

SOP 13

Text formatieren

Projektleitung

Geschäftsführung

IT

Recht

Was du hier machst	Du lernst, deine Texte technisch korrekt mit Überschriften, Absätzen und Listen auszuzeichnen, um eine klare semantische Struktur zu schaffen.
Voraussetzung	Du erstellst Inhalte für Webseiten wie Ratgeber, Produkttexte, AGB, Blogartikel, Anleitungen, FAQs oder Landingpages.
Dauer / Aufwand	Initialaufwand: 2–3 Stunden (für Einrichtung des Kanals, Erstellung der Tracking-Liste, Formulierung von Textbausteinen). Laufend: 15–30 Minuten pro eingegangener Meldung.
Gesetzlich verpflichtend	Ja – das Barrierefreiheitsstärkungsgesetz (BFSG), der European Accessibility Act (EAA) und die WCAG 2.1 (insbesondere 1.3.1 und 2.4.6) fordern eine verständliche und maschinenlesbare Struktur. Die DSGVO (Art. 12 Abs. 1) verlangt, dass Informationen in leicht zugänglicher Form bereitgestellt werden.
Nächster Schritt	SOP 14: Alternativtexte für Bilder.

 SOP 13: Formatiere deine Texte barrierefrei

Wie du eine Struktur, schaffst, die jede/r versteht – ganz egal, in welchem Zustand oder welcher Situation

Wer sollte diese SOP lesen?

Was du hier machst

Du lernst, deine Texte technisch korrekt mit Überschriften, Absätzen und Listen auszuzeichnen, um eine klare semantische Struktur zu schaffen.

 Voraussetzung


Du erstellst Inhalte für Webseiten wie Ratgeber, Produkttexte, AGB, Blogartikel, Anleitungen, FAQs oder Landingpages.

Ergebnis

Eine klar strukturierte, logisch aufgebaute Textseite, bei der alle Inhalte technisch korrekt für Screenreader formatiert sind und eine sichtbare sowie maschinenlesbare Struktur aufweisen.

 Dauer

Initialaufwand: 2–3 Stunden (für Einrichtung des Kanals, Erstellung der Tracking-Liste, Formulierung von Textbausteinen). Laufend: 15–30 Minuten pro eingegangener Meldung.

 Gesetzlich verpflichtend?

Ja – das Barrierefreiheitsstärkungsgesetz (BFSG), der European Accessibility Act (EAA) und die WCAG 2.1 (insbesondere 1.3.1 und 2.4.6) fordern eine verständliche und maschinenlesbare Struktur. Die DSGVO (Art. 12 Abs. 1) verlangt, dass Informationen in leicht zugänglicher Form bereitgestellt werden.

 Nächster Schritt

SOP 14: Alternativtexte für Bilder.

Warum machst du das überhaupt?

Weil nicht nur deine Sprache und deine Form barrierefrei sein muss, sondern auch die technische Umsetzung deines Textes in HTML und CSS. Nur wenn du auf deiner Website z.B. Überschriften, Fließtext, Absätze, Listen usw. sauber und korrekt auszeichnest, können Screenreader-Nutzer:innen, Menschen mit kognitiven Einschränkungen und auch alle anderen deine Texte gut erfassen und verstehen.

Wichtig: Falls du keinen technischen Background hast und nicht weißt, was HTML genau ist, solltest du dir ab dieser SOP zumindest die Grundlagen aneignen oder jemanden fragen, der z.B. eure Website betreut und dir weiterhelfen kann.

Am Ende hast du einen klaren Prozess, um alle deine Texte technisch so aufzubereiten, dass sie eine klare sogenannte „semantische“ Struktur haben. Dank dieser semantischen Struktur können Hilfstechnologien deine Inhalte leicht scannen und verstehen und sie so für alle Nutzer:innen lesbar machen.

In deinem Text-Dokument mit einem Titel wie „Dokumentation des Barrierefreiheit-Feedback-System“ .

Wenn du mit dieser SOP fertig bist, legst du dieses Dokument an und trägst die Ergebnisse dieser SOP dort ein.

