

---

# Erstellen eines für Sehbehinderte optimierten Werkzeugs zur Erstellung von barrierefreien Internetangeboten

**Diplomarbeit**

zur Erlangung des akademischen Grades

**Diplom-Informatiker / Diplom-Informatikerin**

an der Fachhochschule für Technik und Wirtschaft Berlin

Fachbereich Wirtschaftswissenschaften II

Studiengang Angewandte Informatik

**Eingereicht bei** 1. Betreuer Prof. Dr.-Ing. Thomas Jung

2. Betreuer Prof. Dr.-Ing. Dierk Langbein

**Eingereicht von** Daniela Bonnert, Matrikelnummer s0504724

Fachsemester 10, Studienschwerpunkt Multimedia

**Eingereicht zum** 25.04.2005

**Daniela Bonnert**

Bouchéstraße 19

12435 Berlin

Telefon 030-25768034

Mobil 0163-5772200

Mail dannibonnert@gmx.net

---

---

# Inhaltsverzeichnis

<b>1 Einleitung.....</b>	<b>1</b>
1.1 Motivation.....	1
1.2 Ziel der Arbeit.....	2
1.3 Aufbau der Arbeit .....	4
<b>2 Behinderung.....</b>	<b>6</b>
2.1 Körperbehinderung.....	7
2.2 Intelligenzbehinderung.....	8
2.3 Sinnesbehinderungen.....	8
<b>3 Sehbehinderung.....</b>	<b>9</b>
3.1 Sehbehinderung und hochgradige Sehbehinderung.....	10
3.2 Blindheit.....	10
3.3 Farbenblindheit und -fehlsichtigkeit.....	11
3.4 Sehbehinderte im Umgang mit dem Computer.....	11
3.4.1 Ausgabemöglichkeiten.....	12
3.4.2 Eingabemöglichkeiten.....	16
3.4.3 Vorteile der Computerbenutzung.....	17
<b>4 Barrierefreie Internetauftritte.....</b>	<b>19</b>
4.1 Rechtliche Grundlagen und Standards.....	20
4.1.1 Behindertengleichstellungsgesetz.....	20

---

4.1.2	Barrierefreie Informationstechnik-Verordnung.....	21
4.1.3	Web Content Accessibility Guidelines (WCAG).....	22
4.1.4	BITV - WCAG 1.0.....	24
4.1.5	Anforderungen der BITV .....	25
<b>5</b>	<b>Gebrauchstaugliche Internetauftritte.....</b>	<b>41</b>
5.1	Gebrauchstauglichkeit.....	43
5.2	Benutzerverhalten im Internet.....	44
5.2.1	Lesen von Texten am Bildschirm .....	44
5.2.2	Leseverhalten im Internet.....	48
5.2.3	Ladezeiten im Internet.....	50
5.2.4	Struktur eines Internetauftritts.....	52
5.2.5	Konventionen im Internet.....	53
5.2.6	Unterstützung des menschlichen Gehirns.....	55
<b>6</b>	<b>Werkzeuge zur Erstellung von Internetangeboten....</b>	<b>56</b>
6.1	(X)HTML-Editor.....	56
6.2	Content Management System.....	58
<b>7</b>	<b>Datenhaltung.....</b>	<b>60</b>
7.1	XML.....	60
7.1.1	XML-Dokumente.....	61
7.1.2	Document Type Definition.....	64
7.1.3	XML-Schema.....	68
7.1.4	Verarbeitung von XML-Dokumenten.....	71
7.1.5	Persistieren von XML-Daten.....	73
7.2	Datenbanken.....	74

---

<b>8 Analyse.....</b>	<b>76</b>
8.1 Zielsetzung.....	76
8.2 Anwendungsfälle des Werkzeugs.....	77
8.3 Anforderungen an das Werkzeug.....	80
<b>9 Design.....</b>	<b>84</b>
9.1 Lokal oder Internet? .....	84
9.2 Möglichkeiten zur Datenhaltung.....	86
9.3 Die Programmiersprache.....	87
9.4 Gesamtarchitektur des Werkzeugs.....	89
9.5 Die Arbeit mit dem Werkzeug.....	92
<b>10 Implementierung.....</b>	<b>94</b>
10.1 Model: Aufbau der XML-Dokumente.....	94
10.2 View: (X)HTML-Vorlagen.....	97
10.3 Controller: Umsetzung in PHP5.....	103
<b>11 Test.....</b>	<b>107</b>
<b>12 Zusammenfassung und Ausblick.....</b>	<b>108</b>
12.1 Ergebnisse.....	108
12.2 Mögliche Erweiterungen.....	110

---

# Abbildungsverzeichnis

**Seite 13:** Bildschirmansicht in normaler Größe und bei vierfacher Vergrößerung

**Seite 15:** Benutzerin an einem Arbeitsplatz mit Braillezeile

**Seite 51:** Bewertungen der Antwortzeiten von Internetseiten [BKB]

**Seite 54:** Erwartungen bezüglich des Verweises zur Startseite (links) und bezüglich interner Verweise (rechts) [vgl. BS]

**Seite 77:** UML-Anwendungsfalldiagramm

**Seite 80:** Standardablauf "Seite bearbeiten" als UML-Aktivitätsdiagramm

**Seite 90:** Darstellung einer 3-tier-Architektur

**Seite 91:** Klassendiagramm der Geschäftslogik

**Seite 92:** Darstellung eines Internetangebots mit unterschiedlichen Stylesheet-Dateien

**Seite 97:** Aufbau der (X)HTML-Vorlage

**Seite 98:** Ansicht des Internetangebots

**Seite 99:** Vergrößerung der Schriftgröße durch den Internetbrowser

**Seite 101:** Navigation in der Entwickleransicht mit und ohne Verwendung eines Stylesheets

---

# Abkürzungsverzeichnis

<b>ASCII:</b>	American Standard Code for Information Interchange
<b>BGG:</b>	Gesetz zur Gleichstellung behinderter Menschen, Bundesgleichstellungsgesetz
<b>BITV:</b>	Verordnung zur Schaffung barrierefreier Informationstechnik nach dem Behindertengleichstellungsgesetz, Barrierefreie Informationstechnik-Verordnung
<b>CMS:</b>	Content Management System
<b>CSS:</b>	Cascading Style Sheets
<b>dpi:</b>	Dots per inch
<b>EJB:</b>	Enterprise JavaBeans
<b>FTP:</b>	File Transfer Protocol
<b>HTML:</b>	HyperText Markup Language
<b>HTTP:</b>	Hypertext Transfer Protocol
<b>JSP:</b>	Java Server Pages
<b>PHP:</b>	PHP: Hypertext Preprocessor
<b>MIT/LCS:</b>	Massachusetts Institute of Technology, Laboratory for Computer Science
<b>MVC:</b>	Model-View-Controller
<b>UML:</b>	Unified Modeling Language
<b>W3C:</b>	World Wide Web Consortium
<b>WAI:</b>	Web Accessibility Initiative
<b>WCAG:</b>	Web Content Accessibility Guidelines
<b>XHTML:</b>	eXtensible HyperText Markup Language
<b>XML:</b>	eXtensible Markup Language

# 1 Einleitung

Vorab möchte ich die Leser der vorliegenden Diplomarbeit darauf aufmerksam machen, dass einer besseren Lesbarkeit zugute auf geschlechtsspezifische Unterscheidungen verzichtet wird. Wird bei möglichen Unterscheidungen nur die männliche Form angeführt, gilt diese immer für Frau und Mann in gleicher Weise.

## 1.1 Motivation

Bis zu meinem Praxissemester war mir das Thema *barrierefreie Informations-technik* völlig unbekannt. Während des Praktikums wurde ich zum ersten Mal mit einer blinden Frau konfrontiert, deren Aufgabe darin bestand, Internetangebote auf ihre Zugänglichkeit hin zu überprüfen. Da es jenseits meiner Vorstellung lag, wie blinde Menschen mit dem Computer arbeiten, war das Thema sofort sehr interessant für mich. Im Internet konnte ich meinen Wissensdurst stillen. Neben rein fachlichen Informationen fand ich auch ein Internetangebot [Rai], in welchem sehbehinderte und blinde Menschen ihre alltäglichen Erfahrungen in Form von Kurzgeschichten veröffentlichen. Die Autoren beschreiben sehr ausdrucksstark, welche Probleme für sie bestehen und wie sie diese meistern. Für mich war das Lesen dieser Geschichten sehr eindrucksvoll.

Nach dem Besuch mehrerer Internetangebote, die von blinden Menschen erstellt wurden, eröffnete sich mir eine neue Problematik. Die Internetangebote sind für sehbehinderte und blinde Menschen zugänglich, aber gilt dies auch für Sehende? Sehende Menschen haben, verwöhnt durch das Informationsüberangebot im Internet, bestimmte Anforderungen an Internetangebote entwickelt. Bei schwarzem Hintergrund mit weißer Schrift und einer am Seitenende untergebrachten Navigation verlieren sehende Menschen schnell das Interesse, sich näher mit den Inhalten zu beschäftigen. Aus dieser Schwierigkeit heraus entwickelte sich das Thema der vorliegenden Diplomarbeit.

Zum einen soll in dieser Arbeit herausgearbeitet werden, was *Barrierefreiheit* bedeutet und wie diese mit *Gebrauchstauglichkeit* zu vereinen ist. Zum anderen wird ein Werkzeug erstellt, das es blinden Menschen ermöglicht, eigenständig Internetangebote zu gestalten, die für blinde, sehbehinderte und sehende Menschen gleichermaßen attraktiv sind.

Bei der Erwähnung des Wortes *barrierefrei* fallen vielen Menschen bisher lediglich die sichtbaren baulichen Anpassungen ein, durch die verschiedene Bereiche des öffentlichen Raumes behinderten Menschen zugänglich gemacht werden. Besonders bekannt sind hier Rampen und Aufzüge, die es Rollstuhlfahrern ermöglichen, Höhenunterschiede selbstständig zu überwinden oder sanitäre Anlagen, die durch angebrachte Haltegriffe sowie vergrößerte Flächen vor und neben den Toiletten behindertengerecht gestaltet sind.

*Barrierefreiheit* wurde in den letzten Jahren jedoch auch bezüglich der Entwicklung von Computeranwendungen und Internetangeboten zu einem Schlagwort, an dem man als Entwickler immer schwieriger vorbei kommt.

## 1.2 Ziel der Arbeit

Das Ziel der vorliegenden Diplomarbeit besteht in der Konzeption und der Implementierung eines Werkzeugs, das es sehbehinderten und blinden Benutzern ermöglicht, Internetangebote zu erstellen. Zu beachten gilt, dass sowohl das Werkzeug als auch die damit erstellbaren Internetangebote *barrierefrei*, das heißt benutzbar für Menschen mit Behinderungen, aufzubauen sind. Darüber hinaus ist bei der Gestaltung der entstehenden Internetangebote darauf zu achten, dass durch die Unterstützung verschiedener Verhaltensmuster von Internetbenutzern eine hohe Gebrauchstauglichkeit entsteht. Um ein Werkzeug dieser Art speziell an die Bedürfnisse sehbehinderter und blinder Benutzer anzupassen, besteht die Aufgabe darin, zu untersuchen, inwieweit die Gestaltung der Internetangebote durch ein Werkzeug übernommen werden kann.



Welche Anforderungen entstehen, wenn Computeranwendungen und Internetangebote auch von behinderten Menschen zu benutzen sind, ist durch folgende Fragen zu klären:

- Was ist eine Behinderung und welche Arten von Behinderungen gibt es?
- Welche Behinderungen beeinflussen die Arbeit am Computer?
- Welche Hilfsmittel ermöglichen behinderten Menschen die Arbeit am Computer?
- Was ist zu beachten, damit Computeranwendungen und Internetangebote mit diesen Hilfsmitteln benutzbar sind?
- Welche rechtlichen Grundlagen und Standards sind zu berücksichtigen?

Die durch das Werkzeug erstellbaren Internetangebote sind nicht nur für behinderte Benutzer zugänglich zu gestalten. Gleichmaßen gilt es die Angebote auch für nichtbehinderte Benutzer zu optimieren. Um die Anforderungen nichtbehinderter Benutzer zu erläutern, kann nicht auf rechtliche Grundlagen zurückgegriffen werden. Vielmehr müssen Verhaltensmuster untersucht werden und abhängig davon Regeln aufgestellt werden. Folgende Fragen sind in diesem Zusammenhang zu beantworten:

- Welche Verhaltensmuster gibt es im Internet?
- Wie kann man Internetangebote diesen Verhaltensmustern anpassen?
- Ist dadurch eine kontextübergreifende Gebrauchstauglichkeit möglich?

Ist geklärt, welche Anforderungen die entstehenden Internetangebote erfüllen müssen, steht noch immer offen, wie das Werkzeug den Benutzer beim Erstellen der Internetangebote unterstützen soll. Die Möglichkeiten reichen von einfacher Syntaxhervorhebung bis zur automatisierten Erstellung vollständiger Seitenstrukturen. Abhängig von der Art des Werkzeugs stellt sich die Frage, in welcher Form die Inhalte sinnvoll zu speichern sind. Hier ergeben sich folgende Fragestellungen:

- Welche Unterstützung kann und soll das Werkzeug bieten?
- Welche Form der Datenhaltung kommt in Frage?

Nach Beantwortung der gestellten Fragen gilt es verschiedene Lösungswege zur Umsetzung eines solchen Werkzeugs zu ermitteln. Die Eignung der Um-

setzungsmöglichkeiten wird durch ein Vergleich mit den gestellten Anforderungen diskutiert, damit im Anschluss die am besten geeignete Lösung prototypisch umsetzbar ist.

Zu entwickeln ist ein Werkzeug, mit dem sehbehinderte und blinde Menschen ohne tiefere technische Kenntnisse in der Lage sind, vollständige Internetangebote zu erstellen. Das Werkzeug muss daher den Regeln der Barrierefreiheit entsprechen und alle Funktionalitäten bereitstellen, die beim Erstellen oder Bearbeiten von Internetangeboten entstehen. Um für möglichst viele Benutzer zugänglich zu sein, sind die entstehenden Internetangebote sowohl barrierefrei als auch gebrauchstauglich umzusetzen. Daher ist zusätzlich die optische Gestaltung der Angebote vollständig durch das Werkzeug auszuführen. Dies ermöglicht eine ausreichende Unterstützung sehbehinderter und blinder Benutzer.

## 1.3 Aufbau der Arbeit

**Kapitel 1 Einleitung:** Zu Beginn werden Ziel und Aufbau der vorliegenden Diplomarbeit erläutert.

**Kapitel 2 Behinderung:** Dieses Kapitel stellt die verschiedenen Arten von Behinderungen vor und erklärt, wie diese die Arbeit am Computer beeinträchtigen können.

**Kapitel 3 Sehbehinderung:** An dieser Stelle geht es detailliert um die Einschränkungen, die durch eine Sehbehinderung entstehen. Weiterhin werden die Möglichkeiten sehbehinderter Menschen im Umgang mit dem Computer geschildert.

**Kapitel 4 Barrierefreie Internetauftritte:** Um die Barrierefreiheit der entstehenden Anwendung und Internetangebote zu gewährleisten, beschreibt Kapitel 4 vorhandene rechtliche Grundlagen und Standards.

**Kapitel 5 Gebrauchstaugliche Internetauftritte:** Hier werden Gewohnheiten von Internetbenutzern dargestellt und dadurch Anforderungen an benutzerfreundliche Internetangebote herausgearbeitet.

**Kapitel 6 Werkzeuge zur Erstellung von Internetangeboten:** Dieses Kapitel stellt die verschiedenen Werkzeugalternativen vor, die das Erstellen von Internetangeboten unterstützen.

**Kapitel 7 Datenhaltung:** In diesem Kapitel wird erarbeitet, in welcher Form die Datenhaltung des Werkzeugs möglich ist.

**Kapitel 8 Konzeption des Werkzeugs:** Dieses Kapitel zeigt die an das Werkzeug gestellten Anforderungen auf und ermittelt mögliche Lösungswege.

**Kapitel 9 Umsetzung:** Die Erläuterung der prototypischen Implementierung des Werkzeugs steht im Mittelpunkt dieses Kapitels.

**Kapitel 10 Zusammenfassung und Ausblick:** Die Ergebnisse der Arbeit werden zusammengetragen und diskutiert. Anschließend erfolgt ein kurzer Ausblick auf mögliche Erweiterungen.

# Theorie

## 2 Behinderung

Das Ziel der vorliegenden Arbeit besteht in der Entwicklung eines Werkzeugs zur Erstellung von Internetangeboten. Sowohl das Werkzeug als auch die Internetangebote sind barrierefrei zu gestalten, um so die Benutzungsmöglichkeit für behinderte Personen sicherzustellen.

Um zu verstehen, welche Anforderungen sich durch Barrierefreiheit ergeben, gilt es vorerst zu untersuchen, was überhaupt eine Behinderung darstellt, welche Behinderungen sich auf die Arbeit am Computer auswirken und welche Hilfsmittel Menschen mit diesen Behinderungen die Arbeit am Computer ermöglichen. Kapitel 4 geht später darauf ein, wie Informationstechnik zu gestalten ist, um mit diesen Hilfsmitteln benutzbar zu sein.

Es gestaltet sich schwierig, eine einheitliche Definition für die Tatsache einer Behinderung festzulegen. Folgend sind zwei mögliche Definitionen aufgezeigt:

- Als behindert gelten Personen, die aufgrund von Beeinträchtigungen sowie deren Auswirkungen in einem existenzwichtigen sozialen Beziehungsfeld wie Bildung, Erwerbstätigkeit, Kommunikation, Wohnen oder Freizeit auf Dauer beeinträchtigt sind [vgl. KM83, S.60].
- Als behindert gelten entsprechend dem Behindertengleichstellungsgesetz (BGG) Menschen, die unter einer Beeinträchtigung körperlicher, geistiger oder seelischer Art leiden, sofern diese Beeinträchtigung länger als sechs Monate anhält und von dem für das Lebensalter üblichen Zustand beträchtlich abweicht. Die Teilnahme am Leben in der Gesellschaft ist bei Menschen mit Behinderungen infolgedessen beeinträchtigt [vgl. BuD, §3].

Durch Unterscheidung der vorhandenen Beeinträchtigung ergibt sich die Möglichkeit, Behinderungen nach ihrer Art in drei verschiedene Gruppen einzutei-

len: Körperbehinderungen, Intelligenzbehinderungen und Sinnesbehinderungen [vgl. KM83, S.60]. In den folgenden Unterkapiteln werden die einzelnen Behinderungen erklärt und mögliche Barrieren, die bei der Arbeit mit einem Computer entstehen, näher erläutert. Der Schwerpunkt der vorliegenden Arbeit liegt auf der Untersuchung der Anforderungen von sehbehinderten Benutzern an Computeranwendungen. Aus diesem Grund werden die verschiedenen Arten der Sehbehinderungen anschließend in Kapitel 3 gesondert erläutert.

## 2.1 Körperbehinderung

Als körperbehindert gelten Personen, deren Bewegungsfähigkeit auf Dauer eingeschränkt ist. Die Ursachen und Erscheinungsformen dieser Behinderungen sind sehr unterschiedlich. Alle Formen der Körperbehinderung haben gemeinsam, dass betroffene Personen oft nicht die Möglichkeit besitzen, sich alleine ohne Hilfsmittel und ohne Hilfe bewegen zu können [vgl. KM83, S.60].

Im Umgang mit Computern können körperbehinderte Menschen aufgrund vorhandener motorischer Einschränkungen möglicherweise nicht in der Lage sein, die üblichen Eingabegeräte wie Maus und Tastatur zu benutzen. Um Eingaben zu ermöglichen, stehen ihnen verschiedene Möglichkeiten zur Auswahl. Bei leichten motorischen Einschränkungen kann schon der Einsatz einer Großfeldtastatur in Kombination mit einer Tastaturmaus die üblichen Eingabegeräte vollständig ersetzen [vgl. Hel05, S.22]. Hier werden dem Benutzer durch größere Tasten sowie durch die freie Belegbarkeit von Tasten, was die Benutzung von Tastaturkürzeln ermöglicht, Eingaben erleichtert. Bei größeren Einschränkungen können Eingaben, abgesehen von der Möglichkeit der Spracheingabe, auch durch alternative Zeigegeräte in Verbindung mit einer Bildschirmtastatur vorgenommen werden. Ein alternatives Zeigegerät kann beispielsweise mit dem Mund bedienbar sein und alle Funktionen einer Standardmaus zur Verfügung stellen [vgl. Hel05, S.21].

Um Menschen mit motorischen Einschränkungen Informationstechnik voll zugänglich zu gestalten, muss sichergestellt sein, dass Software sowohl mit

alternativen Zeigegeräten als auch mit einer Tastatur bedienbar ist [vgl. Hel05, S.23].

## 2.2 Intelligenzbehinderung

Intelligenzbehinderungen stellen Schädigungen der geistigen Funktionen dar, wobei von geistiger Behinderung nur dann gesprochen werden kann, wenn die Schädigung der Intelligenz in besonders schwerer Form vorliegt [vgl. KM83, S.60].

Für Menschen mit Lernbehinderungen kann die Zugänglichkeit eines Internetangebotes durch die möglichst verständliche Darstellung des Seiteninhalts erreicht werden. Zu berücksichtigen sind hier neben übersichtlich strukturierten Internetseiten vor allem die in einem Internetangebot enthaltenen Texte, da diese zu einem großen Teil die Informationsübermittlung übernehmen. Verständliche Texte bedienen sich einer einfachen Sprache, in der nach Möglichkeit Fremdwörter und Abkürzungen vermieden werden, sowie einem klaren Satz- und Textaufbau, um den Leser nicht durch überflüssige Schachtelsätze und Querverweise zu verwirren.

## 2.3 Sinnesbehinderungen

Des Weiteren gibt es die Gruppe der Sinnesbehinderungen. Als sinnesbehindert gelten Menschen, bei denen eine Schädigung im Seh- oder im Hörvermögen oder eine Schädigung der Sprache vorliegt [vgl. KM83, S.60].

Da Computeranwendungen oft visuell ausgerichtet sind, treten für hörbehinderte Menschen nur in seltenen Fällen Beeinträchtigungen auf. Bei Internetangeboten werden Töne oft lediglich als Nebeneffekt zur Untermalung eingesetzt und sind nur selten zum Verständnis der Information nötig [vgl. Nie00, S.308]. Auf Ersatz der enthaltenen Audioinformation ist jedoch bei der Verwendung von Video und Animation zu achten. Um die enthaltenen Audioinformationen auch hörbehinderten Menschen zugänglich zu machen, müssen

Multimedia-Präsentationen mit äquivalenten Alternativen ausgestattet werden [Vgl. Bul]. Die Übersetzung in Gebärdensprache oder eine Untertitelung sind als optimale Alternativen zu einer Tonspur anzusehen.

Durch die bereits erwähnte visuelle Ausrichtung von Computeranwendungen entstehen besonders für sehbehinderte Menschen große Barrieren in der Benutzung. Daher geht das nächste Kapitel genauer auf die unterschiedlichen Arten der Sehbehinderungen ein. Anschließend wird erläutert, wie Sehbehinderte mit dem Computer arbeiten und welche Vorteile sich besonders für blinde Menschen bei der Arbeit am Computer ergeben.

### 3 Sehbehinderung

In der vorliegenden Arbeit nehmen Sehbehinderungen eine besondere Position ein. Zum einen erschweren grafische Benutzeroberflächen sehbehinderten Personen die Benutzung von Computern, zum anderen richtet sich das zu entwickelnde Werkzeug speziell an sehbehinderte Privatpersonen um diese bei der Erstellung von Internetangeboten zu unterstützen. Um die speziellen Anforderungen sehbehinderter Benutzer an das zu erstellende Werkzeug zu verstehen, geht dieses Kapitel detailliert auf die verschiedenen Ausprägungen von Sehbehinderungen ein. Darüber hinaus werden Hilfsmittel, die sehbehinderten Menschen Computerarbeit ermöglichen, vorgestellt.

Sehbehinderungen sind auf verschiedene Ursachen zurückzuführen und haben unterschiedliche Ausprägungen. Abhängig von der vorhandenen Sehstärke kann eine Unterteilung in drei Gruppen vorgenommen werden: Sehbehinderung, hochgradige Sehbehinderung und Blindheit. Zusätzlich sind auch Farbwahrnehmungsstörungen sowie Gesichtsfeldeinschränkungen zu berücksichtigen.

Unterschiedliche Arten von Sehbehinderungen erfordern auch unterschiedliche Anpassungen, um Einrichtungen entsprechend dieser Behinderung zugänglich zu gestalten. Für die Benutzung von Informationstechnik gibt es für Sehbehinderte und Blinde spezielle Hilfsmittel, die nonvisuelle Ein- und Ausgaben ermöglichen. Computeranwendungen müssen diese Möglichkeiten berücksichtigen und erlauben. Bevor erläutert wird, wie Informationstechnik für sehbehinderte und blinde Benutzer zugänglich gestaltet werden kann, zeigen die nächsten Absätze auf, welche Arten der Sehbehinderungen unterschieden werden.

### **3.1 Sehbehinderung und hochgradige Sehbehinderung**

Liegt eine erhebliche Beeinträchtigung der optischen Wahrnehmung vor, so bezeichnet man das als Sehschädigung. Von einer Sehbehinderung wird erst bei einer schwerwiegenden Einschränkung des Sehvermögens gesprochen. Gesetzlich wird zwischen einer Sehbehinderung und einer hochgradigen Sehbehinderung unterschieden. Eine hochgradige Sehbehinderung liegt vor, wenn die verbleibende Sinnesempfindung nur noch zwischen zwei und fünf Prozent beträgt. Von einer Sehbehinderung ist die Rede bei einer Sehkraft von fünf bis hin zu 33 Prozent [vgl. Bro99, S.1147].

Über die Anzahl sehbehinderter Menschen in Deutschland gibt es keine zuverlässigen Angaben. Der Deutsche Blinden und Sehbehinderten Verband (DBSV) schätzt die Gesamtzahl auf 500.000 Menschen [vgl. Deu].

### **3.2 Blindheit**

Von völliger Blindheit spricht man, wenn über das Auge keinerlei Sinneswahrnehmungen übermittelt werden und somit keine Lichtempfindung mehr möglich ist. Völlig blinde Menschen können nicht anhand der Lichtunterschiede zwischen Tag und Nacht unterscheiden. Gesetzlich ist ein Mensch blind, wenn eine Sehstärke unter zwei Prozent vorliegt oder sonstige schwerwiegende Seh-



störungen, wie beispielsweise eine starke Gesichtsfeldeinschränkung, gegeben sind [vgl. Bro99, S.1147].

In Deutschland leben etwa 155.000 Menschen, die von Blindheit betroffen sind [vgl. Deu]. Etwa 28 Prozent der Blinden im erwerbsfähigen Alter sind erwerbstätig, das entspricht etwa 10.000 Menschen [vgl. Deu].

### **3.3 Farbenblindheit und -fehlsichtigkeit**

Ist das Auge nicht in der Lage, die Farben des gesamten Farbspektrums wahrzunehmen, spricht man von einer totalen Farbenblindheit [vgl. Bro99, S.373]. Betroffene Menschen empfinden keinerlei Farben und können somit nur Grautöne unterscheiden. Häufig gehen mit einer solchen Störung starke Sehschwächen und Lichtscheu einher [vgl. Bro99, S.374]. Handelt es sich nur um eine Empfindungsstörung bei bestimmten Farben, nennt sich das Krankheitsbild Farbfehlsichtigkeit. Verbreitet sind hier besonders Störungen in der Wahrnehmung der Farben rot und grün. Weniger oft vertreten ist eine Wahrnehmungsstörung der Farben gelb und blau [vgl. Bro99, S.373].

Unter Farbfehlsichtigkeit leiden etwa acht Prozent der männlichen Bevölkerung und weniger als ein halbes Prozent der weiblichen Bevölkerung. Betroffen von einer vollständigen Farbenblindheit sind weniger als 0,01 Prozent der Bevölkerung [Hel05, S.11].

### **3.4 Sehbehinderte im Umgang mit dem Computer**

Abhängig von der vorliegenden Beeinträchtigung der Sehfähigkeit gibt es unterschiedliche Möglichkeiten, wie Benutzer mit einem Computer interagieren. Im Folgenden werden diese Möglichkeiten erläutert. Vielen sehenden Computerbenutzern ist die Vorstellung, dass hochgradig sehbehinderte und blinde Menschen mit einem Computer arbeiten, fremd. Dies ist nicht verwunderlich angesichts der Tatsache, dass der Bildschirm als übliches Hilfsmittel, das für die Ausgaben eines Computers benutzt wird, rein visueller

Natur ist. Blinde Anwender können diese Form der Ausgabe nicht benutzen. Ihnen stehen für die Interaktion mit dem Computer nonvisuelle Hilfsmittel zur Verfügung. Auch hochgradig Sehbehinderte arbeiten, in Anbetracht der wieder kleiner werdenden Bildschirme, nicht unter den für viele Menschen üblichen Bedingungen am Computer.

Bei der Arbeit mit einem Computer gilt es zwei Schnittstellen zwischen Mensch und Maschine zu betrachten: Die Ausgabe von Daten sowie deren Eingabe. Welche Hilfsmittel sehbehinderte und blinde Computerbenutzer zur Datenein- und -ausgabe einsetzen und wie sich die Arbeit mit ihnen gestaltet, wird in den folgenden Abschnitten aufgezeigt.

### 3.4.1 Ausgabemöglichkeiten

Gewöhnlich stehen Benutzern folgende Möglichkeiten der Datenausgabe zur Verfügung:

- Ein Bildschirm zur Darstellung von Grafik- und Textdaten.
- Eine Audioausgabe.

Die Ausgabe durch einen Drucker ist in dieser Übersicht nicht berücksichtigt, da die Ausgabe auf Papier nicht für ein Arbeiten mit dem Computer benutzt werden kann. Jedoch gibt es auch auf diesem Gebiet Geräte, die für Sehbehinderte eine vergrößerte Darstellung ermöglichen oder Blinden die Ausgabe in *Brailleschrift* erlauben [vgl. DKR69, S.13].

Für blinde und sehbehinderte Benutzer stellen durchschnittliche Bildschirme keine geeigneten Hilfsmittel dar. Abhängig von den Einschränkungen des Sehvermögens stehen ihnen andere Ausgabemedien zur Verfügung:

- Sprachausgabe zur Ausgabe von Textinformation.
- *Braillezeile*, um textuelle Daten auszugeben.
- Eine vergrößerte Ausgabe visueller und textueller Daten.

Die Ausgabemöglichkeiten für blinde Computerbenutzer - Sprachausgabe und *Braillezeile* - unterscheiden sich stark von den Möglichkeiten für sehbehinder-

te Benutzer. Durch diesen Unterschied ergibt sich die in der vorliegenden Arbeit vorhandene Trennung in sehbehinderte und blinde Benutzer. Die folgenden Abschnitte zeigen auf, wie sehbehinderte Benutzer die Darstellung auf dem Bildschirm benutzen können. Anschließend wird erläutert, wie blinde Benutzer Ausgaben wahrnehmen.

Ist ein Benutzer nicht in der Lage, die Bildschirmdarstellung bei korrekter physiologischer Sitzhaltung und dem daraus entstehenden Leseabstand wahrzunehmen, stehen ihm unterschiedliche Lösungsmöglichkeiten zur Verfügung. Bei einer geringen Sehbehinderung kann bereits das Einstellen einer größeren Schrift im Betriebssystem oder der Anwendung dem Benutzer Erleichterung verschaffen. Reicht eine Vergrößerung durch Systemeinstellungen nicht aus oder lässt eine Anwendung die Skalierung der Schrift nicht zu, hilft die Unterstützung durch ein Vergrößerungssystem. Diese Systeme vergrößern den gesamten Bildschirminhalt. Dadurch zeigt der Bildschirm zu jeder Zeit nur einen Ausschnitt des gesamten Inhalts an. Bei vierfacher Vergrößerung kann der Benutzer lediglich 1/16 des ursprünglichen Inhalts wahrnehmen [vgl. He105, 15]. Abbildung 3.1 zeigt die Bildschirmdarstellung in normaler Größe und dazu im Vergleich die Ansicht bei vierfacher Vergrößerung. Diese Darstellung schränkt die Übersichtlichkeit und damit die Orientierung des Benutzers stark ein.



