

Kleine Tipps – Große Wirkung: Barrierefreies E-Learning Arbeitspapier

Eine Investition in Wissen bringt noch immer die besten Zinsen.

(Benjamin Franklin, 17. 01. 1706 – 14. 04. 1790, Naturforscher, Ökonom und US-Staatsmann)

Zentraler Informatikdienst
Neue Medien und Lerntechnologien
DI (FH) Ortrun Gröbinger

Inhaltsverzeichnis

Einleitung	2
Empfehlungen zur Gestaltung von Word-Dokumenten	2
Alternativtexte in Word einpflegen	3
Strukturhinweise	3
Nicht vergessen bei der Strukturierung von Word-Dokumenten	4
Empfehlungen zur Gestaltung von Word-Dokumenten	4
Versionen	4
Umwandeln von Word zu PDF	5
Word und PDFMaker 5	7
Word und PDFMaker 7	7
Einstellungen in Acrobat	8
Aussagekräftige Dateinamen und Metadaten	8
Fazit PDF	9
Empfehlungen für den Webbereich	10
Verständliche Dokumentationsgestaltung	10
Klare Navigationsmechanismen	10
Natürliche Sprache	13
Tabellen	14
Alternativen zu Audio und visuellen Inhalten	15
Farbe	17
Markup	18
Stylesheets	20
Benutzerkontrolle über zeitgesteuerte Abläufe	21
Technologien und Richtlinien von W3C	21
Schluss	21
Abbildungsverzeichnis	21
Quellen	22

Einleitung

Was sind Barrieren im Internet? Diese Frage lässt sich nicht direkt beantworten, da Barrieren für jeden etwas anderes bedeuten. Müssen in erster Linie die Bedürfnisse von Menschen mit Behinderungen wie Sehschwächen, Hörschäden, eingeschränkter Beweglichkeit usw. berücksichtigt werden, gibt es eine breite Benutzergruppe, die z.B. nur über eine langsame Internetanbindung oder gering auflösende Bildschirme verfügt. Man denke weiters an den aktuellen Trend Mobile Endgeräte auch im Lernbereich einzusetzen. Die Individualisierung der Lernumgebung und Lernmaterialien werden mittlerweile als ein wichtiger Aspekt für gutes Lernen angesehen. Das führt zu sehr hohen Ansprüchen an die Materialentwicklung, und zwar sowohl auf inhaltlicher, als auch auf technischer Ebene.

Als Materialproduzent bzw. Materialproduzentin befinden Sie sich heutzutage in einer Zwickmühle. Einerseits gibt es eine geradezu gigantische Auswahl an Möglichkeiten um Inhalte am Computer, insbesondere für das Internet, bunt, vertont, mit Animationen, Grafiken, Videos, Textgestaltungen usw. lernfördernd, motivierend und künstlerisch aufzuarbeiten. Andererseits müssen bestimmte Richtlinien beachtet werden um Barrierefreiheit zu wahren, nicht zuletzt aufgrund des E-Government-Gesetzes welches mit 1. Jänner 2008 in Österreich in Kraft treten wird. In diesem Gesetz ist unter anderem verankert, dass alle Behördlichen und Öffentlichen Seiten, für alle Menschen zugänglich sein müssen. In diese Definition fallen sowohl Ämter, als z.B. auch Universitäten und bis zu einem gewissen Grad auch die dort angebotenen Lernmaterialien.

Wie barrierearm sich eine Universität präsentieren möchte, welchen Standard sie für sich realisieren möchte, ist eine Profilerscheidung. Die Universität Innsbruck ist daran interessiert, möglichst barrierearm aufzutreten. Aus diesem Grund wurde nach Richtlinien, Vereinbarungen und Kompromissen gesucht, die sich auf den Bereich der Neuen Medien wie folgt auswirken:

Werden Word oder PDF Dokumente erstellt, wird dringend darauf hingewiesen, sich an bestimmte Einstellungen, in Folge in diesem Dokument behandelt, zu halten. Für den Webbereich wurden zehn Bereiche ausgewählt, zu welchen Richtlinien ausgearbeitet wurden und welche im Zusammenhang mit geförderten eLearning-Projekten verpflichtend sind.

Diese Richtlinien decken in erster Linie die Basisvoraussetzungen für Barrierefreiheit ab und sind bei weitem nicht vollständig – die vollständigen Richtlinien entnehmen Sie bitte z. B. den W3C-Richtlinien. Eine selbstständige, weitere Verbesserung der Projektmitarbeiter am Produkt ist gewünscht.

Empfehlungen zur Gestaltung von Word-Dokumenten

Microsoft Word ist eines der am weitesten verbreiteten Tools zur Aufbereitung von Texten. Word stellt auch Möglichkeiten für Interaktivität bereit, indem multimediale Ressourcen hinzugefügt oder Möglichkeiten angeboten werden, z. B. Formulare auszufüllen. Zusätzliche Funktionalitäten werden mittels Makros programmiert.

Sind bei Grafiken oder hinzugefügten multimedialen Inhalten keine Alternativdarstellungen geboten, so kann dies für Menschen mit Sehbehinderungen und ScreenReader-Anwender

problematisch werden. Daher ist es hier besonders wichtig, darauf zu achten, dass Bilder und Grafiken mit Alternativtexten ausgestattet werden.

Alternativtexte in Word einpflegen

Word bietet die Möglichkeit, Alternativtexte bei Grafiken zu hinterlegen. Diese Texte werden von Endgeräten, die keine Grafiken darstellen, ausgegeben. Gehen Sie dabei wie folgt vor:

- Grafik in Dokument wie gewohnt einfügen.
- Mit der rechten Maustaste auf die Grafik klicken.
- Das Dialogfeld „Grafik formatieren“ aufrufen.
- Im Reiter (Tab-Sheet) „WEB“ den Alternativtext angeben.

Logische Blöcke einer Seite können bspw. sein:

