

# **Dokumentation der betrieblichen Projektarbeit**

Von:

Norman Scheffler

zur Erlangung des Abschlusses als Informatikkaufmann

Komplettierung vorhandener PHP- und Datenbank-Module  
zu einem Web-Content-Management-System  
für den firmeninternen Einsatz und Vertrieb über die Marburger Bürobedarf GmbH

Marburger Bürobedarf GmbH  
Sybelstraße 6  
35037 Marburg  
Telefon 0 64 21 / 1 21 39  
Projektbetreuer: Herr Michael Stock  
Durchführungszeitraum: März 2006 - April 2006

# Inhaltsverzeichnis

1. Beschreibung des Projektauftrages .....	1
1.1 Die Ausgangssituation .....	1
1.2 Gründe für den Projektauftrag .....	1
1.3 Projektziele .....	2
1.4 Prozessschnittstellen .....	2
1.4.1 Persönliche Ansprechpartner.....	2
1.4.2 Nutzung existierender Programmmodule.....	2
2. Projektplanung .....	2
2.1. Zeit- und Ablaufplanung .....	2
2.2 Ressourcenplanung .....	3
2.3. Kostenplanung.....	4
3. Projektdurchführung .....	4
3.1 Analysephase .....	4
3.1.1 Ermittlung des Ist-Zustandes .....	4
3.1.2 Definition des Soll-Zustandes/Erstellung des Lastenhefts.....	4
3.1.3 Erstellen des Pflichtenhefts und der Checkliste zur Qualitätssicherung ...	5
3.2 Entwurfsphase.....	5
3.2.1 Entwurf einer Rechteverwaltung .....	5
3.2.2 Entwurf der Style-Verwaltung .....	5
3.2.3 Programmmodul zum Festlegen der Startseite.....	5
3.2.4 Planung von Modul-Schnittstellen.....	5
3.3 Implementierungsphase .....	6
3.3.1 Erweiterung der CMS-Datenbank.....	6
3.3.2 Programmierung der Web-Oberflächen und PHP-Funktionen .....	6
3.3.3 Anpassungen des Terminkalenders.....	7
4. Testphase .....	7
4.1 Globaler Funktionstest.....	7
4.2 Softwaretest anhand erstellter Checkliste zur Qualitätssicherung.....	7
4.3 Nachbearbeitung .....	7
5. Übergabephase.....	8
5.1 Installation der Anwendung auf den Zielsystemen .....	8
5.2 Schulungsmaßnahmen .....	8
5.3 Dokumentation.....	8
5.3.1 Erstellung der Installationsanleitung für das Web-CMS .....	8
5.3.2 Amortisationsrechnung bei betriebsinterner Nutzung .....	8
5.3.3 Analyse/Kalkulation von Möglichkeiten zum Softwarevertrieb.....	8
6. Projektauswertung .....	9
Anlagenverzeichnis.....	10
Quellenverzeichnis .....	10

## **1. Beschreibung des Projektauftrages**

### **1.1 Die Ausgangssituation**

Der ausbildende Betrieb ist die Marburger Bürobedarf GmbH (MBB), ein zur Deutschen Blindenstudienanstalt Marburg e. V., gehörendes Unternehmen, das sehbehinderte und blinde Menschen zu Informatikkaufleuten ausbildet. Die MBB wird als Übungsfirma betrieben, die virtuell mit Produkten aus dem Bereich Schreibwaren, Büromöbel und -technik sowie Hard- und Software handelt. Dieser Handel erfolgt über den Zentralen Übungsfirmenring (ZÜF), dem auch die Marburger Bürobedarf GmbH angehört. Derzeit beschäftigt die MBB 23 Angestellte, davon 4 Ausbilder und 19 in den Abteilungen Einkauf, Verkauf, Lohn- und Finanzbuchhaltung sowie Softwareentwicklung tätige Auszubildende. Die Bearbeitung des Projekts findet in der vier Mitarbeiter umfassenden Abteilung Softwareentwicklung statt.

### **1.2 Gründe für den Projektauftrag**

Im firmeninternen Netz der Marburger Bürobedarf GmbH steht ein Windows 2003 Domänencontroller zur Verfügung, auf dem zusätzlich die Apache-Webserversoftware installiert wurde. Dieser Rechner wird von den Angestellten des Ausbildungsbetriebes für ein webbasiertes Informationsportal genutzt. Komponenten dieses Intranets sind derzeit ein auf eine Datenbank zugreifender Küchenplan, ein Terminkalender, dessen Einträge in Textdateien abgespeichert werden, ein "schwarzes Brett" sowie nur temporär relevante Inhalte wie einen Tippplan für Fußballspiele.

Von Nachteil in der Handhabung dieses Informationsangebots sind insbesondere die folgenden Argumente:

- Die allermeisten Veränderungen/Aktualisierungen an Inhalten setzen zumindest Kenntnisse des Redakteurs im Bereich der Webprogrammierung voraus.
- Das Einbinden neuer Komponenten in die Web-Struktur erfordert Kenntnis im Bereich der Referenzierung von Dokumenten und ein Verändern existierender Quellcodes.
- Aufgrund vorhandener Zugriffsrechte auf das Rootverzeichnis des Webserver ist im Betrieb jedermann potentiell imstande, die Dateien des Intranets zu verändern.
- Alle eingetragenen Termine sowie alle Publikationen im Intranet sind für jedermann zugänglich, eine diesbezügliche Rechtebeschränkung ist derzeit nicht realisiert.
- Selbst bei vorhandenen Kenntnissen im Bereich der Webprogrammierung ist das Einbinden von Inhalten dennoch mit erheblichem Zeitaufwand verbunden.
- Eingefügte Inhalte können oft bezüglich ihres Layouts den Ansprüchen an ein Corporate Design nicht entsprechen.

