kostet, kann man ganze Websites überprüfen lassen.

Siteimprove Accessibility Checker

Issues:

- Non-text Content (27 occurrences)
- Info and Relationships (57 occurrences)
- Use of Color (7 occurrences)
- Contrast (Minimum) (1 occurrence)
- Images of Text (67 occurrences)
- Timing Adjustable (1 occurrence)
- Bypass Blocks (1 occurrence)
- Focus Order (112 occurrences)

Abb. 4: Siteimprove Chrome Extension

## 2) Welche Regeln zum Prüfen werden von den Tools jeweils angeboten?

- 1) **WAVE**: Keine Auswahlmöglichkeit, wird nach WCAG 2.1 Level A + AA, Section 508 geprüft
- 2) **AChecker**: Auswahlmöglichkeit bei den Guidelines: BITV 1.0 (Level 2), Section 508, Stanca Act, WCAG 1.0 (Level A – AAA), WCAG 2.0 (Level A – AAA)
- 3) **Functional Accessibility Evaluator 2.1**: Keine Auswahlmöglichkeit, wird nach WCAG 2.0 geprüft
- 4) **Siteimprove Accessibility Checker**: Auswahlmöglichkeit bei den Filtern zum Testen zwischen WCAG 2 Level A, AA und AAA, Auswahl der Gewichtigkeit: Fehler, Warnungen und Review (sollte manuell gecheckt werden), Auswahl der Verantwortung: Editor, Webmaster, Entwickler.

## 3) Wie viele Probleme fand das Tool jeweils, und wie steht das in Relation zu den Ergebnissen der manuellen Prüfung (vorherige Aufgaben)?

- 1) **WAVE**: 47 Errors, 188 Alerts, 1 Contrast Error
- 2) **AChecker (Geprüft mit WCAG 2.0 Level AA)**: Known Problems 31, Likely Problems 1, Potential Problems 1228
- 3) **Functional Accessibility Evaluator 2.1**: Violations 4, Warnings 3, Manual checks 36, Passed 16
- 4) **Siteimprove Chrome Extension**: Errors ca. 70, Warnings 28, Manual checks ca. 218

Relation zu den Ergebnissen [s. Punkt 6\)](#)

## 4) Was sind Ihre jeweiligen Erfahrungen mit den Tools? Was fanden Sie hilfreich, was nicht? War es schwierig, das Tool zu benutzen, oder hatten Sie Schwierigkeiten? Wenn ja, welche?

- 1) **Wave** ist recht übersichtlich und recht gut designt, und wenn man sich eine Weile damit beschäftigt hat, gut lesbar. Es gibt Symbole, die anzeigen, welcher Art der

Fehler/Hinweis ist, bei Mouseover wird ein Rahmen um den betreffenden Bereich gelegt und bei Klick darauf bekommt man mehr Informationen: Um was es geht und Links zur Referenz und der Möglichkeit den Code einzublenden.

Auf der linken Seite ist eine Übersicht über alles, dort kann man zum Beispiel direkt zur Referenz springen und bekommt einiges an Informationen. Man kann dort auch per Klick zu einem Problem im Haupt-Screen springen.

Alles in allem ein gutes und leicht zu bedienendes Tool für den Start der Prüfung einer Seite.

Man muss englisch können, da es ein englisch-sprachiges Tool ist.

- 2) Der **AChecker** ist grafisch weniger ansprechend gemacht. Es dauert eine Weile, bis das Ergebnis der Prüfung bereit steht, wenn man eine Website-URL eingibt. Es wird eine ausführliche Ergebnis-Liste mit den Fehlern ausgegeben, aufgeteilt in Known problems, likely problems und potential problems und es gibt in PopUps immer nähere Infos dazu, was falsch ist und wie es gelöst werden kann. Was auch sehr gut ist, sind Code-Beispiele, die zeigen, was richtig und falsch ist, vor allem für Nutzer die sich mit html nicht so gut auskennen.

