



Barrierefreiheit



Best Practices



SEO

▲ 0-49

■ 50-89

● 90-100



Barrierefreiheit

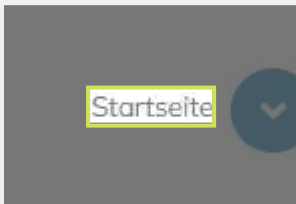
Mit diesen Prüfungen wirst du auf Möglichkeiten hingewiesen, [mit denen du die Barrierefreiheit deiner Web-App verbessern kannst](#). Durch die automatische Erkennung werden nicht alle Probleme erkannt. Auch die Barrierefreiheit deiner Web-App kann damit nicht garantiert werden, weshalb [manuelle Tests](#) empfohlen werden.

KONTRAST

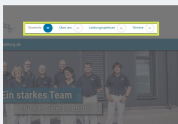
▲ Das Kontrastverhältnis von Hintergrund- und Vordergrundfarben ist nicht ausreichend. ^

Text mit geringem Kontrast ist für viele Nutzer schlecht oder gar nicht lesbar. [Informationen zu einem ausreichenden Farbkontrast](#).

Fehlerhafte Elemente



a



nav#main-menu.menu

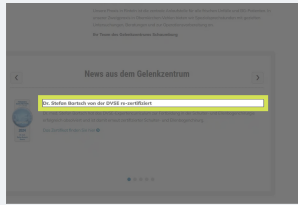


h2

Fehlerhafte Elemente



body



h3



div.newsBox



a



div.footer_bottom

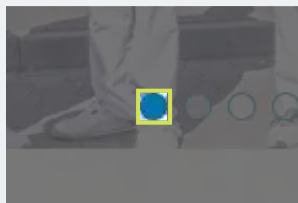
Anhand dieser Möglichkeiten kannst du die Lesbarkeit deiner Inhalte verbessern.

BEST PRACTICES

- ▲ Berührungszielbereiche haben eine unzureichende Größe oder einen unzureichenden Abstand. ^

Berührungszielbereiche mit ausreichender Größe und ausreichend Abstand können Nutzern, die eventuell Schwierigkeiten haben, kleine Steuerelemente zu aktivieren, die Bedienung erleichtern. [Weitere Informationen zu Berührungszielbereichen](#)

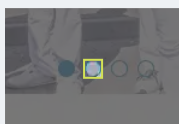
Fehlerhafte Elemente



li

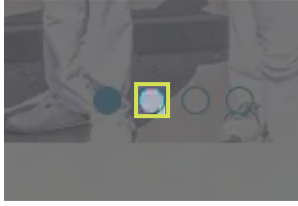


a



li

Fehlerhafte Elemente



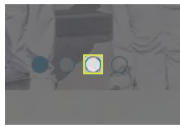
li



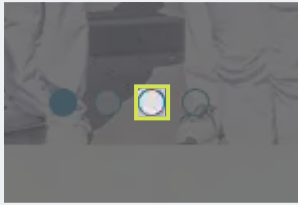
a



li



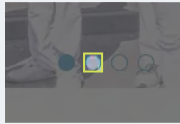
li



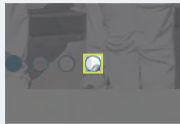
li



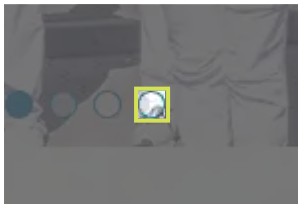
a



li



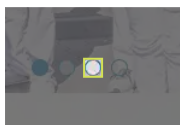
li.cameracurrent



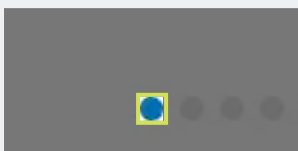
li.cameracurrent



a

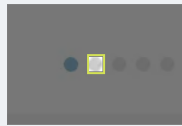


li

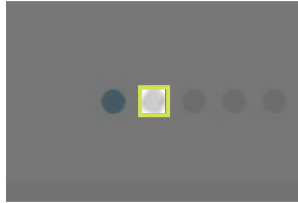


li

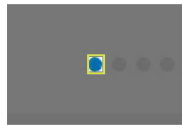
Fehlerhafte Elemente



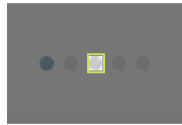
li.active



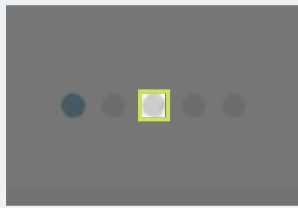
li.active



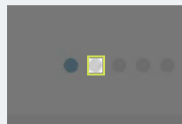
li



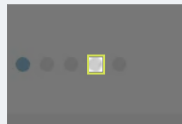
li



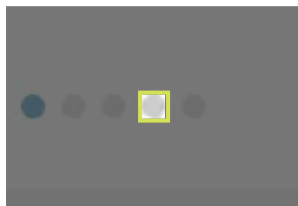
li



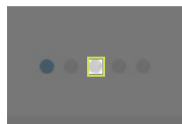
li.active



li



li

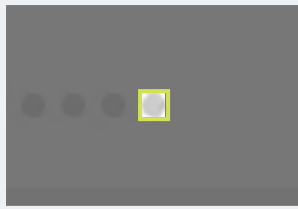


li

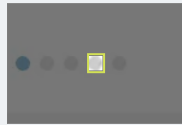


li

Fehlerhafte Elemente



li



li

Hier findest du häufig genutzte Best Practices für Barrierefreiheit.

