AUTOMATISIERTER BARRIEREFREIHEITS-CHECK

www.christ.de

Prüfdatum: 06. November 2026

Prüfgrundlage: WCAG 2.2 Stufe AA · EN 301 549 · Barrierefreiheitsstärkungsgesetz (BFSG)

3	8	91
Seiten geprüft	Problemtypen	betroffene Elemente

1	6	1	0
Kritisch	Ernst	Mittel	Gering

Die Schweregrad-Zahlen geben die Anzahl der Problemtypen je Einstufung an.

Zusammenfassung

Wir haben 3 Seiten des Shops www.christ.de am 06. November 2026 automatisiert auf Barrieren geprüft. 2 weitere Seiten konnten nicht geladen werden und sind nicht in die Auswertung eingeflossen. Dabei haben wir 8 verschiedene Problemtypen an insgesamt 91 Stellen gefunden, davon 1 als kritisch und 6 als ernst eingestuft. Am häufigsten betroffen: ARIA-Rolle ohne erforderliches Elternelement (36 Elemente), Versteckte Elemente sind per Tastatur erreichbar (27 Elemente), Sichtbarer Text und zugänglicher Name stimmen nicht überein (14 Elemente). Für Ihre Besucherinnen und Besucher bedeutet das: Menschen, die einen Screenreader nutzen, erhalten fehlende oder falsche Informationen; wer nur mit der Tastatur navigiert, stößt auf unerreichbare oder irreführende Bedienelemente; Nutzer von Sprachsteuerung können Elemente nicht zuverlässig ansprechen. Viele dieser Punkte lassen sich zentral im Theme beheben und wirken dann auf allen Seiten gleichzeitig.

Befunde im Detail

Sortiert nach Schweregrad und Anzahl betroffener Elemente.

1. ARIA-Rolle ohne erforderliches Elternelement

Kritisch

`aria-required-parent` · 36 betroffene Elemente

WCAG-Kriterien: 1.3.1 Info und Beziehungen

BETROFFENE SEITEN

- Startseite `https://www.christ.de` – 12 Elemente
- Produktseite `https://www.christ.de/product/89216010/christ-damenring-89216010/index.html` – 12 Elemente
- Kategorieseite `https://www.christ.de/product/40008362/christ-damenring/index.html` – 12 Elemente

WAS BEDEUTET DAS FÜR NUTZER?

Elemente tragen eine ARIA-Rolle (z. B. menuitem, tab, option), stehen aber nicht im dafür vorgesehenen Container. Assistive Technologien können die Bedienelemente dadurch falsch oder gar nicht ansagen.

EMPFEHLUNG

Den passenden Container mit der richtigen Rolle ergänzen (z. B. role="menu" um role="menuitem") – oder die ARIA-Rollen ganz entfernen, wenn normales HTML genügt.

BEISPIELE AUS DEM SHOP

```
Selektor: .js-level-1-item.nav__list-item--has-children.nav__list-item:nth-child
(1) > .tab-focus-dropdown.js-nav-toggle-on-tab-focus[role="menuitem"]
HTML:     <button class="tab-focus-dropdown js-nav-toggle-on-tab-focus" role="men
uitem" tabindex="0">
<svg class="icon" role="img">
<title>Menü öffnen</title>
<use xlink:
```

```
Selektor: .js-level-1-item.nav__list-item--has-children.nav__list-item:nth-child
(2) > .tab-focus-dropdown.js-nav-toggle-on-tab-focus[role="menuitem"]
```

2. Versteckte Elemente sind per Tastatur erreichbar

Ernst

`aria-hidden-focus` · 27 betroffene Elemente

WCAG-Kriterien: 4.1.2 Name, Rolle, Wert

BETROFFENE SEITEN

- Startseite `https://www.christ.de` – 9 Elemente
- Kategorieseite `https://www.christ.de/product/40008362/christ-damenring/index.html` – 18 Elemente

WAS BEDEUTET DAS FÜR NUTZER?

Elemente sind für Screenreader ausgeblendet (aria-hidden), lassen sich aber per Tab-Taste fokussieren. Tastatur- und Screenreader-Nutzer landen auf „unsichtbaren“ Links, z. B. in geklonten Slider-Folien oder geschlossenen Menüs.

EMPFEHLUNG

Fokussierbare Elemente in ausgeblendeten Bereichen mit tabindex="-1" versehen oder den Bereich mit dem Attribut inert deaktivieren. Bei Slider-Plugins hilft oft ein Update oder die Barrierefreiheits-Option des Plugins.