- Initialaufwand: 2–3 Stunden (für Einrichtung des Kanals, Erstellung der Tracking-Liste, Formulierung von Textbausteinen).
- Laufend: 15–30 Minuten pro eingegangener Meldung.
- DU (Projektleitung, EAA-Verantwortliche:r): Steuerst und überwachst den Prozess, führst die Triage durch und verantwortest die Kommunikation mit den Nutzer:innen.
- GF / Recht: Werden bei kritischen Meldungen oder potenziellen Eskalationen informiert und treffen strategische Entscheidungen.
- IT / Entwicklung / Agentur: Analysieren die gemeldete Barriere technisch und sind für die Behebung verantwortlich.
- Kundenservice: Kann als erste Anlaufstelle dienen und leitet Meldungen standardisiert an dich weiter.

Wann musst du das machen?

- Bei allen Website-Inhalten, aber besonders bei:
- Ratgeberseiten, Produkttexten, AGB und Datenschutzerklärungen
- Blogartikeln, Anleitungen und Erklärtexten
- FAQs und Landingpages

Welches Gesetz verlangt das?

- Barrierefreiheitsstärkungsgesetz (BFSG) / European Accessibility Act (EAA) : Eine verständliche Struktur gehört zu den Mindestanforderungen.
- WCAG 2.1: Insbesondere die Erfolgskriterien 1.3.1 (Info and Relationships) und 2.4.6 (Headings and Labels).
- DSGVO Art. 12 Abs. 1: Fordert, dass Informationen „in leicht zugänglicher Form“ bereitgestellt werden müssen.

Der Prozess im Detail

1 Betrachte Überschriften als Wegweiser und formatiere sie entsprechend

Überschriften sind das wichtigste Werkzeug, um Text zu strukturieren. Screenreader-Nutzer:innen navigieren oft nur über sie, um eine Seite zu überfliegen, und auch Google versteht Inhalte anhand ihrer Hierarchie.

Halte dich an folgende Regeln, wenn du Überschriften in HTML auszeichnest:

- Nur eine H1 pro Seite: Die `<h1>` ist der Haupttitel der Seite und darf nur einmal vorkommen.
- Logische Hierarchie einhalten: Strukturiere deine Inhalte streng absteigend (nach `<h1>` kommt `<h2>`, danach `<h3>` usw.). Überspringe niemals eine Ebene.
- Keine optische Formatierung: Formatiere Überschriften niemals nur durch fette (bold) oder größere Schrift. Nutze immer die echten HTML-Tags (`<h1>`, `<h2>` etc.).

2 Verbessere die Lesbarkeit deiner Texte mit Absätzen und Listen

Lange Textblöcke sind schwer zu lesen. Lockere sie gezielt auf, indem du folgende Techniken nutzt:

- Kurze Absätze: Trenne Gedanken klar und halte Absätze kurz, idealerweise nicht länger als vier Zeilen.
- Verwende echte Listen: Nutze für Aufzählungen immer die korrekten HTML-Listen-Elemente, also `` für unsortierte Bullet Points und `` für nummerierte Listen. Erstelle bitte keine Pseudolisten mit Bindestrichen oder Sternchen.
- Tabellen nur für Daten: Verwende Tabellen ausschließlich zur Darstellung von tabellarischen Daten – niemals zur Gestaltung des Layouts.

3 Setze semantisches HTML korrekt ein

Stelle nun konsequent sicher, dass die Struktur nicht nur sichtbar, sondern auch für Maschinen lesbar ist.

Das heißt:

- Gestalte dein Layout mit CSS : Vermeide es, Layout über die Textstruktur zu erzeugen z. B. durch leere Absätze oder `
`-Tags. Stattdessen nutze `<p>` für einen neuen Absatz (`<p>` steht für „Paragraph“) und `
` nur, wenn du einen Zeilenumbruch INNERHALB eines Absatzes möchtest. `
` steht nämlich für „Line Break“.
- Achte darauf, dass in deinem Pagebuilder alles korrekt konfiguriert ist: Vieles davon kannst du mit visuellen Editoren anstatt in HTML machen. Aber Achtung: Auch in Pagebuildern wie Elementor oder Divi müssen die semantischen Elemente korrekt konfiguriert sein.

