

Keine Angst vor Barrierefreiheit

©Foto Freisinger, Thomas Eisenpass



Bildungsthema Juli – September 2015

Themen Expert/innen

Mag^a Monika Hirschmugl-Fuchs, CMC

Dipl.-Ing. Gerald Kortschak, BSc, CMC

Dipl.-Ing. Klaus D. Tolliner, MBA, CMC

Arbeitskreis **ECOBILITY** – barrierefreie Wirtschaft

www.ecobility.at

Inhaltsverzeichnis

Inhaltsverzeichnis.....	2
Abbildungsverzeichnis	5
Abkürzungsverzeichnis	6
Kapitel 3 Digitale Barrierefreiheit.....	8
Anmerkung zum Inhalt.....	9
Abgrenzung	10
Barrierefreiheit digitaler Inhalte verstehen.....	11
Selbstbestimmungsprinzip.....	12
Zwei-Sinne Prinzip.....	13
Universelles Design	13
Technische Grundlagen / Richtlinien.....	14
Prinzipien des WCAG.....	17
Wahrnehmbar	17
Bedienbar.....	18
Verständlich	18
Robust	18
Zusätzliche Dokumente	19
Accessible Rich Internet Applications.....	19
Umsetzung bei Websites	20
Barrierefrei HTML.....	22
Seitentitel, Meta-Tags und Dokumentsprache setzen	22
Seitentitel	23
Meta Tags.....	23
Anweisungen für Suchmaschinen-Roboter.....	24

Dokumentsprache.....	24
Seitenaufbau	25
Verwendung von <div>-Containern.....	26
CSS	26
HTML	27
Grafische Gestaltung.....	27
Trennung von Inhalt und Design	27
Gute Farbkontraste	27
Relative Textgrößen.....	28
Navigation	28
Robuste Funktionalität	29
Abgetrennt vom Inhalt.....	29
Aktueller Standort klar erkennbar.....	30
Sprunglinks	30
Accesskeys	31
Überschriften	32
Grafiken.....	33
Image-Maps.....	33
Schriftgrafiken	33
Logo als Grafik.....	34
Layoutgrafiken	34
Tabellen.....	34
Formulare	35
Gruppierung von Eingabefeldern	36
Rich Internet Applications	36
Rich Internet Applications und Barrierefreiheit	36
Barrierefreies AJAX	37

Live-Regionen	37
aria-atomic	38
aria-busy	38
aria-relevant	38
Barrierefreie Widgets	39
tabindex-Attribut für alle HTML-Elemente	39
Rollen von Widgets	39
Barrierefreies Flash	40
Flash für blinde Nutzer	40
Flash für Personen mit Sehbehinderung	40
Tastaturnavigation	41
Flash für Personen mit Hörbehinderung	41
Weiterführende Informationen:	41
Barrierefreies Silverlight	41
Silverlight für Blinde	42
Silverlight für Sehbehinderte Personen	42
Weiterführende Informationen unter:	42
Umsetzung bei PDF	43
Überprüfen von PDF-Dokumenten	44
Umsetzung bei Word	45
Literatur aus Kapitel 3	46

Abbildungsverzeichnis

Abbildung 1: Wirkungskreis positiver Effekte der digitalen Barrierefreiheit	10
Abbildung 2: Digitale Wechselbeziehungen als Ursache von Barriere	11
Abbildung 3: Wechselspiel WAI-Richtlinien.....	15
Abbildung 4: Prinzipien des WCAG.....	17
Abbildung 5: Schaukasten Farbkontraste	28

Abkürzungsverzeichnis

WKO	Wirtschaftskammer Österreich ist die Interessensvertretung von Unternehmer/innen in Österreich.
UBIT	Unternehmensberatung, Buchhaltung, Informationstechnologie ist eine Fachgruppe der Sparte Information und Consulting (IC).
IC	Information und Consulting ist eine der sieben Sparten der Österreichischen Wirtschaftskammer.
CMC	Certified Management Consultant ist ein internationales Zertifikat des „International Council of Management Consulting Institutes“ (ICMCI) und repräsentiert das weltweit stärkste Qualitätssiegel für Unternehmensberater/innen.
UN	UN steht für United Nations (VN Vereinte Nationen), oft auch UNO für United Nations Organization (Organisation der Vereinten Nationen) sind ein Zwischenstaatlicher Zusammenschluss und ein uneingeschränkt anerkanntes Völkerrechtssubjekt.
PA	Persönliche Assistenz ist eine Form der persönlichen Hilfe, die behinderten Menschen ein selbstbestimmtes Leben ermöglicht.
ÖGS	Österreichische Gebärdensprache, die gesetzlich anerkannte Sprache gehörloser Menschen.
ÖGLB	Österreichischer Gehörlosenbund ist die Interessenvertretung der Gehörlosengemeinschaft in Österreich. Zu ihr gehören alle, die bevorzugt in ihrer Gebärdensprache kommunizieren.
BGStG	Bundesbehindertengleichstellungsgesetz hat das Ziel, die Diskriminierung von Menschen mit Behinderungen zu beseitigen oder zu verhindern und damit die gleichberechtigte Teilhabe von Menschen mit Behinderungen am Leben in der Gesellschaft zu gewährleisten.
DfA	Design für Alle ist ein Konzept für die Planung und Gestaltung von Produkten, Dienstleistungen und Infrastrukturen, mit dem Ziel, allen Menschen deren Nutzung ohne individuelle Anpassung oder besondere Assistenz zu ermöglichen. DfA wird auch als universelles Design bezeichnet.
WAI	Web Accessibility Initiative ist ein Bereich innerhalb des World Wide Web Consortiums (W3C), in der sich mehrere Arbeitsgruppen und Interessensgruppen mit dem barrierefreien Zugang zum Web und seinen Inhalten beschäftigen.
W3C	World Wide Web Consortium ist das Gremium zur Standardisierung der Techniken im World Wide Web.
WCAG	Web Content Accessibility Guidelines sind eine Empfehlung der Web Accessibility Initiative (WAI) des World Wide Web Consortiums (W3C) zur barrierefreien Gestaltung der Inhalte von Internetangeboten. Webseiten, die diesen Richtlinien entsprechen, sind auch für Menschen mit sensorischen und motorischen (und in gewissem Rahmen mentalen) Einschränkungen zugänglich, d.h. sie können die

angebotenen Informationen erfassen und notwendige Eingaben tätigen.

GUI	Graphical user interface ist die englische Bezeichnung für grafische Benutzeroberfläche oder auch grafische Benutzerschnittstelle.
ATAG	Authoring Tools Accessibility Guidelines sind Anforderungen an Werkzeugen zur Website-Erstellung.
UAAG	User Agent Accessibility Guidelines sind Anforderungen an Browser und Mediaplayer.
WHO	World Health Organization, Weltgesundheitsorganisation ist eine Sonderorganisation der Vereinten Nationen mit Sitz in Genf.
§	Paragraf
Abs.	Absatz
Z	Ziffer
GewO	Gewerbeordnung
EGVG	Einführungsgesetz zu den Verwaltungsverfahrensgesetzen

Kapitel 3 Digitale Barrierefreiheit

Unser Jahrhundert ist durch den digitalen Austausch von Informationen gekennzeichnet. Unternehmen legen hierbei besonderes Augenmerk auf die Bereitstellung von Informationen über Produkte und Dienstleistungen. Vom Onlineshop über die Bewerbung der Dienstleistung auf der eigenen Website oder der Kommunikation mit Kunden. Wie üblich gilt auch hier die Grundregel des Internets, wer seine Informationen schnell und übersichtlich zur Verfügung stellt hat einen Wettbewerbsvorteil.

Kapitel 3 gibt einen Überblick über digitale Barrierefreiheit. Dabei wird gezeigt, warum digitale Barrierefreiheit so wichtig ist, welche Personengruppen ohne digitale Barrierefreiheit diskriminiert werden und was beim Erstellen von Web Content wichtig ist.

Hierbei bietet eine barrierefreie Gestaltung von digitalen Inhalten den Vorteil, dass eine Vielzahl von Personenkreisen und Anwendungsbereichen davon abgedeckt werden. Beispielsweise wird die Auffindbarkeit durch Suchmaschinen unterstützt, sowie die Darstellung von Inhalten auf mobilen Endgeräten. Die nachfolgende Grafik stellt eine kurze Übersicht dieses Wirkungsfeldes dar.

Ihre Vorteile hierbei sind:

- Verbesserte Reichweite
 - Nicht nur, dass eine eventuell sogar neue Kundengruppe Ihre Inhalte nun betrachten kann, mit entsprechender Umsetzung ist die Seite auch wesentlich kompatibler für Browser und mobile Endgeräte
- Gute Grundlage für Suchmaschinen
 - Suchmaschinenanbieter nutzen sogenannte Suchroboter. Kleine Programme die Webinhalte durchsuchen um so die Website für die Suchmaschine zu indizieren und in der Folge zu listen. Eine schlechte Indizierung führt zu einer Reihung als Schlusslicht. Suchroboter scheitern bei Websites aufgrund ihres automatisierten Vorgehens an den gleichen Hürden wie behinderte Personen. Daher führen barrierefreie Websites auch zu einer besseren Indizierung
- Image und Einhaltung des Behindertengleichstellungsgesetzes

- Vor dem Gesetz stand an dieser Stelle lediglich Image, da es keine Verpflichtung gab, die Inhalte barrierefrei aufzubereiten. Neben der Einhaltung eines Gesetzes gilt aber nach wie vor, dass ein barrierefreier Auftritt sich positiv auf das Corporate-Image eines Unternehmens auswirkt und insbesondere auch nachhaltig wirkt. Es ist bekannt, dass eine Überalterung der Bevölkerungsschicht vorliegt und im Steigen begriffen ist. Wie im Wirkungskreis der Abbildung dargestellt wird auch dieser Bereich damit abgedeckt und das Unternehmen bekundet niemanden auszugrenzen.
- Nachhaltig und Kostensparend
 - Zur Umsetzung ist es erforderlich Standards und Richtlinien der Webentwicklung einzuhalten. Dies führt dazu, dass Ihre Website eine sogenannte Rückwärts- und Vorwärtskompatibilität aufweist. Dies bedeutet, dass sie in Browsern unterschiedlicher Entwicklungsstufen betrachtbar ist und so auch auf neuesten Systemen nutzbar ist. Wenn im Fall der Fälle dennoch eine Anzeige aufgrund einer völlig neuen Browsertechnologie nicht mehr möglich ist, so werden die Kosten für eine Anpassung reduziert. Darüber hinaus führen barrierefreie Inhalte zu einer strukturierten Darstellung und Verarbeitung von Informationen, womit auch Wartungskosten reduziert werden können.