BEISPIELE AUS DEM SHOP

```
Selektor: div[data-scarabitem="88832001"]
HTML:     <div class="carousel-cell" ref="productSlide" data-product-code="888320
01" data-scarabitem="88832001" aria-hidden="true" style="position: absolute; lef
t: 0px; t
```

```
Selektor: div[data-scarabitem="89239255"]
```

3. Sichtbarer Text und zugänglicher Name stimmen nicht überein

Ernst

`label-content-name-mismatch` · 14 betroffene Elemente

WCAG-Kriterien: 2.5.3 Beschriftung im Namen

BETROFFENE SEITEN

- Startseite `https://www.christ.de` – 14 Elemente

WAS BEDEUTET DAS FÜR NUTZER?

Der per aria-label vergebene Name enthält nicht den sichtbaren Text. Nutzer von Sprachsteuerung sagen „Klick auf In den Warenkorb“ – und nichts passiert.

EMPFEHLUNG

Den zugänglichen Namen mit dem sichtbaren Text beginnen lassen oder das aria-label entfernen, wenn der sichtbare Text bereits aussagekräftig ist.

BEISPIELE AUS DEM SHOP

```
Selektor: .ci-cms213-teaser-inside-fullsize > .ci-cms213-teaser-inside-item > .ci-cms213-teaser-inside-item-inner > .ci-teaser-style-light.ci-item-wrapper > .ci-link-whole-module[data-ci-element="link"]
HTML:     <a href="/category/geschenkideen-weihnachten/index.html" data-ci-element="link" class="ci-link-whole-module" aria-label="Entdecke Geschenke mit Bedeutung" title
```

```
Selektor: a[title="CHRIST Damenring 89216010"][aria-label="CHRIST Damenring 89216010"][data-product-code="89216010"]
```

4. Links ohne erkennbaren Namen

Ernst

`link-name` · 6 betroffene Elemente

WCAG-Kriterien: 2.4.4 Linkzweck (im Kontext) · 4.1.2 Name, Rolle, Wert

BETROFFENE SEITEN

- Startseite `https://www.christ.de` – 2 Elemente
- Produktseite `https://www.christ.de/product/89216010/christ-damenring-89216010/index.html` – 2 Elemente
- Kategorieseite `https://www.christ.de/product/40008362/christ-damenring/index.html` – 2 Elemente

WAS BEDEUTET DAS FÜR NUTZER?

Links, die nur aus einem Icon oder Bild bestehen (Social Media, Logo, Warenkorb), werden vom Screenreader nur als „Link“ ohne Ziel angesagt. Nutzer wissen nicht, wohin der Link führt.

EMPFEHLUNG

Icon-Links mit einem aria-label (z. B. „Zum Warenkorb“) oder visuell verstecktem Text versehen und das Icon selbst mit aria-hidden="true" ausblenden.

BEISPIELE AUS DEM SHOP

```
Selektor: .header__metanav-logo > .logo > a[data-promo-name="SiteLogo"][data-promo-id="site-logo-banner"][href="/"]
HTML:     <a href="/" data-promo-name="SiteLogo" data-promo-id="site-logo-banner">
```

```
Selektor: .ci-footer-logo-container > .logo > a[data-promo-name="SiteLogo"][data-promo-id="site-logo-banner"][href="/"]
```

5. Zu geringer Farbkontrast bei Texten

Ernst

`color-contrast` · 3 betroffene Elemente

WCAG-Kriterien: 1.4.3 Kontrast (Minimum)

BETROFFENE SEITEN

- Startseite `https://www.christ.de` – 1 Element
- Produktseite `https://www.christ.de/product/89216010/christ-damenring-89216010/index.html` – 1 Element
- Kategorieseite `https://www.christ.de/product/40008362/christ-damenring/index.html` – 1 Element

WAS BEDEUTET DAS FÜR NUTZER?

Texte heben sich nicht ausreichend vom Hintergrund ab. Menschen mit Sehschwäche, ältere Kundinnen und Kunden oder alle, die bei Sonnenlicht am Smartphone einkaufen, können Preise, Hinweise oder Buttons schlecht lesen.

EMPFEHLUNG

Textfarben so anpassen, dass ein Kontrastverhältnis von mindestens 4,5:1 (große Schrift 3:1) erreicht wird. Meist genügt es, wenige zentrale Farbvariablen zu ändern – häufig betroffen sind graue Hilfstexte, Footer, Platzhalter und Sale-Badges.