Folge dieser genannten Punkte ist eine mangelhafte Aktualität, ein durchaus begrenzter Informationsgehalt und eine daraus resultierende spärliche Nutzung des Webangebots durch die Angestellten der MBB.

Tests frei verfügbarer Software für Web-Content-Management-Systeme wie EZ Publish oder PHP-Nuke verliefen aufgrund ihres sehr hohen Konfigurationsaufwandes bzw. ihrer mangelnden Zugänglichkeit für Benutzer von so genannten Screen-readern nicht zufriedenstellend.

Angesichts dieser Problematik und einiger Kundenanfragen bezüglich einer Intranet-Lösung erteilte mir die Geschäftsleitung der Marburger Bürobedarf GmbH den Auftrag zur Entwicklung eines Programms zum einfachen Publizieren und Verwalten von firmenrelevanten Inhalten auf Basis eines Webservers.

### **1.3 Projektziele**

Intention dieser Aufgabe ist die Entwicklung einer Anwendung, die

- jedermann im Ausbildungsbetrieb die redaktionelle Nutzung des Intranets ohne Programmierkenntnisse ermöglicht,
- den zeitlichen Aufwand für betriebsinterne Publikationen deutlich verringert,
- die Verfügbarkeit von betriebsrelevanten Informationen erhöht und den Zugriff auf diese erleichtert,
- den Zugang zu sensiblen Dokumenten auf Benutzergruppen einschränkt,
- für jedermann einen personenspezifischen und nur für diesen zugänglichen Terminkalender bereit hält,
- sowie jeweils den Verfasser von Beiträgen und Terminen mit Datum der Erstellung ausweist.

### **1.4 Prozessschnittstellen**

#### **1.4.1 Persönliche Ansprechpartner**

Informationen über die bisherige Nutzung und Bedienbarkeit des Intranets der Marburger Bürobedarf GmbH konnten mir sowohl alle Mit-Auszubildenden, als auch die Ausbilder im Betrieb liefern. Insbesondere die Azubis der jüngeren Jahrgänge zeigten sich diesbezüglich aufgrund des mangelnden bisherigen Informationsgehalts eher uninteressiert an der bisherigen Lösung. Mitglieder des mittleren Ausbildungsjahrganges äußerten dagegen den Wunsch nach einer komfortablen Möglichkeit, Berufsschulinhalte zur Prüfungsvorbereitung in einem zentralen Wissensspeicher verwalten zu können.

Informationen zur geplanten Serverumgebung und zu den gewünschten Programmfunktionen und -eigenschaften lieferte mir Herr Michael Stock, stellvertretender Geschäftsleiter der MBB, in 2 Einzelgesprächen.

Angaben über die durchschnittlichen Kosten für eine Arbeitsstunde im Betrieb konnte mir Herr Rainer Datzer, Geschäftsführer der MBB machen.

#### **1.4.2 Nutzung existierender Programmmodule**

Die Anwendung baut funktionell auf bereits während meines Praktikums bei der MarMed GmbH Cölbe programmierten Teilkomponenten zur Verwaltung von Intranet-Inhalten auf. In einem Pflichtenheft (s. Anlage C) ist detailliert dargestellt, welche Tätigkeiten zur Erreichung der Projektziele zusätzlich vonnöten waren.

## ***2. Projektplanung***

### **2.1. Zeit- und Ablaufplanung**

In der Zeit- und Ablaufplanung wird eine tägliche Projektarbeitszeit von 2 bis 3 Stunden angenommen, da neben dieser Tätigkeit noch weitere Aufgaben im Ausbildungsbetrieb zu erledigen sind. 2 von 5 Arbeitstagen stehen pro Woche aufgrund Berufsschulunterrichts nicht zur Verfügung. Daraus resultiert unter Einberechnung von

Pufferzeiten eine Gesamtlaufzeit des Projektes, die inklusive Berufschulzeit zwischen 19,5 und 29 Arbeitstagen liegt.

Beginn ist der Tag der Genehmigung meines Antrags für die betriebliche Projektarbeit durch die IHK Hessen am 28.02.2006; das Ende ist vom Auftraggeber für den 4. Mai festgesetzt worden.

Die einzelnen Phasen gliedern sich in verschiedene Aktivitäten. In der unten stehenden Tabelle werden diese aufgeführt, einer Projektphase zugeordnet und die geplante Zeit angegeben.