Anmerkung zum Inhalt

Im Inhalt werden zum besseren Verständnis auch Angaben in Quellcode angeführt. Zur Vereinfachung der Ansicht wurde hierbei bewusst auf eine codierte Darstellung von Umlauten (ä == ä) verzichtet.



Abbildung 1: Wirkungskreis positiver Effekte der digitalen Barrierefreiheit¹

Abgrenzung

Dieses Kapitel behandelt hierbei den barrierefreien Zugang zu digitalen Inhalten im Raum des Internet, nicht jedoch die barrierefreie Gestaltung von Arbeitsplätzen, da dies ausreichend definiert ist und in einer derartigen Situation davon auszugehen ist, dass eine entsprechend produktive Absprache zwischen Arbeitgeber und Arbeitnehmer durchgeführt wird. Für weiterführende Informationen wird hier auf die Website des Zentral-Arbeitsinspektorats des Bundesministeriums für Arbeit, Soziales und Konsumentenschutz², sowie das Portal help.gv.at³ verwiesen.

¹ Quelle: <http://www.einfach-barrierefrei.net> - gesichtet 12.08.2015

² <http://www.arbeitsinspektion.gv.at/AI/Arbeitsstaetten/Allgemeines/arbeitsstaetten030.htm>

³ <http://www.help.gv.at>

Barrierefreiheit digitaler Inhalte verstehen

Um die Vorgehensweise verstehen zu können ist es erforderlich zu verstehen, wo und wie diese Barrieren entstehen. Dies veranschaulicht die nachstehende Abbildung.



Abbildung 2: Digitale Wechselbeziehungen als Ursache von Barriere⁴

Hierbei wiederholt es sich immer, dass Personen mit Behinderung auf folgende primäre Barrieren treffen:

- fehlende Navigationshilfen
- schlechter Farbkontrast
- nicht mit der Tastatur bedienbar
- zu kurze Timeouts

Nun gibt es natürlich unterstützende Hilfsmittel die diesem Personenkreis dabei helfen, eine Website mit den oben angeführten Barrieren zu bedienen. Diese sogenannten assistiven Technologien wären beispielsweise:

- Screenreader
- Braillezeile
- Bildschirmlupe
- Joystick
- Mundmaus

⁴ Quelle: <http://www.einfach-barrierefrei.net/verstehen> - gesichtet 12.08.2015

Diese Technologien gibt es, aber diese kosten auch entsprechend Geld und sind nicht jederzeit und überall zur Stelle. Darüber hinaus wird der Konsument quasi in die Pflicht genommen, über die entsprechenden Hilfsmittel zu verfügen, um die vom Unternehmen angebotene Information aufzunehmen. Das hat ungefähr den gleichen Beigeschmack als würde man einer Person im Rollstuhl erklären, dass es keinen Lift im Gebäude gibt, da er eine Treppenraupe an seinem Rollstuhl montieren könnte und so ebenfalls in der Lage ist die Stufen zu bewältigen. Viel besser wäre es daher auch digitale Inhalte nach drei Grundprinzipien auszurichten. Diese wären:

- Selbstbestimmungsprinzip
- Zwei-Sinne Prinzip
- Universelles Design

Selbstbestimmungsprinzip

Das Selbstbestimmungsprinzip definiert, dass Websites nicht durch eine starre Gestaltung umfassend eingegrenzt sind und es daher für den User keine Möglichkeit mehr gäbe, selbstbestimmt Teilbereiche der Darstellung zu verändern. Hierunter fallen Möglichkeiten wie die Veränderung der Schrittgröße, die Anpassung der Kontraste oder die Einstellung der Farbschemas. Wird das Selbstbestimmungsprinzip eingehalten, so ist jeder User eigenständig in der Lage mit den ihm verfügbaren Hilfsmitteln die Seite so einzustellen, dass die Aufnahme der Informationen ein Leichtes ist.

In diesem Zusammenhang kann auf der Website <http://www.eyesyde.de/de/> kostenfrei ein Tool geladen werden, mit dessen Hilfe Webseiten mit unterschiedlichen simulierten Farbsehschwächen von normalsichtigen Menschen betrachtet werden können. So kann man einen Eindruck erhalten, wie eine Website ohne Selbstbestimmungsprinzip auf diese Personen wirkt. Gerade bei Webseiten als Marketing-Instrument spielt die Farbgestaltung eine wichtige Rolle. Das Programm ist leider nur für Windows Systeme mit .NET Framework 4.0 verfügbar. Dieser Umstand soll aber nicht als Hinweis an Entwickler und Programmierer ungenutzt bleiben. Es besteht ein Mangel insbesondere an ausgefeilten Open-Source-Tools für Analysen von Webseiten bzw. zur Darstellung

von Webinhalten mit unterschiedlichen Einschränkungen nach der Auffassung des Autors zum Zeitpunkt der Erstellung dieses Kapitels.

Zwei-Sinne Prinzip

Das Zwei-Sinne-Prinzip basiert grundsätzlich darauf, dass die Wahrscheinlichkeit für eine Person mit gleich zwei eingeschränkten Sinnen sehr gering ist. So ist es eher wahrscheinlich, dass eine akustisch beeinträchtigte Person noch sehen kann, oder eine sehbeeinträchtigte Person noch hören kann. Dies ist leider nicht allgemein gültig, dennoch findet das Zwei-Sinne Prinzip in vielen Bereichen Anwendung und auch hohe Akzeptanz. Im digitalen Raum sind dies beispielsweise Untertitel von Filmen für Gehörlose. Versuchen Sie es selbst indem Sie bei einem Film auf Ihrer DVD (oder welches Medium auch immer) den Untertitel für Gehörlose einblenden lassen. Dies hilft auch besser zu verstehen, wie beispielsweise Bildmaterial auf Websites zu beschreiben ist.

In der Schlussfolgerung besagt das Zwei-Sinne-Prinzip daher, dass alle Informationen so angeboten werden sollten, dass sie gleichzeitig zwei Sinne bedienen können:

- Akustische Signale durch visuelle Signale ergänzen
- Lese-Text auch als Sprachausgabe anbieten und umgekehrt

Universelles Design

Dieses Prinzip ist entstanden durch eine sehr bedauernswerte Entwicklung. Oftmals wurde aus der Not heraus, oder im guten Glauben es „noch“ besser zu tun, für Personen mit Behinderung ein alternativer Zugang geschaffen. Man kennt dies durch Menüpunkte auf Websites mit der Beschriftung „Zur barrierefreien Version“ oder durch separate Gebäudezugänge. Im digitalen Bereich ist ein Beispiel für jeden auch die oft nervende Option „Darstellung der Website für mobile Geräte“ oder die Frage „Wollen Sie zur mobilen Version wechseln?“. Gelegentlich erkennt die Website das Gerät falsch und man erhält im Browser des Notebooks oder PCs schon eine mobile Website oder umgekehrt. Vielen wird hierbei auch der große Nachteil aufgefallen sein: So wie mobile Websites oft nicht alle Inhalte im gleichen Umfang wiedergeben oder es zu einer verfälschten Darstellung kommen kann, so wurden auch digitale Inhalte in der „barrierefreien“ Version oftmals nur reduziert angeboten. Weiters müssen somit zwei

Versionen aktualisiert und gewartet werden, der Button muss erst gefunden werden und der Aufwand während der Gestaltung ist fast doppelt so hoch - dies schlägt sich auch in den Kosten nieder und ist natürlich keine optimale Lösung.

Das Prinzip des universellen Designs geht daher den Weg, dass es einen Zugang, eine einzige Lösung für alle gibt. Die ausführliche Umsetzung dieser Grundprinzipien wurde in den Richtlinien für Barrierefreiheit des W3C umfassend erfasst und wird daher im nächsten Unterkapitel behandelt.

