

Website-Qualitätsbericht

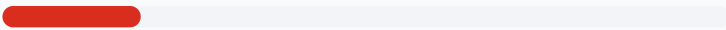
bremen.de

Technischer Website-Check mit Fokus auf Accessibility, SEO und Performance

GESAMTSCORE

19

Kritischer technischer Zustand



WCAG-VORKOMMEN

190

69 kritisch/hoch

MODULE · 0-100, HÖHER IST BESSER



Barrierefreiheit

MANAGEMENT-SUMMARY

Management-Sicht

Gesamtstatus, Hauptrisiken, wichtigste Maßnahmen und Hebel auf einen Blick.

WCAG-Vorkommen	Kritisch	Hoch	Mittel
190	1	68	120

Gesamturteil

<https://www.bremen.de/> erreicht nur 19/100 im Accessibility-Audit. Akuter Handlungsbedarf: Wesentliche Inhalte und Funktionen sind für einen Teil der Nutzer nicht zugänglich.

Wo sich Optimierung lohnt

Barrierefreiheit

19



Kritisch — wesentliche Anforderungen an die Barrierefreiheit werden nicht erfüllt.

Die 5 wichtigsten Risiken

- | | |
|-------------------------------------|--|
| ● 1. Nutzbarkeit | Screenreader und Tastaturnavigation an kritischen Stellen blockiert oder beeinträchtigt. |
| ● 2. Rechtliche Konformität (BFSG) | BFSG-relevante Verstöße (WCAG Level A/AA) gefunden — Behebung empfohlen. |
| ● 3. Conversion & Business-Risiko | Hohes Risiko von Prozessabbrüchen und Conversion-Verlusten. |
| ● 4. SEO & KI-Sichtbarkeit | Sehr gute Auffindbarkeit und strukturierte Schema-Daten. |
| ● 5. Ladezeit & Mobilfreundlichkeit | Schnelle Ladezeit und optimale responsive Darstellung. |

Die 5 wichtigsten Maßnahmen

- 1. Fehlende Name/Rolle bei Bedienelementen: Alle interaktiven Elemente mit einem verständlichen, zugänglichen Namen versehen. Native HTML-Elemente bevorzugen. Bei Icon-Buttons: aria-label verwenden.
- 2. Doppelte IDs im Dokument: Jede ID nur einmal im Dokument vergeben. Bei wiederverwendeten Komponenten die ID dynamisch generieren (z. B. mit Index oder Slug).
- 3. Tastaturfalle — Fokus kann nicht verlassen werden: Sicherstellen, dass der Fokus jedes Element mit Tab und Shift+Tab verlassen kann. Modale Dialoge mit Escape schließbar machen und den Fokus danach korrekt zurücksetzen.
- 4. Sichtbarer Text stimmt nicht mit dem zugänglichen Namen überein: Sicherstellen, dass der zugängliche Name (aria-label) den sichtbaren Text enthält oder identisch ist.
- 5. Fehlende Barrierefreiheit bei Zusatzinhalten (Tooltips): Verwende eine sichtbare Beschriftung oder moderne, barrierefreie Tooltips (schließbar mit Escape, verknüpft über aria-describedby und hoverbar).

Inhalt

Management-Sicht	2
Inhalt	4
Befunde nach Ursache	5
Ursachenanalyse	5
Maßnahmenplan	7
Technische Details für Entwickler	10
Systemdiagnose	10
Klassifizierung der Befunde	12
Systemische Template- & Komponentenfehler (WCAG A/AA)	12
Einzelfälle & Redaktionelle Befunde (WCAG A/AA)	22
Tastatur-Accessibility-Journey	33
Screenreader-Lesereihenfolge	36
Qualitäts-Analysen	37
Dark Mode	38
KI & Vertrauen	40
Anhang & Methodik	47
Prüfumfang	47
Vollständige Fundstellen	51

01

TEIL 1 VON 3

Befunde nach Ursache

Welche Ursachen die meisten Befunde erklären – und der priorisierte Plan, sie zu beheben.

Zielgruppe: Inhaber, Entscheider und Entwickler. Dieser Teil erklärt, was die Befunde verursacht und was als Nächstes zu tun ist.

URSACHEN

Ursachenanalyse

Viele Einzelbefunde gehen auf wenige wiederkehrende Ursachen zurück – deren Behebung wirkt gebündelt.

Das bedeutet konkret: Ein Fix an der richtigen Stelle – etwa einer wiederverwendeten Komponente oder einem Template – behebt oft mehrere Befunde gleichzeitig, statt jeden einzeln reparieren zu müssen.

Erkannte Kernursachen

- Ursache A: 65 Vorkommen — Fehlende Name/Rolle bei Bedienelementen
- Ursache B: 60 Vorkommen — Sichtbarer Text stimmt nicht mit dem zugänglichen Namen überein
- Ursache C: 19 Vorkommen — Fehlende Barrierefreiheit bei Zusatzinhalten (Tooltips)
- Ursache D: 15 Vorkommen — Links nur durch Farbe erkennbar
- Ursache E: 12 Vorkommen — Fehlende semantische Struktur
- Ursache F: 9 Vorkommen — Unzureichende Klickzielgröße

Verteilung der Vorkommen nach Ursache

A — Fehlende Name/Rolle bei Bedienelementen



B — Sichtbarer Text stimmt nicht mit dem zugängl...



C — Fehlende Barrierefreiheit bei Zusatzinhalten...



D — Links nur durch Farbe erkennbar



E — Fehlende semantische Struktur



F — Unzureichende Klickzielgröße



Sonstige (6 weitere Ursachen)



Verteilung der Mängel nach Ursache

Ursache	Vorkommen	Anteil
A — Fehlende Name/Rolle bei Bedienelementen	65	34 %
B — Sichtbarer Text stimmt nicht mit dem zugängl...	60	32 %
C — Fehlende Barrierefreiheit bei Zusatzinhalten...	19	10 %
D — Links nur durch Farbe erkennbar	15	8 %
E — Fehlende semantische Struktur	12	6 %
F — Unzureichende Klickzielgröße	9	5 %
Sonstige (6 weitere Ursachen)	10	5 %

MASSNAHMEN

Maßnahmenplan

Empfohlene Maßnahmen, gruppiert nach Ebene des Problems — inklusive ergänzender SEO- und Qualitätsempfehlungen ohne Rechtsbezug.

Systemische Maßnahmen

Einmal in Vorlage oder Komponente beheben — wirkt auf allen betroffenen Seiten zugleich.

Interaktive Elemente (Buttons, Links) verständlich benennen

Weitverbreitet: Kann Nutzer ausschließen und rechtliches Risiko erhöhen.

→ Ursache A, 65 Vorkommen (34 %)

Verwende eine sichtbare Beschriftung oder moderne, barrierefreie Tooltips (schließbar mit Escape, verknüpft über aria-describedby und hoverbar).

Häufig: Beeinträchtigt Lesbarkeit für Nutzer mit Sehschwäche.

→ Ursache C, 19 Vorkommen (10 %)

Links im Fließtext mit einem nicht-farblichen Merkmal kennzeichnen: Unterstrich (Standard und empfohlen), Fettschrift oder ein kleines Icon. Der Unterstrich ist die stärkste Konvention.

Häufig: Beeinträchtigt Lesbarkeit für Nutzer mit Sehschwäche.

→ Ursache D, 15 Vorkommen (8 %)

Inhalte semantisch korrekt auszeichnen: Tabellen mit `<table>`, `<th>`, `<td>`; Listen mit `<ul>/<ol>`; Formulare mit `<fieldset>` und `<legend>` gruppieren.

Häufig: Beeinträchtigt Qualität und Nutzererlebnis der Website.

→ Ursache E, 12 Vorkommen (6 %)

Lokale Maßnahmen & Einzelfälle

Punktuelle Korrekturen an einzelnen Seiten, Bildern oder redaktionellen Inhalten.

Sicherstellen, dass der Fokus jedes Element mit Tab und Shift+Tab verlassen kann. Modale Dialoge mit Escape schließbar machen und den Fokus danach korrekt zurücksetzen.

Kann Nutzer ausschließen und rechtliches Risiko erhöhen.

Jede ID nur einmal im Dokument vergeben. Bei wiederverwendeten Komponenten die ID dynamisch generieren (z. B. mit Index oder Slug).

Kann Nutzer ausschließen und rechtliches Risiko erhöhen.

Klickflächen auf mindestens 24×24 CSS-Pixel vergrößern, z. B. durch Padding, oder ausreichend Abstand zu benachbarten Zielen einhalten.

Häufig: Beeinträchtigt Qualität und Nutzererlebnis der Website.

→ Ursache F, 9 Vorkommen (5 %)

Seitenstruktur mit Orientierungspunkten auszeichnen

Beeinträchtigt Qualität und Nutzererlebnis der Website.

Formularfelder eindeutig beschriften

Beeinträchtigt Qualität und Nutzererlebnis der Website.

Die DOM-Reihenfolge an die visuelle Reihenfolge anpassen. Auf positive tabindex-Werte verzichten. Bei dynamischen Inhalten den Fokus programmatisch steuern.

Beeinträchtigt Qualität und Nutzererlebnis der Website.

Tastaturnavigation und Fokus-Reihenfolge sicherstellen

Beeinträchtigt Qualität und Nutzererlebnis der Website.