Abbildung 3.1: Bildschirmanzeige in normaler Größe und bei vierfacher Vergrößerung

Farbfehlsichtige Menschen benutzen oft Systemeinstellungen, die von den Standardeinstellungen stark abweichen. Die von ihnen im Betriebssystem festgelegten Farben werden von Standardanwendungen übernommen [vgl. Hei05, S.12]. Benutzer können dadurch Hintergrund sowie Schriftfarben an ihre individuellen Bedürfnisse anpassen.

Eine *Braillezeile* stellt ausgewählten Bildschirminhalt für Blinde lesbar in *Brailleschrift* dar. *Brailleschrift* setzt sich aus Zeichen einzelner tastbarer Punkte zusammen. Eine nähere Erläuterung dieser Schrift ist in Anhang A zu finden. Die Zeichen werden durch Einheiten zu je acht *piezoelektronischen* Elementen dargestellt. *Piezoelektronische* Elemente bestehen aus Kristallen, die sich beim Anlegen einer elektrischen Spannung deformieren und dadurch einen Stößel als Punkt aus einer Fläche herausragen lassen [vgl. Rüb]. Jede Einheit kann dadurch genau ein Zeichen in *Computerbraille* ausgeben. *Computerbraille* besteht aus acht Punkten, während ein normales *Braillezeichen* lediglich aus sechs Punkten zusammengesetzt ist. Diese Erweiterung ist nötig, um den gesamten Zeichensatz, 256 Zeichen, des *American Standard Code for Information Interchange* (ASCII) abzubilden. *Braillezeilen* gibt es in Größen zu 20, 40 und 80 Einheiten [vgl. Hän]. Der ertastbare Bildschirmausschnitt beschränkt sich dadurch auf maximal 80 Zeichen. Um diesen Bildschirmausschnitt optimal zu nutzen, besitzen *Braillezeilen* zusätzlich zur Ausgabefunktion auch Möglichkeiten zur Eingabe. Steuertasten ermöglichen die Bewegung des dargestellten Ausschnittes durch den Benutzer [vgl. Hän]. Zusätzlich kann die *Braillezeile* darauf programmiert werden, bestimmte Bildschirmzustände, wie beispielsweise ein geöffnetes Menü, automatisch anzusteuern [vgl. Hän]. Moderne *Braillezeilen* stellen darüber hinaus sogenannte *Cursorroutingfunktionen* zur Verfügung. Jedem Zeichen der *Braillezeile* ist eine Taste zugeordnet, durch deren Betätigung der *Cursor* an der entsprechenden Stelle platziert werden kann. In Verbindung mit den Betriebssystemen der *Windowsfamilie* kann das *Cursorrouting* auch für eine Mausemulation verwendet werden. Die Aktivierung der Tasten kann dadurch auch einfache und doppelte Mausklicks auslösen [vgl. Hän]. Besonders bei Textkorrekturen kann die Arbeitsgeschwindigkeit durch *Cursorrouting* erhöht werden. Abbildung 3.2 zeigt eine

Anwenderin an einem mit einer *Braillezeile* ausgestattetem Bildschirmarbeitsplatz.



**Abbildung 3.2:** Benutzerin an einem Arbeitsplatz mit *Braillezeile*

Da *Braillezeilen* nicht in Massenproduktion hergestellt werden, sind die Preise dieser Produkte entsprechend hoch. Eine Zeile mit 20 Einheiten kostet bereits etwa 5.000 €. Bei Zeilen, die 80 Zeichen darstellen, „*wird die Grenze von 10.000 € überschritten*“ [Hän]. Über die finanzielle Belastung hinaus ist die Benutzung einer *Braillezeile* nur bei einer guten Beherrschung der *Brailleschrift* möglich. Späterblindete, deren *Braillelesefähigkeit* oft eingeschränkt ist, arbeiten daher häufig mit Sprachausgabesystemen [vgl. Hei97, S.9].

Sprachausgaben übermitteln Informationen durch die Ausgabe synthetischer Sprache. Auf diesem Gebiet stehen *Soft-* und *Hardwarelösungen* zur Verfügung. Je nach Hersteller erfolgt die Erzeugung der Sprache intern, durch die im Computer vorhandene *Soundkarte* in Verbindung mit Lautsprechern oder durch ein an die serielle Schnittstelle angeschlossenes externes Gerät [vgl. Hän].

Im Unterschied zu der zweidimensionalen Darstellung mittels eines Bildschirms sind durch *Braillezeile* und Sprachausgabe nur eindimensionale Ausgaben möglich [vgl. Hei05, S.9]. Daher ist, unabhängig davon, ob ein Medium oder die Kombination beider Medien genutzt wird, eine *Software* nötig, um die Informationen abzufangen und in linearer Form aufzubereiten. Diese Aufgabe

übernimmt ein sogenannter *Screenreader*. Das Programm bezieht Informationen von Anwendungen und dem Betriebssystem, interpretiert diese neu und übermittelt das Ergebnis an das Ausgabegerät [vgl. Hel05, S.8].

*Screenreader* arbeiten sowohl mit dem Betriebssystem als auch mit einzelnen Anwendungen zusammen. *Browser*, Anwendungen, die den Zugriff auf und die Darstellung von Internetangeboten ermöglichen [vgl. Bul], stellen daher nur einen kleinen Teil des Anwendungsgebiets von *Screenreadern* dar. Um eine optimale Zusammenarbeit von *Browser* und *Screenreader* zu erreichen, ist der Einsatz eines *Webreaders* hilfreich [vgl. Hel05, S.9]. Diese *Software* spezialisiert sich auf die Interpretation der Strukturen von Internetseiten und stellt praktische Funktionen, wie das Zusammenstellen einer Liste mit allen Verweisen einer Seite, bereit. Ein oft verwendeter *Webreader* ist der *Webformator* von *Audiodata* [vgl. Hel05, S.9]. Dieses kostenlose Zusatzprogramm zum *Internet Explorer* ermöglicht es auch sehenden Entwicklern von Internetseiten einen Eindruck davon zu bekommen, wie die Seiten von blinden Benutzern wahrgenommen werden. In diesem Zusammenhang findet das Programm im Praxisteil dieser Arbeit Verwendung.

### 3.4.2 Eingabemöglichkeiten

Nachdem im letzten Unterkapitel erläutert wurde, wie die Übermittlung von Informationen eines Computers an sehbehinderte und blinde Benutzer erfolgt, wird nun dargelegt, wie sehbehinderte und blinde Menschen Anwendungen durch Eingaben steuern.

Eingaben werden üblicherweise durch die Kombination zweier Medien vorgenommen:

- Zeigegerät, beispielsweise eine Maus, zur direkten Manipulation.
- Eingaben über eine Tastatur.

Für Sehbehinderte ergeben sich durch die Verwendung benutzerdefinierter Schriftgrößen oder eines Vergrößerungssystems keine Einschränkungen bei der Verwendung dieser Eingabegeräte. Da die Bedienung der Tastatur ohnehin oft

ohne eine visuelle Überprüfung stattfindet, stellt eine geminderte Sehfähigkeit hier keine Einschränkung dar. Ebenfalls werden Zeigegeräte aufgrund der präzisen Darstellung des Zeigers sowie der Fokusverfolgung durch Vergrößerungssysteme von Sehbehinderten in der üblichen Weise benutzt.

Abweichend ist die Verwendung eines Zeigegerätes für blinde Computerbenutzer nicht möglich. Um den Zeiger auf dem Bildschirm zu positionieren, ist eine visuelle Kontrolle erforderlich [vgl. Cla]. Da Blinden die benötigte Hand-Auge-Koordination fehlt, müssen alle Eingaben über die Tastatur erfolgen.

### 3.4.3 Vorteile der Computerbenutzung

Nach der Erklärung, auf welche Weise Sehbehinderte und Blinde den Computer benutzen, geht dieses Kapitel auf die Vorteile der Computerbenutzung ein.

Die fortschreitende Technisierung im Alltag bedeutet für sehbehinderte Menschen eine erhebliche Erleichterung. Durch die Verwendung von Computern und die Benutzung des Internets werden viele Barrieren im Alltag von sehbehinderten und blinden Menschen selbstständig überwunden. Viele Vorgänge erfolgen nun unabhängig von der Hilfe sehender Personen. Im Folgenden werden einige dieser Vorgänge näher betrachtet.

Das Senden und Empfangen von *Emails* ersetzt heute größtenteils den traditionellen Briefverkehr. Vorteile, die sich hierbei auch für Sehende ergeben, bestehen in einer schnelleren und kostengünstigere Zustellung sowie einer leichteren Mitteilungserstellung. Darüber hinaus entfallen für blinde und sehbehinderte Menschen die für sie umständliche Adressierung und korrekte Frankierung der Briefumschläge. Wichtiger ist jedoch die Tatsache, dass durch das elektronische Versenden von Post sehbehinderte Empfänger die Möglichkeit besitzen, Mitteilungen ohne fremde Hilfe zu lesen. Somit wird das Briefgeheimnis gewahrt. Elektronische Medien wie *Email* oder *Chat* ermöglichen Sehbehinderten und Blinden auch erstmals eine Kontaktaufnahme zu anderen Menschen, ohne dass physische oder psychische Barrieren bestehen.

Ein weiteres Geheimnis, dessen Wahrung für sehbehinderte und blinde Menschen erst eine entsprechende Abwicklung durch das Internet möglich gemacht hat, ist das Bankgeheimnis. Durch die elektronische Bearbeitung können nun auch sehbehinderte und blinde Menschen ihre Bankgeschäfte selbstständig ausführen.

Die fortschreitende Technisierung ist auch im Berufsleben zu spüren. Durch den verbreiteten Einsatz von Computern werden von blinden und sehbehinderten Menschen neue Berufsfelder erschlossen. Die Ausübung von Berufen, die eine Verarbeitung von Informationen, die lediglich in gedruckter Form vorliegen, erfordern, bleibt Blinden und Sehbehinderten verwehrt. Heute erfolgt die Verarbeitung von Textinformation häufig mit Hilfe von Computern. Dadurch stehen blinden und sehbehinderten Menschen bei der Berufswahl nicht mehr nur die als typisch angesehenen Blindenberufe wie Telefonist oder Masseur offen. Für sie sind heute auch viele computergestützte Tätigkeiten im Bürobereich, wie die Arbeit eines Sachbearbeiters, ausübbar. Durch diese mögliche höhere Qualifizierung ergeben sich im Berufsleben weitere interessante Aufgaben und die Akzeptanz im gesellschaftlichen Umfeld steigt.

Die Zugänglichkeit von Textinformationen erweitert nicht nur das Berufswahlfeld von blinden und sehbehinderten Menschen, auch im Privatleben ergeben sich hier weitere Vorteile. Aktuelle Informationen wie beispielsweise Reiseverbindungen sind durch die Bereitstellung im Internet für Blinde und Sehbehinderte benutzbar, während Informationen auf Papier diese Menschen oft ausschließen. In diesem Zusammenhang sind auch weitere Aspekte zu erwähnen. Durch die Möglichkeit Einkäufe, Reservierungen und Buchungen über das Internet zu erledigen, sind Sehbehinderte und Blinde erstmals in der Lage, sich selbstständig einen Überblick über das Angebot zu verschaffen und auf Sonderangebote zu reagieren. Dies besitzt besondere Wichtigkeit, da Sehbehinderte und Blinde durch den verbreiteten Einsatz von Automaten im Dienstleistungsbereich oft ausgeschlossen sind.

Informationen, die aufgrund ihres Umfangs nicht in der für Blinde lesbaren *Brailleschrift* erfasst werden, sind durch eine elektronische Aufbereitung für



Sehbehinderte und Blinde lesbar. *Brailleschrift* hat einen erheblich höheren Platzbedarf als die für Sehende verwendete Schwarzschrift. Aufgrund dieses Platzbedarfs und den sehr viel höheren Produktionskosten ist eine flächendeckende Information durch Bücher oder Tageszeitungen nicht möglich. Elektronisch erfasste Bücher und Tageszeitungen im Internet treten so an die Stelle herkömmlicher Informationsmedien.

## 4 Barrierefreie Internetauftritte

Im letzten Kapitel wurde herausgestellt, wie behinderte Menschen mit dem Computer arbeiten. Behandelt wurden unter anderem verschiedene Hilfsmittel, die ein Interagieren mit dem Computer für viele Menschen erst ermöglichen. Computeranwendungen und Internetangebote sind nicht prinzipiell durch die verschiedenen Hilfsmittel benutzbar. Vielmehr müssen bei der Entwicklung von Werkzeugen oder Internetangeboten bestimmte Regeln beachtet werden, um eine Benutzung durch die erwähnten assistive Technologien zu ermöglichen. Im Folgenden wird erläutert, was Barrierefreiheit bedeutet und wie sie umzusetzen ist.

Wie ist es möglich, ein Internetangebot barrierefrei zu gestalten? Um dies zu verstehen, ist vorab zu klären, was *barrierefrei* bedeutet. Der Begriff basiert auf der Übersetzung des englischen Schlagwortes *accessibility*, Zugänglichkeit. Ein Auftritt gilt dann als zugänglich, wenn die technische Umsetzung eine Benutzung auch für Menschen mit Behinderungen ermöglicht [vgl. Hel05, S.6]. Barrierefreiheit geht jedoch darüber hinaus. Sie schließt Aspekte der Benutzerfreundlichkeit mit ein und fordert eine Benutzbarkeit der Auftritte ohne besondere Erschwernis und in der allgemein üblichen Weise [vgl. BuD, Artikel1 §4].

## 4.1 Rechtliche Grundlagen und Standards

In den letzten Jahren wurde eine immer größere Anzahl vorhandener Internetangebote barrierefrei oder zumindest barrierearm gestaltet. Diese Entwicklung entstand nicht aus einer sozialen Verantwortlichkeit heraus. Eher ist ein gesetzlicher und öffentlicher Druck als Ursache dafür zu sehen, dass immer mehr Techniken entwickelt werden, die behinderten Menschen den Umgang mit Informationstechnik erleichtern oder sogar erst ermöglichen.

In den USA hat die Behindertengesetzgebung bereits eine sehr viel längere Tradition als in Deutschland. Dort wurde schon im Jahr 1998 die *Section 508* als Teil des *Rehabilitation Acts* erlassen. Durch diese Gesetzesergänzung wurde erneut bekräftigt, dass alle Informations- und Kommunikationstechniken der amerikanischen Regierung spätestens ab Juni 2001 auch für alle behinderten Menschen nutzbar sein müssen. In Folge dessen stellten auch viele gewerbsmäßige Anbieter in den USA ihre Angebote um, da nur so auch auf lange Sicht eine Zusammenarbeit mit einer der Regierung unterstellten Organisation möglich ist.

### 4.1.1 Behindertengleichstellungsgesetz

Die gesetzliche Grundlage in Deutschland bildet das Gesetz zur Gleichstellung behinderter Menschen, das Behindertengleichstellungsgesetz (BGG). Dieses Gesetz trat am 01.05.2002 in Kraft [vgl. BuD, Artikel 56] und baut auf der Ergänzung der Verfassung am 15.11.1994 in Artikel 3, Absatz 3, einem Benachteiligungsverbot für Behinderte, auf. Das BGG ist diesbezüglich das erste Gleichstellungsgebot in der Bundesrepublik Deutschland. Behinderten Menschen soll durch das BGG ein selbstbestimmtes Leben ermöglicht werden und sie zu einem gleichberechtigten Teil der Gesellschaft machen. Das Ziel des Gesetzes besteht darin, Benachteiligungen behinderter Menschen zu beseitigen [vgl. BuD, Artikel 1 §1]. Durch das Gesetz sind Träger der öffentlichen Gewalt in der Bundesrepublik Deutschland erstmals auf Bundesebene dazu verpflichtet, Benachteiligungen zu vermeiden.

Dies wiederum bedeutet eine barrierefreie Gestaltung. Abgesehen von baulichen und sonstigen Anlagen, Verkehrsmitteln, technischen Gebrauchsgegenständen, akustischen und visuellen Informationsquellen, Kommunikationseinrichtungen und anderen gestalteten Lebensbereichen ist das BGG auch auf Systeme der Informationsverarbeitung anwendbar [vgl. BuD, Artikel1 §4]. Im Rahmen der barrierefreien Informationstechnik sind Internetangebote sowie öffentlich zugängliche grafische Programmoberflächen, die durch Träger der öffentlichen Gewalt zur Verfügung gestellt werden, schrittweise technisch so zu gestalten, dass sie von behinderten Menschen grundsätzlich uneingeschränkt benutzbar sind [vgl. BuD, Artikel1 §11].

Neben einer Verpflichtung der Träger der öffentlichen Gewalt des Bundes sieht das Gesetz auch ein positives Einwirken der Bundesregierung hinsichtlich der barrierefreien Gestaltung von Internetseiten und grafischen Programmoberflächen, die von gewerbsmäßigen Anbietern zur Verfügung gestellt werden, vor [vgl. BuD, Artikel1 §11]. Hier sind Zielvereinbarungen auf freiwilliger Basis zwischen gewerblichen Anbietern und anerkannten Behindertenverbänden angedacht [vgl. Beh].

#### **4.1.2 Barrierefreie Informationstechnik-Verordnung**

Durch das BGG ist festgelegt, welche Angebote barrierefrei zu gestalten sind. Eine Aussage darüber, wie diese Barrierefreiheit erreicht werden kann, wird im Gesetzestext jedoch nicht getroffen. Um die im BGG festgelegten Inhalte verbindlich zu gestalten, erließ das Bundesministerium des Innern im Einvernehmen mit dem Bundesministerium für Arbeit und Sozialordnung eine zusätzliche Rechtsverordnung. Die Verordnung zur Schaffung barrierefreier Informationstechnik nach dem Behindertengleichstellungsgesetz, die Barrierefreie Informationstechnik-Verordnung (BITV), trat am 24.07.2002 in Kraft [vgl. Hel05, S.40]. Diese Rechtsverordnung regelt im Detail, welche Voraussetzungen Inter- und Intranetauftritte, die von Behörden der Bundesverwaltung angeboten werden, erfüllen müssen, um die geforderte Barrierefreiheit umzusetzen. Auf öffentlich zugängliche grafische Programmoberflächen wird jedoch in den Anforderungen, die die Verordnung stellt, nicht eingegangen. Durch die

föderale Struktur der Bundesrepublik benötigt jede Bestimmung auf Bundesebene ihre Entsprechung auf Landesebene. Die Einarbeitung dieses Bundesrahmengesetzes in entsprechende Ländergesetze und damit auch die technischen Umsetzungen stehen jedoch noch am Anfang ihrer Entwicklung.

Für die Umsetzung der BITV gelten verschiedene Fristen [vgl. Bul, §4]. Seit Inkrafttreten der Verordnung müssen Angebote, die neu gestaltet oder in wesentlichen Bestandteilen oder größerem Umfang verändert werden, barrierefrei sein. Bereits veröffentlichte Angebote, die sich speziell an behinderte Menschen richten, sind seit dem 31.12.2003 barrierefrei zu gestalten. Richten sich Angebote nicht speziell an behinderte Menschen, gilt eine Frist bis zum 31.12.2005.

Die BITV regelt neben den bereits erläuterten Fristen, Geltungsbereichen und dem Inkrafttreten auch die einzubeziehenden Gruppen, anzuwendende Standards und die spätere Überarbeitung der Verordnung. Technische Einzelheiten beschreibt Anlage (Teil 1) der Verordnung. Diese Einzelheiten sind in 14 Anforderungen gegliedert. Jeder Anforderung sind Details in Form verschiedener Bedingungen zugeordnet.

Da die BITV die von der Europäischen Union geforderte [vgl. Hel05, S.29] nationale Umsetzung der *Web Content Accessibility Guidelines 1.0* (WCAG 1.0) des *World Wide Web Consortium* (W3C) darstellt, sind die Anforderungen und Bedingungen der BITV Anlage (Teil1) eine genaue Übersetzung der Richtlinien und Checkpunkte der WCAG 1.0.

### 4.1.3 web Content Accessibility Guidelines (WCAG)

Die *Web Content Accessibility Guidelines* (WCAG) stellen Richtlinien dar, welche die Möglichkeiten erläutern, Internetinhalte für behinderte Menschen zugänglich zu gestalten [vgl. WCAG1]. Entwickelt wurden die WCAG vom *World Wide Web Consortium* (W3C), der höchsten Instanz für Entwicklungen im Internet [vgl. Hel05, S.33]. Das W3C sieht seine Aufgabe darin, das Internet „zu seiner

*vollen Entfaltung zu führen*" [W3C]. Um dieses Ziel zu erreichen, stehen dem W3C das Wissen, die Erfahrung und der Einfluss seiner mittlerweile mehr als 350 Mitgliedsorganisationen [vgl. W3C] zur Verfügung. Mitglieder sind neben bedeutenden Softwarefirmen wie *Apple, IBM, Microsoft* oder *SAP* auch namhafte Vertreter anderer Bereiche, wie *T-Online* als Vertreter der Telekommunikationsbranche [vgl. W3C]. Seit seiner Gründung im Oktober 1994 am *Massachusetts Institute of Technology, Laboratory for Computer Science (MIT/LCS)* hat das W3C bereits mehr als 50 Spezifikationen und 40 Empfehlungen herausgegeben [vgl. W3C]. Spezifikationen und Empfehlungen werden in selbstständigen, themenbezogenen Arbeitsgruppen entwickelt. Die WCAG entstanden durch die Arbeit der *Web Accessibility Initiative (WAI)*, einer Arbeitsgruppe, deren Aufgabenbereich in der Berücksichtigung von Aspekten der Zugänglichkeit in den Veröffentlichungen des W3C besteht.

Die WCAG stehen zur Zeit in zwei Versionen zur Verfügung. Die erste Version WCAG 1.0 stammt bereits aus dem Jahr 1999, entspricht aber den zur Zeit vorherrschend verwendeten Internettechniken wie der *HyperText Markup Language (HTML)* in der Version 4.01, der *eXtensible HyperText Markup Language (XHTML)* in der Version 1.0 und den *Cascading Style Sheets (CSS)* in der Version 1.

Während sich die WCAG 1.0 auf den Standard (X)HTML beschränken und sich an Entwickler von Internetinhalten sowie an Entwickler von Werkzeugen zur Erstellung von Internetinhalten richten, wird der Wirkungsgrad in den WCAG 2.0 erweitert. Die WCAG 2.0 werden seit 1999 entwickelt [vgl. Hel05, S.32] und bestehen aus technikübergreifenden Richtlinien, die nicht mehr nur auf (X)HTML beschränkt sind. Sie richten sich an alle am Entstehungsprozess von Internetinhalten beteiligten Personen, von den Entscheidungsträgern bis zu den Entwicklern. Um diese zu erreichen, sind die Richtlinien der WCAG 2.0 verständlicher und weniger technisch verfasst und in vier Gruppen aufgeteilt: Wahrnehmbarkeit des Inhalts, Bedienbarkeit der Oberfläche, Verständlichkeit des Inhalts und der Bedienung sowie Robustheit der Technik [vgl. WCAG2].

Die WCAG 2.0 liegen bisher nur in einer Arbeitsversion vor. Das bedeutet, dass die Richtlinien noch nicht dem Konsens des gesamten W3Cs entsprechen und noch nicht als W3C-Empfehlung gebilligt wurden. Folglich bezieht sich diese Arbeit auf die Version 1.0 der WCAG, da diese zur Zeit die gültige Referenz für Zugänglichkeitsgrundsätze darstellt.

#### 4.1.4 BITV - WCAG 1.0

BITV und WCAG 1.0 entsprechen einander. Beide Regelwerke führen 66 Punkte aus. In der BITV werden diese Punkte Bedingungen genannt, während in den WCAG 1.0 von Checkpunkten die Rede ist. Jeder dieser Punkte ist sowohl einer Priorität als auch einer Überschrift zugeordnet. Diese Überschriften werden in der BITV als Anforderung bezeichnet. In den WCAG 1.0 werden sie Richtlinien genannt. Der Unterschied zwischen BITV und WCAG 1.0 liegt in der Festlegung der verschiedenen Stufen der Zugänglichkeit.

Die WCAG 1.0 legen drei Stufen der Zugänglichkeit fest: A, AA und AAA. Um eine Konformität nach Stufe A zu erhalten, muss ein Internetangebot alle Checkpunkte erfüllen, die der Priorität 1 entsprechen. Erfüllt ein Angebot diese Checkpunkte nicht, *„wird es für eine oder mehrere Gruppen unmöglich sein, auf die Information im Dokument zuzugreifen“* [WCAG1]. Werden Checkpunkte außer Acht gelassen, die der Priorität 1 oder 2 entsprechen, erreicht das Angebot keine Konformität nach Stufe AA. Dadurch *„wird es für eine oder mehrere Gruppen schwierig sein, auf die Information im Dokument zuzugreifen“* [WCAG1]. Erfüllt ein Angebot alle Checkpunkte einschließlich der Checkpunkte, die der Priorität 3 zugeordnet sind, erreicht das Angebot eine Konformität der Stufe AAA. Dadurch wird ausgeschlossen, dass *„es für eine oder mehrere Gruppen etwas schwierig sein“* wird, *„auf die Information im Dokument zuzugreifen“* [WCAG1].

Bevor die BITV in Kraft trat, wurden die ihr zugrunde liegenden WCAG 1.0 auf ihre praktische Relevanz hin überprüft. Angaben von Behindertenverbänden sowie praktische Tests ergaben, dass Barrierefreiheit erst erreicht werden

kann, wenn die Konformitätsstufe AA der WCAG 1.0 umgesetzt ist [vgl. Hel05, S.43]. Um diesem Ergebnis Rechnung zu tragen, wurde von einer Übernahme der Konformitätsstufen der WCAG 1.0 in die BITV abgesehen. In der BITV gibt es nur zwei Prioritätsstufen: Priorität I und Priorität II. Priorität I der BITV fasst die Prioritäten 1 und 2 der WCAG 1.0 zusammen, während Priorität II der Priorität 3 entspricht. Um die Trennung der Prioritäten vornehmen zu können, waren formale Anpassungen weniger Checkpunkte der WCAG 1.0 nötig.

Die Umsetzung der Barrierefreiheit betreffend bezieht sich die vorliegende Arbeit überwiegend auf die BITV. Da diese auf den WCAG 1.0 aufbaut, werden die Richtlinien des W3C für nähere Erläuterungen herangezogen.

#### 4.1.5 Anforderungen der BITV

Um genau zu wissen, welchen Regeln das zu entwickelnde Werkzeug sowie die damit erstellbaren Internetangebote folgen müssen, um für behinderte Menschen benutzbar zu sein, werden anschließend die durch die BITV gestellten Anforderungen erläutert. Die Trennung der Bedingungen nach Priorität I und II wird in der Auflistung ausgelassen, da Barrierefreiheit in der vorliegenden Arbeit durch Priorität II der BITV (entspricht Priorität 3 der WCAG 1.0) umzusetzen ist. Daraus folgend sind alle Bedingungen zu bearbeiten.

##### 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.“ [Bul]*

Um die BITV umzusetzen, muss jedes Nicht-Text-Element durch ein Text-Äquivalent ergänzt werden. Nicht-Text-Elemente können in zwei Gruppen eingeteilt werden: Audio- und visuelle Elemente.

Für Menschen, die bei der Computerbenutzung nichtgrafische Ausgabemöglichkeiten wie die Ausgabe durch Sprache oder eine *Braillezeile* benutzen, blei-

ben Inhalte grafischer Elemente unzugänglich. *Screenreader* sind nicht in der Lage, den Inhalt grafischer Elemente zu interpretieren. Auch programmierte Objekte bleiben ihnen häufig unzugänglich, da deren Benutzung meist zusätzliche *Software* benötigt, deren Unterstützung vielen *Screenreadern* noch fehlt [vgl. Hel05, S.232 und S.284]. Um sicherzustellen, dass diese Inhalte dennoch erfassbar sind, benötigen sie zusätzlich eine nichtgrafische Form der Darstellung. Für Menschen, die Audioinformationen nicht direkt nutzen können, müssen diese durch Text-Äquivalente ergänzt werden. Text-Äquivalente sind Texte, die nicht nur den Inhalt der zu ersetzenden Elemente wiedergeben, sondern darüber hinaus deren Funktion beschreiben. Äquivalente visueller Schaltflächen geben beispielsweise nicht unbedingt Auskunft über den Inhalt der verwendeten Grafik, sondern über die Auswirkungen einer Betätigung der Schaltfläche.

Mit Text-Äquivalenten zu ergänzen sind Grafiken, auch wenn diese Text grafisch darstellen, als Listenpunkte, Platzhalter oder zur Erstellung von Schaltflächen verwendet werden. Weiterhin sind Zeichnungen, die durch die Anordnung von ASCII-Zeichen entstehen sowie die Bildschirmunterteilung durch mehrere *Frames* (Bedingung 1.1) zu nennen. Um aktive Regionen von *Imagemaps*, verweissensitiven Grafiken, zu ersetzen, ist die Bereitstellung redundanter Verweise nötig (Bedingung 1.2 und 1.5). Zu ergänzende Audioelemente sind Töne, die entweder aus Audio-Dateien oder auch aus der Tonspur eines Videos stammen (Bedingung 1.1). Bei dem Einsatz von zeitgesteuerten Multimedia-Präsentationen wie Video oder Animation ist die äquivalente Bereitstellung von Audio- und Videoinformation zusätzlich mit der Präsentation zu synchronisieren (Bedingung 1.4), wobei Inhalte der Videospur zusätzlich als Audiobeschreibung anzubieten sind (Bedingung 1.3).



## Anforderung 2

*„Texte und Grafiken müssen auch dann verständlich sein, wenn sie ohne Farbe betrachtet werden.“ [Bul]*

Für Menschen, die Probleme haben, verschiedene Farben zu erkennen oder zu unterscheiden, können in zwei Fällen Schwierigkeiten beim Verständnis von Internetinhalten auftreten.

Werden Farben verwendet, um Informationen zu übermitteln, können diese Informationen für farbfeldsichtige oder farbenblinde Menschen verloren gehen. Daher dürfen Farben in Internetauftritten lediglich als Ergänzung und in keinem Fall als alleinige Informationsträger verwendet werden (Bedingung 2.1).

Darüber hinaus kann es sich für Menschen mit Farbfeldsichtigkeit oder Farbenblindheit als schwierig oder unmöglich gestalten, Unterschiede zwischen Vorder- und Hintergrundfarbe wahrzunehmen, wenn diese sich lediglich im Farbton und nicht im Kontrast unterscheiden (Bedingung 2.2 und 2.3). Da es ein breites Spektrum an Farbfeldsichtigkeiten gibt, sollten Farbkombinationen, die als kritisch bekannt sind, vermieden werden. Kritische Farbpaare sind abgesehen von den Kombinationen komplementärer Farben auch Rot/Schwarz, Gelb/Weiß, Rot/Blau sowie Gelb/Rot oder Gelb/Grün [vgl. Hel05, S.82].

Zusätzlich ist es notwendig, dass der Anwender die Darstellung individuell an seine Bedürfnisse anpassen kann [vgl. Hel05, S.74]. Ermöglicht wird dies durch die Trennung von Inhalt und Präsentation durch die Verwendung von *Stylesheets*. Grafische *Browser*, Programme, die den Zugriff auf Internetangebote ermöglichen [vgl. Bul], können *Stylesheets* deaktivieren und erlauben dadurch eine benutzerdefinierte Darstellung des Inhalts.

