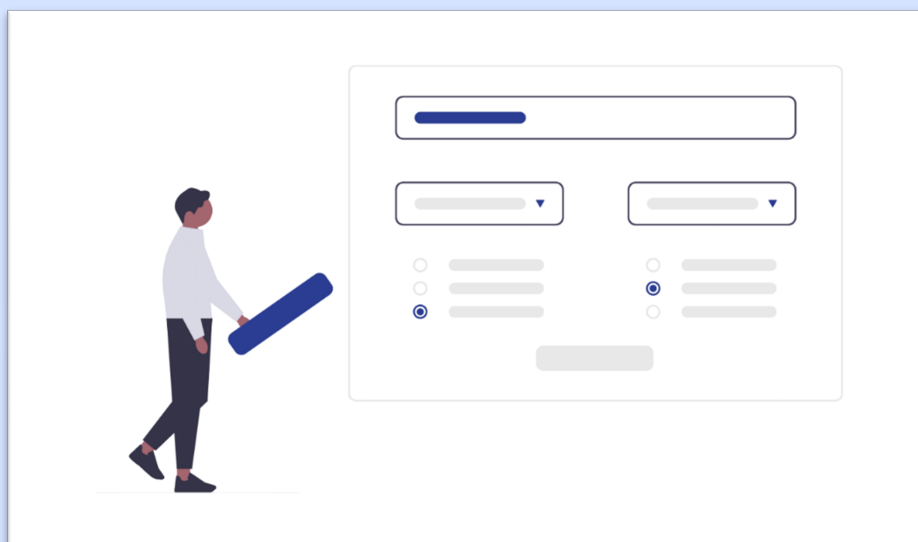




# Barrierefreie Formulare

Ob Kontaktformular, Anmeldung oder Online-Antrag: Formulare sind im Web allgegenwärtig und müssen so gestaltet sein, dass niemand ausgeschlossen wird. Barrierefreie Formulare zeichnen sich durch klare Beschriftungen, nachvollziehbare Struktur und zugängliche Interaktionen aus. Das bedeutet zum Beispiel, dass alle Felder korrekt benannt und per Tastatur erreichbar sind und Rückmeldungen wie Fehlermeldungen verständlich präsentiert werden. So wird sichergestellt, dass wirklich jeder das Formular ausfüllen und absenden kann.



In diesem Guide wird hauptsächlich über HTML-Formulare gesprochen, doch lassen sich gewisse Aspekte auch auf Formulare in anderen Formaten beziehen.



## Warum sind barrierefreie Formulare wichtig?

Gut gestaltete Formulare kommen vielen zugute, aber besonders profitieren Menschen mit Beeinträchtigungen.

Zum Beispiel sind klare Beschriftungen und Hinweise essenziell für blinde und sehbehinderte Personen, die Screenreader nutzen. Ohne korrekte Labels können Screenreader nicht ausgeben, was in ein Feld einzugeben ist.

Auch Personen mit motorischen Beeinträchtigungen oder jene, die keine Maus verwenden, sind auf eine durchdachte Tastaturbedienung angewiesen. Wenn interaktive Elemente nicht per Tastatur erreichbar sind oder der Fokus nicht sichtbar ist, entsteht für diese Nutzenden eine unüberwindbare Hürde.

Menschen mit kognitiven Beeinträchtigungen profitieren von verständlicher Sprache, klaren Anleitungen und vorhersehbaren Abläufen, zum Beispiel helfen ihnen konkrete Fehlermeldungen und Beispielergebnisse dabei, Eingabefehler zu vermeiden und zu korrigieren.

Barrierefreie Formulare fördern Inklusion und verbessern für viele die User Experience.

## Relevante WCAG-Erfolgskriterien

Die Web Content Accessibility Guidelines (WCAG) definieren verschiedene Erfolgskriterien, die für barrierefreie Formulare besonders wichtig sind:

- » **1.3.1 Informationen und Beziehungen (Level A):** Alle Informationen und strukturellen Beziehungen müssen programmatisch erkennbar sein.
- » **2.1.1 Tastatur (Level A):** Alle Funktionen im Formular müssen mit der Tastatur nutzbar sein.
- » **2.4.3 Fokusreihenfolge (Level A):** Die Tab-Reihenfolge beim Durchlaufen des Formulars mit der Tastatur muss logisch und nachvollziehbar sein.
- » **2.4.7 Sichtbarer Fokus (Level AA):** Wenn sich ein Element im Fokus befindet, muss dies klar erkennbar sein.
- » **3.3.1 Fehlererkennung (Level A):** Wenn ein Fehler bei der Eingabe erkannt wird, muss dies dem Nutzenden eindeutig mitgeteilt werden.
- » **3.3.2 Beschriftungen oder Anleitungen (Level A):** Jedes Formularfeld muss eine eindeutige und verständliche Beschriftung oder Anleitung haben.
- » **3.3.3 Fehlerhilfe (Level AA):** Es müssen konkrete Hinweise gegeben werden, wie ein Eingabefehler korrigiert werden kann.
- » **3.3.4 Fehlervermeidung (Level AA):** Bei kritischen Formularen (z. B. rechtlich verbindlichen Vorgängen) soll die Möglichkeit bestehen, Eingaben zu überprüfen, zu ändern oder den Vorgang abubrechen

# Wichtige Aspekte bei der Umsetzung

## Aussagekräftige Beschriftungen und Anleitungen:

- » Jedes Formularfeld braucht eine sichtbare Text-Beschriftung oder eine klare Anleitung. In HTML erreicht man das durch das `<label>`-Element, das mit dem Eingabefeld verknüpft wird.
- » **Wichtig:** Platzhaltertexte im Eingabefeld (sog. *Placeholder*) reichen nicht aus und dürfen nicht als alleinige Beschriftung dienen. Sie verschwinden beim Tippen und sind für Screenreader oft nicht zugänglich. Besser ist es, neben jeder Eingabe ein dauerhaft sichtbares Label anzuzeigen. Gegebenenfalls kann man zusätzliche Hinweise unter dem Feld platzieren, um z. B. das erwartete Format zu erläutern („TT.MM.JJJJ“ bei einem Datumsfeld). Sprechende Beschriftungen wie „Telefonnummer (optional)“ machen sofort klar, welche Eingabe erwartet wird und ob ein Feld Pflicht oder freiwillig ist.

## Tastaturbedienbarkeit und Fokusführung:

- » Ein barrierefreies Formular muss vollständig ohne Maus bedienbar sein. Das bedeutet, dass alle interaktiven Elemente wie z. B. Felder, Buttons und Links mit der Tabulator-Taste erreichbar sind. Die Tab-Reihenfolge sollte der visuellen Anordnung entsprechen.
- » Weiterhin muss der Fokus deutlich sichtbar sein. Verwendet CSS, um den Fokuszustand hervorzuheben (z. B. durch einen klaren Rahmen oder Hintergrundfarbe). Vermeiden Sie es, den Standardfokus einfach zu entfernen, ohne Ersatz anzubieten. Gute Fokusmarkierungen helfen Nutzenden mit Sehbehinderung oder Aufmerksamkeitsproblemen, sich im Formular zu orientieren.

## Fehlermeldungen und Validierung:

- » Eine häufige Barriere sind unklare oder nicht wahrnehmbare Fehlermeldungen. Stellen Sie sicher, dass Fehler sowohl visuell als auch programmatisch ermittelt werden. Konkret heißt das: Falls eine Eingabe fehlt oder ungültig ist, sollte neben dem betreffenden Feld eine gut verständliche Fehlermeldung erscheinen und diese Meldung idealerweise mit dem Feld verknüpft sein (in HTML z. B. via aria-describedby). So liest ein Screenreader die Fehlermeldung vor, sobald das Feld fokussiert wird.
- » Die Fehlermeldung selbst sollte präzise formuliert sein: Statt nur „Ungültige Eingabe“ lieber einen konkreten Hinweis geben, etwa „Bitte geben Sie eine fünfstellige Postleitzahl ein“. Das hilft allen Nutzenden, den Fehler zu verstehen. Zeigen Sie Fehler am besten **dynamisch** an, also sofort bei oder nach der Eingabe, und nicht erst nach dem Absenden des gesamten Formulars. So können Fehler schneller korrigiert werden.
- » Ebenfalls empfehlenswert: Nach dem Absenden sollte das Formular eine eindeutige Bestätigung anzeigen (z. B. „Vielen Dank, Ihre Nachricht wurde gesendet.“), damit auch Nutzende von Screenreadern oder mit kognitiven Einschränkungen sicher wissen, dass der Vorgang abgeschlossen ist.