- **Shopware:** In den Theme-Einstellungen bzw. den SCSS-Variablen des eigenen Themes anpassen.
- **Shopify:** Im Theme-Editor unter „Farben“ bzw. in den Farbschemata anpassen.
- **WooCommerce:** Im Customizer oder per CSS im Child-Theme anpassen.

BEISPIELE AUS DEM SHOP

```
Selektor: .nav__list-item-link[data-title="Sale"] > .nav__list-item-link-wrapper
> span
HTML:     <span style="">Sale</span>
```

6. Prüfregel „role-img-alt“

Ernst

`role-img-alt` · 1 betroffenes Element

WCAG-Kriterien: 1.1.1 Nicht-Text-Inhalte

BETROFFENE SEITEN

- Startseite `https://www.christ.de` – 1 Element

WAS BEDEUTET DAS FÜR NUTZER?

Die automatische Prüfung meldet hier ein Problem, das die Nutzung mit assistiven Technologien erschweren kann. Originalbeschreibung der Prüfregel (englisch): [role="img"] and [role="image"] elements must have alternative text.

EMPFEHLUNG

Die betroffenen Elemente anhand der Beispiele unten prüfen und gemäß der genannten WCAG-Kriterien anpassen. Gern bewerten wir den konkreten Fall im Rahmen einer Nachbesserung.

BEISPIELE AUS DEM SHOP

```
Selektor: .ci-text-active
HTML:     <div class="ci-text-active" role="img">
<svg class="icon">
<title>Discount Code wurde kopiert</title>
<use xlink:href="#check"></use>
</svg>
Code kopiert
</div>
```

7. Prüfregel „aria-prohibited-attr“

Ernst

`aria-prohibited-attr` · 1 betroffenes Element

WCAG-Kriterien: 4.1.2 Name, Rolle, Wert

BETROFFENE SEITEN

- Produktseite `https://www.christ.de/product/89216010/christ-damenring-89216010/index.html` – 1 Element

WAS BEDEUTET DAS FÜR NUTZER?

Die automatische Prüfung meldet hier ein Problem, das die Nutzung mit assistiven Technologien erschweren kann. Originalbeschreibung der Prüfregel (englisch): Elements must only use permitted ARIA attributes.

EMPFEHLUNG

Die betroffenen Elemente anhand der Beispiele unten prüfen und gemäß der genannten WCAG-Kriterien anpassen. Gern bewerten wir den konkreten Fall im Rahmen einer Nachbesserung.

BEISPIELE AUS DEM SHOP

```
Selektor: .pdp-compare-toggle
HTML:     <label for="toggle_1" class="pdp-compare-toggle" data-is-link="false" aria-label="Zum Produktvergleich hinzufügen" data-haskeylistener="true"></label>
```

8. Zoomen auf Mobilgeräten wird verhindert

Mittel

`meta-viewport` · 3 betroffene Elemente

WCAG-Kriterien: 1.4.4 Textgröße ändern

BETROFFENE SEITEN

- Startseite `https://www.christ.de` – 1 Element
- Produktseite `https://www.christ.de/product/89216010/christ-damenring-89216010/index.html` – 1 Element
- Kategorieseite `https://www.christ.de/product/40008362/christ-damenring/index.html` – 1 Element

WAS BEDEUTET DAS FÜR NUTZER?

Die Seite unterbindet das Vergrößern per Fingergeste (user-scalable=no oder maximum-scale). Menschen mit Sehschwäche können Texte am Smartphone nicht vergrößern.

EMPFEHLUNG

Im viewport-Meta-Tag user-scalable=no und maximum-scale entfernen. Die Änderung erfolgt im zentralen Layout-Template (Shopware: base.html.twig, Shopify: theme.liquid, WooCommerce: header.php).

BEISPIELE AUS DEM SHOP

```
Selektor: meta[name="viewport"]
HTML:     <meta name="viewport" content="width=device-width, initial-scale=1, user-scalable=no">
```

Sichtprüfung und Tastatur

Ergänzend zur automatischen Regelprüfung haben wir die unten genannte Seite so untersucht, wie sie Screenreader, Tastatur und stark vergrößerte Ansichten erleben. Die folgenden Punkte stammen aus dieser Sichtprüfung und sind mit Screenshots belegt.