Projektphase	Aufgabenbeschreibung	Soll-Stunden
<b>Analyse</b>	Ermittlung des Ist-Zustandes	1:00 h
	Ermittlung des Soll-Zustandes	1:00 h
	Erstellung des Lastenhefts	2:00 h
	Erstellung des Pflichtenhefts	2:00 h
	Erstellen Checkliste zur Qualitätssicherung	1:00 h
<b>Entwurf</b>	Konzeptioneller Entwurf einer Rechteverwaltung	2:00 h
	Entwurf einer Style-Verwaltung	0:30 h
	Planung von Modul-Schnittstellen	0:30 h
<b>Implementierung</b>	Anlegen von Datenbankkomponenten für Rechteverwaltung	2:00 h
	Datenbankkomponenten für die Style-Verwaltung	0:30 h
	PHP-Scripte für Rechte- und Style-Verwaltung	5:00 h
	Einbinden neuer Module in bisherige Programmstruktur	1:00 h
	Anpassen von Programmmodulen an Vorgaben aus Lastenheft	2:00 h
<b>Test</b>	Globaler Funktionstest	1:00 h
	Maßnahmen zur Qualitätssicherung	2:00 h
	Testdokumentation/Prüfprotokoll	2:00 h
<b>Übergabe</b>	Installation auf Zielsystem	0:30 h
	Einweisung	1:00 h
	Dokumentation	8:00 h
<b>Gesamtzeit</b>		<b>35:00 h</b>

## 2.2 Ressourcenplanung

Das Projekt wird an einem Arbeitsplatzrechner mit dem Betriebssystem Windows XP Professional durchgeführt.

Zum Erstellen der PHP-Scriptdateien und SQL-Anweisungen steht der standardmäßig installierte Windows-Editor zur Verfügung, für lokale Tests der Anwendung wurden auf dem Arbeitsplatzrechner der Interbase-Datenbankserver für Windows Version 6.0 der Firma Borland und das XAMPP-Softwarepaket Version 1.51 mit dem hierin enthaltenen Apache-Webserver installiert.

Der Zugriff auf die Scriptdateien erfolgt über den Internet Explorer 6 und vergleichshalber über den Mozilla Firefox 1.0.6.

Die Projektdokumentation erfolgt unter Verwendung der Programme MS Word 2003, MS Excel 2003 und MS Access 2003.

## 2.3. Kostenplanung

Die Kosten des Projekts setzen sich aus der Anzahl der aufzuwendenden Stunden und dem Kostensatz für einen Mitarbeiter und einer Arbeitsplatzausstattung pro Stunde zusammen.

durchschnittliche jährlicher Bruttoverdienst:	7.200,00 €
+ 25 % Arbeitgeberanteil für Sozialabgaben:	9.000,00 €
+ 15 % *) Handlungskostenzuschlag:	10.350,00 €
Anzahl Arbeitstage/Jahr abzüglich 26 Urlaubstage:	226
durchschnittliche wöchentliche betriebliche Arbeitszeit (38:30 h Wochenarbeitszeit - 14:00 h Berufsschulzeit):	24:30 h
durchschnittliche tägliche betriebliche Arbeitszeit:	4:54 h
durchschnittliche Kosten pro Arbeitsstunde:	9,35 €

---

<b>Projektgesamtkosten bei einer Projektdauer von 35 h:</b>	<b>327,12 EUR</b>
---	-------------------

---

\*) Angaben durch Herrn Rainer Datzer, Geschäftsführer MBB

## 3. Projektdurchführung

### 3.1 Analysephase

#### 3.1.1 Ermittlung des Ist-Zustandes

In mehreren Gesprächen mit den Mitarbeitern der MBB wurden der bisherige Publikationsaufwand und der Nutzungsgrad der bisherigen Intranet-Lösung ermittelt. Geäußerte Kritikpunkte habe ich gesammelt, um sie in meiner Projektarbeit berücksichtigen zu können. In weiteren Gesprächen mit meinem Auftraggeber, Herrn Michael Stock erfolgte zunächst eine Analyse der Ausgangssituation verbunden mit der Vorstellung der aus meiner Praktikumsstätigkeit stammenden Programmmodule zum Verwalten von Web-Inhalten.

#### 3.1.2 Definition des Soll-Zustandes/Erstellung des Lastenhefts

Im Anschluss hieran wurden die Projektziele und die genauen, fachlichen Anforderungen an ein Web-CMS festgelegt.

Auf Grundlage dieser Gespräche hielt ich diese Anforderungen in einem in Namen der Geschäftsleitung erstellten Lastenheft fest, dass ich zur Abstimmung der inhaltlichen Korrektheit und Vollständigkeit Herrn Stock zur Abnahme vorlegte. Dieses Dokument befindet sich im Anhang (Anlage B) an die Projektdokumentation.

### 3.1.3 Erstellen des Pflichtenhefts und der Checkliste zur Qualitätssicherung

Zum Zwecke der Abgrenzung der Projektstätigkeit zu dem bereits während meines Praktikums Geleisteten erstellte ich zudem ein Pflichtenheft. Hierin sind die konkreten Realisierungsvorgaben aufgrund der Umsetzung des vom Auftraggeber vorgegebenen Lastenheftes festgehalten. Im Lastenheft gestellte Qualitätsanforderungen konkretisierte ich in einer detaillierten Checkliste für Web-Standards und CMS-Funktionen. Beide genannten Dokumente befinden sich im Anhang an diese Projektdokumentation (Anlagen C und D).

## 3.2 Entwurfsphase

### 3.2.1 Entwurf einer Rechteverwaltung

In dieser Phase der Softwareentwicklung stellte ich die für die Verwirklichung der im Pflichtenheft konkretisierten Vorgaben an eine umfassende Rechteverwaltung notwendigen Komponenten zusammen.