Technische Grundlagen / Richtlinien

Es gibt für die barrierefreie Gestaltung von Webseiten keine direkten Vorgaben. Sehr wohl gibt es aber Richtlinien. Die bekannteste und zumeist auch zur Anwendung gebrachte ist die WCAG 2.0⁵ die auch in einer deutschen Übersetzung aufliegt. WCAG steht für Web Content Accessibility Guideline und diese Richtlinie stellt eine Sammlung von Empfehlungen dar, wie Inhalte barrierefrei gestaltet werden können und ist quasi die Pflichtliteratur für jeden Webdesigner im Bereich Barrierefreiheit. WCAG in der Version 2.0 stellt die zentralen Richtlinien dar. Erstellt wurden diese vom W3C⁶, dem World Wide Web Consortium im Rahmen der WAI, der Web Accessibility. Diese wurden auch als Komponenten in Zusammenhang gebracht und stellen sich wie folgt dar:

⁵ <http://www.w3.org/Translations/WCAG20-de/>

⁶ <http://www.w3.org>



Abbildung 3: Wechselspiel WAI-Richtlinien⁷

WCAG 2.0 wurde bereits angeführt. WAI-AIRI⁸ ist eine Erweiterung der WAI zu Accessible Rich Internet Applications (kurz ARIA) und ist speziell für Websites mit dynamischen Inhalten entworfen worden, die beispielsweise mit AJAX (einer Programmiersprache) realisiert wurden. ATAG⁹ steht für Authoring Tool Accessibility Guidelines und definiert die Richtlinien für Software mit der Webseiten erstellt werden (Authoring Tools – Autoren Werkzeuge). Daneben gibt es noch die User Agent Accessibility Guidelines¹⁰. Die UAAG definieren die Richtlinien für sogenannte User Agents, das sind beispielsweise Web-Browser, Media-Player oder die Eingangs erwähnten assistiven Technologien.

Hieraus resultiert das Zusammenspiel der Komponenten und die W3C-WAI Konformität. Diese Konformität wird auch als „Level“ bezeichnet und es gibt diese in drei Stufen.



Konformität Stufe "A": Alle Checkpunkte der Priorität 1 sind erfüllt;

⁷ Quelle: <http://www.einfach-barrierefrei.net/verstehen/standards/wai.html> - gesichtet 12.08.2015

⁸ <http://www.w3.org/TR/wai-aria/>

⁹ <http://www.w3.org/WAI/intro/atag.php>

¹⁰ <http://www.w3.org/WAI/intro/uaag.php>



- **Konformität Stufe "Double-A"**: Alle Checkpunkte der Priorität 1 und 2 sind erfüllt;



- **Konformität Stufe "Triple-A"**: Alle Checkpunkte der Priorität 1, 2 und 3 sind erfüllt;¹¹

Mit Hilfe der entsprechenden Logos kann auf der eigenen Website dargestellt werden, welche Konformitätsstufe eingehalten wird.

Diese drei Stufen sind auch entscheidend bei der Herangehensweise an eine barrierefreie Website. Sie sollten zu Beginn klar definieren, welche Konformitäts-Stufe von Ihrer Website und damit der realisierenden Stelle einzuhalten ist, wobei die Stufe realisierbar und adäquat sein sollte. Aus diesem Grund wird auch schon im WAI darauf hingewiesen, dass es keinen Sinn ergibt grundsätzlich die Stufe AAA einzufordern, da es Inhalte gibt deren konforme Darstellung technisch noch nicht möglich ist. Weiters basieren moderne Website überwiegend auf Content-Management-Systemen (CMS) und hierbei bedient man sich auch sogenannter Plug-Ins. Dies sind Erweiterungsmodule, beispielsweise umfassende Kontaktformulare. Diese Plug-Ins stammen von Drittanbietern und können nicht immer auf ihre Konformität der Richtlinie entsprechend geprüft werden und dürfen daher auch nicht als konform ausgewiesen werden. Für eine Überprüfung kann beispielsweise folgende Website genutzt werden: <http://www.cynthiasays.com>. Auf dieser Website kann angegeben werden, welcher Level eingehalten worden sein sollte. Die Interpretation der Testergebnisse erfordert jedoch entsprechendes technisches Wissen über Webentwicklung, da sämtliche Ergebnisse gelistet werden mit Bezug auf die entsprechende HTML-Syntax. Eine Überprüfung nach dem ja/nein-Prinzip findet nicht statt. Dies gilt grundsätzlich für sämtliche kostenfrei verfügbaren Analyse-Tools im Zusammenhang mit barrierefreien Webseiten.

¹¹ Quelle: <http://www.w3c.at/Trans/WAI/webinhalt.html#Conformance> - gesichtet am 08.08.2015

Prinzipien des WCAG

Das WCAG besteht im Wesentlichen aus vier Grundprinzipien von welchen zwölf Richtlinien abgeleitet werden, wobei diese Richtlinien als Anweisungen als Entwickler formuliert wurden, womit die Umsetzung und Einhaltung der Richtlinien grundsätzlich erleichtert wird.

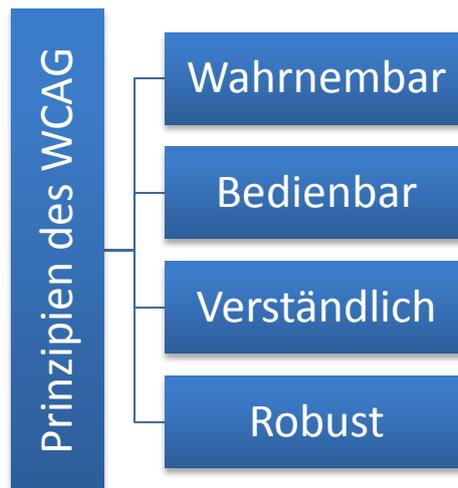


Abbildung 4: Prinzipien des WCAG

Im folgenden werden die vier Prinzipien mit den zugehörigen Richtlinien ausgeführt, wobei diese als direkte Zitate der Website www.w3.org übernommen wurden.

Wahrnehmbar

Wahrnehmbar bedeutet: *Informationen und Bestandteile der Benutzerschnittstelle müssen den Benutzern so präsentiert werden, dass sie diese wahrnehmen können.*¹²

- Stellen Sie Textalternativen für alle Nicht-Text-Inhalte zur Verfügung, so dass diese in andere vom Benutzer benötigte Formen geändert werden können, wie zum Beispiel Großschrift, Braille, Symbole oder einfachere Sprache.
- Stellen Sie Alternativen für zeitbasierte Medien zur Verfügung.
- Erstellen Sie Inhalte, die auf verschiedene Arten dargestellt werden können (zum Beispiel mit einfacherem Layout), ohne dass Informationen oder Strukturen verloren gehen.

¹² Quelle: <http://www.w3.org/Translations/WCAG20-de/#perceivable> gesichtet am 8.8.2015

- Machen Sie es für den Benutzer leichter, Inhalte zu sehen und zu hören, einschließlich der Trennung zwischen Vordergrund und Hintergrund.

Bedienbar

Bedienbar bedeutet: *Bestandteile der Benutzerschnittstelle und Navigation müssen bedienbar sein.*¹³

- Sorgen Sie dafür, dass alle Funktionalitäten von der Tastatur aus verfügbar sind.
- Geben Sie den Benutzern ausreichend Zeit, Inhalte zu lesen und zu benutzen.
- Gestalten Sie Inhalte nicht auf Arten, von denen bekannt ist, dass sie zu Anfällen führen.
- Stellen Sie Mittel zur Verfügung, um Benutzer dabei zu unterstützen zu navigieren, Inhalte zu finden und zu bestimmen, wo sie sich befinden.

Verständlich

Verständlich bedeutet: *Informationen und Bedienung der Benutzerschnittstelle müssen verständlich sein.*¹⁴

- Machen Sie Textinhalte lesbar und verständlich.
- Sorgen Sie dafür, dass Webseiten vorhersehbar aussehen und funktionieren.
- Helfen Sie den Benutzern dabei, Fehler zu vermeiden und zu korrigieren.

Robust

Robust bedeutet: *Inhalte müssen robust genug sein, damit sie zuverlässig von einer großen Auswahl an Benutzeragenten einschließlich assistierender Techniken interpretiert werden können.*¹⁵

- Maximieren Sie die Kompatibilität mit aktuellen und zukünftigen Benutzeragenten, einschließlich assistierender Techniken.

¹³ Quelle: <http://www.w3.org/Translations/WCAG20-de/#operable> gesichtet am 8.8.2015

¹⁴ Quelle: <http://www.w3.org/Translations/WCAG20-de/#understandable> gesichtet am 8.8.2015

¹⁵ Quelle: <http://www.w3.org/Translations/WCAG20-de/#robust> gesichtet am 8.8.2015

Zusätzliche Dokumente

Weiterführende Informationen zum WCAG finden sich auch unter:

- How To Meet WCAG 2.0 (englisch): Eine Kurzreferenz für WCAG 2.0
- <http://www.w3.org/WAI/WCAG20/quickref/>
- Understanding WCAG 2.0 (englisch): Zusätzliche Details und Hintergrundwissen zu den Richtlinien und Erfolgskriterien
<http://www.w3.org/TR/UNDERSTANDING-WCAG20/>
- Techniques for WCAG 2.0 (englisch): Allgemeine und technologiespezifische Beispiele
<http://www.w3.org/TR/WCAG20-TECHS/>

Accessible Rich Internet Applications

Die WAI-ARIA ist eine Erweiterung der Web Accessibility Initiative zum Erstellen von barrierefreien Rich Internet Applikationen. Sie wurde erst 2010 verabschiedet und ist damit der jüngste Teil der WAI.

Die Spezifikation basiert auf drei Funktionsbausteinen:

1. Landmark Roles

Durch Landmarks können die Regionen einer HTML-Seite (Navigation, Header, Footer) bezeichnet und dadurch von Browsern und assistiven Technologien besser verstanden werden.