Hinweise zur Aussagekraft: Automatisierte Befunde sind belastbar für erkennbare Muster, aber kontextabhängige WCAG-Kriterien, semantische Inhaltsqualität und Nutzerwege benötigen weiterhin manuelle Prüfung. Indikator-Module wie KI-Sichtbarkeit, Content Visibility und UX sind Hinweiswerte, keine Garantien.

Audit-Hinweise**● Unvollständiger Messumfang**

Eine oder mehrere angeforderte Prüfungen konnten nicht abgeschlossen werden. Die Scores beschreiben nur den erfolgreich gemessenen Umfang.

● Consent-Banner

Ein Consent-Banner wurde erkannt. Teile der Seite können ohne Zustimmung verborgen oder nicht vollständig messbar gewesen sein.

02

TEIL 2 VON 3

Technische Details für Entwickler

Detaillierte WCAG-Verstöße, HTML-Evidenzen und Code-Beispiele, gruppiert nach systemischen Komponenten vs. Einzelfällen.

Zielgruppe: Entwickler und QA-Teams. Dieser Teil liefert konkrete Code-Befunde zur Behebung aller Barrieren.

Modulübersicht

● Barrierefreiheit 19/100

Systemdiagnose

Einzelnes dominantes Problem: Ein Regeltyp verursacht den Großteil aller kritischen/hohen Findings. Die Behebung dieser einen Ursache hat den größten Einzeleffekt.

Einzelnes dominantes Problem: "Missing name/role on controls" verursacht den Großteil der kritischen/hohen Findings.

Kategorie-Übersicht

Kategorie	Vorkommen	Schwerster Schweregrad
Accessibility	190	Kritisch

Problemcluster

Cluster	Befunde	Vorkommen	Max. Schwere- grad
Barrierefreiheits-Barrieren (13 Befunde)	13	190	Kritisch

BEFUNDE

Klassifizierung der Befunde

Alle technischen Befunde, getrennt nach systemischen Komponentenfehlern und Einzelfällen.

Systemisch (Pflicht)	Systemisch (Optimierung)	Lokal (Pflicht)	Lokal (Optimierung)
5	0	7	0

SYSTEMISCHE PFLICHT

Systemische Template- & Komponentenfehler (WCAG A/AA)

Diese Befunde betreffen wiederkehrende Template- oder Komponentenfehler. Eine zentrale Behebung in der Vorlage behebt die Fehler auf allen betroffenen Seiten gleichzeitig.

[Hoch] Fehlende Name/Rolle bei Bedienelementen — WCAG 4.1.2 (A)

Interaktive Elemente (Buttons, Links, Formularfelder) haben keinen zugänglichen Namen oder keine erkennbare Rolle. Assistive Technologien können nicht vermitteln, worum es sich handelt. Screenreader-Nutzer hören z. B. nur 'Button' ohne Beschreibung der Funktion, oder ein klickbares Element wird gar nicht als interaktiv erkannt.

Einstufung	Zuständig	Aufwand	Umsetzungspriorität
Hoch	Entwicklung	Geringer Aufwand	Hoch

PRÜFMETADATEN

Erkennungssicherheit	Sehr hoch
Falschpositiv-Risiko	Sehr niedrig
BFSG-/EAA-Relevanz (Barrierefreiheitsstärkungsgesetz / European Accessibility Act)	Hoch
Elemente	65
Vorkommen	65
Referenz	https://www.w3.org/WAI/WCAG21/Understanding/name-role-value.html

MASSNAHMENBEWERTUNG

Erwartete Wirkung	Behebt 65 Vorkommen; erwartete Auswirkung auf den Score: hoch; WCAG-Level: A.
Komplexitätsgrund	65 Vorkommen deuten auf ein Komponenten- oder Template-Problem hin.

Element-Typen 2x https

Empfehlung

Alle interaktiven Elemente mit einem verständlichen, zugänglichen Namen versehen. Native HTML-Elemente bevorzugen. Bei Icon-Buttons: aria-label verwenden.

× Falsch

```
<button><svg> ... </svg></button>
```

✓ Richtig

```
<button aria-label="Open menu"><svg  
aria-hidden="true"> ... </svg></button>
```

Repräsentative Fundstellen

Fundstelle	Hinweis
a [href: ...n-in-bremen/mobilitaet-und-verkeh...	Link is missing accessible text
a [href: https://www.senatsp ressestelle.breme...	Link is missing accessible text
a [href: https://www. frauenseiten.bremen.de]	Link is missing accessible text

Fundstelle: a [href: ...n-in-bremen/mobilitaet-und-verkehr/verkehrsme...

Node **4029**

Hinweis **Link is missing accessible text**

HTML **<a href="/leben-in-bremen/mobilitaet-und-verkehr/verkehrsmeldungen" target="_self" tabindex="-1" role="pres...**

DOM-Pfad **div.relative > div.relative:nth-of-type(1) > a**

Fundstelle: a [href: https://www.senatspressestelle.bremen.de/]

Node 4020

Hinweis Link is missing accessible text

HTML `<a href="https://www.senatspressestelle.bremen.de/" target="_blank" tabindex="-1" role="presentation" rel="..."`

DOM-Pfad `div.relative > div.relative:nth-of-type(1) > a`

Fundstelle: a [href: https://www.frauenseiten.bremen.de]

Node 4038

Hinweis Link is missing accessible text

HTML `<a href="https://www.frauenseiten.bremen.de" target="_blank" tabindex="-1" role="presentation" rel="noopener..."`

DOM-Pfad `div.relative > div.relative:nth-of-type(1) > a`

Häufige Muster

Muster	Vorkommen
a [href: /leben-in-bremen/fahrradstadt]	1
a [href: /leben-in-bremen/shopping/komiko]	1
a [href: https://www.frauenseiten.bremen.de]	1

Diese Tabelle gruppiert Vorkommen nach struktureller Ähnlichkeit (gleiches Element- und Auswahlmuster) über die gesamte Seite hinweg, nicht nach exakt identischem Wert — die Zahl kann daher von der Vorkommenszahl eines einzelnen Befunds abweichen.

Root Cause

Root Cause: 1 Komponentenproblem erzeugt 65 Vorkommen. Wahrscheinlich ein gemeinsam genutztes Template oder eine Komponente — ein einmaliger Fix behebt alle Vorkommen gleichzeitig.

[Mittel] Sichtbarer Text stimmt nicht mit dem zugänglichen Namen überein — WCAG 2.5.3 (A)

Der sichtbare Text eines Bedienelements stimmt nicht mit dem Namen überein, den assistive Technologien vorlesen. Sprachsteuerungsnutzer können das Element nicht ansprechen. Nutzer, die Sprachsteuerung verwenden, können Elemente nicht per Sprachbefehl aktivieren, weil der vorgelesene Name nicht dem sichtbaren Text entspricht.

Einstufung	Zuständig	Aufwand	Umsetzungspriorität
Mittel	Entwicklung	Geringer Aufwand	Hoch

PRÜFMETADATEN

Erkennungssicherheit	Sehr hoch
Falschpositiv-Risiko	Sehr niedrig
BFSG-/EAA-Relevanz	Mittel
Elemente	60
Vorkommen	60
Referenz	https://www.w3.org/WAI/WCAG21/Understanding/label-in-name.html

MASSNAHMENBEWERTUNG

Erwartete Wirkung	Behebt 60 Vorkommen; erwartete Auswirkung auf den Score: mittel; WCAG-Level: A.
Komplexitätsgrund	60 Vorkommen deuten auf ein Komponenten- oder Template-Problem hin.

Empfehlung

Sicherstellen, dass der zugängliche Name (aria-label) den sichtbaren Text enthält oder identisch ist.

× Falsch

✓ Richtig

```
<button aria-label="Close menu">X</button>
```

```
<button aria-label="Close">Close</button>
```

Root Cause

Root Cause: 1 Komponentenproblem erzeugt 60 Vorkommen. Wahrscheinlich ein gemeinsam genutztes Template oder eine Komponente — ein einmaliger Fix behebt alle Vorkommen gleichzeitig.

[Mittel] Fehlende Barrierefreiheit bei Zusatzinhalten (Tooltips) — WCAG 1.4.13 (AA)

Zusatzinhalte, die erst bei Hover (Mauszeiger) oder Fokus (Tastatur) erscheinen (wie Tooltips oder native title-Attribute), sind nicht barrierefrei zugänglich oder unvollständig verknüpft. Nutzer von Screenreadern hören diese Inhalte gar nicht, Tastaturnutzer können sie oft nicht schließen, und auf Touchgeräten (Mobilgeräten) sind sie in der Regel unbrauchbar.

Einstufung	Zuständig	Aufwand	Umsetzungspriorität
Mittel	Entwicklung	Mittlerer Aufwand	Hoch

PRÜFMETADATEN

Erkennungssicherheit	Mittel
Falschpositiv-Risiko	Mittel
BFSG-/EAA-Relevanz	Mittel
Elemente	19
Vorkommen	19
Referenz	https://www.w3.org/WAI/WCAG21/Understanding/content-on-hover-or-focus.html

MASSNAHMENBEWERTUNG

Erwartete Wirkung	Behebt 19 Vorkommen; erwartete Auswirkung auf den Score: mittel; WCAG-Level: AA.
Komplexitätsgrund	19 Vorkommen deuten auf ein Komponenten- oder Template-Problem hin.
Prüfung	Manuelle Prüfung empfohlen

Empfehlung

Verwende eine sichtbare Beschriftung oder moderne, barrierefreie Tooltips (schließbar mit Escape, verknüpft über aria-describedby und hoverbar).