### Anforderung 3

*„Markup-Sprachen (insbesondere HTML) und Stylesheets sind entsprechend ihrer Spezifikationen und formalen Definitionen zu verwenden.“ [Bul]*

*Screenreader* benutzen die Angaben der Auszeichnungssprache, um den Aufbau einer Internetseite strukturiert wiederzugeben und es dem Benutzer so zu erleichtern, diese Struktur zu verstehen [vgl. WCAG1]. Werden Elemente der Auszeichnungssprache nicht korrekt entsprechend ihrer Spezifikation verwendet, schränkt das die Verständlichkeit und die Navigierbarkeit der Internetseiten für Benutzer spezialisierter *Software* ein.

Strukturauszeichnung ist ausschließlich für die Seitenstruktur zu verwenden, während die Verwendung von Präsentationsauszeichnung auf Präsentationszwecke beschränkt ist (Bedingungen 3.5, 3.6 und 3.7). Ein häufiger Verstoß gegen diese Anforderung ist die Verwendung des `blockquote`-Elementes, welches Zitate kennzeichnet, zur Formatierung eines Absatzes mit Texteinrückung. Um die Präsentation zu beeinflussen, sind statt dessen von der Auszeichnungssprache getrennte *Stylesheets* zu verwenden (Bedingung 3.3).

Internetinhalte sollen für ein breites Publikum benutzbar sein. Daher ist es notwendig, dem Benutzer die Möglichkeit zu geben, die Größe des Seiteninhalts an seine Bedürfnisse anzupassen (Bedingung 3.4). Wird die Präsentation einer (X)HTML-Seite durch *Stylesheets* umgesetzt, besteht die Möglichkeit, Elementen eine relative statt einer absoluten Größe zuzuordnen. Relative Größenangaben bei Schriften und anderen Elementen ermöglichen dem Benutzer die Anpassung der Seitendarstellung an seine Sehfähigkeit. Bei der Verwendung von absoluten Angaben ist eine Skalierung der Inhalte oft nicht möglich [vgl. Hel05, S.140]. Auch die Verwendung von Schriftgrafiken schränkt die Skalierbarkeit des enthaltenen Textes ein. Daher ist die Verwendung einer Auszeichnungssprache in Verbindung mit *Stylesheets* zur Textformatierung der Verwendung von Schriftgrafiken vorzuziehen, sofern eine entsprechende Auszeichnungssprache existiert (Bedingung 3.1). Die Abbildung eines Textes

durch eine Grafikdatei vermischt Inhalt mit Präsentation und verhindert darüberhinaus in vielen grafischen Benutzeragenten die Skalierbarkeit der dargestellten Schrift [vgl. Hel05, S.86].

Mittels Auszeichnungssprachen erstellte Dokumente müssen den Regeln, die für diese Auszeichnungssprache gelten, entsprechen (Bedingung 3.2). Daher verweisen (X)HTML-Dokumente zu Beginn des Dokumentes auf die für sie gültigen Regeln, die *Document Type Definition* (DTD).

#### Anforderung 4

*„Sprachliche Besonderheiten wie Wechsel der Sprache oder Abkürzungen sind erkennbar zu machen.“ [Bul]*

Bei der Benutzung einer Sprachausgabe wird der Bildschirminhalt in synthetischer Sprache ausgegeben. Die Aussprache und Betonung dieser synthetischen Sprache ist dabei fest an eine Landessprache gebunden. Diese lässt sich durch die Konfiguration der Software festlegen. Erfolgt die Ausgabe einer Landessprache mit einer dafür ungeeigneten Konfiguration, sind Worte durch die falsche Aussprache schwer zu verstehen. Um diese Schwierigkeiten zu umgehen, sieht die BITV eine Kennzeichnung der vorherrschend verwendeten natürlichen Sprache sowie deren Änderung vor (Bedingungen 4.1 und 4.3). Dies ermöglicht ein automatisches Wechseln der Sprache durch die Ausgabe-Software.

Verständnisschwierigkeiten entstehen auch bei der Ausgabe von Abkürzungen oder Akronymen, da die *Software* zur Sprachausgabe versucht, Abkürzungen als eigenständige Worte auszugeben. Um eine korrekte Ausgabe zu gewährleisten müssen Abkürzungen und Akronyme durch *Markup* gekennzeichnet und bei ihrem ersten Auftreten inhaltlich erläutert werden (Bedingung 4.2).

### 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.“ [Bul]*

(X) HTML-Tabellenkonstrukte sind zweidimensionale Objekte. Ausgabemedien wie der Sprachausgabe oder der Ausgabe über eine *Braillezeile* steht jedoch nur eine Dimension zur Verfügung. Um eine korrekte Ausgabe zu ermöglichen, werden Tabellen vor der Ausgabe durch die Zugangssoftware linearisiert. In der Regel geschieht dies, indem die Tabellenzellen Zeile für Zeile von links nach rechts abgearbeitet werden [vgl. Hel05, S.154]. Wird eine Tabelle zur Darstellung tabellarischer Daten verwendet und werden Zeilen und Spalten durch Markup mit aussagekräftigen Überschriften versehen, gestaltet sich das Verstehen des Tabelleninhalts auch für Benutzer eindimensionaler Ausgabemedien problemlos. Verschachtelte Tabellenkonstrukte, die zur Positionierung der enthaltenen Elemente auf dem Bildschirm dienen, *Layouttabellen*, führen in linearisierter Form selten zu brauchbaren Ausgaben [vgl. Hel05, S.154].

Werden Tabellen zur Darstellung tabellarische Daten verwendet, sind die Zeilen und Spalten mit aussagekräftigen Überschriften zu versehen. Werden diese mittels Auszeichnungselementen als solche gekennzeichnet (Bedingung 5.1) und zusätzlich in verkürzter Schreibweise angeboten (Bedingung 5.6), ermöglicht dies eine entsprechende Interpretation durch *Screenreader* sowie eine visuelle Hervorhebung, auch wenn *Stylesheets* vom Benutzer deaktiviert wurden [vgl. Hel05, S.196]. Um Sprachausgabebenutzern besonders bei Tabellen mit mehreren logischen Ebenen (Bedingung 5.2) die Zuordnung einer Zelle zu den entsprechenden Überschriften zu ermöglichen, müssen Zuweisungen durch Auszeichnungselemente vorgenommen werden. Die zusätzliche Zusammenfassung einer Tabelle (Bedingung 5.5) ist für Benutzer der Sprachausgabe hilfreich, da diese ihnen einen schnellen Gesamtüberblick ermöglicht.

Tabellen sind nicht zur visuellen Formatierung des Seiteninhalts zu benutzen. Dies ist nur erlaubt, soweit eine Linearisierung der *Layouttabelle* möglich ist

und zu einer sinnvollen Ausgabe führt (Bedingung 5.3). Da in diesem Fall die Strukturierung der Tabelle durch Markup dem Benutzer überflüssige Information bieten würde, ist von zusätzlichem Markup abzusehen (Bedingung 5.4).

### Anforderung 6

*„Internetangebote müssen auch dann nutzbar sein, wenn der verwendete Benutzeragent neuere Technologien nicht unterstützt oder diese deaktiviert sind.“ [Bul]*

Unabhängig davon, wie groß die Verbreitung einer Technologie ist, gibt es Benutzer, die diese nicht einsetzen können. Daher muss ein Internetauftritt auch ohne neuere Technologien benutzbar sein. Zu berücksichtigen sind neben dynamischen Inhalten durch *Scripts*, *Applets* oder anderen programmierten Objekten auch *Stylesheets* und *Frames*.

Hinsichtlich der Seitengestaltung durch *Stylesheets* ist zu beachten, dass der Inhalt von (X)HTML-Dokumenten auch bei deaktivierten *Stylesheetdateien* benutzbar bleibt (Bedingung 6.1). Werden (X)HTML-Dokumente, deren Gestaltung vollständig durch CSS-Dateien umgesetzt ist, ohne die Unterstützung von *Stylesheets* angezeigt, geht die Positionierung der Elemente verloren. Um dennoch eine zweckmäßige Ausgabe zu erhalten, muss im Quelltext eine schlüssige Reihenfolge der Elemente vorliegen.

Bei der Verwendung von dynamischen Inhalten gilt es sicherzustellen, dass die Eingaben aller programmierten Objekte geräteunabhängig möglich sind (Bedingung 6.4) und die Internetauftritte auch bei der Deaktivierung von programmierten Objekten benutzbar bleiben (Bedingung 6.3). Zu berücksichtigen sind hier beispielsweise Seiten, deren Inhalte vollständig durch *JavaScript* generiert werden oder deren Navigation nur durch *JavaScript* oder *Flash* benutzbar ist. Da *„die Schnittstelle zwischen Flash und den Computerhilfsmitteln nicht ausgereift und die Bedienung in der Regel nur unter erschwerten Bedingungen möglich“* [Hel05, S.104] ist und *JavaScript clientseitig* dynamische Seiteninhalte erstellt, *Screenreader* Seiten jedoch statisch erfassen [vgl. Hel,

S.179], müssen bei der Verwendung dieser oder anderer programmierter Objekte Äquivalente des dynamischen Inhalts angeboten werden (Bedingung 6.5). Hierbei muss sichergestellt sein, dass alternativ angebotene Inhalte aktualisiert werden, wenn eine Änderung des dynamischen Inhalts erfolgt (Bedingung 6.1).

### Anforderung 7

*„Zeitgesteuerte Änderungen des Inhalts müssen durch die Nutzerin/den Nutzer kontrollierbar sein.“ [Bul]*

Diese Anforderung bezieht sich auf zwei Problemfelder. Geregelt wird zum einen der Umgang mit blinkenden, flackernden und beweglichen Seiteninhalten, zum anderen die automatische Änderung des Seiteninhaltes.

Bei Menschen mit photosensitiver Epilepsie kann ein Flackern oder Blinken der Bildschirminhalte im Bereich von vier bis 59 Hertz Anfälle auslösen [WCAG1]. Um dies zu verhindern sind sowohl das Flackern des gesamten Bildschirms (Bedingung 7.1) als auch blinkende Inhalte (Bedingung 7.2) zu vermeiden. Darüber hinaus kann die Bewegung von Inhalten, beispielsweise bei der Verwendung einer Laufschrift, sehbehinderte oder leseschwache Benutzer ausschließen, da Inhalte nicht schnell genug erfasst werden können. Bewegungen der Seiteninhalte sind daher zu vermeiden oder deaktivierbar zu gestalten (Bedingung 7.3).

Die automatische Änderung des Seiteninhalts kann auf unterschiedliche Weise durchgeführt werden. Durch den Einsatz von *JavaScript* kann dynamisch ein Teil einer Internetseite ausgetauscht werden, ohne dass die gesamte Seite neu geladen wird. Da *Screenreader* lediglich „*ein statisches Bild einer Seite erfassen*“ [Hel05, S. 179], gehen ihnen diese Änderungen verloren. Darüber hinaus ist ein Neuladen der gesamten Seite mit Hilfe von *meta*-Elementen möglich. Hier werden unaufgefordert periodische Aktualisierungen des Seiteninhalts oder eine Weiterleitung zu einer anderen Seite ausgeführt. Problematisch sind solche Aktualisierungen besonders für leseschwache Benutzer oder für Benutzer eindimensionaler Ausgaben, da *Screenreader* bei jedem Aufruf

einer Seite erneut am Seitenanfang beginnen, diese vorzulesen. Durch das zu kurze Zurverfügungstehen der Texte können Inhalte am Seitenende nicht erfasst werden [vgl. Hän]. Um diese Schwierigkeiten zu umgehen, wird auf unaufgeforderte Änderungen des Seiteninhalts durch periodische Aktualisierungen (Bedingung 7.4) sowie *clientseitige* automatische Weiterleitungen (Bedingung 7.5) verzichtet.

Die WCAG 1.0 stuft diese Anforderung als temporär ein. Entwickler müssen die genannten Punkte berücksichtigen bis Benutzeragenten eine Kontrolle dieser Fälle zur Verfügung stellen [WCAG1].

### Anforderung 8

*„Die direkte Zugänglichkeit der in Internetangeboten eingebetteten Benutzerschnittstellen ist sicherzustellen.“ [Bul]*

Programmierte Objekte, die eigene Benutzerschnittstellen aufweisen, müssen die Zugänglichkeit direkt oder mit Hilfe assistiver Technologien garantieren (Bedingung 8.1). Da diese Objekte eine Bedienung unabhängig vom Benutzeragenten ermöglichen, kann dieser nicht für eine geräteunabhängige Bedienung herangezogen werden. Es muss sichergestellt sein, dass Schnittstellen programmierter Objekte den „*Prinzipien zugänglichen Designs*“ [WCAG1], wie sie durch die BITV aufgestellt werden, folgen. Dazu gehört unter anderem eine geräteunabhängige Bedienung, die in Anforderung 9 näher erläutert wird.

### Anforderung 9

*„Internetangebote sind so zu gestalten, dass Funktionen unabhängig vom Eingabegerät oder Ausgabegerät nutzbar sind.“ [Bul]*

Wie bereits erwähnt sind Menschen mit Behinderungen unter Umständen nicht in der Lage, die üblichen Eingabegeräte (Standardtastatur) in Verbindung mit einem Zeigegerät (Maus), zu verwenden. Blinde Menschen können auf Grund

der fehlenden Hand-Auge-Koordination keine Zeigegeräte benutzen. Die Bedienung des Computers erfolgt daher lediglich über die Tastatur. Menschen mit motorischen Behinderungen haben je nach dem Grad ihrer motorischen Einschränkungen verschiedene Eingabegeräte zur Auswahl. Bei Koordinationsproblemen kann bereits eine Großfeldtastatur in Verbindung mit einer Tastaturmaus die üblichen Eingabegeräte voll ersetzen [vgl. Hel05, S.22]. Darüber hinaus gibt es die Möglichkeit, Dokumente über eine Spracheingabe zu erstellen [vgl. Hel05, S.22] oder Eingaben mit Hilfe eines mundgesteuerten Zeige Gerätes in Verbindung mit einer Bildschirmtastatur vorzunehmen [vgl. Hel05, S.21]. Um eine uneingeschränkte Nutzung von Internetangeboten zu ermöglichen, müssen diese *„sowohl mit alternativen Zeige geräten als auch mit einer Tastatur zu bedienen“* sein [Hel05, S.23], da Internetseiten, die *„eine Bedienung über die Tastatur erlauben, auch über Spracheingabe oder eine Kommandozeilen-Schnittstelle zugänglich“* sind [WCAG1]. Infolgedessen müssen alle Elemente, die eine Interaktion des Benutzers mit der Anwendung ermöglichen, über Tastatureingaben bedienbar sein (Bedingung 9.2). Da *serverseitige Imagemaps* eine Tastaturbedienung nicht ermöglichen sind *clientseitige Imagemaps* zu verwenden, soweit die definierbaren Formen für verweissensitive Regionen dies zulassen (Bedingung 9.1).

Bei der Bedienung einer Internetseite über Tastatureingaben springt der Benutzer mit Hilfe der Tabulatortaste von einem aktivierbaren Element zum nächsten. Um eine Tastaturbedienung der Internetseiten nicht nur zu ermöglichen sondern zusätzlich zu unterstützen, müssen die Elemente in einer sinnvollen Reihenfolge angesprungen werden (Bedingung 9.4). Zusätzlich müssen bedeutende Verweise sowie Formularkontrollelemente und Gruppen dieser Elemente durch Tastaturkurzbefehle erreichbar sein (Bedingung 9.5). Die Ausführung von Tastaturkurzbefehlen erfolgt, indem eine Steuerungs- sowie eine Buchstaben- oder Zifferntaste gleichzeitig betätigt werden. Nicht nur Elemente innerhalb einer Internetseite sind über diese Kurzbefehle zu erreichen, auch Befehle des Benutzeragenten sind ausführbar. Überschneiden sich die verwendeten Buchstaben- oder Zifferntasten, wird die Bedienung des Benutzeragenten eingeschränkt. Daher dürfen Tastenkombinationen, die in Benutzeragenten gebraucht werden, nicht zur Navigation in den Internetseiten



selbst verwendet werden. Bei Berücksichtigung aller gängigen Benutzeragenten und Betriebssysteme bleiben zur freien Belegung lediglich die Zifferntasten [vgl. HeI05, S.139].

*Scripts* werden in Internetauftritten verwendet, um auf bestimmte Ereignisse zu reagieren. Folgt die Reaktion auf ein geräteabhängiges Ereignis, wie das Bewegen der Maus über ein Element, darf die Reaktion keine zusätzlichen Informationen bieten, da diese Benutzern ohne Maus verborgen bleibt. Um Geräteunabhängigkeit zu gewährleisten, sind logische Ereignisse an Stelle von geräteabhängigen zu definieren (Bedingung 9.3).

### 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.“ [Bul]*

Unter dieser Anforderung sind Punkte zusammengefasst, die beim heutigen Stand der Technik zu berücksichtigen sind, um Barrierefreiheit zu ermöglichen, soweit die Anpassungen nicht unverhältnismäßig viel Aufwand bedeuten. In den WCAG 1.0 sind diese Punkte als vorläufig ausgezeichnet, da sie nur bis zu einer Anpassung der Benutzeragenten berücksichtigt werden müssen [WCAG1]. Die BITV verzichtet auf eine solche Kennzeichnung, statt dessen ist eine Überprüfung der gesamten Verordnung spätestens drei Jahre nach deren Inkrafttreten vorgesehen [vgl. Bul].

Die Sicherstellung der Barrierefreiheit bei der Verwendung älterer Benutzeragenten gilt als unverhältnismäßig, wenn diese älter als drei Jahre sind oder der Verbreitungsgrad der Benutzeragenten in der diesbezüglichen Benutzergruppe unter fünf Prozent liegt [vgl. Beh]. Die Vorbelegung von Formularelementen mit Platzhaltern (Bedingung 10.4) ist aus diesem Grund nicht mehr zur Erfüllung der BITV notwendig. Die Problematik, dass leere Eingabefelder nicht

erkannt wurden, trat bei der *Software Outspoken* für den *Macintosh* sowie in *Netscape*-Versionen der zweiten Generation auf [vgl. Hel05, S.187].

Noch immer aktuell scheint hingegen die Problematik, die für die Benutzer von eindimensionalen Ausgabemedien entsteht, wenn aufeinander folgende Verweise nicht deutlich (das heißt durch ein von Leerzeichen umgebenes druckbares Zeichen) voneinander getrennt werden (Bedingung 10.5). Vorleseprogramme hängen die Texte mehrerer Verweise aneinander, sodass der Benutzer nicht in der Lage ist, die Verweise voneinander zu trennen. Eine Trennung ist besonders in der Navigation zu beachten, da diese das Rückgrat eines Internetangebots darstellt. Hier können Schwierigkeiten entstehen, wenn mehrere aufeinanderfolgende Verweise von der Sprachausgabe lediglich durch eine Stimmänderung gekennzeichnet werden.

Ebenfalls problematisch kann die eindimensionale Ausgabe eines zweiseitigen Textes sein. Wird der Text durch die Verwendung einer Tabelle mehrspaltig dargestellt, produzieren ältere Screenreader oft Ausgabefehler. Der Text wird Zeile für Zeile abgearbeitet, ohne die Spalten zu berücksichtigen (vgl. Hel05, S.155). Wird diese Art der *Layouttabelle* verwendet, muss der enthaltene Text zusätzlich in linearer Form bereitgestellt werden (Bedingung 10.3).

Weiter ist für Benutzer einer eindimensionalen Ausgabe das automatische Öffnen neuer Fenster nicht so offensichtlich wie bei der Verwendung grafischer Ausgabemedien. Das Wechseln des Fensterfokusses kann den Benutzer dieser Ausgabemöglichkeiten leicht verunsichern. Da ältere Benutzeragenten die Unterdrückung von unaufgefordert erscheinenden Fenstern nicht unterstützen, sind solche Wechsel der Ansicht zu vermeiden (Bedingung 10.1). Wird das Ziel einer Verknüpfung in einem neuen Fenster geöffnet, darf dies nur nach Aufforderung oder Ankündigung geschehen, um Benutzer einer Sprachausgabe nicht zu verwirren.

Solange Benutzeragenten die explizite Beschriftung von Formularelementen durch die Verwendung entsprechender Elemente noch nicht unterstützen, müssen die impliziten Beschriftungen der Elemente korrekt positioniert sein

(Bedingung 10.2). Dies gewährleistet, dass blinde Benutzer Formularelemente eindeutig identifizieren können [vgl. Hel05, S.191]. Korrekt positioniert bedeutet, dass die Konvention, die durch die verschiedenen Betriebssysteme vorgegeben wurde, eingehalten wird. Nach dieser sind Beschriftungen von Formularelementen unmittelbar vor dem Element zu platzieren. Ausnahmen bilden Auswahlwähler, hier wird die Beschriftung erst hinter dem Formularelement eingefügt [vgl. Hel05, S.191 ff.].

### 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.“ [Bul]*

Um Internetangebote zu realisieren, sind aktuelle Versionen öffentlich zugänglicher und dokumentierter Technologien zu verwenden (Bedingung 11.1). Überholte Funktionalitäten sind zu Gunsten der Neuerungen nicht mehr einzusetzen (Bedingung 11.2). Internetangebote, die auf öffentlichen Standards statt proprietären Technologien beruhen, sind mit Benutzeragenten ohne die Verwendung zusätzlicher Software benutzbar. Zur Zeit stehen zwei Standards des W3C zur Auswahl: XHTML 1.0 und HTML 4.01, wobei XHTML 1.0 die Empfehlung des W3C für Internetauftritte darstellt. Durch die vollständige Kompatibilität zu XML und damit zu anderen Auszeichnungssprachen ist XHTML die zukunftssichere Variante [vgl. Hel05, S.212]. Werden dennoch nicht-zugängliche Technologien eingesetzt, gilt es zusätzlich ein barrierefreies Angebot mit äquivalenten Inhalten bereitzustellen (Bedingung 11.3).

Über die Verwendung standardisierter Technologien hinaus benötigen manche Benutzergruppen weitere Unterstützung durch die Bereitstellung zusätzlicher Seiten, die den Inhalt in einer anderen, empfängerorientierten Form aufbereiten (Bedingung 11.4). Denkbar sind hier beispielsweise Audioinformationen statt Text für leseschwache Benutzer.

## Anforderung 12

*„Der Nutzerin/dem Nutzer sind Informationen zum Kontext und zur Orientierung bereitzustellen.“ [Bul]*

Für Menschen mit kognitiven Behinderungen oder Benutzer eines *Screen-readers* ist es schwierig, komplexe Beziehungen innerhalb der Seitenstruktur eines Internetauftrittes zu verstehen. Daher sieht die BITV explizite Beschreibungen für die Beziehungen bestimmter Seitenelemente vor.

Bei der Benutzung eines eindimensionalen Ausgabemediums ist die gleichzeitige Darstellung mehrerer durch *Frames* voneinander getrennten Bildschirmbereiche nicht möglich. Der Benutzer erhält eine Liste mit den Kennzeichnungen der vorhandenen *Frames*. Mit Hilfe dieser Übersicht kann er zwischen den einzelnen Bereichen wechseln. Sind die *Frames* nicht aussagekräftig benannt, kann sich der Benutzer nicht orientieren und muss jeden einzelnen *Frame* auswählen, um dessen Inhalt zu erkennen. Um dies zu vermeiden, muss jedem Bildschirmbereich ein aussagekräftiger Titel (Bedingung 12.1) sowie wenn nötig seine Beziehungen zu anderen Bereichen (Bedingung 12.2) zugeordnet werden.

Um das Verständnis von großen Informationsblöcken zu erleichtern, gilt es diese in Internetauftritten mittels *Strukturmarkup* in mehrere kleine Einheiten zu unterteilen (Bedingung 12.3). Zu berücksichtigen sind hier lange Texte, die strukturiert durch Paragraphen und Überschriften in verschiedenen Ebenen leichter zu verarbeiten sind. Darüber hinaus können umfangreiche Formulare in kleine Gruppen mit zusammengehörigen Kontrollelementen gegliedert werden, wobei jedem Kontrollelement eine Beschriftung zuzuordnen ist (Bedingung 12.4).

**Anforderung 13**

*„Navigationsmechanismen sind übersichtlich und schlüssig zu gestalten.“ [Bul]*

Die Navigation bildet das Rückgrat eines Internetauftrittes, ohne sie kann ein Benutzer einen Internetauftritt nicht vollständig erleben. Ist sie nicht schlüssig angelegt, wird es für einen Benutzer schwer, sich einen Überblick über die Seite zu verschaffen. Da die Navigation eines Auftrittes aus der Gesamtheit der enthaltenen Verweise besteht, ist die Basis für eine Orientierung eine geeignete Kennzeichnung der vorhandenen Verweise (Bedingung 13.1). *Screenreader* ziehen die Information eines Verweises aus dem vorhandenen Text in dem Verweis selbst oder, bei Verweisen, die durch Grafiken umgesetzt sind, aus deren Text-Äquivalent. *Screenreader* stellen Funktionen zur Verfügung, die es dem Benutzer ermöglichen, Verweise getrennt vom Inhalt einer Seite zu betrachten. Damit Verweise auch isoliert benutzt werden können, ist es wichtig, dass Verweistexte Informationen über das Ziel des Verweises enthalten.

*Screenreader* geben bei dem Aufruf einer neuen Seite häufig einleitend deren Dokumententitel aus. Dies ermöglicht dem Benutzer ein sofortiges Erkennen der aktuellen Seite. Um solche Ausgaben zu erlauben, müssen (X)HTML-Dokumente semantische Informationen enthalten (Bedingung 13.2). (X)HTML sieht für solche Informationen Metadaten vor.

Jede Seite eines Internetauftrittes stellt Navigationsmöglichkeiten in Form einer Liste an Verweisen zur Verfügung (Bedingung 13.5). In dieser Liste, dem Navigationsmenü, sollte der aktuelle Standpunkt des Benutzers hervorgehoben werden, um dessen Orientierung zu unterstützen. Für die Benutzer eindimensionaler Ausgabemedien ist es nötig, diese Hervorhebung nicht nur visuell zu gestalten, sondern sie auch durch textuelle Information zu unterstützen. Zusätzlich ist das Umgehen einer solchen Gruppierung von Verweisen durch entsprechende Sprungmarken zu ermöglichen (Bedingung 13.6), da das Durcharbeiten einer solchen Liste bei Tastaturbedienung einen hohen Zeitauf-

wand bedeutet. Entsprechend ist das Überspringen von ASCII-Zeichnungen sicherzustellen (Bedingung 13.10).

Um Benutzern, denen es nicht möglich ist, sich durch das Navigationsmenü einen Überblick des gesamten Internetauftritts zu verschaffen, die Orientierung in einem Internetauftritt zu erleichtern, sollte eine Gesamtübersicht des Angebots zur Verfügung gestellt werden (Bedingung 13.3). Zusätzlich zu einer Übersichtsseite kann eine numerische Beschriftung des Navigationsmenüs Benutzern einer eindimensionalen Ausgabe helfen, die hierarchische Struktur des Auftritts zu verstehen (Bedingung 13.4). Um einen benutzerdefinierten Zugriff auf die Inhalte des Internetangebots zu gewährleisten, wird oft eine Suchfunktion als Zugangsmöglichkeit angeboten. Um eine solche Funktion für einen möglichst großen Personenkreis zugänglich zu gestalten, sind verschiedene Arten der Suche anzubieten (Bedingung 13.7). Denkbar sind hier neben der Volltextsuche eine phonetische Deutung der Suchbegriffe oder eine Filterfunktion in den gesamten Seiteninhalten.

Um die Informationsaufnahme zu erleichtern, ist es sinnvoll, Information in handhabbare Blöcke zu zerlegen. Aussagekräftige Kurzbeschreibungen über die in einem Informationsblock enthaltene Information (Bedingung 13.8) helfen besonders Benutzern von Screenreadern, da durch sie ein Überspringen von Blöcken ermöglicht wird und die gesuchten Informationsblöcke schneller erreicht werden. Neben reinen Textblöcken sind zusätzliche Beschreibungen auch für Listen, wie beispielsweise die Navigation, vorgesehen. Bei der getrennten Darstellung zusammenhängender Informationen muss diese auch als Zusammenstellung zur Verfügung stehen (Bedingung 13.9). Dadurch kann sichergestellt werden, dass die Information dem Benutzer vollständig zugänglich ist.

**Anforderung 14**

*„Das allgemeine Verständnis der angebotenen Inhalte ist durch angemessene Maßnahmen zu fördern.“ [Bul]*

Für Menschen mit kognitiven Problemen, Lese- oder Lernschwierigkeiten stellen komplexe Internetauftritte hohe Anforderungen dar. Durch eine einfach gehaltene Sprache wird die Informationsvermittlung besonders für diese Menschen effektiver gestaltet (Bedingung 14.1). Um eine klare Sprache zu erreichen, ist die Verwendung von Fremdwörtern und Fachbegriffen, Abkürzungen sowie zusammengesetzten Wörtern zu vermeiden.

Für Menschen mit Einschränkungen des Lese- und Lernvermögens ist es ebenfalls hilfreich, textuell dargestellte Information durch Nicht-Text-Elemente zu ergänzen (Bedingung 14.2). Bieten Internetauftritte lange Texte zusätzlich in akustischer Form an, hilft das Benutzern mit Leseschwierigkeiten die enthaltenen Information selbstständig aufzunehmen [vgl. Hel05, S.116-117].

Schließlich kommt eine konsistente Gestaltung der Seiten eines Internetauftrittes (Bedingung 14.3) unter anderem Menschen mit Lernschwierigkeiten zugute. Wurde die Funktion des Auftrittes einmal verstanden, kann das Gelernte auf jeder Seite erneut angewendet werden.

## 5 Gebrauchstaugliche Internetauftritte

Im letzten Kapitel wurden verschiedene technische und nichttechnische Anforderungen aufgezeigt, denen Informationstechnik entsprechen muss, um für behinderte Menschen benutzbar zu sein. Die im Rahmen der Diplomarbeit entstehenden Internetangebote sind nicht nur für behinderte Menschen benutzbar zu gestalten. Auch nichtbehinderte Benutzer haben durch das Überangebot an Information im Internet bestimmte Anforderungen und wenden sich von Internetangeboten ab, die bestimmte Mindestanforderungen

nicht erfüllen. Dieses Kapitel geht auf diese Anforderungen ein und erläutert, wie benutzerfreundliche Angebote zu gestalten sind.

Das Internet hat in den letzten Jahren die Benutzung traditioneller Medien wie Druck, Fernsehen und Rundfunk in vielen Bereichen ergänzt oder abgelöst. Als Handelsmedium, zur Kommunikation, Information sowie zur Unterhaltung ist die Benutzung des weltumspannenden Netzverbundes für viele Menschen alltäglich geworden. Während im ersten Quartal 1999 nur 13,1 Prozent der deutschen Bevölkerung das Internet benutzten, lag dieser Wert im ersten Quartal 2004 bereits bei 55,1 Prozent. Dies entspricht einem Anstieg von 8,3 Millionen auf 35,3 Millionen Menschen in den vergangenen fünf Jahren [vgl. SIL, S.2]. Die Entwicklung zu einem Massenmedium wurde durch die in den letzten Jahren stark gestiegene Verbreitung, die sinkenden Zugriffszeiten sowie die immer billiger und einfacher werdenden Zugriffsmöglichkeiten gefördert.