ZUSÄTZLICHE ELEMENTE ZUR MANUELLEN ÜBERPRÜFUNG (10)

Ausblenden

 Interactive controls are keyboard focusable ^

Custom interactive controls are keyboard focusable and display a focus indicator. [Learn how to make custom controls focusable.](#) Nicht bewertet

 Interactive elements indicate their purpose and state ^

Interactive elements, such as links and buttons, should indicate their state and be distinguishable from non-interactive elements. [Learn how to decorate interactive elements with affordance hints.](#) Nicht bewertet

 The page has a logical tab order ^

Tabbing through the page follows the visual layout. Users cannot focus elements that are offscreen. [Learn more about logical tab ordering.](#) Nicht bewertet

 Visual order on the page follows DOM order ^

DOM order matches the visual order, improving navigation for assistive technology. [Learn more about DOM and visual ordering.](#) Nicht bewertet

 User focus is not accidentally trapped in a region ^

A user can tab into and out of any control or region without accidentally trapping their focus. [Learn how to avoid focus traps.](#) Nicht bewertet

 The user's focus is directed to new content added to the page ^

If new content, such as a dialog, is added to the page, the user's focus is directed to it. [Learn how to direct focus to new content.](#) Nicht bewertet

HTML5 landmark elements are used to improve navigation ^

Landmark elements (<main>, <nav>, etc.) are used to improve the keyboard navigation of the page for assistive technology. [Learn more about landmark elements.](#) Nicht bewertet

Offscreen content is hidden from assistive technology ^

Offscreen content is hidden with display: none or aria-hidden=true. [Learn how to properly hide offscreen content.](#) Nicht bewertet

Custom controls have associated labels ^

Custom interactive controls have associated labels, provided by aria-label or aria-labelledby. [Learn more about custom controls and labels.](#) Nicht bewertet

Custom controls have ARIA roles ^

Custom interactive controls have appropriate ARIA roles. [Learn how to add roles to custom controls.](#) Nicht bewertet

Diese Prüfungen sind für Bereiche vorgesehen, für die automatische Testtools nicht geeignet sind. Weitere Informationen findest du in unserem Leitfaden zur [Durchführung einer Prüfung auf Barrierefreiheit.](#)

BESTANDENE PRÜFUNGEN (26)

Ausblenden

[aria-*]-Attribute entsprechen ihren Rollen ^

Jede ARIA-role unterstützt eine bestimmte Untergruppe von aria-*-Attributen. Wenn sie jedoch falsch zugeordnet sind, werden die aria-*-Attribute ungültig. [Informationen zum Zuordnen von ARIA-Attributen zu ihren Rollen.](#)

[aria-hidden="true"] ist in dem Dokument <body> nicht vorhanden ^

Hilfstechnologien wie Screenreader funktionieren nicht richtig, wenn für den <body> des Dokuments aria-hidden="true" festgelegt ist. [Informationen zu den Auswirkungen von aria-hidden auf den Textbereich des Dokuments.](#)

[role]-Elemente verfügen über alle erforderlichen [aria-*]-Attribute ^

Für einige ARIA-Rollen sind Attribute erforderlich, die Screenreadern den Zustand des Elements beschreiben. [Weitere Informationen zu Rollen und erforderlichen Attributen.](#)

Die Elemente mit einer ARIA-[role], deren untergeordnete Elemente eine bestimmte [role] enthalten müssen, haben alle erforderlichen untergeordneten Elemente. ^

Einige übergeordnete ARIA-Rollen müssen bestimmte untergeordnete Rollen enthalten, damit sie die beabsichtigten Hilfsfunktionen erfüllen können. [Weitere Informationen zu Rollen und erforderlichen untergeordneten Elementen.](#)

- [role]-Elemente sind ihren jeweils erforderlichen übergeordneten Elementen untergeordnet ^

Einige untergeordnete ARIA-Rollen müssen in bestimmten übergeordneten Rollen enthalten sein, damit sie die beabsichtigten Hilfsfunktionen erfüllen können. [Weitere Informationen zu ARIA-Rollen und erforderlichen übergeordneten Elementen.](#)

- [role]-Werte sind gültig ^

Für ARIA-Rollen müssen gültige Werte angegeben sein, damit sie die beabsichtigten Hilfsfunktionen erfüllen können. [Weitere Informationen zu gültigen ARIA-Rollen.](#)

- [aria-*]-Attribute weisen gültige Werte auf ^

Hilfstechnologien wie Screenreader können ARIA-Attribute mit ungültigen Werten nicht interpretieren. [Weitere Informationen zu gültigen Werten für ARIA-Attribute.](#)