× Falsch

```
<button title="Mehr Info">...</button>
<div role="tooltip">Hilfetext</div>
```

✓ Richtig

```
<button aria-describedby="tip1">Mehr
Info</button>
<div id="tip1" role="tooltip">Hilfe-
text</div>
```

Root Cause

Root Cause: 1 Komponentenproblem erzeugt 19 Vorkommen. Wahrscheinlich ein gemeinsam genutztes Template oder eine Komponente — ein einmaliger Fix behebt alle Vorkommen gleichzeitig.

[Mittel] Fehlende semantische Struktur — WCAG 1.3.1 (A)

Inhalte sind visuell strukturiert (z. B. durch Größe oder Farbe), aber die Struktur ist nicht im HTML-Code hinterlegt. Screenreader und andere Hilfstechnologien können die Beziehungen zwischen Inhalten nicht erkennen. Nutzer mit Screenreader können Tabellen, Listen und Formulargruppen nicht korrekt navigieren. Die logische Struktur der Seite geht verloren.

Einstufung	Zuständig	Aufwand	Umsetzungspriorität
Mittel	Entwicklung	Mittlerer Aufwand	Hoch

PRÜFMETADATEN

Erkennungssicherheit	Sehr hoch
Falschpositiv-Risiko	Sehr niedrig
BFSG-/EAA-Relevanz	Mittel
Elemente	12
Vorkommen	12
Referenz	https://www.w3.org/WAI/WCAG21/Understanding/info-and-relationships.html

MASSNAHMENBEWERTUNG

Erwartete Wirkung	Behebt 12 Vorkommen; erwartete Auswirkung auf den Score: mittel; WCAG-Level: A.
Komplexitätsgrund	12 Vorkommen deuten auf ein Komponenten- oder Template-Problem hin.

Empfehlung

Inhalte semantisch korrekt auszeichnen: Tabellen mit `<table>`, `<th>`, `<td>`; Listen mit `<ul>/<ol>`; Formulare mit `<fieldset>` und `<legend>` gruppieren.

× Falsch

```
<div class="table">... </div>
```

✓ Richtig

```
<table><thead><tr><th scope="col">... </th></tr></thead>... </table>
```

Repräsentative Fundstellen

Fundstelle	Hinweis
input#CybotCookiebotDialogBodyLevelButtonNecessary	Element with role 'checkbox' is not contained within a landmark region
a [href: #social_media_anchor]	Element with role 'link' is not contained within a landmark region
a [href: #footer_anchor]	Element with role 'link' is not contained within a landmark region

Fundstelle: input#CybotCookiebotDialogBodyLevelButtonNecessary [type=...

Node 2891

Hinweis Element with role 'checkbox' is not contained within a landmark region

HTML `<input type="checkbox" id="CybotCookiebotDialogBodyLevelButtonNecessary" class="CybotCookiebotDialogBodyLev...`

DOM-Pfad `input#CybotCookiebotDialogBodyLevelButtonNecessary`

Vorgeschlagene Code-Korrektur

```
<label    for="CybotCookiebotDialogBodyLevelButtonNecessary">Field    label</label>    <input
type="checkbox" id="CybotCookiebotDialogBodyLevelButtonNecessary" class="CybotCookiebotDi-
alo...
```

Fundstelle: a [href: #social_media_anchor]Node **3462**Hinweis **Element with role 'link' is not contained within a landmark region**HTML **Zu Social-Media-Links**DOM-Pfad **ul > li:nth-of-type(3) > a****Fundstelle: a [href: #footer_anchor]**Node **3464**Hinweis **Element with role 'link' is not contained within a landmark region**HTML **Zur Fußzeile**DOM-Pfad **ul > li:nth-of-type(4) > a****Häufige Muster**

Muster	Vorkommen
a [href: #footer_anchor]	1
a [href: #social_media_anchor]	1
input#CybotCookiebotDialogBodyLevelButtonNecessary [type=checkbox]	1

Diese Tabelle gruppiert Vorkommen nach struktureller Ähnlichkeit (gleiches Element- und Auswahlmuster) über die gesamte Seite hinweg, nicht nach exakt identischem Wert — die Zahl kann daher von der Vorkommenszahl eines einzelnen Befunds abweichen.

Root Cause

Root Cause: 1 Komponentenproblem erzeugt 12 Vorkommen. Wahrscheinlich ein gemeinsam genutztes Template oder eine Komponente — ein einmaliger Fix behebt alle Vorkommen gleichzeitig.

[Mittel] Links nur durch Farbe erkennbar — WCAG 1.4.1 (A)

Links im Fließtext lassen sich ausschließlich durch ihre Farbe von normalem Text unterscheiden. Für Menschen mit Farbsehschwäche sind diese Links nicht als klickbar erkennbar. Nutzer mit Rot-Grün-Schwäche oder anderen Farbsehschwächen können Links im Text nicht erkennen und verpassen so wichtige Navigationsmöglichkeiten.

Einstufung	Zuständig	Aufwand	Umsetzungspriorität
Mittel	Design / UX	Geringer Aufwand	Hoch

PRÜFMETADATEN

Erkennungssicherheit	Sehr hoch
Falschpositiv-Risiko	Sehr niedrig
BFSG-/EAA-Relevanz	Mittel
Elemente	15
Vorkommen	15
Referenz	https://www.w3.org/WAI/WCAG21/Understanding/use-of-color.html

MASSNAHMENBEWERTUNG

Erwartete Wirkung	Behebt 15 Vorkommen; erwartete Auswirkung auf den Score: mittel; WCAG-Level: A.
Komplexitätsgrund	15 Vorkommen deuten auf ein Komponenten- oder Template-Problem hin.

Empfehlung

Links im Fließtext mit einem nicht-farblichen Merkmal kennzeichnen: Unterstrich (Standard und empfohlen), Fettschrift oder ein kleines Icon. Der Unterstrich ist die stärkste Konvention.

× Falsch

✓ Richtig

```
a { color: #0057b8; text-decoration: none; }
```

```
a { color: #0057b8; text-decoration: underline; }
```

Root Cause

Root Cause: 1 Komponentenproblem erzeugt 15 Vorkommen. Wahrscheinlich ein gemeinsam genutztes Template oder eine Komponente — ein einmaliger Fix behebt alle Vorkommen gleichzeitig.

Einzelfälle & Redaktionelle Befunde (WCAG A/AA)

Punktuelle Barrieren, die nur einzelne Seiten, spezifische Bilder oder redaktionelle Texte betreffen und meist individuell behoben werden müssen.

[Mittel] Unzureichende Klickzielgröße — WCAG 2.5.8 (AA)

Bedienelemente wie Buttons, Links oder Icons haben eine Klickfläche von weniger als 24×24 CSS-Pixeln. Nutzer mit motorischen Einschränkungen oder auf Touchscreens treffen kleine Zielflächen leicht daneben und lösen versehentlich falsche Aktionen aus.

Einstufung	Zuständig	Aufwand	Umsetzungspriorität
Mittel	Design / UX	Geringer Aufwand	Normal

PRÜFMETADATEN

Erkennungssicherheit	Sehr hoch
Falschpositiv-Risiko	Sehr niedrig
BFSG-/EAA-Relevanz	Niedrig
Elemente	9
Vorkommen	9
Referenz	https://www.w3.org/WAI/WCAG22/Understanding/target-size-minimum.html

MASSNAHMENBEWERTUNG

Erwartete Wirkung	Behebt 9 Vorkommen; erwartete Auswirkung auf den Score: niedrig; WCAG-Level: AA.
Komplexitätsgrund	9 Vorkommen erfordern einheitliche Anpassungen in Inhalten oder Templates.

Empfehlung

Klickflächen auf mindestens 24×24 CSS-Pixel vergrößern, z. B. durch Padding, oder ausreichend Abstand zu benachbarten Zielen einhalten.

× Falsch

```
<button style="width: 16px; height: 16px; padding: 0;">×</button>
```

✓ Richtig

```
<button style="width: 24px; height: 24px; padding: 4px;">×</button>
```

Strukturelle Ursache

Dieses Problem tritt bei 9 Elementen auf — möglicherweise eine gemeinsam genutzte Komponente oder ein Template.

[Hoch] Doppelte IDs im Dokument — WCAG 4.1.1 (A)

Mehrere Elemente auf der Seite tragen dieselbe id. IDs müssen im gesamten Dokument eindeutig sein, damit assistive Technologien Referenzen (z. B. aria-owns, aria-labelledby) korrekt auflösen können. Screenreader können den Accessibility-Tree nicht korrekt aufbauen und überspringen Elemente oder springen zum falschen Ziel.

Einstufung	Zuständig	Aufwand	Umsetzungspriorität
Hoch	Entwicklung	Mittlerer Aufwand	Quick Win

PRÜFMETADATEN

Erkennungssicherheit	Sehr hoch
Falschpositiv-Risiko	Sehr niedrig
BFSG-/EAA-Relevanz	Hoch
Elemente	3
Vorkommen	3
Referenz	https://www.w3.org/WAI/WCAG21/Understanding/parsing.html

MASSNAHMENBEWERTUNG

Erwartete Wirkung	Behebt 3 Vorkommen; erwartete Auswirkung auf den Score: mittel; WCAG-Level: A.
Komplexitätsgrund	Wenige Vorkommen und ein klar abgegrenzter Fix.

Empfehlung

Jede ID nur einmal im Dokument vergeben. Bei wiederverwendeten Komponenten die ID dynamisch generieren (z. B. mit Index oder Slug).