Diese Entwicklung löste auch eine Veränderung der Internetbenutzung aus. Während in den Anfangsjahren das Internet von Experten dominiert wurde, handelt es sich heute um ein Alltagsmedium für jedermann. Für die angebotenen Inhalte bedeutet das die Notwendigkeit einer hohen Benutzerorientierung. Ist diese nicht oder kaum vorhanden, bringen heutige Benutzer diesem Umstand kein Verständnis mehr entgegen und wenden ihre Aufmerksamkeit von den entsprechenden Internetangeboten ab. Nach Jakob Nielsen sind „*alle weltweiten Mitbewerber [...] nur einen Mausklick entfernt*“ [Nie00, S.9]. Dadurch wird ein bedeutender Unterschied zwischen herkömmlicher *Software* und Internetangeboten deutlich. *Software* wird käuflich erworben und erst nach dem Bezahlen kann der Benutzer die *Software* verwenden und ihre Nützlichkeit erfahren [vgl. MM02, S.16]. Durch diese zeitliche Abfolge ist der Benutzer bemüht, die Handhabung einer neuen *Software* zu verstehen und sie mit Hilfe von Handbüchern zu erlernen. Bei Internetinhalten liegt diese zeitliche Abfolge in umgekehrter Reihenfolge vor. Die Inhalte sind vor einer Bezahlung, die dem Ersteller zugute kommt, benutzbar. Da Benutzer aus diesem Grund nicht bereit sind, die Handhabung eines Internetangebots ähnlich der einer *Software* zu erlernen, muss sich die Gestaltung von Internetauftritten besonders an den Erwartungen der Benutzer orientieren.



## 5.1 Gebrauchstauglichkeit

Entspricht ein Internetangebot den Erwartungen eines Benutzers und ermöglicht ihm dadurch eine benutzerfreundliche Bedienung, so entsteht für den Benutzer eine hohe Gebrauchstauglichkeit des Internetauftrittes. Häufig wird für diese Tatsache auch im deutschen Sprachraum das englische Wort *Usability* verwendet. Da hinter diesem englischen Schlagwort viele unterschiedliche Ansätze stehen, ist Gebrauchstauglichkeit nur eine der möglichen Übersetzungen. Unter anderem finden auch die Begriffe Benutzerfreundlichkeit oder Benutzbarkeit Verwendung.

Eine allgemein gültige Definition des Begriffs ist in der ISO-Norm 9241 festgehalten: Gebrauchstauglichkeit *„bezeichnet das Ausmaß, in dem ein Produkt durch bestimmte Benutzer in einem bestimmten Nutzungskontext genutzt werden kann, um bestimmte Ziele effektiv, effizient und mit Zufriedenheit zu erreichen“* [zitiert nach vGi02, S.2]. Auf die Gebrauchstauglichkeit eines Internetangebots wirken, abhängig von dem Nutzungszweck und der Zielgruppe, drei Größen ein:

- **Effektivität:** *„Die Genauigkeit und Vollständigkeit, mit der Benutzer ein bestimmtes Ziel erreichen“* [zitiert nach vGi02, S.2]. Effektivität umschreibt daraus folgend die Tatsache, dass ein Benutzer sein Ziel erreicht, ohne zu beachten wie viel Aufwand er dafür benötigt.
- **Effizienz:** *„Der im Verhältnis zur Genauigkeit und Vollständigkeit eingesetzte Aufwand, mit dem Benutzer ein bestimmtes Ziel erreichen“* [zitiert nach Bar03, S.7]. *„Der relative Aufwand kann psychische oder physische Beanspruchung, Zeit, Material oder monetäre Kosten enthalten“* [zitiert nach vGi02, S.3]. Effizienz bedeutet, dass ein Benutzer zum Erreichen seiner Ziele den geringstmöglichen Einsatz aufbringen muss.
- **Zufriedenheit:** *„Maße der Zufriedenheit beschreiben die Beeinträchtigungsfreiheit und die Akzeptanz der Nutzung. [...] Maße der Zufriedenheit können sich auf Einstellungen beziehen, ein Produkt zu benutzen, oder auf das Benutzerurteil über Aspekte wie Effizienz, Nützlichkeit und Lernförderlichkeit“* [zitiert nach vGi02, S.3]. Die Zufriedenheit orientiert sich stark

an den Erwartungen des Benutzers und wird erreicht, wenn diese Erwartungen mindestens erfüllt oder auch übertroffen werden.

Aus diesen drei Größen ergibt sich, dass ein gebrauchstaugliches Internetangebot dem Benutzer ermöglicht, seine selbstgestellten Bedürfnisse mit der kleinstmöglichen Aufwendung an Zeit, Geld, Kraft und Nerven zu befriedigen. Um den Benutzer dabei zu unterstützen, seine Ziele mit einem möglichst geringen Aufwand zu erreichen, muss sich der Aufbau eines Internetauftritts am Verhalten des Benutzers orientieren. Im nächsten Abschnitt wird das Verhalten eines Benutzers im Internet näher untersucht und erläutert, wie ein Internetangebot auf dieses Verhalten ausgerichtet werden kann.

## 5.2 Benutzerverhalten im Internet

Bereits die Definition in der ISO-Norm 9241 sagt aus, dass die Gebrauchstauglichkeit eines Produktes nicht allgemein gültig ist, sondern sich auf einen bestimmten Benutzer in einem bestimmten Benutzungskontext und auf das Erreichen bestimmter Ziele bezieht. Daraus folgend sind kontextübergreifende Lösungen nicht möglich. Bezogen auf Internetangebote ist jedoch davon auszugehen, dass verschiedene Verhaltensmuster der Benutzer vorausgesetzt werden können. Davon ausgehend kann ein Internetangebot für eine unbestimmte Zielgruppe mit eingeschränkter Gültigkeit an die Benutzer angepasst und dadurch eine eingeschränkte Gebrauchstauglichkeit erreicht werden. Nachfolgend sind verschiedene Verhaltensmuster aufgezeigt. Zudem wird erläutert, wie es möglich ist, ein Internetangebot an diese anzupassen.

### 5.2.1 Lesen von Texten am Bildschirm

Im Vergleich zu dem Lesen von auf Papier gedruckten Texten benötigen Menschen beim Lesen am Bildschirm circa 25 Prozent mehr Zeit [Nie00, S.106]. Dieser erhöhte Zeitaufwand resultiert aus der Tatsache, dass das Lesen am Bildschirm für das menschliche Auge größere Anstrengung bedeutet als das Lesen auf Papier. Im Gegensatz zu Papier ist ein Bildschirm eine aktive Lichtquelle. Müssen sich die Augen beim Lesen auf diese Lichtquelle konzentrieren,

führt das zu einer schnelleren Ermüdung. Eine gute typographische Gestaltung kann eine bessere Lesbarkeit der Texte am Bildschirm ermöglichen. Das menschliche Auge wird durch klare optische Kontraste zwischen Text und Hintergrund, aber auch zwischen den einzelnen Textabsätzen, entlastet und geführt. In den folgenden Abschnitten werden verschiedene Textmerkmale genannt. Weiterhin wird dargestellt, welche Eigenschaften das Lesen am Bildschirm positiv beeinflussen.

### **Hintergrund**

Die größte Kontrastwirkung wird durch die Kombination der Farben schwarz und weiß erreicht unabhängig davon, ob es sich um schwarzen Text auf weißem Hintergrund handelt oder umgekehrt. Da schwarzer Text auf weißem Hintergrund jedoch sehr viel häufiger (auch im Druckbereich) angewendet wird, ist diese Kombination besser lesbar, während die Umkehrung dieses Verhältnisses den Benutzer eher verwirrt [vgl. Nie00, S.125]. Auch wenn die Kombination schwarz auf weiß den besten Kontrast ergibt, ist die Verwendung eines hellen, farbig abgetönten Hintergrundes für Internetanwendungen besser geeignet. Beim Lesen auf Papier gedruckter Texte passt sich die Helligkeit des Papiers durch das Reflektieren des Umgebungslichtes den Lichtverhältnissen der Umgebung an. Ein Bildschirm ist dazu nicht in der Lage, da dieser eine aktive Lichtquelle darstellt. Beim ständigen Wechsel zwischen dem Lesen am Bildschirm und dem Betrachten der Umgebung muss das Auge immer wieder seine Wahrnehmungssensibilität anpassen [vgl. Pus01, S.37]. Bei durchschnittlichen Lichtverhältnissen ist der Intensitätsunterschied zwischen dem Umgebungslicht und einem Bildschirm mit weißem Hintergrund wesentlich größer als bei der Verwendung eines leicht abgetönten Bildschirmhintergrundes. Zusätzlich entstehen bei großen weißen Flächen am Bildschirm starke Blendeffekte [vgl. Hel05, S.82], daher kann die Benutzung eines weißen Hintergrundes die Augen des Benutzers noch schneller ermüden lassen.

Die Verwendung von Hintergrundgrafiken sollte vermieden werden, da durch sie der Kontrast zwischen Text und Hintergrund nicht mehr einheitlich, son-

dem unterschiedlich ist. Dadurch wird die Wahrnehmung einzelner Buchstaben oder Wörter gestört [vgl. Nie00, S.126].

### **Schriftart**

Die Darstellung von Text auf einem Bildschirm ist abhängig von dessen technischen Gegebenheiten. Computerbildschirme arbeiten meist mit einer Auflösung kleiner als 85 *dots per inch* (dpi) [vgl. HL99, S.97]. Das bedeutet im Vergleich zu der Auflösung, mit der Bücher oder Zeitschriften erstellt werden, häufig 1200 dpi [vgl. HL99, S.97], eine sehr geringe Auflösung. Daher sind Schriften, die in gedruckter Form über eine sehr gute Lesbarkeit verfügen, nicht automatisch auch in der Bildschirmdarstellung gut lesbar. Gedruckte Texte werden durch die Darstellung in einer Serifenschrift besser lesbar. Die kleinen Begrenzungsstriche an den Buchstabenenden leiten das Auge und erleichtern dadurch das Beibehalten der Zeile. Bei der niedrigen Auflösung am Bildschirm kann die Darstellung von Serifen jedoch zu Problemen führen. Besonders bei kleinen Schriftgrößen kann es zu unproportionaler Darstellung kommen, wenn Serifen auf Pixelgrenzen fallen und sich immer über mindestens einen Pixel erstrecken müssen [vgl. Wil]. Da unregelmäßige Darstellungen der Buchstaben die Lesbarkeit negativ beeinflussen, sind für die Fließtexte bei Internetangeboten serifenlose Schriften zu bevorzugen.

### **Schriftgröße**

Im Allgemeinen gilt eine Schrift in der Schriftgröße 12 bis 14 Punkt am Bildschirm als gut lesbar. Jedoch wirkt sich nicht nur die Größe der Schrift auf die Lesbarkeit aus, sondern auch die Größe des Raumes zwischen den Textzeilen. Der Raum zwischen den Grundlinien zweier aufeinander folgender Zeilen wird als Durchschuss bezeichnet. Bei gedruckten Texten sollte der Durchschuss zwei Punkte größer sein als die Schriftgröße [vgl. HL99, S.105]. Aufgrund der geringeren Auflösung am Bildschirm empfehlen Sarah Horton und Patrick J. Lynch hier einen Durchschuss von 14 bis 16 Punkt bei einer Schriftgröße von 12 Punkt [vgl. HL99, S.105].

### **Zeilenlänge**

Bei normalem Leseabstand kann das menschliche Auge nur etwa acht Zentimeter fokussieren [vgl. HL99, S.103]. Bei der Wahl der Zeilenlänge sollte man diesen Wert berücksichtigen und nicht zu sehr überschreiten. Beim Lesen längerer Zeilen muss der Fokus durch die Augenmuskulatur oder durch Bewegung des Kopfes an der Zeile entlang geführt werden. Deswegen führt der Sprung zum Zeilenbeginn häufig zum Verlust der Zeile und die Lesbarkeit des Textes wird gemindert. Sarah Horton und Patrick J. Lynch geben einen Richtwert von 365 Pixeln für die Breite eines Textes an. Dies ergibt mit einer *„Zeilenlänge von rund 50 Zeichen mit durchschnittlich sechs bis zehn Wörtern pro Zeile“* eine gute Lesbarkeit und eine optimale Raumnutzung [vgl. HL99, S.103 ff.].

### **Textausrichtung**

Für die Textausrichtung gibt es vier Möglichkeiten: Linksbündig, rechtsbündig, zentriert und Blocksatz.

Im Blocksatz schließt jede Zeile mit dem linken und dem rechten Textrand ab. Dadurch entstehen blockartige Absätze. Um ein bündiges Abschließen auf beiden Seiten zu ermöglichen, muss das ausführende Programm selbstständig Silbentrennungen einfügen und einzelne Zeilen durch Skalieren der Wortzwischenräume an die Zeilenbreite anpassen. Spezielle Textprogramme greifen auf ein Wörterbuch zurück, um die Silbentrennungen korrekt einzufügen. Benutzeragenten ermöglichen solche Silbentrennungen nicht [vgl. He105, S.93]. Dadurch werden Zeilen lediglich mit einem groben Ausgleich durch die Skalierung der Wortzwischenräume der Textbreite angepasst. Die entstehenden Lücken führen zu einer schlechten Lesbarkeit [vgl. HL99, S.101 ff.].

Bei der linksbündigen, rechtsbündigen und der zentrierten Textausrichtung bleibt mindestens ein Textrand im Flattersatz, das heißt mit unregelmäßigen Zeilenenden, bestehen. Ein Ausgleich der Zeilenlänge ist daher nicht erforderlich.

Das Lesen erfolgt im europäischen Sprachraum von links nach rechts. Das Auge wandert beim Lesen vom linken zum rechten Textrand und springt dann zurück nach links zum Anfang der nächsten Zeile. Befindet sich am linken Rand keine einheitliche Linie sondern ein zerklüfteter Text hat es das Auge schwer, die nächste Zeile zu finden, da eine vertikale Orientierungslinie fehlt. Diese Linie ist nur bei einer linksbündigen Ausrichtung zu finden, da alle Zeilen bündig am linken Textrand beginnen und so einen gleichmäßigen und absehbaren Rand ergeben [vgl. HL99, S.102].

Je weniger Anstrengung der Benutzer für den Lesevorgang an sich benötigt, desto mehr Reserven stehen ihm für die Verarbeitung des Inhalts zur Verfügung. Auch bei optimaler Lesbarkeit der Texte unterscheidet sich das Leseverhalten im Internet vom Lesen gedruckter Texte. Da das Leseverhalten Auswirkungen auf die Verarbeitung der im Text enthaltenen Informationen hat, geht der nächste Abschnitt über das Lesen am Bildschirm hinaus auf die Besonderheiten beim Lesen von Internettexten ein.

### 5.2.2 Leseverhalten im Internet

Das Leseverhalten im Internet wird von weiteren Faktoren beeinflusst. Jakob Nielsen nennt diesbezüglich folgende Punkte [vgl. Nie00, S.106]:

- Das Internet stellt ein benutzergesteuertes Medium dar, durch welches sich der Benutzer aktiv bewegt. Diese Aktivität nimmt dem Benutzer die Ruhe, sich auf einen Text konzentrieren zu können.
- Der Benutzer wird im Internet mit einem Informationsüberangebot konfrontiert.
- Der Benutzer steht unter Zeitdruck.

Im Internet werden Texte nicht mit einer gleich bleibend hohen und gezielten Aufmerksamkeit gelesen, wie das bei einem Buch der Fall ist. Vielmehr werden Texte in einem dreistufigen Leseprozess durchgearbeitet. Dieser Prozess ermöglicht dem Benutzer die Aufnahme der im Text enthaltenen relevanten Informationen, ohne den Text vollständig Wort für Wort zu lesen. Somit wird das anstrengende Lesen von Text am Bildschirm minimiert.

Thomas Wirth benennt die drei Phasen in dieser zeitlichen Reihenfolge mit den Begriffen *Scannen*, *Skimmen* und Lesen [vgl. Wir04, S.206 ff.]:

- **Scannen:** Die Inhalte einer Internetseite werden zuerst mit einer hohen *schwebenden Aufmerksamkeit* überflogen. Die *schwebende Aufmerksamkeit* ermöglicht ein schnelles Absuchen hervorgehobener Informationen wie zum Beispiel Überschriften nach relevanten Inhalten. Beim *Scannen* kann in kurzer Zeit sehr viel Information verarbeitet werden, jedoch ist die Qualität der Verarbeitung und damit die Menge der aufgenommenen Information sehr gering.
- **Skimmen:** Wurden relevante Informationen gefunden, wird die Internetseite mit einer geringeren Geschwindigkeit, aber einer noch immer hohen *schwebenden Aufmerksamkeit* betrachtet. Neben den Überschriften werden nun auch Teile des Fließtextes wie Absatzanfänge und hervorgehobener Text beachtet. Sowohl Gedächtnisleistung als auch Aufnahme von Information steigen.
- **Lesen:** Bei niedriger Lesegeschwindigkeit nimmt die Verarbeitungstiefe und damit die aufgenommene Informationsmenge zu. Durch die nun dominierende *fokussierende Aufmerksamkeit* ist eine systematische und vollständige Aufnahme der Informationen möglich.

In einer Studie stellten Jakob Nielsen und John Morkes fest, dass 79 Prozent der Teilnehmer aufgerufene Internetseiten mit einer schwebenden Aufmerksamkeit überflogen, anstatt die enthaltenen Texte Wort für Wort zu lesen [vgl. Nie00, S.104]. Daraus folgend sollte die Aufbereitung der Textinformationen eines Internetauftritts dem Leseverhalten der Benutzer angepasst werden. Um das *Scannen* eines Textes zu erleichtern, sollte dieser in Absätze angemessener Größe gegliedert sein. Durch hervorgehobene Überschriften ist das

Thema eines Absatzes schnell zu erfassen. Wird ein Absatz angelesen, kann eine angepasste Strukturierung des Inhalts den Benutzer beim *Skimmen* unterstützen. Der Aufbau eines Absatzes sollte dem Prinzip der *umgekehrten Pyramide* folgen. Hier stehen wichtige Informationen oder eine Kurzzusammenfassung bereits zu Beginn des Absatzes.

### 5.2.3 Ladezeiten im Internet

Die Länge der Wartezeiten, die bei dem Aufruf einer Internetseite entstehen, wirkt sich auf die Zufriedenheit des Benutzers aus. Kurze Zeiten beeinflussen diese Größe positiv, lange hingegen negativ.

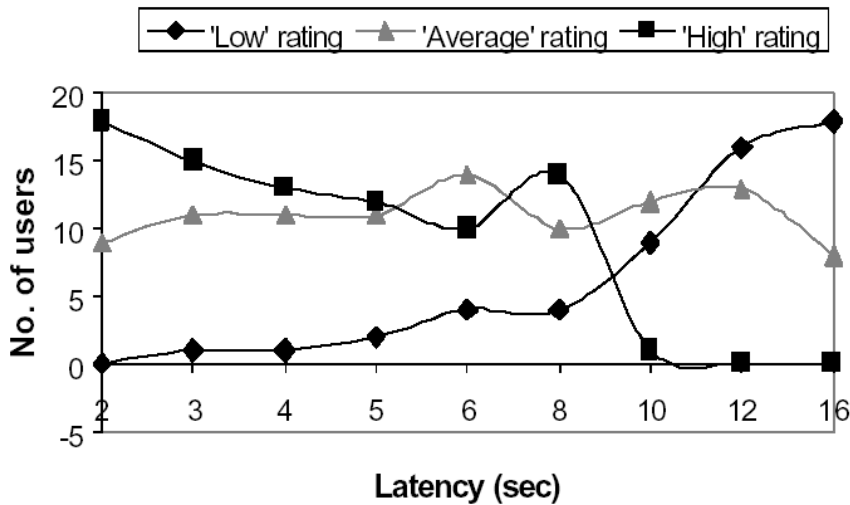
Grundsätzlich kann man bei den Antwortzeiten von Computeranwendungen mit folgenden Auswirkungen beim Benutzer rechnen [vgl. Nie00, S.42 ff.]:

- Bis 0,10 Sekunde: Die Verzögerung wird vom Benutzer nicht wahrgenommen, er bekommt eine unmittelbare Reaktion der Anwendung.
- Bis eine Sekunde: Die Verzögerung wird wahrgenommen, wobei die Gedankengänge des Benutzers ununterbrochen bleiben.
- Bis zehn Sekunden: Die Verzögerung wird als Wartezeit wahrgenommen. Die Aufmerksamkeit des Benutzers bleibt bei der Anwendung.
- Länger als zehn Sekunden: Die Verzögerung wird zur Unterbrechung. Der Benutzer wendet seine Aufmerksamkeit von der Anwendung ab, unter Umständen wird der Dialog beendet.

Eine Studie von Anna Bouch, Allan Kuchinsky und Nina Bhatti zeigt, dass diese Aussagen auch auf die Antwortzeiten bei einem Aufrufen von Internetseiten übertragbar sind [vgl. BKB].

Die Teilnehmer sollten in einem Teil der Studie die Antwortzeiten von Internetseiten in drei Stufen (schnell, durchschnittlich und mittel) einteilen. Die Antwortzeiten der Internetseiten waren vorbestimmt und variierten zwischen zwei und 73 Sekunden.





**Abbildung 5.1:** Bewertungen der Antwortzeiten von Internetseiten [BKB]

Das Ergebnis ist in Abbildung 5.1 [BKB] dargestellt. Bis zu einer Antwortzeit von acht Sekunden wurde der Ladevorgang einer Internetseite von den meisten Teilnehmern als schnell oder durchschnittlich empfunden. Nur wenige Teilnehmer empfanden bereits diese Zeitspanne als zu lang. Bei einer Antwortzeit von acht bis zehn Sekunden kehrt sich dieses Verhältnis um und bereits bei zwölf Sekunden Antwortzeit wird der Ladevorgang von keinem der Teilnehmer mehr als schnell bewertet. Daraus folgt, dass auch die Antwortzeiten von Internetseiten die Grenze von zehn Sekunden nicht überschreiten dürfen.

Wie lange das Laden einer Internetseite dauert, ist abhängig von Seitengröße und Qualität der Internetverbindung. Die Seitengröße bezeichnet die Summe aller Dateigrößen, die eine Internetseite ausmachen. Dazu gehören neben der (X)HTML-Datei alle in sie eingebetteten Objekte wie zum Beispiel Bilddateien und CSS-Dateien. Die Internetverbindung wird in diesem Zusammenhang mit der Übertragungsrate charakterisiert. Sie sagt aus, wie viele Daten zeitabhängig auf den Computer des Benutzers übertragen werden.

Jakob Nielsen legte im Jahr 2000 die Obergrenze der Seitengröße bei 34 KByte fest [vgl. Nie00, S.48]. Frank Puscher wählte 2001 einen Wert von 50 KByte [vgl. Pus01, S.18]. Den Berechnungen von Nielsen wurde die Übertragungsrate eines 28.8k-Modems zu Grunde gelegt. Heute ist die kleinste zu berücksichti-

gende Übertragungsrate die eines 56k-Modems. Bei der optimalen Übertragung können über ein 56k-Modem 56 KBit, also sieben KByte, pro Sekunde übertragen werden. Damit die maximale Antwortzeit von zehn Sekunden nicht überschritten wird, muss die Größe jeder Seite unter 70 KByte bleiben. Um eine längere Antwortzeit auch bei einer geringeren als der optimalen Übertragungsrate von sieben KByte pro Sekunde zu gewährleisten, sollte die Summe der Größen aller für eine Internetseite benötigten Dateien den von Puscher gewählten Wert von 50 KByte nicht überschreiten.

## 5.2.4 Struktur eines Internetauftritts

Ein Internetangebot besteht aus einzelnen Internetseiten. Durch die Verknüpfung der einzelnen Seiten entsteht eine logische Struktur des Internetangebots. Diese Strukturierung hilft dem Benutzer, die Anordnung der Informationen zu verstehen und sich zu orientieren. Auf der Struktur eines Internetangebots baut dessen Navigation auf. Nach Sarah Horton und Patrick J. Lynch gibt es vier Möglichkeiten, die Information eines Internetauftrittes zu strukturieren: sequenziell, gerastert, hierarchisch und netzartig [vgl. HL99, S.40 ff.].

Sequenzen stellen die Informationen als lineare Abfolge einzelner Seiten dar. Weisen Informationen bereits lineare Strukturen auf, werden diese sehr gut durch eine Sequenz wiedergegeben. Bei umfangreichen Internetauftritten gibt es neben der Möglichkeit, die Information linear abzuarbeiten, noch Zusatzinformationen, die abzweigend vom Hauptstrang angeboten werden.

Die gerasterte Struktur bietet sich für Internetauftritte an, die mehrere Variablen zueinander in Beziehung setzen und dabei eine einheitliche Ordnung von Themen und Unterthemen aufweisen. Es entsteht eine umfangreiche Datentabelle.

Hierarchisch aufgebaute Internetauftritte ordnen die Informationen mit zunehmender Detailgenauigkeit um eine Startseite an. Die Informationen werden bei tieferer Struktur immer detaillierter.

Eine netzartige Struktur versucht, mit untereinander stark verlinkten Internetseiten die Assoziationen des Benutzers zu unterstützen. Da hier keine geordnete Struktur vorliegt, kann ein netzartiger Aufbau bei umfangreichen Internetangeboten den Benutzer verwirren.

Nach Nielsen weisen die meisten Internetauftritte eine hierarchische Struktur auf [vgl. Nie00, S.198]. Da Benutzer mit hierarchischen Darstellungen vertraut sind, fällt es ihnen leicht, diese Struktur zu verstehen. Oft liegen Mischformen mit anderen Strukturen vor. So werden auch in hierarchisch aufgebauten Internetauftritten die Informationen in detaillierteren Ebenen durch Querverweise netzartig verknüpft und beispielsweise Bestellvorgänge oft in linearer Form Schritt für Schritt abgearbeitet. Die Navigation stellt zu jeder Zeit dar, wo sich der Benutzer in der Hierarchie befindet und welche Bewegungsmöglichkeiten ihm offen stehen. Um dem Benutzer das Verstehen der Seitenstruktur zu erleichtern, verdeutlicht die Navigation die Hierarchieebenen durch Größenverhältnisse oder Texteinrückungen.

## 5.2.5 Konventionen im Internet

Wie bereits erläutert ermöglicht es ein gebrauchstauglicher Internetauftritt dem Benutzer, seine Ziele zu erreichen. Führt ein Auftritt den Benutzer nicht zu seinem Ziel, verlässt er diesen und wendet seine Aufmerksamkeit einem anderen Internetauftritt zu [vgl. Pus01, S.48]. Um dem Benutzer das Gefühl zu geben, seine Ziele erreichen zu können, muss ihm ein Internetangebot ermöglichen, zielgerichtet durch die einzelnen Seiten des Angebots zu navigieren. Eine solche Navigation erfordert, dass sich der Benutzer in dem Internetangebot orientieren kann. Deshalb muss der Benutzer zu jeder Zeit wissen, an welcher Stelle er sich befindet und welche weiteren Möglichkeiten ihm offen stehen.

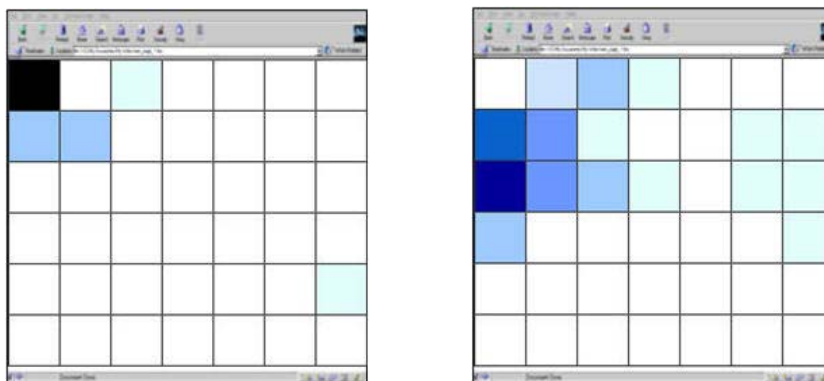
Der Standpunkt des Benutzers bezieht sich hier nicht nur auf seine Position in dem Internetangebot an sich, sondern auch auf dessen Position im Internet. Da Internetangebote häufig nicht über die Startseite aufgerufen werden, sondern durch Verweise aus Suchmaschinen heraus direkt Unterseiten angezeigt

werden, kann es für den Benutzer schwierig sein zu erkennen, in welchem Internetauftritt er sich befindet.

Um dem Benutzer die zur Orientierung benötigte Sicherheit bezüglich des eigenen Standpunktes im Internet zu geben ist sicherzustellen, dass jede Seite eines Internetangebotes aufzeigt, zu welchem Internetangebot sie gehört und welche Navigationsmöglichkeiten dem Benutzer offen stehen.

Damit die angebotenen Hilfsinformationen für den Benutzer möglichst leicht auffindbar sind, sollten bei der Seitengestaltung übliche Konventionen beachtet werden. Diese ermöglichen dem Benutzer eine schnelle Orientierung auf einer neuen Internetseite, da er vorhandene Erfahrungen auf neue Gegebenheiten anwenden kann.

In einer Studie untersuchten Michael Bernard und Ashwin Sheshadri diese Konventionen bezüglich verschiedener Elemente eines Internetangebots zum Onlineeinkauf. Die Teilnehmer sollten vorgefertigte Elemente in einem leeren mit Zeilen und Spalten versehenen Browserfenster platzieren [vgl. BS]. Zu den Elementen zählten unter anderem ein Verweis zu Startseite, Werbeflächen, Einkaufskorb sowie Verweise zu internen und externen Internetseiten.



**Abbildung 5.2:** Erwartungen bezüglich des Verweises zur Startseite (links) und bezüglich interner Verweise (rechts) [vgl. BS]

Es ist davon auszugehen, dass diese Erkenntnisse nicht nur auf Angebote zum Onlineeinkauf zutreffen, sondern darüber hinaus auch bei anderen Internetangeboten Anwendung finden. Zu beachten sind hier besonders die Erwartungen

bezüglich der Verweise zu weiteren Internetseiten des Angebots, der Navigation sowie dem Verweis zur Startseite des Angebots. Wie in Abbildung 5.2 durch die farbig markierten Zellen zu sehen, wird der Verweis zur Startseite in der oberen linken Ecke des Browserfensters, die Navigation direkt darunter am linken Bildschirmrand erwartet.

### 5.2.6 Unterstützung des menschlichen Gehirns

Nicht nur erlernte Konventionen schränken die Gestaltungsfreiheit bei dem Entwurf einer Internetseite ein, auch verschiedene Gegebenheiten des menschlichen Körpers sind bei Gestaltungsentscheidungen zu berücksichtigen.

Die Konvention, dass sich die Hauptnavigation am linken Bildschirmrand befindet, hat sich nicht unbegründet gebildet. Optische Reize werden mit den Augen aufgenommen, in Impulse umgewandelt und über Nervenbahnen zum Gehirn geleitet. Die Nervenbahnen, die von den Augen zum Gehirn führen, kreuzen sich, sodass die durch das linke Auge aufgenommenen Reize in der rechten Gehirnhälfte ankommen und die durch das rechte Auge aufgenommenen Reize entsprechend die linke Gehirnhälfte zuerst erreichen. Die Gehirnhälften unterscheiden sich durch die Aufgaben, die sie ausführen. Die rechte Hälfte zeichnet sich dadurch aus, dass sie ganzheitliche Strukturen nonverbal verarbeitet, während die linke Hälfte eher verbal und analytisch vorgeht. Werden Inhalte auf einer Internetseite so platziert, dass die Informationen direkt in die zuständige Gehirnhälfte geleitet werden, entfällt ein weiteres Austauschen der Informationen zwischen den Gehirnhälften und eine schnellere Informationsverarbeitung wird unterstützt [vgl. WvG02, S.34]. Für die Seitengestaltung bedeutet dies, dass die Navigationsstruktur mit nonverbaler Unterstützung am linken Bildschirmrand und textuelle Informationen auf der rechten Bildschirmseite angeordnet sein sollten.

