

Freie wissenschaftliche Arbeit zur Erlangung des akademischen Grades  
Bachelor of Science in Wirtschaftsinformatik

# **Barrierefreie Gestaltung von Internetseiten**

## **Bachelorthesis**

im Fachbereich Wirtschaftswissenschaften II  
im Studiengang Wirtschaftsinformatik  
der Fachhochschule für Technik und Wirtschaft Berlin

vorgelegt von: Christiane Müller  
Am Falkenberg 98  
12524 Berlin  
Matrikel-Nr: 0504010

Erstbetreuer: Dipl. Math. Petra Utke  
Zweitbetreuer: Prof. Dr. Thomas Pietsch

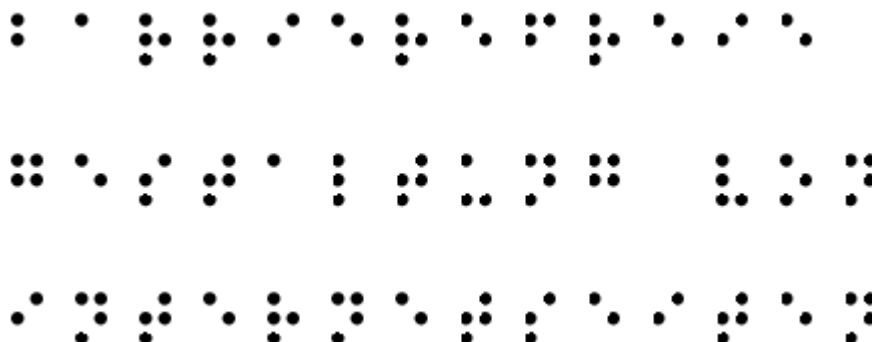
Abgabetermin: 21.03.2005



## Vorwort

In meiner Ausbildung zur Informations- und Telekommunikationssystem Kauffrau entdeckte ich ein neues Hobby: Webdesign. Am 01.11.2000 erstellte ich meine erste Internetseite.

HTML wurde zu meiner zweiten Muttersprache, der Umstieg auf XHTML glich dem Ablegen eines umgangssprachlichen Dialektes. Und wie es sich für ein richtiges Hobby gehört, bei der Perfektionierung der eigenen Internetpräsenz [www.it-infothek.de](http://www.it-infothek.de) spielte Zeit bald keine Rolle mehr. Resultierend aus der Vorliebe zum Webdesign entstand dann auch das Thema dieser Bachelor-Thesis: „Barrierefreie Gestaltung von Internetseiten“. Die Faszination dieses Themas kann sehr einfach verdeutlicht werden:



Das ist Blindenschrift und die meisten Leser werden aus diesen Zeilen keine Informationen ziehen können. Ähnlich geht es blinden, sehbehinderten und vielen anderen behinderten Nutzern des Internets. Ist eine Internetpräsenz nicht optimal gestaltet, bleibt behinderten Nutzern und vor allem Blinden der Zugriff auf die Informationen verwehrt.

Ich möchte an dieser Stelle allen danken, die mich auf dem Weg meiner Webdesign-Geschichte begleitet oder während der letzten drei Bachelor-Thesis-Monate unterstützt haben. Vielen Dank an Herrn Prof. Dr. Thomas Pietsch, der mich in meinem Eifer gebremst und mir den Blick für's Wesentliche gezeigt hat. Mein Dank geht auch an Frau Dipl. Math. Petra Utke, die mir mit gutem Rat zur Seite stand und in den letzten Wochen noch einmal für frischen Wind sorgte.

Und ganz besonderer Dank geht an Frau Dipl. Ing. Christin Wilde, meiner persönlichen Mentorin, die mich in der Studienplatzwahl unterstützte, mir einen Arbeitsplatz und mehrere Aufträge im Bereich des Webdesign verschaffte und immer wie eine große Schwester war. Danke!

Berlin, März 2005

Christiane Müller  
[christiane@it-infothek.de](mailto:christiane@it-infothek.de)

# Inhaltsverzeichnis



Vorwort.....	I
Inhaltsverzeichnis.....	II
Abbildungsverzeichnis.....	IV
1 Einleitung.....	1
2 Begriffsbestimmungen.....	2
2.1 Barrierefreiheit.....	2
2.1.1 Standardkonformität.....	2
2.1.2 Zugänglichkeit.....	2
2.1.3 Gebrauchstauglichkeit.....	2
2.1.4 Verständlichkeit.....	2
2.1.5 Design.....	2
2.2 Nutzerorientierte Gestaltung.....	3
2.2.1 Behinderungen.....	3
2.2.2 Screenreader.....	4
2.2.3 Braillezeile.....	4
2.2.4 Sprachausgabe.....	4
2.3 World Wide Web Consortium (W3C).....	4
3 Grundlagen.....	5
3.1 Web Content Accessibility Guidelines 1.0 (WCAG1).....	5
3.1.1 Zugänglichkeitsrichtlinien.....	5
3.1.2 Priorität 1.....	5
3.1.3 Priorität 2.....	6
3.1.4 Priorität 3.....	7
3.1.5 Konformitätsstufen.....	7
3.2 Barrierefreie Informationstechnik-Verordnung (BITV).....	8
4 Technologien.....	9
4.1 Aktuelle Technologien.....	9
4.2 Extensible Hypertext Markup Language (XHTML).....	10
4.2.1 Deklaration und Definition.....	10
4.2.2 Kopfbereich und Textkörper.....	11
4.3 Cascading Stylesheets (CSS).....	12
4.3.1 Stylesheet-Ebenen.....	13
4.3.2 Verschiedene Formatvorlagen.....	14
4.4 Dynamische Inhalte.....	16
5 Grafiken und Farbe.....	17
5.1 Schriftgrafiken.....	17
5.2 Informative Grafiken.....	18
5.3 Layoutgrafiken.....	19
6 Schrift und Sprache.....	20
6.1 Schrift.....	20
6.1.1 Textkörper.....	20
6.1.2 Vererbungsprinzip.....	21
6.2 Sprache.....	22
6.2.1 Sprachwechsel.....	22
6.2.2 Zitate.....	23
6.2.3 Abkürzungen und Akronyme.....	24

---

7 Tabellen und Formulare.....	25
7.1 Tabellen.....	25
7.1.1 Datentabellen.....	25
7.1.2 Layouttabellen.....	27
7.2 Formulare.....	28
8 Frames und Container.....	29
8.1 Frames.....	29
8.2 Container.....	30
9 Seitenaufbau.....	32
9.1 Dokumentenaufbau.....	32
9.1.1 Überschriften.....	32
9.1.2 Listen.....	33
9.2 Kopfdaten.....	33
9.2.1 Metadaten.....	34
9.2.2 Seitennavigation.....	34
9.3 Navigation.....	35
9.4 Tastaturbedienung.....	36
10 Testwerkzeuge.....	38
10.1 Browser-Tests.....	38
10.1.1 Lynx.....	38
10.1.2 Opera.....	38
10.1.3 Internet Explorer.....	39
10.2 Automatische und manuelle Tests.....	40
10.2.1 Bobby.....	40
10.2.2 Barrierefinder.....	41
10.2.3 WebFormator.....	41
10.2.4 Validierungsservices.....	42
11 Ergebnisse.....	43
11.1 Von traditioneller zu standardbasierter Entwicklung.....	43
11.2 Chancen und Risiken.....	44
11.3 Barrierefreiheit in Wirtschaft und Gesellschaft.....	45
12 Fazit.....	46
Literaturverzeichnis.....	V
Anhang.....	IX
A Richtlinien der Web Content Accessibility Guidelines 1.0.....	X
B Checkpunkte der Web Content Accessibility Guidelines 1.0.....	XI
B.1 Checkpunkte der Priorität 1.....	XI
B.2 Checkpunkte der Priorität 2.....	XII
B.3 Checkpunkte der Priorität 3.....	XIII
C Anlage 1 der Barrierefreie[n] Informationstechnik-Verordnung.....	XV
C.1 Anforderungen und Bedingungen der Priorität 1.....	XV
C.2 Anforderungen und Bedingungen der Priorität 2.....	XVIII

# Abbildungsverzeichnis

Abb. 1: Aufbau der Web Content Accessibility Guidelines 1.0.....	5
Abb. 2: Konformitätsstufen.....	7
Abb. 3: Aufbau der Barrierefreie[n] Informationstechnik-Verordnung.....	8
Abb. 4: Browser-Ansicht mit Formatvorlage.....	15
Abb. 5: Druckvorschau mit Formatvorlage.....	15
Abb. 6: Wechsel zwischen Formatvorlagen im Mozilla.....	15
Abb. 7: Schriftgrößen im Internet Explorer.....	20
Abb. 8: Vererbungsprinzip.....	21
Abb. 9: Sprachwechsel.....	23
Abb. 10: Zitate im Internet Explorer.....	23
Abb. 11: Zitate im Mozilla.....	23
Abb. 12: Abkürzungen im Internet Explorer.....	24
Abb. 13: Abkürzungen im Mozilla.....	24
Abb. 14: Datentabelle.....	26
Abb. 15: Layouttabelle.....	27
Abb. 16: Formulare.....	28
Abb. 17: Frame-Ansicht.....	30
Abb. 18: Noframe-Ansicht.....	30
Abb. 19: Container.....	30
Abb. 20: Quelltext Container 1.....	31
Abb. 21: Quelltext Container 2.....	31
Abb. 22: Ausgabe Container 1.....	31
Abb. 23: Ausgabe Container 2.....	31
Abb. 24: Linearisierung Container 1.....	31
Abb. 25: Linearisierung Container 2.....	31
Abb. 26: Überschriften.....	32
Abb. 27: Liste bei eingeschalteten Grafiken.....	33
Abb. 28: Liste bei ausgeschalteten Grafiken.....	33
Abb. 29: Site-Navigationsleiste im Mozilla.....	35
Abb. 30: Original AccessKey-Pad.....	37
Abb. 31: Benutzermodus im Opera.....	38
Abb. 32: Web Accessibility Toolbar im Internet Explorer.....	39
Abb. 33: Überprüft mit Bobby.....	40
Abb. 34: Überprüft mit WebFormator.....	41
Abb. 35: Überprüft mit Validierungsservices des W3C.....	42
Abb. 36: BIENE-Award.....	45

# 1 Einleitung



Öffentliche Verkehrsmittel und Gebäude sind längst behindertengerecht gestaltet, die Debatte um abgesenkte Bordsteine vergessen. Behindertengerechte Gehwege sind selbstverständlich im Straßenbild und nicht nur Rollstuhlfahrer, sondern auch Eltern mit Kinderwagen, Radfahrer oder ältere Menschen möchten sie nicht mehr missen.

Viele Menschen mit Behinderungen nutzen das Internet, da es ihnen neue Möglichkeiten eröffnet, aktiver am öffentlichen Leben teilzunehmen und ihnen den Alltag erleichtern kann. Leider sind die Betroffenen oftmals Barrieren ausgesetzt, die sie daran hindern, an Informationen zu gelangen oder Online-Services in Anspruch zu nehmen.

Ziel dieser Arbeit ist es, die wesentlichsten Kriterien herauszuarbeiten, die der Umsetzung barrierefreier Internetseiten dienen. Hierbei sollen vor allem ergänzende Angaben innerhalb einer Internetseite identifiziert werden, die behinderten Nutzern das Arbeiten mit dem Medium Internet erleichtern.

Tabellenbasierte Layouts, Vermischung von Inhalt und Gestaltung, semantisch fehlerhafter Code: Wesentliche Merkmale der „traditionellen“ Internetseite, entstanden durch den Versuch, Internetseiten als eine Erweiterung der Print-Medien zu sehen und auf eine pixelgenaue Darstellung für einen bestimmten Browser (meist den Internet Explorer) ausgelegt.

Im Verlauf dieser Arbeit sollen Standards und Richtlinien vorgestellt werden, deren Umsetzung Barrieren der traditionellen Internetseite abbauen. Das Streben nach einer **nutzerorientierten Gestaltung von elektronischen Informationen** ist ein großer Schritt in Richtung barrierefreie Informationstechnik.<sup>1</sup>

Das barrierefreie Gestalten von Internetseiten ist einerseits für die Nutzergruppe der behinderten Menschen erforderlich, schafft andererseits jedoch auch beste Voraussetzungen für medienneutrales Publizieren auf kleinen Displays, Mobiltelefonen und zunehmend verbreiteten Zugangsarten in Zug, Auto oder Flugzeug.<sup>2</sup>

Dieser Ausarbeitung ist eine CD-ROM beigelegt, die alle nachfolgenden Beispiele sowie die genutzten Online-Quellen zum Zeitpunkt des Zugriffs enthält. Die in den Fußnoten genannten Seitenzahlen von Online-Quellen (dargestellt in Klammern) entsprechen den Seitenzahlen des jeweiligen auf CD-ROM archivierten Dokumentes.

---

1) Vgl. Hellbusch 2005 a, S. 7

2) Vgl. Bittner 2005 (S. 6)

---

## 2 Begriffsbestimmungen

### 2.1 Barrierefreiheit

Am 01.05.2002 trat in Deutschland das *Gesetz zur Gleichstellung behinderter Menschen (Behindertengleichstellungsgesetz - BGG)* in Kraft. In BGG § 4 heißt es: „Barrierefrei sind [...] Systeme der Informationsverarbeitung [...], wenn sie für behinderte Menschen in der allgemein üblichen Weise, ohne besondere Erschwernis und grundsätzlich ohne fremde Hilfe zugänglich und nutzbar sind.“<sup>1</sup> Barrierefreiheit ist unterteilbar in die fünf Dimensionen Standardkonformität, Zugänglichkeit, Gebrauchstauglichkeit, Verständlichkeit und Design.<sup>2</sup>

#### 2.1.1 Standardkonformität

Die Standardkonformität, auch als Validität bezeichnet, bildet das Fundament für ein barrierefreies Umsetzen von Internetseiten. Standardkonformität wird durch das Verwenden aktueller Technologien und die Trennung von Inhalt und Layout sichergestellt und garantiert eine plattformübergreifende und zukunftssichere Darstellung.

#### 2.1.2 Zugänglichkeit

Unter Zugänglichkeit (**Accessibility**) werden unterschiedliche Nutzergruppen und Nutzungskontexte berücksichtigt. Aufgabe ist es, das Informationsangebot für möglichst viele Benutzergruppen, also z.B. auch blinde, sehbehinderte oder motorisch eingeschränkte Menschen, zugänglich zu machen.

#### 2.1.3 Gebrauchstauglichkeit

Gebrauchstauglichkeit (**Usability**) beschreibt Aspekte wie nachvollziehbare Navigation, klare Orientierungshilfen und konsistente Nutzerführung.

#### 2.1.4 Verständlichkeit

Die Dimension Verständlichkeit bezieht sich vor allem auf den inhaltlichen Aspekt und das Umsetzen in einfacher Sprache.

#### 2.1.5 Design

Die oftmals unterschätzte Rolle des Designs, inklusive der Gestaltung des Layouts, und dem damit verbundenen Bilden der Struktur beschreibt die fünfte Dimension.

---

1) BGBl 2002 b, S. 1468

2) Vgl. Bittner 2005 (S. 3)



## 2.2 Nutzerorientierte Gestaltung

Allen Nutzern werden äquivalente Inhalte zur Verfügung gestellt und vor allem die Bedürfnisse von Menschen mit Sehbeeinträchtigungen oder motorischen Behinderungen sowie von Personen, die im Umgang mit ihrem PC unsicher oder unerfahren sind, berücksichtigt.

Doch auch für Nutzer von Textbrowsern<sup>1</sup> oder mobilen Endgeräten (Handy, PDA<sup>2</sup>) ist die barrierefreie Gestaltung von Internetseiten von Vorteil.

### 2.2.1 Behinderungen

Nach dem Ergebnis des *Mikrozensus*<sup>3</sup> lebten im Mai 2003 in der Bundesrepublik Deutschland 8,4 Millionen amtlich anerkannte behinderte Menschen. Resultierend aus dem Ergebnis dieser Statistik ist im Durchschnitt jeder zehnte Einwohner der Bundesrepublik Deutschland anerkannt behindert und diese Nutzergruppe somit nicht zu vernachlässigen.<sup>4</sup>

Besonders hervorzuheben sind Sehbehinderungen, insbesondere die Blindheit, die schwerwiegendste Beeinträchtigung in Bezug auf das primär optisch zu erfassende Medium Internet.<sup>5</sup> Auch für Nutzer mit Sehschwächen oder Farbenblindheit ist die barrierefreie Gestaltung von Internetseiten eine Grundvoraussetzung für das uneingeschränkte Nutzen der Inhalte. Nutzer mit ausgeprägter Sehschwäche sind beispielsweise auf das Anpassen der Schriftgröße an persönliche Bedürfnisse angewiesen, wobei sichergestellt werden muss, dass das Layout auch nach einem Vergrößern der Schrift nicht zerstört oder unleserlich ist. Für von Farbenblindheit betroffene Nutzer ist zu beachten, dass bei einer Graustufenbetrachtung die Internetseite mit ausreichenden Kontrasten lesbar bleibt.

Auch Nutzer mit einer motorischen Behinderung können vor Problemen bei der Internetnutzung stehen. In Rücksicht auf diese Zielgruppe ist es besonders wichtig, Möglichkeiten der Tastatursteuerung anzubieten, so dass zum Beispiel Navigationselemente mit Hilfe der Tastatur auswählbar sind.

---

1) Textbrowser = Spezieller nicht-grafischer Browser, der Internetseiten in Textform zeilenorientiert darstellt.

2) PDA = Personal Digital Assistant. PC-Gerät in Westentaschen-Größe, das batterie- oder akkubetrieben vorwiegend von Berufstätigen unterwegs für das persönliche Informationsmanagement verwendet wird. (Vgl. Hansen/Neumann 2001, S. 65)

3) Mikrozensus = Die EU-weit größte Haushaltsbefragung und eine Mehrzweckstichprobe, die ausführliche Informationen über die wirtschaftliche und soziale Lage der Bevölkerung liefert. Die Beantwortung der Fragen zur Behinderung und Gesundheit ist freiwillig, 2003 haben 85% der Befragten Angaben zur Behinderung gemacht. (Vgl. Pfaff 2004, S. 1181)

4) Vgl. Pfaff 2004, S. 1181

5) Vgl. Richter 2004 (S. 2)

### 2.2.2 Screenreader

Blinde und stark sehbehinderte Menschen stehen bei der Internetnutzung vor technischen Problemen. Sie benötigen einen sogenannten **Screenreader**, eine Software, die es ihnen ermöglicht, die Inhalte der Internetseiten zu nutzen. Diese Software interpretiert Bildschirminformationen wie Texte, Grafiken oder den Bildschirmaufbau und **gibt diese Informationen an Hilfsmittel weiter**.<sup>1</sup> Da blinde Nutzer Grafiken nicht optisch erfassen können, sollte für diese Nutzergruppe beispielsweise für jede Grafik eine alternative Bildbeschreibung hinterlegt werden, die dann von einem Screenreader alternativ zur Grafik weitergegeben werden kann.

### 2.2.3 Braillezeile

Ein **Hilfsmittel**, mit dem Blinde die vom Screenreader ausgelesenen Informationen ertasten können, ist die **Braillezeile**, bei der Stifte entsprechend den Buchstaben der Blindenschrift hochgedrückt werden. Die Grundform bilden sechs Punkte, die in zwei senkrechten Reihen zu je 3 Punkten nebeneinander angeordnet und so optimal ertastbar sind.<sup>2</sup>

### 2.2.4 Sprachausgabe

Eine weiteres **Hilfsmittel** ist die **Sprachausgabe**, die den am Bildschirm sichtbaren Text, ausgelesen von einem Screenreader, akustisch wiedergibt. Für den Test der Sprachausgabe wird im Rahmen dieser Ausarbeitung der Screenreader *JAWS*<sup>3</sup> mit integrierter Sprachausgabe genutzt, der bei Einsatz eines Windows-Betriebssystems alle gängigen Anwendungen unterstützt und den Zugang zum Internet ermöglicht.

