

Website-Qualitätsbericht

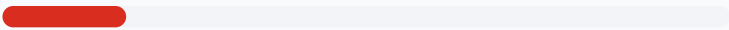
[schleswig-holstein.de](#)

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

GESAMTSCORE

17

Kritischer technischer Zustand



WCAG-VORKOMMEN

204

120 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
204	1	119	60

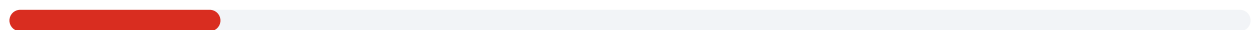
Gesamturteil

<https://www.schleswig-holstein.de/> erreicht nur 17/100 im Accessibility-Audit. Akuter Handlungsbedarf: Wesentliche Inhalte und Funktionen sind für einen Teil der Nutzer nicht zugänglich.

Wo sich Optimierung lohnt

Barrierefreiheit

17



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. Tastaturfokus auf unsichtbaren Elementen: Wenn ein Element vollständig versteckt sein soll, `display:none` oder `visibility:hidden` verwenden. Soll es nur vor Screenreadern versteckt sein, muss es unfokussierbar sein (`tabindex="-1"`).
- 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. Fehlende semantische Struktur: Inhalte semantisch korrekt auszeichnen: Tabellen mit `<table>`, `<th>`, `<td>`; Listen mit `<ul>/<ol>`; Formulare mit `<fieldset>` und `<legend>` gruppieren.
- 5. ARIA-Rollen-Konflikt in Navigation: 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.

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)	20
Tastatur-Accessibility-Journey	29
Screenreader-Lesereihenfolge	31
Qualitäts-Analysen	32
Dark Mode	33
KI & Vertrauen	35
Anhang & Methodik	42
Prüfumfang	42
Vollständige Fundstellen	46

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: 95 Vorkommen — Tastaturfokus auf unsichtbaren Elementen
- Ursache B: 25 Vorkommen — Fehlende semantische Struktur
- Ursache C: 25 Vorkommen — Fehlende Name/Rolle bei Bedienelementen
- Ursache D: 23 Vorkommen — ARIA-Rollen-Konflikt in Navigation
- Ursache E: 22 Vorkommen — Doppelte IDs im Dokument
- Ursache F: 8 Vorkommen — Unzureichende Klickzielgröße

Verteilung der Vorkommen nach Ursache



Verteilung der Mängel nach Ursache

Ursache	Vorkommen	Anteil
A — Tastaturfokus auf unsichtbaren Elementen	95	47 %
B — Fehlende semantische Struktur	25	12 %
C — Fehlende Name/Rolle bei Bedienelementen	25	12 %
D — ARIA-Rollen-Konflikt in Navigation	23	11 %
E — Doppelte IDs im Dokument	22	11 %
F — Unzureichende Klickzielgröße	8	4 %
Sonstige (5 weitere Ursachen)	6	3 %

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.

Wenn ein Element vollständig versteckt sein soll, `display:none` oder `visibility:hidden` verwenden. Soll es nur vor Screenreadern versteckt sein, muss es unfokussierbar sein (`tabindex="-1"`).

Weitverbreitet: Kann Nutzer ausschließen und rechtliches Risiko erhöhen.

→ Ursache A, 95 Vorkommen (47 %)

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

Weitverbreitet: Beeinträchtigt Qualität und Nutzererlebnis der Website.

→ Ursache B, 25 Vorkommen (12 %)

Interaktive Elemente (Buttons, Links) verständlich benennen

Weitverbreitet: Beeinträchtigt Qualität und Nutzererlebnis der Website.

→ Ursache C, 25 Vorkommen (12 %)

Seitenstruktur mit Orientierungspunkten auszeichnen

Weitverbreitet: Beeinträchtigt Qualität und Nutzererlebnis der Website.

→ Ursache D, 23 Vorkommen (11 %)

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

Weitverbreitet: Kann Nutzer ausschließen und rechtliches Risiko erhöhen.

→ Ursache E, 22 Vorkommen (11 %)

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.

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, 8 Vorkommen (4 %)

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

Beeinträchtigt Lesbarkeit für Nutzer mit Sehschwäche.