Anstatt grafisch zu zeigen, wo der Fehler ist, bekommt man beim AChecker die Angabe der Zeilennummer, was es etwas schwieriger macht den Fehler zu finden. Das ist nicht so gut gelöst, aber man findet die entsprechenden Stellen. Natürlich muss man wissen, wie man sich den Quelltext anzeigen lassen kann.

Es gibt die Möglichkeit, sich das Ergebnis in verschiedene Formate exportieren zu lassen: pdf, EARL, CSV, HTML, das ist vorteilhaft, wenn man die Liste nach und nach abarbeiten möchte. Man auch einstellen, was man exportieren möchte.

Es gibt den AChecker auch auf deutsch – aber das ist eine automatische Übersetzung und teilweise nicht verständlich. Deshalb ist es am besten, man lässt die Einstellung auf englisch.

- 3) Der **Functional Accessibility Evaluator 2.1** ist zunächst schwierig zu verstehen. Es gibt auf den ersten Blick keine Struktur, nach der die Ergebnisse gegliedert sind. Man erhält zwar viele und umfangreiche Informationen, weiß aber gar nicht so recht, was man jetzt damit machen soll, bzw. wo man beginnen soll. Es gibt die Möglichkeit, sich das Ergebnis als csv exportieren zu lassen, oder per E-Mail zusenden zu lassen.

Wenn man sich das Tool eine Weile genauer angesehen hat, wird es langsam klarer, und die farbliche Darstellung von Violation, Manual Check und Passed ist sehr hilfreich. Auf der Detailansicht bekommt man weitere Hinweise zum Fehler und wo er auf der Website zu finden ist, aber hier fehlt eine grafische Darstellung, bzw. die Website selbst und man muss anhand der „Element Positionen“ herausfinden wo das Element auf der Website zu finden ist.

Die Hilfe, die man auf der Detailseite bekommt, ist gut und verständlich. Hier findet man dann auch einen Link zu WCAG, sowie weitere Links.

Man muss englisch können, da es ein englisch-sprachiges Tool ist.

- 4) **Siteimprove Chrome Extension** fand ich am einfachsten zu benutzen, weil ich damit schon öfters gearbeitet habe. Ich finde es sehr übersichtlich, da es gut nach den WCAG Prüfschritten gegliedert ist und auch die Stellen markiert, an denen Probleme auftreten, sowie Links zu entsprechenden Artikeln bereitstellt, die zur Lösung der Probleme hilfreich sind. Auch hier gibt es eine direkte Anzeige auf der Website, wo das Problem ist, wie bei Wave. Dieses Feature ist allerdings im Vergleich zu Wave etwas versteckt und erst sichtbar, wenn man sich links in der Liste zur entsprechenden Detailansicht durch geklickt hat. Die Einstellmöglichkeiten und Filter sind sehr gut. Außerdem ist die Hilfe zu jedem Detailpunkt auch gut gelöst, man findet dort immer einen Link, der lautet „How do I meet this criteria?“
- Man muss englisch können, da es die Erweiterung ein englisch-sprachiges Tool ist.

## 5) Was sind jeweils die Stärken und Schwächen der Tools? Für welches Anwendungsszenario ist das Tool jeweils besonders gut geeignet?