4 Optimierte auf visuelle und kognitive Lesbarkeit

Eine gute visuelle Gestaltung unterstützt die Struktur. Falls du mit der visuellen Gestaltung deines Page Builders nicht zufrieden bist, kannst du in den meisten Fällen selbst definieren, wie deine Elemente (z.B. Überschriften) aussehen sollen.

Beachte auf jeden Fall folgende Regeln:

- Deutliche Hervorhebung: Überschriften müssen sich visuell klar von Fließtext abheben (größer, fetter, mit mehr Abstand).
- Großzügige Abstände: Sorge für einen Zeilenabstand von mindestens 1,5 und großzügige Abstände zwischen den Absätzen.
- Vermeide Blocksatz: Linksbündiger Flattersatz ist besser lesbar, da er keine unregelmäßigen Lücken zwischen den Wörtern erzeugt.
- Nutze eine barrierefreie Schriftgröße: Die Schriftgröße sollte responsiv sein und mindestens 16px betragen.

5 Teste und überprüfe deine Formatierung

Wie bei allen Prozessen, solltest du natürlich am Ende auch das Ergebnis kontrollieren, um sicherzustellen, dass die Struktur funktioniert. Wie? Indem du folgende Techniken nutzt:

- Screenreader-Test (NVDA , VoiceOver): Prüfe, ob die Struktur korrekt und logisch vorgelesen wird.
- Tastatur-Navigation: Teste, ob du mit der Tab-Taste alle interaktiven Elemente erreichen kannst.
- Accessibility-Checker (z. B. WAVE oder axe Accessibility Linter): Diese Tools helfen, falsch verwendete oder fehlende Strukturelemente zu erkennen.

Du hast nun:

- Eine klar strukturierte, logisch aufgebaute Textseite.
- Alle Inhalte technisch korrekt für Screenreader formatiert.
- Lesbare, verständliche Blöcke statt unübersichtlicher Textwüsten.
- Eine sichtbare und maschinenlesbare Struktur – für Menschen und Suchmaschinen.

Copy-Paste-Baustein: Barrierefreiheits-Feedback

Und zum Schluss noch eine simple E-Mail-Vorlage, mit der du den Eingang der Meldung bestätigen kannst. Du kannst natürlich auch in deinem Mail-Programm das Ganze als Autoresponder einrichten, damit diese E-Mail sofort rausgeht, wann immer du eine erhältst. Kein Muss, aber in der Praxis doch sehr empfehlenswert.

Betreff: Wir haben Ihr Feedback zur Barrierefreiheit erhalten

Hallo [Name des/der Meldenden, falls angegeben],

vielen Dank für Ihre Nachricht an barrierefreiheit@dein-unternehmen.de. Wir haben Ihr Anliegen am [Datum des Eingangs] erhalten und kümmern uns darum.

Wir schätzen Ihr Engagement sehr, unsere digitalen Angebote noch zugänglicher zu machen. Ihr Hinweis ist wichtig für uns!

Sie erhalten eine weitere Nachricht von uns, sobald wir Ihr Feedback detailliert geprüft haben.

Mit freundlichen Grüßen,

Ihr Team von [Name deines Unternehmens]

Fragen & Antworten

Warum ist es wichtig, sich mit HTML-Grundlagen zu beschäftigen, auch wenn ich einen Page Builder nutze?

Auch wenn Page Builder viel Arbeit abnehmen und oft eine visuelle Bearbeitung ermöglichen, musst du wissen, was „unter der Haube“ passiert. Page Builder sind Werkzeuge, aber du bist derjenige, der die korrekten Einstellungen vornimmt.