Geprüfte Seite: `https://www.christ.de`

Umfang: 206 Bedienelemente im Accessibility-Tree ausgewertet · 25 Tab-Schritte nachverfolgt · Seitenbreite bei 320 px Fensterbreite: 320 px

1. Screenreader-Namen

Ernst

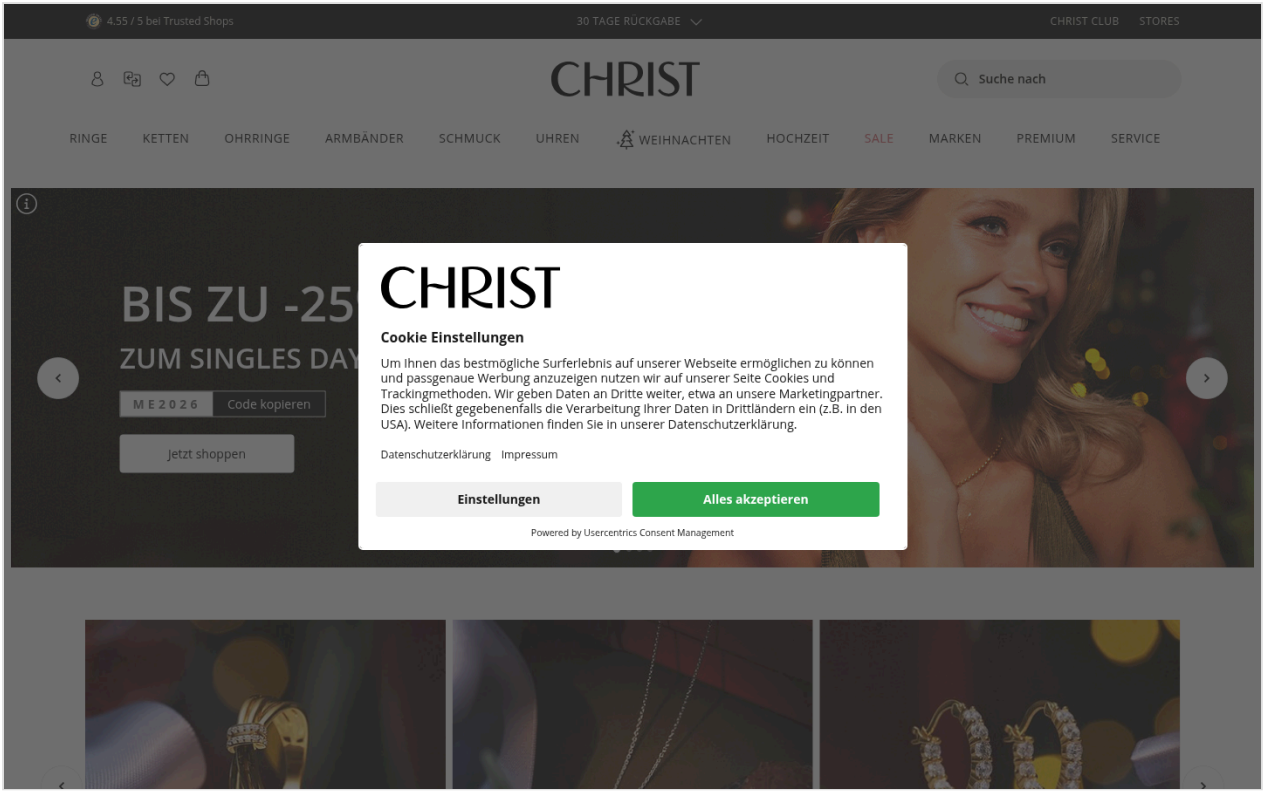
WCAG-Kriterien: 2.4.4 Linkzweck (im Kontext) · 4.1.2 Name, Rolle, Wert

BEFUND

38 Bedienelemente haben keinen brauchbaren Screenreader-Namen (leer, Dateiname, URL oder Platzhalterwort).

BETROFFENE ELEMENTE (AUSZUG)

Rolle	Name
link	(leer)
image	(leer)
link	(leer)
image	(leer)
image	(leer)
image	(leer)
image	(leer)
image	(leer)



Screenshot des sichtbaren Ausschnitts der geprüften Seite in der Desktop-Ansicht (1440 × 900 Pixel) als Kontext zu den Bedienelementen ohne brauchbaren Screenreader-Namen.

2. Fokus landet auf unsichtbaren Elementen

Ernst

WCAG-Kriterien: 2.4.3 Fokus-Reihenfolge · 2.4.7 Fokus sichtbar

BEFUND

Die Tabulatortaste landet fast nur auf unsichtbaren Elementen (25 von 25) — für Tastaturnutzer ist nicht erkennbar, wo sie sich befinden.