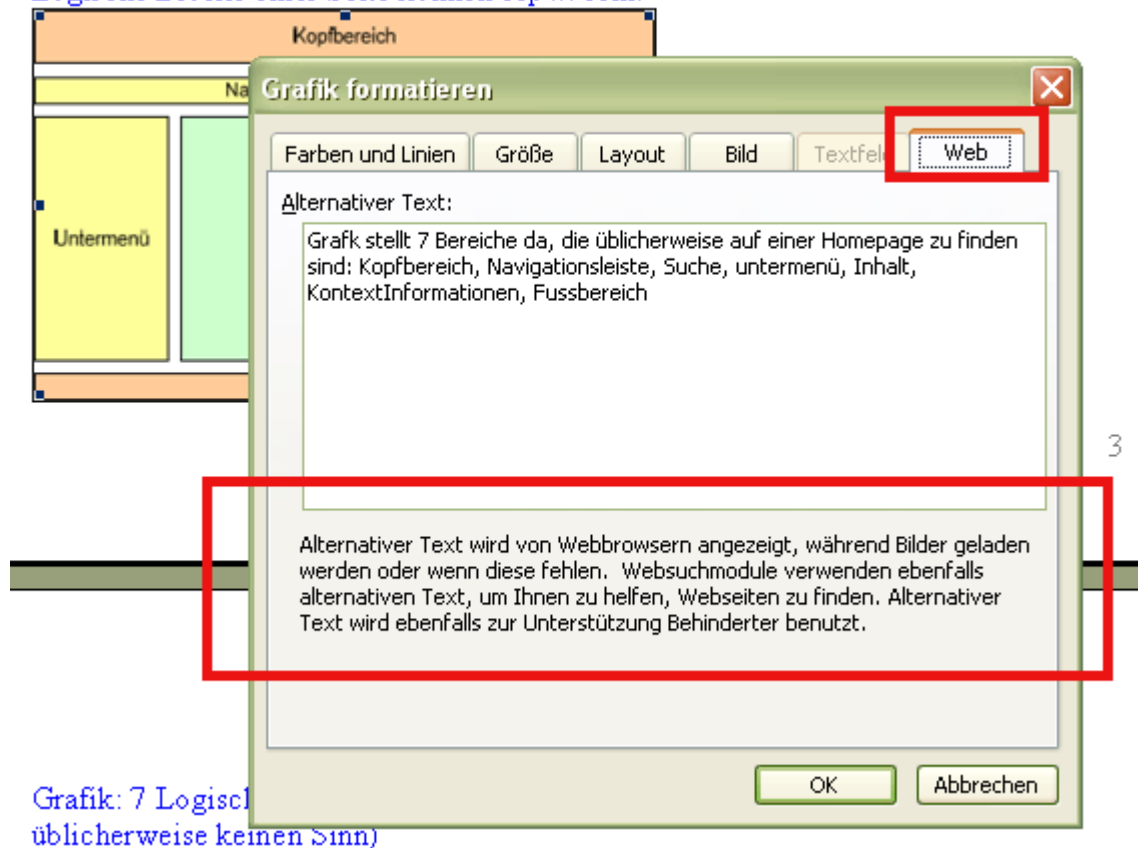


Abbildung 1: Erklärende Grafik, wie Alternativtexte in Word richtig vergeben werden können.

Strukturhinweise

Einheitliche Strukturen in der Formatierung bringen sowohl sehenden als auch sehbehinderten Menschen große Vorteile, da richtig formatierte Dokumente einen schnelleren Überblick bieten. Am einfachsten ist es, auf bereits vorhandene Formatvorlagen zurückzugreifen, Sie können aber auch eigene Formatvorlagen entwickeln.

Besonders wichtig sind Überschriftformate. Diese Überschriftformate helfen außerdem beim automatischen Erstellen eines Inhaltsverzeichnisses und somit einer einfachen Navigation im Dokument.

Vermeiden Sie viele Zeilenschaltungen, da diese Screenreaderbenutzer verwirren können. Wenn Sie größere Abstände zwischen den einzelnen Zeilen oder Absätzen wünschen, definieren Sie dies über eine entsprechende Formatvorlage.

Kennzeichnen Sie Tabellen und Spalten.

Vermeiden Sie HTML-Konvertierung von Word-Dokumenten. Word produziert sehr viel unnötigen Code, der für ScreenReader-Anwender und Anwender von Textbrowsern ausgesprochen mühsam ist. Eine nachträgliche Überarbeitung des Codes um Reparaturen daran vorzunehmen ist fast immer zeitlich unrentabel. Die Problematik wird noch gesteigert, wenn die Texte nicht mit Alternativtexten versehen werden und keine Strukturbeschreibungen beinhalten. Sollten Sie trotzdem Word-Dokumente in HTML umwandeln müssen, verwenden Sie Programme, die Word-generierten HTML-Code „Filtern“ und in einen standardkonformen optimierten Code umwandeln.

Nicht vergessen bei der Strukturierung von Word-Dokumenten

- Versehen Sie Grafiken mit Alternativtexten.
- Verwenden Sie Formatvorlagen, insbesondere für Überschriften und Absatzgestaltung.
- Beschriften Sie Tabellen und Spalten
- Beschriften Sie Grafiken.
- Erstellen Sie Inhalts-, Abbildungs- und Quellenverzeichnisse

Empfehlungen zur Gestaltung von Word-Dokumenten

PDF ist ein weit verbreitetes proprietäres (d.h. software- und herstellerabhängiges) Format, das von Adobe entwickelt wurde. Ursprünglich war das Format dazu gedacht, gelayoutete, für den Druck bestimmte, Inhalte auch im Web zugänglich zu machen. Da sich PDF aber sehr rasch auch dahingehend etablieren konnte, Informationen so abzuspeichern, dass sie Plattformunabhängig zugänglich wurden und die Texte nicht ohne weiteres verändert werden konnten, muss dem Format auch hinsichtlich Barrierefreiheit Aufmerksamkeit geschenkt werden.

PDF-Dokumente können mit freien Acrobat Readern gelesen werden. Es gibt Acrobat Reader für Windows und Mac, Open Source PDF-lesende Applikationen sowie für UNIX/Linux Betriebssysteme.

Versionen

PDFs die mit älteren Acrobat-Versionen als Acrobat 5.0 erstellt wurden bereiten insbesondere ScreenReader-Anwendern bezüglich Barrierefreiheit große Probleme. Verwenden Sie daher bitte wenn irgendwie möglich Acrobat Version 6.0 oder höher. Ab Version Acrobat 6.0 sind die meisten PDF-Dokumente mit ScreenReadern lesbar, da es sich um „tagged“ (ausgezeichnete) PDFs handelt. Dokumente, die mit der Version 5.0 erstellt wurden, müssen in „tagged PDFs“ konvertiert werden. Dazu ist das Acrobat 5.0 Make Accessible Plug-in erforderlich. Zum Erstellen von PDFs aus gescannten Dokumenten hat Adobe Capture 3.0 Funktionen, die Barrierefreiheit unterstütze

In der Acrobat-Version 7.0 hat die Firma Adobe sehr viele Verbesserungen für die barrierefreie Nutzung von PDFs getan. Diese Version unterstützt viele Voreinstellungen

bereits automatisch. So integriert Acrobat 7.0 Professional z.B. einen Barrierefreiheits-Checker zur „Zugriffsprüfung“.

Acrobat 7.0 für Windows verwendet Microsoft Active Accessibility (MSAA). Mithilfe dieser API (Application Programming Interface) können Windows-basierte Anwendungen Informationen zum Inhalt und zur Benutzeroberfläche an Computer-Hilfsmittel übertragen und gesprochen oder als Braille-Text ausgegeben werden (vgl. [4]). MSAA ist ein weit verbreiteter Standard für Barrierefreiheit.

Umwandeln von Word zu PDF

Neben den im Abschnitt „Empfehlungen zur Gestaltung von Word-Dokumenten“ beschriebenen Einstellungen müssen folgende Hinweise beachtet werden.

