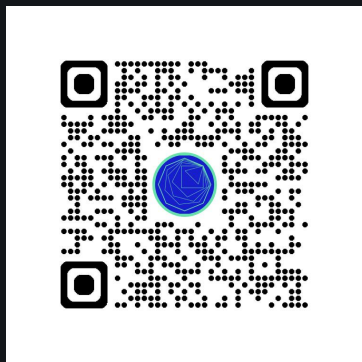




A11Y

Barrierefrei hacken

Incluthon



incluthon.com



A11Y

AIM Studie

Quiz

Kontraste

ALT

Links

Überschriften

Tools





Web AIM Studie

- Studie begann 2019
- 1.000.000 Homepages werden erfasst
- Bericht ist eine quantifizierte und zuverlässige Darstellung
- deckt aktuellen Zustand der Zugänglichkeit der einflussreichsten Seiten im Internet auf
- Barrieren wurden mit dem Tool WAVE gefunden
- nach Web Content-Richtlinien (WCAG) 2.2 Level A/AA-Konformitätsfehler getestet
- WAVE ist ein automatisiertes Tool -> Es kann nur wenige Fehler finden
- Die Studie hat eine hohe Aussagekraft, trotz Teil Erfassung

Quelle: <https://webaim.org/projects/million/>



Auf den eine Million Homepages wurden 50.960.288
eindeutige Barrierefreiheits-Verstösse festgestellt

Durchschnittlich 51 Fehler pro Seite



2024 wurden bei **95,9% der Homepages**
WCAG-2-Fehler festgestellt



Was sind die größten Fehler?

Lasst uns ein Quiz draus machen!



Wie viel Proze

Wie viel Prozent aller Websites hatten zu niedrigen
Farb-Kontrast bei Texten?



79%

Wie viel Prozent aller Websites hatten zu niedrigen
Farb-Kontrast bei Texten?



Wie viel Prozent aller Websites hatten einen fehlenden
Alternativ Text?



55%

Wie viel Prozent aller Websites hatten einen fehlenden
Alternativ Text?



Wie viel Prozent hatten fehlende
Formular-Eingabe-Beschriftungen?



48%

Wie viel Prozent hatten fehlende
Formular-Eingabe-Beschriftungen?



Wie viel Prozent aller Websites hatten leere Links
oder fehlerhafte Beschreibungen der Links?



45%

Wie viel Prozent aller Websites hatten leere Links
oder fehlerhafte Beschreibungen der Links?



Wie viel Prozent hatten fehlerhafte Überschrifts-Ebenen?



39%

Wie viel Prozent hatten fehlerhafte Überschrifts-Ebenen?



5 Hacks

Wir machen jetzt die Quote besser!



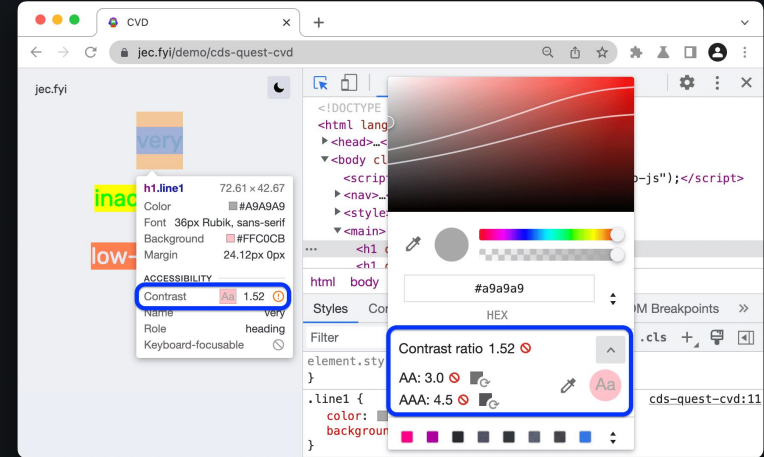
Farb-Kontrast (bei Texten)

Color Contrast Checker

<https://coolors.co/contrast-checker/112a46-acc8e5>

Contrast Checker in Chrome

<https://developer.chrome.com/blog/devtools-tips-5?hl=en>





Alternativ Text

KI Gernerator

<https://barrierefreies.design/werkzeuge/ki-generator-alternativer-text-von-bildern>

Geht auch bei ChatGPT und Co

→ Achtung wegen Datenschutz

→ Apertus ist die Schweizer Alternative

<https://www.swiss-ai.org/apertus>



Alternativ Text

1. Ziel des Bilds verstehen

→ Alt-Text beschreibt die Funktion oder Bedeutung, nicht das Aussehen.

Beispiel:

statt „Ein blauer Button mit weißer Schrift“

besser „Absenden-Button des Kontaktformulars“

2. redundante Informationen vermeiden!

Wenn die Bildunterschrift den Inhalt schon erklärt oder das Bild dekorativ ist, reicht ein leeres `alt=""`.

3. Kontext zählt

Ein Firmenlogo im Header braucht: „Logo der Stadt Berlin“ nicht „Logo“.