## 2.3 World Wide Web Consortium (W3C)

Das World Wide Web Consortium (W3C) wurde gegründet, um die Möglichkeiten des Internets zu erschließen und den Fortschritt innerhalb dieses Mediums durch die Entwicklung einheitlicher Technologien zu fördern und sicherzustellen.<sup>4</sup> Ein Zitat vom Direktor des W3C, Tim Berners-Lee, zeigt das Anliegen des Internet-Gremiums: „W3C is where the future of the Web is made. Our Members work together to design and standardize Web technologies that build on its universality, giving the power to communicate, exchange information, and to write effective, dynamic applications - for anyone, anywhere, anytime, using any device.“<sup>5</sup>

---

1) Vgl. Freudenfeld 2003 (S. 1)

2) Vgl. Lenk 2004 (S. 2)

3) JAWS = Job Access With Speech. Getestet wurde mit „JAWS for Windows“, Version 4.51.221. Als 40-Minuten-Demo zum Download verfügbar unter <http://www.freedomsci.de/serv01.htm>

4) Vgl. Birkenbihl 2005 (S. 1)

5) W3C 2004 (S. 1)

## 3 Grundlagen



### 3.1 Web Content Accessibility Guidelines 1.0 (WCAG1)

1999 veröffentlichte das *World Wide Web Consortium (W3C)* die *Web Content Accessibility Guidelines 1.0 (WCAG1)*. Die WCAG1 wurden von der *Web Accessibility Initiative (WAI)* des W3C erarbeitet und werden deshalb auch als *WAI-Richtlinien* bezeichnet. Die Richtlinien enthalten umfangreiche Empfehlungen (in englischer Sprache) für das Gestalten von Internetseiten und bilden die **Basis aller Vorlagen für barrierefreies Internet**. Das Befolgen der Richtlinien fördert die Zugänglichkeit der Internetseite und stellt allen Nutzern unabhängig von Zugangsort und Zugangsart äquivalente Inhalte zur Verfügung.<sup>1</sup>

#### 3.1.1 Zugänglichkeitsrichtlinien

Insgesamt bestehen die WCAG1 aus 14 Richtlinien (siehe Anhang A) mit insgesamt 65 Checkpunkten verteilt auf drei Prioritätsstufen (siehe Anhang B).

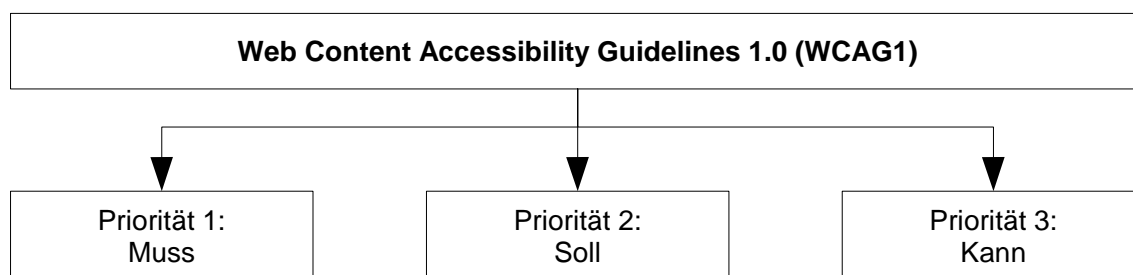


Abb. 1: Aufbau der Web Content Accessibility Guidelines 1.0

#### 3.1.2 Priorität 1

Priorität 1 beinhaltet sogenannte **Muss-Kriterien**, die grundlegende Erfordernisse abbilden, damit bestimmte Gruppen Web-Dokumente verwenden können.<sup>2</sup>

Priorität 1 beinhaltet beispielsweise

- das Bereitstellen eines Text-Äquivalents für jedes Nicht-Text-Element,
- die zusätzliche Kennzeichnung von mit Farbe dargestellter Information durch Hervorhebung oder andere Stilmittel,
- das Kenntlichmachen von Änderungen der natürlichen Sprache innerhalb des Dokumententextes,

1) Vgl. Hartmann 2002 (S. 2)

2) Vgl. Hartmann 2002 (S. 7f)

- die Lesbarkeit des Dokumentes auch ohne Stylesheets,
- das Vermeiden von Bildschirmflackern und
- das Formulieren der Inhalte in klarer und einfacher Sprache.

Des Weiteren beinhalten die Richtlinien und Checkpunkte der Priorität 1 Anforderungen zum Gestalten von Tabellen und Frames<sup>1</sup>: Das Kennzeichnen von Zeilen- und Spaltenüberschriften bei Datentabellen, die Zuordnung von Daten- und Überschriftenzellen sowie das Betiteln von Frames, um die Navigation und Identifikation zu erleichtern.

### 3.1.3 Priorität 2

Das Umsetzen der unter Priorität 2 erfassten sogenannten **Soll-Kriterien** beseitigt signifikante Hindernisse für den Zugriff auf Web-Dokumente.<sup>2</sup>

Priorität 2 fordert beispielsweise

- einen ausreichenden Kontrast von Vorder- und Hintergrundfarbe,
- die Angabe von relativen anstelle von absoluten Einheiten,
- das Strukturieren eines Dokumentes durch Überschriften-, Listen- und Absatz-Elemente,
- die Auszeichnung von Zitaten,
- die Beschriftung von Formularfeldern,
- das Bereitstellen von Textalternativen für dynamische Inhalte und
- die Identifizierung von Verweis-Zielen.

Weiterhin wird unter Priorität 2 das Verwenden von Stylesheets und W3C-Technologien, das Hinzufügen semantischer Informationen über die Einbindung von Metadaten, das Bereitstellen eines Inhaltsverzeichnisses sowie die konsistente Verwendung von Navigationsmechanismen erfasst.

Außerdem sind Bilder zur Informationsdarstellung, blinkende Elemente, automatische Aktualisierungen, Weiterleitungen, das Erscheinen neuer Fenster, überholte Elemente, Layouttabellen sowie Frames zu vermeiden und die Funktionsweise der Internetseiten bei abgeschalteten Skripten sicherzustellen.

---

1) Frames = Bereiche, die das Browserfenster in verschiedene Segmente unterteilen (siehe Kapitel 8.1).

2) Vgl. Hartmann 2002 (S. 8)

### 3.1.4 Priorität 3

Priorität 3 umfasst sogenannte **Kann-Kriterien**. Das Umsetzen der Richtlinien und Checkpunkte dieser Priorität erleichtert den Zugriff auf Web-Dokumente.<sup>1</sup>

Dazu müssen beispielsweise

- Abkürzungen und Akronyme bei ihrem ersten Auftreten im Text spezifiziert,
- die vorherrschende natürliche Sprache des Dokumentes kenntlichgemacht,
- eine logische Tabulator-Reihenfolge definiert,
- Tastatur-Kurzbefehle bereitgestellt,
- nebeneinander angeordnete Verweise durch Zeichen wie „|“ eindeutig getrennt und
- Informationen über die Zusammenhänge verschiedener Dokumente eingebunden werden.

Informationen sind zu gruppieren und in einem konsistenten Präsentationsstil darzustellen. Wenn es das Verständnis der Seite erleichtert, sind für Texte grafische oder akustische Präsentationen zu ergänzen. Zur Erfüllung der Priorität 3 müssen Tabellen jeweils eine Zusammenfassung und lange Überschriftenzellen eine Abkürzung enthalten.

### 3.1.5 Konformitätsstufen

Für Anbieter von Internetseiten hat das W3C Konformitätslogos entwickelt, mit denen der Umsetzungsgrad der WCAG1 angezeigt werden kann (A = Priorität 1, AA = Priorität 2, AAA = Priorität 3). Der Gebrauch dieser Logos auf Seiten, die die Richtlinien und Checkpunkte erfüllen, soll das **Bewusstsein für Barrierefreiheit steigern**.



Abb. 2: Konformitätsstufen (Quelle: <http://www.w3.org/WAI/WCAG1-Conformance.html>)

Barrierefreie Internetseiten verbessern für die Einen die Nutzbarkeit, für andere Nutzergruppen sind sie Voraussetzung, überhaupt mit dem Medium Internet arbeiten zu können.<sup>2</sup> Das Umsetzen der WCAG1 fördert die Barrierefreiheit im Internet. Für Unternehmen und Privatpersonen ist die barrierefreie Gestaltung von Internetseiten freiwillig. Doch auch im Internet gewinnt das Thema Gleichstellung an Bedeutung. Wie der nächste Abschnitt zeigt, sind öffentliche Institutionen demnächst zum Erstellen eines barrierefreien Internetangebotes verpflichtet.

1) Vgl. Hartmann 2002 (S. 8)

2) Vgl. Hellbusch 2005 a, S. 5

### 3.2 Barrierefreie Informationstechnik-Verordnung (BITV)

Aufbauend auf das Behindertengleichstellungsgesetz wurde am 17.07.2002 die *Verordnung zur Schaffung barrierefreier Informationstechnik nach dem Behindertengleichstellungsgesetz (Barrierefreie Informationstechnik-Verordnung - BITV)* vom Bundesministerium herausgegeben.

In der Anlage<sup>1</sup> enthält die BITV nach BGG §§ 3 und 4 Abs.1 eine Liste mit Anforderungen. Im direkten Vergleich mit der deutschen Übersetzung der WCAG1 von René Hartmann werden „Richtlinien“ (siehe Anhang A) zu „Anforderungen“ (erste Gliederungsebene) und „Checkpunkte“ (siehe Anhang B) zu „Bedingungen“ (zweite Gliederungsebene). Zusätzlich sind die BITV-Anforderungen auf nur zwei statt drei Prioritäten beschränkt (siehe Anhang C).

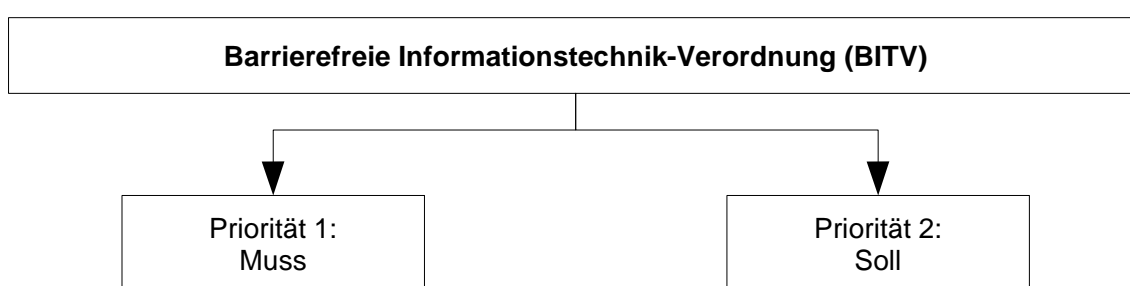


Abb. 3: Aufbau der Barrierefreie[n] Informationstechnik-Verordnung

Priorität 1 beschreibt die Anforderungen, die für eine barrierefreie Internetseite erfüllt werden müssen und definiert unerlässliche Mindestanforderungen. Priorität 2 beinhaltet Anforderungen, die erfüllt werden sollten oder auch müssen, wenn es sich um zentrale Einstiegsangebote handelt.<sup>2</sup>

Der große **Vorteil der WCAG1** gegenüber der BITV liegt in der Untergliederung in drei Prioritäten, was den **stufenweisen Ausbau** einer barrierefreien Internetseite besser ermöglicht.<sup>3</sup>

Der **Vorteil der BITV** gegenüber der WCAG1 liegt in der Rechtsverbindlichkeit, nach der **öffentliche Institutionen verpflichtet** sind, bis Ende 2005 ihre Internetauftritte und -angebote behindertengerecht umzugestalten und somit die Benachteiligung behinderter Menschen zu beseitigen.<sup>4</sup>

Letztendlich **basieren die BITV-Vorgaben auf den WCAG1-Vorgaben** und sind ähnlich formuliert.

1) Vgl. BGBl 2002 a, S. 2655ff

2) Vgl. Bühler 2004 (S. 2)

3) Vgl. Bittner 2004 (S. 3)

4) Vgl. BGBl 2002 a, S. 2654

## 4 Technologien



### 4.1 Aktuelle Technologien

Die Verwendung aktueller W3C-Technologien ist aus drei Gründen empfehlenswert<sup>1</sup>:

- W3C-Technologien beinhalten bereits Zugänglichkeitsaspekte,
- berücksichtigen Fragen der Zugänglichkeit schon in der Design-Phase und
- werden in einem offenen, auf Industrie-Konsens basierenden Prozess entwickelt.

Im Gegensatz zu Nicht-W3C-Formaten wie beispielsweise PDF<sup>2</sup> sind keine Plug-Ins<sup>3</sup> oder eigenständige Anwendungen zum Betrachten erforderlich. Das Vermeiden proprietärer Technologien verbessert die Zugänglichkeit von Internetseiten. W3C-Formate ermöglichen das Betrachten oder Navigieren mit Standard-Benutzeragenten<sup>4</sup> und zeichnen sich durch die schnelle und korrekte Interpretation seitens der Browser, Handys, PDAs und Hilfsmitteln wie der Braillezeile aus.

Das Umsetzen von Internetseiten war bisher durch die Verwendung der *Hypertext Markup Language* (HTML) und *Cascading Stylesheets* (CSS) geprägt. Als zukunftsorientiert gilt die **Extensible Hypertext Markup Language** (XHTML). XHTML ist eine durch das W3C entwickelte Neuformulierung von HTML basierend auf der *Extensible Markup Language* (XML), einer Beschreibungssprache für Strukturen und Datenformate, die dazu dient proprietäre Formate abzulösen.

Resultierend aus der XML-Basis sind XHTML-Dateien wohlgeformt und valide zu gestalten. **Wohlgeformte Dokumente** sind nach syntaktischen Regeln aufgebaut, beispielsweise müssen alle Elemente korrekt ineinander geschachtelt, mit einem abschließenden Element beendet oder explizit durch eine Endmarkierung geschlossen werden. **Valide Dokumente** enthalten zusätzlich ausschließlich die in einer Dokumenttypdefinition (Document Type Definition) anwendungsspezifisch definierten Elemente und Markierungen.<sup>5</sup>

---

1) Vgl. Hartmann 2002 (S. 19)

2) PDF = Portable Document Format. Von Adobe entwickeltes, weit verbreitetes Format für den Austausch von unveränderlichen elektronischen Dokumenten. (Vgl. Hansen/Neumann 2001, S. 382)

3) Plug-In = Hilfsprogramm. Browser benötigen diverse Plug-Ins um Multimediainhalte darzustellen.

4) Benutzeragent = Software zum Zugriff auf Internetseiten; dies umfasst beispielsweise Browser, Mobiltelefone, Screenreader und Spracherkennungssoftware. (Vgl. Hartmann 2002 (S. 27))

5) Vgl. Hansen/Neumann 2001, S. 1045f

## 4.2 Extensible Hypertext Markup Language (XHTML)

### 4.2.1 Deklaration und Definition

XHTML-Dokumente beinhalten

- eine XML-Deklaration,
- die Einbindung einer Dokumenttypdefinition und
- die Angabe des XML-Namensraumes.

Zur expliziten Kennzeichnung als XHTML-Dokument wird eine spezielle Kopfzeile (`<?xml ... ?>`) eingefügt, in der zum einen die zugrundeliegende **XML-Version** (`version="1.0"`) und zum anderen die Kennung des benutzten Zeichensatzes (`encoding="UTF-8"`)<sup>1</sup> festgelegt werden.

Über eine weitere Kopfzeile (`<!DOCTYPE ...>`) wird der Name des Wurzelementes (`html`) genannt und die zu verwendende **Document Type Definition** (DTD) eingebunden.

Die DTD ist eine Textdatei mit der Endung `.dtd`, enthält die zur Validität benötigten Element- und Attributdeklarationen und definiert die Struktur eines Dokumentes: Die Reihenfolge, in der Elemente und Attribute auftreten und wie diese verschachtelt werden dürfen.<sup>2</sup>

In XHTML gibt es drei mögliche DTD-Angaben:

- Strict
- Frameset
- Transitional

Die **Strict-DTD** definiert eine „normale“ XHTML-Datei und wird für Inhaltsseiten verwendet. Mit dem Einbinden der **Frameset-DTD** wird eine mehrere Frames enthaltende Datei definiert. Die **Transitional-DTD** ist eine Verbindung der beiden vorgenannten DTDs, definiert sowohl Inhalts- als auch Frameset-Elemente und stellt eine Übergangslösung dar. Empfehlenswert und vor allem zukunftsorientiert sind die beiden erstgenannten Varianten *Strict* und *Frameset*.

Im öffnenden `<html>`-Element wird über das `xmlns`-Attribut der **XML-Namensraum** angegeben, die Sprachangabe des Dokumentes im `lang`-Attribut ist optional.

1) UTF-8 = Unicode Transformation Format. Zeichensatz mit variabler Zeichenlänge und der Fähigkeit, die international verwendeten Schriften repräsentieren zu können. (Vgl. Hansen/Neumann 2001, S. 1007)

2) Vgl. Hansen/Neumann 2001, S. 1046



```

Beispiel: <?xml version="1.0" encoding="UTF-8"?>
          <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
          "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
          <html xmlns="http://www.w3.org/1999/xhtml" lang="de" xml:lang="de">
          <head>...</head>
          <body>
          ...
          </body>
          </html>

```

XHTML-Dokumente können wie HTML-Dokumente als *.htm* und *.html*-Dateien gespeichert werden. Auch die Endung *.xhtml* kann verwendet werden; besonders zu beachten ist dann, dass neuere, XML-fähige Browser (beispielsweise der Internet Explorer ab Version 5) wenig fehlertolerant agieren und die Aufbereitung eines Dokumentes bei Auftreten von Fehlern gegebenenfalls abbrechen.<sup>1</sup>

#### 4.2.2 Kopfbereich und Textkörper

XHTML-Dokumente bestehen weiterhin generell

- aus einem Kopfbereich (*<head>*-Element) und
- einem Textkörper (*<body>*-Element).

Beide Bereiche werden, wie schon in HTML üblich, in das *<html>*-Element eingebettet (siehe vorangegangenes Beispiel).

Im Unterschied zu Notationen in HTML ist zu beachten, dass

- alle Inhalte des Textkörpers in Blockelemente wie beispielsweise Absätze (*<p>*-Element) und Überschriften eingebettet werden müssen,
- Elemente, Attribute und Attributwerte immer in Kleinbuchstaben dargestellt werden,
- Elemente, die kein Abschlusselement beinhalten, durch einen Schrägstrich (*<br />*) beendet werden und
- Attribute immer in Anführungszeichen (*height="..."*) stehen.

Alle Elemente werden korrekt verschachtelt, unsauber geschachtelte Elemente, wie sie in HTML-Notation möglich waren, verursachen einen Fehler.

1) Vgl. Münz/Nefzger 2002, S. 388f

Beispiel:     `<p>Das ist ein <strong>verschachtelter Absatz</strong>.</p>`  
                   `<p>Und jetzt folgt ein Bild:<br />`  
                   `</p>`

Mögliche HTML-Notation:     `<p>...<strong>...</p></strong>`

Das Konzept von XHTML wird ständig weiterentwickelt. Elemente, die **der Trennung von Inhalt und Gestaltung entgegenstehen**, werden aus der Spezifikation entfernt und gelten als veraltet („deprecated“, auch übersetzt mit „missbilligt“ oder „überholt“).

Dazu zählen die Elemente:<sup>1</sup>

<code>&lt;basefont&gt;</code>	<code>&lt;center&gt;</code>	<code>&lt;dir&gt;</code>
<code>&lt;font&gt;</code>	<code>&lt;isindex&gt;</code>	<code>&lt;menu&gt;</code>
<code>&lt;s&gt;</code>	<code>&lt;strike&gt;</code>	<code>&lt;u&gt;</code>

Das in HTML oft verwendete `<marquee>`-Element, das eine Laufschrift erzeugt sowie das `<blink>`-Element werden zwar von einigen Browsern unterstützt, sind jedoch nicht standardisiert.<sup>2</sup>

Das `<font>`-Element beispielsweise, das in HTML zum Formatieren von Schriften innerhalb eines Dokumentes genutzt wurde, entfällt durch das Trennen von Inhalt und Layout und wird, ebenso wie alle anderen veralteten Elemente, über externe Formatierungen in Stylesheet-Dateien realisiert.

### 4.3 Cascading Stylesheets (CSS)

Das Gestalten und Positionieren der Elemente einer Internetseite wird über Cascading Stylesheets (CSS, oder auch: Stylesheets) festgelegt.

Sinn des Einsatzes von Stylesheets ist **das klare Trennen von Inhalt und Layout**.

Über Stylesheets hat der Entwickler von Internetseiten außerhalb des eigentlichen Inhaltes genaue Kontrolle über beispielsweise Zeichenabstände, Textausrichtung, Objektpositionen, Audio- und Sprachausgabe und Schriftcharakteristiken.<sup>3</sup>

1) Vgl. Mintert et al. 2003, S. 426f

2) Vgl. Hellbusch 2005 a, S. 178

3) Vgl. Meiert 2004 (S. 1f)

Das Extrahieren aller Präsentations-Informationen aus dem Inhalt macht die Internetseite zugänglicher für ein breites Spektrum von Geräten, erleichtert umfangreiche Änderungen, schafft kleinere Dateien und ermöglicht ein bedürfnisorientiertes Anpassen.

#### 4.3.1 Stylesheet-Ebenen

Es existieren drei Stylesheet-Ebenen, die kaskadenartig ineinander verschachtelt werden können:<sup>1</sup>

- Browser-Stylesheet,
- User-Stylesheet und
- Autoren-Stylesheet.