2. Unterstützung von emulierten Bedienelementen

HTML unterstützt nur sehr einfache Eingabeelemente. Elemente wie Slider, Groupbox oder Tree, werden in modernen AJAX Applikationen deshalb aus "Einzelteilen" nachgebaut und sind nicht barrierefrei. Mit den ARIA-Erweiterungen können diese emulierten Elemente durch folgende Erweiterungen barrierefrei gemacht werden:

- Rollen zur Beschreibung der Funktion
- Tastaturnavigation für alle HTML-Elemente über das tabindex-Attribut
- Status und Eigenschaften, als Info für assistive Technologien

3. Live Regions

In AJAX Applikationen können Teile der Webseite dynamisch nachgeladen oder verändert werden. Assistive Technologien, wie Braillezeilen oder Screenreader bekommen dies jedoch nicht mit. Dazu wurde mit der ARIA sogenannte Live-Regions eingeführt, welche über ein Attribut beschreiben, dass diese Region dynamisch verändert werden kann.

Weiterführende Informationen unter:

- Accessible Rich Internet Applications Spezifikation <http://www.w3.org/TR/wai-aria/>
- Einführung der WAI zu WAI-ARIA <http://www.w3.org/WAI/intro/aria>

Umsetzung bei Websites

Bei der Gestaltung von barrierefreien Webseiten gilt es in mehreren Stufen voranzuschreiten. Einerseits wurde eingangs bereits erklärt, dass es drei unterschiedliche Level der Konformität nach WAI gibt. Daher gilt es zu definieren welche dieser Stufen erreicht werden sollte. Diese Definition kann nicht vom Webentwickler vorgegeben werden, sondern ist eine Entscheidung der Unternehmensführung, bei der selbstverständlich beratende Informationen eingeholt werden können. In der Folge muss man sich bewusst sein, dass bei einem universellen Design eventuell Abstriche bei den Realisierungswünschen in Kauf genommen werden müssen. Es ist nicht immer möglich, sowohl Barrierefreiheit als auch marketingtechnische Vorgaben unter einen Hut zu bringen. Aus diesem Grund ist es bei der Gestaltung von barrierefreien Webseiten von Vorteil, wenn jemand eine beratende Vermittlerrolle zwischen Marketing, Geschäftsführung und Webentwicklung wahr nimmt, um Konflikte vorzubeugen, Probleme frühzeitig aufzudecken und ganzheitlich ein besseres und nachhaltigeres Resultat sicher zu stellen.

Grundsätzlich wurden und werden Webseiten im reinem HTML (HyperTextMarkupLanguage¹⁶) aufgebaut. HTML wurde von Tim Berners-Lee als Sprache geschaffen, um Inhalte über Browser im Internet zu verteilen. Heute kennt

¹⁶ <http://wiki.selfhtml.org/wiki/Startseite>

diese jeder durch die Endung .html bei einzelnen Webseiten. Es wurde als reine Auszeichnungssprache (Markup Language) geschaffen, damit Elemente eines textbasierenden Dokumentes dargestellt werden können. Derartige Elemente sind Überschriften (z.B. <h1/>), Textabsätze (z.B. <p/>), Listen (z.B.), Tabellen (<table><tr><td></td></tr></table>) und Bildelemente (). HTML steht aktuell in der Version HTML5¹⁷ zur Verfügung, wobei das W3C bereits an der Version HTML5.1¹⁸ arbeitet. Zusätzlich gibt es noch XHTML. Das Extended HTML wurde vor HTML5 als Zukunft betrachtet, hatte sich aber auch in der Version XHTML2 nicht durchsetzen können. Daher sollten heutige Webentwickler sich primär nach HTML5 orientieren. Diese kurze Einführung in HTML sollte Ihnen einen grundsätzlichen Überblick verschaffen. Auf der Website des Web-Projektes SELFHTML (<http://www.selfhtml.org>) finden Sie alle Informationen, sowie auch Anleitungen zum Erlernen von HTML.

Moderne Webseiten werden nicht mehr Zeile für Zeile in HTML oder in sogenannten WYSIWYG¹⁹-Editoren aufgebaut, sondern basieren auf Content-Management-Systemen wie Joomla²⁰, Typo3²¹, Wordpress²² und vielen weiteren mehr. Diese CMS²³ bieten den Vorteil, dass die Inhalte dynamisch verändert werden können und der Fokus der Verwaltung auf dem Inhalt der Webseite liegt. Dieser Inhalt ist in Datenbanken gespeichert und wird anhand sogenannter Templates grafisch im Webbrowser umgesetzt. Für eine ausführliche Erklärung zum Thema CMS wird auf das Internet verwiesen bzw. die Webseiten unterschiedlicher Anbieter die dieses Thema dort ausführlich darstellen. Wesentlich ist für den Aspekt der Barrierefreiheit, dass es für die meisten CMS entsprechende barrierefreie Templates, also Darstellungsformen, sowie Erweiterungsmodule gibt, um beispielsweise Textelemente vorzulesen, zu vergrößern oder die Farbgestaltung anzupassen. Es ist bei Webseiten mit einem CMS nicht erforderlich sämtliche Passagen des HTML-Codes zu durchlaufen. Bei einem CMS kann über ein entsprechendes Kontextmenü ganz einfach der Alternativ-Text für ein Bild eingetragen werden und es muss nicht der „alt“-Tag im HTML definiert werden. Dies

¹⁷ <http://www.w3.org/TR/html5/>

¹⁸ <http://www.w3.org/html/wg/drafts/html/master/>

¹⁹ What-You-See-Is-What-You-Get

²⁰ <http://www.joomla.org>

²¹ <http://typo3.org>

²² <https://wordpress.org>

²³ <https://de.wikipedia.org/wiki/Content-Management-System>

stellt für die Erstellung, Wartung und den Betrieb von barrierefreien Webseiten eine enorme Erleichterung dar.

Barrierefrei HTML

Es gilt 6 Grundregeln zu beachten:

1. Seitentitel, Meta-Tags und Dokumentsprache setzen
2. Layout über <div> Elemente
3. Inhalt und Design durch CSS trennen, gute Farbkontraste und flexible Größen
4. Robuste Navigation, vom Inhalt abgetrennt, mit aktueller Auswahl erkennbar
5. Überschriften richtig setzen und einsetzen
6. Grafiken mit dem alt-Attribut beschreiben und Schriftgrafiken vermeiden

Diese werden in den nachstehenden Unterkapiteln kurz erklärt. Diese Beschreibung erhebt keinen Anspruch auf Vollständigkeit und ersetzt keine HTML-Schulung.

Seitentitel, Meta-Tags und Dokumentsprache setzen

Schon längst genügt es nicht mehr, nur guten Inhalt auf der Website zu haben, um sie richtig und ideal für ein gutes Suchmaschinen-Ranking darzustellen. Ganz im Gegenteil, ohne die Angabe zusätzlicher Meta-Informationen ist es schwierig, sich gut zu positionieren und barrierefrei zu agieren.

Zu den benötigten Metatags gehören zum Beispiel folgende Informationen:

- Sprache
- Verwendeter Zeichensatz
- Titel
- Kurzbeschreibung
- Keywords
- Autor

Diese Informationen werden in Form von <meta>-Tags im Kopf der HTML-Seite platziert und Browser, Proxy-Server, Suchmaschinenroboter oder assistive Technologien zur

Verfügung gestellt. Diese interpretieren diese Daten und nutzen sie, um die Inhalte unterschiedlich zu behandeln.

Seitentitel

Der Seitentitel sollte kurz und prägnant sein und möglichst präzise den Inhalt der Seite beschreiben. Verwenden Sie dafür Schlüsselwörter, die sich auch im Inhalt der Seite widerspiegeln. Es gibt verschiedene Zwecke, für die der Titel wichtig ist:

- Titelzeile des Browserfensters
- Bezeichnung von Lesezeichen
- Überschrift in Suchresultaten

```
<title>Keine Angst vor Barrierefreiheit - Kontakt</title>
```

Meta Tags

Die nachfolgend beschriebenen Meta-Tags sollten im Header jeder HTML-Seite enthalten sein. „title“, „description“ und „keywords“ sind vor Allem im Zusammenhang mit einem guten Suchmaschinen-Ranking besonders wichtig.

```
<head>
  <!-- Zeichensatz -->
  <meta http-equiv="content-type" content="text/html;
charset=UTF-8" />
  <!-- Inhaltssprache -->
  <meta name="content-language" content="de" />
  <!-- Seitentitel -->
  <meta name="title" content="Keine Angst vor
Barrierefreiheit" />
  <!--Inhaltsbezogene eindeutige Schlüsselwörter -->
  <meta name="keywords"
content="Barrierefreiheit, Behindertengleichstellung, ..." />
  <!-- Kurzbeschreibung -->
  <meta name="description" content="Alle Informationen zu
Barrierefreiheit" />
  <!-- Autor -->
  <meta name="author" content="Max Mustermann" />
  <!-- Publiziert durch -->
  <meta name="publisher" content="Max Mustermann" />
  <!-- Copyright -->
```

```
<meta name="copyright" content="Max Mustermann" />
</head>
```

Anweisungen für Suchmaschinen-Roboter

Suchmaschinen senden Ihre „Roboter“ regelmäßig ins Internet aus, um Seiten zu indexieren, Inhalte zu erfassen, und laufend alle Hinweise zu Websites zu aktualisieren.