1. Grafiken müssen als „Mit Text in Zeile“ formatiert sein.

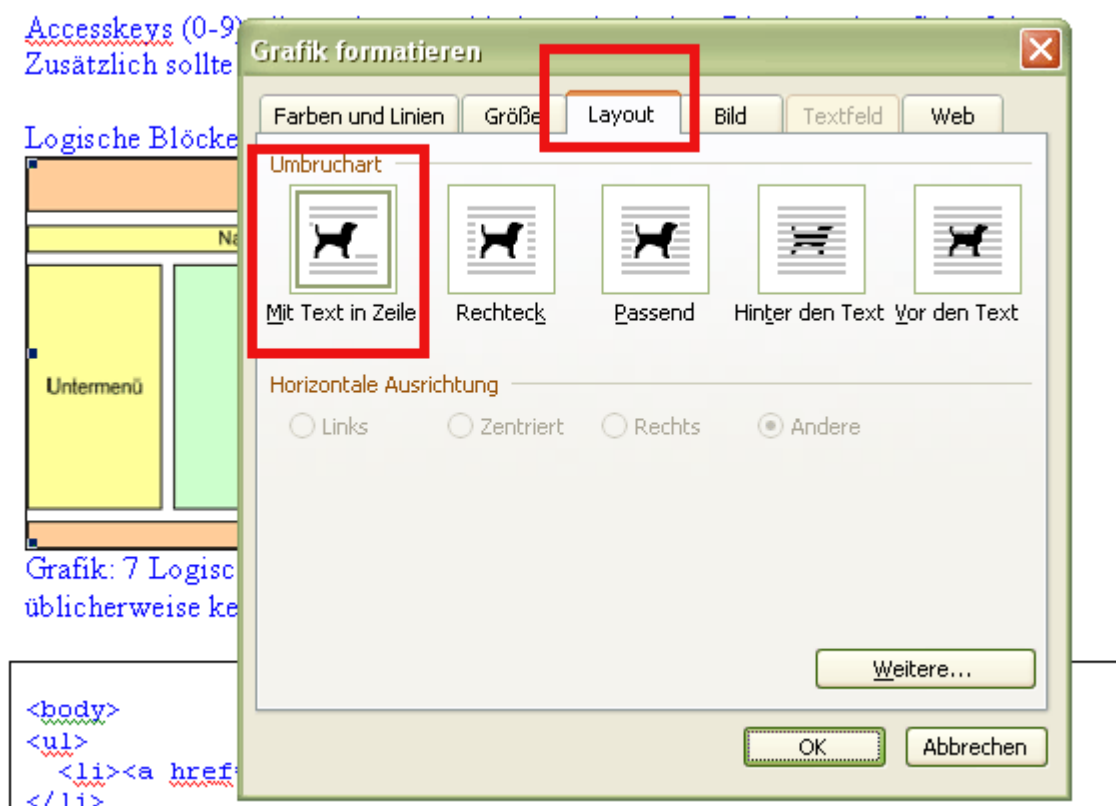


Abbildung 2: Grafik beschreibt, wie Bilder in Word "Mit Text in Zeile" formatiert werden, um später korrekt in barrierefreien PDFs angezeigt zu werden.

- In Grafiken und Textfeldern dürfen keine Füllfarben verwendet werden. Vermeiden Sie Hintergrundflächen.

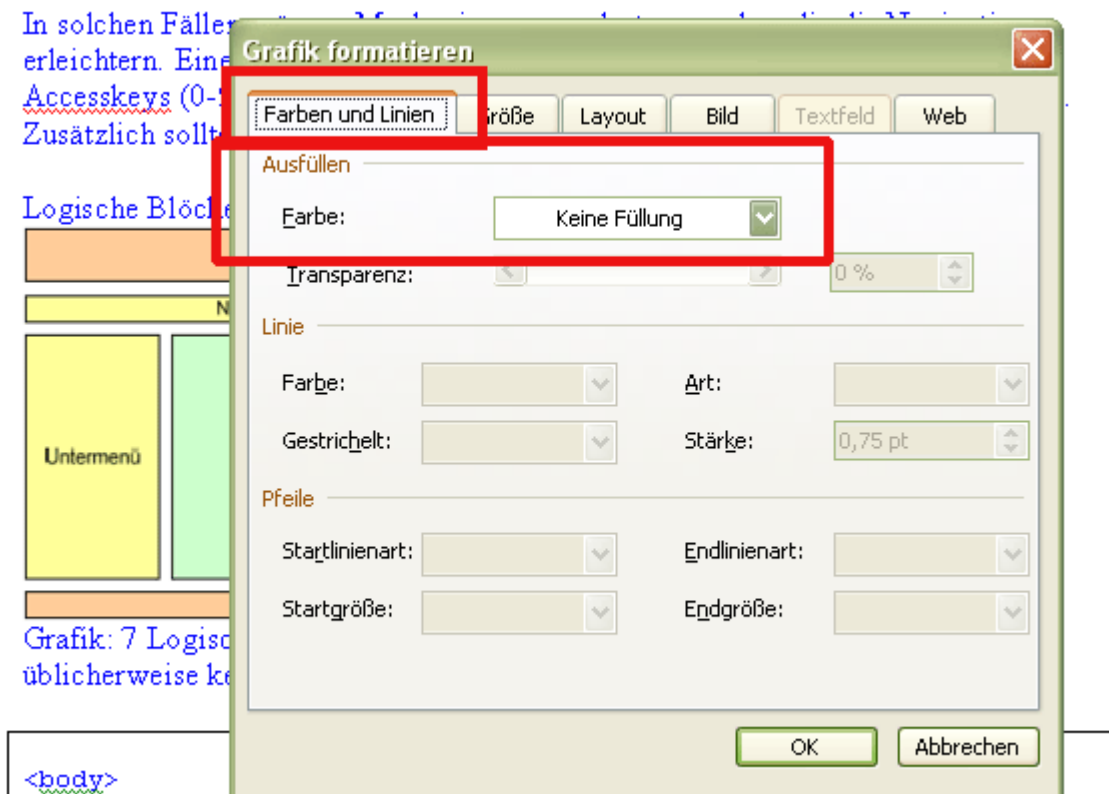


Abbildung 3: Wenn Bilder korrekt in tagged PDFs übernommen werden sollen, dürfen sie keine Füllungen haben.

- Ab Word 2000 ist die Übertragung von Strukturinformationen aus einem Word-Dokument in ein strukturiertes PDF-Dokument möglich. Bei gut strukturierten Word-Dokumenten werden diese bei der Konvertierung mit einem Lesezeichenbaum als interaktives Inhaltsverzeichnis erzeugt.

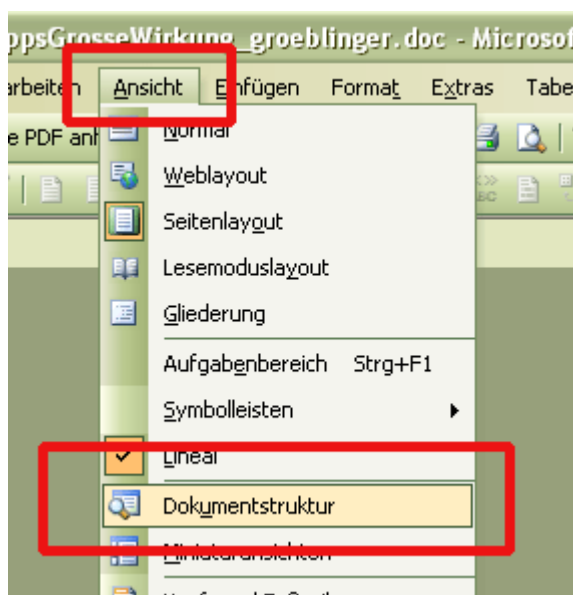


Abbildung 4: Menü in Word - Anzeigen von Lesezeichen

Lesezeichen kann man sich in Word anzeigen lassen, indem man in den Menüpunkt „Ansicht“ die „Dokumentenstruktur“ aktiviert.
Inhalte von Kopf- und Fußzeilen werden von Acrobat nicht in die PDF-Tag-Struktur übernommen.

4. Die Sprache für das Dokument kann im Menü „Extras“ beim Punkt „Sprache“ eingestellt werden. Der PDFMaker überträgt diese Information jedoch noch nicht ins PDF. In diesem Fall muss die Sprache bei Acrobat unter Menü „Datei“, „Dokumenteneigenschaften“ unter „Erweitert“ – „Leseoptionen“ bei „Sprache“ oder „Dokumentensprache“ eingestellt werden.
5. Starten Sie Ihr PDF-Konvertierungsprogramm. Je nachdem, mit welcher Software Sie arbeiten, müssen Sie unterschiedliche Einstellungen vornehmen:

Word und PDFMaker 5

- Dokument in Word öffnen
- Im Menü Acrobat den Menüpunkt „Konvertierungseigenschaften ändern“ auswählen.
- In der Registerkarte „Einstellungen“ wählen Sie E-Book aus. Die Schaltfläche „Konvertierungseinstellungen bearbeiten“
- Im neuen Fenster in der Registerkarte „Allgemein“ bei „Kompatibilität“ den Eintrag „Acrobat 5.0 (PDF 1.4)“ wählen. – OK
- Registerkarte „Office“ auswählen und überprüfen, ob „Tags in PDF einbetten“ aktiviert ist.
- Überprüfen, ob für alle Textfelder die „Ausfüllfarbe“ auf „keine“ gestellt ist.
- Registerkarte „Lesezeichen“ auswählen und festlegen, für welche Elemente Lesezeichen erzeugt werden sollen.
- In der Registerkarte „Anzeigeoptionen“ „Lesezeichen und Seite“ aktivieren. Als Verknüpfungsdarstellung das unsichtbare Rechteck beibehalten.
- OK
- Im Menü „Acrobat“ den Menüpunkt „In Adobe PDF konvertieren“ auswählen und speichern.