Textcontainer flexibel gestalten (min-height statt fester Höhe, kein overflow:hidden auf Textbereichen) und mit den WCAG-Mindestabständen testen.

Beeinträchtigt Lesbarkeit für Nutzer mit Sehschwäche.

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 17/100

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: "Focusable element hidden from screen readers" verursacht den Großteil der kritischen/hohen Findings.

Kategorie-Übersicht

Kategorie	Vorkommen	Schwerster Schweregrad
Accessibility	204	Kritisch

Problemcluster

Cluster	Befunde	Vorkommen	Max. Schwere- grad
Barrierefreiheits-Barrieren (12 Befunde)	12	204	Kritisch

BEFUNDE

Klassifizierung der Befunde

Alle technischen Befunde, getrennt nach systemischen Komponentenfehlern und Einzelfällen.

Systemisch (Pflicht)	Systemisch (Optimierung)	Lokal (Pflicht)	Lokal (Optimierung)
5	0	6	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] Tastaturfokus auf unsichtbaren Elementen – WCAG aria-hidden-focus (A)

Ein interaktives Element ist über aria-hidden="true" vor Screenreadern versteckt, kann aber dennoch mit der Tastatur (Tabulator) fokussiert werden. Tastaturnutzer steuern auf unsichtbare Elemente (Fokus-Stolperfallen), während Screenreader-Nutzer keinerlei Rückmeldung über dieses Element erhalten.

Einstufung	Zuständig	Aufwand	Umsetzungspriorität
Hoch	Entwicklung	Geringer Aufwand	Hoch

PRÜFMETADATEN

Erkennungssicherheit	Hoch
Falschpositiv-Risiko	Niedrig
BFSG-/EAA-Relevanz (Barrierefreiheitsstärkungsgesetz / European Accessibility Act)	Niedrig
Elemente	95
Vorkommen	95
Referenz	https://www.w3.org/WAI/WCAG21/Understanding/name-role-value.html

MASSNAHMENBEWERTUNG

Erwartete Wirkung	Behebt 95 Vorkommen; erwartete Auswirkung auf den Score: hoch; WCAG-Level: A.
Komplexitätsgrund	95 Vorkommen deuten auf ein Komponenten- oder Template-Problem hin.

Empfehlung

Wenn ein Element vollständig versteckt sein soll, `display:none` oder `visibility:hidden` verwenden. Soll es nur vor Screenreadern versteckt sein, muss es unfokussierbar sein (`tabindex="-1"`).

× Falsch

```
<button aria-hidden="true">Menu</button>
```

✓ Richtig

```
<button aria-hidden="true"
tabindex="-1">Menu</button>
```

Root Cause

Root Cause: 1 Komponentenproblem erzeugt 95 Vorkommen. Wahrscheinlich ein gemeinsam genutztes Template oder eine Komponente — ein einmaliger Fix behebt alle Vorkommen gleichzeitig.

[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	Hoch

PRÜFMETADATEN

Erkennungssicherheit	Sehr hoch
Falschpositiv-Risiko	Sehr niedrig
BFSG-/EAA-Relevanz	Hoch
Elemente	22
Vorkommen	22
Referenz	https://www.w3.org/WAI/WCAG21/Understanding/parsing.html

MASSNAHMENBEWERTUNG

Erwartete Wirkung	Behebt 22 Vorkommen; erwartete Auswirkung auf den Score: hoch; WCAG-Level: A.
Komplexitätsgrund	22 Vorkommen deuten auf ein Komponenten- oder Template-Problem hin.

Empfehlung

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

Root Cause

Root Cause: 1 Komponentenproblem erzeugt 22 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	25
Vorkommen	25
Referenz	https://www.w3.org/WAI/WCAG21/Understanding/info-and-relationships.html

MASSNAHMENBEWERTUNG

Erwartete Wirkung	Behebt 25 Vorkommen; erwartete Auswirkung auf den Score: mittel; WCAG-Level: A.
Komplexitätsgrund	25 Vorkommen deuten auf ein Komponenten- oder Template-Problem hin.