Das Meta-Tag „robots“ gibt den Suchrobotern Hinweise, ob der Inhalt dieser Webseite gelesen und verzeichnet werden soll oder nicht. Mit der Zusatzinformation „follow“ kann außerdem bestimmt werden, ob auch die Links der jeweiligen Seite indexiert werden sollen. Eine weitere Zusatzinformation „revisit-after“ gibt an, wie häufig Sie die Inhalte aktualisieren, und damit einen Wiederbesuch und Neu-Indexierung der Seite durch die Suchroboter wünschen.

```
<!-- Indexierung dieser Seite: index oder no-index -->
<!-- Links ebenfalls indexieren: follow oder no-follow -->
<meta name="robots" content="index,follow" />

<!-- Wiederbesuchszeit in Tagen (days), Wochen (weeks) oder
Monaten (months) -->
<meta name="revisit-after" content="2 days" />
```

Dokumentsprache

Werkzeuge zur Unterstützung behinderter Menschen benötigen eine Hilfestellung, um den vorhandenen Text in der richtigen Sprache zu interpretieren. Diese Hilfestellung ist die Angabe der verwendeten Sprache im Kopf der jeweiligen Seite:

```
<html lang="de" xml:lang="de">
...
</html>
```

Sollten einzelne Textteile oder Wörter in anderer Sprache vorkommen werden diese mit einem -Tag umschlossen und gesondert definiert. Das ermöglicht es zum Beispiel Screenreadern, alle Wörter mit richtiger Betonung und Aussprache auszugeben.

```
<p>Diese Definition hilft Ihrem <span lang="en"
xml:lang="en">Screenreader</span> bei der richtigen
Aussprache</p>
```

Seitenaufbau

Einzelne Seiten Ihres Webauftrittes bestehen immer aus mehreren Inhaltblöcken. Bereiche wie Seitenkopf, Menü, Inhalt und Fußzeile sollten in einer definierten Reihenfolge und Anordnung angezeigt werden – für diese richtige Anordnung bietet HTML mehrere Möglichkeiten. Tabellen, Frames und div-Container liefern optisch dasselbe Ergebnis, in Hinblick auf Barrierefreiheit jedoch stellen Sie große Unterschiede dar.

- Tabellen – NICHT EMPFOHLEN
 - Layouttabellen haben zwei gravierende Nachteile:
 - ein mit Tabellen erstelltes Layout ist starr und kaum durch Stylesheets anpassbar
 - Screenreader erkennen Tabellen als Darstellungsform für Daten. Aus diesem Grund erfolgt ein zeilenweises Vorlesen, was nicht immer der gewünschten Aussage zugute kommt und Inhalte oft zerstückeln kann
- Frames – EINGESCHRÄNKT EMPFOHLEN
 - Eine bessere Alternative sind Frames, da diese von den meisten modernen Browsern und assistiven Technologien verstanden und interpretiert werden können. Nachteil bei Frames ist, dass ähnlich wie bei Tabellen Inhalte von Screenreadern zeilenweise gelesen werden, was zum Verlust der Semantik Ihrer Inhalte führen kann.
- div-Container - EMPFOHLEN
 - Als Layoutierungsmöglichkeit der Wahl sollten div-Container gelten. Sie werden von allen Browsern und assistiven Technologien richtig interpretiert und können vollkommen frei und flexibel auf der Seite

angeordnet werden. Die Reihenfolge der Interpretation für Screenreader kann frei definiert werden, die dargestellte Reihenfolge muss hier nicht zwingend der Reihenfolge im Quellcode entsprechen. Zusätzlich sind auch Textbrowser in der Lage, div-Container richtig zu interpretieren.

Verwendung von <div>-Containern

Auf den ersten Blick mag der Aufbau Ihres Seitenlayouts mittels div-Container umfangreicher und komplexer wirken als der Einsatz von Frames oder Tabellen. Sobald Sie jedoch kleine Layoutänderungen, ein zweites Design oder die Erstellung einer mobiltauglichen Variante Ihrer Seite anstreben werden Sie feststellen, dass Sie durch den Aufbau mit Containern erheblich flexibler sind und sehr viel Zeit sparen.

Das folgende Beispiel zeigt ein einfaches Layout mit zwei Spalten: einer Navigations- und einer Inhalts-Spalte. Diese teilen sich die Breite im Verhältnis 30% zu 70%.

CSS

```
html, body
{
  margin: 0;
  padding: 0;
  text-align: center;
}

#container
{
  width: 100%;
  text-align: left;
  margin: 0 auto;
}

#navigation
{
  width: 30%;
  float: left;
  position: relative;
}

#content
{
  position: relative;
```

```
width: 70%;  
float: right;  
}
```

HTML

```
<div id="container">  
  <div id="navigation"></div>  
  <div id="content"></div>  
</div>
```

Grafische Gestaltung

Mitunter den größten Einfluss auf die Barrierefreiheit Ihrer Webseite hat die Wahl der grafischen Gestaltung. Werden Farben, Kontraste und Textgrößen ohne Bedacht gewählt kann es schnell passieren, dass der Inhalt Ihrer Seite von Menschen mit Einschränkungen nur sehr schwer oder überhaupt nicht wahrgenommen werden kann.

Trennung von Inhalt und Design

Der erste Schritt in die richtige Richtung ist die technische Trennung von Inhalt und Design. Mit Hilfe von CSS-Stylesheets ist es möglich, vollkommen getrennt vom Inhalt Schriftvarianten, Farben, Größen, Layout etc zu definieren.

Daraus ergeben sich viele Vorteile bezüglich Barrierefreiheit:

- Flexible Anpassung auf individuelle Bedürfnisse
- Flexible Schriftgrößen und Layout
- Einfach austauschbare Darstellung

Gute Farbkontraste

Schon für Menschen ohne Behinderung ist es wichtig, gute Kontraste zwischen Vorder- und Hintergrund zu wählen, um einen Text gut lesen zu können. Noch schwieriger wird es für Menschen mit Behinderung, kontrastarme Passagen lesen zu können.

WCAG schlägt dafür folgende [Kontrastverhältnisse](#) vor:

- Mindestens **4.5:1** für Schriften kleiner als 18pt (Level AA)

- Mindestens **3:1** für Schriften grösser als 18pt (Level AA)



Abbildung 5: Schaukasten Farbkontraste

Auch beim Verwenden eines Hintergrundbildes sollten Sie auf diese Verhältnisse achten. Ebenfalls wichtig und zu bedenken ist, dass die Schrift auch bei ausgeschaltetem Hintergrund kontrastreich und somit gut sichtbar bleiben sollte.

- Snook - Farbkontrastanalyser:
http://www.snook.ca/technical/colour_contrast/colour.html

Relative Textgrößen

Vermeiden Sie absolute Textgrößen in Pixel (px). Verwenden Sie stattdessen eine relative Vermassung in em. Ein em entspricht der Breite des Buchstabens M bei normaler Schriftgrösse.

```
h1  
{  
    font-size: 2em;  
}
```

Nicht nur Textgrößen, sondern auch Breiten und Höhen von Elementen können und sollten mit em angegeben werden, damit sie sich bei einer Änderung der Textgröße in der Größe automatisch anpassen.

Navigation

Das Herzelement jeder Website ist die Menüführung um durch die Seite und ihre Inhalte navigieren zu können. Achten Sie darauf, dass die Navigation immer funktioniert, auch wenn die Seite

- ohne CSS und JavaScript

- ohne Bilder
- ohne Maus, rein über die Tastatur

angesteuert und gelesen wird.

Robuste Funktionalität

Die Menüführung sollte unter allen Umständen und immer gut und einfach bedienbar bleiben. Vor allem der Einsatz verschiedener Browser, diverser Sicherheitseinstellungen und assistiver Technologien sollte gewährleistet sein.

Auch eine Darstellung ohne Java-Script und CSS sollte die Website nicht in Ihrer Bedienbarkeit einschränken.

Abgetrennt vom Inhalt

Gerade für Menschen mit einer Sehbehinderung ist es wichtig, die Navigation eindeutig vom Inhalt abzutrennen. Diese Abtrennung erfolgt im HTML-Code üblicherweise mit dem Einsatz einer versteckten Überschrift als „Einleitung“ für den Navigationsbereich. Somit wird dem behinderten Menschen ermöglicht, klar zu erkennen, wo sich die Navigation befindet.

Beispiel für eine versteckte Überschrift

CSS:

```
.invisible  
{  
    position: absolute;  
    left: -1000px;  
    font-size: 0px;  
}
```

HTML:

```
<h2 class="invisible">Navigation</h2>
```

Aktueller Standort klar erkennbar

Ebenfalls wichtig für Menschen mit Behinderung ist die einfache Möglichkeit festzustellen, in welchem Bereich der Website er sich gerade bewegt. Für sehende Menschen erfolgt dies meist durch die farbliche Kennzeichnung des aktuellen Menüpunktes. Für jene, die diese optische Kennzeichnung nicht erfassen können, gibt es die Möglichkeit, den Navigationstext „ausgewählt“ anzuhängen, welcher unsichtbar dargestellt wird.

CSS:

```
span.inivisble  
{  
    position: absolute;  
    left: -1000px;  
    font-size: 0px;  
}
```

HTML:

```
<a href="barrierefreiheit.html">  
    Barrierefreiheit <span  
class="inivisble">(ausgewählt)</span>  
</a>
```

Noch besser und gängiger ist der Einsatz so genannter Breadcrumbs. Diese „Brotkrumen“ zeigen dem Besucher der Website jederzeit an, in welchem Bereich er sich gerade befindet. So kann für alle Besucher der Website gleichermaßen ein Mehrwert durch den gewonnenen Überblick geschaffen werden.

Sprunglinks

Viele Websites starten im Kopfbereich mit umfangreichen Informationspunkten wie Suche, Impressum, Kontakt, News... etc. Besucht nun ein Mensch die Website, der ausschließlich mit der Tastatur arbeiten kann, muss er jeden dieser Links mittels Tabulator-Taste überspringen bevor er zum relevanten Inhalt gelangt.