Word und PDFMaker 7

- Dokument in Word öffnen
- Im Menü „Adobe PDF“ „Konvertierungseigenschaften ändern“ auswählen.
- In der Registerkarte „Einstellungen“ wählen Sie Standard. Damit sind Lesezeichen, Verknüpfungen und Tags automatisch abgedeckt.
- Unter „Erweiterte Einstellungen“ wählen Sie die Karte „Allgemein“ aus und überprüfen Sie bei „Kompatibilität“ den Eintrag „Acrobat 5.0 (PDF 1.4)“ und ob die „schnelle Webanzeige“ aktiviert ist. – OK
- Registerkarte „Word“ auswählen, Registerkarte „Lesezeichen“ auswählen und festlegen, für welche Elemente Lesezeichen erzeugt werden sollen.
- OK
- Im Menü „Acrobat“ den Menüpunkt „In Adobe PDF konvertieren“ auswählen und speichern.

Einstellungen in Acrobat

Acrobat 7

- Im Menü „Acrobat“ den Menüpunkt „Grundeinstellungen – In PDF konvertieren“ – Microsoft Office“ auswählen und „Einstellungen bearbeiten“ anklicken.
- Als „Adobe PDF-Einstellungen“ „Standard“ auswählen.
- „Bearbeiten“ auswählen. In der Karte „Allgemein“ bei „Kompatibilität“ überprüfen, ob der Eintrag „Acrobat 5.0 (PDF 1.4)“ und „schnelle Webanzeige“ eingestellt sind.
- OK
- Datei – PDF erstellen – aus Datei: Word-Datei auswählen und konvertieren.

Sicherheitseinstellungen

Dokumente, die lese- und kopiergeschützt sind, sind für ScreenReader nicht lesbar. Falls die Dokumente geschützt werden müssen, so muss unbedingt folgende Einstellung vorgenommen werden:

- Mit der Einstellung der 128bit-Verschlüsselung im Register Sicherheit anstatt der 40bit-Verschlüsselung können Dokumente im Acrobat Reader 5 (und höher) mit dem ScreenReader geöffnet werden und sind somit für sehbehinderte Menschen zugänglich. Achtung es muss sichergestellt sein, dass der „Inhaltszugriff für Sehbehinderte aktiviert ist“.

Verzichtet man auf einen Kennwortschutz zum Öffnen und Ändern durch Kennwortschutz, so können tagged PDFs bei entsprechender Kompatibilitätswahl sogar noch von Acrobat 3 gelesen werden.

Aussagekräftige Dateinamen und Metadaten

Aussagekräftige Dateinamen helfen nicht nur bei PDF, Dokumente später leichter wieder zu finden, oder überhaupt einmal zu entscheiden, ob man diese z.B. aus dem Internet herunterladen möchte. Acrobat bietet eine besonders komfortable Möglichkeit um einen Dokumententitel zu vergeben, welcher nach dem Öffnen des Dokuments erscheint. Diesen können Sie unter „Datei – Dokumenteigenschaften – Ansicht beim Öffnen – Anzeigen – Dokumententitel“ unter dem Reiter „Beschreibung“ eingeben.

Metadaten, wie sie in diesem Menübereich eingegeben werden können, sind insbesondere für externe Suchfunktionen relevant.

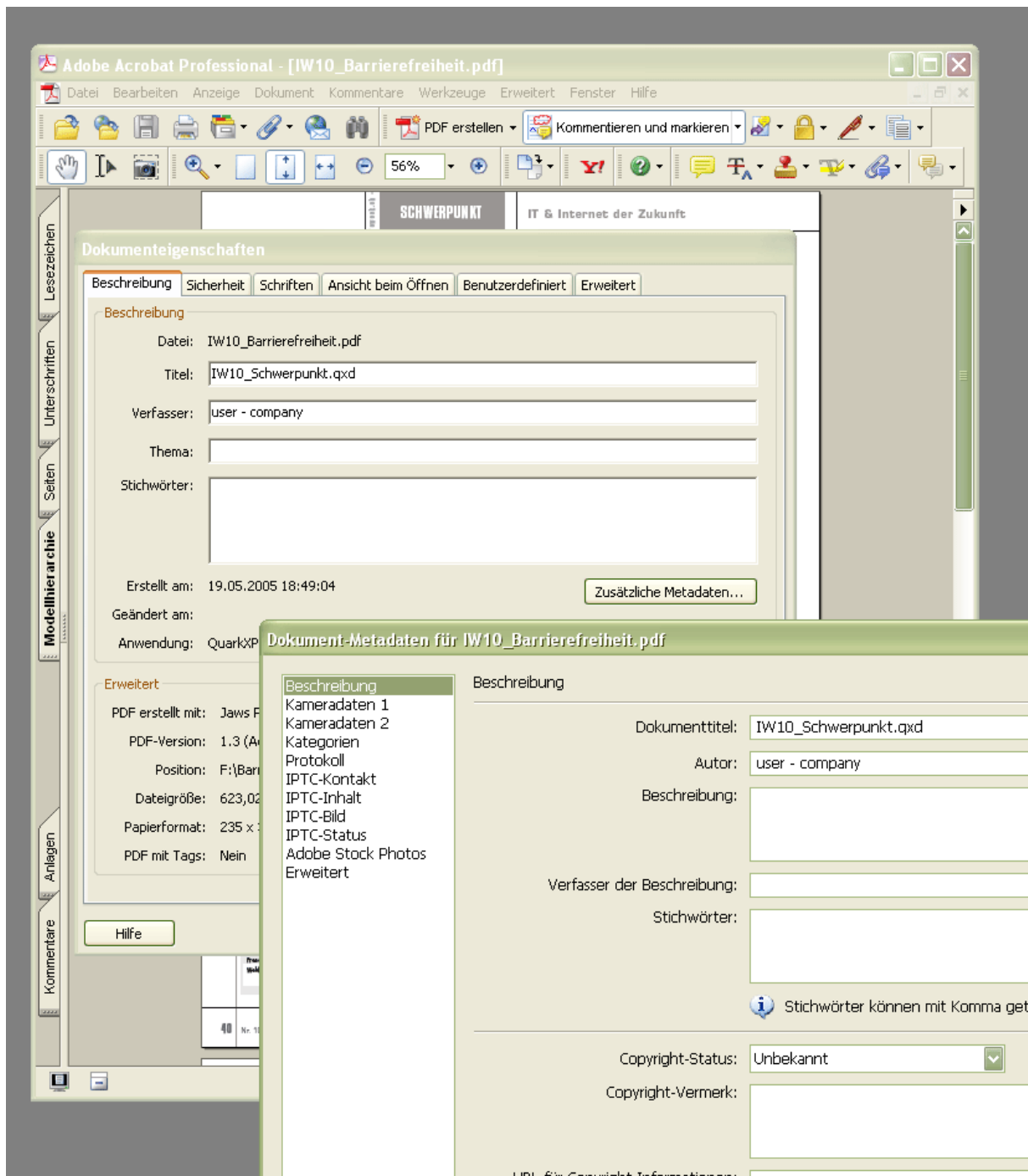


Abbildung 5: Acrobat Dialogfeld zur Beschreibung einer Datei mit Metadaten

Fazit PDF

Zusammenfassend ist zu sagen, dass obwohl PDF-Dokumente in den letzten Jahren stark verbessert wurden, sie gegenüber klassischen (X)HTML-Seiten bzgl. der Zugänglichkeit noch immer unterlegen sind, was vor allem aus dem großen Aufwand resultiert, der betrieben werden muss, um die Dokumente annähernd barrierefrei zu gestalten. Wenn es also notwendig ist PDF Dokumente anzubieten, ist immer zu überlegen, die darin enthaltenen Texte eventuell auch zusätzlich als Rich Text Format (RTF) anzubieten. Und denken Sie immer daran: PDF-Dokumente sind nur so gut wie deren Vorlagen.