## Wie sehen barrierefreie Formulare in der Praxis aus?

Bereits im Markup sind alle Eingabefelder mit ihren Labels verknüpft, Pflichtfelder sind deutlich markiert (etwa durch „(Pflichtfeld)“ im Label) und required Attribut sind gesetzt. Die Absende-Schaltfläche hat einen aussagekräftigen Text wie „Nachricht senden“ statt nur „Abschicken“.

Wird das Formular abgeschickt und es treten Fehler auf, so erscheinen direkt über dem Formular eine verständliche Fehler-Zusammenfassung und an jedem fehlerhaften Feld eine spezifische Fehlermeldung. Beide Hinweise sind mit den jeweiligen Feldern verknüpft, sodass Screenreader-Nutzende diese zugeordnet wahrnehmen können.

Der Tastaturfokus springt nach dem Absenden automatisch zum ersten Fehlerhinweis, damit niemand diesen übersieht. Bei einer Gruppe von Auswahlmöglichkeiten werden in HTML ein <fieldset> und <legend> genutzt, um diese Optionen als zusammengehörig zu strukturieren.

## Beispiel:

(nach Absenden des Formulars mit Fehlern)

```
<form aria-labelledby="form-title" novalidate>
  <h2 id="form-title">Kontaktformular</h2>

  <!-- Fehler-Zusammenfassung -->
  <div id="error-summary" role="alert" tabindex="-1">
    <p>Bitte beheben Sie die folgenden Fehler:</p>
    <ul>
      <li><a href="#anrede-herr">Bitte wählen
        Sie eine Anrede aus.</a></li>
      <li><a href="#name">Name ist ein
        Pflichtfeld.</a></li>
      <li><a href="#email">Bitte geben Sie eine
        gültige E-Mail-Adresse (Format: team@team.de) ein.</a></li>
    </ul>
  </div>

  <!-- Pflichtfeld-Hinweis nur visuell -->
  <p aria-hidden="true"><strong>Hinweis:</strong>
    Felder mit <span>*</span> sind Pflichtfelder.</p>
```

```
<!-- Anrede -->  
  
<fieldset aria-describedby="error-anrede">  
  
    <legend>Anrede <span aria-  
hidden="true">*</span></legend>  
  
    <label>  
  
        <input type="radio" name="anrede"  
id="anrede-herr" value="herr" required autocomplete="honorific-  
prefix">  
  
        Herr  
  
    </label>  
  
    <label>  
  
        <input type="radio" name="anrede"  
id="anrede-frau" value="frau" autocomplete="honorific-prefix">  
  
        Frau  
  
    </label>  
  
    <p id="error-anrede" class="error"  
role="alert">Bitte wählen Sie eine Anrede aus.</p>  
  
</fieldset>
```

```
<!-- Name -->

<label for="name">Name <span aria-
hidden="true">*</span></label>

<input id="name" name="name" type="text" required
autocomplete="name" aria-describedby="error-name">

<p id="error-name" class="error" role="alert">Bitte
geben Sie Ihren Namen an.</p>

<!-- E-Mail -->

<label for="email">E-Mail <span aria-
hidden="true">*</span></label>

<input id="email" name="email" type="email" required
autocomplete="email" aria-describedby="error-email">

<p id="error-email" class="error" role="alert">Bitte
geben Sie eine gültige E-Mail-Adresse (Format: team@team.de)
ein.</p>

<!-- Absenden -->

<button type="submit">Nachricht senden</button>

</form>
```

## Kontaktformular

Bitte beheben Sie die folgenden Fehler:

- Bitte wählen Sie eine Anrede aus.
- Name ist ein Pflichtfeld.
- Bitte geben Sie eine gültige E-Mail-Adresse (Format: team@team.de) ein.

Hinweis: Felder mit \* sind Pflichtfelder.

Anrede \*

☐ Herr ☐ Frau

Bitte wählen Sie eine Anrede aus.

Name \*

Bitte geben Sie Ihren Namen an.

E-Mail \*

Bitte geben Sie eine gültige E-Mail-Adresse (Format: team@team.de) ein.

**Nachricht senden**

## Drittanbieter-Tools

Oft greifen Redakteure auf Online-Formulardienste wie Google Forms oder Microsoft Forms zurück. Die gute Nachricht: Viele dieser Tools berücksichtigen grundlegende Barrierefreiheitsaspekte bereits von Haus aus.