Das **Browser-Stylesheet** orientiert sich an den Grundangaben von (X)HTML: Ein Dokument, dem keine CSS-Datei zugewiesen ist, wird trotzdem in jedem Browser dargestellt. Standardmäßig wird beispielsweise die Schriftfarbe auf schwarz und die Hintergrundfarbe auf weiß gesetzt, Überschriften werden größer dargestellt, Listen eingerückt.

Das **User-Stylesheet** ermöglicht ein bedürfnisorientiertes Anpassen. Das User-Stylesheet ist im Normalfall nicht definiert, bietet aber Benutzern die Möglichkeit ein eigenes Stylesheet einzurichten und beispielsweise für alle Elemente eine größere Schrift oder andere Farbuweisungen zu definieren.

Das **Autoren-Stylesheet** gibt den Entwicklern die Möglichkeit, verwendete Elemente zu überschreiben, neu zu definieren und mit speziellen Angaben zu ergänzen. Grundeigenschaften des Browser-Stylesheets, die nicht neu definiert werden, bleiben weiterhin erhalten.

Wichtig ist auch, das Kaskaden-Konzept in unterschiedlichen Element-Ebenen zu beachten. Das Verschachteln von Elementen bewirkt, dass innere Elemente die Eigenschaften der äußeren Elemente übernehmen. Werden den inneren Elementen eigene Zuweisungen zugeteilt, ist diese Vererbung aufgehoben: die innerste CSS-Eigenschaft ist stärker als die umgebende.

---

1) Vgl. Jenzer 2004 (S. 1)

### 4.3.2 Verschiedene Formatvorlagen

Unter Zuhilfenahme von Stylesheets können für unterschiedliche Ausgabemedien auch unterschiedliche Layouts festgelegt werden. Das Zuweisen verschiedener CSS-Dateien erfolgt über das **media**-Attribut im `<link>`-Element.<sup>1</sup>

Mögliche Angaben sind:

- *all* für alle Medientypen,
- *aural* für computergesteuerte, synthetische Sprachausgabe,
- *braille* für Ausgabegeräte mit Braille-Zeile,
- *embossed* für Braille-Drucker,
- *handheld* für die Anzeige auf tragbaren Kleincomputern,
- *print* für den Ausdruck auf Papier,
- *projection* für die Datenprojektion mit Beamern,
- *screen* für die Bildschirmanzeige,
- *tty* für nicht-grafische Ausgabemedien und
- *tv* für TV-ähnliche Ausgabegeräte.

Das Zuweisen unterschiedlicher Formatvorlagen ermöglicht beispielsweise das Ausblenden von Bildern oder Werbebannern in der Druckvariante (`media="print"`).

Im nachfolgenden Beispiel wird der Absatz `<p class="banner">...</p>` durch die Anweisung `"display:none"` in der speziell angelegten CSS-Datei beim Druck nicht mehr angezeigt, in der grafischen Ansicht bleibt dieser Absatz aber weiterhin sichtbar.

Beispiel:      `<link rel="stylesheet" type="text/css" href="print.css" title="Druckansicht" media="print" />`  
                  `<link rel="stylesheet" type="text/css" href="format.css" title="Standard" media="screen" />`

Quelltext (Ausschnitt):      `<p class="banner">`  
                                  ``  
                                  `Bachelor-Thesis<br />`  
                                  `Testseite Formatvorlagen<br />`  
                                  `Christiane Müller</p>`  
                                  `<h1>Barrierefreie Gestaltung von Internetseiten</h1>`

**print.css:**      `p.banner { display:none; }`

1) Vgl. Münz/Nefzger 2002, S. 408ff

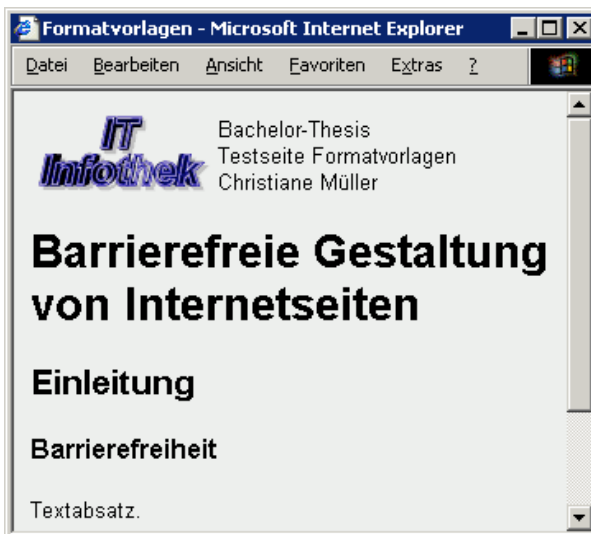


Abb. 4: Browser-Ansicht mit Formatvorlage

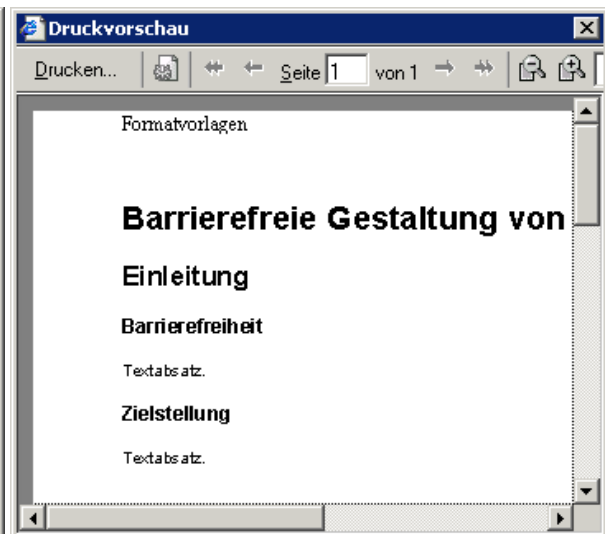


Abb. 5: Druckvorschau mit Formatvorlage

Einige Browser ermöglichen das Umschalten zwischen verschiedenen Formatvorlagen direkt in der grafischen Anzeige. Dazu muss im `<link>`-Element noch zusätzlich das `title`-Attribut eingebunden werden.

Im Mozilla sind die Formatvorlagen über „Ansicht – Style verwenden“, im Opera über „Ansicht – Seitendarstellung“ erreichbar.

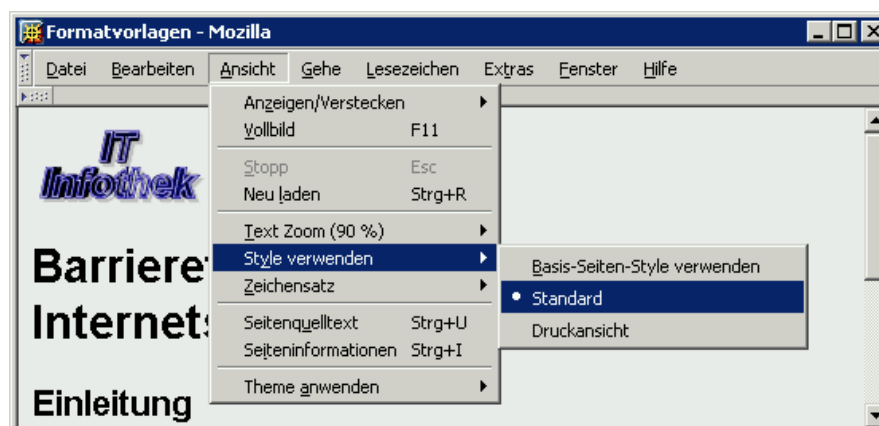


Abb. 6: Wechsel zwischen Formatvorlagen im Mozilla

Durch das Einbinden verschiedener Formatvorlagen können nutzergruppenorientierte Layouts bereitgestellt werden. Die Möglichkeit, zwischen verschiedenen Formatvorlagen zu wechseln, wird aufgrund der fehlenden Unterstützung seitens der Browserhersteller in der Praxis oft skriptbasiert angeboten. Dadurch entstehende Barrieren werden nachfolgend erläutert.

#### 4.4 Dynamische Inhalte

Die Bedienbarkeit einer Internetseite bei Abschalten von Skripten wird sowohl in den WCAG1- als auch in den BITV-Richtlinien mit Priorität 1 gekennzeichnet. Screenreader haben Schwierigkeiten mit Skripten, ältere Browser können Skripte unter Umständen nicht verarbeiten und in vielen Institutionen sind Skripte aus Sicherheitsgründen deaktiviert.

Diese Einschränkungen betreffen vor allem den verbreiteten Einsatz von **JavaScript**, einer von Netscape entwickelte Skriptsprache, um Internetseiten mit dynamischen Inhalten ausstatten zu können.<sup>1</sup> Das Erzeugen einer barrierefreien Internetseite schließt den Einsatz von JavaScript nicht aus, zu vermeiden sind jedoch Befehle, die eine Benutzung der Maus verlangen oder Befehle zum Öffnen neuer Fenster, so dass bestimmte Ausgabegeräte diese nicht mehr anzeigen können.<sup>2</sup>

Dynamische Inhalte werden auf Internetseiten auch häufig durch Flash-Filme realisiert. **Flash** ist ein Programm von Macromedia zum Erstellen von Animationen und interaktiven Anwendungen im Internet. Zur Darstellung im Browser ist die Installation eines speziellen Flash-Player-Plug-Ins notwendig.<sup>3</sup>

Auch das Einbinden von Flash-Filmen in eine barrierefreie Internetseite ist aufgrund der Bemühungen des Herstellers Macromedia, behindertengerechte Funktionen anzubieten, zwar nicht ausgeschlossen, führt aber weiterhin zu Schwierigkeiten, da diese Funktionen von den Hilfsmitteln nur begrenzt unterstützt werden.<sup>4</sup>

Für Browser, die keine Skripte unterstützen und für alle Nutzer, die Skripte deaktiviert haben, kann über das `<noscript>`-Element alternativer Inhalt bereitgestellt werden. Dieses Element ist direkt im Anschluss an das `<script>`-Element zu notieren und kann XHTML-Formatierungselemente enthalten.

```
Beispiel: <script src="gebaeude.js"></script>
          <noscript>
          <p>Die Animation zeigt die Entstehung unseres Hauses von der ersten
          Vermessung bis zum Richtfest.</p>
          </noscript>
```

1) Vgl. Hansen/Neumann 2001, S.955f

2) Vgl. Hein 2004 (S. 1)

3) Vgl. Münz/Nefzger 2002, S. 89

4) Vgl. Hellbusch 2005 a, S. 237

## 5 Grafiken und Farbe



### 5.1 Schriftgrafiken

Der größte Vorteil, den Internetseiten gegenüber Print-Medien haben, ist die variable Form. Texte können in Größe, Farbe und Gestalt angepasst oder von spezieller Software vorgelesen werden.<sup>1</sup> Texte, die zu Grafiken umgewandelt wurden, werden als Schriftgrafiken bezeichnet.

Schriftgrafiken werden meist eingesetzt, um das Corporate Design<sup>2</sup> zu wahren. Für das Anfertigen einer barrierefreien Internetseite ist diese Anwendung umstritten. Einige Autoren verwenden Bilder, um Text in einer bestimmten Schriftart darzustellen, deren Verfügbarkeit nicht auf allen Nutzeragenten gewährleistet ist. Text, der als Grafik dargestellt wird, kann weder von Spezialsoftware wie Screenreadern noch von Suchmaschinen erfasst werden.<sup>3</sup> Schriftgrafiken können nicht an Benutzeranforderungen angepasst werden, da Farben und Schriftgrößen in Grafiken nicht individuell eingestellt werden können.

Schriftgrafiken und Symbole werden häufig für Navigationselemente und Schaltflächen verwendet, sogenannte Bedienelemente. Da für blinde Nutzer oder für Benutzer von einfachen Textbrowsern diese Grafiken nicht zugänglich sind, müssen Alternativtexte hinterlegt werden. Alternativtexte sollen den Text des Elementes wiedergeben und somit das Grafikelement ersetzen.

Oftmals sind neben dem grafischen Verweis gleichwertige Textverweise angeordnet. In diesem Fall ist ein leeres *alt*-Attribut zu notieren, da das grafische Navigationselement dann nur dekorativen Charakter hat.

Beispiel:	a) <code>&lt;a href="kontakt.html"&gt;&lt;img src="briefkasten.gif" alt="Kontakt" /&gt;&lt;/a&gt;</code> b) <code>&lt;a href="kontakt.html"&gt;&lt;img src="briefkasten.gif" alt="" /&gt; Kontakt&lt;/a&gt;</code>
JAWS:	a) „Link, Grafik Kontakt“ b) „Link, Kontakt“

Eine weitere Ausnahme ist das Verwenden von Schrift im Firmenlogo. Solange dieses nicht als Überschrift oder Bedienelement eingesetzt wird, ist das Einbinden von Schriften nicht untersagt.

1) Vgl. DIAS 2004 (S. 3)

2) Corporate Design = Ein optisch einheitliches Bild durch Schrift, Farbe und Formen in allen Dateien und Projekten schaffen, beispielsweise mit einem Logo, grafischen Symbolen und Textformaten. (Vgl. Münz/Nefzger 2002, S. 1148f)

3) Vgl. Meiert 2004 (S. 2)

## 5.2 Informative Grafiken

Ebenso wie für grafische Bedienelemente ist es für informative Grafiken wichtig, einen Alternativtext zu hinterlegen. Informative Grafiken unterstützen den Inhalt und das Thema der Seite. Für Schriften ist der entsprechende Text zu notieren, für Symbole und Logos ist deren Bedeutung wiederzugeben, Fotos sollten knapp beschrieben werden. Für ausführlichere Beschreibungen ist das *longdesc*-Attribut zu nutzen. Dieses kann den Namen einer Datei enthalten, in der dann wiederum das Bild umfangreich beschrieben werden kann.

Das *longdesc*-Attribut wird bisher kaum unterstützt; Browser könnten aber beispielsweise einen Verweis zu dieser Datei anbieten, wenn die Grafik nicht anzeigbar ist.<sup>1</sup>

Beispiel:	<code>&lt;img src="bild1.gif" alt="Die höchsten Gebäude der Welt" longdesc="gebaeude.html" /&gt;</code>
JAWS:	„Grafik Die höchsten Gebäude der Welt, Runde Klammer auf, Drücken Sie Eingabe für lange Beschreibung, Runde Klammer zu“

Die Bedeutung eines Bildes kann auch im Kontext näher beschrieben werden. Wichtig ist, dass die Beschreibung dem Bild eindeutig zugeordnet werden kann. In diesem Fall ist im Alternativtext des Bildes nur eine kurze Bezeichnung des im Kontext näher beschriebenen Gegenstandes notwendig.

Farbblinde und sehbehinderte Nutzer können Farben nur eingeschränkt wahrnehmen, so dass Informationen, die ausschließlich über die Farbe vermittelt werden, noch anderweitig hervorgehoben werden müssen. Verschiedenfarbige Linien in Grafiken können beispielsweise zusätzlich über die Linienart unterschieden werden. Informative Grafiken müssen desweiteren auch vor wechselndem Hintergrund erkennbar sein, was beim Setzen von anwenderspezifischen Hintergrundfarben teilweise nicht gegeben ist. Grafiken mit transparentem Hintergrund sind meist für helle Hintergrundfarben optimiert und auf schwarzem Bildschirm nicht mehr erkennbar.

Ein weiterer wichtiger Aspekt im Bereich von Grafiken ist der Einsatz animierter Bilder. Flackernde, blinkende oder sich bewegende Elemente ziehen die Aufmerksamkeit des Benutzers auf sich. Das kann sowohl zu Konzentrationsschwierigkeiten führen als auch bei Nutzern mit photosensitiver Epilepsie Anfälle auslösen.<sup>2</sup> So lange der Nutzer keine Möglichkeit hat, solche Elemente abzustellen, sind diese zu vermeiden.

1) Vgl. Münz/Nefzger 2002, S. 218

2) Vgl. Hartmann 2002 (S. 16)



### 5.3 Layoutgrafiken

Für das Verwenden von Layoutgrafiken, welche ausschließlich ausschmückenden Charakter haben, gibt es zwei Anwendungsbereiche. Einerseits können Bilder eingesetzt werden, um einen Text visuell zu unterstützen. Genaue Informationen können aus einer solchen Grafik nicht entnommen werden und es reicht, die Grafik kurz und allgemein zu beschreiben.

Beispiel: Ein wissenschaftlicher Text über Pflanzen, die Natur und deren Umgebung. Das Bild einer Pflanze bessert den Text grafisch auf.

```
<p> Text über Pflanzen.</p>
```

JAWS: „Grafik eine Pflanze, Text über Pflanzen“

Anders ist es bei Layoutgrafiken wie Abstandshaltern, Farbflächen oder Mustern. Diese Grafiken haben keinen Informationsgehalt, benötigen daher auch keinen Alternativtext und sind mit einem leeren *alt*-Attribut auszuzeichnen.

Für Nutzer von Screenreadern ist das leere *alt*-Attribut besonders wichtig. Würde gar kein *alt*-Attribut notiert sein, liest der Screenreader den Namen des Bildes vor. Das kann zu endlosen Listen mit überflüssigen Bilddateinamen führen und den Nutzer erheblich beeinträchtigen. Ist in diesen Fall ein leerer Alternativtext notiert, wird das Bild als Layoutgrafik erkannt und übersprungen.<sup>1</sup>

Beispiel: Ein wissenschaftlicher Text über Pflanzen, die Natur und deren Umgebung. Diesmal dient das Bild einer Pflanze als Abstandshalter ohne Informationsgehalt.

```
<p> Text über Pflanzen.</p>
```

JAWS: „Text über Pflanzen“

1) Vgl. DIAS 2004 (S. 2)

## 6 Schrift und Sprache



### 6.1 Schrift

In Bezug auf die barrierefreie Gestaltung von Internetseiten ist es vor allem für ältere und leicht sehbehinderte Nutzer wichtig, Schriftgrößen bedürfnisorientiert einstellen zu können. Schriftgrößen können in absoluten und in relativen Werten angegeben werden. Obwohl die meisten Browser auch Schriften skalieren, für die absolute Größen vorgegeben sind, ist die Angabe von Variablen, also relativen Schriftgrößen empfehlenswert, da der Internet Explorer als der meistgenutzte Browser das Skalieren von absoluten Größen nicht ermöglicht.

Relative Größenangaben sind in den Maßeinheiten „em“ (bezogen auf die Schriftgröße des Elementes, 1.2em entspricht 120% des Elternelementes) und „%“ (Prozent) möglich. Die Entscheidung, Größenangaben in em oder % zu hinterlegen, ist Geschmackssache.

#### 6.1.1 Textkörper

Empfehlenswert ist jedoch, die **Schriftgröße für den <body>-Bereich** eines Dokumentes in 100.01% festzulegen. Ältere Browser-Versionen von Opera und Safari haben mit Angaben wie 100% und 101% Schwierigkeiten, die Angabe von 1em führt im Internet Explorer bei Skalierungen zu unverhältnismäßig kleiner (siehe nachfolgende Grafik, rechte Spalte) bzw. größerer Schrift.

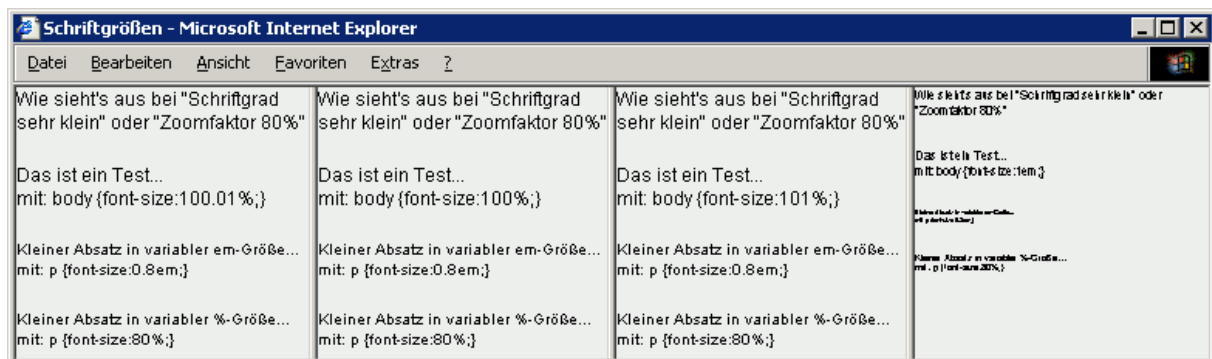


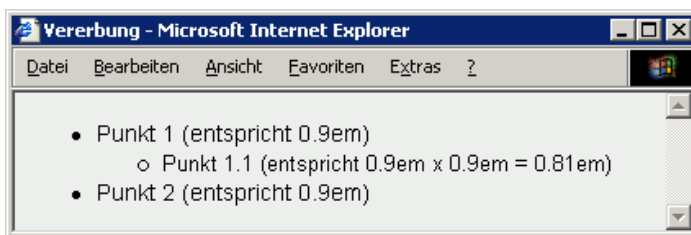
Abb. 7: Schriftgrößen im Internet Explorer

### 6.1.2 Vererbungsprinzip

Um den Stil der Schrift zu beeinflussen und das veraltete `<font>`-Element zu eliminieren, wird das CSS-Attribut `font` verwendet. Über dieses können Angaben zu Schriftart, Schriftgröße und Schriftgewicht eingebunden werden.

```
CSS-Datei:  ul { font:arial 0.9em normal; }
```

Wenn skalierbare Schriften verwendet werden, ist besonderes Augenmerk auf das CSS-Vererbungsprinzip zu richten.