Abhilfe schaffen Sprunglinks. Sie werden am Anfang einer Webseite platziert und führen direkt zur Navigation oder zum Inhalt einer Webseite.

```
<div id="header">
  <a href="#inhalt" class="sprunglink">Direkt zum Inhalt</a>
  ...
</div>
...
<div id="content">
  <h1 name="inhalt">Keine Angst vor Barrierefreiheit </h1>
</div>
```

Sprunglinks können für andere Benutzer mit dem Einsatz von CSS unsichtbar gemacht werden. Ein Screenreader, der CSS nicht interpretiert, sieht die Links trotzdem und kann sie richtig interpretieren. Empfehlenswert ist, dass die Links sichtbar werden, wenn sie per Tab fokussiert werden.

```
a.sprunglink
{
  position: relative;
  left: -9999px;
}

a.sprunglink:focus
{
  position: absolute;
  left: 0px;
}
```

Accesskeys

Eine weitere Erleichterung bei der Navigation durch die Webseite kann der Einsatz von Accesskeys sein. Hierbei werden einzelne Links mit Kürzeln hinterlegt, um einen schnellen Zugriff auf wichtige Bereiche der Seite über die Tastatur zu ermöglichen.

Dazu fügt man im Header einen Link auf die entsprechende Region oder Seite ein und setzt das accesskey-Attribut.

```
<a href="#navigation" accesskey="1" title="[ALT+1]">Zur
Navigation</a>
<a href="#content" accesskey="2" title="[ALT+2]">Zum
Inhalt</a>
```

Es wird empfohlen, Accesskey ausschließlich aus Zahlen zu erstellen, um nicht Konflikte mit fix im Browser hinterlegten Accesskeys hervorzurufen.

Damit die Accesskeys auf allen Webseiten gleich sind, hat die Stiftung «Zugang für alle» eine [Empfehlung für Accesskeys](#)²⁴ herausgegeben:

0. Startseite
1. Navigation (Link innerhalb der Webseite)
2. Inhalt (Link innerhalb der Webseite)
3. Kontakt
4. Sitemap
5. Suche

Überschriften

Die richtige Verwendung von Überschriften-Tags stellt einen wichtigen Punkt im Bereich der Barrierefreiheit dar. Menschen ohne eingeschränktem Sehvermögen erkennen leicht durch Schriftgrößen und –stärken, wenn ein Textteil eine Überschrift darstellt.

Für behinderte Menschen ist diese Interpretation nur mittels unterstützender Technologien möglich, welche wiederum von technischer Seite mit den richtigen Informationen gefüttert werden müssen.

Durch den Einsatz der Überschriften-Tags <h1>-<h6> kann ein Screenreader leicht interpretieren, wenn es sich um eine Überschrift handelt, und welcher Ebene sie zuzurechnen ist.

```
<h1>Erste Überschrift der Ebene 1</h1>  
...  
<h2>Erste Überschrift der Ebene 2</h2>  
...  
<h3>Erste Überschrift der Ebene 3</h3>  
...
```

²⁴ <http://www.access-for-all.ch/ch/barrierefreiheit/barrierefreies-webdesign/tools-fuer-barrierefreies-webdesign/accesskeys.html>

Grafiken

Grafiken transportieren Emotionen, beschreiben bildlich Inhalte, lenken Aufmerksamkeit und geben einen Einblick in das beworbene Unternehmen.

In Hinblick auf Barrierefreiheit jedoch stellen Grafiken zumeist ein Problem dar. Sie können oft von sehbehinderten Menschen nicht gesehen oder erkannt werden, und Ihre Farben können nicht angepasst werden um die Sichtbarkeit zu verbessern.

Um nicht auf Grafiken verzichten zu müssen, und trotzdem alle Inhalte auch barrierefrei zu transportieren kommt bei jedem einzelnen Bild das alt-Attribut zum Einsatz. Hiermit bieten Sie beeinträchtigten Menschen eine Textalternative, die sowohl von assistiven Systemen als auch von Suchmaschinen interpretiert und verarbeitet werden kann.

```

```

Image-Maps

Sollten Sie Image-Maps nutzen, um einzelne Bereiche eines Bildes mit verschiedenen Links zu hinterlegen, kann jede einzelne <area> mit einem Alternativtext im alt-Attribut hinterlegt werden.

```


<map name="mp1234" id="mp1234">
  <area shape="circle" coords="210,47,33"
        href="start.html" alt="Zur Startseite"/>
  <area shape="circle" coords="327,87,33"
        href="kontakt.html" alt="Kontakt"/>
  <area shape="circle" coords="329,313,33"
        href="foerderungen.html" alt="Förderinfos"/>
</map>
```

Schriftgrafiken

Für Ihre Website stehen beim Einsatz von HTML4 nur wenige Schriften zur Verfügung. Diese Tatsache führte dazu, dass oft Texte in Grafiken eingefügt wurden, um das Corporate Design eines Unternehmens nicht zu missachten. Daraus ergeben sich jedoch gravierende Nachteile:

- Suchmaschinen, Screenreader und sonstige assistive Tools sind nicht in der Lage, Text in Bildern zu lesen und zu interpretieren
- Bilder können durch den Benutzer nicht in Farbe und Größe angepasst werden, was ein Lesen für manche Menschen unmöglich macht

Von Schriftgrafiken wird aus der Sicht der Barrierefreiheit daher **stark abgeraten**.

Logo als Grafik

Es gibt eine entscheidende Ausnahme beim Einsatz von Schrift in Grafiken: Ihr eigenes Firmenlogo. Dies ist die einzige Ausnahme, bei der der Wiedererkennungswert Ihrer Firma, Ihrer Website und Ihres Corporate Designs vor Barrierefreiheit gehen.

Layoutgrafiken

Gelegentlich kommen auf Webseiten Grafiken zum Einsatz, die nicht als Information sondern als Layoutgrafik dienen (zB als Abstandhalter). Gibt man hier das alt-Attribut an, lässt es aber leer, erkennen Screenreader, dass es sich nicht um eine Inhaltsgrafik handelt und überspringen diese.

Geben Sie immer ein leeres alt-Attribut an – ohne würde der Screenreader bei jeder einzelnen dieser Layoutgrafiken den Namen der Bilddatei vorlesen da er sie als Inhaltsgrafik interpretiert.

```

```

Tabellen

Tabellen sollten ausschließlich für die Darstellung tabellarischer Daten verwendet werden – und nicht zur Gestaltung eines Layouts.

Auch für Tabellen gibt es verschiedene Attribute und Tags, die es sehbehinderten Menschen erleichtern, sich in der Tabelle zurecht zu finden:

- summary-Attribut: Zusammenfassung der Struktur der Tabelle
- <caption>-Tag: Tabellen- Überschrift
- <thead>,<th>: Kopfzeilen definieren

Preise für Früchte in Europa

	Äpfel	Birnen
Russland	2.90€	1.50€
Österreich	1.70€	2.30€

```
<table summary="Tabelle mit einfacher Spalten- und
Zeilenüberschrift">
  <caption>Preise für Früchte in Europa</caption>
  <tr>
    <td></td>
    <th>Äpfel</th>
    <th>Birnen</th>
  </tr>
  <tr>
    <th>Russland</th>
    <td>2.90€</td>
    <td>1.50€</td>
  </tr>
  <tr>
    <th>Österreich</th>
    <td>1.70€</td>
    <td>2.30€</td>
  </tr>
</table>
```

Formulare

Ähnlich wie bei Tabellen gibt es für Formulare einige Tags und Attribute, die es einem sehbehinderten Benutzer erleichtern, sich zurecht zu finden:

- `<label>`: Beschriftung für ein Eingabefeld inkl. Verbindung zum Eingabefeld, zur leichteren Steuerung mit der Tastatur und richtigen Interpretation durch Screenreader

Versuchen Sie, das Label so nah wie möglich bei dem zugehörigen Eingabefeld zu platzieren. Durch die extreme Vergrößerung einer Website ist es selten möglich, beide gleichzeitig anzeigen zu lassen – durch die örtlich Nähe wird die Orientierung auf der Seite erleichtert.

```
<form action="...">
```

```
<label for="name">Name:</label>  
<input type="text" id="name" />  
</form>
```

Gruppierung von Eingabefeldern

Verschiedene Eingabefelder können zu fieldsets gruppiert werden, um die Bedienbarkeit eines Formulars weiter zu erleichtern. So können beispielsweise Vorname, Nachname, Straße und PLZ und Ort zu einer Adresse zusammengefasst werden.

In HTML stehen dafür die Tags `<fieldset>` und `<legend>` zur Verfügung.

```
<form action="...">  
  <fieldset>  
    <legend>Adresse</legend>  
    <label for="firstname">Vorname:</label><input  
type="text" id="firstname" />  
    <label for="name">Name:</label><input type="text"  
id="name" />  
  </fieldset>  
</form>
```

Rich Internet Applications

Das Web wird immer interaktiver und aus statischen Internetseiten werden dynamische Anwendungen, welche jenen auf dem Desktop immer mehr ähneln.