Google Forms zum Beispiel erstellt Formulare, die für eine breite Zielgruppe nutzbar sind, einschließlich Menschen mit Screenreader oder andere assistive Technologien.

Auch Microsoft Forms bietet Unterstützung für Tastaturnavigation und Screenreader. Microsoft stellt sogar Anleitungen bereit, wie man ihre Formulare mit Screenreader bedient.

Dennoch ist Vorsicht geboten: Die Verantwortung für verständliche Texte, klare Eingabeaufforderungen und barrierefreie Inhalte liegt weiterhin bei den Erstellenden. Achten Sie also auch bei Google/Microsoft Forms darauf, verständliche Feldbeschriftungen zu verwenden, alle notwendigen Hinweise bereitzustellen und Medien (Bilder, Videos) mit Alternativtext bzw. Untertiteln zu versehen. Testen Sie das fertige Formular unbedingt selbst mit der Tastatur und, falls möglich, mit einem Screenreader, um sicherzugehen, dass alles funktioniert. Beachten Sie auch, dass die Ergebnisse solcher Tools (z. B. Auswertungsdigramme oder Tabellen) unter Umständen nicht vollständig barrierefrei dargestellt werden. Hier kann es sinnvoll sein, alternative Darstellungen oder Exportmöglichkeiten (CSV/Excel) anzubieten.

## HTML-Formulare vs. PDF-Formulare

Formulare können nicht nur als HTML auf Webseiten, sondern auch in PDF-Dokumenten realisiert werden. Allerdings gibt es wichtige Unterschiede in Bezug auf Barrierefreiheit: HTML-Formulare sind in der Regel flexibler und mit weniger Aufwand barrierefrei zu gestalten.

Durch native HTML-Strukturen und Webstandards lassen sich viele Anforderungen (Beschriftungen, Fokus-Reihenfolge, Fehlermeldungen) relativ einfach umsetzen. Moderne Browser und assistive Technologien unterstützen HTML-Formulare sehr gut, wenn sie korrekt aufgebaut sind.

PDF-Formulare dagegen erfordern spezielle Schritte und Tools, um barrierefrei zu sein. Es gibt derzeit keine kostenfreie Software, um vollständig barrierefreie PDF-Formulare zu erstellen. Der Aufwand ist deutlich höher. PDF-Formulare eignen sich vor allem, wenn das ausgefüllte Dokument unterschrieben oder heruntergeladen werden muss, z. B. bei offiziellen Anträgen oder Verträgen. Doch sollten hier Papierformulare nicht 1 zu 1 ins Digitale umgebaut werden, da diese meist nicht gut dafür konzeptioniert sind. Hier macht sich ein neuer Aufbau des Formulars meist besser.

Für interaktive Umfragen, Feedback-Formulare oder allgemeine Web-Formulare sind hingegen HTML-Lösungen oder spezialisierte Online-Tools besser geeignet. Wir empfehlen daher HTML-Formulare, weil sie mit Standardtechnologien breiter zugänglich sind.

## Checkliste für die Erstellung eines barrierefreien Formulars

- ✓ **HTML-Standardelemente werden verwendet: Semantik** und Funktion sind klar erkennbar.
- ✓ **Labels sichtbar und korrekt verknüpft:** Jedes Eingabefeld besitzt ein sichtbares Label, das korrekt mit dem Feld verbunden ist.
- ✓ **Gruppierungen richtig strukturiert:** Zusammengehörige Felder sind mit <fieldset> und <legend> gegliedert.
- ✓ **Hilfetexte sind verbunden:** Zusätzliche Hinweise oder Formatangaben sind mit dem jeweiligen Feld verknüpft.
- ✓ **Vollständige Tastaturbedienbarkeit:** Das gesamte Formular ist ohne Maus vollständig nutzbar.
- ✓ **Fehlermeldungen dynamisch und zugänglich:** Fehlermeldungen erscheinen automatisch nach Absenden oder Verlassen des Feldes.
- ✓ **Formular selbst testen:**
  - » Tastaturnavigation prüfen: Sind alle interaktiven Elemente bedienbar?
  - » Screenreader-Test durchführen: Werden alle Informationen übermittelt?
  - » Responsives Verhalten prüfen: Sind alle Inhalte in der mobilen Ansicht sichtbar und bedienbar?