Datenbankvoraussetzungen	Web-/Scriptvoraussetzungen
Tabelle Nutzer	⇔ Webformular zum Anlegen u. Verwalten von Nutzern
Tabelle Gruppen	⇔ Webformular zum Anlegen u. Verwalten von Gruppen
Tabelle Gruppenmitgliedschaften	⇔ Webformular zum Verwalten von Gruppenmitgliedschaften
Tabelle seitenspezifische Schreibrechte	⇔ Webformular zum Verwalten seitenspez. Schreibrechte
Tabelle seitenspezifische Leserechte	⇔ Webformular zum Verwalten seitenspez. Leserechte
Prozedur zur Überprüfung seitenspezifischer Schreibrechte	⇔ PHP-Funktion zur Auswertung seitenspezifischer Schreibrechte
Prozedur zur Überprüfung seitenspezifischer Leserechte	⇔ PHP-Funktion zur Auswertung seitenspezifischer Leserechte

### 3.2.2 Entwurf der Style-Verwaltung

Als Voraussetzung für zusätzliche elementbezogene Layoutmöglichkeiten sah ich auf Datenbank-Ebene lediglich eine Tabelle für die einzelnen Style-Klassen und als zugehörige Web-Komponente eine Oberfläche zum Anlegen, Löschen und Ändern von diesen Klassen an.

### 3.2.3 Programmmodul zum Festlegen der Startseite

Zum Festlegen des grundlegenden Navigationslayouts, des Titels des Webangebots und einer CMS-Startseite sah ich auf Datenbank-Ebene wiederum lediglich eine Tabelle vor, in die der angemeldete Administrator über eine Weboberfläche die diesbezüglichen Informationen ablegen kann.

### 3.2.4 Planung von Modul-Schnittstellen

Sowohl der Schutz von Inhaltsseiten vor unbefugtem Lese- bzw. Schreibzugriff, als auch der Schutz des gesamten Administrationsbereichs des Web-CMS bedingt Verbindungen zu den neuen Modulen zur Rechteverwaltung. Ich kam zu dem Schluss, dass die benötigten Schnittstellen effektiv, ohne jedoch allzu in den Quellcode bereits

existierender Programmmodule eingreifen zu müssen, durch das Einbinden von Include-Dateien zu realisieren waren.

### 3.3 Implementierungsphase

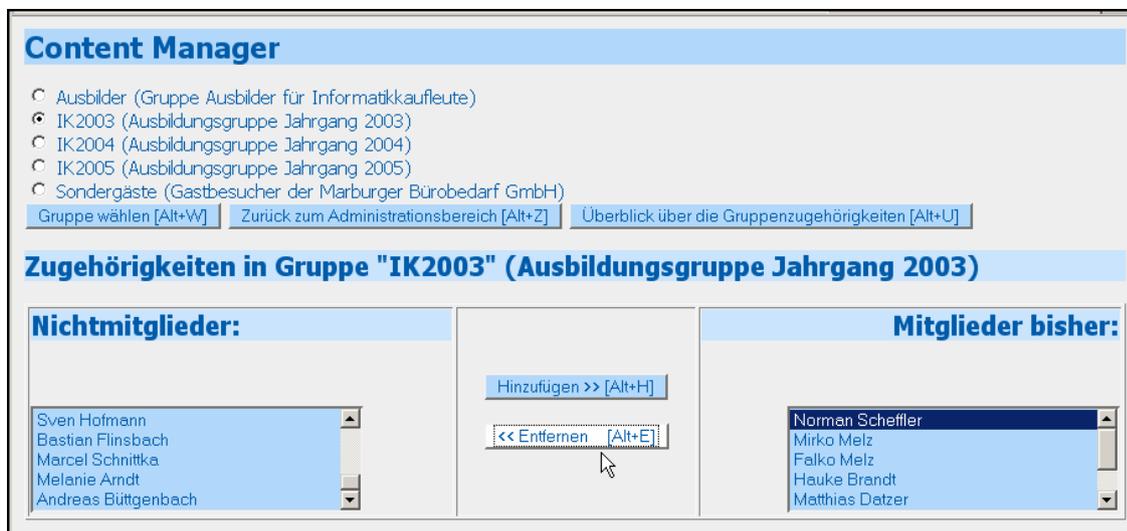
#### 3.3.1 Erweiterung der CMS-Datenbank

Unter Verwendung der IBConsole, einem zum Borland-Interbase-Datenbankserver gehörendem Tool zur direkten Eingabe von SQL-Befehlen ergänzte ich die bisherige CMS-Datenbankstruktur um die für die im Lastenheft geforderten Rechte- und Styleverwaltungen notwendigen Tabellen und Prozeduren.

Bereits vorhandene Tabellen mussten zum Teil um weitere Attribute bzw. Fremdschlüsselspalten ergänzt werden. Sämtliche neuen Tabellen und Veränderungen sind im in der Anlage I befindlichen CMS-Datenbankdiagramm farblich gekennzeichnet.

#### 3.3.2 Programmierung der Web-Oberflächen und PHP-Funktionen

Unter Verwendung des Windows-Editors erstellte ich die für die Rechte- und Layoutverwaltungen unter 3.2.1 bis 3.2.3 genannte notwendige Anzahl von Weboberflächen in Form von PHP-Scriptdateien. Das Auslagern immer wiederkehrender Codebestandteile in so genannte Include-Dateien vereinfachte die Programmierarbeit und birgt den Vorteil, Änderungen für diese Programmteile nun zentral vornehmen zu können. Die nachfolgende Abbildung zeigt beispielhaft für diese Arbeit die Weboberfläche zum Verwalten von Gruppenmitgliedschaften, der Quellcode dieses Programmmoduls befindet sich exemplarisch im Anhang zu dieser Dokumentation (s. Anlage J).