Element-Typen 1× h2

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#_ffx4qicgc [type=checkbox]	Element with role 'checkbox' is not contained within a landmark region
h2#_jzs9sch02	Element with role 'heading' is not contained within a landmark region
a.RichTextIntlLink [href: / impressum.html]	Element with role 'link' is not contained within a landmark region

Fundstelle: input#_ffx4qicgc [type=checkbox]

Node 4443

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

HTML `<input class="cookiebanner-control__element js-banner-control__element" id="_ffx4qicgc" name="externalVideo..."`

DOM-Pfad input#_ffx4qicgc

Vorgeschlagene Code-Korrektur

```
<label for="_ffx4qicgc">Field label</label> <input class="cookiebanner-control__element js-banner-control__element" id="_ffx4qicgc" name="externalVideos" type="checkbox" aria-de...
```

Fundstelle: h2#_jzs9sch02

Node 4440

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

HTML `<h2 class="cookiebanner-section__title" id="_jzs9sch02">Externe Videos (z.B. Youtube/Vimeo)</h2>`

DOM-Pfad `h2#_jzs9sch02`

Fundstelle: a.RichTextIntLink [href: /impressum.html]

Node 4429

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

HTML `<a class="RichTextIntLink NavNode" href="/impressum.html" title="Impressum">Impressum</a>`

DOM-Pfad `div.cookiebanner-section:nth-of-type(1) > p:nth-of-type(2) > a.RichTextIntLink`



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

Häufige Muster

Muster	Vorkommen
h2#_ap3jrb9l4	1
h2#_jzs9sch02	1
h1#_seo6vtkbi	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 25 Vorkommen. Wahrscheinlich ein gemeinsam genutztes Template oder eine Komponente — ein einmaliger Fix behebt alle Vorkommen gleichzeitig.

[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	Hoch

PRÜFMETADATEN

Erkennungssicherheit	Sehr hoch
Falschpositiv-Risiko	Sehr niedrig
BFSG-/EAA-Relevanz	Mittel
Elemente	23
Vorkommen	23
Referenz	https://www.w3.org/WAI/WCAG21/Techniques/failures/F92

MASSNAHMENBEWERTUNG

Erwartete Wirkung	Behebt 23 Vorkommen; erwartete Auswirkung auf den Score: mittel; WCAG-Level: A.
Komplexitätsgrund	23 Vorkommen deuten auf ein Komponenten- oder Template-Problem hin.

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>
```

Root Cause

Root Cause: 1 Komponentenproblem erzeugt 23 Vorkommen. Wahrscheinlich ein gemeinsam genutztes Template oder eine Komponente — ein einmaliger Fix behebt alle Vorkommen gleichzeitig.

[Gering] 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
Niedrig	Entwicklung	Geringer Aufwand	Normal

PRÜFMETADATEN

Erkennungssicherheit	Sehr hoch
Falschpositiv-Risiko	Sehr niedrig
BFSG-/EAA-Relevanz	Mittel
Elemente	25
Vorkommen	25
Referenz	https://www.w3.org/WAI/ARIA/apg/patterns/tabs/

MASSNAHMENBEWERTUNG

Erwartete Wirkung	Behebt 25 Vorkommen; erwartete Auswirkung auf den Score: mittel; WCAG-Level: A.
Komplexitätsgrund	25 Vorkommen deuten auf ein Komponenten- oder Template-Problem hin.

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>
```

Root Cause

Root Cause: 1 Komponentenproblem erzeugt 25 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	8
Vorkommen	8
Referenz	https://www.w3.org/WAI/WCAG22/Understanding/target-size-minimum.html

MASSNAHMENBEWERTUNG

Erwartete Wirkung	Behebt 8 Vorkommen; erwartete Auswirkung auf den Score: niedrig; WCAG-Level: AA.
Komplexitätsgrund	8 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 8 Elementen auf — möglicherweise eine gemeinsam genutzte Komponente oder ein Template.