- 1) **Wave** ist gut für einen schnellen ersten Test geeignet, und außerdem für Benutzer, die HTML/CSS Kenntnisse haben, um die Ergebnisse auch beurteilen zu können. Die Stärken von Wave liegen in der guten Oberfläche und der einfachen Verständlichkeit, sowie in der Tatsache, dass man auf der Website-Oberfläche sehen kann, wo der Fehler ist. Die Hilfestellung ist gut. Ich habe keinen Unterschied zwischen der Online-Version und der Chrome-Extension gesehen.
- 2) **AChecker** eignet sich dann, wenn man nicht mit einem Plugin bzw. einer Extension arbeiten und sich die Ergebnisse exportieren lassen möchte. Man kann eine URL eingeben, html-Files uploaden oder html direkt eingeben, und sogar nur einen Teil HTML-Code checken lassen. Gerade letzteres ist ein guter Service. Die Code-Beispiele sind hilfreich für Benutzer, die sich mit HTML nicht so gut auskennen. Der Nachteil des Tools ist die Unübersichtlichkeit auf den ersten Blick, es erschließt sich aber nach und nach, wenn man sich einige Zeit mit dem Tool und dem Thema beschäftigt. Man sollte sich mit der Barrierefreiheit für Websites schon etwas besser auskennen, wenn man das Tool benutzt. Die fehlende grafische Darstellung der Website macht es etwas schwieriger, die Fehler zu finden. Die Hilfestellung und weiterführenden Links sind gut.
- 3) **Functional Accessibility Evaluator 2.1** eignet sich auch, wenn man nicht mit einem Plugin oder einer Extension arbeiten und sich die Ergebnisse exportieren lassen möchte. Um damit zu arbeiten, muss man sich einen kostenlosen Nutzer-Account anlegen. Man hat so immer Zugriff auf das Ergebnis, da es im Account unter Reports gespeichert wird. Die farbliche Darstellung der Probleme / Hinweise ist gut verständlich. Es fehlt ein bisschen die Struktur, und das macht es schwierig einen Ansatzpunkt zum Verbessern der Probleme zu finden. Die fehlende grafische Darstellung der Website macht es etwas schwieriger, die Fehler zu finden. Die Hilfestellung ist sehr ausführlich und gut.

Insgesamt eignet sich das Tool, um über einen längeren Zeitraum an einer Website zu arbeiten, da der Bericht gespeichert wird und man immer wieder darauf zugreifen, oder einen neuen erstellen lassen kann. Man kann einzelne Webseiten oder die ganze Website untersuchen lassen (bis zur Ebene 3 und bis zu 25 Seiten).

- 4) **Siteimprove Chrome Extension** ist so wie Wave für einen schnellen ersten Test geeignet, sowie für Benutzer, die HTML/CSS Kenntnisse haben. Hier ist es sogar noch etwas schwieriger, den Code zu finden, da man den Web Dev Tools einschalten und bedienen können muss. Die Stärken von Siteimprove liegen in der guten Strukturiertheit der Fehler, da sie nach den WCAG Richtlinien sortiert sind. Auch die Hilfestellung und die weiterführenden Links sind sehr gut.

**Generell ist zu sagen, dass die Tools sich immer dazu eignen, sich einen ersten Überblick über die Probleme zu verschaffen, und erste Probleme zu beseitigen.**

## 6) Relation zur manuellen Prüfung und Schlussfolgerung

Die Relation zur manuellen Prüfung ist schwierig herzustellen. Ich habe leider nicht mitnotiert in den Sitzungen, was jetzt die genauen „Lösungen“ waren, und meine eigenen Dokumente sind nicht komplett richtig gewesen, weil ich Dinge manchmal falsch eingeschätzt hatte.

Ich denke aber, dass die Tools, die Probleme, die sie herausfinden **können**, besser herausfinden, als es bei einer manuellen Überprüfung möglich ist. Bei den Problemen, die sie nicht herausfinden könnten, aber darauf hinweisen, dass sie manuell überprüft werden müssen, sind sie auch sehr hilfreich, da man selbst bei einer manuellen Prüfung viel übersehen kann.

**Von daher halte ich die Vorgehensweise für sinnvoll:** bei der Überprüfung einer Website auf Barrierefreiheit sollte man immer erst eines der Tools verwenden, am besten das, mit dem man am besten zurechtkommt, oder vielleicht sogar zwei. Auch die Überprüfung der HTML Struktur (Validierung über den W3C Checker) und des css halte ich für wichtig. Danach sollte man in die manuelle Prüfung gehen. Den BITV Selbst-Test halte ich ebenfalls für sehr wichtig, da man hier strategisch die Punkte durchgehen kann, und bewerten welche erfüllt sind und welche Probleme noch beseitigt werden müssen.