Durch Veränderung vorhandener PHP-Scriptdateien schuf ich Schnittstellen zu den neu erstellten Komponenten. Die nachfolgende Abbildung stellt als Beispiel den Dialog zum Bestimmen von gruppenbezogenen Leserechten dar, welcher direkt aus dem Modul zum Editieren einer bestimmten Webseite aufgerufen werden kann.

**Leserechte ändern**

Sie können durch Hinzufügen von Gruppen das Leserecht an der ausgewählten Seite ändern und auf selbige einschränken. Wird keine Gruppe gewählt, so genießen alle Nutzer Leserechte an dieser Seite.

Bitte entsprechende Gruppen markieren!

- Ausbilder - Gruppe Ausbilder für Informatikkaufleute
- IK2003 - Ausbildungsgruppe Jahrgang 2003
- IK2004 - Ausbildungsgruppe Jahrgang 2004
- IK2005 - Ausbildungsgruppe Jahrgang 2005
- Sondergäste - Gastbesucher der Marburger Bürobedarf GmbH

Analog zu diesem Webformular habe ich den Dialog zum Verwalten von gruppenspezifischen Schreibrechten konzipiert.

### 3.3.3 Anpassungen des Terminkalenders

Aufgrund der Forderungen aus dem Lastenheft passte ich die PHP-Scripte des Terminkalenders dahingehend an, dass die Anzeige von Terminen auf der Session-ID des angemeldeten Nutzers basiert. Diese ID wird dann zur Überprüfung von Gruppenzugehörigkeiten verwandt, so dass zwischen allgemeinen, persönlichen und sogar gruppenspezifischen Terminen unterschieden wird. Beim Eintragen von Terminen wird die Nutzer-ID zur Identifizierung des Erstellers neben den anderen Textbestandteilen in der Datenbank abgespeichert, später kann daher nicht nur der "Empfänger", sondern auch der Ersteller eines Eintrags zum Löschen auf diesen Zugreifen.

## 4. Testphase

### 4.1 Globaler Funktionstest

Nach der Implementierung sämtlicher neuer Programmmodule und Abschluss aller notwendig gewordenen Veränderungen an den bestehenden CMS-Komponenten testete ich das gesamte System auf seine Funktionen und Fehleranfälligkeit. Aufgrund der Tatsache, dass ich bereits während der Programmierung der Datenbank- und Webkomponenten permanent Testsituationen simulierte, wurden in dieser Phase keinerlei Korrekturen nötig.

### 4.2 Softwaretest anhand erstellter Checkliste zur Qualitätssicherung

Im Anschluss an diesen globalen Funktionstest überprüfte ich das Web-CMS in Bezug auf die in der Checkliste zur Qualitätssicherung definierten Anforderungen. Hierin dokumentierte ich zudem, ob und wie die von mir erstellte Software die genannten Vorgaben erfüllt.

### 4.3 Nachbearbeitung

Aufgrund genannter Checkliste korrigierte ich die gesamte Anwendung um die folgenden Eigenschaften und Funktionen:

- eine PHP-Funktion zum automatischen Erzeugen von Formulschaltern mit Shortcut
- eine zusätzliche CSS-Datei für ein Print-Layout
- eine eigene 404-Fehlerseite mit zugehöriger HTACCESS-Datei

## **5. Übergabephase**

### **5.1 Installation der Anwendung auf den Zielsystemen**

Nach nochmaliger Rücksprache mit Herrn Stock kamen wir überein, dass die Anwendung künftig auf 2 Systemen betrieben werden soll. Die auf den Betrieb eines Webservers basierenden PHP-Scriptdateien installierte ich im Root-Verzeichnis desjenigen Webservers, der schon bisher zur Bereitstellung des firmeninternen Intranets genutzt wurde. Der erforderliche Interbase-Datenbankserver wurde zusätzlich auf einem bisher ausschließlich als Druck- und Backup-Server mit Windows 2003-Server-Betriebssystem installiert. Bei der Konfiguration der Anwendung traten keine erheblichen Schwierigkeiten auf, es musste lediglich die den Datenbankzugriff initiiierende PHP-Scriptdatei auf den Rechner des Datenbankservers angepasst werden.

### **5.2 Schulungsmaßnahmen**

Unmittelbar nach der Installation des Web-CMS informierte ich die Kollegen und Ausbilder über die neue Software, legte alle Mitarbeiter der MBB als Nutzer in der CMS-Nutzerdatenbank an und fasste diese entsprechend ihrer Zugehörigkeit zu einen Ausbildungsjahrgang bzw. zum Ausbilderteam zu Gruppen zusammen. Anschließend verschaffte ich den anwesenden Kollegen ein Bild über die Funktionsweise der Software.

### **5.3 Dokumentation**

Neben der Erstellung dieser Projektdokumentation, dem in Auftrag der Geschäftsleitung der MBB erstellten Lastenheft und des hieraus resultierenden Pflichtenheftes hatte ich zur Erfüllung des Projektauftrages die für die Installation des CMS notwendigen Schritte sowie die Amortisation der Software bei verschiedenen Möglichkeiten ihrer Nutzung zu dokumentieren.