Da HTML nicht für diese Art Anwendungen konzipiert wurde, haben sich im Laufe der Zeit viele Technologien etabliert, welche durch Browser-Erweiterungen oder Plugins die fehlende Funktionalität ersetzen. Dies sind ein paar Beispiele dafür:

- JavaScript und AJAX
- Adobe Flex und Air
- Microsoft Silverlight
- Google Web Toolkit

Rich Internet Applications und Barrierefreiheit

Rich Internet Applications sind zwar aus der Sicht der Usability oft besser als gewöhnliche HTML Seiten, da sie unmittelbares Feedback bieten und komplexe

Interaktionen ermöglichen. Die Barrierefreiheit bleibt dadurch jedoch oft auf der Strecke. Gründe dafür sind:

- Schlechte Tastaturbedienbarkeit
- Fehlende Textalternativen für Bilder und Videos
- Schlechte Integration von assistiven Technologien wie Screenreadern oder Braillezeilen
- Schlechte Suchmaschinenoptimierung, da die Roboter den Inhalt nicht interpretieren können

Trotzdem gibt es viele Bemühungen, diese Nachteile aus der Sicht der Barrierefreiheit wett zu machen. Dazu gehören folgende Initiativen:

- ARIA-WAI Accessible Rich Internet Applications
http://www.einfach-barrierefrei.net/verstehen/standards/wai_aria.html
- Adobe Accessibility for Flash
http://help.adobe.com/de_DE/FlashPlatform/reference/actionscript/3/flash/accessibility/package-detail.html
- Microsoft Silverlight Accessibility
[http://msdn.microsoft.com/en-us/library/cc707824\(v=vs.95\).aspx](http://msdn.microsoft.com/en-us/library/cc707824(v=vs.95).aspx)

Barrierefreies AJAX

Um das lästige Nachladen der ganzen Seite zu vermeiden, versucht man bei AJAX, durch asynchrone Callbacks zum Server Teile der Webseite dynamisch nachzuladen oder zu aktualisieren.

Dies bringt jedoch einige Probleme mit sich. Beispielsweise merkt ein Screenreader nichts von Änderungen an bereits gelesenen Inhalten. Eine blinde Person wird somit nicht darüber informiert, wenn sich auf der Webseite etwas geändert hat.

Live-Regionen

Zur Lösung dieses Problems erweitert die WAI-ARIA die HTML-Syntax um sogenannte Live-Regionen. Dabei kann auf jedem HTML-Element das aria-live-Attribut gesetzt

werden, welches den assistiven Technologien mitteilt, dass diese Region sich ändern kann.

Das *aria-live*-Attribut kann auf folgende Werte gesetzt werden:

- *off* ist der Standard und sagt aus, dass kein dynamisches Nachladen erfolgt
- *polite* informiert den Benutzer dann, wenn der Screenreader im Idle-Mode ist. Eine laufende Sprachausgabe wird nicht unterbrochen
- *assertive* bedeutet, dass der Benutzer unmittelbar informiert werden muss. Eine laufende Sprachausgabe wird unterbrochen

```
<div id="news"  
  
  aria-live="polite"  
  
  aria-atomic="true"  
  
  aria-relevant="additions">Zur Zeit keine News.</div>
```

aria-atomic

Durch das *aria-atomic*-Attribut kann festgelegt werden, ob bei einer Änderung der ganze Inhalt, oder nur die Änderung vorgelesen werden soll.

- *true* bedeutet: bei einer Änderung **alles** nochmals vorlesen
- *false* bedeutet: **nur den geänderten Text** nochmals vorlesen

aria-busy

Durch das *aria-busy*-Attribut kann das Mitteilen der Änderung verzögert werden, wenn beispielsweise noch weitere Änderungen anstehen.

- *true* bedeutet: Mitteilung verzögern
- *false* bedeutet: Änderung kann jetzt mitgeteilt werden

aria-relevant

Durch das *aria-relevant*-Attribut kann festgelegt werden, welche Art von Änderungen für den Benutzer interessant sind:

- *all*: alle Änderungen sind relevant

- additions: nur hinzugefügte Elemente sind relevant
- removals: nur entfernte Elemente sind relevant

text: bei Änderung des Textinhaltes

Barrierefreie Widgets

Eine typische Eigenschaft von Rich Internet Applications sind Widgets wie Groupboxen, Sliders oder Trees, welche in HTML aus einfachen Elementen wie <div> oder zusammengebaut werden und denen dann mit JavaScript Leben eingehaucht wird.

Diese emulierten Widgets haben jedoch das Problem, dass sie voller Barrieren sind:

- Eine korrekte Tastaturnavigation ist oft nicht möglich, da HTML das tabindex-Attribut nicht für alle Elemente unterstützt.
- Assistive Technologien kennen die Bedeutung dieser Widgets nicht. Für sie sind es nur lose <div> Elemente. Dadurch ergeben sich folgende Probleme:
 - Die Rolle des Widgets ist unbekannt
 - Status kann nicht interpretiert werden
 - Aktualisierungen werden nicht erkannt

Glücklicherweise bietet der ARIA-Standard der Web Accessibility Initiative eine einfache und gute Lösung für all diese Probleme:

tabindex-Attribut für alle HTML-Elemente

Durch die ARIA-Erweiterung, kann das tabindex-Attribut auf allen Elementen gesetzt werden. In HTML4 wurden nur die Elemente a, area, button, input, select, textarea und object unterstützt.

```
<div id="rating_slider" tabindex="10">...</div>
```

Eine weitere Besonderheit ist die Verwendung von **negativen tabindex-Werten**. Wenn bei einem Element ein negativer tabindex gesetzt wird, dann ist es nicht Teil der normalen Tab-Sequenz, es kann jedoch **per JavaScript fokussiert** werden.

Rollen von Widgets

Die ARIA-Erweiterung führt das role-Attribut ein, welches den assistiven Technologien den Verwendungszweck eines Elementes beschreibt. Über weitere Eigenschafts-

Attribute können zudem der Status und die aktuellen Werte des Widgets für assistive Technologien zugänglich gemacht werden.

```
<div id="rating_slider"  
  role="slider"  
  aria-valuemin="0"  
  aria-valuemax="5"  
  aria-valuenow="4"  
  aria-valuetext="Good">  
  ...  
</div>
```

Eine Auflistung der von WAI-ARIA unterstützten Rollen ist auf folgender Website zu finden: http://www.w3.org/TR/wai-aria/roles#role_definitions

Barrierefreies Flash

Mit Adobe Flash ist es möglich, barrierefreie Inhalte zu erstellen. Die Technologie bringt alle nötigen Bausteine dafür mit. Die Entwickler müssen diese jedoch richtig umsetzen.

Flash für blinde Nutzer

Adobe Flash unterstützt Microsoft Active Accessibility (MSAA)²⁵. Das ist eine Schnittstelle, welche assistiven Technologien wie Screenreadern den Zugang auf Flash-Inhalte ermöglicht.

Sobald über MSAA ein Screenreader erkannt wird, kann per Up/Down-Taste zwischen den Inhalten einer Flash-Applikation navigiert werden.

Textinhalte und Buttons mit einem Text können so ohne weitere Anpassungen bereits von einem Screenreader vorgelesen werden. Für Bilder und Videos muss der Entwickler entsprechende Textalternativen hinterlegen. Diese können über das Accessibility-Panel gesetzt werden.

Die Reihenfolge der Elemente kann durch das Feld "Tab index" festgelegt werden. Über das Menu **View->Show Tab Order** kann die Reihenfolge als Overlay eingeblendet werden.

Flash für Personen mit Sehbehinderung

Für Personen mit einer Sehbehinderung sollte man folgende Punkte beachten:

²⁵ [https://msdn.microsoft.com/en-us/library/ms697707\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/ms697707(v=vs.85).aspx)

- Die Schrittgröße anpassbar machen
- Farben mit ausreichend Kontrast zueinander wählen
- Informationen nie ausschließlich über die Farbe codieren

Tastaturnavigation

Die Anwendung sollte rein mit der Tastatur bedienbar sein. Mit der TAB-Taste kann man durch die Elemente navigieren. Um die Tab-Reihenfolge festzulegen, kann man den "Tab Index" im Accessibility-Panel setzen. Um ein Element aus der Navigation auszuschließen kann man per ActionScript `tabEnabled` auf `false` setzen:

```
myImage.tabEnabled = false;
```

Eingabeelemente wie ListBox oder ComboBox unterstützen die Tastaturnavigation. Sie muss allerdings per ActionScript mittels folgendem Befehl aktiviert werden:

```
comboBox.enableAccessibility();
```

Flash für Personen mit Hörbehinderung

Hörbehinderte Personen haben hauptsächlich Probleme beim Verstehen von Videos, bei denen gesprochen wird. Die gesprochenen Informationen können in Flash über eine zeitbasierte XML-Datei (DFXP) hinterlegt und über den Video eingeblendet werden.

Weiterführende Informationen:

WebAIM hat eine Übersicht mit kleinen Anleitungen erstellt:

<http://webaim.org/techniques/flash/>

Adobe stellt auf seiner Seite ein eigenes Tutorial zur Verfügung:

<http://www.adobe.com/accessibility/products/flash/tutorial.html>

Barrierefreies Silverlight

Microsoft Silverlight ist eine plugin basierte Technologie, welche es erlaubt plattformübergreifende .NET Anwendungen zu erstellen, welche in eine HTML-Seite eingebettet werden können.

Plugin basierte Technologien sind aus Sicht der Barrierefreiheit immer kritisch, da assistive Technologien das proprietäre Format verstehen müssen, um auf die Inhalte zugreifen zu können. Microsoft bietet mit Silverlight jedoch über die UI-Automation

Schnittstelle eine standardisierte Schnittstelle, über welche auf die Inhalte der Anwendung zugegriffen werden können.

Silverlight für Blinde

Blinde Personen nutzen einen Screenreader, der ihnen den Inhalt des Bildschirms vorliest. Silverlight unterstützt Screenreader über die UI-Automation Schnittstelle. Standardmäßig werden so alle Texte und Elemente mit einem Namen vorgelesen, wenn der Fokus darauf gesetzt wird.