[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	Quick Win

PRÜFMETADATEN

Erkennungssicherheit	Mittel
Falschpositiv-Risiko	Mittel
BFSG-/EAA-Relevanz	Mittel
Elemente	2
Vorkommen	2
Referenz	https://www.w3.org/WAI/WCAG21/Understanding/content-on-hover-or-focus.html

MASSNAHMENBEWERTUNG

Erwartete Wirkung	Behebt 2 Vorkommen; erwartete Auswirkung auf den Score: niedrig; WCAG-Level: AA.
Komplexitätsgrund	Wenige Vorkommen und ein klar abgegrenzter Fix.
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>
```

[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	1× 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.mfp-content	Potential keyboard trap detected (modal dialog)

Fundstelle: div.mfp-content

Node 4421

Hinweis Potential keyboard trap detected (modal dialog)

HTML <div class="mfp-content" role="dialog" aria-modal="true" aria-labelledby="_seo6vtkbi"><section id="cookieba...

DOM-Pfad div.mfp-wrap:nth-of-type(2) > div.mfp-container > div.mfp-content:nth-of-type(1)



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

Häufige Muster

Muster	Vorkommen
div.mfp-content	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] Inhalt wird bei größerem Zeilen-/Wortabstand abgeschnitten — WCAG 1.4.12 (AA)

Wenn Nutzer den Zeilen-, Wort- oder Buchstabenabstand für bessere Lesbarkeit erhöhen, schneidet ein Container mit fester Höhe oder `overflow:hidden` Teile des Textes ab. Nutzer mit Leseschwäche oder Sehbbeeinträchtigung, die größere Abstände zur besseren Lesbarkeit einstellen, verlieren Inhalte oder können sie nicht mehr vollständig lesen.

Einstufung	Zuständig	Aufwand	Umsetzungspriorität
Mittel	Entwicklung	Mittlerer 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/text-spacing.html

MASSNAHMENBEWERTUNG

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

Empfehlung

Textcontainer flexibel gestalten (min-height statt fester Höhe, kein `overflow:hidden` auf Textbereichen) und mit den WCAG-Mindestabständen testen.

× Falsch

```
<div style="height: 40px; overflow: hidden;">Langer Textinhalt... </div>
```

✓ Richtig

```
<div style="min-height: 40px; overflow: visible;">Langer Textinhalt... </div>
```

[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.mfp-content	Dialog contains no focusable elements — keyboard users cannot interact with it.

Fundstelle: div.mfp-content

Node 4421

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

HTML `<div class="mfp-content" role="dialog" aria-modal="true" aria-labelledby="_seo6vtkbi"><section id="cookieba...`

DOM-Pfad `div.mfp-wrap:nth-of-type(2) > div.mfp-container > div.mfp-content:nth-of-type(1)`

Häufige Muster

Muster	Vorkommen
div.mfp-content	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.mfp-wrap	Focusable element without interactive role

Fundstelle: div.mfp-wrap

Node 4419

Hinweis **Focusable element without interactive role**

HTML `<div class="mfp-wrap mfp-auto-cursor mfp-ready" tabindex="-1" style="overflow: hidden auto;"><div class="mf...`

DOM-Pfad `div.mfp-wrap:nth-of-type(2)`



Figure 3: Element auf der Seite (Desktop-Ansicht)

Vorgeschlagene Code-Korrektur