- [aria-*]-Attribute sind gültig und richtig geschrieben ^

Hilfstechnologien wie Screenreader können ARIA-Attribute mit ungültigen Namen nicht interpretieren. [Weitere Informationen zu gültigen ARIA-Attributen.](#)

- Die Namen der Schaltflächen sind für Screenreader zugänglich ^

Wenn eine Schaltfläche keinen barrierefreien Namen hat, wird sie von Screenreadern nur als „Schaltfläche“ angesagt. Dadurch ist sie für Nutzer, die auf Screenreader angewiesen sind, unbrauchbar. [Informationen zum barrierefreien Gestalten von Schaltflächen.](#)

- Bildelemente verfügen über [alt]-Attribute ^

Für informative Elemente sollte ein kurzer, beschreibenden alternativer Text verwendet werden. Dekorative Elemente können mit einem leeren ALT-Attribut ignoriert werden. [Weitere Informationen zum Attribut alt.](#)

- [user-scalable="no"] wird nicht im <meta name="viewport">-Element verwendet und das [maximum-scale]-Attribut ist nicht kleiner als 5. ^

Wenn du die Zoomfunktion deaktivierst, können Nutzer mit eingeschränktem Sehvermögen, die auf die Bildschirmvergrößerung angewiesen sind, den Inhalt einer Webseite nicht richtig sehen. [Weitere Informationen zum Darstellungsbereich-Meta-Tag.](#)

- ARIA-Attribute werden wie angegeben für die Rolle des Elements verwendet ^

Einige ARIA-Attribute sind nur unter bestimmten Umständen für ein Element zulässig. [Weitere Informationen zu bedingten ARIA-Attributen.](#)

● [aria-hidden="true"]-Elemente enthalten keine fokussierbaren Unterelemente ^

Fokussierbare Nachfolgerelemente in einem [aria-hidden="true"]-Element führen dazu, dass Nutzer von Hilfstechnologien wie Screenreadern solche interaktiven Elemente nicht verwenden können. [Informationen zu den Auswirkungen von aria-hidden auf fokussierbare Elemente.](#)

● Elemente verwenden nur zulässige ARIA-Attribute ^

Die Verwendung von ARIA-Attributen in Rollen, für die sie nicht zulässig sind, kann bedeuten, dass wichtige Informationen nicht an Nutzer von Hilfstechnologien weitergegeben werden. [Weitere Informationen zu unzulässigen ARIA-Rollen](#)

● Dokument enthält ein <title>-Element ^

Der Titel gibt Screenreader-Nutzern einen Überblick über die Seite. Nutzer von Suchmaschinen verlassen sich stark auf ihn, um zu entscheiden, ob eine Seite für ihre Suche relevant ist. [Weitere Informationen zu Dokumenttiteln.](#)

● <html>-Element hat ein [lang]-Attribut ^

Wenn für eine Seite kein lang-Attribut angegeben ist, nehmen Screenreader an, dass sie in der Standardsprache vorliegt, die der Nutzer beim Einrichten des Screenreaders ausgewählt hat. Ist das nicht der Fall, gibt der Screenreader den Inhalt der Seite möglicherweise falsch aus. [Weitere Informationen zum lang-Attribut.](#)

● Das <html>-Element hat einen gültigen Wert für sein [lang]-Attribut ^

Wenn ein gültiger [BCP-47-Sprachcode](#) angegeben wird, kann der Text von einem Screenreader korrekt wiedergegeben werden. [Informationen zur Verwendung des Attributs lang.](#)

● Links sind ohne Farbe erkennbar. ^

Text mit geringem Kontrast ist für viele Nutzer schlecht oder gar nicht lesbar. Gut erkennbarer Linktext erhöht die Nutzerfreundlichkeit für Nutzer mit eingeschränktem Sehvermögen. [Informationen dazu, wie Links gut erkennbar gemacht werden](#)

● Links haben einen leicht erkennbaren Namen ^

Linktext, der erkennbar, einzigartig und fokussierbar ist, erleichtert Screenreader-Nutzern die Verwendung. Dies gilt auch für alternativen Text für Bilder, die als Links verwendet werden. [Informationen zu barrierefreien Links.](#)

● Listen enthalten nur -Elemente und Elemente zur Skriptunterstützung (<script> sowie <template>). ^

Screenreader sagen Listen auf bestimmte Art und Weise an. Wenn die Liste richtig strukturiert ist, kann der Screenreader sie besser ausgeben. [Weitere Informationen zum Strukturieren von Listen.](#)

● Listenelemente () befinden sich in übergeordneten -, - oder <menu>-Elementen ^

Listenelemente () müssen sich in einem übergeordneten -, - oder <menu>-Element befinden, damit sie von Screenreadern richtig angesagt werden können. [Weitere Informationen zum Strukturieren von Listen.](#)

- Kein Element hat einen `[tabindex]`-Wert größer als 0 ^

Ein Wert größer als 0 impliziert eine explizite Navigationsanordnung. Das ist zwar technisch möglich, aber für Nutzer, die auf Hilfstechnologien angewiesen sind, häufig frustrierend. [Weitere Informationen zum Attribut `tabindex`.](#)