Nicht nur die Anordnung der Navigation auf dem Bildschirm sollte den menschlichen Gegebenheiten angepasst sein. Auch die Navigationsstruktur hat sich am menschlichen Gedächtnis zu orientieren. Das Kurzzeitgedächtnis ist dafür verantwortlich, Einzelheiten aufzunehmen und diese zu behalten, bis die

Verarbeitung abgeschlossen ist und ein Gesamtzusammenhang entsteht [vgl. WvG02, S.39]. Die Kapazität des Kurzzeitgedächtnisses variiert, durchschnittlich sind sieben Objekte gleichzeitig verarbeitbar. Beim Überschreiten der Maximalanzahl werden Objekte ausgetauscht, und der Gesamtüberblick kann verloren gehen. Um die Navigationsstruktur zu jeder Zeit übersichtlich zu gestalten, sollten sich auf einer Hierarchieebene nie mehr als fünf bis sieben Navigationsmöglichkeiten befinden.

## 6 Werkzeuge zur Erstellung von Internetangeboten

Im letzten Kapitel wurde herausgestellt, wie die Anforderungen von behinderten und nichtbehinderten Menschen an Internetauftritte lauten. Das Ziel der Diplomarbeit besteht in der Entwicklung eines Werkzeugs zur Erstellung von Internetangeboten. Daher ist vorerst zu klären, welche Unterstützung Werkzeuge bei der Erstellung von Internetangeboten bieten. Dieses Kapitel zeigt, welche Arten von Werkzeugen es gibt.

Für den Benutzer bestehen die zu betrachtenden Internetauftritte aus einzelnen (X)HTML-Seiten, die durch Verweise miteinander verbunden sind. Unabhängig von diesem Erscheinungsbild gibt es die Möglichkeit, die Internetinhalte nicht statisch in (X)HTML-Dateien zu speichern, sondern eine andere Form der Datenhaltung zu benutzen und die Inhalte dynamisch in (X)HTML-Seiten einzufügen. Abhängig davon sind die Möglichkeiten, diese Seiten herzustellen und zu bearbeiten, sehr vielfältig.

### 6.1 (X)HTML-Editor

Internetangebote, deren Inhalte statisch als (X)HTML-Seiten gespeichert sind, können auf unterschiedliche Art erstellt werden. Da es sich bei (X)HTML-Sei-

ten um Textdateien handelt, besteht die Möglichkeit, Internetseiten durch reine Texteingabe zu erstellen. Darüber hinaus ermöglichen (X)HTML-Editoren unterschiedliche Grade der Automatisierung verschiedener Arbeitsschritte. Zu unterscheiden sind hier zwei Arten von (X)HTML-Editoren, textbasierte Editoren sowie grafische Editoren.

Bei der Verwendung eines Texteditors zur Erstellung von (X)HTML-Seiten erstellt und bearbeitet der Benutzer unmittelbar den Quelltext des Dokuments. Texteditoren können den Benutzer bei der Erstellung von (X)HTML-Seiten auf unterschiedliche Art und in unterschiedlicher Intensität unterstützen. Um dem Benutzer die Orientierung im Quelltext zu erleichtern, bieten viele Editoren die farbliche Hervorhebung der Quelltextsyntax an. Hier werden vorhandene Auszeichnungen, *Tags*, in unterschiedlichen Farben dargestellt. Darüber hinaus können solche Editoren den Zeitaufwand beim Erstellen von (X)HTML-Seiten erheblich minimieren, indem sie *Tags* in vorgefertigter Form zum Einfügen bereitstellen. Unabhängig davon, welche Unterstützung der Texteditor bietet, muss der Benutzer neben (X)HTML-Kenntnissen auch über das Abstraktionsvermögen verfügen, aus dem logisch-analytisch aufgebauten Quelltext eine Seitenansicht zu konstruieren [vgl. Lan01].

Genau dieses Abstraktionsvermögen sowie genaue Kenntnisse der Auszeichnungssprache (X)HTML werden bei dem Einsatz eines (X)HTML-Editors mit grafischer Oberfläche nicht mehr benötigt. Die sogenannten „*What-you-see-is-what-you-get*“-Editoren (WYSIWYG) ermöglichen es dem Benutzer, die grafischen Elemente der (X)HTML-Seite auf dem Bildschirm ähnlich wie bei der Verwendung eines normalen Grafikprogramms anzuordnen. Die Umsetzung in (X)HTML-Quelltext wird von dem Editor übernommen. Der erzeugte Quelltext weist jedoch häufig Unreinheiten wie die Verwendung redundanter oder proprietärer *Tags* auf [vgl. Nie99, S.69]. Da aus diesem Grund der erzeugte Quelltext manueller Nachbearbeitung bedarf, sind auch diese Editoren nicht ohne grundlegende (X)HTML-Kenntnisse verwendbar [vgl. Lan01, S.185].

Über die Eigenschaft des WYSIWYG hinaus bieten professionelle (X)HTML-Editoren oft Funktionalitäten, ohne die sich komplexere Internetangebote kaum realisieren und pflegen lassen. Hierzu gehören beispielsweise die automatische Anpassung von Verweisen beim Verschieben von einzelnen (X)HTML-Seiten innerhalb der Ordnerstruktur oder die Verwaltung der Navigation [vgl. Lan01, S.184].

Beide Arten von (X)HTML-Editoren bieten dem Benutzer mehr oder weniger eine Arbeitserleichterung beim Erstellen von Internetangeboten. Jedoch führt die Benutzung eines Editors nur mit fundierten (X)HTML-Kenntnissen sowie Kenntnissen in der grafischen Gestaltung von Internetseiten zu einem annehmbaren Ergebnis. Um auch Benutzern ohne (X)HTML-Kenntnisse die Möglichkeit zur Erstellung von Internetangeboten zu geben, sind neben den Automatisierungen, die bereits ein WYSIWYG-Editor bietet, weitere Automatisierungen durch die Anwendung nötig. Zu betrachten sind hier verschiedene Umsetzungen eines *Content Management Systems*.

## 6.2 Content Management System

*Content Management Systeme* bieten dem Benutzer je nach Systemgröße sehr umfangreiche Funktionalitäten. Im Rahmen dieser Funktionalitäten kann auch eine automatisierte Gestaltung der einzelnen Internetseiten sowie der Struktur des gesamten Angebots durch das System übernommen werden. Da im Rahmen der Diplomarbeit ein Werkzeug entsteht, welches sehbehinderte und blinde Benutzer besonders unterstützt, wird nun erläutert, was ein CMS in diesem Zusammenhang bieten kann.

Zunächst gilt es zu definieren, was unter einem CMS zu verstehen ist. Danach werden verschiedene Umsetzungsmöglichkeiten betrachtet. Nach Gunther Rothfuss und Christian Ried beschäftigt sich *Content Management* mit der „*systematischen Sammlung und Verwaltung von Informationsbausteinen zu einem einzigen (logischen) Bestand*“ [RR01, S.59]. Ein CMS umfasst nach Rothfuss/Ried folgende Funktionen [vgl. RR01, S.73 ff.]:



- **Benutzerverwaltung:** Abhängig von den ihm zugeteilten Rechten kann ein Benutzer nur die für ihn vorgesehenen Aktionen ausführen.
- **Protokollfunktionen:** Änderungen im Datenbestand werden erfasst und protokolliert.
- **Datensicherung und *Rollback*:** Aufbauend auf den Protokollfunktionen und einer Datensicherung können Änderungen im Datenbestand rückgängig gemacht werden. Der Datenbestand befindet sich nach einem *Rollback* in einem konsistenten Zustand.
- **Mehrplatzfähigkeit:** Der Datenbestand lässt sich von mehreren Benutzer zur gleichen Zeit bearbeiten. Um dies zu ermöglichen, muss das System sicherstellen, dass die Änderungen nicht voneinander abhängig sind. Mehrplatzfähigkeit kann beispielsweise durch *Check-In* und *Check-Out* umgesetzt werden.
- ***Check-In* und *Check-Out*:** Möchte ein Benutzer bestimmte Inhalte ändern, muss er diese für sich reservieren (*Check-Out*). Reservierte Inhalte sind bis zu einer Freigabe (*Check-In*) von keinem anderen Benutzer benutzbar.
- **Verwaltungsfunktionen:** Aufbauend auf vor- und benutzerdefinierten Metainformationen der Inhalte ist eine Verwaltung des Datenbestands möglich.
- **Anfragefunktionen:** Das CMS stellt Möglichkeiten bereit, Inhalte abhängig von unterschiedlichen Eigenschaften, wie enthaltene Zeichenketten oder vorhandene Dateigrößen, zur Bearbeitung auszuwählen.
- **Massenoperationen:** Bei einem umfangreichen Datenbestand können durch eine Aktion mehrere Inhalte gleichzeitig bearbeitet werden.

Systeme, die auf eine Verarbeitung von Inhalten für das Internet spezialisiert sind, heißen auch *Web Content Management Systeme* (WCMS) [vgl. RR01, S.61]. Der Leistungsumfang von verschiedenen Produkten dieser Sparte ist sehr unterschiedlich, oft werden nicht alle der oben genannten Funktionen unterstützt. Generell ist davon auszugehen, dass ein WCMS den *Content Life Cycle* von Inhalten weitgehend automatisiert [vgl. BTZZ00, S.99]. Der *Content Life Cycle* umfasst Erstellung, Kontrolle, Freigabe, Veröffentlichung und schließlich die Archivierung oder das Löschen von Internetinhalten [vgl. BTZZ00, S.83 ff.]. Durch diese Automatisierung ermöglicht ein WCMS das Erstellen, Ändern und

die Pflege von Internetinhalten durch Benutzer, die technisch weniger versiert sind.

Werden Internetangebote durch ein WCMS erstellt, besteht das Angebot oft nicht aus statischen (X)HTML-Seiten. Statt dessen ist die Datenhaltung von der Darstellung getrennt, und die Daten werden erst beim Aufruf einer Internetseite dynamisch in bestehende (X)HTML-Vorlagen eingefügt. Um zu untersuchen, welche Art der Datenhaltung für ein WCMS benutzt werden kann, befasst sich das nächste Kapitel mit verschiedenen Möglichkeiten der Datenhaltung und ihren Vorteilen.

## 7 Datenhaltung

Die Datenhaltung für dynamisch erzeugte Internetangebote lässt sich auf unterschiedliche Art und Weise umsetzen. In diesem Kapitel gilt es verschiedene Umsetzungsmöglichkeiten sowie deren Vor- und Nachteile zu untersuchen.

Die Inhalte von Internetangeboten entsprechen bereits einer (X)HTML-Struktur. Das heißt, in einem CMS sind Daten hierarchischer Art zu speichern. Da die Daten im Falle von XHTML bereits der XML-Struktur entsprechen, liegt es nahe, sich diese Auszeichnungssprache als Form der Datenhaltung genauer anzusehen.

### 7.1 XML

Die *eXtensible Markup Language* (XML) wird seit 1996 durch das *World Wide Web Consortium* (W3C) [vgl. XML] entwickelt. Trotz der erst kurzen Zeit seit Beginn der Entwicklung wird XML bereits in einem breiten Spektrum an Anwendungen in der Informations- und Kommunikationstechnik verwendet.

XML bietet als Metasprache Regeln für die Erzeugung von Auszeichnungssprachen [vgl. Arc01, S.28] und wird zum Datenaustausch, zur Datenpräsentation sowie zur Datenhaltung genutzt. Dieses Kapitel bietet einen kurzen Überblick über die Eigenschaften und Möglichkeiten von XML.

XML wurde 1996 von einer Arbeitsgruppe des W3C entwickelt. 1998 wurde die Sprache erstmals in der Version 1.0 als Empfehlung (*recommendation*) verabschiedet [vgl. And00, S.35]. XML entstand als echte Teilmenge der *Standard Generalized Markup Language* (SGML), einer Metasprache zur Definition weiterer Auszeichnungssprachen besonders im Bereich der Dokumentenverarbeitung. Bei SGML handelt es sich um eine mächtige aber komplizierte Auszeichnungssprache. Diese Komplexität erschwert das Verständnis sowie eine performante Verarbeitung. Als eines der Ziele bei der Entwicklung von XML kann daher die Vereinfachung von SGML bei Beibehaltung der Ausdrucksmöglichkeiten angesehen werden.

Über die Eigenschaft der Metaauszeichnungssprache hinaus gewinnt XML zunehmend auch in den Bereichen Datenhaltung und Datenaustausch an Bedeutung. Gründe für diese Verwendung der Sprache sind vor allem in der Plattformunabhängigkeit, der selbstbeschreibenden Eigenschaft sowie der Möglichkeit, hierarchische Informationen abzubilden, zu sehen [vgl. And00, S.170]. Um die Speicherung von Daten in XML darzustellen, zeigt das nächste Unterkapitel den Aufbau von XML-Dokumenten.

### 7.1.1 XML-Dokumente

Ein XML-Dokument ist in drei Teile gegliedert: den *Prolog*, den *Rumpf* und den *Epilog*.

Der *Prolog* besteht im Wesentlichen aus einer XML-Deklaration. Diese Deklaration weist ein Dokument als XML-Dokument aus. Sie legt einerseits durch das Attribut *version* die Versionsnummer des XML-Standards fest, andererseits durch das Attribut *encoding* die zu verwendenden Schriftzeichen. Zusätzlich

kann der *Prolog* eine Strukturbeschreibung des Dokuments enthalten. Möglichkeiten zur Strukturbeschreibung sind im Anschluss an diesen Abschnitt erläutert. Der *Prolog* ist als optionaler, jedoch empfohlener Teil eines XML-Dokumentes anzusehen [vgl. And00, S.58].

Der *Rumpf* bildet den eigentlichen Dokumenteninhalt. Er besteht aus *Markup* und Text. Durch das *Markup* werden dem enthaltenen Text Struktur und Bedeutung zugeordnet. Hierzu wird der Text in Elementen gekapselt. Diese werden durch öffnende und schließende *Tags* gekennzeichnet und können neben dem Text auch weitere Elemente enthalten. Alle Elemente sind baumartig verschachtelt in einem Wurzelement enthalten.

Der *Epilog* ist wie der *Prolog* als optionaler Teil zu betrachten. Jedoch ist von der Verwendung eines *Epilogs* abzusehen [vgl. Arc01, S.32], da er lediglich eine unstrukturierte Sammlung verschiedener Kommentare und Verarbeitungsanweisungen enthält.

Das folgende *Listing* zeigt anhand eines praktischen Beispiels den formalen Aufbau eines XML-Dokuments. Es besteht aus der XML-Deklaration sowie dem *Rumpf*. Auf einen *Epilog* wurde verzichtet.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<project counter="31">
  <css>grau</css>
  <imprint>
    <author>
      <name>Bonnert</name>
      <firstname>Daniela</firstname>
      <street>Bouchestrasse</street>
      <zip>12435</zip>
      <city>Berlin</city>
      <state>Deutschland</state>
    </author>
    <title>Seitentitel</title>
    <subtitle>Seitenuntertitel</subtitle>
  </imprint>
  <description>Beschreibung</description>
  <keywords>ein paar treffende Worte</keywords>
</project>
```

Listing 7.1: XML-Dokument

Entspricht ein XML-Dokument neben diesem formalen Aufbau auch den syntaktischen Regeln der XML-Syntax, nennt man das Dokument *wohlgeformt* (*well-formed*) [vgl. Arc01, S.45]. Im Folgenden werden einige Bedingungen, welchen ein *wohlgeformtes* XML-Dokument entsprechen muss, dargestellt:

- **XML-Deklaration:** Weist ein Dokument als XML-Dokument aus.
- **Genau ein Wurzelement:** Auf der obersten Ebene im Dokumentenrumpf enthält das Dokument genau ein Element. Dieses Element hat keine Vorgänger oder Geschwister und alle weiteren Elemente sind Nachfolger dieses Elements.
- **Korrekte Elemente:** Jedem *Starttag*, mit „< >“ gekennzeichnet, ist ein entsprechendes *Endtag* zugeordnet, mit „</ >“ gekennzeichnet. Ausnahmen hierzu bilden lediglich leere Elemente. Leere Elemente enthalten keinen weiteren Inhalt und sind mit „< />“ zu begrenzen.
- **Korrekte Schachtelung:** Elemente dürfen nicht kreuzweise überlappen. Das bedeutet, dass jeweils das Start- und Endtag eines Elements im selben Vorgängerelement enthalten sind.
- **Attributwerte in Anführungszeichen:** Die Attributwerte in einem Starttag müssen jeweils in Anführungszeichen gekapselt werden.
- **Attributname eindeutig:** In einem Element darf ein Attributname maximal einmal auftreten.

Zusätzlich zur Wohlgeformtheit kann ein Dokument auch weiteren Strukturbeschreibungen entsprechen. Entspricht ein XML-Dokument zusätzlich festgelegten Strukturbeschreibungen, nennt man das Dokument *gültig* (*valid*). Strukturbeschreibungen legen die genaue Verwendung von Elementen, Attributen, Entitäten, Notationen und Datentypen in einer Gruppe von XML-Dokumenten fest [vgl. And00, S.148].

Was sind die Vorteile solcher Strukturbeschreibungen? Ist die Dokumentenstruktur durch eine Beschreibung festgelegt, so ist diese bereits bekannt, bevor die entsprechenden XML-Dokumente durch eine Anwendung bearbeitet werden. Diese Tatsache bietet mehrere Vorteile. Anwendungen, die mit XML-Dokumenten arbeiten, gehen oft sehr genau auf die darin enthaltene Struktur

ein. Ändert sich diese Struktur, ist auch die damit arbeitende Anwendung anzupassen. Die Entwickler, deren Aufgabe darin besteht, die Anwendung zu ändern, können auf die Strukturbeschreibung als Dokumentation zurückgreifen [vgl. And00, S.90]. Zusätzlich können fehlerhafte Dokumente bereits ausgefiltert werden, bevor durch sie die Stabilität weiterer Anwendungen gefährdet wird [vgl. And00, S.90]. Um die Struktur eines XML-Dokuments zu beschreiben, haben sich zwei Möglichkeiten durchgesetzt: *Document Type Definition* (DTD) und XML-Schema. Die folgenden Unterkapitel erläutern beide Möglichkeiten und untersuchen die Unterschiede.

## 7.1.2 Document Type Definition

Eine *Document Type Definition* (DTD) legt Strukturregeln für eine Gruppe von XML-Dokumenten fest. Insbesondere beschreibt eine DTD [vgl. And00, S.91 und Arc01, S.44 ff.]

- welche Elemente erlaubt sind,
- in welcher Reihenfolge und Häufigkeit sie vorkommen,
- welche Attribute sie enthalten und
- in welcher Beziehung sie zu anderen Elementen stehen.

Die Strukturregeln einer DTD können direkt im Prolog eines XML-Dokumentes angegeben werden (interne Teilmenge, *internal subset*) oder in einer externen Datei abgespeichert sein (externe Teilmenge, *external subset*).

```
<!DOCTYPE project [ ... ]>
```

**Listing 7.2:** Interne Teilmenge, *internal subset*

```
<!DOCTYPE project SYSTEM "xml/project.dtd">
```

**Listing 7.3:** Externe Teilmenge, *external subset*

Extern in einer DTD gespeicherte Strukturregeln sind weitaus flexibler einsetzbar als interne Teilmengen [vgl. And00, S.93]. Bei der Verwendung einer externen Teilmenge ist eine Instanz an Strukturregeln von mehreren XML-Dokumenten nutzbar. Besonders bei Änderungen reduziert dies den Arbeitsaufwand erheblich. Üblich ist auch eine Mischung beider Verfahren. Hier erfolgt eine Definition allgemeiner Regeln in der externen Teilmenge, während speziellere Anforderungen durch eine interne Teilmenge ergänzt werden. Die Möglichkeit, dass externe Deklarationen durch interne überdeckt werden, besteht [vgl. And00, S.92].

In DTDs gibt es vier Deklarationsmöglichkeiten [vgl. And00, S.94]:

- `ELEMENT` wird verwendet um ein Element zu definieren.
- `ATTLIST` definiert die für ein Element möglichen Attribute sowie deren Namen und Wertemöglichkeiten.
- `ENTITY` deklariert einen beliebigen Dokumentteil, der häufiger Verwendung findet.
- `NOTATION` behandelt Inhalte, die nicht aus XML bestehen.

Das folgende Listing mit der DTD zu dem in Listing 7.1 dargestellten XML-Dokument zeigt die Verwendbarkeit dieser Deklarationsmöglichkeiten in einer internen DTD. Anschließend wird erläutert, wie Elemente und Attribute in einer DTD definiert werden.

```
<!DOCTYPE project [  
<!ELEMENT project (css,imprint,description,keywords)>  
<!ELEMENT css (#PCDATA)>  
<!ELEMENT imprint (author,title,subtitle)>  
<!ELEMENT author (name,firstname,street,zip,city,state)>  
<!ELEMENT name (#PCDATA)>  
<!ELEMENT firstname (#PCDATA)>  
<!ELEMENT street (#PCDATA)>  
<!ELEMENT zip (#PCDATA)>  
<!ELEMENT city (#PCDATA)>  
<!ELEMENT state (#PCDATA)>  
<!ELEMENT title (#PCDATA)>  
<!ELEMENT subtitle (#PCDATA)>  
<!ELEMENT description (#PCDATA)>  
<!ELEMENT keywords (#PCDATA)>  
<!ATTLIST project counter CDATA #IMPLIED]>
```

**Listing 7.4:** Interne DTD

In einer DTD sind Elemente durch ein `ELEMENT-Tag` definiert. Angegeben werden der Name des Elementes und dessen Inhalt:

```
<!ELEMENT elementname (content model)>
```

**Listing 7.5:** DTD Elementdefinition

In DTDs stehen drei Elementtypen zur Verfügung:

- **Leere Elemente** enthalten weder Text noch andere Elemente. Als Schlüsselwort findet `EMPTY` Verwendung.
- **Elemente mit beliebigem Inhalt** spezifizieren ihren Inhalt nicht genauer. Gebraucht wird hier das Schlüsselwort `ANY`.
- Elemente, die andere Elemente in bestimmter Form enthalten dürfen, werden mit dem Begriff **Inhaltsmodell** (*content model*) beschrieben.

Inhaltsmodelle können aus den Namen der möglichen Kindelemente sowie dem Schlüsselwort `#PCDATA`, um Texte zu spezifizieren, bestehen. Die einzelnen Elemente werden mit Klammern und Operatoren kombiniert. Operatoren geben hierbei die Kardinalität und die Verwendung der Elemente an. Möglich ist der Einsatz folgender Operatoren [vgl. And00, S.99]:

- „*a,b*“ bildet eine Sequenz der Elemente *a* und *b*.
- „*a|b*“ trennt optionalen Elemente *a* und *b*.
- „*a?*“ Element *a* ist optional und kann einmal oder keinmal auftreten.
- „*a\**“ Element *a* kann keinmal oder beliebig oft auftreten.
- „*a+*“ Element *a* tritt beliebig oft, aber mindestens einmal auf.

Zusätzlich zu den Elementen kann eine DTD auch die auftretenden Attribute spezifizieren:

```
<!ATTLIST elementname attributname attributtyp modus>
```

**Listing 7.6:** DTD Attributdefinition



Der Modus kann angeben, ob ein Attribut optional ist oder welchen Wert es gegebenenfalls annehmen soll. Folgende Vorgabewerte stehen in DTDs zur Verfügung [vgl. And00, S.102]:

- #REQUIRED bedeutet, dass das Attribut in jeder Instanz des Elements erforderlich ist.
- #IMPLIED weist optionale Attribute aus.
- #FIXED tritt in Verbindung mit einem Vorgabewert auf, den das Attribut annehmen muss.
- Ein *Standardwert* setzt einen Wert, falls das Attribut nicht auftritt.

Mit Hilfe des Attributtyps kann der Wert eines Attributes näher beschrieben werden. In DTDs stehen unter anderem folgende Attributtypen zur Auswahl:

- CDATA stellt Text dar.
- ID ist ein eindeutiger Bezeichner.
- IDREF stellt eine Referenz zu einem ID-Attribut dar.
- IDREFS: Aufzählung von IDREFs.
- ENTITY bedeutet eine vordefinierte Entität.
- ENTITIES: Aufzählung von Entitäten.
- [Wert aus einer Aufzählung] akzeptiert einen Wert aus einer benutzerdefinierten Aufzählung.

In den letzten Abschnitten wurde gezeigt, wie mit Hilfe von DTDs gültige Dokumente erstellt werden. DTDs sind jedoch nicht die einzige Möglichkeit Strukturbeschreibungen vorzunehmen. Wo liegen die Beschränkungen von DTDs [vgl. And00, S.256 ff.]?

- DTDs folgen nicht der XML-Syntax, sondern sind nach einer eigenen Syntax aufgebaut. Das verhindert eine Verarbeitung von DTDs durch die genutzten XML-Anwendungen.
- DTDs unterstützen Datentypen nur sehr mangelhaft. Dadurch ist es nicht möglich, Attributwerte näher zu spezifizieren.
- Die Unterstützung von Namensräumen fehlt.
- Die Erweiterbarkeit von DTDs ist auf Parameterentitäten eingeschränkt.

Um diese Beschränkungen zu beheben, wurden eine Reihe weiterer Standards entwickelt. Durchgesetzt hat sich vor allem XML-Schema.

### 7.1.3 XML-Schema

XML-Schema löst einige der Problemfelder, die bei der Arbeit mit DTDs bestehen. XML-Schemas sind in XML-Syntax aufgebaut und entsprechen selbst wohlgeformten Dokumenten. Dadurch sind die Strukturdefinitionen für Entwickler, die die XML-Syntax beherrschen, leichter verständlich. Darüber hinaus sind diese Definitionen auch für XML-Anwendungen zugänglich und können mit diesen verarbeitet werden. Durch die Unterstützung von Namensräumen ist es mit XML-Schema möglich, Strukturdefinitionen modular aufzubauen und Namenskonflikte zu vermeiden. Zusätzlich sind in XML-Schema Werte bereits in der Strukturdefinition näher spezifiziert. XML-Schema stellt zu diesem Zweck ein strenges Typenkonzept bereit.

Entwickelt wurde XML-Schema durch eine Arbeitsgruppe des W3C. Seit Mai 2001 handelt es sich um eine offizielle Empfehlung des W3C. Langfristig ist daher damit zu rechnen, dass die Verwendung von DTDs durch XML-Schema abgelöst wird [vgl. Arc01, S.311].

XML-Schema bildet eine sehr umfangreiche Möglichkeit, Strukturbeschreibungen vorzunehmen. In den folgenden Abschnitten können daher nur erste Einblicke vermittelt werden. Das folgende Listing zeigt den Aufbau eines XML-Schemas an Hand eines praktischen Beispiels:

```
<?xml version="1.0"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:element name="project" type="projectType"/>
  <xsd:complexType name="projectType">
    <xsd:sequence>
      <xsd:element name="css" type="xsd:string"/>
      <xsd:element name="imprint" type="imprintType"/>
      <xsd:element name="description" type="xsd:string"/>
      <xsd:element name="keywords" type="xsd:string"/>
    </xsd:sequence>
    <xsd:attribute name="counter" type="xsd:int"/>
  </xsd:complexType>
</xsd:schema>
```

```

</xsd:complexType>
<xsd:complexType name="imprintType">
  <xsd:sequence>
    <xsd:element name="author" type="authorType"/>
    <xsd:element name="title" type="xsd:string"/>
    <xsd:element name="subtitle" type="xsd:string"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="authorType">
  <xsd:sequence>
    <xsd:element name="name" type="xsd:string"/>
    <xsd:element name="firstname" type="xsd:string"/>
    <xsd:element name="street" type="xsd:string"/>
    <xsd:element name="zip">
      <xsd:simpleType>
        <xsd:restriction base="xsd:positiveInteger">
          <xsd:totalDigits value="5"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:element>
    <xsd:element name="city" type="xsd:string"/>
    <xsd:element name="state" type="xsd:string"/>
  </xsd:sequence>
</xsd:complexType>
</xsd:schema>

```

**Listing 7.7:** XML-Schema

In XML-Schema gibt es die Unterscheidung zwischen *einfachen Typen* und *komplexen Typen*. Die Verwendung dieser Typen wird im Folgenden aufgezeigt.

*Einfache Typen* können Zeichendaten enthalten, jedoch keine weiteren Elemente oder Attribute. Diese *einfachen Typen* sind direkt zur Definition von Elementen und Attributen benutzbar.

```

<xsd:element name="elementname" type="typ"/>
<xsd:attribute name="attributname" type="typ"/>

```

**Listing 7.8:** Definition von einfachen Typen

Das `type`-Attribut gibt bei *einfachen Typen* den Datentyp an. Die folgende Tabelle zeigt einige vordefinierte *einfache Typen* in XML-Schema [vgl. XMLS].

Zusätzlich können neue *einfache Typen* durch die Ableitung von einem bestehenden *einfachen Typen* definiert werden.

Einfacher Typ	Beispiel, Bemerkungen
string	Einfach Text!
integer	-126789, -1, 0, 1, 126789
int	-1, 126789675
long	-1, 12678967543233
short	-1, 12678
decimal	-1.23, 0, 123.4, 1000.00
float	-INF, -1E4, -0, 0, 12.78E-2, 12, INF, NaN
boolean	true, false, 1, 0
time	13:20:00.000, 13:20:00.000-05:00
date	1980-04-14
Name	XML 1.0 Name-Typ
anyURI	http://www.example.com/
ID	XML 1.0 ID Attribut-Typ
IDREF	XML 1.0 IDREF Attribut-Typ

**Tabelle 7.1:** Vordefinierte einfache Typen in XML-Schema [vgl. XMLS]

Die für *einfache Typen* erlaubten Werte lassen sich in XML-Schema an Hand sogenannter *Facetten* einschränken. Abhängig von dem verwendeten Typen sind folgende *Facetten* möglich: `length`, `minLength`, `maxLength`, `pattern`, `enumeration` und `whiteSpace`.

Die Deklaration neuer *komplexer Typen* erfolgt vor der Definition. Somit können diese anschließend, ähnlich der *einfachen Typen*, definiert werden. Die Deklaration von *komplexen Typen* erfolgt über das `complexType`-Element. In diesem Element sind eine Reihe von Element-Deklarationen, Element-Referenzen und Attribut-Deklarationen enthalten.