Empfehlungen für den Webbereich

1. Verständliche Dokumentationsgestaltung
2. Klare Navigationsmechanismen
3. natürliche Sprache
4. Tabellen
5. Alternativen zu Audio und visuellen Inhalten
6. Farbe
7. Markup
8. Stylesheets
9. Benutzerkontrolle über zeitgesteuerte Abläufe
10. Technologien und Richtlinien von W3C

Verständliche Dokumentationsgestaltung

1.1 Verwenden Sie für die Inhalte einer Seite die klarste und einfachste Sprache, die angemessen ist.

Die gewählte Sprache muss der Zielgruppe entsprechen.

1.2 Ergänzen Sie Text mit grafischen oder audiovisuellen Präsentationen, wo dies das Verständnis der Seite erleichtert.

1.3 Verwenden Sie einen Präsentationsstil, der über Seiten hinweg konsistent ist.

Achten Sie beispielsweise darauf:

- Immer dieselben Aufzählungssymbole für dieselben Kategorien zu verwenden.
- Farbkodierungen, Schriftarten, Schriftgrößen,... je nach Einsatz beizubehalten.
- Überschriften immer nach demselben Prinzip zu kennzeichnen.
- Navigationselemente immer auf der gleichen Stelle am Bildschirm zu platzieren.

Klare Navigationsmechanismen

2.1 Identifizieren Sie das Ziel jedes Links auf klare Weise (d.h. vermeiden Sie z.B. ständig links mit „hier klicken“, „weiter“, „mehr“ oder „next“ zu setzen.

Vor allem blinde Anwender verschaffen sich gerne einen raschen Überblick über eine ihnen unbekannte Seite, indem Sie mit der Tabulatortaste von Link zu Link springen und im ersten Schritt nur diese lesen. Wenn sich auf einer Homepage ausschließlich Links befinden, die z.B. mit „weiter“ beschriftet wurden, die jedoch nicht sagen, welche Information sich hinter dem Link verbirgt, bedeutet das einen vielfachen Arbeitsaufwand zur Erschließung der Seite für blinde Benutzer.

2.2 Verwenden Sie klar strukturierte Navigationen.

Problem: Nicht jeder kann auf einer Seite schnell navigieren. Manche Benutzer müssen sehr viel überlesen, um sich auf der Seite zu Recht zu finden (z.B. wenn die Navigationsleiste sehr viele Einträge hat).

In solchen Fällen müssen Mechanismen angeboten werden, die die Navigation erleichtern. Eine oft praktizierte Möglichkeit ist das Vergeben von internen

Accesskeys (0-9), die zu den verschiedenen logischen Blöcken einer Seite führen. Zusätzlich sollte jeder dieser Blöcke Metainformation beinhalten.

Logische Blöcke einer Seite können bspw. sein:

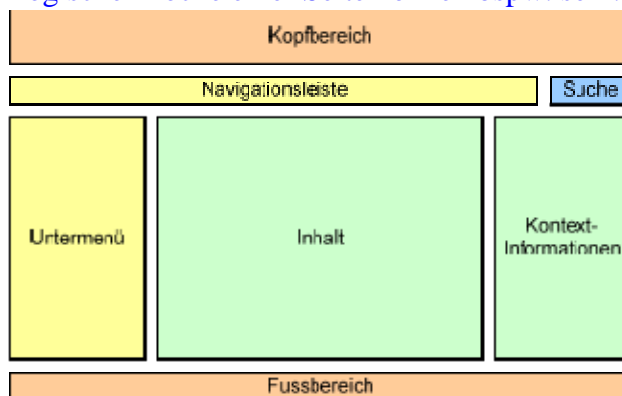


Abbildung 6: 7 Logische Blöcke einer Website (mehr als 7 zu identifizieren macht normalerweise keinen Sinn)

```
<body>
<ul>
  <li><a href="#content" accesskey="0">Zum Inhalt (ALT+0)</a> |
</li>
  <li><a href="#navbar" accesskey="1">Zum Inhalt (ALT+1)</a> | </li>
  ...
</ul>
...
<div class="navbar"> <a name="navbar">Navigationsleiste:</a> ...
  <hr />
</div>
<div class="content"> <a name="content">Inhalt:</a> ...
  <hr />
</div>
...
</body>
```

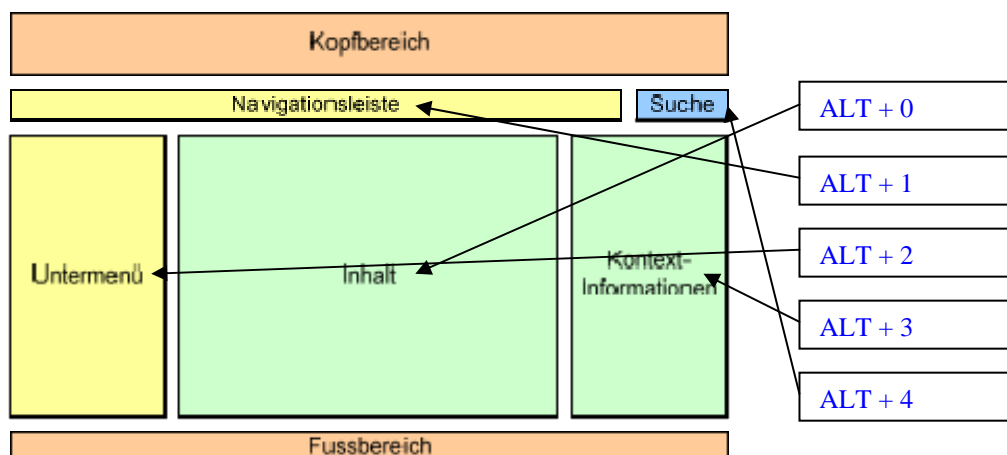


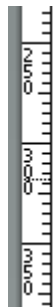
Abbildung 7: Übliche Vergabe der Accesskeys (es existieren bislang keine verbindlichen Richtlinien für die Reihenfolge. Die hier angegebene ist jedoch durchaus geläufig)

2.2.1 Stellen Sie Navigationsleisten zur Verfügung

Manche Benutzer haben Probleme damit, geschachtelten Untermenüs zu verstehen (z.B. lesen die meisten Screenreader die Menüpunkte alle der Reihe nach vor, ohne darauf Rücksicht zu nehmen, ob es sich um ober- oder untergeordnete Punkte handelt. Weiters werden im Normalfall alle Menüpunkte vorgelesen, auch solche, die zugeklappt sind – d.h. für einen sehenden Benutzer nicht sichtbar sind. Bei sehr langen Menüs ergeben sich dadurch Schwierigkeiten).