- Überschriftenelemente werden in einer fortlaufenden absteigenden Reihenfolge angezeigt ^

Richtig geordnete Überschriften, die keine Ebenen überspringen, geben der Seite eine semantische Struktur. Nutzer von Hilfstechnologien können sich so leichter auf der Seite zurechtfinden und die Inhalte besser verstehen. [Weitere Informationen zur Reihenfolge von Überschriften.](#)

- Sprunglink sind fokussierbar. ^

Ein Sprunglink kann Nutzern dabei helfen, zum Hauptinhalt zu springen und so Zeit zu sparen. [Weitere Informationen zu Sprunglinks](#)

- Dokument hat eine Hauptmarkierung. ^

Eine Hauptmarkierung hilft Nutzern von Screenreadern, Webseiten zu bedienen. [Weitere Informationen zu Markierungen](#)

- Es wurden keine verworfenen ARIA-Rollen verwendet ^

Verworfenen ARIA-Rollen werden möglicherweise nicht richtig von Hilfstechnologien verarbeitet. [Weitere Informationen zu verworfenen ARIA-Rollen.](#)

NICHT ZUTREFFEND (32)

Ausblenden

- `[accesskey]`-Werte sind eindeutig ^

Mithilfe von Tastenkombinationen können Nutzer schnell den Fokus auf einen Bereich der Seite verschieben. Damit die Navigation richtig funktioniert, darf jede Tastenkombination nur einmal vergeben sein. [Weitere Informationen zu Tastenkombinationen.](#) Nicht bewertet

- `button`-, `link`- und `menuitem`-Elemente haben zugängliche Namen ^

Wenn ein Element keinen barrierefreien Namen hat, wird es von Screenreadern mit einer allgemeinen Bezeichnung angesagt. Dadurch ist es für Nutzer, die auf Screenreader angewiesen sind, unbrauchbar. [Informationen zum barrierefreien Gestalten von Befehlselementen.](#) Nicht bewertet

- Elemente mit `role="dialog"` oder `role="alertdialog"` haben barrierefreie Namen. ^

ARIA-Dialogelemente ohne barrierefreie Namen können Nutzer von Screenreadern daran hindern, den Zweck dieser Elemente zu erkennen. [Informationen zum Verbessern der Barrierefreiheit von ARIA-Dialogelementen](#) Nicht bewertet

ARIA-Eingabefelder haben zugängliche Namen ^

Wenn ein Eingabefeld keinen barrierefreien Namen hat, wird es von Screenreadern mit einer allgemeinen Bezeichnung angesagt. Dadurch ist es für Nutzer, die auf Screenreader angewiesen sind, unbrauchbar. [Weitere Informationen zu Labels für Eingabefelder](#). Nicht bewertet

ARIA `meter`-Elemente haben zugängliche Namen ^

Wenn ein Messtool-Element keinen barrierefreien Namen hat, wird es von Screenreadern mit einer allgemeinen Bezeichnung angesagt. Dadurch ist es für Nutzer, die auf Screenreader angewiesen sind, unbrauchbar. [Weitere Informationen zum Benennen von meter-Elementen](#) Nicht bewertet

ARIA `progressbar`-Elemente haben zugängliche Namen ^

Wenn ein progressbar-Element keinen barrierefreien Namen hat, wird es von Screenreadern mit einer allgemeinen Bezeichnung angesagt. Dadurch ist es für Nutzer, die auf Screenreader angewiesen sind, unbrauchbar. [Informationen zum Kennzeichnen von progressbar-Elementen](#). Nicht bewertet

Elemente mit dem Attribut „`role=text`“ haben keine fokussierbaren Nachfolgerelemente. ^

Wenn du einen Textknoten, der durch Markup aufgeteilt ist, mit `role=text` auszeichnest, kann VoiceOver diesen als eine Wortgruppe behandeln. Die fokussierbaren Nachfolgerelemente des Elements werden jedoch nicht angesagt. [Weitere Informationen zum Attribut `role=text`](#) Nicht bewertet

ARIA-Ein-/Aus-Schaltflächen haben zugängliche Namen ^

Wenn eine Ein-/Aus-Schaltfläche keinen barrierefreien Namen hat, wird sie von Screenreadern mit einer allgemeinen Bezeichnung angesagt. Dadurch ist sie für Nutzer, die auf Screenreader angewiesen sind, unbrauchbar. [Weitere Informationen zu Ein-/Aus-Schaltflächen](#). Nicht bewertet

ARIA `tooltip`-Elemente haben zugängliche Namen ^

Wenn ein Kurzinfo-Element keinen barrierefreien Namen hat, wird es von Screenreadern mit einer allgemeinen Bezeichnung angesagt. Dadurch ist es für Nutzer, die auf Screenreader angewiesen sind, unbrauchbar. [Weitere Informationen zum Benennen von tooltip-Elementen](#) Nicht bewertet

ARIA `treeitem`-Elemente haben zugängliche Namen ^

Wenn ein `treeitem`-Element keinen barrierefreien Namen hat, wird es von Screenreadern mit einer allgemeinen Bezeichnung angesagt. Dadurch ist es für Nutzer, die auf Screenreader angewiesen sind, unbrauchbar. [Weitere Informationen zum Kennzeichnen von treeitem-Elementen](#). Nicht bewertet