[Mittel] ARIA-Rollen-Konflikt in Navigation — WCAG 1.3.1 (A)

Ein Element ist mit `role="none"` als semantisch leer markiert, enthält aber Kindelemente mit eigenen ARIA-Rollen. Das verwirrt Screenreader: Der Container hat keine Semantik, aber die enthaltenen Elemente referenzieren Rollentypen, die einen bestimmten Container voraussetzen. Screenreader können Navigationsmenüs nicht korrekt interpretieren. Tastaturnutzer erhalten möglicherweise falsche oder fehlende Navigationshinweise.

Einstufung	Zuständig	Aufwand	Umsetzungspriorität
Mittel	Entwicklung	Geringer Aufwand	Quick Win

PRÜFMETADATEN

Erkennungssicherheit	Sehr hoch
Falschpositiv-Risiko	Sehr niedrig
BFSG-/EAA-Relevanz	Mittel
Elemente	3
Vorkommen	3
Referenz	https://www.w3.org/WAI/WCAG21/Techniques/failures/F92

MASSNAHMENBEWERTUNG

Erwartete Wirkung	Behebt 3 Vorkommen; erwartete Auswirkung auf den Score: niedrig; WCAG-Level: A.
Komplexitätsgrund	Wenige Vorkommen und ein klar abgegrenzter Fix.

Empfehlung

Website-Navigation ohne ARIA-Menubar-Muster aufbauen: `<nav><ul><li><a>` reicht für Screenreader aus. `role="menuitem"` und `role="menubar"` sind für Desktop-App-Menüs reserviert, nicht für Seitennavigation. `role="none"` nur auf Elementen verwenden, die wirklich keine Kinder mit Eigenrollen haben.

✗ Falsch

```
<li role="none"><a href="/home"
role="menuitem">Home</a></li>
```

✓ Richtig

```
<li><a href="/home">Home</a></li>
```

[Kritisch] Tastaturfalle — Fokus kann nicht verlassen werden — WCAG 2.1.2 (A)

An bestimmten Stellen der Website bleibt der Tastaturfokus hängen. Nutzer können mit der Tastatur nicht mehr vor- oder zurücknavigieren. Tastaturnutzer sind an dieser Stelle gefangen und können den Rest der Seite nicht mehr erreichen — die Seite wird faktisch unbenutzbar.

Einstufung	Zuständig	Aufwand	Umsetzungspriorität
Kritisch	Entwicklung	Mittlerer Aufwand	Sofort

PRÜFMETADATEN

Erkennungssicherheit	Sehr hoch
Falschpositiv-Risiko	Sehr niedrig
BFSG-/EAA-Relevanz	Hoch
Elemente	1
Vorkommen	1
Referenz	https://www.w3.org/WAI/WCAG21/Understanding/no-keyboard-trap.html

MASSNAHMENBEWERTUNG

Erwartete Wirkung	Behebt 1 Vorkommen; erwartete Auswirkung auf den Score: mittel; WCAG-Level: A.
Komplexitätsgrund	Die Behebung ist technisch, betrifft aber nur wenige Muster.

Element-Typen	1x div
---------------	--------

Empfehlung

Sicherstellen, dass der Fokus jedes Element mit Tab und Shift+Tab verlassen kann. Modale Dialoge mit Escape schließbar machen und den Fokus danach korrekt zurücksetzen.

Repräsentative Fundstellen

Fundstelle	Hinweis
div#CybotCookiebotDialog	Potential keyboard trap detected (modal dialog)

Fundstelle: div#CybotCookiebotDialog

Node 2827

Hinweis Potential keyboard trap detected (modal dialog)

HTML <div id="CybotCookiebotDialog" name="CybotCookiebotDialog" role="dialog" aria-modal="true" aria-labelledby=...

DOM-Pfad div#CybotCookiebotDialog

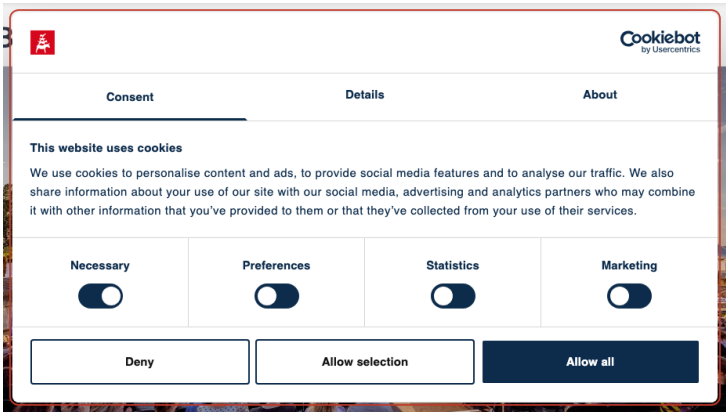


Figure 1: Element auf der Seite (Desktop-Ansicht)

Häufige Muster

Muster	Vorkommen
div#CybotCookiebotDialog	1

Diese Tabelle gruppiert Vorkommen nach struktureller Ähnlichkeit (gleiches Element- und Auswahlmuster) über die gesamte Seite hinweg, nicht nach exakt identischem Wert — die Zahl kann daher von der Vorkommenszahl eines einzelnen Befunds abweichen.

[Mittel] Fehlende Beschriftungen und Anleitungen bei Formularen — WCAG 3.3.2 (A)

Formularfelder haben keine sichtbare Beschriftung oder Anleitung. Nutzer wissen nicht, welche Eingaben erwartet werden. Alle Nutzer, besonders Menschen mit kognitiven Einschränkungen und Screenreader-Nutzer, können Formulare nicht korrekt ausfüllen.

Einstufung	Zuständig	Aufwand	Umsetzungspriorität
Mittel	Entwicklung	Geringer Aufwand	Quick Win

PRÜFMETADATEN

Erkennungssicherheit	Sehr hoch
Falschpositiv-Risiko	Sehr niedrig
BFSG-/EAA-Relevanz	Mittel
Elemente	1
Vorkommen	1
Referenz	https://www.w3.org/WAI/WCAG21/Understanding/labels-or-instructions.html

MASSNAHMENBEWERTUNG

Erwartete Wirkung	Behebt 1 Vorkommen; erwartete Auswirkung auf den Score: niedrig; WCAG-Level: A.
Komplexitätsgrund	Wenige Vorkommen und ein klar abgegrenzter Fix.

Empfehlung

Jedes Formularfeld mit einem sichtbaren, per `<label>` verknüpften Label versehen. Pflichtfelder kennzeichnen. Bei speziellen Formaten (z. B. Datum) das erwartete Format angeben.

× Falsch

```
<input type="text" placeholder="Name">
```

✓ Richtig

```
<label for="name">Name *</label>
<input type="text" id="name" required
aria-required="true">
```

Repräsentative Fundstellen

Fundstelle	Hinweis
input.bg-gray-300 [type=search]	Placeholder used as only label

Fundstelle: input.bg-gray-300 [type=search]

Node 18

Hinweis Placeholder used as only label

HTML `<input name="text" type="search" placeholder="Suche" class="bg-gray-300 border border-blue-500 leading-norm...`

DOM-Pfad `form#header_search_bar > input.bg-gray-300`

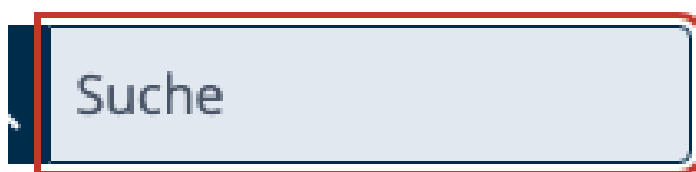


Figure 2: Element auf der Seite (Desktop-Ansicht)

Häufige Muster

Muster	Vorkommen
input.bg-gray-300 [type=search]	1

Diese Tabelle gruppiert Vorkommen nach struktureller Ähnlichkeit (gleiches Element- und Auswahlmuster) über die gesamte Seite hinweg, nicht nach exakt identischem Wert — die Zahl kann daher von der Vorkommenszahl eines einzelnen Befunds abweichen.

[Mittel] Unlogische Fokus-Reihenfolge — WCAG 2.4.3 (A)

Die Reihenfolge, in der interaktive Elemente per Tastatur erreicht werden, entspricht nicht der visuellen oder logischen Reihenfolge der Seite. Tastaturnutzer erleben eine verwirrende Navigation — der Fokus springt zwischen unzusammenhängenden Bereichen hin und her.

Einstufung	Zuständig	Aufwand	Umsetzungspriorität
Mittel	Entwicklung	Mittlerer Aufwand	Normal

PRÜFMETADATEN

Erkennungssicherheit	Hoch
Falschpositiv-Risiko	Mittel
BFSG-/EAA-Relevanz	Mittel
Elemente	1
Vorkommen	1
Referenz	https://www.w3.org/WAI/ARIA/apg/patterns/dialog-modal/

MASSNAHMENBEWERTUNG

Erwartete Wirkung	Behebt 1 Vorkommen; erwartete Auswirkung auf den Score: niedrig; WCAG-Level: A.
Komplexitätsgrund	Die Behebung ist technisch, betrifft aber nur wenige Muster.
Prüfung	Manuelle Prüfung empfohlen

Element-Typen 1× div

Empfehlung

Die DOM-Reihenfolge an die visuelle Reihenfolge anpassen. Auf positive tabindex-Werte verzichten. Bei dynamischen Inhalten den Fokus programmatisch steuern.

Repräsentative Fundstellen

Fundstelle	Hinweis
div#CybotCookiebotDialog	Dialog contains no focusable elements — keyboard users cannot interact with it.

