

## Checkliste bezüglich Barrierefreiheit im WWW

Checkpunkte	Umgesetzt?
<b>Allgemein</b>	
Eine einzige barrierefreie Version für alle statt alternativer Nur-Text-Seite	
Vorgegebene Reihenfolge für die nachträgliche Anpassung beachten	
HTML gemäß seines ursprünglichen Zweckes einsetzen – Inhalt statt Darstellung	
<b>1. Planung und Gestaltung</b>	
Trennung von Inhalt und Layout durch Formatierung anhand von Cascading Style Sheets (CSS)	
Steuerung auch mittels der Tastatur statt der Maus	
Durchgängig einheitliches Layout	
Bereitstellung eines Inhaltsverzeichnisses	
Vermeidung von sich automatisch öffnenden Fenster, automatischen Weiterleitungen und Aktualisierungen	
<i>Layout</i>	
Anhand von Frames vermeiden	
Anhand von Layouttabellen vermeiden	
Anhand von CSS durch das <DIV>-Element	
<b>2. HTML</b>	
<b>2.1 Frames</b>	
Wenn möglich, auf sie verzichten	
Struktur so einfach wie möglich halten	
Sinnvolle Frame-Titel über das „TITLE“-Attribut definieren	
Korrekte <NOFRAMES>-Programmierung realisieren	
<b>2.2 Tabellen</b>	
Größenangaben in relativen statt absoluten Maßeinheiten	
<i>Datentabellen</i>	
Einsatz von strukturierenden Elementen bzw. Attributen wie <TH>, <CAPTION> und „SUMMARY“	
<i>Layouttabellen</i>	
Linearität der Tabelle gewährleisten	
Kein Einsatz von strukturierenden Elementen bzw. Attributen wie <TH>, <CAPTION> und „SUMMARY“	
Möglichst kompletter Verzicht auf Layouttabellen; besser: Layouterstellung anhand von des <DIV>-Elementes und CSS	
<b>2.3 Farben</b>	
Hoher Kontrast zwischen Vorder- und Hintergrund	
Keine Kombinationen von Rot und Grün	
Kein Informationsverlust in der inversen Farbdarstellung	
Informationen auch ganz ohne Farbdarstellung vermittelbar	

Checkpunkte	Umgesetzt?
<b>2.4 Grafiken und Multimedia</b>	
Vor Einsatz deren Zweckmäßigkeit prüfen	
Keine transparenten Bildhintergründe	
Angabe beschreibender Alternativtexte über das „ALT“-Attribut oder einen D-Link	
Bei transparenten Grafiken und Bildern mit rein dekorativem Charakter: leeren Alternativtext (ALT=“ “)	
Bei Grafiken anstelle von HTML-Listenpunkten: Alternativtext entsprechend der Darstellung der Grafik (ALT=“-“ oder ALT=“**“)	
<i>Diagramme</i>	
Regulären beschreibenden Alternativtext	
Zu Grunde liegende Daten in Form einer Datentabelle angeben	
Gesamtaussage des Diagramms in Form eines ausführlichen Textes über einen D-Link anbieten	
<i>Buttons</i>	
Alternativtext mit Verweis auf das Ziel des Links anbieten	
Mindestgröße beachten	
Alternative Bedienbarkeit auch per Tastatur durch Verwendung des „TABINDEX“-Attributes ermöglichen	
<b>2.5 Imagemaps</b>	
Clientseitige statt serverseitige Imagemaps verwenden	
Bild und Links aussagekräftig benennen – inkl. Anzahl und Ziele	
Linkbereiche ausreichend groß definieren	
Wenn möglich vermeiden	
Alternative Textseite ohne Imagemap anbieten	
<b>2.6 Animationen</b>	
Nach Möglichkeit auf Animationen verzichten	
Textalternativen für alle Formen von Animationen	
Niedrige Durchlaufgeschwindigkeiten	
Keine raschen Hell-Dunkel-Wechsel	
Möglichkeit des Abschaltens oder Anhaltens aller Animationen	
<b>2.7 Textaufbau</b>	
Texte kurz und übersichtlich gestalten	
Trennung von Inhalt und Layout	
Text nie als Grafik einbinden	
Inhaltsverzeichnis am Dokumentenanfang	
Linksbündige Textausrichtung	
HTML-Elemente nur ihrem ursprünglichen Sinn entsprechend verwenden	