Hat ein XHTML-Element eine skalierbare Schriftgröße zugewiesen bekommen, gilt diese auch für alle darin enthaltenen Elemente.<sup>1</sup>

Abb. 8: Vererbungsprinzip

```
Beispiel:  <ul>
            <li>Punkt 1 (entspricht 0.9em)
            <ul>
            <li>Punkt 1.1 (entspricht 0.9em x 0.9em = 0.81em)</li>
            </ul></li>
            <li>Punkt 2 (entspricht 0.9em)</li>
            </ul>
```

Die innere Liste erbt in diesem Fall die Schriftgröße der äußeren Liste und wird zusätzlich noch auf 0.9em verkleinert, was letztendlich zu einer realen Schriftgröße von 0.81em führt.

1) Vgl. Hellbusch 2005 a, S. 149

## 6.2 Sprache

Insbesondere für lernbehinderte, leseschwache und fremdsprachige Nutzer ist es vorteilhaft, wenn allgemein gebräuchliche, einfache, kurze Wörter und Sätze verwendet werden.

Eine Untersuchung zum Thema „Nutzungsfreundliche Bezeichnung von Navigationselementen im Internet“<sup>1</sup> zeigt Tendenzen. 46% der Nutzer erwarten, dass die erste Seite eines Internetangebotes als „Startseite“ bezeichnet wird. Alternativ wünschen sich junge Nutzer den Begriff „Home“. Der Begriff „Download“ wird für eine Schaltfläche zum Speichern von Programmen oder Informationen auf der eigenen Festplatte erwartet, Anfänger tendieren zu der Bezeichnung „Auf Festplatte speichern“. Die Möglichkeit, Seiten nach einem bestimmten Begriff zu durchsuchen, wird mit der Bezeichnung „Suche“ (41%) oder „Suchen“ (28%) bevorzugt. Für die Möglichkeit, im Internet Waren oder Dienstleistungen zu bestellen, sollte die Bezeichnung „Online-Shop“ (67%) verwendet werden. Auch für das Anmelden in einem geschützten, nur für registrierte Mitglieder zugänglichen Bereiches wird die englische Bezeichnung bevorzugt: Der deutsche Begriff „Anmelden“ landet mit 11% weit hinter der Bezeichnung „Login“ (56%).

Ein Internetauftritt, der nicht allgemein ausgerichtet ist (z.B. von Universitäten), sollte zumindest auf der Startseite in allgemein verständlicher Weise präsentieren, welches Angebot bereitgestellt wird.

### 6.2.1 Sprachwechsel

Der Wechsel der Sprache innerhalb eines Dokumentes ist durch das *lang*-Attribut kenntlich zu machen. Die Angabe *xml:lang=“...“* stellt die XHTML-konforme Schreibweise dar, für ältere Browser ist die Angabe *lang=“...“* ergänzend notwendig. Das Kennzeichnen von Sprachwechseln ist besonders für Nutzer von Screenreadern wichtig, da Screenreader über interne Wortlisten die Aussprache der Wörter festlegen.

Sprachwechsel können beispielsweise durch Fremdwörter oder fremdsprachige Zitate entstehen.

Beispiel:	Barrierefreiheit und <code>&lt;span xml:lang="en" lang="en"&gt;Accessibility&lt;/span&gt;</code>
-----------	--

1) Vgl. eResult 2005 (S. 7ff)

Innerhalb eines Fließtextes können Bereiche, die von einem Sprachwechsel betroffen sind, in das `<span>`-Element, ein sogenanntes **Inline-Element**<sup>1</sup>, eingebettet werden.

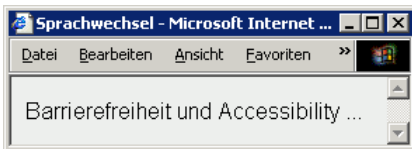


Abb. 9: Sprachwechsel

Das `<span>`-Element kann Text (hier: „Accessibility“) und andere Elemente enthalten, hat selbst aber keinerlei Eigenschaften und beeinflusst somit auch nicht das Layout.<sup>2</sup>

### 6.2.2 Zitate

Zitate sind durch die Elemente `<q>` oder `<blockquote>` zu kennzeichnen.

- `<blockquote>` erzeugt einen Absatz und wird in den meisten Browsern eingerückt dargestellt.
- `<q>` hingegen ist ein Inline-Element, muss zusätzlich in einen Absatz eingebettet werden und wird weder von Screenreadern noch vom Internet Explorer unterstützt.

Browser wie Mozilla und Opera setzen den vom `<q>`-Element umschlossenen Text in Anführungszeichen.

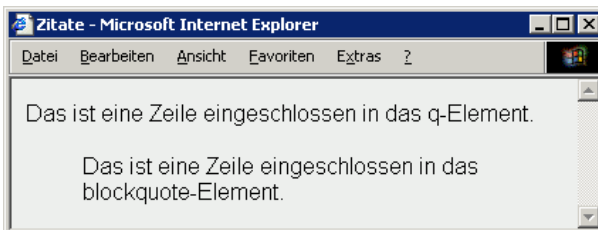


Abb. 10: Zitate im Internet Explorer

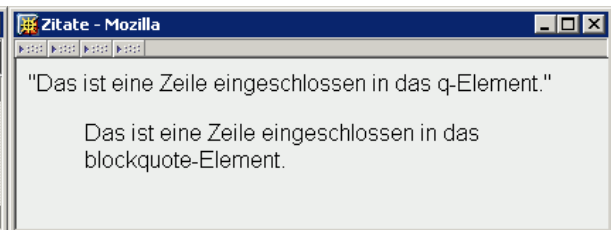


Abb. 11: Zitate im Mozilla

Zur Angabe der Quelle ist das `cite`-Attribut einzubinden.

Beispiel:	a) <code>&lt;q cite="Christiane Müller"&gt;Das ist eine Zeile [...]/&lt;q&gt;</code> b) <code>&lt;blockquote cite="Christiane Müller"&gt;Das ist eine Zeile [...]/&lt;/blockquote&gt;</code>
JAWS:	a) „Das ist eine Zeile eingeschlossen in das q-Element“ b) „Zitat von Doppelpunkt Christiane Müller, Das ist eine Zeile [...], Zitatende“

1) Inline-Element = Element, dass keine neue Zeile im Textfluss erzeugt, als untergeordnetes, „inneres“ Element für Blockelemente gedacht ist und Text sowie weitere Inline-Elemente enthalten kann. (Vgl. Münz/Nefzger 2002, S. 1207)

2) Vgl. Münz/Nefzger 2002, S. 176

### 6.2.3 Abkürzungen und Akronyme

Ebenfalls in die Rubrik sprachlicher Besonderheiten fallen Abkürzungen (<*abbr*>-Element) und Akronyme (<*acronym*>-Element).

- Abkürzungen werden im täglichen Sprachgebrauch ausgesprochen, in der Schriftform jedoch gekürzt. Die in Schriftform notierte Abkürzung „z.B.“ wird gesprochen zu „zum Beispiel“.
- Akronyme sind Buchstabenkombinationen, die im täglichen Sprachgebrauch zu finden sind und aus den Anfangsbuchstaben mehrerer Wörter gebildet werden. „BGB“ (das Akronym zu „Bürgerliches Gesetzbuch“) beispielsweise wird gesprochen zu „be-ge-be“.

In beiden Fällen ist über das *title*-Attribut das Hinterlegen der Ausschreibung erforderlich.

Beispiel:	a) <abbr <b>title</b> ="zum Beispiel">z.B.</abbr> b) <acronym <b>title</b> ="Bürgerliches Gesetzbuch">BGB</acronym>
JAWS:	a) „zum Beispiel“ b) „Bürgerliches Gesetzbuch“

Zur Verbesserung der Verständlichkeit kann natürlich innerhalb des Fließtextes die vollständige Schreibweise eines Akronymes noch zusätzlich angegeben werden.<sup>1</sup>

Wird mit der Maus über eine Abkürzung oder ein Akronym gegangen, erscheint die vollständige Schreibweise. Im Mozilla werden Abkürzungen und Akronyme zusätzlich durch Unterstreichungen visualisiert.

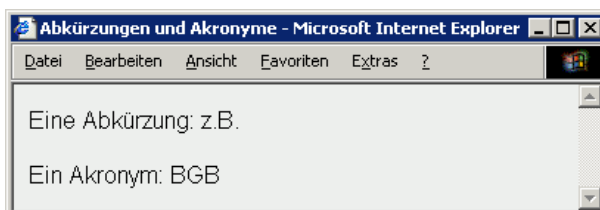


Abb. 12: Abkürzungen im Internet Explorer

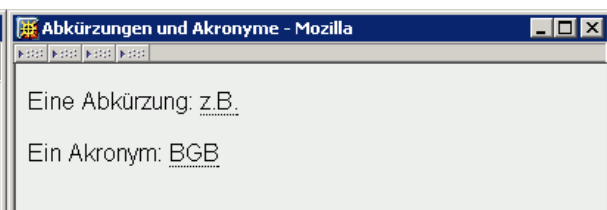


Abb. 13: Abkürzungen im Mozilla

Empfehlenswert ist die ergänzende Hervorhebung von Abkürzungen und Akronymen über stilistische CSS-Mittel wie Unterstreichungen oder farbliche Kennzeichnung. Der Internet Explorer ignoriert jedoch alle CSS-Angaben zum <*abbr*>-Element.

1) Vgl. Hartmann 2002 (S. 13)

## 7 Tabellen und Formulare

### 7.1 Tabellen

"Tabellen sollten verwendet werden, um tatsächlich tabellarische Daten ('Datentabellen') zu kennzeichnen. Entwickler von Inhalten sollten es vermeiden, sie für das Seitenlayout zu verwenden ('Layout-Tabellen')."<sup>1</sup>

#### 7.1.1 Datentabellen

Für das Umsetzen barrierefreier Tabellen stehen Elemente und Attribute für Überschriften, Zusammenfassungen, Beschreibungen und Beziehungszuordnungen zur Verfügung.

Das *summary*-Attribut, welches direkt im einleitenden `<table>`-Element zu notieren ist, dient der Angabe einer Zusammenfassung. Diese ist besonders für blinde Nutzer hilfreich, die so einen Überblick über den Tabelleninhalt erhalten. Über das `<caption>`-Element wird der Tabelle eine Überschrift zugewiesen, die standardmäßig zentriert über der Tabelle angezeigt wird.

Einzelne Tabellenzellen sind in Daten- und Kopfzellen zu unterscheiden. Datenzellen werden durch das Element `<td>` gekennzeichnet, Kopfzellen durch das Element `<th>`. Jede Datentabelle ist mit Zeilen- oder Spaltenüberschriften zu versehen. Für lange Überschriften innerhalb einer Kopfzelle wird empfohlen, das *abbr*-Attribut zu verwenden. Im Gegensatz zum `<abbr>`-Element, welches Abkürzungen kennzeichnet, dient hier *abbr* als Attribut und beinhaltet eine Kurzbezeichnung. Dies ist besonders bei mehrstufigen Tabellen für Nutzer von Screenreadern von Vorteil, der in diesem Fall nur beim ersten Lesen der Überschriftenzelle den langen Text vorliest und bei späteren Verknüpfungen dann nur den Kurztext ausgibt.

Einfache Datentabellen haben nur jeweils eine Zeile oder Spalte für Überschriften. Wenn Tabellen jedoch zwei oder mehr logische Ebenen von Zeilen- oder Spaltenüberschriften haben, sollen die Datenzellen über Attributauszeichnungen den Überschriften zugeordnet werden. Aktuell ist für den Einsatz in Screenreadern nur auf die Attribute *id* und *headers* Verlass.

Werden diese Attribute entsprechend notiert, ist es dem Screenreader möglich, Datenzellen mit Kopfzellen in Verbindung zu setzen. Ein Screenreader liest zum Beispiel automatisch die neue Zeilen- oder Spaltenüberschrift vor, wenn der Benutzer die Tabellenzeile oder -spalte wechselt.<sup>2</sup>

---

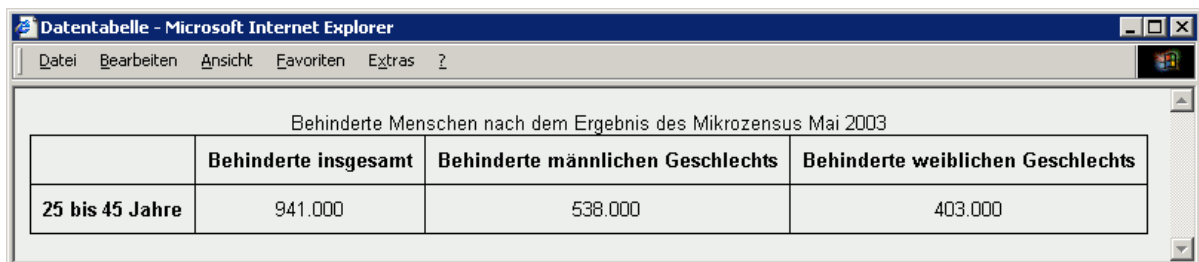
1) Hartmann 2002 (S. 13)

2) Vgl. DIAS 2004 (S. 4)

Beispiel: Die Daten sind den „Lebenslagen der behinderten Menschen, Ergebnis des Mikrozensus 2003“<sup>1</sup> entnommen.

```
<table summary="Behinderte Menschen in verschiedenen Altersstufen, unterteilt in weiblich, männlich und gesamt.">
<caption>Behinderte Menschen nach dem Ergebnis des Mikrozensus Mai 2003</caption>
<tr>
<th></th>
<th abbr="insgesamt" id="gesamt">Behinderte insgesamt</th>
<th abbr="männlich" id="mann">Behinderte männlichen Geschlechts</th>
<th abbr="weiblich" id="frau">Behinderte weiblichen Geschlechts</th>
</tr>
<tr>
<th id="arbeiter">25 bis 45 Jahre</th>
<td headers="arbeiter gesamt">941.000</td>
<td headers="arbeiter mann">538.000</td>
<td headers="arbeiter frau">403.000</td>
</tr>
</table>
```

JAWS: „Tabellenüberschrift Behinderte Menschen nach dem Ergebnis des Mikrozensus Mai zweitausenddreißig, Zusammenfassung Behinderte Menschen in verschiedenen Altersstufen unterteilt in weiblich männlich und gesamt, Tabelle mit vier Spalten und zwei Reihen, Behinderte insgesamt Spalte zwei, Behinderte männlichen Geschlechts Spalte drei, Behinderte weiblichen Geschlechts Spalte vier, fünf und zwanzig bis fünf und vierzig Jahre Spalte eins, Behinderte insgesamt neun hundert ein und vierzigtausend, Behinderte männlichen Geschlechts fünf hundert achtunddreißig, Behinderte weiblichen Geschlechts vier hundert dreitausend“



	Behinderte insgesamt	Behinderte männlichen Geschlechts	Behinderte weiblichen Geschlechts
25 bis 45 Jahre	941.000	538.000	403.000

Abb. 14: Datentabelle

Über Strg + Alt + Pfeiltasten (links, rechts, hoch, runter) kann im JAWS zwischen Tabellenzellen navigiert werden. Durch die Angabe der Kurzbezeichnung *abbr*="weiblich" in der Kopfzelle der 4.Spalte gibt JAWS beim Wechsel von Spalte 3 auf Spalte 4 in Zeile 2 beispielsweise aus:

JAWS: „weiblich, vierhundertdreitausend, Spalte vier“

1) Vgl. Pfaff 2004, S. 1182



### 7.1.2 Layouttabellen

Screenreader verarbeiten Tabellen zeilenweise und geben die Tabellenzellen von links nach rechts als einzelne Informationsblöcke aus.<sup>1</sup> Dieses Vorgehen beschreibt die sogenannte **Linearisierung**. Eine Tabelle ist linearisierbar, wenn der vom Screenreader vorgelesene Inhalt in einer sinnvollen Reihenfolge wiedergegeben wird.

Barrieren entstehen bei durch Layouttabellen erzeugtem mehrspaltigen Text, wenn die von Print-Medien übernommene Darstellung des Mehrspalten-Layouts in der Linearisierung zu unverständlichen Inhalten führt.

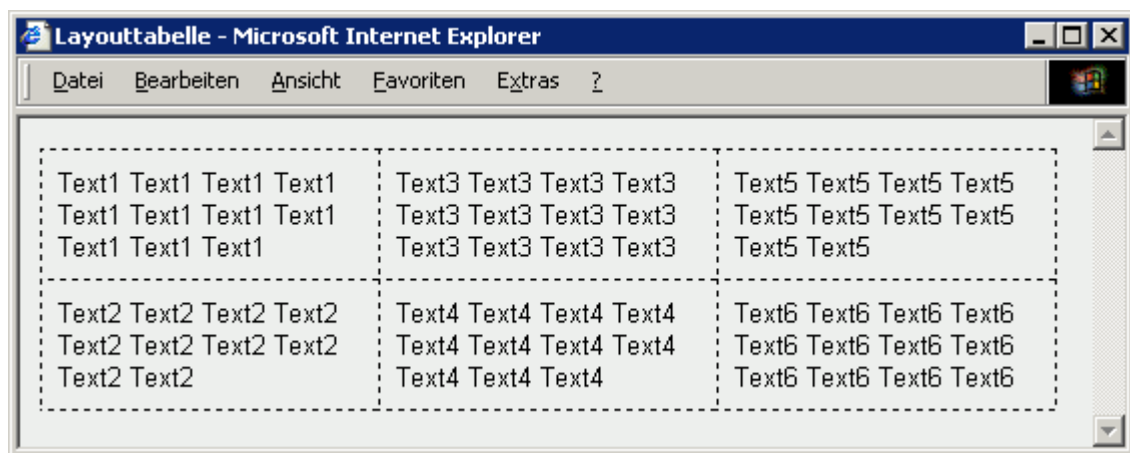


Abb. 15: Layouttabelle

JAWS: „Text 1 [...], Text 3 [...], Text 5[...], Text 2 [...], Text 4 [...], Text 6 [...]“

In der „traditionellen“ Seitengestaltung werden Layouttabellen des Weiteren für die Gestaltung von Formularen eingesetzt, um Beschriftungen links von Formularfeldern zu positionieren. Der barrierefreie Einsatz von Formularen wird im nächsten Abschnitt näher betrachtet.

1) Vgl. Hellbusch 2005 a, S. 154

## 7.2 Formulare

Mit dem `<label>`-Element wird eine Beschriftung eingebunden und diese über das `for`-Attribut mit dem Formularelement verknüpft. Dazu muss der Inhalt des `for`-Attributes identisch sein mit dem `id`-Attribut des zugehörigen Formularelementes.<sup>1</sup>

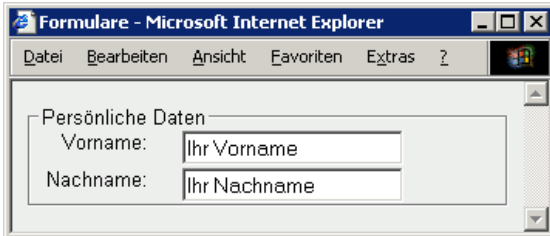


Abb. 16: Formulare

Screenreader verbinden Beschriftungen nach festen Regeln mit den Eingabefeldern von Formularen. Die Beschriftung von Formularelementen soll daher über Stylesheets links neben oder über dem zugehörigen Eingabefeld angeordnet werden.