#### **5.3.1 Erstellung der Installationsanleitung für das Web-CMS**

Zur Erstellung der Installationsanleitung dokumentierte ich diejenigen Schritte, die zur unter 5.1 beschriebenen Installation der Software auf ihren betriebsinternen Zielsystemen nötig waren und ergänzte diese um die jeweiligen betriebssystemabhängigen Handlungsalternativen. Zuvor legte ich eine übersichtliche Ordnerstruktur mit den Komponenten der Software und Linux- bzw. Windows-Versionen des Apache-Webservers und des Interbase-Datenbankservers an, welche ich auf eine CD-ROM kopierte.

#### **5.3.2 Amortisationsrechnung bei betriebsinterner Nutzung**

Nach der in der Anlage F zu dieser Projektdokumentation dargelegten Amortisationsrechnung werden die angefallenen Projektkosten binnen eines Jahres bereits bei einer monatlichen Zeitersparnis von 5 Minuten pro Mitarbeiter durch die Nutzung des neuen Web-CMS ausgeglichen.

#### **5.3.3 Analyse/Kalkulation von Möglichkeiten zum Softwarevertrieb**

Aufgrund des in der Anlage G zu dieser Projektdokumentation vorgenommenen Vergleichs von verschiedenen Möglichkeiten zur wirtschaftlichen Verwertung der von mir erstellten Software und zugehörigen Kalkulationsrechnungen zum Erreichen der Ge-

winnschwelle erarbeitete ich eine begründete Vertriebsempfehlung für die MBB GmbH. Inhalt dieser Empfehlung ist der kostenlose Vertrieb des CMS mit integriertem MBB-Produktkatalog, um hierdurch zu erwartende Umsatzsteigerungen zu erwirken.

## ***6. Projektauswertung***

Abschließend kann ich festhalten, dass alle im Lastenheft zu diesem Projektauftrag definierten Vorgaben an das Content-Management-System erfolgreich in die Software umgesetzt werden konnten. Den Mitarbeitern der MBB steht von nun an ein Programm zur Verfügung, das ihnen das Publizieren von betriebsinternen Inhalten deutlich vereinfacht und den hierfür bisher notwendigen Zeitaufwand erheblich verringert. Mit der zu erwartenden stärkeren Nutzung des Intranets dürften weitere positive Effekte in Bezug auf Informationsfluss und Zeitmanagement verbunden sein. Bei einem wie von mir empfohlenen kostenlosen Vertrieb der Software mit integriertem MBB-Warenkatalog kann zudem künftig von einer gesteigerten Nachfrage nach Artikeln des Bürobedarfs an die MBB GmbH ausgegangen werden.

Zeitliche Vorgaben an das Projekt konnten insgesamt eingehalten werden, die Projektgesamtkosten übersteigen somit nicht die kalkulierten Kosten.

Sehr zufrieden bin ich in Bezug auf die Kommunikation mit den Kollegen und Ausbildern, die Ermittlung von Anforderungen an die Software gestaltete sich unkompliziert, für die jeweiligen Kalkulationen notwendige Auskünfte aus dem Bereich der Lohn- und Finanzbuchhaltung wurden bereitwillig zur Verfügung gestellt.

Über die Anforderungen aus dem Lastenheft hinaus implementierte ich für einen besseren Überblick über die jeweiligen Rechtegruppen eine Anzeige aller Nutzer mit Namen und Foto, gestaffelt nach Gruppenzugehörigkeit.

Insgesamt haben sich meine Vorgehensweise und die ungefähre zeitliche Einteilung meiner Arbeitsschritte als richtig erwiesen, mit dem Projektverlauf und den Ergebnissen meiner Arbeit bin ich sehr zufrieden.

Marburg, den

---

Unterschrift

## Anlagenverzeichnis

- A. Einverständniserklärung MarMed GmbH zur Software-Weiterentwicklung
- B. Lastenheft zum Projekt "Web-CMS"
- C. Pflichtenheft zum Projekt "Web-CMS"
- D. Programmablaufplan Login
- E. Checkliste für Webstandards/CMS-Funktionen
- F. Amortisationsrechnung bei betriebsinternem Softwareeinsatz
- G. Analyse/Kalkulation und Empfehlung zum Softwarevertrieb
- H. Installationsanleitung für Web-CMS
- I. Datenbankdiagramm zum Web-CMS
- J. Quellcode Modul zur Verwaltung von Gruppenmitgliedschaften

## Quellenverzeichnis

Selfhtml 8.1 von Stefan Münz  
<http://www.teamone.de/selfhtml>

SelfPHP 2.0 von Damir Enseleit, Matthias Hupp  
<http://www.selfphp.de>

PHP Handbuch - InterBase-Funktionen  
<http://www.phpforum.de>

Web standards checklist by Russ Weakley  
<http://maxdesign.com.au/presentation/checklist.htm>

IBPhoenix Development SQL Statement and Function Reference  
<http://www.ibphoenix.com>

Borlands  
InterBase 6 SQL Reference

IT-Handbuch (2004): IT-Handbuch (Tabellenbuch) IT-Systemkaufmann/-frau,  
Informatikkaufmann/-frau; Braunschweig: Westermann Schulbuchverlag GmbH,  
3. Auflage, 2004.