Fundstelle: div#CybotCookiebotDialog

Node **2827**

Hinweis **Dialog contains no focusable elements — keyboard users cannot interact with it.**

HTML **<div id="CybotCookiebotDialog" name="CybotCookiebotDialog" role="dialog" aria-modal="true" aria-labelledby=...**

DOM-Pfad **div#CybotCookiebotDialog**

Häufige Muster

Muster	Vorkommen
div#CybotCookiebotDialog	1

Diese Tabelle gruppiert Vorkommen nach struktureller Ähnlichkeit (gleiches Element- und Auswahlmuster) über die gesamte Seite hinweg, nicht nach exakt identischem Wert — die Zahl kann daher von der Vorkommenszahl eines einzelnen Befunds abweichen.

[Gering] Inhalte nicht per Tastatur bedienbar — WCAG 2.1.1 (A)

Bestimmte Funktionen der Website können nur mit der Maus bedient werden. Nutzer, die auf die Tastatur angewiesen sind, können diese Inhalte nicht erreichen oder nutzen. Menschen mit motorischen Einschränkungen, die keine Maus verwenden können, sowie Screenreader-Nutzer sind von diesen Funktionen ausgeschlossen.

Einstufung	Zuständig	Aufwand	Umsetzungspriorität
Niedrig	Entwicklung	Mittlerer Aufwand	Normal

PRÜFMETADATEN

Erkennungssicherheit	Sehr hoch
Falschpositiv-Risiko	Sehr niedrig
BFSG-/EAA-Relevanz	Mittel
Elemente	1
Vorkommen	1
Referenz	https://www.w3.org/WAI/WCAG21/Understanding/keyboard.html

MASSNAHMENBEWERTUNG

Erwartete Wirkung	Behebt 1 Vorkommen; erwartete Auswirkung auf den Score: niedrig; WCAG-Level: A.
Komplexitätsgrund	Die Behebung ist technisch, betrifft aber nur wenige Muster.
Element-Typen	1× div

Empfehlung

Alle interaktiven Elemente mit nativen HTML-Elementen (button, a, input) umsetzen. Bei Custom-Widgets Tastaturnavigation (Tab, Enter, Escape, Pfeiltasten) implementieren.

× Falsch

```
<div onclick="doAction()">Klick mich</div>
```

✓ Richtig

```
<button type="button" onclick="doAction()">Klick mich</button>
```

Repräsentative Fundstellen

Fundstelle	Hinweis
div#main_anchor	Focusable element without interactive role

Fundstelle: div#main_anchor

Node 3599

Hinweis **Focusable element without interactive role**

HTML `<div id="main_anchor" tabindex="-1"></div>`

DOM-Pfad **div#main_anchor**

Vorgeschlagene Code-Korrektur

```
<!-- Add tabindex="0" and a keydown handler to make this element keyboard-operable --> <div id="main_anchor" tabindex="0"></div>
```

Häufige Muster

Muster	Vorkommen
div#main_anchor	1

Diese Tabelle gruppiert Vorkommen nach struktureller Ähnlichkeit (gleiches Element- und Auswahlmuster) über die gesamte Seite hinweg, nicht nach exakt identischem Wert — die Zahl kann daher von der Vorkommenszahl eines einzelnen Befunds abweichen.

Tastatur-Accessibility-Journey

1 interaktive Befunde mit hoher Schwere gefunden.

HIGH

Tab-Navigation

Die Pfeiltasten-Navigation verschiebt die Auswahl nicht zwischen den Tabs. Tastaturnutzer können die Tab-Liste nicht bedienen. — Das Roving-Tabindex-Muster implementieren: Pfeil-rechts verschiebt Fokus und aria-selected zum nächsten Tab.

Geprüfte Journey-Sequenzen

tab_walk	26 Schritte
disclosure_0	4 Schritte
disclosure_1	4 Schritte
disclosure_2	4 Schritte
disclosure_3	4 Schritte
disclosure_4	4 Schritte
disclosure_5	4 Schritte
disclosure_6	4 Schritte
disclosure_7	4 Schritte
disclosure_8	4 Schritte
disclosure_9	4 Schritte
disclosure_10	4 Schritte
disclosure_11	4 Schritte
disclosure_12	4 Schritte
disclosure_13	4 Schritte
disclosure_14	4 Schritte
disclosure_15	4 Schritte
disclosure_16	4 Schritte
disclosure_17	4 Schritte
disclosure_18	4 Schritte
disclosure_19	4 Schritte
disclosure_20	4 Schritte
disclosure_21	4 Schritte
disclosure_22	4 Schritte
disclosure_23	4 Schritte
disclosure_24	4 Schritte
tabs_0	3 Schritte
skip_link_0	2 Schritte
disclosure_25	4 Schritte
disclosure_26	4 Schritte
disclosure_27	4 Schritte
disclosure_28	4 Schritte
disclosure_29	4 Schritte
disclosure_30	4 Schritte
disclosure_31	4 Schritte
disclosure_32	4 Schritte
disclosure_33	4 Schritte
disclosure_34	4 Schritte
disclosure_35	4 Schritte
disclosure_36	4 Schritte
disclosure_37	4 Schritte
disclosure_38	4 Schritte
disclosure_39	4 Schritte
disclosure_40	4 Schritte
disclosure_41	4 Schritte
disclosure_42	4 Schritte

Diese Tests prüfen, ob Browser, DOM, Fokus und Accessibility Tree eine robuste Grundlage für Screenreader-Nutzung liefern. Sie simulieren nicht den exakten Output von NVDA, JAWS oder VoiceOver.

Screenreader-Lesereihenfolge

Wie die Seite Screenreader-Nutzern angekündigt wird: Lesereihenfolge, Landmark- und Heading-Qualität sowie zugängliche Namen. Strukturelle Prüfungen am Accessibility Tree, keine Simulation von NVDA, JAWS oder VoiceOver.

Heading-Qualität	Landmark-Qualität	Namens-Qualität	Angekündigte Knoten	Tab-Stopps
98 / 100	100 / 100	92 / 100	876	162

Die drei Qualitätswerte nutzen eine Skala von 0–100 (höher ist besser) und verdichten erkannte Überschriftenhierarchie, Landmark-Struktur und zugängliche Namen. Angekündigte Knoten und Tab-Stopps sind reine Anzahlen, keine Qualitätswerte.

BFSG-Prüfung: Verstöße im geprüften Umfang festgestellt (siehe Befunde unten).

Screenreader-Befunde

Problem	Schweregrad
Link- oder Button-Name enthält Icon-Font-Zeichen (Unicode Private Use Area). Screen Reader lesen diese als kryptische Zeichencodes vor.	Mittel
Leerer Linktext kommt 10 mal vor und erschwert die Linkliste im Screenreader.	Niedrig
Langer Abschnitt ab Sequenzposition 26 ohne Landmark, Überschrift oder Fokusziel (34 Einträge).	Niedrig
Überschriftenebene wird übersprungen: Some(3) an Sequenzposition 136.	Mittel
Erste Überschrift im Dokument ist H2 statt H1 (Sequenzposition 25).	Mittel
H1 erscheint erst an Sequenzposition 119 — nach einer H2/H3-Überschrift.	Mittel
162 Tab-Stops ohne erkennbaren Skip-Link erschweren Tastatur- und Screenreader-Navigation.	Mittel
Interaktive Elemente ohne zugänglichen Namen werden im Screenreader nicht verständlich angekündigt.	Hoch

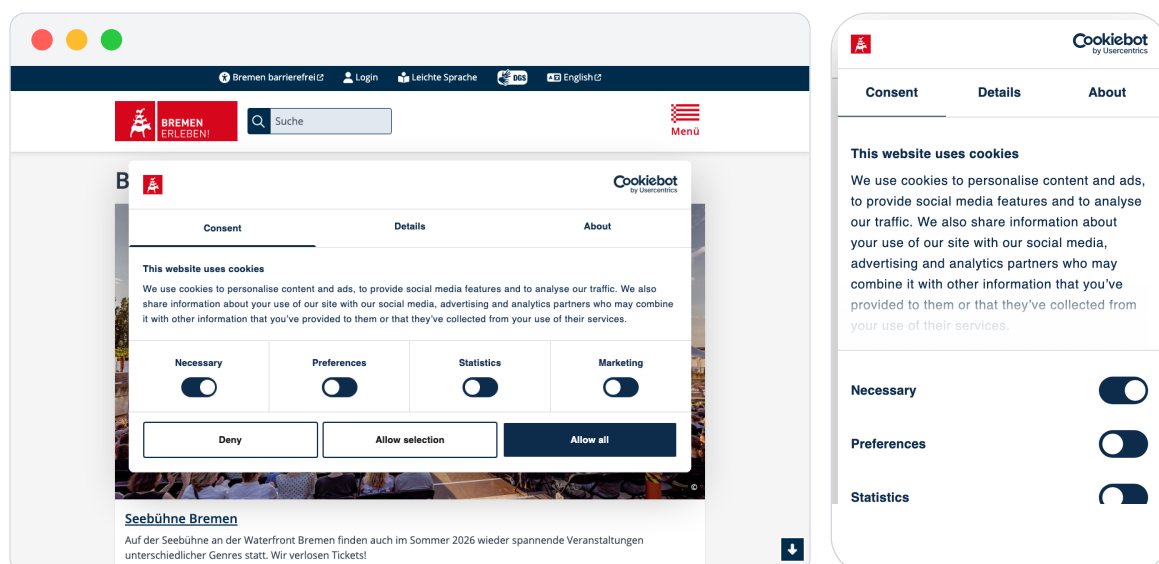
03

TEIL 3 VON 3

Qualitäts-Analysen

Suchmaschinenoptimierung, Ladezeiten, mobiles Layout und Sicherheits-Metriken.