BETROFFENE ELEMENTE (AUSZUG)

Element	Fokusrahmen	Position y (px)
	outline: none 3px	6.897
	outline: none 3px	6.897
	outline: none 3px	6.897

Screenshot (focus-state.png) liegt nicht vor.

3. Reduzierte Bewegung

Mittel

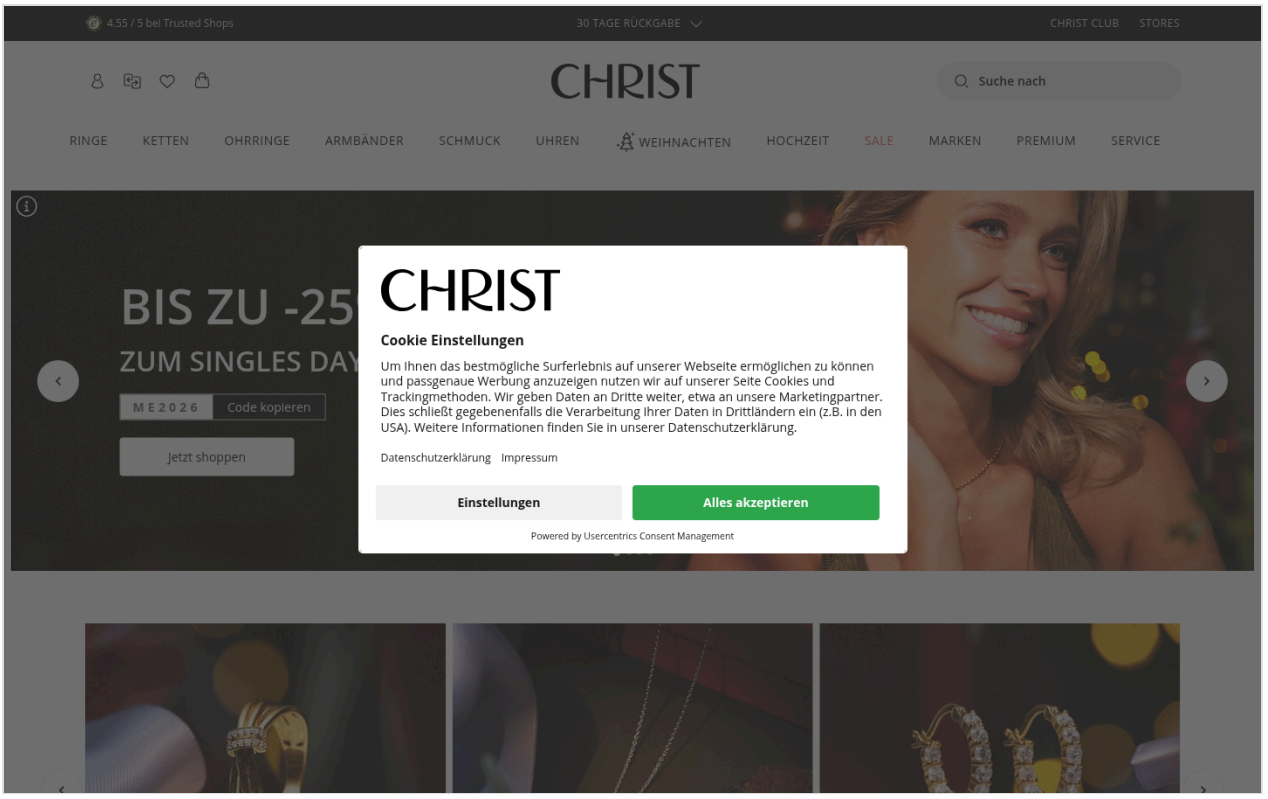
WCAG-Kriterien: 2.3.3 Animation durch Interaktionen (Stufe AAA, Empfehlung)

BEFUND

1 Elemente animieren länger als 0,5 s, auch wenn "reduzierte Bewegung" aktiv ist.

BETROFFENE ELEMENTE (AUSZUG)

Element	Animationsdauer
	Animation 0.75s



Screenshot des sichtbaren Ausschnitts der geprüften Seite in der Desktop-Ansicht (1440 × 900 Pixel) als Kontext zu den animierten Elementen.