Eine Lösung wäre der Einsatz von geschachtelten Listen. Bei dieser Verschachtelung werden zusätzliche Informationen über die Hierarchieebene angeboten.

```
<body>
<map name="menu" title="Menü">
  <ul>
    <li><a href="..."><dfn>1</dfn>Eintrag 1</a></li>
    <li><a href="..."><dfn>2</dfn>Eintrag 2</a></li>
    <li><a href="..."><dfn>3</dfn>Eintrag 3</a>
      <ul>
        <li><a href="..."><dfn>3.1</dfn>Untereintrag 1</a></li>
        <li><a href="..."><dfn>3.2</dfn>Untereintrag 2</a></li>
        ...
      </ul>
    </li>
    <li><a href="..."><dfn>4</dfn>Eintrag 4</a></li>
    ...
  </ul>
</map>
</body>
```



- 1Eintrag 1
- 2Eintrag 2
- 3Eintrag 3
 - 3.1Untereintrag 1
 - 3.2Untereintrag 2...
- 4Eintrag 4...

Grafik: Darstellung des Codebeispiels. Das Erscheinungsbild dieser Liste kann mit CSS beinahe beliebig angepasst werden.

2.2.2 Gruppieren Sie verwandte Links

Es sollte sich schon in der Navigation zeigen, welchen zusammenhängende Themenblöcke auf einer Seite behandelt werden.

Natürliche Sprache

3.1 Geben Sie zu Beginn des Dokuments dessen natürliche Sprache an.

D.h. geben Sie bitte die Sprache bekannt, in der der Großteil der Texte auf der jeweiligen Website verfasst wurde.

3.2. Kennzeichnen Sie Änderungen in der natürlichen Sprache in HTML mit dem „lang“-Attribut und in XML mit „xml:lang“. (Anmerkung: lang ist ein Universalattribut. D.h. das es in mehreren HTML-Elementen vorkommen darf. „lang“ kann in allen Attributen eingesetzt werden, außer: applet, base, basefont, br, frame, frameset, hr, iframe, meta, param, script)

Beispiel HTML:

```
<body>
...
<p> Heute unterhalten wir uns über <ACRONYM lang="eng"
TITLE="englisch: Compact Disc - Read Only Memory">CD-ROM</ACRONYM>.
</p>
...
</body>
```

Beispiel XML:

```
<body>
...
<p> Hier können Sie mehr zum Thema <span lang="en"
xml:lang="en">Copyright </span> lesen. </p>
...
</body>
```

3.3 Erklären Sie jede Abkürzung und jedes Akronym an der Stelle ihres ersten Auftretens (in HTML mit „title“-Attribut und den Elementen ABBR [für Abkürzungen] und ACRONYM).

Beispiel ABBR:

```
<body>
...
<p> Gestern kam es zu einem verheerenden Unfall, in welchem zwei
<ABBR TITLE="Personenkraftwagen">PKW </ABBR>und <ABBR
TITLE="Lastkraftwagen">LKW</ABBR> verwickelt waren. </p>
...
</body>
```

Beispiel ACRONYM:

```
<body>
...
<p> Heute unterhalten wir uns über <ACRONYM lang="eng"
TITLE="englisch: Compact Disc - Read Only Memory">CD-ROM</ACRONYM>.
</p>
...
</body>
```

Tabellen

4.1 Tabellen sind nicht zu Layout-Zwecken gedacht. Verwenden Sie Tabellen daher ausschließlich zur Übermittlung von Daten, deren Präsentation in Tabellenform zweckdienlich ist.

4.1.1 Kennzeichnen Sie bei Datentabellen Zeilen- und Spaltenüberschriften (in HTML mit TD für Datenzellen, TH für Überschriften).

4.1.2 Stellen Sie eine Zusammenfassung zu jeder größeren Tabelle bereit (Attribut: summary).

Einfache Tabelle:

```
<table border="1" summary="Accesskeys der Seite">
  <caption>Accesskeys in der Navigation:</caption>

  <thead>
    <tr>
      <th id="t">Taste</th>
      <th id="b">Beschreibung</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td headers="t">ALT + 0</td>
      <td headers="b">Inhalt</td>
    </tr>
    <tr>
      <td headers="t">ALT + 1
      <td headers="b">Hauptnavigation</td>
    </tr>
    [...]
  </tbody>
</table>
```

...

Accesskeys in der Navigation:

Taste	Beschreibung
ALT + 0	Inhalt
ALT + 1	Hauptnavigation
ALT + 2	Untermenü
ALT + 4	Kontextinformation
ALT + 5	Suche

...

Abbildung 8: Zeigt einen Screenshot des ausgeführten vorangegangenen Codebeispiels

4.1.3 Gruppieren Sie zueinander gehörende Daten (in HTML mit THEAD, TFOOT und TBODY, COL und COLGROUP).

4.1.4 Vermeiden Sie leere GIFs (Bilder) zu Platzhalterzwecken. Sollte es unbedingt notwendig sein ein leeres GIF einzusetzen, verwenden Sie bitte folgende Beschriftung:

```
<p></p>
```

Alternativen zu Audio und visuellen Inhalten

5.1 Stellen Sie für jedes Nicht-Text-Element ein Text-Äquivalent zur Verfügung; d.h. beschreiben Sie jedes Bild, jedes Video, jede Audioaufnahme, selbstablaufende Präsentationen etc, in textlicher Form.

5.1.1 Verwenden Sie „alt“ für die IMG-, INPUT- und APPLET-Element oder stellen Sie ein Text-Äquivalent im Inhalt des OBJECT- und APPLET-Element bereit.

Beispiel: Eingebundene Grafik

```
<p><p>
```

Beispiel: Formular mit Input-Attribut

```
<body>
<h1>Formular für Namenseingabe</h1>
<form action="input_text.htm">
  <p>Vorname:<br><input name="vorname" type="text" size="30"
maxlength="30"></p>
  <p>Zuname:<br><input name="zuname" type="text" size="30"
maxlength="40"></p>
</form>
</body>
```

Formular für Namenseingabe

Vorname:

Zuname:

[zurück](#)

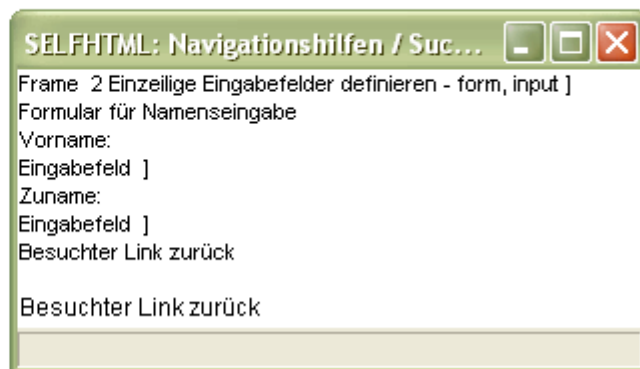


Abbildung 9: Bei diesem Beispiel sehen Sie im rechten Bereich die Ausgabe eines Screenreaders. Da die Formatierung der Formularefelder korrekt vorgenommen wurde, kann ein eindeutiger Zusammenhang zwischen der Beschriftung und dem jeweiligen Eingabefeld hergestellt werden.

5.1.2 Stellen Sie für komplexen Inhalt, bei dem das „alt“-Tag nicht genügt, zusätzliche Beschreibungen z.B. mit „longdesc“ im IMG und FRAME, einen Link innerhalb eines OBJECT-Elements, oder einen Beschreibungs-Link.

Bei Bildern (IMG):

Wenn die Möglichkeit, mit dem alt-Attribut einen gewöhnlichen Alternativtext für eine Grafik zu notieren, nicht genügt, kann seit HTML 4.0 auch zu einer anderen Stelle oder Datei verwiesen, wo sich nähere Textinformation zu der Grafik befindet. Der Browser könnte beispielsweise nur dann einen Verweis zu dieser Stelle anbieten, wenn die Grafik nicht angezeigt werden kann.