Datum/Date: 20. März 2006

---

**Betreff / REF:** / 10

Name/name: **Lutz Kempe**  
Tel. Durchwahl/direct: 87373-10  
Email: lutz.kempe@ttm-germany.de

---

## **Weiterentwicklung von betriebsintern erstellten Softwarekomponenten**

Die Firmen MarMed GmbH & TTM e.V. erklären sich mit der freien Nutzung, wirtschaftlichen Verwertung sowie Weiterentwicklung der von Herrn Norman Scheffler während seines Betriebspraktikums vom 15. August bis 02. Dezember 2005 in unserem Auftrag erstellten Softwarekomponenten für ein webbasiertes

### **Content-Management-System**

einverstanden.

Diese Freigabe erfolgt unter der Bedingung der Kenntnisnahme und uneingeschränkten Teilhabe an den Ergebnissen der programmiererischen Weiterentwicklung durch Herrn Norman Scheffler seitens der Firmen MarMed GmbH & TTM e.V.

Lutz Kempe  
Geschäftsführer TTM e.V.

Wolfram Aeckersberg  
Geschäftsführer MarMed GmbH

# **Lastenheft für den Projektauftrag "Web-CMS"**

Verfasst durch:

**Norman Scheffler**

im Namen der Geschäftsleitung der Marburger Bürobedarf GmbH

Erstellt am 14. März 2006

# Inhalt

1. Zielbestimmung .....	1
2. Produkteinsatz .....	1
3. Produktfunktionen .....	1
3.1 Authentisierungsaufgaben .....	1
3.2 Administrationsaufgaben .....	1
3.2.1 Grundeinstellungen, Grundstruktur .....	1
3.2.2 Benutzerverwaltung .....	1
3.2.3 Verwaltung des Administrationsrechts .....	1
3.2.4 Gruppenverwaltung .....	2
3.2.4 Layoutverwaltungsaufgaben .....	2
3.3 Aufgaben für redaktionelle Tätigkeiten .....	2
3.3.1 Publikationsaufgaben .....	2
3.3.2 Inhalt-Verwaltungsaufgaben .....	2
3.3.3 Verwaltung von Lese- und Schreibrechten .....	2
3.4 Groupware-Funktionen .....	2
3.5 Transparenz .....	3
4. Produktdaten .....	3
5. Anforderungen an die Serverumgebung/Kompatibilität .....	3
6. Qualitätsanforderungen .....	3
6.1 Bedienung .....	3
6.2 Barrierefreiheit .....	3
6.3 Überprüfung von Formulareingaben .....	3
6.4 Browsereignung, Validität .....	3
7. Erweiterbarkeit, Schnittstellen .....	4
8. Zeitlicher Rahmen/Amortisation der Projektkosten bei betriebsinterner Nutzung .....	4
9. Analyse und Empfehlung zum Softwarevertrieb .....	4

## ***1. Zielbestimmung***

Die Geschäftsleitung der Marburger Bürobedarf GmbH beauftragt Herrn Norman Scheffler mit der Erstellung eines webbasierten Content-Management-Systems, das den jeweiligen Nutzern das Anlegen, die Publikation und das Verwalten von Inhalten im internen Netz ohne Programmierkenntnisse ermöglichen soll. Das Produkt soll die Kommunikation und die Dokumentenverwaltung von Benutzern am jeweiligen Einsatzort vereinfachen.

## ***2. Produkteinsatz***

Neben der firmeninternen Nutzung soll die zu programmierende Software auch an die Kunden der Marburger Bürobedarf GmbH vertrieben werden. Das CMS soll als Informations- und Kommunikationsportal im jeweiligen Intranet zur Verfügung stehen. Für Messen oder Informationsveranstaltungen soll es sich zudem als Präsentationsmedium eignen.

## ***3. Produktfunktionen***

Die zu erstellende Software hat folgende Aufgaben zu erfüllen:

### **3.1 Authentisierungsaufgaben**

Bei der in Auftrag gegebenen Software handelt es sich um ein Mehrbenutzersystem. Zur Identifikation des Nutzers und der damit verbundenen Rechtezuweisung ist dessen Authentisierung über einen Login-Bereich zu realisieren.

Nach der Installation des Systems muss zunächst ein Default-Nutzer mit Administrationsrechten zur Verfügung stehen. Ein vom Administrator (s. 3.2.2) angelegter Nutzer soll über die Dauer seines "Besuchs" hinaus mit seinen Anmeldeinformationen gespeichert bleiben um eine ständige Neuansmeldung zu vermeiden. Dies soll ausdrücklich nicht für die Administrationseigenschaft gelten.

### **3.2 Administrationsaufgaben**

#### **3.2.1 Grundeinstellungen, Grundstruktur**

Grundlegende Konfigurationsaufgaben wie das Festlegen der Startseite, des Titels und des Navigationslayouts sollen einem Nutzer mit Administrationseigenschaft vorbehalten bleiben. Vorgaben für eine strukturierte Referenzierung von Dokumenten in der Navigation gibt der Administrator durch Anlegen von Haupt- und Unterkategorien.

#### **3.2.2 Benutzerverwaltung**

Durch Eingabe von grundlegenden Benutzerdaten (Name, Vorname usw.) soll der Administrator weitere Nutzer am System anmelden können. Login-Name und Passwort sind automatisch zu generieren, sie sollen jedoch änderbar sein.