Was zusätzlich geprüft wurde (und was nicht)

Der Plus-Check ergänzt die automatische Regelprüfung um eine Auswertung des Accessibility-Trees (welche Namen ein Screenreader für Bedienelemente erhält), eine Nachverfolgung der Tastaturführung per Tab-Taste einschließlich des sichtbaren Fokus, eine Prüfung des Umbruchs bei 320 px Breite (Reflow) sowie des Verhaltens bei aktivierter Einstellung „reduzierte Bewegung“. Alle Beobachtungen sind mit Screenshots aus Desktop-, Mobil- und 320-px-Ansicht dokumentiert. Einige der im folgenden Abschnitt „Methodik & Grenzen“ als nicht geprüft genannten Punkte (Tastatur, mobile Ansicht) sind damit für die geprüfte Seite teilweise abgedeckt. Nicht ersetzen kann der Plus-Check echte Screenreader-Durchläufe mit NVDA oder VoiceOver, eine Bewertung der Verständlichkeit von Inhalten, einen Testkauf im Bestellprozess und reale Tests mit Nutzerinnen und Nutzern mit Behinderung.

Hinweis: Auch der Plus-Check ist keine rechtliche Bewertung und stellt keine Rechtsberatung dar.

Methodik & Grenzen

WAS GEPRÜFT WURDE

- Automatisierte Prüfung mit der Open-Source-Prüfbibliothek axe-core (Deque Systems) in einem echten Chromium-Browser.
- Regelsatz: WCAG 2.0, 2.1 und 2.2 jeweils Stufe A und AA (Grundlage der EN 301 549 und damit des BFSG).
- Desktop-Ansicht mit 1440 × 900 Pixeln, deutsche Spracheinstellung; Cookie-Banner wurden vor der Prüfung nach Möglichkeit geschlossen.
- 3 Seiten erfolgreich geprüft (von 5 angefragten Seiten):
- Zusätzlich (Bericht Plus): Tastatur-Durchlauf mit 25 Tab-Schritten auf der Startseite, Accessibility-Tree (welche Namen Screenreader ansagen), Reflow-Prüfung bei 320 Pixeln Breite, Verhalten bei reduzierter Bewegung, Screenshots je Befund.

Seite	Adresse	Status	Problemtypen
Startseite	`https://www.christ.de`	geprüft	7
Produktseite	`https://www.christ.de/product/89216010/christ-damenring-89216010/index.html`	geprüft	5
Kategorieseite	`https://www.christ.de/product/40008362/christ-damenring/index.html`	geprüft	5
Warenkorb	`https://www.christ.de/warenkorb`	nicht geprüft (Error: HTTP 404)	–
Suchergebnisse	`https://www.christ.de/suche?q=schuhe`	nicht geprüft (Error: HTTP 404)	–

WAS NICHT GEPRÜFT WURDE

- Echte Screenreader-Läufe mit NVDA oder VoiceOver (der Plus-Check liest den Accessibility-Tree aus, nicht die Sprachausgabe)
- Tastatur-Durchlauf auf allen Seiten außer der geprüften Startseite (Fokusreihenfolge, Tastaturfallen, sichtbarer Fokus)
- Verständlichkeit von Inhalten, Texten und Fehlermeldungen
- PDF-Dokumente und andere Downloads
- Checkout, Login und Bestellprozess
- Mobile Ansicht und Touch-Bedienung außerhalb der 320-Pixel-Reflow-Prüfung

Automatische Tests erkennen erfahrungsgemäß nur einen Teil der möglichen Barrieren. Eine fehlende Meldung bedeutet daher nicht, dass ein Bereich barrierefrei ist.

> **Wichtiger Hinweis:** Dieser Bericht ist eine technische Bestandsaufnahme und keine rechtliche Bewertung. Er stellt keine Rechtsberatung dar und begründet keine Zusicherung oder Garantie der Konformität mit dem BFSG, der EN 301 549 oder den WCAG.

Nächste Schritte

1. **Kritische Befunde zuerst beheben** (1 Problemtyp): Sie sperren Menschen mit Behinderung ganz von einzelnen Funktionen aus.
2. **Danach ernste Befunde angehen** (6 Problemtypen), anschließend mittlere und geringe.
3. **Erneut prüfen:** Nach den Korrekturen einen Re-Scan durchführen und ergänzend eine manuelle Prüfung von Tastaturbedienung und Bestellprozess einplanen.

Die meisten Korrekturen betreffen das Theme bzw. Templates und können als fertige Patches für Ihr Shop-Theme geliefert werden – ohne Eingriff in Ihre Produktdaten.

Erstellt am 06. November 2026 · Automatisierter Check, keine Rechtsberatung · Auch der Plus-Check ist keine rechtliche Bewertung.