Checkpunkte	Umgesetzt?
<i>Überschriften</i>	
Vorgegebene Hierarchie beachten	
Mit kennzeichnendem Merkmal beginnen	
Mit Satzzeichen beenden	
<b>2.8 Sprache</b>	
Klare, allgemein verständliche Sprache ohne Fachbegriffe	
Abkürzungen vermeiden bzw. anhand der HTML-Elemente <ABBR> und <ACRONYM> ausschreiben	
Dokumentensprache angeben anhand des „LANG“-Attributes	
Sprachwechsel kennzeichnen	
Tipfehler u. ä. durch entsprechende Prüfungen vor einer Veröffentlichung entfernen	
<b>2.9 Verlinkungen</b>	
Alternativ zur Maus auch mit der Tastatur bedienbar	
Linklisten untereinander statt nebeneinander	
Nebeneinander stehende Links durch mindestens ein nichtverlinktes Zeichen trennen	
Text des Linkes auch ohne unmittelbaren Kontext verständlich	
Angemessene Größe des anklickbaren Bereiches	
<b>2.10 Schriften</b>	
Ausreichend große Schriftgrößen	
Relative Größenangaben	
Formatierungen nur anhand von CSS	
Serifenlose Schriftarten	
Keine typografischen Änderungen an den Schriften wie Sperrungen oder Unterschneidungen	
Nicht zu viele Schriftarten gleichzeitig verwenden	
<b>2.11 Formulare</b>	
Klare Bezeichnung der Formularfelder mittels <LABEL>-Element	
Vorbelegung der Formularfelder durch Verwendung des „VALUE“-Attributes	
Bedienbarkeit aller Formularfelder auch über die Tastatur anhand des Einsatzes der HTML-Attribute „ACCESSKEY“ und „TABINDEX“	
Sinnvolle Gruppierung von Informationsblöcken durch Anwendung des „FIELDSET“- und des „LEGEND“-Elementes	
Überprüfung aller Eingaben auf Rechtschreibung	

Checkpunkte	Umgesetzt?
<b>3. Cascading Style Sheets</b>	
Einbindung der externen CSS-Angaben über das <LINK>-Element	
Unterschiedliche Stylesheets für verschiedene Endgeräte	
Sicherstellung des Weiterfunktionierens ganz ohne Formatvorlagen	
Größenangaben in relativen statt absoluten Werten	
<b>4. JavaScripts und Applets</b>	
Alternativen anhand von HTML schaffen	
„MouseOver“-Aktionen nicht zusätzlich mit Alternativen belegen	
Auf Eventhandler „OnClick“ verzichten	
Bei Deaktivierung sicherstellen, dass die Navigation und der Informationswert erhalten bleiben und weiter funktionieren	
Navigierbarkeit unabhängig vom Eingabegerät sicherstellen	
Auf den Einsatz neuester Technologien verzichten	
Wenn nicht in einer einzigen Version umsetzbar, alternative HTML-Seite anbieten	
<b>5. Flash</b>	
Einsatz auf Zweckmäßigkeit prüfen	
Grafiken mit Alternativtexten versehen über eine Betitelung im Feld „Name“ oder eine Beschreibung im Feld „Beschreibung“	
Klare, kontrastreiche Farben	
Keine Bewegungen/Animationen für wichtige Informationen	
Zusammengehörende Objekte nah beieinander platzieren	
Einhaltung von Standardvorgaben	
<b>6. Portable Document Format-Dokumente</b>	
Nach Möglichkeit vermeiden	
Bei einem Einsatz: Inhalte auch in alternativen, zugänglichen Dateiformaten anbieten	