Wenn du die Grundlagen von HTML und semantischer Auszeichnung verstehst, kannst du sicherstellen, dass dein Page Builder die gewünschte, barrierefreie Struktur auch wirklich umsetzt und nicht nur optisch etwas darstellt, was für Screenreader oder Suchmaschinen unverständlich ist.

Es ist wie das Verständnis der Architektur, selbst wenn du ein Fertighaus baust – du musst wissen, wo die tragenden Wände sind, um Probleme zu vermeiden.

Ich habe von ARIA-Attributen gehört, die bei der barrierefreien Formatierung helfen sollen – sollte ich sie nutzen?

ARIA-Attribute (Accessible Rich Internet Applications) sind spezielle HTML-Attribute, die zusätzliche semantische Informationen für assistierende Technologien wie Screenreader bereitstellen, wenn die native HTML-Semantik nicht ausreicht. Sie sind besonders nützlich für dynamische Inhalte oder komplexe Bedienelemente (z.B. Tabs, Slider, Dialogfelder), die nicht durch Standard-HTML-Elemente abgedeckt sind. Aber Achtung: Nutze ARIA nur, wenn es unbedingt notwendig ist und du genau weißt, was du tust. Der Grundsatz lautet: „First rule of ARIA is not to use ARIA.“ Wenn ein natives HTML-Element die gewünschte Semantik bereits bietet (z.B. ein `<button>` für einen Button), solltest du dieses bevorzugen. Eine falsche Anwendung von ARIA kann die Barrierefreiheit sogar verschlechtern!

Reicht es, wenn meine Schriftgröße nur auf dem Desktop 16px ist, oder muss sie auch auf Mobilgeräten so groß sein?

Die Angabe von mindestens 16px bezieht sich auf die Basisschriftgröße. Wichtig ist, dass diese Größe responsiv ist, also auch auf Mobilgeräten und kleineren Bildschirmen gut lesbar bleibt. Das bedeutet, du solltest flexible Einheiten wie `em` oder `rem` für deine Schriftgrößen verwenden, die sich an die Basisschriftgröße des Nutzers oder an die Bildschirmgröße anpassen. Stelle dir vor, du zoomst auf einem Handy in eine Website hinein und die Schrift wird immer noch als zu klein empfunden – das wäre nicht barrierefrei. Teste dies unbedingt auf verschiedenen Geräten!

Ich habe viele Bilder auf meiner Seite. Was muss ich bei der Barrierefreiheit von Bildern beachten?

Zu Bildern kommen wir, sobald wir den Abschnitt mit den SOP's zu Texten abgeschlossen haben.

Schon mal vorab: Du solltest immer Alt-Texte (Alternativtexte) für deine Bilder hinterlegen. Ein Alt-Text beschreibt den Inhalt oder die Funktion des Bildes kurz und prägnant für Screenreader-Nutzer:innen. Wenn das Bild rein dekorativ ist und keine Informationen vermittelt, sollte der Alt-Text leer sein (`alt=""`), damit der Screenreader es ignorieren kann. Stell dir vor, du kannst die Bilder nicht sehen – welche Information würdest du benötigen, um den Inhalt der Seite trotzdem vollständig zu verstehen?

Meine Seite ist sehr interaktiv mit vielen Formularen und Schaltflächen. Was muss ich hier bezüglich der Tastatur-Navigation beachten?

Für interaktive Elemente ist die Tastatur-Navigation entscheidend – wir werden sie ALLE in den nachfolgenden SOPs behandeln.

Du kannst dir aber schon jetzt einmal zwei Dinge dazu merken:

1. Alle interaktiven Elemente (Links, Buttons, Formularfelder etc.) müssen mit der Tab-Taste erreichbar sein und in einer logischen Reihenfolge durchlaufen werden können.
2. Außerdem muss der Fokus (eine sichtbare Umrandung oder Hervorhebung) des aktuell ausgewählten Elements klar erkennbar sein. Nur so können Menschen, die keine Maus nutzen können oder wollen, deine Website bedienen.

Denk daran, die Usability ist für alle wichtig, aber für Menschen mit motorischen Einschränkungen ist sie absolut essenziell.