- Die Seite enthält eine Überschrift, einen Link zum Überspringen oder einen Landmark-Bereich ^

Wenn Tastaturnutzer Inhalte umgehen können, die sich wiederholen, sorgt das für eine effizientere Navigation. [Weitere Informationen zum Umgehen von Blockierungen.](#) Nicht bewertet

- `<d1>`-Elemente enthalten ausschließlich Gruppen aus `<dt>`- und `<dd>`-Elementen sowie `<script>`-, `<template>`- oder `<div>`-Elemente, die richtig angeordnet sind. ^

Wenn Definitionslisten nicht korrekt mit Markup versehen sind, kann es zu verwirrenden oder ungenauen Screenreader-Ausgaben kommen. [Informationen zum Strukturieren von Definitionslisten.](#) Nicht bewertet

- Definitionlistenelemente sind in `<d1>`-Elemente eingefasst ^

Definitionlistenelemente (`<dt>` und `<dd>`) müssen in ein übergeordnetes `<d1>`-Element eingefasst sein, damit sie von Screenreadern richtig angesagt werden können. [Informationen zum Strukturieren von Definitionslisten.](#) Nicht bewertet

- ARIA-IDs sind eindeutig ^

Der Wert einer ARIA-ID muss eindeutig sein, damit andere Instanzen nicht von Hilfstechnologien übersehen werden. [Informationen zum Korrigieren doppelter ARIA-IDs.](#) Nicht bewertet

- Kein Formularfeld hat mehrere Labels ^

Formularfelder mit mehreren Labels werden von Hilfstechnologien wie Screenreadern unter Umständen missverständlich angesagt, da sie entweder das erste, das letzte oder alle Labels verwenden. [Informationen zur Verwendung von Formularlabels.](#) Nicht bewertet

- `<frame>`- oder `<iframe>`-Elemente verfügen über einen Titel ^

Screenreader-Nutzer sind auf Frametitel angewiesen, die die Inhalte von Frames beschreiben. [Weitere Informationen zu Frametiteln.](#) Nicht bewertet

- Das `<html>`-Element hat ein `[xml:lang]`-Attribut mit derselben Basissprache wie das `[lang]`-Attribut. ^

Wenn für die Webseite keine konsistente Sprache angegeben ist, kündigt der Screenreader den Text der Seite möglicherweise nicht richtig an. [Weitere Informationen zum Attribut „lang“](#) Nicht bewertet

- Eingabeschaltflächen haben erkennbaren Text. ^

Das Hinzufügen von erkennbarem, zugänglichem Text zu Eingabeschaltflächen hilft den Nutzern von Screenreadern möglicherweise, den Zweck der entsprechenden Schaltfläche zu verstehen. [Weitere Informationen zu Eingabeschaltflächen](#) Nicht bewertet

- `<input type="image">`-Elemente haben `[alt]`-Text ^

Wenn ein Bild als `<input>`-Schaltfläche verwendet wird, kann alternativer Text Screenreader-Nutzern helfen, den Zweck der Schaltfläche besser zu verstehen. [Informationen zum Alt-Text für Eingabebilder](#). Nicht bewertet

- Formularelemente sind mit Labels verknüpft ^

Durch Labels wird gewährleistet, dass Steuerelemente für Formulare von Hilfstechnologien wie Screenreadern richtig angesagt werden. [Weitere Informationen zu Labels für Formularelemente](#). Nicht bewertet

- Dieses Dokument verwendet `<meta http-equiv="refresh">` nicht ^

Nutzer rechnen nicht damit, dass eine Seite automatisch aktualisiert wird. Außerdem wird dadurch der Fokus wieder auf den Seitenanfang verschoben. Das kann für den Nutzer frustrierend oder irritierend sein. [Weitere Informationen zum Meta-Tag „Refresh“](#). Nicht bewertet

- `<object>`-Elemente haben alternativen Text ^

Screenreader können lediglich Textinhalte interpretieren. Wenn du `<object>`-Elementen alternativen Text hinzufügst, können Screenreader-Nutzer besser verstehen, was diese Elemente darstellen. [Weitere Informationen zum Alt-Text für object-Elemente](#). Nicht bewertet

- `select`-Elemente haben zugehörige `label`-Elemente. ^

`form`-Elemente ohne wirkungsvolle Labels können für Nutzer von Screenreadern frustrierend sein. [Weitere Informationen zum Element „select“](#) Nicht bewertet

- Zellen in einem "`<table>`"-Element, die das Attribut "`[headers]`" enthalten, verweisen auf Zellen in derselben Tabelle. ^

Screenreader bieten Funktionen, die die Navigation in Tabellen vereinfachen. Wenn du dafür sorgst, dass `<td>`-Zellen, die das Attribut `[headers]` verwenden, nur auf andere Zellen in derselben Tabelle verweisen, kann dies für Screenreader-Nutzer hilfreich sein. [Weitere Informationen zum Attribut headers](#). Nicht bewertet