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
    „http://www.w3.org/TR/html4/strict.dtd">
<html>
  <head>
    <title>Langbeschreibung f&uuml;r Grafiken</title>
  </head>
  <body>
    <h1>Alte Zeiten</h1>
    <p><img scr="altezeiten.jpg" width=271 height="265" alt="alte
Zeiten" longdesc="altezeiten.htm"></p>
  </body>
</html>

```



Abbildung 10: Grafiken mit Alternativtexten beschriften

Bei Frames (FRAME):

Mit dem Attribut `longdesc` in einem `<frame>`-TAG können Sie auf eine Stelle verweisen, wo der Inhalt dieses Frame-Fensters in einer längeren Beschreibung zusammengefasst wird. Erlaubt ist ein URI, d.h. bei der Angabe gelten die Regeln zum Referenzieren in HTML. Dieses Attribut kommt bei Browsern zum Einsatz, die Frames zwar erkennen, aber bei denen die Anzeige von Frames nicht möglich ist, wie z.B. bei akustische Ausgabesystemen.

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN"
    "http://www.w3.org/TR/html4/frameset.dtd">
<html>
<head>
<title>Langbeschreibung für Frame-Fenster</title>

```



```
</head>
<frameset cols="250,*">
  <frame src="verweise.htm" name="Navigation">
  <frame src="startseite.htm" name="Daten"
longdesc="startseite_extra.htm">
  <noframes>
    <body>
      <p>Alternativ-Inhalt (Zusammenfassung, Sitemap ...)</p>
    </body>
  </noframes>
</frameset>
</html>
```

- 5.1.3 Vermeiden Sie wenn möglich Imagemaps. Sollten Sie Imagemaps einsetzen, verwenden Sie entweder das „alt“-Attribut bei AREA oder das MAP-Element mit A-Elementen (und zusätzlichem Text) als Inhalt; d.h. stellen Sie redundante Textlinks zur Verfügung.

Beispiel siehe: http://de.selfhtml.org/html/grafiken/verweis_sensitive.htm#definieren (Verweis-sensitive Grafiken definieren)

- 5.2 Beim unvermeidbaren Einsatz von Platzhalterbildern, oder auch bei dekorativen Elementen, verwenden Sie ein leeres alt-Attribut: alt="";

```
<p><p>
```

Farbe

- 6.1 Sorgen Sie dafür, dass alle Ihre Informationen auch gänzlich ohne Farbe verständlich dargestellt werden (schwarz-weiß, keine Graustufen).

- 6.2 Achten Sie auf einen ausreichenden Kontrast bei Vorder- und Hintergrundfarbe.

„Von einer guten Sichtbarkeit zweier Farben kann dann ausgegangen werden, wenn der Helligkeitsunterschied und die Farbdifferenz zwischen den beiden Farben größer ist als ein bestimmter Grenzwert. Der von W3C vorgeschlagene Grenzwert ist > 125 für die Helligkeit und > 500 für die Farbdifferenz“.

Es stehen einige Programme zur Verfügung, mit welchen sich diese Kontraste berechnen lassen.

Formel für die Helligkeit von Farben:

$((\text{Rot-Wert} \times 299) + (\text{Grün-Wert} \times 587) + (\text{Blau-Wert} \times 114)) / 1000$

Der Unterschied zwischen der Helligkeit des Hintergrundes und des Vordergrundes sollte größer als 125 sein.

Formel für den Farbunterschied:

$(\text{Maximum}(\text{Rot-Wert1}, \text{Rot-Wert2}) - \text{Minimum}(\text{Rot-Wert1}, \text{Rot-Wert2})) +$
 $(\text{Maximum}(\text{Grün-Wert1}, \text{Grün-Wert2}) - \text{Minimum}(\text{Grün-Wert1}, \text{Grün-Wert2})) +$
 $(\text{Maximum}(\text{Blau-Wert1}, \text{Blau-Wert2}) - \text{Minimum}(\text{Blau-Wert1}, \text{Blau-Wert2}))$

Die Differenz zwischen Hintergrundfarbe und Vordergrundfarbe sollte größer als 500 sein.

Analysiert ob die Kontraste zwischen Hintergrund und Text stark genug sind.

<http://juicystudio.com/services/colourcontrast.php>

6.3. Denken Sie bei der Gestaltung an Personen mit Farbfehlsichtigkeiten (wie z.B. Rot-Grün Farbenblindheit). Informationen wie „Drücken Sie den grünen Button“ sind für diese Benutzergruppe nicht brauchbar.

[Vischeck simuliert, wie Bilder für Farbenblinde dargestellt werden.](http://www.vischeck.com/)

<http://www.vischeck.com/>

Markup

7.1. Erstellen Sie Dokumente, welche den formalen Grammatiken der Programmiersprache entsprechen.

7.1.1 Verwenden Sie eine Dokumententype-Deklaration am Anfang Ihres Dokuments.

7.1.2 Verwenden Sie ausschließlich W3C-konforme TAGS (HTML-TAGs, XHTML-TAGs, etc.)

7.1.3 Zeichnen Sie alle Teile Ihrer Web-Dokumente mit entsprechenden TAGS aus.

`<head>`, `<body>`, `<h1>`, `<p>`, ...

7.1.4 Verwenden Sie Überschriften-Elemente (h1, h2,...), um die Struktur eines Dokuments darzustellen.

7.1.5 Verwenden Sie Markup für Listen und Listenelemente (OL-, UL- und DL-Listen).

`` steht für sortierte Listen (ol = odered list). `` wird in Nummerierungen umgewandelt:

```
<h1>Sortierte Liste</h1>
<ol>
  <li>auwachen</li>
  <li>aufstehen</li>
  <li>waschen</li>
  <li>anziehen</li>
  <li>frühstücken</li>
  <li>Zähneputzen</li>
  <li>frisieren</li>
</ol>
```

`` leitet unsortierte Listen (ul = unordered list) ein. Mit `` beginnt ein neuer Punkt innerhalb der Liste (li = list item).

Listen können auch verschachtelt werden.

```
<h1>Unsortierte Liste</h1>
<ul>
  <li>Großkatzen
    <ul>
      <li>Löwe</li>
      <li>Tiger</li>
      <li>Leopard</li>
    </ul>
  </li>
  <li>Paarhufer
    <ul>
      <li>Schweineartige</li>
      <li>Schwielensohler</li>
    </ul>
  </li>
</ul>
```

```
<li>Wiederkäuer</li>
</ul>
</li>
</ul>
</body>
```

Sortierte Liste

1. auwachen
2. aufstehen
3. waschen
4. anziehen
5. frühstücken
6. Zähneputzen
7. frisieren

Unsortierte Liste

- Großkatzen
 - ◊ Löwe
 - ◊ Tiger
 - ◊ Leopard
- Paarhufer
 - ◊ Schweineartige
 - ◊ Schwielensohler
 - ◊ Wiederkäuer

Abbildung 11: Grafik einer sortierten und einer unsortierten Liste in klassischer Browserdarstellung

Definitionenlisten `<dl>` sind für Glossare gedacht.