Formularelemente können über das `<fieldset>`-Element zu Formularabschnitten zusammengefasst werden. Über das `<legend>`-Element erhält eine durch das `<fieldset>`-Element umschlossene Einheit einen Titel und das Verstehen des Formularnutzens wird unterstützt.<sup>2</sup>

Beispiel:	<pre> &lt;fieldset&gt; &lt;legend&gt;Persönliche Daten&lt;/legend&gt; &lt;label for="vorname"&gt;Vorname:&lt;/label&gt; &lt;input type="text" id="vorname" value="Ihr Vorname" /&gt;&lt;br /&gt; &lt;label for="nachname"&gt;Nachname:&lt;/label&gt; &lt;input type="text" id="nachname" value="Ihr Nachname" /&gt; &lt;/fieldset&gt; </pre>
CSS-Datei:	<pre> label { float:left; text-align:right; width:2cm; margin-right:0.5cm; } </pre>
JAWS:	<pre> „Persönliche Daten, Vorname Doppelpunkt, Eingabefeld Ihr Vorname, Nachname Doppelpunkt, Eingabefeld Ihr Nachname“ </pre>

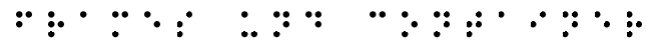
Pflichtfelder sollten in Formularen besonders gekennzeichnet werden, oft verwendet wird das Anfügen eines Sternchens (\*) an die Beschriftung. Die sicherste Methode, Pflichtfelder zu kennzeichnen, ist das Ausschreiben eines Wortes wie „Pflicht“ in der Bezeichnung des jeweiligen Formularelementes.<sup>3</sup>

1) Vgl. Münz/Nefzger 2002, S. 327

2) Vgl. Mintert et al. 2003, S. 280

3) Vgl. Hellbusch 2005 a, S. 189

## 8 Frames und Container



### 8.1 Frames

Frames segmentieren das Browserfenster, wobei jeder Frame eine Datei oder ein weiteres Frameset enthält. Frames haben den Vorteil, dass nur ein einmaliger Navigationsaufbau erfolgt, was jedoch auch zu höheren Ladezeiten führen kann. Ältere Browser können Frames nicht immer korrekt interpretieren, Screenreader lesen Frames nacheinander vor und Textbrowser können Frames gar nicht darstellen und sind auf einen `<noframes>`-Bereich angewiesen. Auch das Setzen von Lesezeichen kann sich als problematisch erweisen.

Mit einem Frameset (`<frameset>`-Element) können beliebig viele Frames (`<frame>`-Element) gleichzeitig eingebunden und gewechselt werden. Durch die dynamische Veränderung der Inhalte in mehreren Frames gleichzeitig werden jedoch rasch die Grenzen der Gebrauchstauglichkeit erreicht.<sup>1</sup> Deshalb ist es besonders wichtig, jeden Frame sinnvoll zu bezeichnen. Über das `name`-Attribut erfolgt eine kurze Bezeichnung des Inhaltes oder Zweckes, das `title`-Attribut kann sogar ganze Sätze enthalten.

Frames, die ausschließlich zu Layoutzwecken eingesetzt werden, sollten die Bezeichnung „leer“ enthalten. Weiterhin ist über den `<noframes>`-Bereich ein äquivalenter Zugang zu den im Frameset enthaltenen Seiten sicherzustellen, da die linearisierte Form eines Framesets aufgrund einer schnell ansteigenden Komplexität für Blinde umständlich bedienbar werden kann.

```
Beispiel: <frameset cols="100,*">
  <frame name="Navigation" title="Navigation" src="menu.html" />
  <frame name="Inhalt" src="inhalt.html" /> /* Gegenbeispiel: name="rechts" */
</noframes>
<body>
  <ul>
    <li><a href="menu.html">zur Navigation</a></li>
    <li><a href="inhalt.html">zum aktuellen Inhalt</a></li>
    <li><a href="sitemap.html">zum Inhaltsverzeichnis</a></li>
  </ul>
</body>
</noframes>
</frameset>
```

JAWS: „Navigation Rahmen, Menü [Anm. d. Verf.: entspricht Titel der Seite], Menü-Frame [Anm. d. Verf.: entspricht Inhalt der Seite], Navigation Rahmen Ende, Inhalt Rahmen, Inhalt [Anm. d. Verf.: entspricht Titel der Seite], Inhalts-Frame [Anm. d. Verf.: entspricht Inhalt der Seite], Inhalt Rahmen Ende“

1) Vgl. Hellbusch 2005 a, S. 163



Abb. 17: Frame-Ansicht



Abb. 18: Noframe-Ansicht

Eine weitere akzeptable Lösung für den `<noframes>`-Bereich ist der Verweis zu einer Übersichtsseite, einer sogenannten Sitemap, mit der ein Nutzer die Möglichkeit erhält, das Konzept eines Auftritts und dessen wichtigste Kategorien und Inhalte zu überblicken.

## 8.2 Container

Eine Alternative zum vorgenannten Frame-Aufbau ist das Aufteilen einer Seite in `<div>`-Elemente, sogenannte Container.

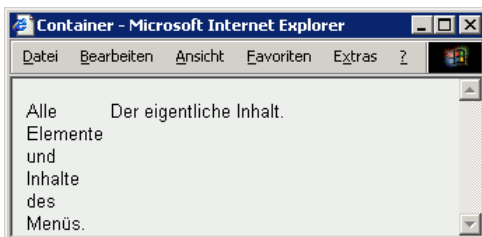


Abb. 19: Container

Container können mehrere Elemente wie Text, Grafiken und Tabellen in einen gemeinsamen Bereich einschließen, über Stylesheets fest positioniert werden und äquivalent zum Framegerüst verschiedene Bereiche einer Internetseite repräsentieren.<sup>1</sup>

Bei der Entwicklung einer Internetseite ist zu beachten, dass diese auch für Nutzer mit geringer Bildschirmauflösung lesbar ist. Das Layout der Seite soll deshalb flexibel sein, so dass sich Elemente des Fließtextes nicht überlagern. Das kann durch eine relative Breitenangabe für die einzelnen Container erreicht werden, beispielsweise „`width:20%`“ für den Navigationscontainer.

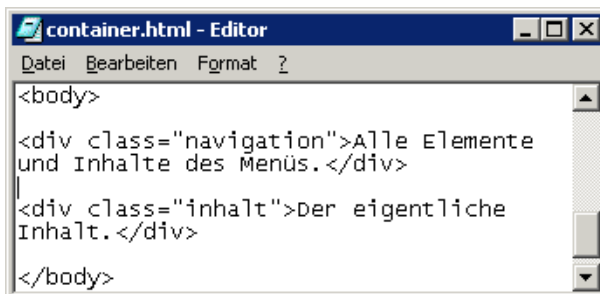
Beispiel: `<div class="navigation">Alle Elemente und Inhalte des Menüs.</div>`  
`<div class="inhalt">Der eigentliche Inhalt.</div>`

CSS-Datei: `div.navigation { position:absolute; width:20%; }`  
`div.inhalt { position:absolute; width:80%; margin-left:20%; }`

JAWS: „Alle Elemente und Inhalte des Menüs, Der eigentliche Inhalt“

1) Vgl. Münz/Nefzger 2002, S. 174

Die Reihenfolge der `<div>`-Elemente im Quelltext ist ausschließlich in Zusammenhang mit der Linearisierung von Bedeutung aber somit für Nutzer von Textbrowsern und Screenreadern entscheidend. Container mit für den Nutzer relevanten Inhalten sind immer an den Anfang eines Dokumentes zu stellen, sich wiederholende Elemente, wie beispielsweise aktuelle Meldungen oder Besucherstatistiken, können in Container eingebettet werden, die im Quelltext erst zum Ende hin erscheinen. Ein die Hauptnavigation enthaltender Container kann beispielsweise nur in der Startdatei im Quelltext an erster Stelle stehen (siehe *Container 1*), in den darauffolgenden Dokumenten dann jedoch an das Ende verlegt werden (siehe *Container 2*).

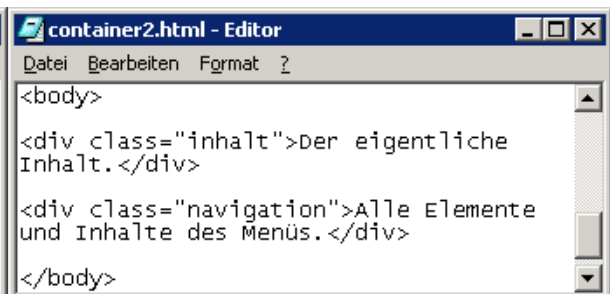


```

<body>
<div class="navigation">Alle Elemente
und Inhalte des Menüs.</div>
<div class="inhalt">Der eigentliche
Inhalt.</div>
</body>

```

Abb. 20: Quelltext Container 1



```

<body>
<div class="inhalt">Der eigentliche
Inhalt.</div>
<div class="navigation">Alle Elemente
und Inhalte des Menüs.</div>
</body>

```

Abb. 21: Quelltext Container 2

Durch das über CSS erfolgte Positionieren der Container werden Nutzer grafischer Ausgabegeräte diese Feinheit nicht weiter bemerken.

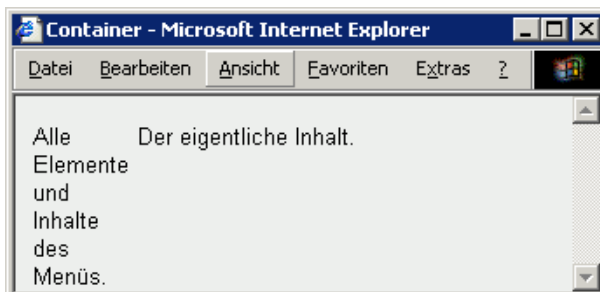


Abb. 22: Ausgabe Container 1

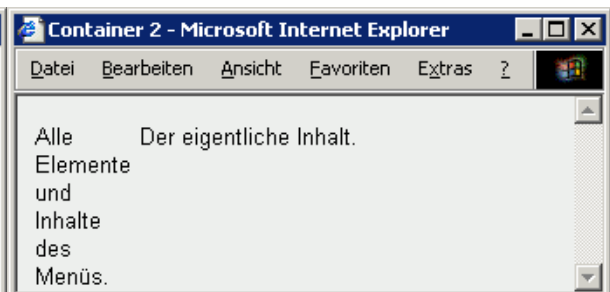


Abb. 23: Ausgabe Container 2

Hilfreich ist das unterschiedliche Positionieren im Quelltext jedoch für die im ersten Absatz beschriebene Nutzergruppe von Textbrowsern und Screenreadern.

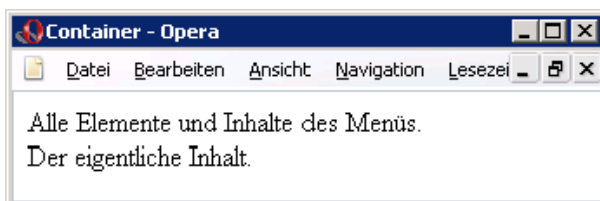


Abb. 24: Linearisierung Container 1

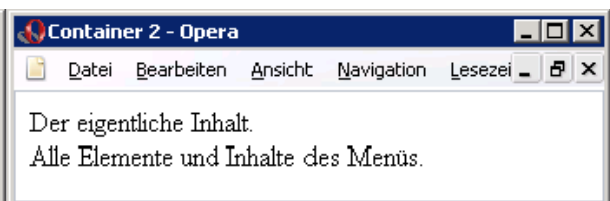


Abb. 25: Linearisierung Container 2

## 9 Seitenaufbau



### 9.1 Dokumentenaufbau

Bei der barrierefreien Gestaltung von Internetseiten ist auf das Strukturieren und Gliedern des Inhaltes zu achten. Besonders hervorzuheben sind der Einsatz von Überschriften und Listen sowie die Angabe von Metadaten<sup>1</sup>.

#### 9.1.1 Überschriften

Überschriften strukturieren Text und gliedern den Inhalt einer Seite. Für sehende Besucher sind Überschriften klar erkennbar: Sie sind größer, haben eine andere Schrift oder sind durch die Farbe hervorgehoben.<sup>2</sup> Blinde und stark sehbehinderte Besucher können solche optischen Merkmale jedoch nicht wahrnehmen und brauchen „logische“, im Quelltext der Seite festgehaltene Auszeichnungen, um sich auf der Seite zu orientieren.

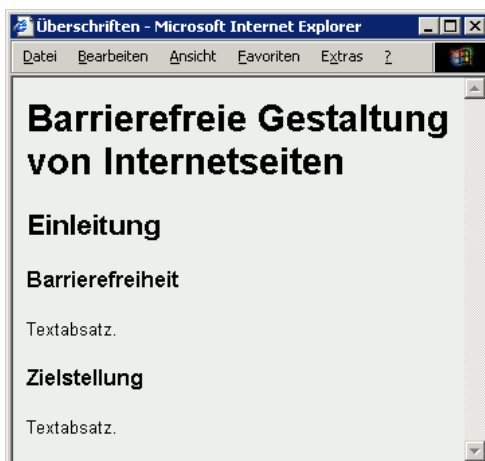


Abb. 26: Überschriften

Wichtig ist, dass Überschriften entsprechend ihrer Ordnung abgestuft werden. Überschriften erster Ordnung dienen der Angabe einer Hauptüberschrift, mit Überschriften zweiter Ordnung können Kapitel ausgezeichnet werden und Überschriften dritter Ordnung können Unterkapitel kennzeichnen.

Eine Überschrift der dritten Ordnung sollte also niemals auf eine Überschrift der ersten Ordnung folgen.

Beispiel: `<h1>Barrierefreie Gestaltung von Internetseiten</h1>`  
`<h2>Einleitung</h2>`  
`<h3>Barrierefreiheit</h3>`  
`<p>Textabsatz.</p>`  
`<h3>Zielstellung</h3>`  
`<p>Textabsatz.</p>`

JAWS: „Überschrift Ebene eins Barrierefreie Gestaltung von Internetseiten, Überschrift Ebene zwei Einleitung, Überschrift Ebene drei Barrierefreiheit, Textabsatz, Überschrift Ebene drei Zielstellung, Textabsatz“

1) Metadaten = Das Dokument näher beschreibende Daten im Kopfbereich einer Internetseite.

2) Vgl. DIAS 2004 (S. 4)

### 9.1.2 Listen

Listen werden über die Elemente `<ul>` (ungeordnete Liste) und `<ol>` (geordnete Liste) gekennzeichnet. Werden eigene Grafiken als Listenelemente des `<ul>`-Elementes verwendet, ist ergänzend eine Alternative anzugeben, die bei Nichtladen der Grafik zum Einsatz kommt.

Beispiel: 

```
<ul class="eigene_grafik">
  <li>Listenpunkt 1</li>
  <li>Listenpunkt 2</li>
  <li>Listenpunkt 3</li>
</ul>
```

CSS-Datei: 

```
ul.eigene_grafik { list-style:url(blauer_kreis.gif) disc; }
```



Abb. 27: Liste bei eingeschalteten Grafiken



Abb. 28: Liste bei ausgeschalteten Grafiken

## 9.2 Kopfdaten

Den Inhalt bezeichnende Dokumententitel dienen der Unterscheidung und Auswahl von Seiten. Der Dokumententitel wird in Browsern in der Kopfzeile angezeigt, beim Setzen von Lesezeichen verwendet und in der Liste der bereits besuchten Seiten angezeigt.

Die beiden vorangegangenen Grafiken zeigen ein Dokument mit dem Titel „Listen“. Der Dokumententitel wird üblicherweise im Kopfbereich des Dokumentes (siehe Abschnitt 4.2.2) über das `<title>`-Element eingebunden.

Beispiel: 

```
<head><title>Listen</title></head>
```

Suchmaschinen bieten den Titel eines Dokumentes als anklickbaren Verweis an und verwenden den Inhalt des Titels für die Suche.<sup>1</sup> Weitere, für Suchmaschinen relevante Angaben werden nachfolgend näher beschrieben.

1) Vgl. Münz/Nefzger 2002, S. 119

### 9.2.1 Metadaten

Für Suchmaschinen relevante Informationen können über das `<meta>`-Element im Kopfbereich einer Seite angegeben werden, auch die Einbindung des Seitentitels kann ergänzend zum `<title>`-Element über eine Metaangabe (`name="title"`) erfolgen. Ebenfalls möglich sind beispielsweise Angaben zur Sprache des Dokumentes (`name="language"`), zum Autor (`name="author"`), zum Inhalt (`name="description"`) und zu Stichwörtern (`name="keywords"`).

```
Beispiel: <meta name="title" content="Seitentitel" />
          <meta name="language" content="de" />
          <meta name="author" content="Name des Autors" />
          <meta name="description" content="Beschreibung der Seite" />
          <meta name="keywords" content="Kommatagetrennte, Stichwörter" />

          <link rel="start" href="index.html" title="Startseite" />
          <link rel="up" href="kapitel.html" title="Eine Ebene hoch" />
          <link rel="prev" href="unterkapitel_1_01.html" title="Unterkapitel 1.1" />
          <link rel="next" href="unterkapitel_1_03.html" title="Unterkapitel 1.3" />
          <link rel="contents" href="sitemap.html" title="Inhaltsverzeichnis" />
          <link rel="chapter" href="kapitel_1.html" title="Ausgewählt: Kapitel 1" />
          <link rel="chapter" href="kapitel_2.html" title="Kapitel 2" />
          <link rel="index" href="stichwort.html" title="Stichwortverzeichnis" />
```

In einem BBC-Interview sagte Dr. Jakob Nielsen, bezogen auf die Bedeutung von Suchmaschinen: „They are becoming like the operating system to the internet [...]“<sup>1</sup> Er untersuchte auch die Nutzungshäufigkeit von Suchmaschinen und stellte fest, dass Nutzer bei einer konkreten Aufgabenstellung in 88% der Fälle zuerst eine Suchmaschine kontaktierten.<sup>2</sup>

### 9.2.2 Seitennavigation

Das `<link>`-Element ist ein weiteres Element, das im Kopfbereich einer Seite notiert wird. Jede Seite kann über dieses Element mit der Startseite (`rel="start"`), einer übergeordneten Ebene (`rel="up"`) und inhaltlich vorangegangenen (`rel="prev"`) oder nachfolgenden (`rel="next"`) Seiten verbunden werden. Weiterhin können beispielsweise Beziehungen zum Inhaltsverzeichnis (`rel="contents"`), verschiedenen Kapiteln (`rel="chapter"`) oder einem Index (`rel="index"`), einer Hilfe, einer Suche, Angaben zum Autor oder Copyright und anderen Versionen des Dokumentes festgehalten werden.

1) Ward 2004 (S. 2)

2) Vgl. Nielsen 2004 (S. 1)



Im Browser Mozilla kann die sogenannte Site-Navigationsleiste aktiviert werden, in der die vorgenannten Seitennavigationselemente angezeigt werden.



Abb. 29: Site-Navigationsleiste im Mozilla

Da diese Navigationsmöglichkeit jedoch bisher nur von wenigen Browsern unterstützt wird, müssen vorerst herkömmliche und vor allem bekannte Navigationsmechanismen und -elemente angewendet und eingesetzt werden.

### 9.3 Navigation

Einheitliche Navigation erleichtert das Bewegen im Internetauftritt. Navigationselemente sollten daher in allen Seiten einer Internetpräsenz in gleicher Weise verwendet werden. Dabei ist das Nutzen von verbreiteten und allgemein bekannten Navigationsmechanismen angebracht. Diese sollten am oberen oder linken Seitenrand angeordnet werden, um zu gewährleisten, dass bei allen Bildschirmgrößen und Browsereinstellungen die Navigationselemente sichtbar und benutzbar sind.<sup>1</sup>

Zum Thema „Nutzergerechte Gestaltung von Homepages“<sup>2</sup> wurden erfahrene Nutzer bezüglich ihrer Erwartungshaltung zu bestimmten Navigationselementen befragt. Über 50% erwarten die Hauptnavigation am linken Rand, die obere linke Ecke sollte für einen Verweis zur Startseite genutzt werden. Anders ist es mit einem Hinweis auf Umfragen. Dieser wird am rechten Rand und vertikal mittig erwartet.

Damit der Nutzer jederzeit einen Überblick über seine Position im Internetauftritt hat, ist der derzeitige Pfad („bread crumb trail“ oder auch „Brotkrumen-Pfad“) mit übergeordneten Seiten bis hin zur Startseite anzugeben. Diese sollten als Verweise eingebunden sein, so dass sie als Navigationselemente genutzt werden können. Das jeweils aktive Dokument, also das Ende des Pfades, ist nicht als Verweis auszuzeichnen.

Beispiel: Sie sind hier: [Startseite](#) > [Kapitel](#) > [Unterkapitel](#) > Abschnitt 1

1) Vgl. Böhringer et al. 2003, S. 175

2) Vgl. eResult 2004 (S. 10ff)

Verweistexte sollen aussagekräftig und auch außerhalb ihres Textzusammenhanges eindeutig und unterscheidbar sein. Dies ermöglicht blinden und sehbehinderten Nutzern, sich einen schnellen Überblick über das Angebot einer Internetseite zu verschaffen.

Beispiel:	„Informationen zu Barrierefreiem Webdesign“
Gegenbeispiel:	„Hier klicken“

In Verbindung mit Navigationselementen werden in der Praxis teilweise Tastatur-Kurzbefehle angeboten, beispielsweise die Taste S für „Startseite“. Dadurch entstehende Probleme werden im nächsten Abschnitt näher betrachtet.

#### 9.4 Tastaturbedienung

Menschen mit einer Sehbehinderung oder mit motorischen Störungen setzen eine Maus nur bedingt ein. Sie profitieren, ebenso wie jeder der lieber mit der Tastatur als der Maus arbeitet, von der Angabe sogenannter Tastatur-Kurzbefehle (Shortcuts). Diese können über das `accesskey`-Attribut für wichtige Verweise und Formular-Kontrollelemente bereitgestellt werden. Zusätzlich zu dem Verweistext kann über das `title`-Attribut das Ziel des Verweises angegeben werden.