- Für `<th>`-Elemente und Elemente mit `[role="columnheader"/"rowheader"]` sind Datenzellen vorhanden, die sie beschreiben. ^

Screenreader bieten Funktionen, die die Navigation in Tabellen vereinfachen. Wenn du dafür sorgst, dass Tabellenüberschriften immer auf bestimmte Zellen verweisen, kann dies für Screenreader-Nutzer hilfreich sein. [Weitere Informationen zu Tabellenüberschriften](#). Nicht bewertet

- `[lang]`-Attribute weisen einen gültigen Wert auf ^

Wenn ein gültiger [BCP-47-Sprachcode](#) für Elemente angegeben wird, kann der Text besser von Screenreadern ausgesprochen werden. [Informationen zur Verwendung des Attributs lang](#). Nicht bewertet

- `<video>`-Elemente enthalten ein `<track>`-Element mit `[kind="captions"]` ^

Wenn ein Video Untertitel enthält, können Menschen mit Hörbehinderung die Informationen im Video besser verstehen.

[Weitere Informationen zu Untertiteln in Videos.](#) Nicht bewertet

- Tabellen haben unterschiedliche Inhalte im Zusammenfassungsattribut und in „<caption>“. ^

Das Zusammenfassungsattribut sollte die Tabellenstruktur beschreiben, während „<caption>“ den Titel auf dem Bildschirm haben sollte. Korrektes Tabellen-Markup ist für Nutzer von Screenreadern hilfreich. [Weitere Informationen zu](#)

[Zusammenfassung und Bildunterschrift](#) Nicht bewertet

- Alle heading-Elemente haben einen Inhalt. ^

Eine Überschrift ohne Inhalt oder nicht zugänglicher Text verhindert, dass Nutzer von Screenreadern auf Informationen in der Seitenstruktur zugreifen. [Weitere Informationen zu Überschriften](#) Nicht bewertet

- ARIA-Rollen werden nur für kompatible Elemente verwendet ^

Vielen HTML-Elementen können nur bestimmte ARIA-Rollen zugewiesen werden. Die Verwendung von ARIA-Rollen für Elemente, für die sie nicht zulässig sind, kann die Barrierefreiheit der Webseite beeinträchtigen. [Weitere Informationen zu](#)

[ARIA-Rollen](#) Nicht bewertet

- Bildelemente haben keine „[alt]“-Attribute, bei denen es sich um redundanten Text handelt. ^

Für informative Elemente sollte ein kurzer und beschreibender alternativer Text verwendet werden. Alternativer Text, der mit dem Text neben dem Link oder Bild identisch ist, kann für Nutzer von Screenreadern verwirrend sein, da der Text zweimal vorgelesen wird. [Weitere Informationen zum Attribut „alt“](#) Nicht bewertet

- Identische Links haben denselben Zweck. ^

Links mit demselben Ziel sollten dieselbe Beschreibung haben, damit Nutzer den Zweck des Links besser verstehen und entscheiden können, ob sie ihm folgen möchten. [Weitere Informationen zu identischen Links](#) Nicht bewertet



Best Practices

VERTRAUEN UND SICHERHEIT

- Sicherstellen, dass CSP effektiv gegen XSS-Angriffe wirkt ^

Eine starke Content Security Policy (CSP) reduziert das Risiko für Cross-Site-Scripting-Angriffe (XSS-Angriffe) erheblich.

[Weitere Informationen](#) Nicht bewertet

Beschreibung	Anweisung	Schweregrad
Kein CSP im erzwungenen Modus gefunden		Hoch

Wirkungsvolle HSTS-Richtlinie verwenden ^

Durch die Implementierung des HSTS-Headers wird das Risiko von Downgrades bei HTTP-Verbindungen sowie von Abhörangriffen erheblich reduziert. Wir empfehlen eine schrittweise Einführung, bei der mit einem niedrigen „max-age“-Wert begonnen wird. [Hier findest du weitere Informationen zur Verwendung einer wirkungsvollen HSTS-Richtlinie.](#)

Nicht bewertet

Beschreibung	Anweisung	Schweregrad
Kein HSTS-Header gefunden		Hoch

Mit COOP für die richtige Isolation des Ursprungs sorgen ^

Mit der Cross-Origin-Opener-Policy (COOP) kann das übergeordnete Fenster von anderen Dokumenten wie Pop-ups isoliert werden. [Hier findest du weitere Informationen zur Implementierung eines COOP-Headers.](#) Nicht bewertet

Beschreibung	Anweisung	Schweregrad
Kein COOP-Header gefunden		Hoch

Clickjacking mit XFO oder CSP abschwächen ^

Der X-Frame-Options-Header (XFO) oder die frame-ancestors-Anweisung im Content-Security-Policy-Header (CSP) steuern, wo eine Seite eingebettet werden kann. Wenn das Einbetten der Seite auf einigen oder allen Websites blockiert wird, können dadurch Clickjacking-Angriffe abgeschwächt werden. [Weitere Informationen](#) Nicht bewertet

Beschreibung	Anweisung	Schweregrad
Keine Richtlinie zur Kontrolle von Frames gefunden		Hoch