```
<xsd:complexType name="elementname">
  <xsd:sequence>
    <xsd:element name="Vorname" type="xsd:string"/>
    <xsd:element name="Nachname" type="xsd:string"/>
  </xsd:sequence>
  <xsd:attribute name="land" type="xsd:NMTOKEN" fixed="DE"/>
</xsd:complexType>
```

**Listing 7.9:** Deklaration eines komplexen Typen

Um das Vorkommen von Elementen in einem *komplexen Typen* zu regeln, bestehen verschiedene Möglichkeiten [vgl. XMLS]:

- **xsd:sequence:** Alle Elemente müssen in entsprechender Reihenfolge auftreten.
- **xsd:choice:** Nur eines der genannten Elemente darf auftreten.
- **xsd:all:** Alle Elemente können einmal oder gar nicht in beliebiger Reihenfolge auftreten.

Um Häufigkeitsbeschränkungen für Elemente festzulegen, existieren in XML-Schema die Attribute `minOccurs` und `maxOccurs` [vgl. XMLS]. Hierbei legt `minOccurs` fest, wie häufig das Element auftreten muss, und `maxOccurs` sagt aus, wie häufig ein Element maximal auftreten darf. Die Werte der Attribute sind numerische Werte. Bei dem Attribut `maxOccurs` kann zusätzlich der Begriff `unbounded` für unbegrenztes Auftreten des Elements verwendet werden. Attribute dürfen im Gegensatz zu Elementen lediglich einmal oder keinmal auftreten. Um die Häufigkeit von Attributen festzulegen, wird das Attribut `use` genutzt. Mögliche Werte für das `use`-Attribut sind `required` für vorgeschriebene Attribute, `optional` und `prohibited` für verbotene Attribute.

## 7.1.4 Verarbeitung von XML-Dokumenten

Um XML-Daten in einer Anwendung zu verarbeiten ist eine Möglichkeit anzubieten, um aus der Anwendung heraus strukturiert auf die in einem Dokument enthaltenen Daten zuzugreifen. Hier gibt es zwei Schnittstellen, deren Umsetzung in vielen Hochsprachen verbreitet ist: *Simple API for XML (SAX)* und *Document Object Model (DOM)*. Diese Schnittstellen bilden unterschiedliche

Ansätze der XML-Verarbeitung nach: ereignisorientiert und baumorientiert. Soll XML im Rahmen dieser Diplomarbeit als Datenhaltung verwendet werden, so ist zu untersuchen, welche Art der Verarbeitung am besten geeignet ist. Nun folgt eine genaue Erläuterung beider Verarbeitungsmöglichkeiten.

Bei einer ereignisorientierten Verarbeitung wird das XML-Dokument sequenziell eingelesen und stellt eine Abfolge von Ereignissen dar. Die Verarbeitung der Daten erfolgt hier als Reaktion auf auftretende Ereignisse. Eine Umsetzung dieses Ansatzes bildet SAX. Bei einer baumorientierten Verarbeitung wird hingegen das komplette XML-Dokument eingelesen und bildet eine hierarchische Baumstruktur, mit welcher gearbeitet werden kann. Ein Beispiel für diese Verarbeitung bildet das DOM.

### **Simple API for XML**

SAX bildet eine leichtgewichtige Möglichkeit, XML-Dokumente ereignisbasiert zu bearbeiten. Leichtgewichtig bezieht sich hierbei sowohl auf die Implementierung von SAX selbst als auch auf die Integration von SAX in Anwendungen [vgl. Jec]. Ursprünglich wurde die Schnittstelle von David Megginson und der XML-Mailingliste XML-DEV entwickelt [vgl. Arc01, S.91]. Ein erster Standard wurde 1998 verabschiedet, im Jahr 2000 folgte die zweite Version SAX2. Ursprünglich enthielt die Spezifikation lediglich generische *Java*-Schnittstellen. Inzwischen wird SAX jedoch in vielen Hochsprachen unterstützt.

Da SAX ein Dokument sequenziell einliest und nicht das komplette Dokument im Hauptspeicher hält, sind die Vorteile von SAX vor allem in einem planbaren Laufzeit- und Speicherplatzverhalten zu sehen [vgl. Jec].

### **Document Object Model**

Das DOM wird seit 1998 vom W3C entwickelt und wurde als Spezifikation verabschiedet. Es bildet eine sprach- und plattformneutrale Definition von Funktionen, die eine Navigation in (X)HTML- und XML-Dokumenten sowie deren

Manipulation ermöglichen. Entwickelt wurde das DOM in mehreren Generationen, Level 1 bis Level 3. Die Generationen sind untereinander abwärtskompatibel und steigern jeweils den Funktionsumfang. Implementierungen sind auch hier in allen üblichen Programmiersprachen vorhanden.

Da bei der Verwendung des DOMs vor der Verarbeitung das gesamte XML-Dokument in den Hauptspeicher geladen wird, entstehen beim Aufbau einer DOM-Instanz möglicherweise größere Speichermengen [vgl. Jec]. Bei umfangreichen Dokumenten kann diese Eigenschaft zu Laufzeitproblemen führen. Durch die genutzte Baumstruktur sind jedoch auch große Änderungen im Dokument oder das Erstellen ganzer Dokumente problemlos möglich.

Es wurden sowohl Aufbau der Auszeichnungssprache XML als auch Arten der Verarbeitung gezeigt. Anschließend folgt die Diskussion, für welche Anwendungen das Speichern der Daten in XML-Dateien geeignet ist und welche alternativen Speichermöglichkeiten verwendet werden können.

### 7.1.5 Persistieren von XML-Daten

Das Persistieren von XML-Daten aus einer Anwendung heraus erfolgt über das Speichern von XML-Dokumenten im Dateisystem. Die Speicherung über das Dateisystem unterliegt einigen Beschränkungen, die diese Art der Speicherung für große Datenmengen unzweckmäßig werden lassen.

Die Grenzen eines Dateisystems sind in [And00, S.420 ff.] näher spezifiziert:

- **Datenmenge:** Die Arbeit mit Dateien wird ab einer bestimmten Datenmenge schwierig. Dies bezieht sich auf die Navigation in großen Dateien sowie auf die Möglichkeit, verschiedene Teile eines Dokuments unabhängig voneinander zu bearbeiten.
- **Parallele Bearbeitung:** Es ist nicht möglich, dass mehrere Personen gleichzeitig an einem Dokument arbeiten.

- **Sicherheit:** Der Zugriff auf bestimmte Teile eines Dokumentes kann nicht kontrolliert werden.
- **Datenintegration:** Es entstehen Schwierigkeiten bei der Eingliederung externer Daten.

Für viele Anwendungsfälle kann die Arbeit mit dem Dateisystem zur Persistierung von Daten ausreichend sein. Jedoch sind vorab auch weitere Speichermöglichkeiten auf ihre Tauglichkeit hin zu untersuchen.

## 7.2 Datenbanken

Eine Datenbank ist eine Sammlung von Daten inklusive „*aller relevanten Informationen über die dargestellte Information (Metainformation)*“ [SW04, S.418]. Die enthaltenen Daten folgen bestimmten Integritätsbedingungen und Regeln und stehen dem Benutzer in nur einem Exemplar ohne Redundanz zur Verfügung [vgl. SW04, S.418]. Datenbanken können beliebig große Mengen an Daten enthalten. Zur Verwaltung einer Datenbank wird üblicherweise ein Datenbankverwaltungssystem (*Database Management System, DBMS*) genutzt.

Durch die Verwendung eines DBMS zur Datenspeicherung können die Beschränkungen bei der Verwendung eines Dateisystems aufgehoben werden [vgl. And00, S.426 ff.]:

- **Datenmenge:** Mit einem DBMS werden sehr große Informationsmengen problemlos verwaltet.
- **Parallele Bearbeitung:** Bei der Datenspeicherung mit einem DBMS ist es mehreren Benutzern möglich, parallel auf dem selben Datenbestand zu arbeiten.
- **Sicherheit:** Durch ein DBMS werden oft verschiedene Ebenen des Zugriffs umgesetzt. Es ist möglich, zwischen dem Einsehen von Datensätzen und dem Bearbeiten zu trennen.
- **Datenintegration:** Datenbanken ermöglichen es, Daten aus verschiedenen Quellen gemeinsam zu verwalten und zu nutzen.



Die theoretische Grundlage für ein Datenbanksystem, ein DBMS in Verbindung mit einer oder mehreren Datenbanken, bildet das Datenmodell. Dieses Modell gibt an, nach welchem Prinzip die Daten abgebildet werden.

Am weitesten verbreitet ist hier das *relationale Datenmodell* [vgl. SW04, S.420]. Relationale Datenbanken speichern Informationen in zweidimensionalen Tabellen, die miteinander über *Primärschlüssel* und *Fremdschlüssel* in Beziehung gebracht werden. Um mit den Daten einer relationalen Datenbank zu arbeiten, wird die standardisierte Abfragesprache *Structured Query Language* (SQL) eingesetzt. Oft finden die bekannten Datenbankverwaltungssysteme *Oracle* und *mySQL* Verwendung.

Darüber hinaus gibt es das Objektdatenmodell, welches ähnlich der objektorientierten Programmiersprachen Daten als Objekte aus zusammengehörigen Attributen auffasst [vgl. SW04, S.423]. Die Performance von Objektdatenbanken reicht selten an die von relationalen Datenbanken heran, jedoch ist die Steuerung durch objektorientierte Sprachen leichter [vgl. And00, S.427].

Ein älteres Datenmodell für Datenbanken ist das hierarchische Datenmodell. Hier werden Daten in einer Baumstruktur gespeichert. Dieses Modell gilt für heutige DBMS weitgehend als überholt, wird hier aber zur Vollständigkeit erwähnt.

# Praxis

## 8 Analyse

Die aus den Grundlagen gewonnenen Erkenntnisse werden nun zum Entwurf eines Werkzeugs verwendet, welches es blinden und sehbehinderten Menschen ermöglicht, ein Internetangebot zu erstellen. Vorerst ist durch eine Analyse festzulegen, was das Werkzeug leisten soll. Zu diesem Zweck ist die Zielsetzung des Werkzeugs zu definieren. Darauf aufbauend gilt es die Anforderungen an das Werkzeug zu ermitteln und zu beschreiben.

### 8.1 Zielsetzung

Es soll ein Werkzeug entwickelt werden, welches blinde und sehbehinderte Benutzer bei der Erstellung eines kompletten Internetangebotes unterstützt. Das Werkzeug stellt alle Funktionen bereit, die im Zusammenhang mit dem Erstellen und Bearbeiten eines Internetangebots entstehen. Hierzu gehören:

- Das Anlegen des Angebots.
- Das Anlegen neuer Internetseiten.
- Das Bearbeiten der Inhalte vorhandener Internetseiten.
- Das Löschen vorhandener Internetseiten
- Die Erstellung der Struktur des Internetangebots.

Das Werkzeug muss, um sehbehinderte und blinde Benutzer ohne technische Kenntnisse ausreichend zu unterstützen, darüber hinaus folgende Funktionen automatisiert übernehmen:

- Die optische Gestaltung der einzelnen Internetseiten.
- Die Umsetzung der Inhalte in barrierefreien Techniken.

## 8.2 Anwendungsfälle des Werkzeugs

Aus der festgelegten Zielsetzung lassen sich die Anwendungsfälle des Werkzeugs ableiten. Die folgende Abbildung zeigt die entstehenden Anwendungsfälle in Notation der *Unified Modeling Language* (UML).

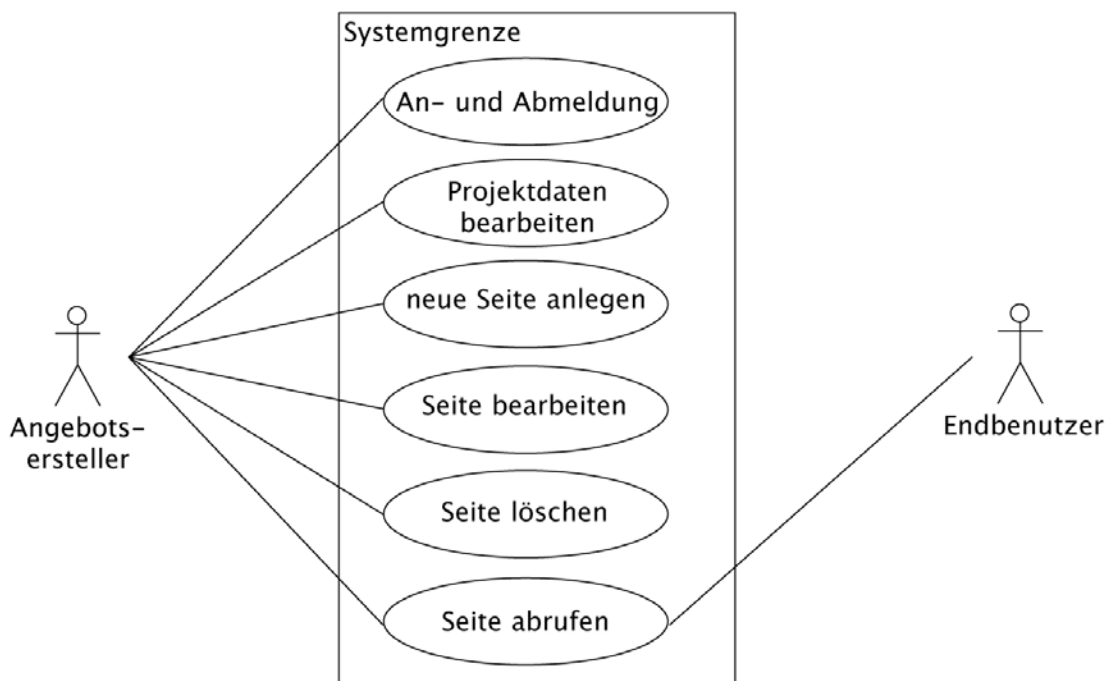


Abbildung 8.1: UML-Anwendungsfalldiagramm

Anwendungsfälle beschreiben das gewünschte externe Verhalten einer Anwendung. Durch die folgende Beschreibung der Anwendungsfälle lassen sich die Anforderungen an das Werkzeug aus Sicht des Benutzers ermitteln. Die Anforderungen werden im Anschluss beschrieben.

**Anwendungsfall:** An- und Abmeldung

**Ziel:** Eine Person authentifiziert sich beim System; Eine Person meldet sich beim System ab.

**Vorbedingung:** Keine; Eine Person ist beim System angemeldet.

**Nachbedingung Erfolg:** Person ist beim System angemeldet und kann den vollen Funktionsumfang benutzen; Person ist beim System abgemeldet und weitere Funktionen sind nicht ausführbar.

**Akteur:** Angebotsersteller

**Auslösendes Ereignis:** Betätigen des Buttons „Anmeldung“; Betätigen des Buttons „Abmeldung“.

**Beschreibung:** Eine Person gibt ihren Benutzernamen und Passwort in ein Formular ein und bestätigt die Eingabe; Eine Person wählt die Option „Abmeldung“ in der Navigation aus.

**Anwendungsfall:** Projektdaten bearbeiten

**Ziel:** Die Projektdaten werden aktualisiert.

**Vorbedingung:** Eine Person ist beim System angemeldet.

**Nachbedingung Erfolg:** Die aktualisierten Projektdaten sind gespeichert.

**Akteur:** Angebotsersteller

**Auslösendes Ereignis:** Auswahl der Option „Projekt bearbeiten“ in der Navigation.

**Beschreibung:** Eine Person gibt die projektbezogenen Daten in ein Formular ein und bestätigt die Eingabe.

**Anwendungsfall:** Neue Seite anlegen

**Ziel:** Eine neue Einzelseite wird dem Internetangebot hinzugefügt.

**Vorbedingung:** Eine Person ist beim System angemeldet.

**Nachbedingung Erfolg:** Die Daten der neuen Seite sind gespeichert. Die neue Seite erscheint in der Navigation und die Seiteninhalte können bearbeitet werden.

**Akteur:** Angebotsersteller

**Auslösendes Ereignis:** Auswahl der Option „neue Seite anlegen“ in der Navigation.

**Beschreibung:** Eine Person gibt die seitenbezogenen Daten in ein Formular ein und bestätigt die Eingabe.

**Anwendungsfall:** Seite bearbeiten

**Ziel:** Die Inhalte der Seite werden aktualisiert.

**Vorbedingung:** Eine Person ist beim System angemeldet, die Seite ist angelegt.

**Nachbedingung Erfolg:** Die Inhalte der Seite sind gespeichert.

**Akteur:** Angebotsersteller

**Auslösendes Ereignis:** Auswahl der Option „*Seiten bearbeiten*“ sowie der zu ändernden Seite in der Navigation.

**Beschreibung:** Eine Person ändert die seitenbezogenen Daten in einem Formular und bestätigt die Eingabe; Eine Person ändert die Elemente des Seiteninhaltes in ein Formular und bestätigt die Eingabe.

**Anwendungsfall:** Seite löschen

**Ziel:** Eine Einzelseite wird aus dem Internetangebot gelöscht.

**Vorbedingung:** Eine Person ist beim System angemeldet, die Seite ist angelegt.

**Nachbedingung Erfolg:** Die Daten der Seite sind gelöscht. Die Seite erscheint nicht mehr in der Navigation.

**Akteur:** Angebotsersteller

**Auslösendes Ereignis:** Betätigen des Buttons „*Seite löschen*“.

**Beschreibung:** Eine Person betätigt den Button „*Seite löschen*“ im Dialog zum Bearbeiten einer Seite.

**Anwendungsfall:** Seite abrufen

**Ziel:** Eine Einzelseite wird in der Anwenderansicht angezeigt.

**Vorbedingung:** Die Seite ist angelegt.

**Nachbedingung Erfolg:** Die Daten der Seite werden in der Anwenderansicht angezeigt.

**Akteur:** Angebotsersteller; Endbenutzer

**Auslösendes Ereignis:** Auswahl einer Seite in der Navigation.

**Beschreibung:** Eine Person wählt eine Seite in der Navigation aus und lässt sich die Inhalte der Seite anzeigen.

Den umfangreichsten Anwendungsfall stellt der Anwendungsfall „*Seite bearbeiten*“ dar. Aus diesem Grund stellt das folgende Aktivitätsdiagramm den Ablauf dieses Anwendungsfalls in UML-Syntax dar.

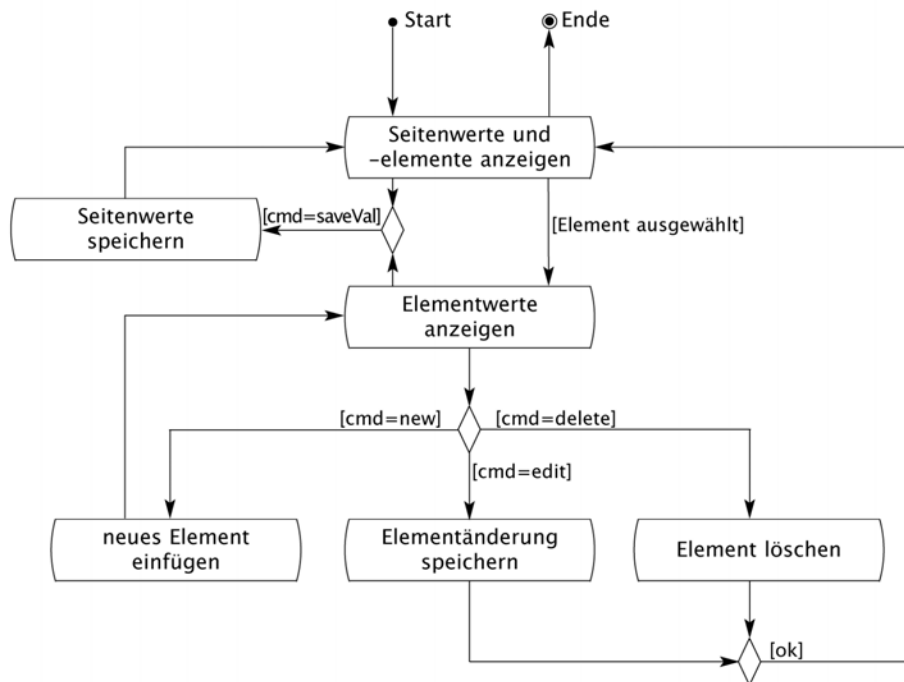


Abbildung 8.2: Standardablauf "Seite bearbeiten" als UML-Aktivitätsdiagramm

## 8.3 Anforderungen an das Werkzeug

Nicht alle Anforderungen lassen sich durch die herausgearbeiteten Anwendungsfälle beschreiben. Über die anwendungsspezifischen Anforderungen hinaus sollen auch allgemein gültige Anforderungen an Computeranwendungen erfüllt werden. Im Folgenden werden die Anforderungen an das Werkzeug aufgezeigt.

### An- und Abmeldung

Sofern das Werkzeug als Internetanwendung umgesetzt wird und dadurch öffentlich zugänglich ist, ist die Ausführung des Werkzeugs durch die Eingabe von Benutzername und Passwort zu schützen.

### **Aufnahme von Projektdaten**

Die Projektdaten sind sowohl in den Metadaten der Einzelseiten des Internetangebotes als auch bei der Erstellung von Impressum und Inhaltsübersicht zu verwenden. Zu speichern sind Name und Adresse des Anbieterstellers sowie Schlüsselwörter, Beschreibung, Titel und Untertitel, um das Angebot näher zu spezifizieren.

### **Aufnahme von Seitendaten**

Das Werkzeug muss das Anlegen, Ändern sowie das Löschen einzelner Internetseiten ermöglichen. Die Seiteninhalte können hierbei aus Überschriften, Texten, Bildern und Verweisen bestehen. Um die Barrierefreiheit der entstehenden Seiten zu ermöglichen, ist die Auszeichnung von Sprachwechseln sowie die Auszeichnung von Bildern durch Alternativtexte umzusetzen.

### **Optische Gestaltung des Internetangebots**

Die Gestaltung der Internetangebote ist automatisiert durch das Werkzeug umzusetzen. Dem Benutzer ist die Auswahlmöglichkeit zwischen verschiedenen Gestaltungsvarianten zu geben.

### **Trennung in Anwenderansicht und Entwickleransicht**

Die Anwenderansicht zeigt das erstellte Internetangebot. Diese Ansicht erlaubt dem Benutzer lediglich eine Anzeige der Seiteninhalte. Getrennt von dieser Ansicht ist die Entwickleransicht umzusetzen. Diese ermöglicht das Bearbeiten des Internetangebotes. Die Ansichten sind über unterschiedliche URLs (*Uniform Resource Locator*) zu erreichen.

### **Barrierefreiheit**

Barrierefrei bedeutet, dass eine Anwendung den Anforderungen der BITV entspricht. Im Rahmen dieser Diplomarbeit wird Barrierefreiheit nach BITV Priori-

tät II sowohl für das Werkzeug als auch für die damit erstellbaren Internetangebote gefordert. Darüber hinaus ist der Ausrichtung des Werkzeugs auf die Benutzung durch sehbehinderte und blinde Benutzer Rechnung zu tragen. Hier sind einerseits spezielle Orientierungshilfen einzufügen. Andererseits sind durch Hilfetexte und Überprüfungen Fehleingaben des Benutzers, die zu Darstellungsfehlern führen können, zu vermeiden.

### Plattformunabhängigkeit

Ist eine Anwendung plattformunabhängig bedeutet das, dass die Anwendung nicht auf die Ausführung unter einem bestimmten Betriebssystem, einer bestimmten Architektur oder auf Verwendung eines bestimmten Prozessors beschränkt bleibt, sondern davon unabhängig in Verbindung mit verschiedenen Komponenten lauffähig ist. Plattformunabhängigkeit wird erreicht, indem eine Anwendung ausschließlich durch plattformunabhängige Technologien umgesetzt wird. Plattformunabhängigkeit wird bezüglich des zu entwickelnden Werkzeugs sowohl für die entstehenden Internetangebote als auch für das Werkzeug selbst gefordert.

### Trennung von Präsentation, Inhalt und Logik

Als eine Basisanforderung bei der Entwicklung einer Computeranwendung mit grafischer Benutzeroberfläche (*Graphical User Interface*, GUI) kann die Trennung in mehrere Zuständigkeitsbereiche angesehen werden. Weit verbreitet ist das Entwurfsmuster *Model-View-Controller* (MVC) nach dem eine GUI-Anwendung in drei Schichten aufzuteilen ist:

- **Model:** Datenhaltung, Verwaltung der persistenten Anwendungsdaten.
- **View:** Präsentation, Visualisierung für den Benutzer.
- **Controller:** Geschäftslogik, Bearbeitung von Benutzeranfragen durch Aufrufe von *Model* oder *View*.

Die Umsetzung des Entwurfsmusters und die dadurch entstehenden Schichten erleichtern Änderungen sowie Erweiterungen der Anwendung und ermöglichen die Wiederverwendung einzelner Komponenten.



### **Modularität**

Ist eine Anwendung modular aufgebaut, lässt sie sich in voneinander unabhängige Teile gliedern. Einzelne Module können modifiziert oder ausgetauscht werden, ohne dass eine Modifikation weiterer Module nötig ist. Modular aufgebaute Anwendungen gelten als leicht änderbar, da Änderungen immer nur Teile der Anwendung betreffen und als leicht erweiterbar, da sich neue Module problemlos in die Anwendungsarchitektur integrieren lassen.

### **Einfache Installation**

Eine einfache Installation ermöglicht auch unerfahrenen Benutzern eine Installation der Anwendung. Erreicht werden kann dies unter anderem durch die Entwicklung einer eigenständigen Anwendung, die das Vorhandensein weiterer Anwendungen auf dem System nicht erfordert.

### **Kostengünstig**

Das im Rahmen dieser Arbeit zu entwickelnde Werkzeug richtet sich an Personen, die Interesse daran haben, privat ein Internetangebot zu erstellen. Da das Werkzeug sehbehinderte und blinde Benutzer besonders unterstützt, ist davon auszugehen, dass die Zielgruppe sich zu einem großen Teil aus diesem Personenkreis zusammensetzt. Diese Zielgruppe ist durch notwendige kostenintensive Anschaffungen, wie beispielsweise die einer *Braillezeile*, in finanzieller Hinsicht bereits einer großen Belastung ausgesetzt. Um diese Belastung nicht zu vergrößern, soll bei der Entwicklung des Werkzeuges darauf geachtet werden, dass die benötigten Technologien und Anwendungen kostengünstig zur Verfügung stehen.

## 9 Design

Im letzten Kapitel wurde aufgezeigt, was das System genau leisten soll. Die Designphase soll klären, wie das Werkzeug die aufgestellten Anforderungen erfüllen kann.

In der Praxis gibt es unterschiedliche Möglichkeiten, die im letzten Kapitel dargestellten Anforderungen umzusetzen. Im Folgenden werden verschiedene Lösungswege aufgezeigt und die optimale Lösung für die spezifischen Anforderungen herausgestellt. Grundlegend gilt es beispielsweise zu klären, ob die Anwendung als lokale Anwendung oder als Internetapplikation umzusetzen ist.

Bei der Implementierung des Werkzeugs sollen ausschließlich plattformunabhängige Technologien und *Open-Source*-Produkte verwendet werden. Die Verwendung von plattformunabhängigen Technologien sichert die plattformunabhängige Einsatzmöglichkeit der Anwendung. Der Einsatz von *Open-Source*-Produkten bedeutet eine Vermeidung anfallender Kosten. Darüberhinaus können *Open-Source*-Produkte durch den öffentlich zugänglichen Quellcode bei Bedarf an die Anforderungen angepasst werden und unterliegen einer ständigen Weiterentwicklung und Verbesserung durch die *Open-Source*-Gemeinde.

In den nächsten Abschnitten werden verschiedene Rahmenbedingungen geklärt, um anschließend die Gesamtarchitektur des zu entwickelnden Werkzeugs festzulegen.

### 9.1 Lokal oder Internet?

Computeranwendungen lassen sich in Desktopanwendungen und Internetanwendungen unterscheiden. Desktopanwendungen werden lokal an einem Computer installiert und vom Benutzer auf diesem ausgeführt. Internetanwendungen hingegen sind auf einem Computer installiert, der ans Internet angeschlossen ist, dem *Server*, während der Benutzer die Anwendung von einem weiteren Computer, dem *Client*, über eine Internetverbindung ausführt.

Prinzipiell ist das Erstellen von Internetangeboten mit deren Veröffentlichung im Internet verbunden. Werden die Inhalte mit einer lokalen Anwendung erstellt, muss der Benutzer ein weiteres Programm verwenden, um das erstellte Internetangebot per *File Transfer Protocol* (FTP) auf dem *Server* zu speichern. Internetangebote, die mit Desktopanwendungen erstellt werden, bestehen oft aus statischen (X)HTML-Dateien, da nicht vorausgesetzt wird, dass der später verwendete *Server* eine *serverseitige* Verarbeitung erlaubt.

Wird das Werkzeug als Internetanwendung umgesetzt, überträgt der Benutzer das Werkzeug einmalig per FTP auf den *Server*, führt eine Installation aus und benutzt das Werkzeug anschließend ähnlich anderer Internetangebote. Die erstellten Internetangebote werden hier durch die Anwendung direkt auf dem *Server* gespeichert und sind sofort veröffentlicht. Eine solche Internetanwendung erfordert eine *serverseitige* Verarbeitung der Anfragen, die der *Client* per *HTTP-Request* (*Hypertext Transfer Protocol*) sendet. Diese *serverseitige* Verarbeitung lässt sich auch für die Umsetzung der Internetangebote nutzen. Die Angebote können aus (X)HTML-Seiten bestehen, die erst bei Anfrage durch einen *Client* dynamisch erzeugt werden. Dadurch ist es möglich, Inhalte, die auf mehreren Einzelseiten erscheinen, nicht redundant in allen statischen (X)HTML-Dateien zu speichern, sondern diese Inhalte von einer zentralen Stelle in alle Einzelseiten einzufügen. Die entstehende Trennung von Darstellung und Inhalt ist, wie bereits in Kapitel 6.2 erläutert, eine grundlegende Anforderung an ein WCMS. Alleine diese Umsetzung des Werkzeugs als WCMS ermöglicht es, das Werkzeug auch für Personen ohne nähere Kenntnisse der Techniken (X)HTML und CSS benutzbar zu machen. Darüber hinaus kann ein WCMS die Anforderungen an die entstehenden Internetangebote bezüglich der Gebrauchstauglichkeit automatisiert umsetzen. Dadurch wird erreicht, dass die entstehenden Internetangebote Anforderungen, die durch Gebrauchstauglichkeit entstehen, auch dann erfüllt werden, wenn das Werkzeug von sehbehinderten Benutzern bedient wird, die optische Gestaltungsregeln nicht nachvollziehen können.

Um die *serverseitige* Verarbeitung auch bei der Anzeige der Seiten des Internetangebots zu benutzen, sind die Inhalte dieser Seiten über ein Datenmodell zu speichern. Im folgenden Abschnitt wird diskutiert, welche Art der Datenhaltung benutzt werden soll.

## 9.2 Möglichkeiten zur Datenhaltung

In Kapitel 7 wurden bereits verschiedene Arten der Datenhaltung erläutert: die Verwendung eines Datenbankverwaltungssystems sowie die Speicherung von XML-Dokumenten.

Um zu entscheiden, welche der Möglichkeiten bei der Erstellung des Werkzeugs Verwendung findet, werden im Folgenden deren Vorteile zusammengefasst. Ein Datenbankverwaltungssystem ermöglicht die Verwaltung großer Datenbestände und erlaubt eine parallele und sichere Verarbeitung der Daten durch mehrere Benutzer. Nachteilig ist hier zu sehen, dass auf eine weitere Anwendung, das Datenbankverwaltungssystem, zugegriffen werden muss. Das bedeutet, dass der Benutzer bei der Providerwahl eine zusätzliche Komponente zu berücksichtigen hat. Sind die Daten hingegen im XML-Format im Dateisystem gespeichert, gilt es lediglich bei der Wahl der Programmiersprache zu berücksichtigen, dass diese eine Möglichkeit der XML-Verarbeitung bietet. Als Nachteile sind hier die Begrenzung der Datenmenge, die nicht mögliche parallele Verarbeitung sowie das Fehlen implementierter Zugriffsmechanismen zu sehen.

Als Fazit ist zu sehen, dass Datenbankverwaltungssysteme einen größeren Aufwand bei der Installation des Werkzeugs sowie bei der Providerwahl bedeuten. Bei größeren Anwendungen gleichen die Vorteile eines solchen Systems jedoch diesen Aufwand aus. Bei einer Anwendung, die für Privatpersonen erstellt wird, kann ein Datenbankverwaltungssystem seine Vorteile jedoch kaum unter Beweis stellen. Daher kann in diesem Fall die benutzerfreundliche Alternative besser geeignet sein. Infolgedessen wird die Datenhaltung des Werkzeugs durch XML-Dokumente umgesetzt, wobei die Strukturbe-

schreibung in XML-Schema erfolgt. Um diese Art der Datenhaltung zu ermöglichen, muss die verwendete Programmiersprache die Verarbeitung von XML-Dokumenten erlauben. Im Folgenden sollen verschiedene Programmiersprachen auf ihre Eignung untersucht werden.

## 9.3 Die Programmiersprache

Da die Umsetzung des Werkzeugs als Internetanwendung erfolgt, stehen zwei Möglichkeiten zur Diskussion: *PHP: Hypertext Preprocessor* (PHP) und *Java*. Beide Sprachen stellen Möglichkeiten zur XML-Verarbeitung zur Verfügung und haben verschiedene Vor- und Nachteile. Diese werden in den folgenden Abschnitten erläutert und mit den Anforderungen an das Werkzeug aus Kapitel 8.3 verglichen.

### Java

*Java* ist eine objektorientierte Programmiersprache und stellt mit *Java Servlets*, *Java Server Pages* (JSP) und den *Enterprise JavaBeans* (EJB) vielversprechende Technologien für die Erstellung von Internetanwendungen zur Verfügung.

Bei der Verwendung von *Java*-Technologien zur Erstellung von Internetanwendungen ergeben sich folgende Vorteile:

- *Java*-Programme sind durch die Zwischenschaltung der *Java Virtual Machine* (JVM) plattformunabhängig.
- *Java* eignet sich zur Erstellung von verteilten Anwendungen, ist skalierbar und dadurch an wachsende Anforderungen anpassbar.
- Durch die Objektorientierung der Sprache ist es möglich Anwendungen modular aufzubauen.

### PHP

PHP ist eine weitverbreitete Skriptsprache. Das Ziel von PHP besteht in der Erstellung dynamischer Internetanwendungen auf einfache Art. PHP kann in (X)HTML-Dokumente eingebettet Verwendung finden und ist leicht zu erlernen, da die Syntax anderen Hochsprachen nachempfunden wurde.

Für den Benutzer des Werkzeugs bietet PHP folgende Vorteile:

- Aufgrund der weiten Verbreitung wird PHP-Unterstützung von Internet Providern kostengünstig angeboten.
- Die Installation einer PHP-Applikation erfordert keine tieferen technischen Kenntnisse.
- Da PHP-Unterstützung für alle gängigen Plattformen angeboten wird, kann durch die Verwendung von PHP Plattformunabhängigkeit erreicht werden.

Für die Entwicklung des Werkzeuges ergeben sich durch die Verwendung von PHP weitere Vorteile:

- Anwendungen in PHP sind schnell zu implementieren, da die Sprache keine große Einarbeitung erfordert.
- PHP stellt in der fünften Generation eine neue Umsetzung des *Document Object Models* zur XML-Verarbeitung zur Verfügung. Dadurch kann einerseits die XML-Verarbeitung effizient erfolgen, andererseits eine Trennung von (X)HTML (*View*) und PHP (*Controller*) erreicht werden.
- Ebenfalls neu in der fünften Generation ist die Möglichkeit der vollständig objektorientierten Programmierung. Dies ermöglicht den modularen Aufbau von PHP-Anwendungen und macht diese dadurch leichter änderbar, erweiterbar und ermöglicht die Wiederverwendung einzelner Module.
- PHP ist ein *Open-Source*-Projekt und wird ständig weiterentwickelt und verbessert.

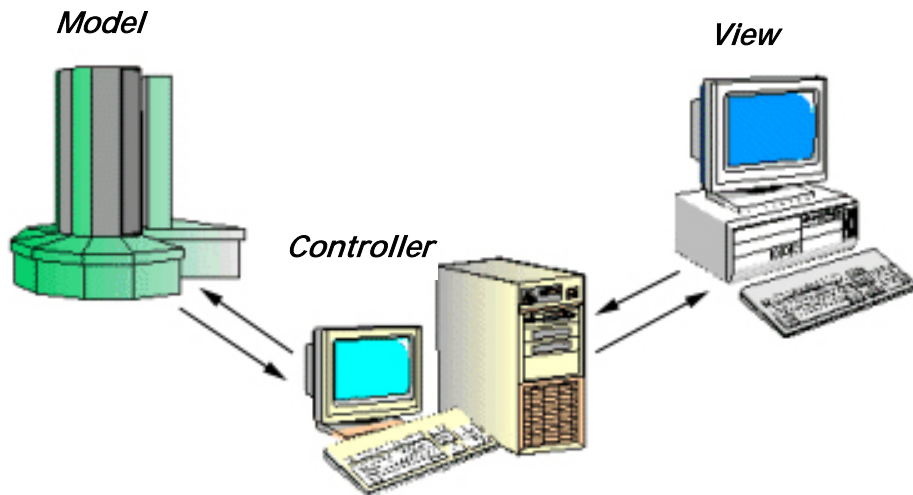
Bei einem direkten Vergleich der beiden Sprachen ist *Java* noch immer als die professionellere Möglichkeit anzusehen. *Java* ist objektorientiert, plattformunabhängig, skalierbar und durch die Internettechnologien steht auch bei der Entwicklung von Internetanwendungen der gesamte Funktionsumfang der Sprache zur Verfügung. Diesen Vorteilen stehen jedoch auch Nachteile gegenüber. Zum einen treibt die benötigte Unterstützung der *Java*-Technologien die Providerkosten in die Höhe, zum anderen erfordert das Installieren einer *Java*-Anwendung auf einem unbekanntem Applikationsserver tiefere Kenntnisse der Materie oder viel Geduld.

Die im Rahmen dieser Arbeit zu entwickelnde Anwendung richtet sich vor allem an Privatpersonen. Da die Nachteile von *Java* in diesem Fall überwiegen wird für die Implementierung des Werkzeuges PHP bevorzugt.

## 9.4 Gesamtarchitektur des Werkzeuges

Nach Bernd Oestereich definiert die Anwendungsarchitektur „*die prinzipiellen Möglichkeiten und Einschränkungen der Problemlösung*“ [Oes01, S.145] und bestimmt, welche Arten von Klassen und Schnittstellen zu entwerfen sind [vgl. Oes01, S.145].

Wie durch die Anforderungen in Kapitel 8.3 aufgezeigt, ist bei der Umsetzung der Anwendung die Trennung der Schichten *Model*, *View* und *Controller*, die durch das MVC-Entwurfsmuster vorgegeben werden, zu berücksichtigen. Es entsteht eine so genannte *3-tier*-Architektur. Das Zusammenspiel der verschiedenen Schichten ist in Abbildung 9.1 dargestellt.



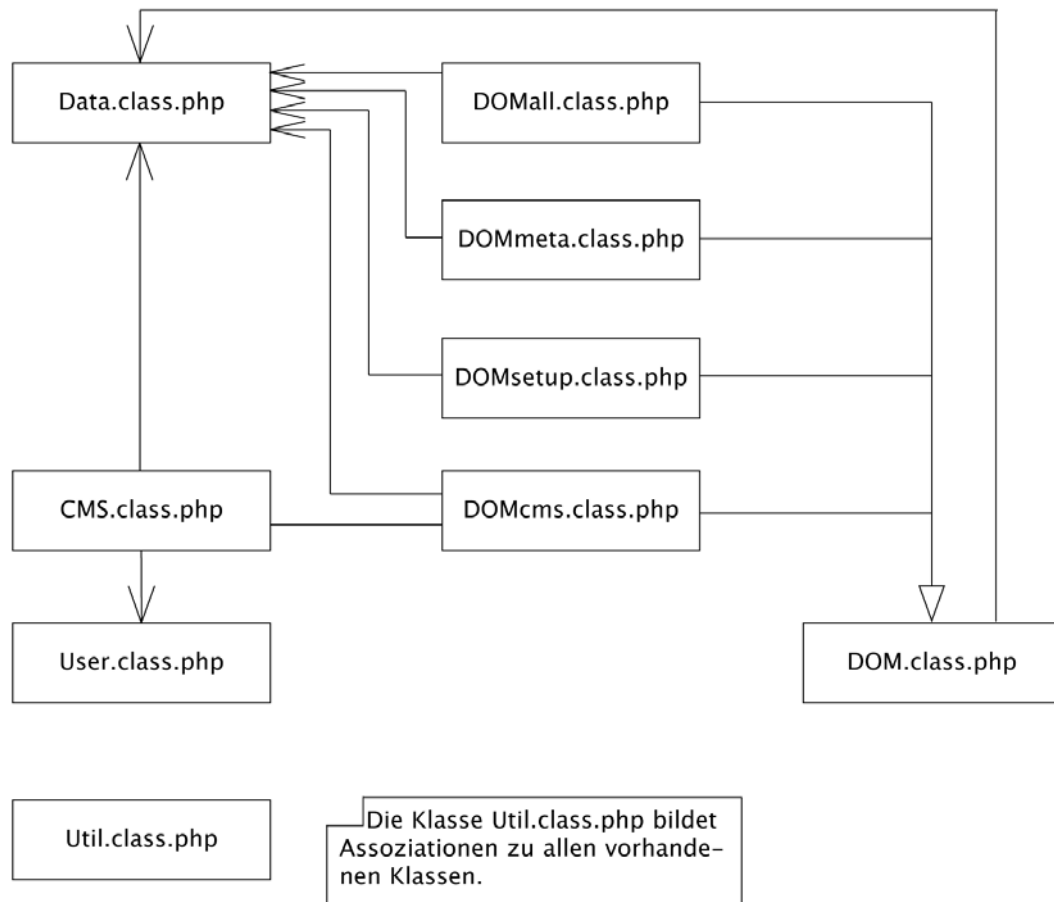
**Abbildung 9.1:** Darstellung einer *3-tier*-Architektur

Nachdem bereits geklärt ist, in welcher Form die Daten des Werkzeugs zu speichern sind und welche Programmiersprache zu verwenden ist, zeigt die folgende Aufzählung, mit welchen Technologien die entstehenden Schichten umzusetzen sind:

- **Model:** Der Zugriff auf die entstehenden XML-Dokumente wird in einer Klasse gekapselt. Diese Modularität ermöglicht ein späteres Austauschen der Datenhaltung.
- **View:** (X)HTML-Seiten ermöglichen dem Benutzer das Absenden von HTTP-Anfragen an den Server.
- **Controller:** Die Programmlogik wird objektorientiert in PHP5 umgesetzt. Der Zugriff auf die Präsentationsschicht erfolgt ausschließlich über die angebotenen DOM-Funktionen. Dadurch entsteht die Trennung der Schichten *View* und *Controller*.

Die objektorientierte Umsetzung der Programmlogik erfolgt nach folgendem Klassendiagramm:





**Abbildung 9.2:** Klassendiagramm der Geschäftslogik

Die Umsetzung der Präsentationsschicht erfolgt durch eine (X)HTML-Vorlage. Während der Programmausführung werden die Inhalte dynamisch über Funktionen der DOM-Implementierung in PHP5 in diese eingefügt und dem Benutzer als Ergebnis angezeigt. Erstellt wird lediglich eine (X)HTML-Vorlage. Diese wird sowohl für die Präsentation des Werkzeugs an sich als auch bei der Darstellung der erstellten Internetangebote verwendet. Dadurch sind später benötigte Änderungen mit geringem Aufwand durchführbar.

Als grundlegende Anforderungen an die (X)HTML-Vorlage sind die korrekte Umsetzung in einer aktuellen Version des Standards sowie die Trennung von Inhalt und Gestaltung zu sehen. Die aktuelle Version des HTML-Standards ist die Version HTML 4.01. Darüber hinaus gilt ebenfalls der Standard XHTML

1.0 als aktuell und zusätzlich als Empfehlung des W3Cs bei der Erstellung von Internetangeboten. Daher werden die Vorlagen in diesem Standard erstellt. Die Gestaltung der Vorlagen wird vollständig von CSS-Dateien übernommen. Dadurch wird einerseits die Trennung von Inhalt und Gestaltung umgesetzt und die Gestaltung ist vollständig änderbar, ohne die (X)HTML-Vorlage zu bearbeiten. Andererseits ermöglicht die Verwendung von CSS die Umsetzung der Seitengestaltung ohne Einsatz von Tabellen zur optischen Gestaltung. Durch diese Form der Gestaltung ist es möglich, dass sich die optische Gestaltung stark von der Reihenfolge der Seitenelemente im Quelltext unterscheidet. Wie unterschiedlich die Darstellung abhängig von der verwendeten CSS-Datei sein kann zeigt Abbildung 9.3.



**Abbildung 9.3:** Darstellung eines Internetangebots mit unterschiedlichen *Stylesheet*-Dateien

## 9.5 Die Arbeit mit dem Werkzeug

Bevor das nächste Kapitel auf die Implementierung des Werkzeuges eingeht, fassen die folgenden Abschnitte zusammen, wie das Werkzeug dem Benutzer das Erstellen und Ändern eines Internetangebotes ermöglicht.

Das Werkzeug ist erreichbar über die Eingabe der URL des Wurzelverzeichnisses mit der Ergänzung „/cms“. Nach der Eingabe der korrekten Zugangsdaten stehen dem Benutzer in der Navigation folgende Möglichkeiten offen:

- **Abmeldung:** Meldet den Benutzer von der Anwendung ab.
- **Projekt bearbeiten:** Ermöglicht das Anlegen und Ändern von Daten, die das gesamte Internetangebot betreffen.
- **Seiten bearbeiten:** Hier erfolgt die Bearbeitung bereits vorhandener Einzel-seiten sowie das Anlegen neuer Seiten.

Über den Navigationspunkt „*Projekt bearbeiten*“ gelangt der Benutzer zu einem (X)HTML-Formular. Dieses Formular ermöglicht dem Benutzer das Eintragen von Daten, die das gesamte Internetangebot betreffen. Das Absenden des Formulars speichert die Werte. Folgende Datengruppen werden abgefragt:

- **Angaben über den Ersteller des Angebots:** Diese Angaben werden in den Metadaten des Internetangebots angezeigt und durch das Impressum des Angebots bereitgestellt.
- **Angaben über das Internetangebot:** Diese Daten enthalten unter anderem Titel, Beschreibung sowie Schlüsselwörter, die das Angebot näher beschreiben. Verwendung finden diese Angaben in den Metadaten des Angebots sowie im Kopf jeder Einzelseite.
- **Auswahl der Gestaltung:** Hier ermöglicht ein Formularelement dem Benutzer das Festlegen der optischen Gestaltung des gesamten Internetangebots.

Der Navigationspunkt „*Seiten bearbeiten*“ ermöglicht dem Benutzer das Anlegen neuer Seiten sowie das Bearbeiten bereits vorhandener Seiten durch entsprechende (X)HTML-Formulare. Das Formular zum Anlegen einer neuen Seite umfasst folgende Angaben:

- Der **Seitentitel** wird in der Navigation als Verweistext verwendet und erscheint zusätzlich im `title-Tag` der Seite.
- Die **Beschreibung** erscheint in der Navigation als *Tooltip*.
- Das Festlegen einer **übergeordneten Seite** bestimmt, an welcher Stelle sich die neue Seite in der Navigation befindet. Alle Seiten entstehen so als direkte oder indirekte Nachfolger der Startseite und bilden eine hierarchische Struktur. Diese Struktur ist wie bereits in Kapitel 5.2.4 gezeigt für Benutzer leicht zu verstehen.

Zusätzlich zu diesen Angaben ermöglicht das Formular zum Bearbeiten vorhandener Seiten das Anlegen, Ändern und Löschen verschiedener Elemente aus welchen der Seiteninhalt aufgebaut ist. Mögliche Elemente sind hier:

- **Überschriften:** Erlauben eine Gliederung der Seite in verschiedene Ebenen.
- **Textabsätze:** Ermöglichen die Gliederung der Texte und unterstützen zusammen mit Überschriften das Leseverhalten von Internetbenutzern, das in Kapitel 5.2.2 dargestellt wurde.
- **Sprachwechsel:** Ermöglichen das Wechseln der natürlichen Sprache innerhalb von Textabsätzen.
- **Verweise:** Bestehen aus dem Verweistext sowie dem Sprungziel.
- **Bilder:** Werden zusätzlich mit einem Text-Äquivalent versehen.

## 10 Implementierung

Es wurde im Einzelnen aufgezeigt, in welche Komponenten das Werkzeug zu gliedern ist. Nun gilt es diese Komponenten umzusetzen. Im Folgenden ist die Implementierung der entstandenen Schichten erläutert.

### 10.1 Model: Aufbau der XML-Dokumente

Die Datenhaltung erfolgt durch die Speicherung von XML-Dokumenten. Um zu umfangreiche Dokumente zu vermeiden, werden einerseits Dokumente angelegt, welche Daten speichern, die das gesamte Internetangebot betreffen, andererseits wird pro Seite des Angebots ein XML-Dokument erstellt, in dem der spezifische Seiteninhalt hinterlegt wird. In den folgenden Abschnitten werden die Verwendung und der Inhalt der einzelnen Dokumente erläutert.

#### **project.xml**

Die Datei `project.xml` speichert Angaben, die das gesamte Internetangebot betreffen. Hierzu gehören die folgenden Angaben:

- **css**: Beinhaltet den Namen der aktuell zu verwendenden CSS-Datei.
- **description, keywords**: Speichern die Werte für die entsprechenden Metaangaben im Header der (X)HTML-Ausgaben.
- **title**: Dieser Wert erscheint sowohl im *title-Tag* der (X)HTML-Seiten und dadurch in der Browserleiste und auch im Seitenkopf und ermöglicht so die sofortige Identifikation des Internetauftritts auf jeder Einzelseite.
- **subtitle**: Beschreibt den Internetauftritt näher und erscheint neben dem Titel im Kopf jeder Seite des Internetauftritts.
- **author**: Die hier hinterlegten Angaben dienen zur Angabe der Metadaten. Darüber hinaus erscheinen die Werte bei der Anzeige des Impressums, welches bei entsprechender Anfrage durch einen *Client* automatisch erstellt wird.