Neben der Angabe von Shortcuts kann auch eine Tabulator-Reihenfolge fest definiert werden. Normalerweise werden die Elemente in der Reihenfolge angesprungen, in der sie in der Datei definiert sind.<sup>1</sup> Über das `tabindex`-Attribut können alle Elemente eines Dokumentes jedoch eine andere Reihenfolge zugeordnet bekommen.

Beispiel:	<code>&lt;a href="index.html" accesskey="0" title="zur Startseite" tabindex="1"&gt;Startseite&lt;/a&gt;</code>
JAWS:	„Zur Startseite Alt Pluszeichen Null Link“

Tastatur-Kurzbefehle und Tabulator-Reihenfolgen können eine Navigation über die Tastatur erheblich beschleunigen und den Zugang für Blinde vereinfachen. Schwierig ist jedoch, eine browserübergreifende Norm zu schaffen, da die Browser zum Teil eigene Shortcuts in Verbindung mit der Alt-Taste verwenden.<sup>2</sup> Für Normierungsbemühungen bleiben daher nur die Ziffern 0 bis 9, da diese bislang nicht für browserspezifische Kurzbefehle verwendet werden.

1) Vgl. Münz/Nefzger 2002, S. 328

2) Hellbusch 2005 b (S. 2)

In einer gemeinsamen Initiative von Jan Eric Hellbusch, Ralf Hertsch und bundesrepublik.org wurde ein **AccessKey-Pad**<sup>3</sup> entwickelt, das ein einheitliches Belegen der Tastatur-Kurzbeefehle fördern soll.

Das AccessKey-Pad lässt sich relativ einfach in die eigene Internetpräsenz einbinden, da es ausschließlich mit HTML und CSS realisiert wurde. Sowohl der dafür nötige Quelltext als auch die CSS-Formatierungszuordnungen werden zum Download angeboten, Anpassungen können problemlos vorgenommen und das AccessKey-Pad durch die Einbettung in einen Container beliebig auf der eigenen Seite positioniert werden.



Abb. 30: Original AccessKey-Pad (Quelle: <http://2bweb.de/accesskey/download.htm>)

Empfohlen werden folgende Belegungen der Ziffern 0 bis 9:

- zur Startseite Alt + 0,
- zum Seitenanfang Alt + 1,
- zur nächsten Seite Alt + 2,
- zur vorherigen Seite Alt + 3,
- zur nächsten Rubrik oder eine Ebene tiefer Alt + 4,
- zur vorherigen Rubrik oder eine Ebene höher Alt + 5,
- zur Gesamtübersicht, Sitemap oder Suchfunktion Alt + 6,
- zum Inhaltsverzeichnis der aktuellen Seite Alt + 7,
- zum Inhalt der aktuellen Seite Alt + 8 und
- Alt + 9 um direkten Kontakt zum Autor der Seite herzustellen oder das Impressum, die AGB oder einen Disclaimer aufzurufen.

Im Internet Explorer muss anschließend noch die Enter-Taste, in Opera ab Version 7 mit Umschalt + Esc die Shortcut-Funktion aktiviert und dann die jeweilige Zahl gedrückt werden.

---

3) Zum Download verfügbar unter <http://2bweb.de/accesskey/download.htm>

## 10 Testwerkzeuge



Zum Überprüfen der Barrierefreiheit einer Internetseite ist eine Kombination aus Browser-Tests, manuellen und automatischen Tests sowie Nutzertests empfehlenswert.

Browser sind in die zwei Kategorien grafisch und nicht-grafisch unterteilbar. Grafische Browser sind beispielsweise der Internet Explorer, Opera und Mozilla. Weniger bekannt hingegen sind nicht-grafische, textbasierte Browser, auch Textbrowser genannt, die nur einfachen Text und Textformate interpretieren und darstellen können.

### 10.1 Browser-Tests

#### 10.1.1 Lynx

*Lynx*<sup>1</sup> ist ein rein textorientierter Browser zur Anzeige von Internetseiten. In *Lynx* ist keine visuelle (durch Stylesheets formatierte) Gliederung verfügbar, Grafiken werden nicht angezeigt. Aus diesem Grund ist ein Textbrowser sehr schnell, aufgrund der fehlenden Grafikanzeige aber auch wenig verbreitet. Mit Hilfe von textbasierten Browsern ist es möglich, die logische Reihenfolge von Navigation und Seiteninhalt zu überprüfen und die klare Gliederung des ausgegebenen Textes zu kontrollieren. Ebenso wie Suchmaschinen können Textbrowser weder JavaScript noch Flash-Animationen interpretieren. Statt dessen werden Alternativbereiche wie das `<noframe>`-Element oder das `<noscript>`-Element ausgewertet.

#### 10.1.2 Opera

Besonders gut geeignet im Bereich der grafischen-Browser-Tests ist der Browser *Opera*<sup>2</sup>, der die Möglichkeit bietet, in den sogenannten „Benutzermodus“ zu wechseln.

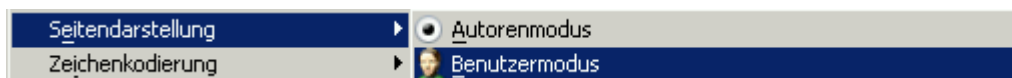


Abb. 31: Benutzermodus im Opera

In diesem können verschiedene Einstellungen gewählt werden: das Emulieren eines Textbrowsers, das ausschließliche Betrachten von Verweisen bei Verstecken des normalen Textes, die kontrastreiche Ansicht in schwarz-weiß und weiß-schwarz, die Linearisierung von Tabellen, das Ausschalten von nicht zur Navigation gehörenden Grafiken und das Anzeigen von strukturierenden Elementen.

1) Zum Download verfügbar unter <http://lynx.isc.org/>

2) Zum Download verfügbar unter <http://www.opera.com/download/>

Über die F12-Taste stehen zusätzlich „Schnelleinstellungen“ zur Verfügung, die je nach Bedarf das An- und Abschalten von Pop-Ups<sup>1</sup>, Java<sup>2</sup>, Plug-Ins, JavaScript, Cookies<sup>3</sup> und akustischen Wiedergaben ermöglichen.

Die Tastenkombination Alt + P führt in die „Einstellungen der Seitendarstellung“, wo die Darstellung von Frames deaktiviert und das Anzeigen des `<noframe>`-Elementes simuliert werden kann. Ein Test zur kontextunabhängigen Lesbarkeit von Verweisen, durch Selektion dieser aus dem gesamten Inhalt, ist über die Tastenkombination Strg + J erreichbar. Über die Taste G kann das Verhalten der Seite bei Ein- und Abschalten von Grafiken bezüglich der Hinterlegung von Alternativtexten getestet werden.

### 10.1.3 Internet Explorer

Für den *Microsoft Internet Explorer*<sup>4</sup> und verschiedene *Mozilla-Browser*<sup>5</sup> gibt es im Internet Programme und Erweiterungen, die nicht standardmäßig integriert sind, sich aber zur Überprüfung der Barrierefreiheit als hilfreich erweisen. Stellvertretend für beide Browserfamilien wird nachfolgend die *Web Accessibility Toolbar*<sup>6</sup> für den Internet Explorer vorgestellt, der nach wie vor der am meisten verbreitete und genutzte Browser ist. Die Web Accessibility Toolbar ist mit elf Rubriken eine sehr umfangreiche Erweiterung.

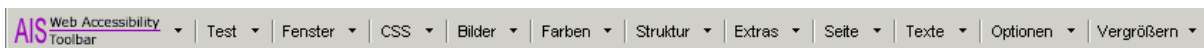


Abb. 32: Web Accessibility Toolbar im Internet Explorer

Über die Rubrik „*Test*“ sind verschiedene W3C Validatoren und Prüfwerkzeuge angebunden. Die Rubrik „*Fenster*“ ermöglicht das Testen von verschiedenen Bildschirmauflösungen und Plattformen. In der Rubrik „*CSS*“ können Stylesheets (sowohl extern als auch intern) ein- und ausgeschaltet, eigene Stylesheet-Zuweisungen getestet und als veraltet geltende Elemente angezeigt werden. Die Rubrik „*Bilder*“ beinhaltet ein Umschalten zwischen der Ansicht mit Grafiken und einer Ansicht mit Alternativtexten. In der Rubrik „*Farben*“ sind ein Graustufen-Test, eine Auflistung der verwendeten Farben sowie der Verweis auf eine Farbkontrastanalyse zu finden.

- 1) Pop-Up = Browser-Fenster, die neu geöffnet werden, ohne dass der Nutzer vorher auf das Öffnen eines neuen Fensters hingewiesen wurde.
- 2) Java = Von Sun Microsystems entwickelte Programmiersprache. Java-Programme sind nur lauffähig, wenn die Java Runtime Environment auf dem Rechner zur Verfügung steht.
- 3) Cookie = Textdatei zur (Zwischen-)Speicherung von Daten, die beim Besuch mancher Internetseiten auf der Festplatte des Besuchers abgelegt wird.
- 4) Zum Download verfügbar unter <http://www.microsoft.com/downloads/>
- 5) Zum Download verfügbar unter <http://www.mozilla.org/products/>
- 6) Zum Download verfügbar unter <http://www.nils.org.au/ais/web/resources/toolbar>

Die Rubrik „*Struktur*“ zeigt den Dokumenten-Titel, Überschriften-, Listen-, Formular-, Akronym- und Abkürzungs-Elemente, Attribute zur Tastatursteuerung, eine Tabellenanalyse und deren Linearisierbarkeit sowie Frame- und Container-Elemente. In der Rubrik „*Extras*“ wird auf weitere Prüfprogramme verwiesen. Der Unterbereich „*Simulationen*“ verschafft einen umfangreichen Überblick über die Auswirkungen verschiedener Sehbeeinträchtigungen. Die Rubrik „*Seite*“ zeigt Informationen (Titel, Dateigröße, Erstelldatum, Zahl der Bilder, Verweise, Formulare, Skripte) über die betrachtete Seite und verschiedene Listen über verwendete Frames, Verweise oder Metadaten sowie die Downloadzeit des Dokumentes.

In der Rubrik „*Texte*“ sind Verweise auf Internetseiten mit Richtlinien zur Barrierefreiheit (WCAG1 und BITV), Usability-Quellen und Sprachreferenzen (HTML, XHTML, CSS) angeführt. Über die Rubrik „*Optionen*“ können Bilder, JavaScript, die CSS-Unterstützung des Internet Explorers und das Dialogfenster „Eingabehilfen“ ein- und ausgeschaltet und die Textgröße geändert werden. In der Rubrik „*Vergrößern*“ ist ein Skalieren des Inhaltes der aktuellen Seite von 25% bis 600% möglich.

## 10.2 Automatische und manuelle Tests

### 10.2.1 Bobby

Das *Center for Applied Special Technology* (CAST) hat in Zusammenarbeit mit dem World Wide Web Consortium (W3C) die Software *Bobby*<sup>1</sup> entwickelt, welche Internetseiten auf Barrierefreiheit gemäß den W3C-Richtlinien prüft. Nachdem der Tester die zu überprüfende URL<sup>2</sup> angegeben hat, erfolgt eine umfangreiche Auswertung, die in die drei Prioritäten der WCAG1 untergliedert ist. Jeder gemeldete Fehler ist der entsprechenden Zeilenzahl des getesteten Dokumentes zugeordnet, mit einem Hinweis zur Vermeidung des Fehlers verbunden und mit einer Referenz auf den zugrundeliegenden WCAG1-Checkpunkt versehen.

*Bobby* liefert eine sehr umfangreiche Überprüfung sowie Fehleranalyse in englischer Sprache und ist für erfahrene Webdesigner ein sinnvolles und hilfreiches Test-Werkzeug. Bei Bestehen des *Bobby*-Tests kann der Tester je nach Umsetzungsgrad die nachfolgenden Grafiken auf der überprüften Seite einbinden:



Abb. 33: Überprüft mit Bobby (Quelle: <http://bobby.watchfire.com/bobby/html/en/icon.jsp>)

1) Siehe <http://bobby.watchfire.com/>

2) URL = Uniform Resource Locator. Ort, an dem ein bestimmtes Dokument gespeichert ist, beispielsweise <http://www.google.com/> für die Startseite der Suchmaschine Google. (vgl. Hansen/Neumann 2001, S. 1192)

### 10.2.2 Barrierefinder

Ein etwas einfacheres und leichter verständliches Testwerkzeug in deutscher Sprache ist der *Barrierefinder*<sup>1</sup>. Der Test basiert auf Sicht- und Vergleichsbeurteilungen. Der Tester gibt eine URL an und bewertet in den nachfolgenden sechs Schritten wesentliche Merkmale der Barrierefreiheit. Dabei muss der Tester in den ersten fünf Schritten selbst entscheiden, inwieweit seine Seite den Anforderungen entspricht und Bildbeschreibungen, Schriftgrößen, Linearisierbarkeit, Graustufenbetrachtung und Funktionstüchtigkeit beim Ausschalten von Skripten bewerten. Möglich sind die Antworten „nein“, „überwiegend nein“, „zum Teil“, „überwiegend ja“ und „ja“. Im sechsten Schritt erfolgt ein automatischer Quellcode-Test: Dokumenttyp, Sprache der Seite, Tastatur-Kurzbefehle, Tabulator-Reihenfolge, Einbindung von Stylesheets, Popup-Fenster, Frames und Strukturierung des Textes.

Am Ende des Tests werden, wenn notwendig, Empfehlungen zum behindertengerechteren Umsetzen gegeben. Diese Empfehlungen sind mit Verweisen zu entsprechenden Kapitelabschnitten von *SELFHTML*<sup>2</sup>, der Online-Referenz zu den Themen HTML und CSS von Stefan Münz versehen.

### 10.2.3 WebFormator

Einen groben Überblick über die Funktionsweise eines Screenreaders liefert der *WebFormator*<sup>3</sup>, der den Inhalt einer Internetseite in einem separaten Textfenster darstellt. Ausgegeben werden der Titel der Seite, Frames werden untereinander dargestellt, Tabellen linearisiert und Alternativtexte von Bildern und Objekten angezeigt. Die Ausgabe ist rein textbasiert und gut geeignet für Entwickler, die einen ersten Eindruck gewinnen möchten, wie Blinde eine Internetseite präsentiert bekommen, da die Ausgabe des *WebFormator* in etwa dem entspricht, was ein Screenreader der Seite entnimmt und an die Sprachausgabe oder Braillezeile weitergibt.

Auch für das erfolgreiche Überprüfen mit dem *WebFormator* steht eine Grafik bereit, die den Nutzer der Seite auf diese Kompatibilität aufmerksam machen kann:



Abb. 34: Überprüft mit *WebFormator* (Quelle: <http://www.webformator.de/infos/webdesigner.html>)

1) Siehe <http://www.barrierefinder.de/>

2) Siehe <http://de.selfhtml.org/>

3) Zum Download verfügbar unter <http://www.webformator.de/deutsch/index.php>

### 10.2.4 Validierungsservices

Auf den Internetseiten des W3C gibt es verschiedene Validierungswerkzeuge. Zum Validieren von Dokumenten kann der *W3C Markup Validation Service*<sup>1</sup> genutzt werden, ein frei verfügbarer Service basierend auf den Empfehlungen des W3C zum Überprüfen von (X)HTML-Dokumenten. Entspricht das überprüfte Dokument nicht den formalen Standards, werden die fehlerhaften Stellen mit Zeilen- und Spaltenangaben aufgezeigt, rot sowie unterstrichen hervorgehoben und ein Vorschlag zum Vermeiden des Fehlers gegeben.

Mit dem *W3C CSS-Validierungsservice*<sup>2</sup> können Stylesheets auf Validität geprüft werden. Hier ist das Angeben einer URL, das Eingeben von CSS in einem Textfeld und das Hochladen einer CSS-Datei möglich.

Besteht das Dokument die Validierungstests, können die folgenden Grafiken dies für den Besucher der Seite verdeutlichen.



Abb. 35: Überprüft mit Validierungsservices des W3C (Quelle: <http://www.w3.org/Icons/>)

1) Siehe <http://validator.w3.org/>

2) Siehe <http://jigsaw.w3.org/css-validator/>



# 11 Ergebnisse



## 11.1 Von traditioneller zu standardbasierter Entwicklung

Der Schritt von traditioneller zu standardbasierter Entwicklung von Internetseiten erfordert Zeit, Geduld und Übung. Barrierefreiheit ist ein Sensibilisierungs- und Bewusstseinsprozess, das Umsetzen barriere-ärmerer Seiten wiederum ein Handwerk, das vor allem im Bereich der Cascading Stylesheets mit Lernaufwand verbunden ist.<sup>1</sup>

Resultierend aus den in Kapitel 4 bis 9 genannten Umsetzungskriterien ergeben sich die folgenden „TOP 10“-Fragen, deren positive Beantwortung einen großen Schritt in Richtung barrierefreier Informationstechnik bedeutet:

1. Werden aktuelle Technologien eingesetzt?
2. Ist die Seite auch ohne JavaScript bedienbar?
3. Sind Grafiken mit Alternativtexten und Beschreibungen versehen?
4. Ist die Seite kontrastreich gestaltet?
5. Sind Schriften skalierbar?
6. Sind Frames beschriftet und wird ein Alternativbereich angeboten?
7. Werden Strukturierungselemente richtig eingesetzt?
8. Wird eine einfache und übersichtliche Navigation verwendet?
9. Sind Verweise auch außerhalb des Kontextes sinnvoll bezeichnet?
10. Ist eine Tastatursteuerung möglich?

Begonnen werden kann mit grundlegenden Änderungen wie dem Hinzufügen eines korrekten Dokumententypes, dem Ergänzen von Textalternativen sowie der Vergabe von aussagekräftigen Seitentiteln. Mittelfristig sollten dann Anpassungen bezüglich validem und semantisch korrektem Quelltext erfolgen, integrierte `<font>`-Elemente und Farbinformationen durch Stylesheet-Angaben ersetzt und zugängliche Formulare sowie Datentabellen gestaltet werden. Abschließend ist die komplette Trennung von Inhalt und Layout durch Auslagern aller Formatierungen und Positionierungen in CSS vorzunehmen.

Eine hundertprozentige Barrierefreiheit kann aufgrund der vielen individuell geprägten Barrieren wahrscheinlich nicht erreicht werden. Es ist jedoch möglich, die Dokumente auf Validität sowie Barrierefreiheit mit Hilfe von Testwerkzeugen zu überprüfen und die Empfehlungen des W3C bei der barrierefreien Gestaltung von Internetseiten umzusetzen.

---

1) Vgl. Bittner 2005 (S. 5)

## 11.2 Chancen und Risiken

Kritisch betrachtet birgt die barrierefreie Gestaltung von Internetseiten sowohl Chancen als auch Risiken. Die im Rahmen dieser Bachelor-Thesis durchgeführten Recherchen und Tests haben gezeigt, dass sich weder Browser- noch Hilfsmittelhersteller vollständig an die Standards halten. Doch je mehr Entwickler und Auftraggeber von Internetseiten sich mit dem Thema Barrierefreiheit auseinandersetzen, sich an die Standards halten und die Richtlinien der WCAG1 oder der BITV befolgen, desto eher wird auch von Seiten der Hersteller von Browsern und Hilfsmitteln eine standardisierte Herangehensweise umgesetzt werden.

Der **Besucher profitiert** von schneller geladenen Dateien, einem für ein breiteres Spektrum an Nutzern und eine größere Anzahl von Geräten zugänglichen Inhalt, je nach Bedürfnissen anzupassenden Seiten und der Möglichkeit druckfreundlicher Versionen für alle Seiten.

Aus Kundensicht sind barrierefreie Seiten leichter zu pflegen, das Layout kann unproblematisch geändert und druckfreundliche Versionen ohne viel Mehraufwand bereitgestellt werden. Dadurch entstehen **weniger Kosten** bei Anpassungen, die Suchmaschinen-Platzierung und die Zugänglichkeit werden verbessert.

Für den Webdesigner oder Webentwickler sind Seiten leichter zu erstellen und zu pflegen, die Entwicklung erfolgt geräteunabhängig. Und noch schafft die barrierefreie Gestaltung von Internetseiten einen **Wettbewerbsvorteil**.

Das Umsetzen barrierefreier Internetseiten kann jedoch durchaus mit einigem Mehraufwand verbunden sein. Entscheider im Unternehmen müssen überzeugt, Mitarbeiter sensibilisiert, geschult und motiviert werden.