`<dl>` leitet eine Definitionsliste ein (dl = definition list)

`<dt>` leitet einen zu definierenden Ausdruck ein (dt = definition (list) item)

`<dd>` leitet eine Definition eines Ausdrucks ein (dd = definition (list) definition)

```
<h1>Abkürzungen</h1>
<dl>
  <dt>g</dt>
  <dd>grin</dd>
  <dt>fg</dt>
  <dd>fed grin</dd>
  <dt>eg</dt>
  <dd>evil grin</dd>
</dl>
```

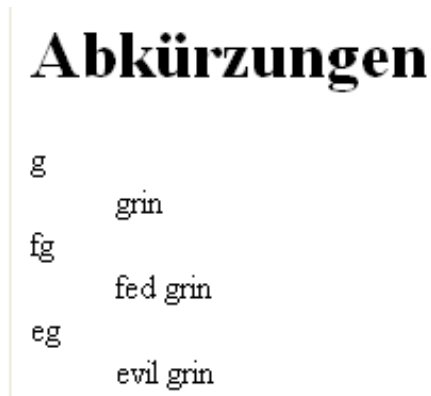


Abbildung 12: Grafik verschiedener Abkürzungen in klassischer Browserdarstellung in Listenform

7.1.6 Wenn eine angemessene Markup-Sprache existiert, verwenden Sie Markup anstelle von Bildern, um Informationen darzustellen (z.B. MathML).

7.1.7 Verwenden Sie Markup für Zitate (in HTML Q und BLOCKQUOTE, um kürzere und längere Zitate zu kennzeichnen).

```
<body>
<h1>Ernest Hemingway</h1>

<p>Hemingway war nicht nur Schriftsteller, sondern auch Reporter und
Kriegsberichterstatter.</p>

<blockquote>
  <p>As you get older it is harder to have heroes, but it is sort of
necessary. </p>
</blockquote>

</body>
```

Stylesheets

8.1 Bauen Sie Dokumente so auf, dass sie ohne Stylesheets gelesen werden können.

Hiermit ist gemeint, dass die wichtigsten Informationen einer Seite auch dann korrekt gelesen werden können, wenn dem Benutzer keinerlei Layout zur Verfügung steht. D.h. dass darauf geachtet werden muss, dass wichtige Informationen

- nicht ausschließlich über die Formatierung vermittelt werden und
- Informationen auch ohne Style-Angaben in der richtigen Reihenfolge übermittelt werden.

8.2. Trennen Sie sämtliche layoutspezifischen Angaben in Websites wie z.B.: Hintergrund- und/oder Schriftfarben, Schriftarten, Schriftgrößen, Bilder, Roll-Over-Effekte etc. vom reinen Homepagecode [(X)HTML], und setzen Sie dafür CSS ein.

8.3 Verwenden Sie relative anstatt absolute Einheiten bei Größenangaben und Platzierungen (anstelle von <px>).

Benutzerkontrolle über zeitgesteuerte Abläufe

9.1 Geben Sie den Benutzern immer die Möglichkeit, Abläufe (z.B. von Videos, Präsentationen) selbst zu steuern.

9.1.1 Vermeiden Sie Bildschirmflackern, blinkende Inhalte, automatische Seitenrefreshs, automatische Seitenweiterleitungen und bewegte Seiten, bis Benutzeragenten die Kontrolle darüber erlauben.

Anmerkung: Benutzeragenten sind Programme, mit denen der Benutzer bestimmte Abläufe steuern kann. Dazu gehören z.B. Browser für HTML oder XHTML, Bildbetrachtungsprogramme oder spezifische Technologien für Behinderte wie Screenreader, Braillezeile, Programme die Bildschirminhalte vergrößert darstellen, etc.

Technologien und Richtlinien von W3C

Verwenden Sie W3C-Technologien, wenn diese verfügbar und angemessen sind, und die dem aktuellen Stand der Technik entsprechen.

W3C-Technologien sind z.B. HTML, XHTML, XML, SMIL, SVG, CSS, RDF,... Viele Nicht-W3C-Formate (z.B. PDF, Shockwave,...) können mit Standard-Benutzeragenten (einschließlich assistiver Technologien) nicht ohne weiteres verarbeitet (betrachtet) werden. Die Vermeidung proprietärer Technologien (proprietäre Elemente, Attribute,...) wird in der Tendenz Seiten die Zugänglichkeit von Seiten für Menschen mit Einschränkungen verbessern. Wenn nicht zugängliche proprietäre Technologien (egal ob proprietär oder nicht) verwendet werden müssen, müssen äquivalente zugängliche Seiten bereitgestellt werden.

Schluss

Wenngleich hier der Anspruch auf Perfektion möglicherweise vermisst wird, stehen wir dafür ein, dass auch kleine Schritte ans Ziel führen. Werden die nun weiter beschriebenen Tipps und Richtlinien befolgt ist das ein erster Anstoß für ein Umdenken, dass Wellen in unserer Gesellschaft schlagen wird.

Abbildungsverzeichnis

Abbildung 1: Erklärende Grafik, wie Alternativtexte in Word richtig vergeben werden können.	3
Abbildung 2: Grafik beschreibt, wie Bilder in Word "Mit Text in Zeile" formatiert werden, um später korrekt in barrierefreien PDFs angezeigt zu werden.	5
Abbildung 3: Wenn Bilder korrekt in tagged PDFs übernommen werden sollen, dürfen sie keine Füllungen haben.	6
Abbildung 4: Menü in Word - Anzeigen von Lesezeichen	6
Abbildung 5: Acrobat Dialogfeld zur Beschreibung einer Datei mit Metadaten	9
Abbildung 6: 7 Logische Blöcke einer Website (mehr als 7 zu identifizieren macht normalerweise keinen Sinn)	11
Abbildung 7: Übliche Vergabe der Accesskeys (es existieren bislang keine verbindlichen Richtlinien für die Reihenfolge. Die hier angegebene ist jedoch durchaus geläufig)	11
Abbildung 8: Zeigt einen Screenshot des ausgeführten vorangegangenen Codebeispiels	14
Abbildung 9: Bei diesem Beispiel sehen Sie im rechten Bereich die Ausgabe eines Screenreaders. Da die Formatierung der Formularfelder korrekt vorgenommen wurde, kann ein eindeutiger Zusammenhang zwischen der Beschriftung und dem jeweiligen Eingabefeld hergestellt werden.	15
Abbildung 10: Grafiken mit Alternativtexten beschriften	16
Abbildung 11: Grafik einer sortierten und einer unsortierten Liste in klassischer Browserdarstellung	19
Abbildung 12: Grafik verschiedener Abkürzungen in klassischer Browserdarstellung in Listenform	20

Quellen

Binder, Urs: Barrierefreiheit kostet nichts, InfoWeek.ch Nr. 10, 17. 05. 2005, S. 40 – 42, letzter Zugriff: 28. August 2006

<http://www.namics.com/wissen/artikel-interviews/artikel/barrierefreiheit-kostet-nichts.html#275>

W3C Recommendation, 5. Mai 1999, letzter Zugriff: 28. August 2006

<http://www.w3c.de/Trans/WAI/webinhalt.html#Guidelines>

Aktion Mensch und Stiftung Digitale Chancen: Kriterien zum BIENE-Award 2006, letzter Zugriff: 28. August 2006

<http://www.biene-award.de/award/kriterien/>

Nussbaum, Gerhard: Vortrag im Rahmen des eLearning Tags „eLearning an der LFUI – Rückblick und Ausblick“: Barrierefreie Web-Gestaltung, 28. September 2006

HTML-Referenz: SELFHTML ,letzter Zugriff: 10. Okt. 2006

<http://de.selfhtml.org>

Angie Radtke, Dr. Michael Charlier: Barrierefreies Webdesign, Attraktive Websites zugänglich gestalten, Addison-Wesley, 2006

eGovernmentgesetz, letzter Zugriff: 14. 1. 2008

<http://www.cio.gv.at/egovernment/law/>

Bundes-Behindertengleichstellungsgesetz, letzter Zugriff: 14. 1. 2008

<http://www.bizeps.or.at/gleichstellung/rechte/bgstg.php>

PDF-Dokumente – lesbar für Alle, letzter Zugriff: 14. 1. 2008

http://www.einfach-fuer-alle.de/artikel/pdf_barrierefrei/