Zielgruppe: Marketing, SEO-Spezialisten und Product Owner. Dieser Teil ergänzt die Barrierefreiheit um Qualitäts-Indikatoren.



Dual-Viewport-Zusammenfassung

Barrierefreiheit - Desktop	Barrierefreiheit - Mobile	Barrierefreiheit - Gesamt
18 / 100 Kritischer Bereich	20 / 100 Kritischer Bereich	19 / 100 70/30 gewichtet

Jeder Barrierefreiheits-Score bewertet Schwere und Vielfalt der automatisiert erkannten WCAG-Befunde seiner Ansicht; 100 bedeutet, dass im automatisierten Prüfumfang kein Problem erkannt wurde. Niedrigere Werte bedeuten höheren Handlungsdruck. WCAG-Vorkommen vor der ansichtsübergreifenden Zusammenführung: 123 auf Desktop und 110 auf Mobile. Die normalisierte gemeinsame Befundliste enthält 190 Vorkommen; diese Anzahl dient Evidenz und Maßnahmenplanung, nicht als dritter Score. Der ausgewiesene Barrierefreiheits-Gesamtwert wird mit 70 % Mobile und 30 % Desktop berechnet.

Dark Mode

Es wurde kein Support für ein dunkles Farbschema erkannt.

Gesamtwertung · 0–100

55



Ausbaufähig

Status	Methoden	CSS Variablen	Print	Forced Colors
Nicht unterstützt	0	0	Ja	Nein

Dark Mode Übersicht

Unterstützung	Nicht unterstützt
color-scheme CSS	Ja
Print-Stylesheet	Ja
Interaktive Seitenelemente im Print verborgen	Nein
Potenziell geclippte Print-Elemente	0
Forced Colors	Nein

Farbsehschwäche-Emulation

Modus	Kontrastprobleme	Neue Probleme	Nur-Farbe-Probleme
protanopia	0	0	9
deutanopia	0	0	9
tritanopia	0	0	9
achromatopsia	0	0	9

Dark Mode Hinweis

Es wurden keine @media (prefers-color-scheme: dark)-Regeln gefunden. Nutzerinnen und Nutzer mit aktiviertem System-Dunkelmodus erhalten weiterhin die helle Ansicht.

Dark Mode Hinweis

Ein Print-Stylesheet ist vorhanden, das Druck-Layout wirkt jedoch riskant: interaktive Bedienelemente ausgeblendet = nein, potenziell abgeschnittene Elemente = 0.

Dark Mode Hinweis

Es wurden keine gezielten Forced-Colors-Regeln gefunden. Nutzerinnen und Nutzer des Windows-Hochkontrastmodus verlieren dadurch möglicherweise Fokusindikatoren, Rahmen oder Statusflächen, wenn Farben rein visuell definiert sind.

VERTRAUEN

KI & Vertrauen

Quellenqualität, KI-Lesbarkeit und Content-Sichtbarkeit — Vertrauens- und Auffindbarkeits-Indikatoren.

| Quellenqualität

Gesamtwertung · 0–100

58



Ausbaufähig

Quellqualität — Überblick: Diese Bewertung basiert ausschließlich auf technischen Signalen (Struktur, Semantik, Metadaten, Sicherheit). Sie beurteilt nicht, ob die dargestellten Inhalte inhaltlich korrekt, vollständig oder aktuell sind.

Der Score zeigt in diesem Bereich erkennbares Verbesserungspotenzial. Quellenqualität zeigt, ob Inhalte substanziell, konsistent und vertrauenswürdig genug wirken, um Entscheidungen zu stützen.

| Substanz

Stark · 0–100

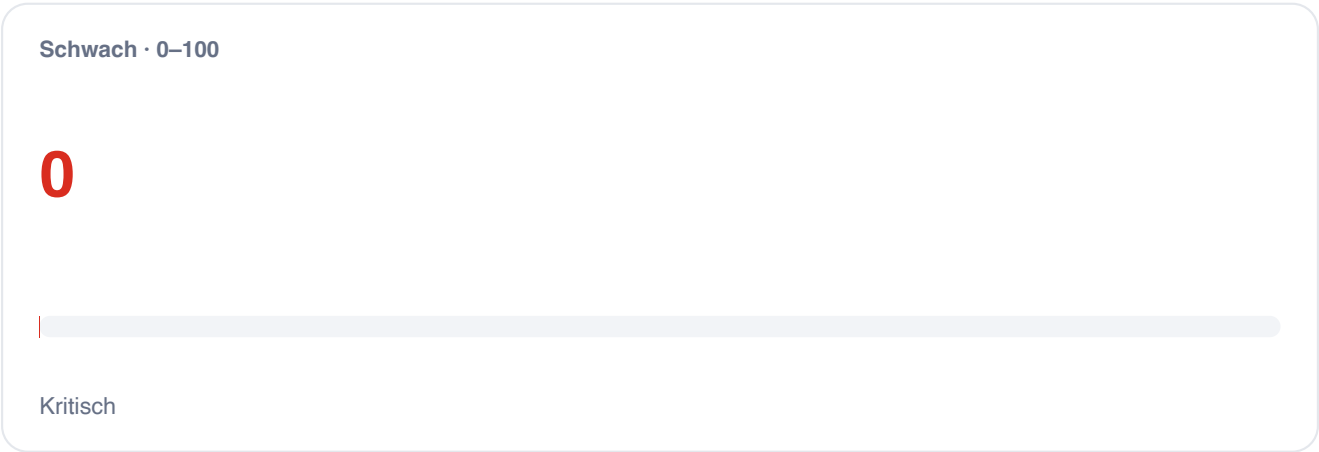
100



Sehr gut

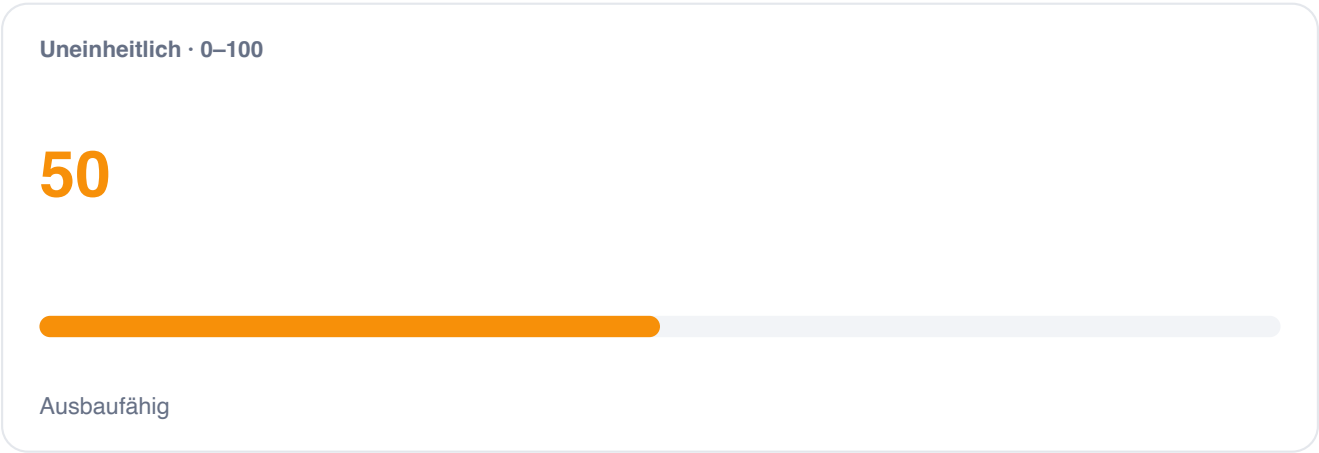
Signal	Status	Detail
Bildbeschreibungen	✓	Alle Bilder haben Alternativtexte
Semantische Struktur	✓	Korrekte Landmark-Regionen

Konsistenz



Signal	Status	Detail
Benannte Bedienelemente	×	65 Elemente ohne zugänglichen Namen
Keine kritischen Fehler	×	1 kritische Verstöße

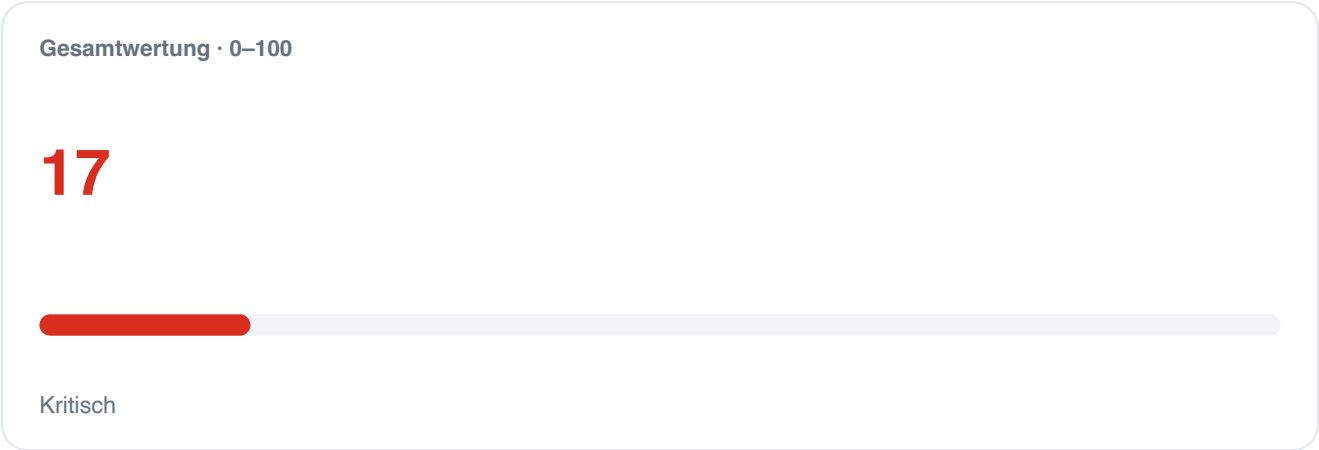
Autorität



Signal	Status	Detail
HTTPS	✓	Verschlüsselte Verbindung

Signal	Status	Detail
Barrierefreiheit	×	Accessibility-Score: 16 (niedrig)