Alternativ Text

4. Keine Formulierungen wie „Bild von ...“ oder „Foto zeigt ...“ das erkennt der Screenreader.

5. Für komplexe Bilder (Diagramme, Infografiken, Karten):

im `alt` nur Kurzbeschreibung verwenden z. B. „Diagramm: Anteil barrierefreier Webseiten 2025 nach Branche“

→ Eine detaillierte Beschreibung bspw. im Fließtext, unter dem Bild oder oder verlinkt zur vollständige Beschreibung-Seite

6. Bei Icons oder Buttons

Alt-Text beschreibt die Funktion, nicht das Symbol:

Beispiel:

„Suche starten“ statt „Lupe“.

„Seite schließen“ statt „X-Symbol“.

7. Für Social Media / CMS

Alt-Texte dürfen etwas emotionaler sein, wenn das Bild eine Stimmung oder Geschichte trägt

→ z. B. für Posts mit emotionalem Inhalt oder Storytelling und Stockfotos.



Alternativ Text

Jedes Eingabefeld sollte ein sichtbares und verknüpftes Label haben

Verwende das HTML-Element `<label>` mit dem Attribut `for` und der passenden `id` des Eingabefelds:

```
<label  
for="email">E-Mail-Adresse</label>
```

```
<input id="email" type="email"  
name="email">
```

- Auf Platzhalter als alleinige Beschriftung verzichten
- Pflichtfelder deutlich kennzeichnen
z. B. „erforderlich“
- Fehlermeldungen und Hilfetexte klar zuordenbar und verständlich machen



Links

Leere Links ohne sichtbaren oder erkennbaren Text sind für Screenreader unverständlich und werden daher vermieden.

Jeder Link braucht einen aussagekräftigen Linktext, der das Ziel oder die Funktion beschreibt:

```
<a href="kontakt.html">Kontaktformular  
öffnen</a>
```

Leere Links werden häufig genutzt, um Aktionen auszulösen z. B. „Menü öffnen“, führen aber nirgendwohin.

Das ist nicht barrierefrei, weil Screenreader einen *Navigationslink* erwarten, der zu einer neuen Seite oder einem neuen Ziel führt.

Beispiel:

```
<button type="button">Menü  
öffnen</button>
```




Überschriften-Ebenen

Nur eine Hauptüberschrift (<h1>) pro Seite oder Dokument

→ Sie beschreibt das Hauptthema oder den Titel

- Keine Ebenen überspringen! Nach <h2> kommt <h3>, nicht direkt <h4>
- Überschriften dienen der Struktur, nicht der Optik
- Schriftgröße, Fettung oder Farbe dürfen kein Ersatz für Überschriften-Elemente sein

Überschriften-Ebenen chronologisch aufbauen:

<h1>Barrierefreiheit in digitalen Formularen</h1>

<h2>Eingabefelder</h2>

<h3>Pflichtfelder</h3>

<h3>Hilfetexte</h3>

<h2>Fehlermeldungen</h2>



Weitere Tools & Ressourcen

- [Blogbeitrag Aria Labels nutzen](#)
- [Fachbuch](#) zur Barrierefreie Programmierung
- [Fachbuch](#) zur allgemeinere barrierefreie Umsetzung im Unternehmen
- [App](#) zum VoiceOver detaillierter als über das iOS System lernen mit

Screenreader [NVDA](#)

Screenreader [JAWS](#)

[ALT App](#) zum Beschreiben von Bildern mittels AI für Microsoft und Apple Smartphones

[Lighthouse](#) zum automatisierten Testen in Chrome

Testing mit [WAVE](#)



Prompt (einfach kopieren)

Structure

The language attribute (lang=en) is set correctly on the page. The page title is clear, consistent, and accurately reflects the content of the page.

Images have alternative texts which describes the image content well for blind people (alt attributes). If an image does not provide informational value, the alt attribute is set to empty (alt="").

Headings are used to structure the page, with proper hierarchy levels maintained (h1, h2, h3, etc.).

Input fields are provided with descriptive labels.

Mandatory input fields, checkboxes, and radio groups are marked with aria-required=true.

Presentation

All content is visible and operable in both portrait and landscape orientation (as is standard with responsive design).

The page remains readable and operable even at 200% text zoom.

Text does not scroll in more than one direction (no two-dimensional scrolling).

Text spacing can be adjusted without impacting content or functionality

Manually specified colors meet contrast requirements, see [Understanding Success Criterion 1.4.3: Contrast (Minimum) | WAI | W3C].

Interaction

User input does not trigger automatic interactions.

Keyboard Navigation

No keyboard traps exist.

Navigation follows the reading order of the page.

Focused elements are visually highlighted.

Changing focus does not trigger any actions.

Every functionality (interactive element) is reachable and operable with the keyboard.

Which browsers do we test: Chrome and Safari

Screenreader

Elements are described as they appear—is standard HTML elements such as button, a, input (not only div elements) are used.

Content reading order is correct.

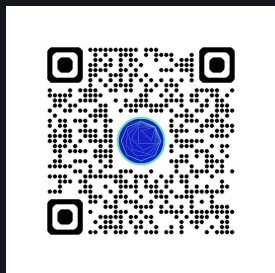
Links have meaningful text."

Lasst uns in Kontakt bleiben!



Claudio Zeni

Alide von Bornhaupt



incluthon.com



[Ressourcen Ordner](#)