```
<!-- Add tabindex="0" and a keydown handler to make this element keyboard-operable --> <div class="mfp-wrap mfp-auto-cursor mfp-ready" tabindex="-1" style="overflow: hidden auto...
```

Häufige Muster

Muster	Vorkommen
div.mfp-wrap	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

4 kleinere interaktive Befunde — keine kritischen oder hohen Barrieren erkannt.

MEDIUM Fokus-Indikator (x3)

Element (div > div > div:nth-of-type(5) > div > p > button:nth-of-type(1)) zeigt im fokussierten Zustand keinen sichtbaren Fokusindikator (kein Outline, kein Box-Shadow, keine Rahmenänderung). Tastaturnutzer verlieren die Orientierung. — Eine CSS-:focus-visible-Regel mit deutlichem Outline, Box-Shadow oder Rahmenwechsel gegenüber dem unfokussierten Zustand ergänzen.

MEDIUM Fokus-Indikator (x3)

Element (div > div > div:nth-of-type(5) > div > p > button:nth-of-type(2)) zeigt im fokussierten Zustand keinen sichtbaren Fokusindikator (kein Outline, kein Box-Shadow, keine Rahmenänderung). Tastaturnutzer verlieren die Orientierung. — Eine CSS-:focus-visible-Regel mit deutlichem Outline, Box-Shadow oder Rahmenwechsel gegenüber dem unfokussierten Zustand ergänzen.

MEDIUM Fokus-Indikator (x3)

Element (body#Start > div:nth-of-type(2)) zeigt im fokussierten Zustand keinen sichtbaren Fokusindikator (kein Outline, kein Box-Shadow, keine Rahmenänderung). Tastaturnutzer verlieren die Orientierung. — Eine CSS-:focus-visible-Regel mit deutlichem Outline, Box-Shadow oder Rahmenwechsel gegenüber dem unfokussierten Zustand ergänzen.

MEDIUM Tab-Reihenfolge

Die Tab-Reihenfolge weicht von der DOM-Reihenfolge ab: 3 rückwärtige Sprünge beobachtet. Zuerst betroffene Elemente: div > div > div > div:nth-of-type(1) > p:nth-of-type(1) > a, div > div > div > div:nth-of-type(1) > p:nth-of-type(1) > a, div > div > div > div:nth-of-type(1) > p:nth-of-type(1) > a. Tastaturnutzer können dem Lesefluss möglicherweise nicht folgen. — Negative oder hohe tabindex-Werte vermeiden. Lese-/DOM-Reihenfolge an die visuelle Reihenfolge angleichen.

Geprüfte Journey-Sequenzen

tab_walk 26 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
100 / 100	0 / 100	100 / 100	87	8

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
Kein Header-Bereich (banner-Landmark) vorhanden. Screen Reader können die Seitenstruktur nicht vollständig navigieren.	Mittel
Keine Navigations-Landmark vorhanden. Tastaturnutzer können nicht direkt zur Navigation springen.	Mittel
Keine Fußzeilen-Landmark (contentinfo) vorhanden. Die Seitenstruktur ist für Screen Reader unvollständig.	Mittel
Kein Main-Landmark in der Screenreader-Landmarkliste erkennbar.	Mittel

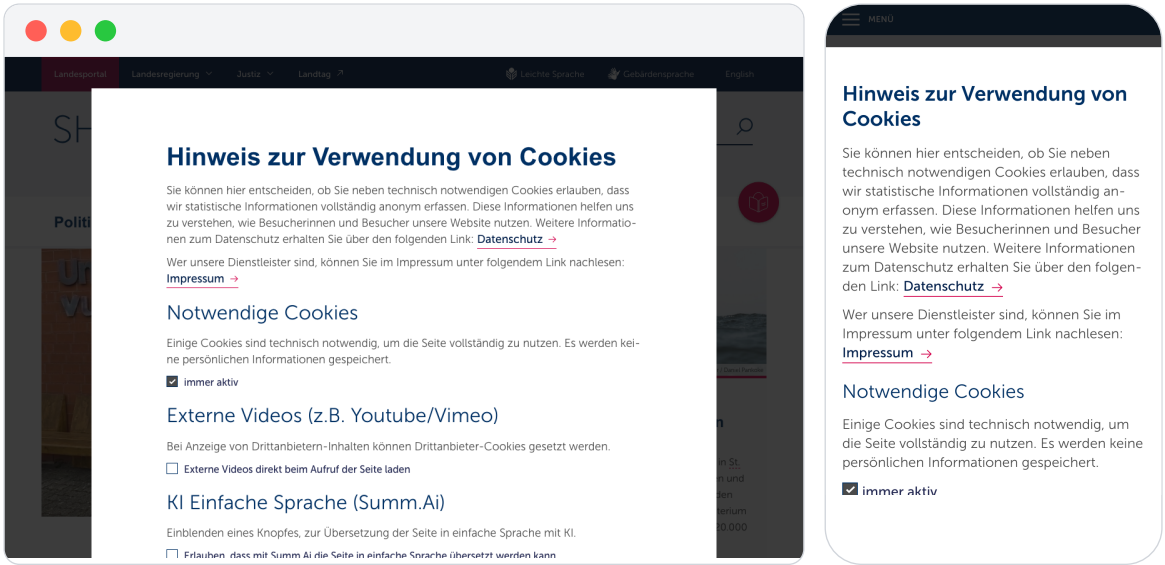
03

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