AI-Sichtbarkeit (Indikator)



Indikator-basierte Metriken: Experimentelle Heuristiken — keine standardisierte Metrik. Die Indikatoren sind heuristische Schätzwerte auf Basis messbarer Signale, kein direktes Ranking-Signal. Sie zeigen Optimierungspotenzial, beweisen aber kein konkretes LLM-Verhalten.

KI-Sichtbarkeit — Überblick: Diese Bewertung basiert auf heuristischen Signalen zur AI-Sichtbarkeit. Sie bewertet, wie gut Inhalte für die maschinelle Extraktion und Zitierung durch KI-Systeme aufbereitet sind — nicht deren inhaltliche Qualität.

Der Score zeigt in diesem Bereich eine deutliche Schwäche. KI-Sichtbarkeit zeigt, ob Inhalte von KI-gestützten Systemen gelesen, gegliedert, zugeordnet und zitiert werden können.

KI-Lesbarkeit

Heuristische Schätzung auf Basis struktureller Signale

~0 / 100 — Schwach

Basis

Kritisch

Signal	Status	Detail
Überschriftenstruktur	×	Keine Überschriften — LLMs können keine Abschnitte erkennen
Inhaltsumfang	×	0 Wörter — zu wenig für gehaltvolle LLM-Extraktion (empfohlen ≥ 300)
Absatzstruktur	×	Weniger als 3 Absätze — kaum Gliederung für Chunk-Extraktion

Signal	Status	Detail
Listen / Aufzählungen	×	Keine Listen — Aufzählungen verbessern LLM-Extrahierbarkeit
Tabellen	×	Keine Tabellen (nicht immer nötig)

5 weitere Signale im detaillierten Anhang.

Zitierbarkeit

Heuristische Schätzung auf Basis struktureller Signale Basis ~23 / 100 — Schwach Kritisch

Signal	Status	Detail
Verschlüsselung	✓	HTTPS — Vertrauenssignal für Zitierwürdigkeit
Herausgeber-Identität	×	Kein Herausgeber-Markup — Autorität nicht maschinell prüfbar
Artikelstruktur	×	Kein Artikel-Schema — Content-Typ nicht maschinenlesbar
Publikationsdatum	×	Kein Publikationsdatum — Aktualität nicht einschätzbar
Kanonische URL	×	Keine Canonical-URL — Duplikate möglich

6 weitere Signale im detaillierten Anhang.

Technische KI-Lesbarkeit

Heuristische Schätzung auf Basis struktureller Signale Basis ~15 / 100 — Schwach Kritisch

Signal	Status	Detail
Abschnittszahl	×	Nur 1 Abschnitte — zu wenig für granulare Chunk-Bildung
Heuristik: Abschnittslänge	×	Heuristik: 0 von 1 Abschnitten liegen im Bereich 100–800 Wörter (0%).

Signal	Status	Detail
		Richtwert, keine standardisierte Metrik.
Keine Übergroßen Abschnitte	✓	Kein Abschnitt über 800 Wörter — gut für Token-Limits
Wenig Fragmente	×	1 Kurzabschnitte — viele Fragmente können Kontext verlieren
Hierarchische Gliederung	×	Flache Heading-Struktur — nur sequenzielles Chunking möglich

3 weitere Signale im detaillierten Anhang.

Strukturierte Daten

Heuristische Schätzung auf Basis struktureller Signale
Basis

~0 / 100 — Schwach
Kritisch

Signal	Status	Detail
Entitäten erkannt	×	0 Entitäten extrahiert — wenig maschinenlesbare Entitäten
Schema.org-Entitäten	×	0 Entitäten aus Schema.org — wenig strukturierte Entitäten
Beziehungen	×	0 Beziehungen zwischen Entitäten — isolierte Entitäten, kein Netz
Typen-Vielfalt	×	0 verschiedene Entitätstypen — wenig Typen-Diversität
Breadcrumb-Hierarchie	×	Kein Breadcrumb — thematische Einordnung fehlt

2 weitere Signale im detaillierten Anhang.

AI-Policy

Heuristische Schätzung auf Basis struktureller Signale
Basis

~55 / 100 — Uneinheitlich
Ausbaufähig

Signal	Status	Detail
robots.txt erreichbar	×	robots.txt nicht erreichbar — AI-Crawler haben keine Policy-Info
Kein Wildcard-Block	✓	Kein globaler Block — Crawler haben grundsätzlich Zugang
KI-Suche erreichbar	✓	KI-Suchbots nicht blockiert — Inhalte für KI-Antworten verfügbar
Explizite AI-Policy	×	Keine robots.txt — keine AI-Policy definierbar
Sitemap-Verweis	×	Keine Sitemap in robots.txt — AI-Crawler müssen Seiten selbst entdecken

1 weitere Signale im detaillierten Anhang.

Content-Abschnitte

Abschnitte

Abschnitt	Level	Wörter
Gesamter Inhalt	H0	0

Empfehlung

Zu wenig Gliederung für effektives Chunking. Zusätzliche Zwischenüberschriften würden die Extrahierbarkeit verbessern.

Erkannte Strukturmuster

● Semantische Hauptnavigation	Vorhanden: 1 navigation landmark found, but none contain link children. Pattern is structurally incomplete.
● Disclosure-Menü	Vorhanden: 25 disclosure triggers with aria-expanded (0 with aria-controls). Controls relationship strengthens screen-reader announcements.
● Modaler Dialog	Vorhanden: 1 dialog detected, 0 structurally well-formed (accessible name, aria-modal, focusable descendants).
● Tab-Liste	Stark: 1 tablist with 3 tabs; 3 have aria-selected, 1 declare aria-controls.
● Skip-Link	Vorhanden: Skip link detected ("Zum Inhalt") but not the first focusable element — keyboard users may have to tab past other content first.
● Accordion	Vorhanden: 25 accordion triggers; 0 with aria-controls; 0 non-button triggers.
● WCAG-Signal	Vorhanden: Skip navigation link detected — keyboard users can bypass repeated blocks
● WCAG-Signal	Vorhanden: <main> landmark detected — assistive technology users can navigate directly to main content

Anhang & Methodik

Prüfumfang, Methodik, WCAG-Coverage und die vollständige Fundstellenliste.

Prinzip-Abdeckung (bestandene Kriterien)

Wahrnehmbar	15/18 (83 %)
Bedienbar	21/26 (81 %)
Verständlich	8/9 (89 %)
Robust	1/3 (33 %)

Prüfumfang

Dieses Audit prüft 36 von ca. 50 WCAG-2.1-AA-Kriterien automatisch. Die unten aufgeführten Kriterien benötigen manuelle Prüfung.

Automatisch geprüft (36)

WCAG 1.1.1 (A) WCAG 1.2.1 (A) WCAG 1.2.2 (A) WCAG 1.2.8 (AAA) WCAG 1.3.1 (A)

WCAG 1.3.2 (A) WCAG 1.3.4 (AA) WCAG 1.3.5 (AA) WCAG 1.3.6 (AAA) WCAG 1.4.1 (A)

WCAG 1.4.3 (AA) WCAG 1.4.4 (AA) WCAG 1.4.7 (AAA) WCAG 1.4.8 (AAA) WCAG 1.4.10 (AA)

WCAG 1.4.11 (AA) WCAG 1.4.12 (AA) WCAG 1.4.13 (AA) WCAG 2.1.1 (A) WCAG 2.1.2 (A)

WCAG 2.2.1 (A) WCAG 2.2.2 (A) WCAG 2.2.3 (AAA) WCAG 2.2.4 (AAA) WCAG 2.2.5 (AAA)

WCAG 2.2.6 (AAA) WCAG 2.3.3 (AAA) WCAG 2.4.1 (A) WCAG 2.4.2 (A) WCAG 2.4.3 (A)

WCAG 2.4.4 (A) WCAG 2.4.6 (AA) WCAG 2.4.7 (AA) WCAG 2.4.8 (AAA) WCAG 2.4.9 (AAA)

WCAG 2.4.10 (AAA) WCAG 2.4.11 (AA) WCAG 2.4.12 (AAA) WCAG 2.5.1 (A) WCAG 2.5.2 (A)

WCAG 2.5.3 (A) WCAG 2.5.4 (A) WCAG 2.5.5 (AAA) WCAG 2.5.8 (AA) WCAG 3.1.1 (A)

WCAG 3.1.3 (AAA) WCAG 3.1.4 (AAA) WCAG 3.2.1 (A) WCAG 3.2.2 (A) WCAG 3.3.1 (A)