DOM-basiertes XSS-Angriffe mit vertrauenswürdigen Typen reduzieren ^

Die require-trusted-types-for-Richtlinie im Content-Security-Policy-Header (CSP) weist User-Agents an, die an DOM-XSS-Senkenfunktionen übergebenen Daten zu steuern. [Weitere Informationen zum Reduzieren von DOM-basierten XSS-Angriffen mit vertrauenswürdigen Typen](#) Nicht bewertet

Beschreibung	Schweregrad
Kein `Content-Security-Policy`-Header mit der Richtlinie zu vertrauenswürdigen Typen gefunden	Hoch

ALLGEMEIN

○ JavaScript-Bibliotheken erkannt ^

Alle Frontend-JavaScript-Bibliotheken auf der Seite wurden erkannt. [Weitere Informationen](#). Nicht bewertet

Name	Version
Bootstrap	>= 2.0.0 & <= 2.3.2
jQuery	1.8.2
Modernizr	2.8.3
jQuery Mobile	

BESTANDENE PRÜFUNGEN (13)

[Ausblenden](#)

● Verwendet HTTPS ^

Alle Websites sollten durch HTTPS geschützt werden – selbst wenn sie keine sensiblen Daten enthalten. Auch [gemischte Inhalte](#), bei denen einige Ressourcen über HTTP geladen werden, obwohl die ursprüngliche Anfrage über HTTPS gestellt wurde, sind zu vermeiden. HTTPS verhindert, dass andere die Website manipulieren oder die Kommunikation zwischen deiner App und deinen Nutzern mitverfolgen können, und ist eine Voraussetzung für HTTP/2 und viele neue Webplattform-APIs. [Weitere Informationen zu HTTPS](#).

● Vermeidet veraltete APIs ^

Verworfen APIs werden aus dem Browser entfernt. [Weitere Informationen zu verworfenen APIs](#).

● Vermeidet Drittanbieter-Cookies ^

Drittanbieter-Cookies werden möglicherweise in bestimmten Kontexten blockiert. [Weitere Informationen zur Vorbereitung auf Einschränkungen für Drittanbieter-Cookies](#)

● Erlaubt Nutzern, Inhalte in Eingabefelder einzufügen ^

Es ist nicht nutzerfreundlich, das Einfügen von Eingaben zu verhindern. Außerdem werden dadurch Passwortmanager blockiert, was die Sicherheit beeinträchtigt. [Weitere Informationen zu nutzerfreundlichen Eingabefeldern](#).

● Fordert während des Seitenaufbaus keine Berechtigung zur Standortbestimmung an ^

Wenn Websites den Standort ohne Begründung anfordern, sind Nutzer schnell misstrauisch oder irritiert. Versuche stattdessen, die Anforderung mit einer Nutzeraktion zu verbinden. [Weitere Informationen zur Standortbestimmung.](#)

● Fordert während des Seitenaufbaus keine Benachrichtigungsberechtigung an ^

Wenn Websites die Berechtigung zum Senden von Benachrichtigungen ohne Begründung anfordern, sind Nutzer schnell misstrauisch oder irritiert. Versuche stattdessen, die Anforderung mit Touch-Gesten zu verbinden. [Weitere Informationen zum verantwortungsvollen Einholen der Berechtigung zum Senden von Benachrichtigungen.](#)

● Zeigt Bilder mit einem korrekten Seitenverhältnis an ^

Die Bildgröße sollte dem natürlichen Seitenverhältnis entsprechen. [Weitere Informationen zum Seitenverhältnis von Bildern.](#)

● Stellt Bilder mit angemessener Auflösung bereit ^

Die originalen Abmessungen eines Bildes sollten proportional zu der Displaygröße und dem Pixel-Verhältnis sein, damit das Bild optimal angezeigt wird. [Weitere Informationen zu responsiven Bildern.](#)

● Seite verfügt über den HTML-DOCTYPE ^

Wenn du einen DOCTYPE angibst, verhinderst du, dass der Browser zum Quirks-Modus wechselt. [Weitere Informationen zum Deklarieren eines DOCTYPE.](#)

● Korrekt definierter Zeichensatz ^

Die Zeichencodierung muss deklariert werden. Dazu kann ein <meta>-Tag in den ersten 1024 Byte des HTML-Codes oder im HTTP-Antwortheader „Content-Type“ angegeben werden. [Weitere Informationen zum Deklarieren der Zeichencodierung.](#)

● Es wurden keine Browserfehler in der Konsole protokolliert ^

In der Konsole protokollierte Fehler weisen auf ungelöste Probleme hin. Sie können durch fehlgeschlagene Netzwerkanfragen und andere Browserprobleme verursacht werden. [Weitere Informationen](#)

● Keine Probleme im **Issues**-Bereich der Chrome-Entwicklertools ^

Im Issues-Bereich der Chrome-Entwicklertools wurden ungelöste Probleme protokolliert. Sie können durch fehlgeschlagene Netzwerkanfragen, unzureichende Sicherheitsmaßnahmen und andere Browser-Probleme verursacht sein. Öffne den Bereich mit den Problemen in Chrome-Entwicklertools, um weitere Details zu den einzelnen Problemen aufzurufen.