17 / 100 Kritischer Bereich	17 / 100 Kritischer Bereich	17 / 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: 159 auf Desktop und 152 auf Mobile. Die normalisierte gemeinsame Befundliste enthält 204 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	0
deutanopia	0	0	0
tritanopia	0	0	0
achromatopsia	0	0	0

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

Schwach · 0–100

0

Kritisch

Signal	Status	Detail
Benannte Bedienelemente	×	25 Elemente ohne zugänglichen Namen
Keine kritischen Fehler	×	1 kritische Verstöße

Autorität

Uneinheitlich · 0–100

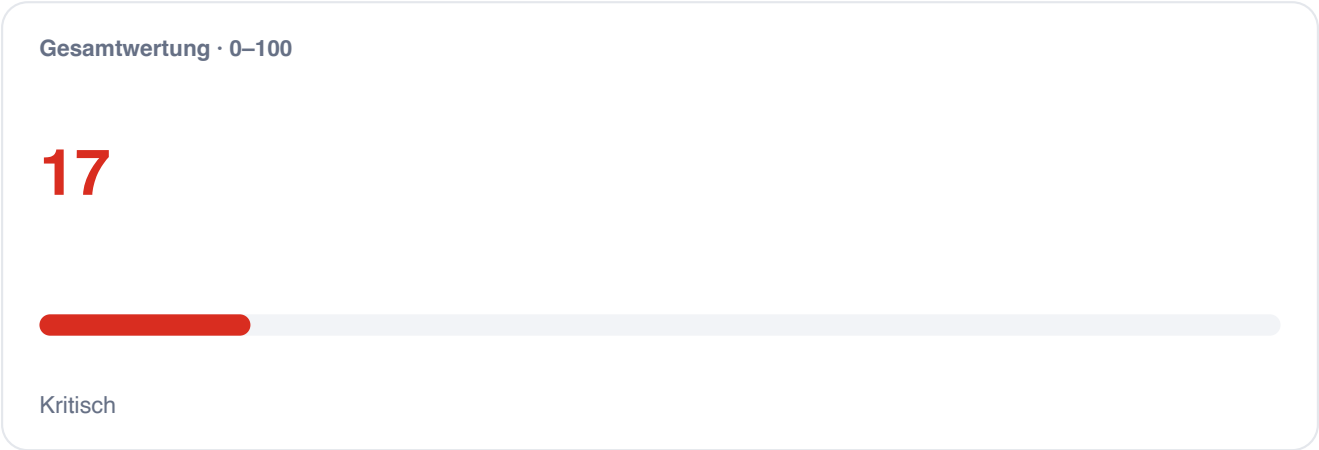
50

Ausbaufähig

Signal	Status	Detail
HTTPS	✓	Verschlüsselte Verbindung

Signal	Status	Detail
Barrierefreiheit	×	Accessibility-Score: 15 (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**● Modaler Dialog**

Vorhanden: 1 dialog detected, 0 structurally well-formed (accessible name, aria-modal, focusable descendants).

Anhang & Methodik

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

Prinzip-Abdeckung (bestandene Kriterien)

Wahrnehmbar	15/18 (83 %)
Bedienbar	22/26 (85 %)
Verständlich	9/9 (100 %)
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>).
- 1.3.2 · Hinweis

If the order carries meaning, reorder the elements in the DOM/source instead of using CSS `order`, or add `aria-owns` on the container listing the children in their visual order so assistive technology announces the same sequence.
- 1.3.2 · Hinweis

If the order carries meaning, reorder the elements in the DOM/source instead of using CSS `order`, or add `aria-owns` on the container listing the children in their visual order so assistive technology announces the same sequence.
- 1.3.2 · Hinweis

If the order carries meaning, reorder the elements in the DOM/source instead of using CSS `order`, or add `aria-owns` on the container listing the children in their visual order so assistive technology announces the same sequence.
- 1.3.2 · Hinweis

If the order carries meaning, reorder the elements in the DOM/source instead of using CSS `order`, or add `aria-owns` on the container listing the children in their visual order so assistive technology announces the same sequence.
- 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: 1 von 1 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	17 / 100 — 70 % Mobile (17) + 30 % Desktop (17)
Audit	Gesamtscore	17 / 100 — beitragende Gewichtung: Barrierefreiheit 100 % Separat ausgewiesene Indikator-Module: Dark Mode, KI-Sichtbarkeit, Quellenqualität.
Audit	Zählweise	12 unterschiedliche WCAG-Befundgruppen, 204 WCAG-Vorkommen; 12 Befundzeilen und 204 Vorkommen über alle Kategorien
Audit	WCAG-Level	AA
Audit	Geprüfte Knoten	3136
Audit	Laufzeit	14.1 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	17 / 100 — Kritisch — wesentliche Anforderungen an die Barrierefreiheit werden nicht erfüllt.. 4 Problemgruppe(n) mit hoher Auswirkung