Das Erstellen einer komplett neuen Internetpräsenz wird **(vorerst) aufwendiger** sein, dafür nötiges **Wissen muss transferiert werden**. Tests müssen durchgeführt werden, möglicherweise auch von externen Stellen. Inhalte sollten in einfacher Sprache aufbereitet, zusätzliche Texte für Bilder, Animationen und Videos erarbeitet werden. Das Anbieten unterschiedlicher Formatvorlagen erfordert zusätzliche Designs und zur Vermeidung von JavaScript müssen Funktionalitäten überprüft und abgewandelt werden.

### 11.3 Barrierefreiheit in Wirtschaft und Gesellschaft

Das Thema Barrierefreiheit findet zunehmend Gehör in Wirtschaft und Gesellschaft.

Um Internetseiten auf Nutzbarkeit für Menschen mit körperlichen Einschränkungen zu testen, hat die Initiative *Barrierefrei Informieren und Kommunizieren* verschiedene Tests unterschiedlichen Umfangs entwickelt, mit denen die Anforderungen der Barrierefreie[n] Informationstechnik-Verordnung (BITV) überprüft werden.

Alle Tests werden von ausgebildeten Experten durchgeführt, Zweifelsfälle im Team diskutiert. Ergebnis ist eine Sammlung von Fragen und Antworten zum BITV-gerechten Webdesign.<sup>1</sup>

Auch die *Stiftung Digitale Chancen* und die *Aktion Mensch* haben auf Grundlage der BITV ein Bewertungsverfahren entwickelt, um Internetangebote auf Barrierefreiheit zu überprüfen. Mit dem sogenannten **BIENE-Award**<sup>2</sup> wurde eine Wettbewerb ins Leben gerufen, bei dem sich Unternehmen mit barrierefreien Internetseiten in den Kategorien E-Business, E-Government / E-Democracy, Kultur und Gesellschaft, Bildung, Wissenschaft und Forschung sowie Medien bewerben können.<sup>3</sup>

Die eingereichten Internetseiten werden umfangreichen Prüfungen unterzogen, die besten barrierefreien Angebote identifiziert und mit der *BIENE* prämiert.



Abb. 36: BIENE-Award (Quelle: <http://www.digitale-chancen.de/award/wettbewerb.cfm>)

In den vergangenen beiden Jahren haben sich insgesamt mehr als 400 Unternehmen und Organisationen an dem Internet-Wettbewerb beteiligt.

---

1) Vgl. DIAS 2004

2) BIENE = Barrierefreies Internet eröffnet neue Einsichten

3) Vgl. Romppel 2005

## 12 Fazit



Eine barrierefreie Internetpräsenz ist keine kostspielige Angelegenheit, vor allem nicht, wenn Empfehlungen und Standards von Anfang an berücksichtigt und eingehalten werden. „Was jedoch eine Investition erfordert, ist die Aneignung von Wissen, welches in den entsprechenden Ausbildungen zu Konzeptionierern, Designern, Programmierern oder Redakteuren vermittelt werden müsste.“<sup>1</sup>

Eine Ende 2004 vom *Barrierekompass* durchgeführte E-Mail-Befragung von über 70 Hoch- und Fachhochschulen, welchen Stellenwert das Thema Barrierefreiheit in der Ausbildung junger Designer hat, lieferte ein ernüchterndes Ergebnis<sup>2</sup>: nur drei Hochschulen antworteten, kaum eine Hochschule hatte das Thema auf dem Lehrplan.

Zwar hat die Thematik des barrierefreien Internets in den vergangenen Jahren kontinuierlich an Beachtung gewonnen, die Umsetzung in der Praxis leidet jedoch unter Nachwuchsmangel. Generationen von Designern und Programmierern haben in der Vergangenheit ihr Studium absolviert, ohne jemals von Barrierefreiheit, Accessibility und Usability gehört zu haben.

Die barrierefreie Gestaltung von Internetseiten bringt allen Nutzern Vorteile. Egal welche Plattform, egal welcher Browser, egal welches Endgerät: allen Nutzern werden äquivalente Inhalte zur Verfügung gestellt; und quasi nebenbei wird auch für Nutzer mit Behinderungen der Zugang zum Medium Internet geebnet.

„Barrierefreiheit ist keine geheime Kunst, sondern die Motivation, die die Erfindung des Webs erst ermöglichte. [...] Behinderung ist dabei nur ein Aspekt.“<sup>3</sup>

---

1) Hellbusch 2005 a, S. 352  
2) Vgl. Morsbach 2004 (S. 1)  
3) Hellbusch 2005 a, S. 351

## Literaturverzeichnis



### BGBl 2002 a

Bundesgesetzblatt Teil I Nr. 49: „Verordnung zur Schaffung barrierefreier Informationstechnik nach dem Behindertengleichstellungsgesetz (Barrierefreie Informationstechnik-Verordnung - BITV)“, vom 17. Juli 2002, ausgegeben zu Bonn am 23. Juli 2002, S. 2654 - 2662

### BGBl 2002 b

Bundesgesetzblatt Teil I Nr. 28: „Gesetz zur Gleichstellung behinderter Menschen (Behindertengleichstellungsgesetz - BGG)“, vom 27. April 2002, ausgegeben zu Bonn am 30. April 2002, S. 1467 - 1482

### Birkenbihl 2005

Klaus Birkenbihl: „Alle Möglichkeiten des Web erschließen“, W3C Deutsch-Österreichisches Büro, <http://www.w3c.de>, Version vom 09.02.2005

### Bittner 2005

Andreas K. Bittner: „BARRIEREFREIHEIT. Internet auf Krankenschein?“, <http://www.onlinejournalismus.de/praxis/barrierefreiheit.php>, Stand vom 11.01.2005

### Bittner 2004

Andreas K. Bittner: „BITV 2006. Bevor die Claims abgesteckt sind“, <http://www.barrierekompass.de/weblog/index.php?itemid=240>, Stand vom 29.09.2004

### Böhringer et al. 2003

Joachim Böhringer et al.: „Kompendium der Mediengestaltung für Digital- und Printmedien“, 2., überarbeitete und erweiterte Auflage, Springer-Verlag Berlin Heidelberg New York 2003

### Bühler 2004

Prof. Dr.-Ing. Christian Bühler u.a.: „Gesetze & Richtlinien“, Forschungsinstitut Technologie-Behindertenhilfe – FTB, <http://wob11.de/gesetze/index.html>, Version vom 30.09.2004

### DIAS 2004

DIAS GmbH: „Warum ist das wichtig?“, <http://www.bik-online.info/test/zeitungen/pruefschritte.php>, Version vom 02.09.2004

eResult 2005

eResult GmbH: „Nutzungsfreundliche Bezeichnung von Navigationselementen“,  
[http://www.eresult.de/Texte/Wording-Studie%202005\\_Kurzversion.pdf](http://www.eresult.de/Texte/Wording-Studie%202005_Kurzversion.pdf), Version vom 15.02.2005

eResult 2004

eResult GmbH: „Nutzergerechte Gestaltung von Homepages“,  
[http://www.eresult.de/Texte/Imagery\\_2\\_Kurzversion.pdf](http://www.eresult.de/Texte/Imagery_2_Kurzversion.pdf), Version vom 13.12.2004

Freudenfeld 2003

Michaela Freudenfeld: „Blinde und Sehbehinderte am Computer“, <http://www.bik-online.info/info/nutzer/computerhilfsmittel.php>, Version vom 10.02.2003

Hansen/Neumann 2001

Prof. Dr. Hans Robert Hansen, Prof. Dr. Gustaf Neumann: „Wirtschaftsinformatik I“, Lucius & Lucius Verlagsgesellschaft mbH, 8., völlig neubearbeitete und erweiterte Auflage, Stuttgart 2001

Hartmann 2002

René Hartmann: „Zugänglichkeitsrichtlinien für Web-Inhalte 1.0“,  
<http://www.w3c.de/Trans/WAI/webinhalt.html>, Version vom 11.01.2002

Hartmann 2000

René Hartmann: „Checkliste der Checkpunkte zu den Zugänglichkeitsrichtlinien für Web-Inhalte 1.0“, <http://www.w3c.de/Trans/WAI/checkliste.html>, Version vom 09.01.2000

Hein 2004

Ansgar Hein: „JavaScript sinnvoll einsetzen“,  
<http://www.barrierekompass.de/weblog/?itemid=203>, Version vom 08.07.2004

Hellbusch 2005 a

Jan Eric Hellbusch, Christian Bühler (Hrsg.): „Barrierefreies Webdesign“, dpunkt.verlag GmbH, Heidelberg 2005

Hellbusch 2005 b

Jan Eric Hellbusch: „Navigieren ohne Maus - Empfehlungen für Shortcuts auf Ihren Webseiten“,  
<http://www.barrierefreies-webdesign.de/knowhow/accesskey/empfehlung.php>, Version vom 05.03.2005

Jenzer 2004

Sven Jenzer: „CSS – Cascading bedeutet gestufte Verschachtelung“, <http://www.design4all.ch/html/018.html>, Version vom 19.08.2004

Lenk 2004

Volker Lenk: „Die Blindenschrift“, Allgemeiner Blinden- und Sehbehindertenverein Berlin gegr. 1874 e.V., [http://www.absv.de/blindenschrift/braille\\_info.html](http://www.absv.de/blindenschrift/braille_info.html), Version vom 01.11.2004

Meiert 2004

Jens Meiert: „Accessibility-Eigenschaften von CSS“, <http://meiert.com/de/w3/TR/css-access/>, Version vom 14.12.2004

Mintert et al. 2003

Stefan Mintert (Hrsg.), Christine Kühnel, Judith Muhr, Stefan Schumacher: „XHTML, CSS & Co“, Addison-Wesley Verlag, ein Imprint der Pearson Education Deutschland GmbH, München 2003

Morsbach 2004

Jörg Morsbach: „Barrierefreies Webdesign in der Ausbildung“, Barrierekompass, <http://www.barrierekompass.de/weblog/index.php?itemid=272>, Version vom 30.11.2004

Münz/Nefzger 2002

Stefan Münz, Wolfgang Nefzger: „HTML & Web-Publishing Handbuch“, Band 1, Franzis' Verlag GmbH, Poing 2002

Nielsen 2004

Jakob Nielsen: „When Search Engines Become Answer Engines“, <http://www.useit.com/alertbox/20040816.html>, Version vom 16.08.2004

Pfaff 2004

Dipl.-Volkswirt Heiko Pfaff und Mitarbeiterinnen: „Lebenslagen der behinderten Menschen, Ergebnis des Mikrozensus 2003“, in: Wirtschaft und Statistik, Heft 10/2004, Statistisches Bundesamt, Wiesbaden 2004, S. 1181 - 1194

Richter 2004

Dipl. Inf. FH Thomas Richter: „Barrierefreiheit - Webseiten für alle zugänglich machen“, <http://www.lehrer-online.de/url/barrierefreiheit>, Version vom 03.03.2004

Romppel 2005

Rebecca Romppel: „BIENE sucht im Netz die Besten“, Stiftung Digitale Chancen,  
<http://www.digitale-chancen.de/award/wettbewerb.cfm>, Version vom 16.03.2005

Ward 2004

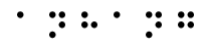
Mark Ward: „A decade of good website design“, BBC News website,  
<http://news.bbc.co.uk/1/hi/technology/4061093.stm>, Version vom 07.12.2004

W3C 2004

World Wide Web Consortium: „Praise for W3C“,  
<http://www.w3.org/Consortium/Prospectus/Testimonials>, Version vom 17.08.2004



# Anhang



- A** Richtlinien der Web Content Accessibility Guidelines 1.0
  
- B** Checkpunkte der Web Content Accessibility Guidelines 1.0
  - B.1 Checkpunkte der Priorität 1
  - B.2 Checkpunkte der Priorität 2
  - B.3 Checkpunkte der Priorität 3
  
- C** Anlage 1 der Barrierefreie[n] Informationstechnik-Verordnung
  - C.1 Anforderungen und Bedingungen der Priorität 1
  - C.2 Anforderungen und Bedingungen der Priorität 2

## A Richtlinien der Web Content Accessibility Guidelines 1.0

Die Web Content Accessibility Guidelines 1.0 enthalten folgende 14 Richtlinien:<sup>1</sup>

1	Stellen Sie äquivalente Alternativen für Audio- und visuellen Inhalt bereit.
2	Verlassen Sie sich nicht auf Farbe allein.
3	Verwenden Sie Markup und Stylesheets und tun Sie dies auf korrekte Weise.
4	Verdeutlichen Sie die Verwendung natürlicher Sprache.
5	Erstellen Sie Tabellen, die geschmeidig transformieren.
6	Sorgen Sie dafür, dass Seiten, die neue Technologien verwenden, geschmeidig transformieren.
7	Sorgen Sie für eine Kontrolle des Benutzers über zeitgesteuerte Änderungen des Inhalts.
8	Sorgen Sie für direkte Zugänglichkeit eingebetteter Benutzerschnittstellen.
9	Wählen Sie ein geräteunabhängiges Design.
10	Verwenden Sie Interim-Lösungen.
11	Verwenden Sie W3C-Technologien und -Richtlinien.
12	Stellen Sie Informationen zum Kontext und zur Orientierung bereit.
13	Stellen Sie klare Navigationsmechanismen bereit.
14	Sorgen Sie dafür, dass Dokumente klar und einfach gehalten sind.

1) Vgl. Hartmann 2002 (S. 3f)

## B Checkpunkte der Web Content Accessibility Guidelines 1.0

Die Web Content Accessibility Guidelines 1.0 enthalten folgende 65 Checkpunkte:<sup>1</sup>

### B.1 Checkpunkte der Priorität 1

<b>Allgemein:</b>	
1.1	Stellen Sie ein Text-Äquivalent für jedes Nicht-Text-Element bereit.
2.1	Sorgen Sie dafür, dass die gesamte mit Farbe dargestellte Information auch ohne Farbe verfügbar ist.
4.1	Machen Sie in klarer Weise Änderungen der natürlichen Sprache des Dokumententexts und sämtlicher Text-Äquivalente kenntlich.
6.1	Bauen Sie Dokumente so auf, dass sie ohne Stylesheets gelesen werden können.
6.2	Sorgen Sie dafür, dass Äquivalente für dynamischen Inhalt aktualisiert werden, wenn sich der dynamische Inhalt ändert.
7.1	Vermeiden Sie Bildschirmflackern, bis Benutzeragenten dem Benutzer eine Kontrolle über das Flackern ermöglichen.
14.1	Verwenden Sie für den Inhalt einer Site die klarste und einfachste Sprache, die angemessen ist.
<b>Und wenn Sie Bilder und Imagemaps verwenden:</b>	
1.2	Stellen Sie redundante Textlinks für jede aktive Region einer Server-seitigen Imagemap bereit.
9.1	Stellen Sie Client-seitige anstelle von Server-seitigen Imagemaps bereit, außer wenn die Regionen mit den verfügbaren geometrischen Formen nicht definiert werden können.
<b>Und wenn Sie Tabellen verwenden:</b>	
5.1	Kennzeichnen Sie bei Datentabellen Zeilen- und Spaltenüberschriften.
5.2	Wenn Datentabellen zwei oder mehr logische Ebenen von Zeilen- oder Spaltenüberschriften haben, verwenden Sie Markup, um Datenzellen und Überschriftenzellen einander zuzuordnen.
<b>Und wenn Sie Frames verwenden:</b>	
12.1	Betiteln Sie jeden Frame, um Navigation und Identifikation zu erleichtern.
<b>Und wenn Sie Applets und Scripts verwenden:</b>	
6.3	Sorgen Sie dafür, dass Seiten verwendbar sind, wenn Scripts, Applets oder andere programmierte Objekte abgeschaltet sind oder nicht unterstützt werden. Ist dies nicht möglich, stellen Sie äquivalente Information auf einer alternativen zugänglichen Seite bereit.
<b>Und wenn Sie Multimedia verwenden:</b>	
1.3	Stellen Sie eine Audio-Beschreibung der wichtigen Information der Videospur einer Multimedia-Präsentation bereit, bis Benutzeragenten das Text-Äquivalent einer Videospur vorlesen können.
1.4	Synchronisieren Sie für jede zeitgesteuerte Multimedia-Präsentation äquivalente Alternativen mit der Präsentation.
<b>Und wenn alles andere fehlschlägt:</b>	
11.4	Wenn Sie auch nach besten Bemühungen keine zugängliche Seite erstellen können, stellen Sie einen Link auf eine alternative Seite bereit, die W3C-Technologien verwendet, zugänglich ist, äquivalente Information (oder Funktionalität) enthält und ebenso oft aktualisiert wird wie die nicht zugängliche (originale) Seite.

1) Vgl. Hartmann 2000 (S. 3ff)

**B.2 Checkpunkte der Priorität 2**

<b>Allgemein:</b>	
2.2	Sorgen Sie dafür, dass die Kombinationen aus Vordergrund- und Hintergrundfarbe ausreichend kontrastieren, wenn sie von jemandem betrachtet werden, dessen Farbsehen beeinträchtigt ist, oder wenn sie mit einem Schwarzweißbildschirm betrachtet werden.
3.1	Wenn eine angemessene Markup-Sprache existiert, verwenden Sie Markup anstelle von Bildern, um Information darzustellen.
3.2	Erstellen Sie Dokumente, die gegen veröffentlichte formale Grammatiken validieren.
3.3	Verwenden Sie Stylesheets, um Layout und Präsentation zu beeinflussen.
3.4	Verwenden Sie relative anstelle von absoluten Einheiten in den Attributwerten der Markup-Sprache und Stylesheet-Property-Werten.
3.5	Verwenden Sie Überschriften-Elemente, um die Struktur eines Dokuments darzustellen und verwenden Sie sie gemäß der Spezifikation.
3.6	Verwenden Sie korrekten Markup für Listen und Listenelemente.
3.7	Verwenden Sie Markup für Zitate. Verwenden Sie keinen Markup, der für Zitate gedacht ist, um visuelle Effekte wie Einrückung zu erzielen.
6.5	Sorgen Sie dafür, dass dynamischer Inhalt zugänglich ist oder stellen Sie eine alternative Präsentation oder Seite bereit.
7.2	Bis Benutzeragenten eine Kontrolle über das Blinken ermöglichen, vermeiden Sie es, Inhalt blinken zu lassen.
7.4	Bis Benutzeragenten es zulassen, den Refresh zu stoppen, erstellen Sie keine Seiten mit automatischer periodischer Aktualisierung.
7.5	Bis Benutzeragenten es zulassen, die automatische Weiterleitung (Redirect) zu stoppen, verwenden Sie keinen Markup, um eine Weiterleitung zu erzielen.
10.1	Lassen Sie keine Pop-Ups oder andere Fenster erscheinen und wechseln Sie das aktuelle Fenster nicht, ohne den Benutzer zu informieren, bis Benutzeragenten es gestatten, die Erzeugung neuer Fenster zu unterbinden.
11.1	Verwenden Sie W3C-Technologien, wenn sie verfügbar und der Aufgabe angemessen sind und benutzen Sie die neueste Version, wenn sie unterstützt wird.
11.2	Vermeiden Sie überholte Features von W3C-Technologien.
12.3	Unterteilen Sie große Informationsblöcke in leichter zu handhabende Gruppen, wo angebracht.
13.1	Identifizieren Sie das Ziel jedes Links auf klare Weise.
13.2	Stellen Sie Metadaten bereit, um semantische Information zu Seiten und Sites hinzuzufügen.
13.3	Stellen Sie Informationen zum allgemeinen Layout einer Site bereit (z.B. über eine Sitemap oder ein Inhaltsverzeichnis).
13.4	Verwenden Sie Navigationsmechanismen in konsistenter Weise.
<b>Und wenn Sie Tabellen verwenden:</b>	
5.3	Verwenden Sie keine Tabellen für Layout, wenn diese in linearisierter Form keinen Sinn ergeben. Ansonsten, wenn die Tabelle keinen Sinn ergibt, stellen Sie ein alternatives Äquivalent bereit (das eine linearisierte Version sein kann).
5.4	Wenn eine Tabelle für Layout verwendet wurde, verwenden Sie keinen Struktur-Markup zum Zweck der visuellen Formatierung.
<b>Und wenn Sie Frames verwenden:</b>	
12.2	Beschreiben Sie den Zweck von Frames und ihre Beziehung untereinander, wenn dies aus den Titeln allein nicht ersichtlich wird.

<b>Und wenn Sie Formulare verwenden (Priorität 2):</b>	
10.2	Sorgen Sie bei allen Formular-Kontrollelementen mit implizit zugeordneten Beschriftungen dafür, dass die Beschriftung korrekt positioniert ist, bis Benutzeragenten eine explizite Zuordnung von Beschriftung und Formular-Kontrollelement ermöglichen.
12.4	Ordnen Sie Beschriftungen explizit ihren Kontrollelementen zu.
<b>Und wenn Sie Applets und Scripts verwenden:</b>	
6.4	Sorgen Sie dafür, dass die Eingabebehandlung von Scripts und Applets vom Eingabegerät unabhängig ist.
7.3	Vermeiden Sie Bewegung in Seiten, bis Benutzeragenten das Einfrieren von Bewegung ermöglichen.
8.1	Machen Sie programmierte Elemente wie Scripts und Applets direkt zugänglich oder kompatibel mit assistiven Technologien
9.2	Sorgen Sie dafür, dass jedes Element, das über seine eigene Schnittstelle verfügt, in geräteunabhängiger Weise bedient werden kann.
9.3	Spezifizieren Sie in Scripts logische Event-Handler anstelle von geräteabhängigen Event-Handlern.