● Seite hat gültige Quellzuordnungen ^

Quellzuordnungen übersetzen reduzierten Code in den ursprünglichen Quellcode. Dies hilft Entwicklern beim Debugging in der Produktionsphase. Zusätzlich kann Lighthouse weitere Informationen liefern. Wir empfehlen, Quellzuordnungen bereitzustellen, um diese Vorteile zu nutzen. [Weitere Informationen zu Quellzuordnungen](#). Nicht bewertet

NICHT ZUTREFFEND (1)

Ausblenden

- HTTP-Traffic wird auf HTTPS weitergeleitet ^

Achte darauf, den gesamten HTTP-Traffic zu HTTPS weiterzuleiten, damit sichere Webfunktionen für alle Nutzer aktiviert werden. [Weitere Informationen](#). Nicht bewertet

**SEO**

Mit diesen Prüfungen ist gewährleistet, dass bei deiner Seite grundlegende Tipps für die Suchmaschinenoptimierung berücksichtigt werden. Es gibt viele verschiedene Faktoren, die Lighthouse hier nicht bewertet und die sich auf das Such-Ranking deiner Seite auswirken können, einschließlich der [Core Web Vitals](#)-Performance. [Weitere Informationen zu Google Search Essentials](#)

ZUSÄTZLICHE ELEMENTE ZUR MANUELLEN ÜBERPRÜFUNG (1)

Ausblenden

- Strukturierte Daten sind gültig ^

Du kannst das [Testtool für strukturierte Daten](#) ausführen, um strukturierte Daten zu validieren. [Weitere Informationen zu strukturierten Daten](#) Nicht bewertet

Du kannst diese zusätzlichen Validierungen für deine Website ausführen, um weitere Best Practices für die SEO zu prüfen.

BESTANDENE PRÜFUNGEN (8)

Ausblenden

- Seite ist nicht von Indexierung ausgeschlossen ^

Suchmaschinen können deine Seiten nicht in die Suchergebnisse aufnehmen, wenn sie nicht berechtigt sind, sie zu crawlen. [Weitere Informationen zu Crawler-Anweisungen](#).

- Dokument enthält ein `<title>`-Element ^

Der Titel gibt Screenreader-Nutzern einen Überblick über die Seite. Nutzer von Suchmaschinen verlassen sich stark auf ihn, um zu entscheiden, ob eine Seite für ihre Suche relevant ist. [Weitere Informationen zu Dokumenttiteln](#).

Dokument enthält eine Meta-Beschreibung ^

Meta-Beschreibungen können in die Suchergebnisse aufgenommen werden, um die Seiteninhalte kurz zusammenzufassen.

[Weitere Informationen zu Meta-Beschreibungen.](#)

Seite hat einen gültigen HTTP-Statuscode ^

Seiten mit ungültigen HTTP-Statuscodes werden möglicherweise nicht richtig indiziert. [Weitere Informationen zu HTTP-Statuscodes.](#)

Links haben beschreibenden Text ^

Wenn du beschreibenden Linktext verwendest, können Suchmaschinen deine Inhalte besser verstehen. [Informationen zu barrierefreien Links.](#)

Links können gecrawlt werden ^

Suchmaschinen verwenden möglicherweise href-Attribute für Links, um Websites zu crawlen. Das href-Attribut von Anchor-Elementen muss auf ein geeignetes Ziel verweisen, damit mehr Seiten auf der Website gefunden werden können.

[Informationen dazu, wie Links für Crawler zugänglich gemacht werden](#)

Bildelemente verfügen über [alt]-Attribute ^

Für informative Elemente sollte ein kurzer, beschreibenden alternativer Text verwendet werden. Dekorative Elemente können mit einem leeren ALT-Attribut ignoriert werden. [Weitere Informationen zum Attribut alt.](#)

Dokument enthält ein gültiges hreflang-Element ^

Anhand von „hreflang“-Links können Suchmaschinen ermitteln, welche Version einer Seite sie in den Suchergebnissen für eine bestimmte Sprache oder Region anzeigen sollen. [Weitere Informationen zu hreflang.](#)

NICHT ZUTREFFEND (2)


Ausblenden

robots.txt ist gültig ^


Wenn deine robots.txt-Datei fehlerhaft ist, können Crawler möglicherweise nicht nachvollziehen, wie deine Website gecrawlt oder indiziert werden soll. [Weitere Informationen zu robots.txt.](#) Nicht bewertet

Dokument enthält ein gültiges rel=canonical-Element ^

Über kanonische Links wird angegeben, welche URL in den Suchergebnissen angezeigt werden soll. [Weitere Informationen zu kanonischen Links.](#) Nicht bewertet


 Captured at 13. Jan. 2026,
14:12 MEZ

 Erster Seitenaufbau

 Desktopemulation with
Lighthouse 13.0.1

 Benutzerdefinierte Drosselung

 Besuch einer einzigen Seite

 Using Chromium 143.0.0.0 with
devtools

Generated by **Lighthouse** 13.0.1 | [Problem melden](#)