Beispiel für UI-Automation in Silverlight:

```
<Button AutomationProperties.Name="Play video"
        AutomationProperties.HelpText="Plays the
introduction video"
        AutomationProperties.AcceleratorKey="P" >
    ...
</Button>
```

Silverlight für Sehbehinderte Personen

Für Personen mit einer Sehbehinderung muss bei der Entwicklung auf folgendes geachtet werden:

- Schrittgröße, oder besser die komplette Oberfläche vergrößerbar machen
- Ausreichende Kontraste anbieten
- Nie eine Information rein über die Farbe kodieren
- Tastaturnavigation sicherstellen

In Silverlight kann festgestellt werden, ob der Benutzer ein High-Contrast Farbschema gewählt hat und dementsprechend die Farben in der Silverlight Applikation anpassen.

```
if (SystemParameters.HighContrast)
{
    // Use a high-contrast color scheme
}
```

Weiterführende Informationen unter:

Microsoft Developer Network: <https://msdn.microsoft.com/enable>

Anleitung zu barrierefreien RIAs in Silverlight:

<https://channel9.msdn.com/Events/MIX/MIX09/T65M>

Umsetzung bei PDF

Nicht vergessen werden dürfen Dokumente in Form von Portable-Document-Format-Dateien (PDF) oder Word-Dateien, da sich beispielsweise PDF als Standard etabliert hat, um Formulare oder Dokumente über das Internet zu teilen. Hierbei wird jedoch, neben einer digitalen Signatur des Dokumentes, auf den Umstand vergessen, dass auch diese barrierefrei zur Verfügung gestellt werden sollten, da sie andernfalls nicht wahrgenommen werden können. Speziell Produktbeschreibungen in Online-Shops die als PDF zum Download zur Verfügung stehen sollten hierbei barrierefrei erstellt worden sein. Um PDFs barrierefrei zu gestalten gibt es sogenannte Tags. Tags sind Strukturinformationen die in das PDF Dokument eingebettet werden können. Eingescannte Bilder im PDF verfügen weder über Tags noch über eine andere Möglichkeit der unterstützten Ausgabe. Es empfiehlt sich daher auch, Dokumente nicht einfach einzuscannen und dies dann als PDF zur Verfügung zu stellen, da der PDF-Container in diesem Fall lediglich ein unbeschriebenes Bild anzeigt. Um Tags nachträglich in ein PDF einzufügen ist der Einsatz von Adobe Acrobat Pro erforderlich.

PDFs treten daher in folgenden Varianten in Erscheinung:

1. Ohne Tags und ohne maschinenlesbaren Text. (z.B. Bilder, Grafiken, gescannte Elemente)
2. Ohne Tags aber mit maschinenlesbarem Text (entspricht der Standarderzeugung von PDFs)
3. Mit automatisch hinzugefügten Tags
4. Mit Tags, die überprüft und angepasst wurden. Dies erfolgt zumeist im nachhinein.

Diese vier Varianten entsprechen dem Setzen von Tags bzw. ihrem Erzeugungszeitpunkt. Für das automatische setzen von Tags ist es beispielsweise unter Word erforderlich, Formatvorlagen zu nutzen oder bei Grafiken auch alternative Texte

zu hinterlegen, andernfalls gibt es keine Anhaltspunkte anhand derer zum Zeitpunkt der PDF Generierung vom Generator ein Tag gesetzt werden kann. Wer jemals ein ausfüllbares PDF-Dokument zu erstellen versuchte kennt hierbei beispielsweise den Umstand, dass gesetzte Formularfelder nach der Erzeugung des PDFs doch nicht digital ausfüllbar waren, sondern erst entsprechende Felder im Originaldokument eingefügt werden mussten, die in der Folge als digital ausfüllbares Feld übernommen werden konnten. Hier fehlten eben die Tags, also die Strukturinformationen. Zu beachten gilt bei PDF-Dokumenten gilt, dass bei getagten PDFs die Inhalte zweifach gespeichert werden, einmal mit und einmal ohne Tags. Dies ist für den Normalverbraucher so nicht ersichtlich. Diese doppelten Inhalte bedeuten vereinfacht formuliert, dass das PDF in zwei Ansichten zur Verfügung steht. Einmal mit und einmal ohne Tag. Einerseits wirkt sich dies in der Folge auf das Tool aus, mit dem das PDF geöffnet wird. Dieses muss in der Lage sein, die Tags auszulesen. Dies ist in fast allen gängigen PDF-Viewern (Anzeigesoftware für PDFs) möglich. Andererseits hat dies zur Folge, dass Änderungen an der einen „Anzeigedarstellungen“ nicht automatisch auf die andere übernommen werden, wenn das PDF nachträglich verändert wird. Es empfiehlt sich daher Änderungen im Originaldokument vorzunehmen und in der Folge ein neues zu generieren. Weiterführende Informationen rund um PDF, das PDF-Dateiformat etc. würden zwar helfen, diesen Umstand besser zu verstehen, sprengen jedoch den Rahmen dieses Dokumentes. Aus diesem Grund wird auf die Website <http://www.barrierefreies-webdesign.de> verwiesen. Jan Eric Hellbusch, ein Sachverständiger aus Deutschland und Autor des Buches „Barrierefreiheit verstehen und umsetzen: Webstandards für ein zugängliches nutzbares Internet“ hat auf seiner Website eine technisch fundierte Information aus seinem Buch zur Verfügung gestellt (<http://www.barrierefreies-webdesign.de/knowhow/rich-media/pdf.html>).

Überprüfen von PDF-Dokumenten

Für eine Überprüfung stehen drei wesentliche Hilfsmittel zur Verfügung. Unter Windows steht der PDF-Accessibility-Checker (PAC) zur Verfügung. Dies ist eine Gratissoftware und kann auf der Webseite von Zugang für alle heruntergeladen werden. <http://www.access-for-all.ch/ch/pdf-werkstatt/pac-pdf-accessibility-checker.html>

Alternativ kann die Prüfung durch Adobe Acrobat Professional erfolgen, setzt jedoch den Kauf von Adobe Acrobat Pro voraus. Eine Anleitung findet sich hier: <http://wac.osu.edu/pdf/checking/checkingyourpdf.html>

Eine weitere Möglichkeit besteht noch darin, den Inhalt aus einem PDF-Dokument per copy-paste in einen Texteditor zu kopieren und sodann die Reihenfolge zu überprüfen.

Umsetzung bei Word

Microsoft Word ist eines der am meisten genutzten Textverarbeitungsprogramme und daher werden auch vielfach .doc oder .docx Dateien über das Internet (Website oder Mails) an Verbraucher übermittelt. Nun bietet Word grundsätzlich sämtliche Funktionen für barrierefreie Dokumente und der Text ist zumeist auch maschinenlesbar. Es ist jedoch gleich wie bei PDF-Dokumenten erforderlich, Tags einzufügen. Im Gegensatz zu PDF sind Tags, also Strukturinformationen, in Word jedoch etwas mit dem ein versierter Nutzer von Word ohnehin arbeitet. Die Formatierungspaletten von Word mit Überschriften, Fußnoten etc. stellen derartige Strukturinformationen bereit und können in der Folge auch als getagtes PDF oder strukturiertes HTML ausgegeben werden.

Eine ausführliche Behandlung dieser Themen wäre mit einer Word-Schulung vergleichbar. Die Organisation WebAIM hat hierfür eine umfassende Anleitung veröffentlicht. Diese steht unter <http://webaim.org/techniques/word/> zur Verfügung und erklärt ausführliche sämtliche Elemente aus Word und was zu beachten ist.

Werden OpenSource bzw. Free-Software Programme wie OpenOffice oder LibreOffice zum Einsatz gebracht, so gilt auch für diese, dass der Einsatz von Tags, also den Formatierungs-Elementen wie Überschrift, Standardtext, Listenelement etc., die Voraussetzung für ein barrierefreies Dokument darstellt. Diese Programme sind ebenso wie Word in der Lage barrierefreie weil getagte PDF-Dokumente zu generieren.

Weiters gibt es auch auf Deutsch eine Umfassende Anleitung von Microsoft unter:

<https://support.office.com/de-at/article/Erstellen-von-barrierefreien-Word-Dokumenten-d9bf3683-87ac-47ea-b91a-78dcacb3c66d>

Literatur aus Kapitel 3

How To Meet WCAG 2.0 (englisch): Eine Art Kurzreferenz für WCAG 2.0 - <http://www.w3.org/WAI/WCAG20/quickref/>

Understanding WCAG 2.0 (englisch): Zusätzliche Details und Hintergrundwissen zu den Richtlinien und Erfolgskriterien) - <http://www.w3.org/TR/UNDERSTANDING-WCAG20/>

Techniques for WCAG 2.0 (englisch): Allgemeine und technologiespezifische Beispiele) - <http://www.w3.org/TR/WCAG20-TECHS/>

<http://www.einfach-barrierefrei.net>

<http://www.w3.org>

<http://www.access-for-all.ch>

HTML-Validator: <http://validator.w3.org/>

Total-Validator: <http://www.totalvalidator.com/>

WAVE: <http://wave.webaim.org>

Compliance-Check: <http://www.cynthiasays.com>

Weiterführende Informationen auch auf: <http://www.barrierefreies-webdesign.de>

Hellbusch (März 2011) Barrierefreiheit verstehen und umsetzen: Webstandards für ein zugängliches und nutzbares Internet von Jan Eric Hellbusch (Autor), Kerstin Probiesch (Autor)