WCAG 3.3.2 (A) WCAG 3.3.5 (AAA) WCAG 3.3.7 (A) WCAG 4.1.1 (A) WCAG 4.1.2 (A)

WCAG 4.1.3 (AA)

Manuelle Prüfung erforderlich (10)

1.2.3 (A) – Audio Description or Media Alternative 1.2.5 (AA) – Audio Description (Prerecorded)

1.4.2 (A) – Audio Control 1.4.5 (AA) – Images of Text 2.1.4 (A) – Character Key Shortcuts

2.3.1 (A) – Three Flashes or Below Threshold 3.2.3 (AA) – Consistent Navigation

3.2.4 (AA) – Consistent Identification

3.3.3 (AA) – Error Suggestion

3.3.4 (AA) – Error Prevention (Legal, Financial, Data)

So testen Sie manuell

- **Tastaturnavigation** Komplette Seite per Tab navigieren. Kein Fokus verloren, kein Keyboard-Trap, jedes interaktive Element erreichbar.
- **Screenreader** Test mit NVDA/JAWS (Windows) oder VoiceOver (Mac/iOS) — Landmark-Navigation und Formular-Interaktion.
- **400% Zoom** Bei 400% Browser-Zoom: Seite ohne horizontales Scrollen bedienbar, kein Inhalt verloren.
- **Reduced Motion** Betriebssystem-Einstellung „Bewegung reduzieren“ aktivieren und prüfen, ob Animationen deaktiviert oder reduziert werden.
- **Modal- / Dropdown-Interaktion** Vollständige Tastaturbedienung: Tab, Enter, Space, Escape, Pfeiltasten. Fokus kehrt nach Schließen zum Trigger zurück.
- **Farbenblindheit** Mit einem Werkzeug wie „Color Oracle“ prüfen, ob Informationen nicht ausschließlich über Farbe vermittelt werden.

Manuelle Prüfpunkte und heuristische Hinweise

- | | |
|----------------------------|--|
| ● 2.1.2 · Manuelle Prüfung | Sicherstellen, dass der Fokus jedes Element mit Tab und Shift+Tab verlassen kann. Modale Dialoge mit Escape schließbar machen und den Fokus danach korrekt zurücksetzen. |
| ● 2.2.1 · Manuelle Prüfung | Provide a mechanism to disable, adjust (at least 10x), or extend any time limit before it expires, with at least 20 seconds to respond. |
| ● 2.5.1 · Manuelle Prüfung | For every multipoint or path-based gesture, provide an equivalent single-pointer alternative (e.g. a button). |
| ● 2.5.2 · Manuelle Prüfung | Trigger actions on the up-event (mouseup/touchend/click) rather than the down-event (mousedown/touchstart), unless the down-event is essential to the function. |
| ● 2.5.4 · Manuelle Prüfung | For any device motion interaction, provide equivalent button controls and allow users to disable motion actuation. |
| ● 2.1.1 · Hinweis | Add tabindex="0" to make the element focusable, or use the native HTML element (e.g. <button>, <a>). |
| ● 2.4.11 · Hinweis | Reserve space for sticky content or add scroll-padding/scroll-margin so focused controls remain visible |
| ● 2.4.11 · Hinweis | Reserve space for sticky content or add scroll-padding/scroll-margin so focused controls remain visible |
| ● 2.4.11 · Hinweis | Reserve space for sticky content or add scroll-padding/scroll-margin so focused controls remain visible |
| ● 2.4.11 · Hinweis | Reserve space for sticky content or add scroll-padding/scroll-margin so focused controls remain visible |
| ● 2.4.11 · Hinweis | Reserve space for sticky content or add scroll-padding/scroll-margin so focused controls remain visible |

Interaktive Abdeckung: 53 von 53 versuchten Journeys abgeschlossen; 0 fehlgeschlagen, 0 übersprungen.

Grenzen automatisierter Tests: Automatisierte Tests können ca. 30–40% aller Barrierefreiheitsprobleme erkennen. Komplexe Aspekte wie korrekte Tab-Reihenfolge, sinnvolle Alt-Texte oder verständliche Sprache erfordern zusätzlich manuelle Prüfung. Dieser Lauf ist unvollständig: Die Scores beschreiben nur den erfolgreich gemessenen Umfang.

Hinweis: Dieser Report stellt eine automatisierte technische Analyse dar. Er ersetzt keine vollständige Konformitätsbewertung nach WCAG 2.1. Für eine rechtsverbindliche Aussage zur Barrierefreiheit ist eine umfassende manuelle Prüfung durch Experten erforderlich.

ANHANG

Vollständige Fundstellen

Rohdaten des Audits und die vollständige Liste aller erkannten Verstöße.

Audit-Daten

Bereich	Signal	Wert
Audit	Barrierefreiheits-Score	19 / 100 — 70 % Mobile (20) + 30 % Desktop (18)
Audit	Gesamtscore	19 / 100 — beitragende Gewichtung: Barrierefreiheit 100 % Separat ausgewiesene Indikator-Module: Dark Mode, KI-Sichtbarkeit, Quellenqualität.
Audit	Zählweise	13 unterschiedliche WCAG-Befundgruppen, 190 WCAG-Vorkommen; 13 Befundzeilen und 190 Vorkommen über alle Kategorien
Audit	WCAG-Level	AA
Audit	Geprüfte Knoten	15148
Audit	Laufzeit	8.6 s
Audit	Aktive Module	accessibility, accessibility_journey, ai_visibility, dark_mode, source_quality
Audit	Fehlgeschlagene Module	Keine
Audit	Audit-Qualität	Partial
Audit	Audit-Hinweise	2
Audit	Vorschau	Desktop und Mobile erfasst
Modul	Barrierefreiheit	19 / 100 — Kritisch — wesentliche Anforderungen an die Barrierefreiheit werden nicht erfüllt.. 4 Problemgruppe(n) mit hoher Auswirkung
Finding	a11y.name_role.missing (4.1.2)	65 Vorkommen — Screenreader-Nutzer hören z. B. nur 'Button' ohne Beschreibung der Funktion, oder ein klickbares Element wird gar nicht als interaktiv erkannt.
Finding	a11y.parsing.invalid (4.1.1)	3 Vorkommen — Screenreader können den Accessibility-Tree nicht korrekt aufbauen und überspringen Elemente oder springen zum falschen Ziel.
Finding	a11y.keyboard_trap.risk (2.1.2)	1 Vorkommen — Tastaturnutzer sind an dieser Stelle gefangen und können den Rest der Seite nicht mehr erreichen — die Seite wird faktisch unbenutzbar.
Finding	a11y.label_in_name.invalid (2.5.3)	60 Vorkommen — Nutzer, die Sprachsteuerung verwenden, können Elemente nicht per Sprachbefehl aktivieren, weil der vorgelesene Name nicht dem sichtbaren Text entspricht.
Finding	a11y.hover.content_visibility (1.4.13)	19 Vorkommen — Nutzer von Screenreadern hören diese Inhalte gar nicht, Tastaturnutzer können sie oft nicht

Bereich	Signal	Wert
		schließen, und auf Touchgeräten (Mobilgeräten) sind sie in der Regel unbrauchbar.

Rohdaten & Weiterverarbeitung

Der begleitende JSON-Report enthält die vollständige maschinenlesbare Fehlerliste mit Selektoren, Vorkommen und allen Detaildaten für automatisierte Weiterverarbeitung.

Alle Verstöße (aggregiert nach Regel)

- **4.1.2 — Fehlende Name/Rolle bei Bedienelementen**

Link is missing accessible text
- **2.5.3 — Sichtbarer Text ≠ zugänglicher Name**

Button aria-label 'Necessary (30)' does not contain its visible text 'Necessary 30'. Speech input users who speak the visible label will not activate the button.
- **4.1.2 — Fehlender Name bei Befehlselement**

Element with role="link" has no accessible name
- **1.4.13 — Unzugängliche Hover- oder Fokus-Inhalte (Tooltips)**

Interactive element uses the native `title` attribute as its only descriptive text (button). Tooltips from `title` are not keyboard-accessible and unreliable on touch devices.
- **1.3.1 — Fehlende semantische Struktur**

Element with role 'link' is not contained within a landmark region
- **1.4.1 — Link nur durch Farbe unterscheidbar**

Inline link distinguishable from surrounding text by color alone (a#Cybot-CookiebotDialogBodyEdgeMoreDetailsLink). Users with low vision or color blindness may not recognize it as a link.
- **2.5.8 — Unzureichende Klickzielgröße**

Interactive target 'Quelle' is 18x24 CSS pixels, below the 24x24 minimum.
- **4.1.1 — Fehlerhaftes Markup (Parsing)**

Duplicate id='header_search_bar' appears 2 times in the DOM. Duplicate IDs cause AT to resolve references incorrectly.
- **1.3.1 — Semantische Kinder in presentational Container**

Element with role="presentation"/"none" contains semantic child tab (a#Cybot-CookiebotDialogNavDeclaration)
- **2.1.2 — Tastaturfalle — Fokus kann nicht verlassen werden**

Potential keyboard trap detected (modal dialog)
- **3.3.2 — Fehlende Formularbeschriftungen**

Placeholder used as only label
- **2.4.3 — Unlogische Fokus-Reihenfolge**

Dialog contains no focusable elements — keyboard users cannot interact with it.
- **2.1.1 — Inhalte nicht per Tastatur bedienbar**

Focusable element without interactive role