### B.3 Checkpunkte der Priorität 3

<b>Allgemein:</b>	
4.2	Spezifizieren Sie die Ausschreibung jeder Abkürzung und jedes Akronyms an der Stelle des ersten Auftretens.
4.3	Machen Sie die vorherrschende natürliche Sprache des Dokuments kenntlich.
9.4	Definieren Sie eine logische Tab-Reihenfolge für Links, Formular-Kontrollelemente und Objekte.
9.5	Stellen Sie Tastatur-Kurzbefehle (Shortcuts) für wichtige Links (einschließlich solcher in Client-seitigen Imagemaps), Formular-Kontrollelemente und Gruppen von Formular-Kontrollelementen bereit.
10.5	Bis Benutzeragenten (einschließlich assistiver Technologien) beieinanderliegende Links getrennt darstellen, platzieren Sie druckbare Zeichen, die nicht zu einem Link gehören, umgeben von Leerzeichen, zwischen Links.
11.3	Stellen Sie Informationen bereit, so dass Benutzer Dokumente entsprechend ihren Vorgaben (Sprache, Typ usw.) erhalten können.
13.5	Stellen Sie Navigationsleisten bereit, um den Navigationsmechanismus hervorzuheben und einen Zugriff darauf zu ermöglichen.
13.6	Gruppieren Sie verwandte Links, identifizieren Sie die Gruppe (für Benutzeragenten), und ermöglichen Sie das Überspringen der Gruppe, bis Benutzeragenten dies gestatten.
13.7	Wenn Suchfunktionen verfügbar sind, stellen Sie verschiedene Arten der Suche bereit, je nach den Fähigkeiten und Vorlieben der Benutzer.
13.8	Platzieren Sie unterscheidungskräftige Information an den Anfang von Überschriften, Absätzen, Listen usw.
13.9	Stellen Sie Informationen über Zusammenstellungen von Dokumenten bereit (z.B. Dokumente, die aus mehreren Seiten bestehen usw.)
13.10	Ermöglichen Sie das Überspringen von mehrzeiligen ASCII-Zeichnungen.
14.2	Ergänzen Sie Text mit grafischen oder Audio-Präsentationen, wo dies das Verständnis der Seite erleichtert.
14.3	Verwenden Sie einen Präsentationsstil, der über Seiten hinweg konsistent ist.

---

<b>Und wenn Sie Bilder und Imagemaps verwenden (Priorität 3):</b>	
1.5	Bis Benutzeragenten Text-Äquivalente für Client-seitige Imagemaps darstellen, stellen Sie für jede aktive Region einer Client-seitigen Imagemap einen redundanten Textlink bereit.
<b>Und wenn Sie Tabellen verwenden:</b>	
5.5	Stellen Sie Zusammenfassungen für Tabellen bereit.
5.6	Stellen Sie Abkürzungen für Überschriften bereit.
10.3	Stellen Sie eine lineare Text-Alternative für alle Tabellen bereit, die Text in parallelen Spalten mit Zeilenumbruch enthalten, bis Benutzeragenten nebeneinander angeordneten Text korrekt behandeln.
<b>Und wenn Sie Formulare verwenden:</b>	
10.4	Bis Benutzeragenten leere Kontrollelemente korrekt behandeln, besetzen Sie Felder mit Platzhalter-Zeichen vor.

## C Anlage 1 der Barrierefreie[n] Informationstechnik-Verordnung

In der Anlage 1 enthält die Barrierefreie Informationstechnik-Verordnung folgende Anforderungen und Bedingungen:<sup>1</sup>

### C.1 Anforderungen und Bedingungen der Priorität 1

Anforderung 1	Für jeden Audio- oder visuellen Inhalt sind geeignete äquivalente Inhalte bereitzustellen, die den gleichen Zweck oder die gleiche Funktion wie der originäre Inhalt erfüllen.
Bedingung 1.1	Für jedes Nicht-Text-Element ist ein äquivalenter Text bereitzustellen.
Bedingung 1.2	Für jede aktive Region einer serverseitigen Imagemap sind redundante Text-hyperlinks bereitzustellen.
Bedingung 1.3	Für Multimedia-Präsentationen ist eine Audio-Beschreibung der wichtigen Informationen der Videospur bereitzustellen.
Bedingung 1.4	Für jede zeitgesteuerte Multimedia-Präsentation sind äquivalente Alternativen mit der Präsentation zu synchronisieren.
Anforderung 2	Texte und Graphiken müssen auch dann verständlich sein, wenn sie ohne Farbe betrachtet werden.
Bedingung 2.1	Alle mit Farbe dargestellten Informationen müssen auch ohne Farbe verfügbar sein, z.B. durch den Kontext oder die hierfür vorgesehenen Elemente der verwendeten Markup-Sprache.
Bedingung 2.2	Bilder sind so zu gestalten, dass die Kombinationen aus Vordergrund- und Hintergrundfarbe auf einem Schwarz-Weiß-Bildschirm und bei der Betrachtung durch Menschen mit Farbfehlsichtigkeiten ausreichend kontrastieren.
Anforderung 3	Markup-Sprachen und Stylesheets sind entsprechend ihrer Spezifikationen und formalen Definitionen zu verwenden.
Bedingung 3.1	Soweit eine angemessene Markup-Sprache existiert, ist diese anstelle von Bildern zu verwenden, um Informationen darzustellen.
Bedingung 3.2	Mittels Markup-Sprachen geschaffene Dokumente sind so zu erstellen und zu deklarieren, dass sie gegen veröffentlichte formale Grammatiken validieren.
Bedingung 3.3	Es sind Stylesheets zu verwenden, um die Text- und Bildgestaltung sowie die Präsentation von mittels Markup-Sprachen geschaffener Dokumente zu beeinflussen.
Bedingung 3.4	Es sind relative anstelle von absoluten Einheiten in den Attributwerten der verwendeten Markup-Sprache und den Stylesheet-Property-Werten zu verwenden.
Bedingung 3.5	Zur Darstellung der Struktur von mittels Markup-Sprachen geschaffener Dokumente sind Überschriften-Elemente zu verwenden.
Bedingung 3.6	Zur Darstellung von Listen und Listenelementen sind die hierfür vorgesehenen Elemente der verwendeten Markup-Sprache zu verwenden.
Bedingung 3.7	Zitate sind mittels der hierfür vorgesehenen Elemente der verwendeten Markup-Sprache zu kennzeichnen.
Anforderung 4	Sprachliche Besonderheiten wie Wechsel der Sprache oder Abkürzungen sind erkennbar zu machen.
Bedingung 4.1	Wechsel und Änderungen der vorherrschend verwendeten natürlichen Sprache sind kenntlich zu machen.

1) Vgl. BGBl 2002 a, S. 2655ff

Anforderung 5	Tabellen sind mittels der vorgesehenen Elemente der verwendeten Markup-Sprache zu beschreiben und in der Regel nur zur Darstellung tabellarischer Daten zu verwenden.
Bedingung 5.1	In Tabellen, die tabellarische Daten darstellen, sind die Zeilen- und Spaltenüberschriften mittels der vorgesehenen Elemente der verwendeten Markup-Sprache zu kennzeichnen.
Bedingung 5.2	Soweit Tabellen, die tabellarische Daten darstellen, zwei oder mehr Ebenen von Zeilen- und Spaltenüberschriften aufweisen, sind mittels der vorgesehenen Elemente der verwendeten Markup-Sprache Datenzellen und Überschriftenzellen einander zuzuordnen.
Bedingung 5.3	Tabellen sind nicht für die Text- und Bildgestaltung zu verwenden, soweit sie nicht auch in linearisierter Form dargestellt werden können.
Bedingung 5.4	Soweit Tabellen zur Text- und Bildgestaltung genutzt werden, sind keine der Strukturierung dienenden Elemente der verwendeten Markup-Sprache zur visuellen Formatierung zu verwenden.
Anforderung 6	Internetangebote müssen auch dann nutzbar sein, wenn der verwendete Benutzeragent neuere Technologien nicht unterstützt oder diese deaktiviert sind.
Bedingung 6.1	Es muss sichergestellt sein, dass mittels Markup-Sprachen geschaffene Dokumente verwendbar sind, wenn die zugeordneten Stylesheets deaktiviert sind.
Bedingung 6.2	Es muss sichergestellt sein, dass Äquivalente für dynamischen Inhalt aktualisiert werden, wenn sich der dynamische Inhalt ändert.
Bedingung 6.3	Es muss sichergestellt sein, dass mittels Markup-Sprachen geschaffene Dokumente verwendbar sind, wenn Scripts, Applets oder andere programmierte Objekte deaktiviert sind.
Bedingung 6.4	Es muss sichergestellt sein, dass die Eingabebehandlung von Scripts, Applets oder anderen programmierten Objekten vom Eingabegerät unabhängig ist.
Bedingung 6.5	Dynamische Inhalte müssen zugänglich sein. Insoweit dies nur mit unverhältnismäßig hohem Aufwand zu realisieren ist, sind gleichwertige alternative Angebote unter Verzicht auf dynamische Inhalte bereitzustellen.
Anforderung 7	Zeitgesteuerte Änderungen des Inhalts müssen durch die Nutzerin/den Nutzer kontrollierbar sein.
Bedingung 7.1	Bildschirmflackern ist zu vermeiden.
Bedingung 7.2	Blinkender Inhalt ist zu vermeiden.
Bedingung 7.3	Bewegung in mittels Markup-Sprachen geschaffener Dokumente ist entweder zu vermeiden oder es sind Mechanismen bereitzustellen, die der Nutzerin/dem Nutzer das Einfrieren der Bewegung oder die Änderung des Inhalts ermöglichen.
Bedingung 7.4	Automatische periodische Aktualisierungen in mittels Markup-Sprachen geschaffener Dokumente sind zu vermeiden.
Bedingung 7.5	Die Verwendung von Elementen der Markup-Sprache zur automatischen Weiterleitung ist zu vermeiden.
Anforderung 8	Die direkte Zugänglichkeit der in Internetangeboten eingebetteten Benutzerschnittstellen ist sicherzustellen.
Bedingung 8.1	Programmierte Elemente (insbesondere Scripts und Applets) sind so zu gestalten, dass sie entweder direkt zugänglich oder kompatibel mit assistiven Technologien sind.
Anforderung 9	Internetangebote sind so zu gestalten, dass Funktionen unabhängig vom Eingabegerät oder Ausgabegerät nutzbar sind.
Anforderung 9.1	Es sind clientseitige Imagemaps bereitzustellen, es sei denn, die Regionen können mit den verfügbaren geometrischen Formen nicht definiert werden.



Bedingung 9.2	Jedes über eine eigene Schnittstelle verfügende Element muss in geräte-unabhängiger Weise bedient werden können.
Bedingung 9.3	In Scripts sind logische anstelle von geräteabhängigen Event-Handlern zu spezifizieren.
Anforderung 10	Die Verwendbarkeit von nicht mehr dem jeweils aktuellen Stand der Technik entsprechenden assistiven Technologien und Browsern ist sicherzustellen, so weit der hiermit verbundene Aufwand nicht unverhältnismäßig ist.
Bedingung 10.1	Das Erscheinenlassen von Pop-Ups oder anderen Fenstern ist zu vermeiden. Die Nutzerin/der Nutzer ist über Wechsel der aktuellen Ansicht zu informieren.
Bedingung 10.2	Bei allen Formular-Kontrollelementen mit implizit zugeordneten Beschriftungen ist dafür Sorge zu tragen, dass die Beschriftungen korrekt positioniert sind.
Anforderung 11	Die zur Erstellung des Internetangebots verwendeten Technologien sollen öffentlich zugänglich und vollständig dokumentiert sein, wie z.B. die vom World Wide Web Consortium entwickelten Technologien.
Bedingung 11.1	Es sind öffentlich zugängliche und vollständig dokumentierte Technologien in ihrer jeweils aktuellen Version zu verwenden, soweit dies für die Erfüllung der angestrebten Aufgabe angemessen ist.
Bedingung 11.2	Die Verwendung von Funktionen, die durch die Herausgabe neuer Versionen überholt sind, ist zu vermeiden.
Bedingung 11.3	Soweit auch nach bestem Bemühen die Erstellung eines barrierefreien Internetangebots nicht möglich ist, ist ein alternatives, barrierefreies Angebot zur Verfügung zu stellen, das äquivalente Funktionalitäten und Informationen gleicher Aktualität enthält, soweit es die technischen Möglichkeiten zulassen. Bei Verwendung nicht barrierefreier Technologien sind diese zu ersetzen, sobald aufgrund der technologischen Entwicklung äquivalente, zugängliche Lösungen verfügbar und einsetzbar sind.
Anforderung 12	Der Nutzerin/dem Nutzer sind Informationen zum Kontext und zur Orientierung bereitzustellen.
Bedingung 12.1	Jeder Frame ist mit einem Titel zu versehen, um Navigation und Identifikation zu ermöglichen.
Bedingung 12.2	Der Zweck von Frames und ihre Beziehung zueinander ist zu beschreiben, soweit dies nicht aus den verwendeten Titeln ersichtlich ist.
Bedingung 12.3	Große Informationsblöcke sind mittels Elementen der verwendeten Markup-Sprache in leichter handhabbare Gruppen zu unterteilen.
Bedingung 12.4	Beschriftungen sind genau ihren Kontrollelementen zuzuordnen.
Anforderung 13	Navigationsmechanismen sind übersichtlich und schlüssig zu gestalten.
Bedingung 13.1	Das Ziel jedes Hyperlinks muss auf eindeutige Weise identifizierbar sein.
Bedingung 13.2	Es sind Metadaten bereitzustellen, um semantische Informationen zu Internetangeboten hinzuzufügen.
Bedingung 13.3	Es sind Informationen zur allgemeinen Anordnung und Konzeption eines Internetangebots, z.B. mittels eines Inhaltsverzeichnisses oder einer Sitemap, bereitzustellen.
Bedingung 13.4	Navigationsmechanismen müssen schlüssig und nachvollziehbar eingesetzt werden.
Anforderung 14	Das allgemeine Verständnis der angebotenen Inhalte ist durch angemessene Maßnahmen zu fördern.
Bedingung 14.1	Für jegliche Inhalte ist die klarste und einfachste Sprache zu verwenden, die angemessen ist.

## C.2 Anforderungen und Bedingungen der Priorität 2

Anforderung 1	Für jeden Audio- oder visuellen Inhalt sind geeignete äquivalente Inhalte bereitzustellen, die den gleichen Zweck oder die gleiche Funktion wie der originäre Inhalt erfüllen.
Bedingung 1.5	Für jede aktive Region einer clientseitigen Imagemap sind redundante Text-hyperlinks bereitzustellen.
Anforderung 2	Texte und Graphiken müssen auch dann verständlich sein, wenn sie ohne Farbe betrachtet werden.
Bedingung 2.3	Texte sind so zu gestalten, dass die Kombinationen aus Vordergrund- und Hintergrundfarbe auf einem Schwarz-Weiß-Bildschirm und bei der Betrachtung durch Menschen mit Farbfehlsichtigkeiten ausreichend kontrastieren.
Anforderung 3	Markup-Sprachen und Stylesheets sind entsprechend ihrer Spezifikationen und formalen Definitionen zu verwenden.
Anforderung 4	Sprachliche Besonderheiten wie Wechsel der Sprache oder Abkürzungen sind erkennbar zu machen.
Bedingung 4.2	Abkürzungen und Akronyme sind an der Stelle ihres ersten Auftretens im Inhalt zu erläutern und durch die hierfür vorgesehenen Elemente der verwendeten Markup-Sprache kenntlich zu machen.
Bedingung 4.3	Die vorherrschend verwendete natürliche Sprache ist durch die hierfür vorgesehenen Elemente der verwendeten Markup-Sprache kenntlich zu machen.
Anforderung 5	Tabellen sind mittels der vorgesehenen Elemente der verwendeten Markup-Sprache zu beschreiben und in der Regel nur zur Darstellung tabellarischer Daten zu verwenden.
Bedingung 5.5	Für Tabellen sind unter Verwendung der hierfür vorgesehenen Elemente der genutzten Markup-Sprache Zusammenfassungen bereitzustellen.
Bedingung 5.6	Für Überschriftenzellen sind unter Verwendung der hierfür vorgesehenen Elemente der genutzten Markup-Sprache Abkürzungen bereitzustellen.
Anforderung 6	Internetangebote müssen auch dann nutzbar sein, wenn der verwendete Benutzeragent neuere Technologien nicht unterstützt oder diese deaktiviert sind.
Anforderung 7	Zeitgesteuerte Änderungen des Inhalts müssen durch die Nutzerin/den Nutzer kontrollierbar sein.
Anforderung 8	Die direkte Zugänglichkeit der in Internetangeboten eingebetteten Benutzerschnittstellen ist sicherzustellen.
Anforderung 9	Internetangebote sind so zu gestalten, dass Funktionen unabhängig vom Eingabegerät oder Ausgabegerät nutzbar sind.
Bedingung 9.4	Es ist eine mit der Tabulatortaste navigierbare, nachvollziehbare und schlüssige Reihenfolge von Hyperlinks, Formularkontrollelementen und Objekten festzulegen.
Bedingung 9.5	Es sind Tastaturkurzbefehle für Hyperlinks, die für das Verständnis des Angebots von entscheidender Bedeutung sind, Formularkontrollelemente und Gruppen von Formularkontrollelementen bereitzustellen.
Anforderung 10	Die Verwendbarkeit von nicht mehr dem jeweils aktuellen Stand der Technik entsprechenden assistiven Technologien und Browsern ist sicherzustellen, so weit der hiermit verbundene Aufwand nicht unverhältnismäßig ist.
Bedingung 10.3	Für alle Tabellen, die Text in parallelen Spalten mit Zeilenumbruch enthalten, ist alternativ linearer Text bereitzustellen.
Bedingung 10.4	Leere Kontrollelemente in Eingabefeldern und Textbereichen sind mit Platzhalterzeichen zu versehen.
Bedingung 10.5	Nebeneinander liegende Hyperlinks sind durch von Leerzeichen umgebene, druckbare Zeichen zu trennen.

Anforderung 11	Die zur Erstellung des Internetangebots verwendeten Technologien sollen öffentlich zugänglich und vollständig dokumentiert sein, wie z.B. die vom World Wide Web Consortium entwickelten Technologien.
Bedingung 11.4	Der Nutzerin/dem Nutzer sind Informationen bereitzustellen, die es ihnen erlauben, Dokumente entsprechend ihren Vorgaben (z.B. Sprache) zu erhalten.
Anforderung 12	Der Nutzerin/dem Nutzer sind Informationen zum Kontext und zur Orientierung bereitzustellen.
Anforderung 13	Navigationsmechanismen sind übersichtlich und schlüssig zu gestalten.
Bedingung 13.5	Es sind Navigationsleisten bereitzustellen, um den verwendeten Navigationsmechanismus hervorzuheben und einen Zugriff darauf zu ermöglichen.
Bedingung 13.6	Inhaltlich verwandte oder zusammenhängende Hyperlinks sind zu gruppieren. Die Gruppen sind eindeutig zu benennen und müssen einen Mechanismus enthalten, der das Umgehen der Gruppe ermöglicht.
Bedingung 13.7	Soweit Suchfunktionen angeboten werden, sind der Nutzerin/dem Nutzer verschiedene Arten der Suche bereitzustellen.
Bedingung 13.8	Es sind aussagekräftige Informationen am Anfang von inhaltlich zusammenhängenden Informationsblöcken (z.B. Absätzen, Listen) bereitzustellen, die eine Differenzierung ermöglichen.
Bedingung 13.9	Soweit inhaltlich zusammenhängende Dokumente getrennt angeboten werden, sind Zusammenstellungen dieser Dokumente bereitzustellen.
Bedingung 13.10	Es sind Mechanismen zum Umgehen von ASCII-Zeichnungen bereitzustellen.
Anforderung 14	Das allgemeine Verständnis der angebotenen Inhalte ist durch angemessene Maßnahmen zu fördern.
Bedingung 14.2	Text ist mit graphischen oder Audio-Präsentationen zu ergänzen, sofern dies das Verständnis der angebotenen Information fördert.
Bedingung 14.3	Der gewählte Präsentationsstil ist durchgängig beizubehalten.

## **Erklärung**

Ich versichere, dass ich die vorliegende Bachelor-Thesis selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Die Arbeit hat keiner anderen Prüfungsbehörde vorgelegen.

Berlin, den

---

Unterschrift