```
<project counter="1">
  <css>css</css>
  <imprint>
    <author>
      <name>name</name>
      <firstname>vorname</firstname>
      <street>strasse</street>
      <zip>11111</zip>
      <city>stadt</city>
      <state>staat</state>
    </author>
    <title>titel</title>
    <subtitle>untertitel</subtitle>
  </imprint>
  <description>beschreibung</description>
  <keywords>schluesselwoerter</keywords>
</project>
```

Listing 10.1: Datei `project.xml`

### **navigation.xml**

Da die Navigation des Internetangebots automatisch durch das Werkzeug generiert wird, werden Angaben über die hierarchische Struktur separat in dieser Datei gespeichert. Der Aufbau der Navigation erfordert so lediglich die Verarbeitung einer XML-Datei: `navigation.xml`.

Die hierarchische Struktur des Internetangebots wird als XML-Baum abgebildet. Zu jedem Navigationseintrag sind folgende Werte gespeichert:

- **id:** Ermöglicht die Identifikation der entsprechenden Seite.
- **text:** Wird als Text des Navigationselementes angegeben.
- **description:** Wird dem Verweis zur näheren Beschreibung des Sprungziels als `title`-Element angefügt.

```
<navigation id="id0">
  <punkt id="id1" text="text 1" description="text 1">
    <punkt id="id2" text="text 2" description="text 2">
      <punkt id="id3" text="text 3" description="text 3"/>
      <punkt id="id4" text="text 4" description="text 4"/>
    </punkt>
    <punkt id="id8" text="text 8" description="text 8"/>
  </punkt>
  <punkt id="id9" text="text 9" description="text 9"/>
</navigation>
```

**Listing 10.2:** Datei `navigation.xml`

### **`idxx.xml`**

Die Datei `idxx.xml` steht exemplarisch für Dateien, die durch das Erstellen einer neuen Seite entstehen. Bereits bei der Installation wird eine solche Datei für die Startseite des Angebots angelegt. Der angegebene Seitentitel erscheint zusätzlich zu dem Titel, der bereits in der Datei `project.xml` gespeichert ist, im `title`-Element der (X)HTML-Seite. Dadurch kann jeder Seite ein individueller Titel zugeordnet werden, was die Identifikation der Einzelseiten ermöglicht.

```
<side id="idxx">
  <title>seitentitel</title>
  <content>
    <h2>Ueberschrift</h2>
    <p>Text eines Absatzes</p>
    <p>Text eines weiteren Absatzes</p>
    <h3>Noch eine Ueberschrift</h3>
    <p>Text eines weiteren Absatzes</p>
  </content>
</side>
```

**Listing 10.3:** Datei `idxx.xml`

Das Element `content` enthält den gesamten Seiteninhalt. Der Inhalt liegt bereits in korrekter XHTML-Syntax vor und ist lediglich in die (X)HTML-Vorlage einzufügen. Auf Überschriften der Ebene `h1` wird im Inhaltsteil verzichtet, da durch Überschriften dieser Ebene die grundlegenden Teile der Seite gekennzeichnet werden.

## 10.2 view: (X)HTML-Vorlagen

Die Präsentationsschicht der Anwendung ist die einzige Schicht, mit welcher der Benutzer direkt interagiert. Aus diesem Grund gilt es hier, einen Großteil der Barrierefreiheit und der Gebrauchstauglichkeit der Anwendung umzusetzen. Um dies zu erreichen, werden beim Aufbau der Präsentationsschicht die Anforderungen der BITV und der Gebrauchstauglichkeit, die im Theorieteil bereits erläutert wurden, berücksichtigt.

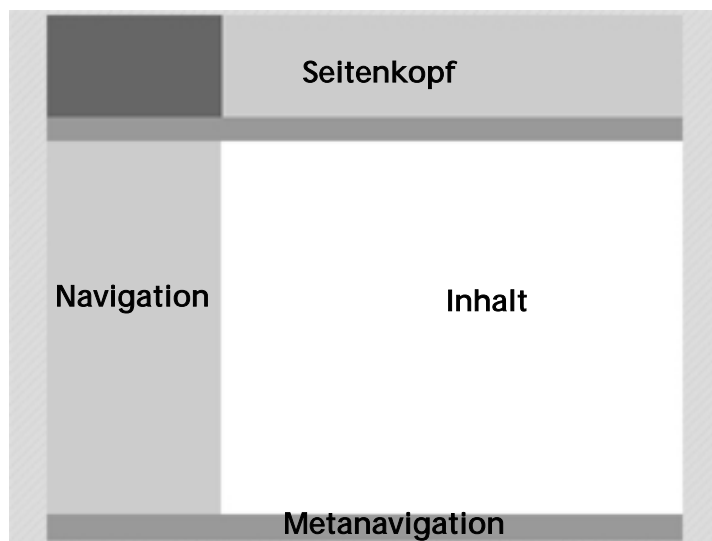


Abbildung 10.1: Aufbau der (X)HTML-Vorlage

Wie in Abbildung 10.1 dargestellt setzt die (X)HTML-Vorlage den klassischen Aufbau einer Internetseite um. Dieser Aufbau unterstützt die allgemeinen Erwartungen der Internetbenutzer an die Anordnung bestimmter Seitenelemente, die bereits in Kapitel 5.2.5 erläutert sind, sowie die Eigenschaften des menschlichen Gehirns, die Kapitel 5.2.6 darstellt. Das ermöglicht eine intuitive

Benutzung des Angebots, da auf bereits vorhandenes Wissen zurückgegriffen werden kann. Im Folgenden sind die einzelnen Seitenelemente sowie deren Funktion erläutert:

- **Seitenkopf:** Enthält drei Bereiche. Links einen Verweis zur Startseite, rechts den Titel sowie einen Navigationsbereich, der den aktuellen Standpunkt des Benutzers in der Seitenhierarchie angibt, die *Breadcrumbs*.
- **Navigation:** Zeigt den gesamten Baum der Seitenstruktur.
- **Inhalt:** Der zentrale Bereich um die Inhalte der Seite darzustellen.
- **Metanavigation:** Enthält Verweise zu Impressum, Inhaltsübersicht und Hilfstexten.

Die Umsetzung weiterer Regeln der Gebrauchtauglichkeit zeigt Abbildung 10.2. Zu sehen ist, wie durch die Anpassung von Hintergrund- und Schriftfarbe eine optimale Kontrastwirkung, wie in Kapitel 5.2.1 beschrieben, erzielt wird. Zusätzlich werden die Ergebnisse aus Kapitel 5.2.1 in Bezug auf Schriftart, Zeilenlänge und Textausrichtung berücksichtigt.



Abbildung 10.2: Ansicht des Internetangebots



Nicht beachtet wurde bei der Festlegung der Schriftgröße der ermittelte Wert von 12 bis 14 Punkten. Statt dessen ist die Schriftgröße in der Maßeinheit *em* festgelegt. Dadurch orientiert sich die Größe der Schrift an der im Betriebssystem eingestellten Standardschriftgröße und ist durch den *Internetbrowser* individuell anpassbar. Abbildung 10.3 zeigt Abbildung 10.2 mit einer im *Browser* eingestellten Skalierung der Schrift auf 200 Prozent.



**Abbildung 10.3:** Vergrößerung der Schriftgröße durch den Internetbrowser

Linearisiert man den in Abbildung 10.1 dargestellten Seitenaufbau durch die Abtastung von der oberen linken Ecke in die untere rechte Ecke, würde die Navigation eine zentrale Stelle zu Beginn der linearisierten Seite einnehmen. Für Benutzer eindimensionaler Ausgabemedien, die auf eine linearisierte Form einer Internetseite angewiesen sind, steht jedoch der Inhalt der Seite im Vordergrund. Die Navigation stellt ein Leitsystem dar, auf welches der Benutzer lediglich bei Bedarf zurückgreift. Die Linearisierung der Seiteninhalte durch *Screenreader* erfolgt jedoch nicht anhand der auf dem Bildschirm dargestellten Gestaltung, sondern aufgrund der Reihenfolge der Elemente im Quelltext. Wie

in Listing 10.4 vereinfacht dargestellt, erscheinen die Navigationselemente im Quelltext erst zum Ende der Seite.

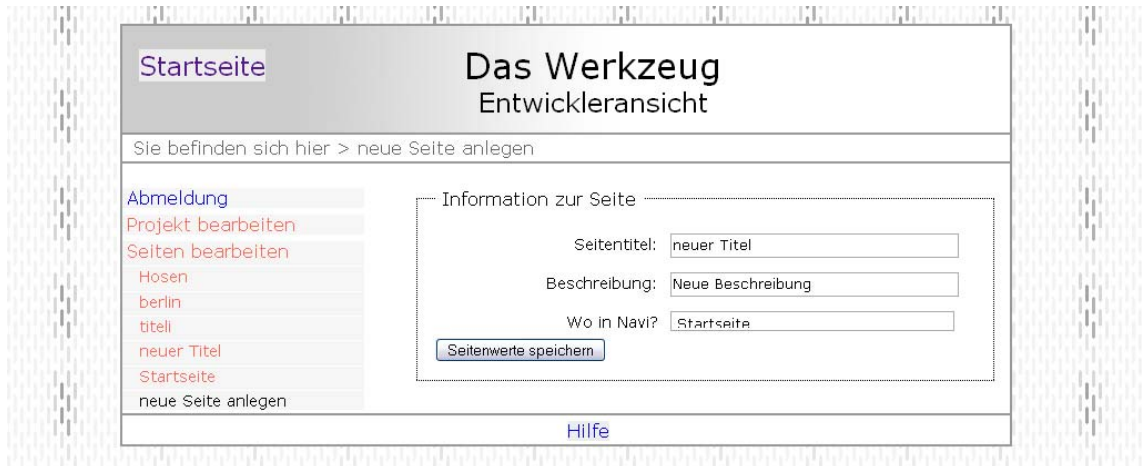
```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head></head>
  <body>
    <div>
      <div id="jumps">Spungmarken</div>
      <div>Seitenkopf</div>
      <div>Breadcrumbs</div>
      <div id="content">Inhalt</div>
      <div id="navi">Navigation</div>
      <div id="metanavi">Metanavigation</div>
    </div>
  </body>
</html>
```

**Listing 10.4:** Vereinfachte Darstellung der Reihenfolge im Seitenquelltext

Weitere Unterstützung für die Benutzer eindimensionaler Ausgabemedien bieten unsichtbare Elemente, die die Orientierung erleichtern:

- Überschriftselemente (`<h1></h1>`) kennzeichnen den Inhaltsteil sowie den Navigationsteil, um diese Seitenelemente möglichst zeitsparend erreichbar zu gestalten.
- Zusätzliche Verweise zu Beginn der Seite ermöglichen ein direktes Erreichen des Seiteninhalts, der Navigation sowie der Metanavigation.
- Die Navigation ist durch verschachtelte Listen umgesetzt, da *Screenreader* vermehrt Möglichkeiten anbieten, diese optimal abzuarbeiten.
- Die Einträge im Navigationsbaum sind mit einer Nummerierung versehen. Der Aufbau entspricht dem Beispiel „1.2.1.“. Der Abschluss durch einen Doppelpunkt verhindert die Interpretation der Nummerierung als Datumsangabe durch *Screenreader*.
- Um einzelne Navigationspunkte voneinander zu trennen, wird die Zeichenfolge „&nbsp;&nbsp; “ verwendet. Diese bewirkt bei Vorleseprogrammen eine Trennung durch eine Senkung der Stimme und auf *Braillezeilen* durch Ausgabe des Punktes.

Abbildung 10.4 zeigt die Entwickleransicht des Werkzeugs mit und ohne Verwendung einer *Stylesheet*-Datei. In der Darstellung ohne CSS sind die verborgenen Orientierungshilfen sichtbar.



[Direkt zum Inhalt](#) . [Direkt zur Navigation](#) . [Direkt zur Metanavigation](#) .

[Startseite](#)

Das Werkzeug

Entwickleransicht

Sie befinden sich hier > neue Seite anlegen

## Inhalt

Information zur Seite

Pflichtfeld Seitentitel:  Pflichtfeld Beschreibung:  Wo in Navi?

## Navigation

- [1: Abmeldung](#) .
- [2: Projekt bearbeiten](#) .
- [3: Seiten bearbeiten](#) .
  - [3.1: Hosen](#) .
  - [3.2: berlin](#) .
  - [3.3: titeli](#) .
  - [3.4: neuer Titel](#) .
  - [3.5: Startseite](#) .
  - [neue Seite anlegen](#) .

## Metanavigation

[Hilfe](#) .

**Abbildung 10.4:** Navigation in der Entwickleransicht mit und ohne Verwendung eines Stylesheets

Um Elemente unsichtbar zu gestalten, ermöglichen CSS die Kennzeichnung durch `display: none`. Jedoch wird diese Angabe auch vermehrt durch Ausleseprogramme interpretiert. Das bedeutet, dass die Elemente ignoriert werden. Um die Anzeige auf dem Bildschirm zu verhindern, jedoch *Screenreader* zur Ausgabe zu zwingen hat sich folgende Möglichkeit bewährt:

```
invisible {
  position: absolute;
  left: -1000px;
  top: -1000px;
  width: 0px;
  height: 0px;
  overflow: hidden;
  display: inline;
}
```

**Listing 10.5:** CSS-Definition nicht sichtbarer Elemente

Hier sind die Elemente durch eine absolute Positionierung aus dem Anzeigebereich eines Bildschirms hinausgeschoben.

Besondere Unterstützung bei der Benutzung eindimensionaler Ausgabemedien bieten ebenfalls die Formulare, die beim Erstellen von Internetauftritten dynamisch in den Inhaltsbereich der (X)HTML-Vorlagen eingefügt werden.

- Zusammengehörige Formularelemente werden durch die Verwendung eines `fieldset`-Elements gruppiert. Diese Gruppen erhalten durch ein `legend`-Element eine explizite Beschreibung.
- Pflichtfelder sind zusätzlich unsichtbar mit dem Wort „*Pflichtfeld*“ gekennzeichnet. Üblich ist die Kennzeichnung durch ein Sternchen, jedoch bieten *Screenreader* die Unterdrückung von Interpunktionszeichen an, um störende Ausgaben zu reduzieren. Bei aktivierter Unterdrückung geht eine solche Kennzeichnung für den Benutzer verloren.
- Eingabefelder, deren Inhalte bei Überbelegung zu Darstellungsproblemen führen können, beispielsweise der Seitentitel, der in der Navigation verwendet wird, sind mit einer Maximallänge gekennzeichnet. Die Eingabe zu langer Texte ist dadurch nicht möglich. Dies garantiert eine fehlerfreie Darstellung der Internetseiten auch bei der Angebotserstellung durch blinde Benutzer.

## 10.3 Controller: Umsetzung in PHP5

Die Umsetzung der Geschäftslogik des Werkzeugs erfolgt objektorientiert in PHP5. Im Folgenden werden die einzelnen Klassen, ihre Aufgabe sowie ihre Zusammenarbeit untereinander genau erläutert.

### Die Klasse `DOM.class.php`

Diese Klasse bildet die Basisklasse der Klassen `DOMall.class.php`, `DOMmeta.class.php`, `DOMsetup.class.php` und `DOMcms.class.php`. Jede dieser Klassen ist für den Aufbau bestimmter Seiten des Werkzeugs sowie der erstellbaren Internetauftritte verantwortlich. Elemente, die in jeder Seite Verwendung finden, sind durch die Basisklasse definiert. Zu diesen Elementen gehören die Sprungmarken zu Beginn einer Seite, die Navigation und Metanavigation sowie der Seitenkopf. Darüber hinaus stellt diese Klasse eine Methode zur Ausgabe der entstandenen (X)HTML-Seiten bereit. Bei spezifischen Anforderungen an den Aufbau eines Elements, folgt die Überschreibung der entsprechenden Methoden in den spezialisierten Subklassen.

### Die Klasse `DOMall.class.php`

Durch die Klassen `DOMall.class.php` und `DOMmeta.class.php` werden aufbauend auf die Klasse `DOM.class.php` die Internetseiten des entstehenden Auftritts dynamisch generiert. Um zu entscheiden, welche Internetseite anzuzeigen ist, wird das Ziel jeder HTTP-Anfrage im Parameter `cmd` übermittelt. Die Klasse `DOMall.class.php` übernimmt hierbei die Zusammenstellung der Startseite sowie aller Unterseiten. Um diese Spezialisierung umzusetzen, ist lediglich eine Methode zu denen der Basisklasse zu ergänzen: `createContent()`. Diese Methode fragt den Inhalt der entsprechenden Seite durch die Klasse `Data.class.php` ab und fügt diesen über DOM-Funktionen in die (X)HTML-Vorlage ein.

### Die Klasse `DOMmeta.class.php`

Diese Klasse spezialisiert sich auf die Generierung der Internetseiten, die über die Metanavigation des Internetangebots erreichbar sind: Impressum und Inhaltsübersicht. Auch hier ist zu den Methoden der Basisklasse lediglich die Methode `createContent()` zu ergänzen. Impressum und Inhaltsübersicht werden aus vorhandenen persistenten Daten automatisch, ohne Einbeziehung des Benutzers des Werkzeugs, generiert.

### Die Klasse `DOMsetup.class.php`

Die Klasse `DOMsetup.class.php` wird bei der Installation des Werkzeugs auf einem *Server* ausgeführt. Die Klasse generiert (X)HTML-Seiten, durch die der Benutzer den zu verwendenden Benutzernamen sowie das Passwort eingeben kann. Anschließend speichert das Werkzeug diese Daten und legt verschiedene, zur weiteren Ausführung benötigte Dateien an.

### Die Klasse `DOMcms.class.php`

Die Klasse `DOMcms.class.php` übernimmt vollständig die Generierung der Internetseiten, die das Werkzeug an sich benötigt. Folgende Vorgänge sind hierbei zu berücksichtigen:

- An- und Abmeldung des Benutzers.
- Anlegen und Ändern der Daten, die das gesamte Internetangebot betreffen, die sogenannten Projektdaten.
- Anlegen, Löschen und Ändern einzelner Seiten des Internetangebots.
- Dokumentationsseiten, die dem Benutzer bei Problemen eine Hilfestellung bieten.

Um diese verschiedenen Vorgänge zu unterscheiden, enthält jede HTTP-Anfrage den Parameter `cmd`. Dadurch wird das Ziel der Anfrage übermittelt. Die Funktion `createContent()` untersucht diesen Parameter und ruft entsprechende Methoden der Klasse `CMS.class.php` auf.

Zusätzlich zu dem Inhaltsteil der Internetseiten sind Seitenkopf, Navigation und Metanavigation dynamisch zu erstellen. Beim Aufbau der Navigation werden einerseits die fest vorhandenen Navigationspunkte wie „*Projekt bearbeiten*“, „*Startseite bearbeiten*“, „*neue Seite anlegen*“ sowie An- oder Abmeldung erstellt. Andererseits wird durch die entsprechende Methode der Basisklasse `DOM.class.php` die Navigation der Seitenstruktur eingefügt. Diese ermöglicht die Bearbeitung der bereits erstellten Seiten.

#### **Die Klasse `CMS.class.php`**

Diese Klasse übernimmt die zentrale Rolle bei der Erstellung von Internetangeboten. Hier werden alle Funktionalitäten bereitgestellt, von der Eingabe der Projektdaten bis zum Löschen bereits erstellter Seiten:

- **Anmeldung:** Vergleich der übertragenen Werte mit den gespeicherten Benutzerdaten.
- **Abmeldung:** Löschen der Session durch Zugriff auf die Klasse `User.class.php`.
- **Projektdaten bearbeiten:** Zeigt die Projektdaten in einem entsprechenden Formular an und ermöglicht so deren Bearbeitung.
- **Anlegen einer neuen Seite:** Fügt eine neu angelegte Seite in die Navigation ein und speichert die neuen Inhalte.
- **Löschen einer Seite:** Löscht den entsprechenden Eintrag in der Navigation sowie die gespeicherten Seiteninhalte.

#### **Die Klasse `Data.class.php`**

Diese Klasse kapselt den gesamten Zugriff auf die persistenten Daten des Werkzeugs. Die entstehende Modularität ermöglicht einen Austausch der Datenschicht, wobei sich dies nicht auf die Geschäftslogik auswirkt. In Falle eines Austauschs ist lediglich die Klasse `Data.class.php` neu zu implementieren. Durch einen Austausch der Datenschicht ist die Anpassung der Anwendung an wachsende Anforderungen möglich, denn wie bereits in Kapitel 7 erwähnt, scheint die Datenhaltung durch XML-Dokumente den Anforderungen größerer Anwendungen nicht gewachsen. Wird zur Datenhaltung ein Datenbankver-

waltungssystem eingesetzt, fragt die Klasse `Data.class.php` die benötigten Daten ab und vermittelt diese an die Geschäftslogik. Als Zwischenformat findet hier weiterhin XML Verwendung.

Die Klasse bildet eine Sammlung verschiedener Methoden um die persistenten Daten abzufragen und zu manipulieren. Die Methoden der Klasse werden durch folgende Klassen aufgerufen:

- **Abfrage:** Die Klassen `DOMall.class.php`, `DOMmeta.class.php`, `DOMcms.class.php` sowie die Oberklasse dieser Klassen `DOM.class.php` benötigen eine Abfrage der Daten, um diese dynamisch in die, durch einen *HTTP-Request* angeforderten, Internetseiten einzufügen. Diese Klassen greifen lediglich auf die *Getter*-Methoden zur Datenabfrage zu.
- **Manipulation:** Die Klassen `DOMsetup.class.php` und `CMS.class.php` benötigen die Möglichkeit der Manipulation der persistenten Daten. Daher erfolgen durch diese Klassen zusätzlich Aufrufe der Methoden zur Datenmanipulation.

#### Die Klasse `Utility.class.php`

Diese Klasse stellt Funktionalitäten zur Verfügung, die in vielen Klassen Verwendung finden. Eine solche Funktionalität stellt die Abfrage der vom *Client* durch *post* oder *get* übermittelten Parameter dar. Um nicht bei jeder Abfrage separat auf die globalen von PHP zur Verfügung gestellten Variablen zugreifen zu müssen, ist die Abfrage in folgender Methode gekapselt:

```
static function getParam($name) {
    global $_GET, $_POST;
    return (isset($_POST[$name]))
        ? $_POST[$name]
        : ((isset($_GET[$name]))
            ? $_GET[$name]
            : null);
}
```

**Listing 10.6:** Methode zur Abfrage von *get*- und *post*-Parametern



### Die Klasse `User.class.php`

In dieser Klasse werden die Zugriffe auf in einer Session hinterlegte Werte gekapselt. In der Einbenutzerversion des Werkzeugs gilt es hier lediglich die Information zu verwalten, ob ein Benutzer bei der Anwendung angemeldet ist oder nicht. Um die Anwendung für mehrere Benutzer auszulegen, ist eine Erweiterung dieser Klasse zur Benutzerverwaltung denkbar.

### Die Datei `text.php`

In dieser Datei sind Textelemente definiert. Alle Textbestandteile des Werkzeugs sind in dieser Datei sowie der (X)HTML-Vorlage selbst gespeichert. Dadurch kann durch den Austausch dieser Dateien eine Mehrsprachigkeit des Werkzeugs erreicht werden.

## 11 Test

Im Rahmen einer Testphase ist abschließend zu klären, ob die Barrierefreiheit des Werkzeugs und den damit erstellbaren Internetangeboten ausreichend umgesetzt wurde.

Der erste Schritt auf dem Weg zu barrierefreien Internetangeboten ist die Umsetzung der Angebote in der aktuellen Version eines öffentlich zugänglichen und vollständig dokumentierten Standards. Im Rahmen der vorliegenden Arbeit findet der Standard XHTML 1.0 des W3Cs Verwendung. Zur Überprüfung, inwieweit Internetseiten der Syntax des Standards entsprechen, stellt das W3C unter [Val] eine automatische Validierung der Syntax von Internetseiten zur Verfügung. Alle Internetseiten, die durch die Benutzung des Werkzeuges entstehen, das heißt die Seiten des Werkzeuges an sich sowie die Seiten der damit erstellbaren Internetangebote, entsprechen dem Standard XHTML 1.0 in korrekter Syntax und wurden mit dem Validator des W3Cs überprüft.

An barrierefreie Internetangebote gibt es über die Umsetzung in korrekter Syntax hinaus weitere Anforderungen. Diese sind in Kapitel 4.1.5 ausführlich erläutert. Die Überprüfung dieser Anforderungen ermöglicht das Werkzeug *Bobby* der Firma *Watchfire* [vgl. Wat]. Die oben genannten Internetseiten wurden mit *Bobby* auf Barrierefreiheit untersucht und entsprechen der Priorität 3 der WCAG 1.0.

Da nicht alle Anforderungen der BITV durch automatisierte Tests zu überprüfen sind, war die Verwendung des Werkzeuges zusätzlich durch Benutzer zu kontrollieren. Diese Tests wurden einerseits durch sehende Benutzer durch den Gebrauch des Programms *Webformator* in Verbindung mit einem *Screenreader* durchgeführt. Andererseits hat eine blinde Benutzerin die Verwendung des Werkzeuges geprüft und dessen Tauglichkeit bestätigt.

## 12 Zusammenfassung und Ausblick

Nachdem in Kapitel 8.3 erläutert wurde, welche Anforderungen an das Werkzeug bestehen und in Kapitel 10 die Details der technischen Umsetzung aufgezeigt wurden, werden im Folgenden die erreichten Ergebnisse zusammengefasst und mögliche Erweiterungen dargestellt.

### 12.1 Ergebnisse

Ziel der vorliegenden Diplomarbeit ist die Konzeption und Entwicklung eines Werkzeugs, das es sehbehinderten Menschen ermöglicht, selbstständig Internetangebote zu erstellen. Als wichtige Anforderungen bei der Umsetzung waren Barrierefreiheit und Gebrauchstauglichkeit der entstehenden Internetangebote sowie des Werkzeugs selbst anzusehen. Um dies zu erreichen, galt es verschiedene Vorgänge durch das Werkzeug zu automatisieren, deren Bearbeitung durch Benutzer ohne technische Kenntnisse und ohne Möglichkeit der optischen Gestaltung nicht möglich wäre.

Um die Möglichkeiten der Umsetzung des Werkzeugs zu erarbeiten, wurde zuerst die Zielgruppe barrierefreier Internetangebote näher untersucht und herausgestellt, welche Arten von Behinderungen die Arbeit am Computer einschränken. Darauf aufbauend galt es die Anforderungen an barrierefreie Informationstechnik sowie die rechtlichen Grundlagen und Standards zu erläutern. Da auch nichtbehinderte Menschen bestimmte Anforderungen an ein benutzerfreundliches Internetangebot stellen und die erstellbaren Internetangebote auch für diese Benutzer zu optimieren sind, wurde untersucht, wie die Gebrauchstauglichkeit eines Internetangebots bei einer unbestimmten Zielgruppe erhöht werden kann. Herausgearbeitet wurden Regeln, die sowohl die technische Umsetzung als auch die optische Gestaltung der Internetangebote betreffen.

Anschließend wurde im Rahmen der theoretischen Diskussion betrachtet, welche Möglichkeiten es gibt, um Internetangebote, die den erläuterten Regeln entsprechen, zu erstellen. Untersucht wurden die Vor- und Nachteile von Editoren und Werkzeugen, die eine weiterführende Unterstützung bieten, *Content Management Systeme*. Da sich die Art des Werkzeuges auch auf die Datenhaltung auswirkt, wurden anschließend zwei Möglichkeiten der Datenhaltung dargestellt: Ein Datenbankverwaltungssystem, welches auch bei großen Anwendungen eine sehr gute Verwaltung von Daten ermöglicht, jedoch eine zusätzliche Applikation erfordert und die Speicherung im XML-Format, die sich bei kleineren Anwendungen verwenden lässt und keine Installation zusätzlicher Anwendungen erfordert.

Als Schlussfolgerung der in die Diskussion eingebrachten Möglichkeiten wurde das Werkzeug mit verschiedenen Eigenschaften eines WCMS entwickelt, wobei Datenhaltung, umgesetzt durch XML-Dokumente, und Darstellung, durch die Auszeichnungssprache XHTML, streng voneinander getrennt wurden.

Da die Anwendung aufgrund einer freundlicheren Benutzung als Internetanwendung umzusetzen war, kamen für die Implementierung der Geschäftslogik die Programmiersprachen PHP und Java in Frage. Beide Sprachen bieten zur XML-Verarbeitung eine Umsetzung des *Document Object Models* an und er-

möglichen eine Trennung der Programmlogik von der Darstellung in (X)HTML. Um sowohl die Entwicklungszeit als auch die Kosten für den Anwender gering zu halten, wurde zur Umsetzung der Geschäftslogik PHP5 gewählt.

Die prototypische Umsetzung des Werkzeugs muss zwar aufgrund des wachsenden Umfangs auf viele Möglichkeiten bei der Erstellung von Internetangeboten verzichten, jedoch wurde gezeigt, wie Internetangebote einerseits auf die Benutzung durch behinderte Menschen, andererseits auf die gewachsenen Anforderungen von Internetbenutzern abgestimmt werden können. Darüber hinaus wurde eine Möglichkeit geschaffen, wie sehbehinderte und blinde Menschen solche Angebote selbstständig und kostengünstig erstellen können.

## 12.2 Mögliche Erweiterungen

Die Umsetzung des Werkzeugs im Rahmen der vorliegenden Arbeit stellt lediglich eine prototypische Implementierung dar. Aus diesem Grund sind Erweiterungsmöglichkeiten zahlreich vorhanden. Die naheliegendsten sollen an dieser Stelle nicht unerwähnt bleiben.

Das erstellte Werkzeug ermöglicht die Erstellung von textlastigen Internetangeboten. Übliche Internetangebote enthalten jedoch neben Textinhalten auch Bilder, Video- oder Audiosequenzen. Um diese Inhalte barrierefrei in die erstellbaren Angebote integrieren zu können, müsste das Werkzeug entsprechende Abläufe und Möglichkeiten bereitstellen.

Zur Zeit ist das Werkzeug lediglich für einen Einbenutzerbetrieb geeignet. Eine Benutzerverwaltung ist daher nicht integriert. Soll das Werkzeug neben Privatpersonen auch Vereine bei der Erstellung von Internetangeboten unterstützen, ist eine entsprechende Benutzerverwaltung hinzuzufügen.

## Anhang A: Die Brailleschrift

Die *Brailleschrift*, auch Blinden- oder Punktschrift genannt, wurde 1825 von dem Franzosen Louis Braille (1809 - 1852) erfunden. Die Schrift basiert auf einem durch in Papier geprägte Punkte, tastbaren Alphabet.

Jedes Zeichen besteht aus einem Grundgerüst von sechs Punkten. Diese Punkte sind, wie die Punkte auf einem Würfel, in zwei vertikalen Reihen zu je drei Punkten angeordnet. Es können 64 ( $2^6$ ) verschiedene Zeichen dargestellt werden, abhängig davon, welche Punkte in das Papier geprägt und welche ausgelassen werden.

In der so genannten Vollschrift werden Worte aus einzelnen Buchstaben des Alphabets zusammengesetzt. Zusätzlich sind Zeichen für Ziffern, Satz- und Sonderzeichen sowie Doppellaute vorgesehen. Darüber hinaus haben sich im Laufe der Zeit in vielen Ländern neben der Vollschrift verschiedene Kurzschriften entwickelt. Hier ermöglicht das Zusammenfassen von Lauten, Worten oder Wortstämmen zu eigenen Zeichen einen geringeren Platzbedarf sowie eine höhere Lese- und Schreibgeschwindigkeit. Durch eine Verwendung dieser Kürzel ist es jedoch nicht mehr möglich, die Schrift zur internationalen Verständigung zu benutzen.

Um in der *Brailleschrift* alle Zeichen des *American Standard Code for Information Interchange* (ASCII) darstellen zu können, wurden in der Vollschrift für Computeranwendungen jedem Zeichen zwei weitere Punkte angefügt [Hän]. Computerbraille besteht damit aus zwei vertikal angeordneten Reihen zu je vier Punkten. Es können nun 256 ( $2^8$ ) Zeichen dargestellt werden. *Computerbraille* wurde europaweit standardisiert.

## Anhang B: BITV – Anlage (Teil 1)

Dieses Dokument enthält keine Vorgaben zur grundlegenden Technik, die für die Bereitstellung von elektronischen Inhalten und Informationen verwendet wird (Server, Router, Netzwerkarchitekturen und Protokolle, Betriebssysteme usw.) und hinsichtlich der zu verwendenden Benutzeragenten. Die Anforderungen und Bedingungen beziehen sich allein auf die der Nutzerin/dem Nutzer angebotenen elektronischen Inhalte und Informationen.

Die Anforderungen und Bedingungen dieser Anlage basieren grundsätzlich auf den Zugänglichkeitsrichtlinien für Web-Inhalte 1.0 (Web Content Accessibility Guidelines 1.0) des World Wide Web Consortiums vom 5. Mai 1999.

Die in Teil 1 dieser Anlage enthaltenen, bei ihrem ersten Auftreten im Text durch Unterstreichung kenntlich gemachten, grundlegenden technischen Fachbegriffe sind in Teil 2 dieser Anlage (Glossar) erläutert.

### **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 originale Inhalt erfüllen.

*Bedingung 1.1:* Für jedes Nicht-Text-Element ist ein äquivalenter Text bereitzustellen. Dies gilt insbesondere für: Bilder, graphisch dargestellten Text einschließlich Symbolen, Regionen von Imagemaps, Animationen (z. B. animierte GIFs), Applets und programmierte Objekte, Zeichnungen, die auf der Verwendung von Zeichen und Symbolen des ASCII-Codes basieren (ASCII-Zeichnungen), Frames, Scripts, Bilder, die als Punkte in Listen verwendet werden, Platzhalter-Graphiken, graphische Buttons, Töne (abgespielt mit oder ohne Einwirkung des Benutzers), Audio-Dateien, die für sich allein stehen, Tonspuren von Videos und Videos.

*Bedingung 1.2:* Für jede aktive Region einer serverseitigen Imagemap sind redundante Texthyperlinks 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 (insbesondere Film oder Animation) sind äquivalente Alternativen (z.B. Untertitel oder Audio-beschreibungen der Videospur) 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 (insbesondere HTML) 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.

#### **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. Insofern auf eine automatische

Weiterleitung nicht verzichtet werden kann, ist der Server entsprechend zu konfigurieren.

### **Anforderung 8**

Die direkte Zugänglichkeit der in Internetangeboten eingebetteten Benutzerschnittstellen ist sicherzustellen.

*Bedingung 8.1:* Programmierte Elemente (insbesonder 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.

*Bedingung 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 geräteunabhängiger 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.

#### **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 originale Inhalt erfüllen.

*Bedingung 1.5:* Für jede aktive Region einer clientseitigen Imagemap sind redundante Texthyperlinks 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 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 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 (einschließlich solcher in clientseitigen Imagemaps), 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 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.

# Literaturverzeichnis

## 1 Monographien

- [And00] **Anderson, Richard; Birbeck, Mark; Kay, Michael; u.a. (2000):**  
XML professionell. Bonn : MITP-Verlag.
- [Arc01] **Arciniegas, Fabio (2001):**  
XML Developer's Guide.  
Erstellen effizienter XML-Anwendungen – Aktuellste Standards –  
Mobiles XML – Web-Publishing – Ergänzende Technologien.  
Poing : Franzis Verlag GmbH.
- [Bar03] **Bartel, Torsten (2003):**  
Die Verbesserung der Usability von Websites auf der Basis  
von Web Styleguides, Usability Testing und Logfile-Analysen.  
Berlin : Verlag für Wissenschaft und Kultur (WiKu-Verlag) Dr. Stein.
- [Bro99] **Brockhaus GmbH (Hrsg.) (1999):**  
Der Gesundheits-Brockhaus.  
5. Auflage. Mannheim : Brockhaus GmbH.
- [BTZZ00] **Büchner, Heino; Traub, Dennis; Zahradka, Rik; Zschau, Oliver (2000):**  
Web Content Management.  
Websites professionell betreiben.  
1. Auflage. Bonn : Galileo Press GmbH.
- [DKR96] **Degenhardt, Sven; Kalina, Ulrich; Rytlewski, Dirk (1996):**  
Der Einsatz des Computers bei blinden und sehbehinderten  
Schülern. Überblick, Stand und Perspektiven: Ergebnisse aus  
dem Modellversuch „Interaktive Informationstechniken für  
Sehgeschädigte in der informationstechnischen Bildung“ (IRIS).  
Hamburg : Hamburger Buchwerkstatt.



- [Hel05] Hellbusch, Jan Eric (2005):**  
Barrierefreies Webdesign. Praxishandbuch für  
Webgestaltung und graphische Programmoberflächen.  
1. Auflage. Heidelberg : dpunkt.verlag GmGH.
- [HL99] Horten, Sarah; Lynch, Patrick J. (1999):**  
Erfolgreiches Web-Design.  
München : Humboldt Taschenbuchverlag Jacobi KG.
- [KM83] Kreft, Dieter (Hrsg.); Mielenz, Ingrid (Hrsg.) (1983):**  
Wörterbuch Soziale Arbeit.  
2. Auflage. Weinheim; Basel : Beltz.
- [Lan01] Lankau, Ralf (2001):**  
Webdesign und -publishing. Projektmanagement für Websites.  
2. Auflage. München; Wien : Carl Hanser Verlag.
- [LT99] Liebrecht, Andrea; Theiß-Klee, Heidi (1999):**  
Sehen Sehbehinderung Blindheit.  
Informationen und Unterrichtshilfen für allgemeine Schulen.  
Frankfurt : DER PARITÄTISCHE, Landesverband Hessen e. V..
- [MM02] Manhartsberger, Martina; Musil, Sabine (2002):**  
Web Usability. Das Prinzip des Vertrauens. Galileo Design.  
Bonn : Galileo Press GmbH.
- [Nie99] Niederst, Jennifer (1999):**  
Web Design in a Nutshell. Sebastopol : O'REILLY.
- [Nie00] Nielsen, Jakob (2000) :**  
Jakob Nielsen's Web Design. Erfolg des Einfachen.  
München : Markt+Technik Verlag.
- [Oes01] Oestereich, Bernd (2001):**  
Objektorientierte Softwareentwicklung.  
Analyse und Design mit der Unified Modeling Language.  
5. Auflage. München; Wien : Oldenbourg.
- [Pus01] Puscher, Frank (2001):**  
Das Usability-Prinzip. Wege zur benutzerfreundlichen Website.  
1. Auflage. Heidelberg : dpunkt.verlag GmGH.

- [RR01] **Rothfuss, Gunther (Hrsg.); Ried, Christian (Hrsg.) (2001):**  
Content-Management mit XML.  
Grundlagen und Anwendungen.  
Berlin u.a. : Springer-Verlag.
- [SW04] **Schneider, Uwe (Hrsg.); Werner, Dieter (Hrsg.) (2004):**  
Taschenbuch der Informatik. 5.Auflage.  
München, Wien : Fachbuchverlag Leipzig im Carl Hanser Verlag.
- [The99] **Theiß-Klee, Heidi (1999):**  
Punktschrift für Anfänger.  
Wie erlerne ich als Sehender die Blindenvollschrift.  
Marburg/Lahn : Deutsche Blindenstudienanstalt e.V..
- [Wir04] **Wirth, Thomas (2004):**  
Missing Links. Über gutes Webdesign.  
2. Auflage. München, Wien : Carl Hanser Verlag.

## 2 Sammelbände

- [BvG02] **Beier, Markus (Hrsg.); von Gizycki, Vittoria (Hrsg.) (2002):**  
Usability. Nutzerfreundliches Webdesign.  
Berlin u.a. : Springer-Verlag.

## 3 Aufsätze

- [vGi02] **von Gizycki, Vittoria (2002):**  
Usability - nutzerfreundliches Web-Design. Seite 1-17  
in: Beier, Markus (Hrsg.); von Gizycki, Vittoria (Hrsg.) (2002):  
Usability. Nutzerfreundliches Webdesign. Berlin u.a. : Springer-Verlag.
- [WvG02] **Weiland, Sascha; von Gizycki, Vittoria (2002):**  
Wahrnehmungspsychologische Erkenntnisse im Web-Design.  
Seite 33-41  
in: Beier, Markus (Hrsg.); von Gizycki, Vittoria (Hrsg.) (2002):  
Usability. Nutzerfreundliches Webdesign. Berlin u.a. : Springer-Verlag.

## 4 onlinequellen

**[Beh] Behindertenbeauftragter:**

Begründung zur 'Verordnung zur Schaffung barrierefreier Informationstechnik nach dem Behindertengleichstellungsgesetz - BITV'.  
<http://www.behindertenbeauftragter.de/files/1027946170.39/Begrueendung.pdf>. 28.09.2004.

**[BKB] Bouch, Anna; Kuchinsky, Allan; Bhatti, Nina:**

Quality is in the Eye of the Beholder: Meeting Users' Requirements for InternetQuality of Service. HLP-2000-4. January, 2000.  
<http://www.hpl.hp.com/techreports/2000/HPL-2000-4.pdf>.  
14.11.2004.

**[BS] Bernard, Michael; Sheshadri, Ashwin:**

Preliminary Examination of Global Expectations of Users' Mental Models for E-Commerce Web Layouts.  
[http://psychology.wichita.edu/surl/usabilitynews/62/web\\_object\\_international.htm](http://psychology.wichita.edu/surl/usabilitynews/62/web_object_international.htm). 03.11.2004.

**[BuD] Bundesregierung Deutschland (Hrsg.):**

Gesetz zur Gleichstellung behinderter Menschen.  
Behindertengleichstellungsgesetz - BGG.  
<http://www.behindertenbeauftragter.de/download/gleichstellungsgesetz.pdf>. 01.09.2004.

**[Bul] Bundesministerium des Innern:**

Verordnung zur Schaffung barrierefreier Informationstechnik nach dem Behindertengleichstellungsgesetz.  
Barrierefreie Informationstechnik-Verordnung - BITV.  
[http://www.behindertenbeauftragter.de/gesetzgebung/behinderten\\_gleichstellungsgesetz/rechtsverordnung/rvo11bgg](http://www.behindertenbeauftragter.de/gesetzgebung/behinderten_gleichstellungsgesetz/rechtsverordnung/rvo11bgg). 01.09.2004.

**[Cla] Clark, Joe:**

Building Accessible Websites.  
[www.joeclark.org/book](http://www.joeclark.org/book). 21.12.2004.

**[Deu] Deutscher Blinden- und Sehbehindertenverband e.V.:**

Infothek. <http://www.dbsv.org/infothek/infothek.html>. 25.09.2004.  
Statistisches. <http://www.dbsv.org/infothek/Statistik.html>. 27.09.2004.

- [Hän] Hänel, Matthias:  
ACCESSIBILITY-Page.  
<http://www.matthias-haenel.de/waautoref.html>. 06.09.2004.  
<http://www.matthias-haenel.de/wabraille.html>. 06.09.2004.
- [Jec] Jeckle:  
Skriptum zur Vorlesung XML.  
<http://www.jeckle.de/vorlesung/xml/script.html#DOM>. 20.03.2005.
- [PHP] PHP:  
PHP-Manual. <http://de.php.net/manual/en/preface.php>. 30.03.2005.
- [Rai] Raissakis, Petra:  
Eine Homepage über den Alltag blinder und sehbehinderter Menschen.  
<http://www.anderssehen.at>. 13.04.2005.
- [Röb] Röbbke-Doerr, Peter:  
Braille & Co. PC-Hilfen für Blinde.  
[http://www.uniability.ch/Part.ct\\_Blindenhilfe/BlindenHilfe\\_ct.shtml](http://www.uniability.ch/Part.ct_Blindenhilfe/BlindenHilfe_ct.shtml).  
23.12.2004.
- [SIL] SevenOne Interactive GmbH (Hrsg.); IP NEWMEDIA GmbH (Hrsg.);  
LYCOS Europe GmbH (Hrsg.):  
@facts basics. 1. Quartal 2004.  
[http://www.atfacts.de/001/pdf\\_studies/atfacts\\_q1\\_2004.pdf](http://www.atfacts.de/001/pdf_studies/atfacts_q1_2004.pdf).  
07.11.2004.
- [Val] World Wide Web Consortium:  
Markup Validation Service.  
<http://validator.w3.org>. 24.04.2005.
- [Wat] Watchfire:  
Bobby. <http://bobby.watchfire.com>. 23.04.2005.
- [W3C] World Wide Web Consortium:  
Das World Wide Web Consortium (W3C) ...in sieben Punkten.  
<http://www.w3c.de/sieben.html>. 06.09.2004.  
About the World Wide Web Consortium (W3C).  
<http://www.w3.org/Consortium/>. 06.09.2004.  
World Wide Web Consortium (W3C) Members.  
<http://www.w3.org/Consortium/Member/List>. 11.12.2004.

- [WCAG1] World Wide Web Consortium:**  
Zugänglichkeitsrichtlinien für Web-Inhalte 1.0.  
<http://www.w3c.de/Trans/WAI/webinhalt.html>. 06.09.2004.
- [WCAG2] World Wide Web Consortium:**  
Web Content Accessibility Guidelines 2.0  
<http://www.w3.org/TR/2004/WD-WCAG20-20040730>. 06.09.2004.
- [Wil] Williams, Thomas R.:**  
Guidelines for Designing and Evaluating the  
Display of Information on the Web.  
<http://www.techcomm-online.com/issues/v47n3/full/0411.html>.  
11.11.2004.
- [XML] World Wide Web Consortium:**  
Extensible Markup Language (XML) 1.0 (Third Edition).  
<http://www.w3.org/TR/REC-xml/>. 15.03.2005.
- [XMLS] World Wide Web Consortium:**  
XML Schema Teil 0: Einführung. Deutsche Übersetzung.  
<http://www.edition-w3c.de/TR/2001/REC-xmlschema-0-20010502/>. 18.03.2005.

# Erklärung

Hiermit versichere ich, Daniela Bonnert, dass ich die vorliegende Diplomarbeit zu dem Thema

*„Erstellen eines für Sehbehinderte optimierten Werkzeugs zur Erstellung von barrierefreien Internetangeboten“*

selbstständig verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt und mich auch sonst keiner unerlaubten Hilfe bedient habe.

Berlin, den 25.04.2005

---