Finding	a11y.aria_hidden_focus.invalid (aria-hidden-focus)	95 Vorkommen — Tastaturnutzer steuern auf unsichtbare Elemente (Fokus-Stolperfallen), während Screenreader-Nutzer keinerlei Rückmeldung über dieses Element erhalten.
Finding	a11y.parsing.invalid (4.1.1)	22 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.structure.missing (1.3.1)	25 Vorkommen — Nutzer mit Screenreader können Tabellen, Listen und Formulargruppen nicht korrekt navigieren.
Finding	a11y.presentation_semantic_children_invalid (1.3.1)	20 Vorkommen — Screenreader können Navigationsmenüs nicht korrekt interpretieren.

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)

- **aria-hidden-focus — Fokussierbares Element versteckt vor Screenreadern**

Focusable element 'header.header > nav.navskip > ul.navSkip > li:nth-of-type(1) > a' is inside an aria-hidden subtree — keyboard users reach it but screen readers cannot announce it
- **4.1.1 — Fehlerhaftes Markup (Parsing)**

Duplicate id='nn-9c9016d7-bca4-4594-9a4a-b69dafeb4f1f-31541551-7176-44b2-a030-2616d31586f0' appears 2 times in the DOM. Duplicate IDs cause AT to resolve references incorrectly.
- **1.3.1 — Fehlende semantische Struktur**

Element with role 'heading' is not contained within a landmark region
- **1.3.1 — Semantische Kinder in presentational Container**

Element with role="presentation"/"none" contains semantic child listitem (li.c-nav-primary__item:nth-of-type(5) > div.c-mobile-nav.js-mobile-nav > div.c-mobile-nav__wrapper:nth-of-type(2) > ul.c-mobile-nav__primary.c-mobile-nav__list:nth-of-type(1) > li.c-mobile-nav__item.c-mobile-nav__item--level1:nth-of-type(1))
- **4.1.2 — Fehlende Name/Rolle bei Bedienelementen**

Tab is missing selected state indication
- **2.5.8 — Unzureichende Klickzielgröße**

Interactive target '1 / 6' is 10x10 CSS pixels, below the 24x24 minimum.
- **4.1.2 — Ungültiger Wert eines ARIA-Attributs**

'aria-labelledby' has invalid value "cbh", expected: existing DOM id: cbh not found
- **1.4.13 — Unzugängliche Hover- oder Fokus-Inhalte (Tooltips)**

Interactive element uses the native `title` attribute as its only descriptive text (input#9fcf45a0-141c-4426-98cc-863b2de29743d30815f5d-6769-4dfc-9c77-a78210f01a69-mobile). Tooltips from `title` are not keyboard-accessible and unreliable on touch devices.
- **2.1.2 — Tastaturfalle — Fokus kann nicht verlassen werden**

Potential keyboard trap detected (modal dialog)
- **1.4.12 — Inhalt bei erhöhtem Zeilen-/Wortabstand abgeschnitten**

Content is clipped by 'div.c-teaser-service__wrapper' when WCAG-minimum text spacing (line-height 1.5x, letter-spacing 0.12em, word-spacing 0.16em, paragraph spacing 2x) is applied.
- **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