#### **3.2.3 Verwaltung des Administrationsrechts**

Die Administrationseigenschaft ist gegebenenfalls von einem Nutzer mit entsprechenden Rechten bei der Neuansmeldung eines Users zu vergeben.

### **3.2.4 Gruppenverwaltung**

Der Administrator soll Gruppen im System definieren und ihnen Mitglieder zuweisen können. Der angemeldete Nutzer soll mehreren Gruppen zugehörig sein können, also Inhaber von Rechten derjenigen Gruppen sein, denen er zugehört. Gruppen sollen löschar sein und damit verbunden auch dazugehörige Rechte der jeweiligen Nutzer.

### **3.2.4 Layoutverwaltungsaufgaben**

Durch Anlegen von Style-Klassen durch den Administrator sollen dem Redakteur zusätzliche elementbezogene Layoutmöglichkeiten bei der Erstellung von Inhaltsseiten verschafft werden.

## **3.3 Aufgaben für redaktionelle Tätigkeiten**

### **3.3.1 Publikationsaufgaben**

Der Nutzer mit Redaktionsrechten soll mittels übersichtlicher Bedienelemente in die Lage versetzt werden, Inhaltsseiten mit den gängigsten Komponenten wie Überschriften, Textblöcken, Linklisten, Tabellen oder Bildern ohne Programmierkenntnis zu erstellen. Dateien wie Images müssen zum Einbinden in Inhalte über einen Upload in das Webverzeichnis kopiert und automatisch adressiert werden können.

### **3.3.2 Inhalt-Verwaltungsaufgaben**

Neben auf oben genannte Weise erstellten Inhaltsseiten sollen auch sonstige Dokumente oder bereits existierende Scriptdateien wie PDFs, HTML-Dateien oder PHP-Scripte auf einfache Weise in eine Navigationsstruktur oder in eine Linkliste einzubinden sein. Die Adressierung dieser Dokumente ist für die jeweilige Referenzierung zu automatisieren.

Dies bedingt zudem die Möglichkeit des Anlegens, Verschiebens, Löschens und der Korrektur von Navigationskategorien und –unterkategorien um die Übersichtlichkeit des Webangebots zu gewährleisten.

### **3.3.3 Verwaltung von Lese- und Schreibrechten**

Der Ersteller einer Seite soll über ein Webformular festlegen können, welche Gruppen Zugriff auf die von ihm erstellten Inhalte haben dürfen. Hierbei hat er zwischen Lese- und Schreibrechten unterscheiden zu können. Ein angemeldeter Nutzer mit Administrationsrecht soll von derartigen Einschränkungen nicht betroffen sein.

## **3.4 Groupware-Funktionen**

Dem angemeldeten Nutzer des Webangebots soll ein Terminkalender zur Verfügung stehen, der es diesem je nach Gruppenzugehörigkeit gestattet, persönliche, gruppenspezifische und allgemeine Termine einzusehen. Es sollen Termine für verschiedene Zielpersonen und –gruppen von einem jeden eingeloggten Nutzer eingetragen werden können. Löschar sollen Termine nur von dessen Ersteller und für denjenigen sein, der allein von einem Termin betroffen ist.

### **3.5 Transparenz**

Jeglichen angelegten Inhalten soll deren Autor und das Datum der letzten Änderung zu entnehmen sein. So ist zu gewährleisten, dass Verfasser von Spam-Beiträgen identifiziert werden können. Auch beim Abspeichern von Terminen sind der Verfasser und das Datum der Eintragung festzuhalten.

### **4. Produktdaten**

Die zu erstellende Software soll Daten, die der angemeldete Nutzer zum Zwecke der Publikation in das CMS einträgt, in Datenbanktabellen verwalten. Zur Identifikation des Nutzers ist das Abspeichern von ihm betreffenden Informationen nötig. Anmeldeinformationen sollen nur vom Nutzer persönlich einsehbar und veränderbar sein.

### **5. Anforderungen an die Serverumgebung/Kompatibilität**

Firmenintern soll die Anwendung auf einem Linux-basierten Datenbankserver und auf einem Windows 2003 Server mit installierter Apache-Webserversoftware verteilt betrieben werden. Im Hinblick darauf, dass die jeweilige Serverumgebung bei den Kunden der MBB nicht vorausgesehen werden kann, ist zu gewährleisten, dass das Produkt flexibel auf Windows- und Linux-Betriebssystemen installierbar ist.

### **6. Qualitätsanforderungen**

#### **6.1 Bedienung**

Die Bedienungsfreundlichkeit soll bei dem Produkt im Vordergrund stehen. Der Nutzer soll durch klare Anweisungen durch die Funktionen des Systems geführt werden und in die Lage versetzt werden, gänzlich ohne eigene Programmierkenntnisse mit geringem Zeitaufwand Inhalte im internen Netz zu publizieren.

Die Antwort des Systems auf eine Benutzereingabe soll lediglich kurze Wartezeiten erfordern, besonders bei den Zugriffen auf relevante Dokumente und Informationen, etwa auf ein PDF oder auf die Anfrage der aktuellen Termine.

#### **6.2 Barrierefreiheit**

Nicht zuletzt aufgrund der Tatsache, dass es sich bei der Marburger Bürobedarf GmbH um einen Ausbildungsbetrieb zu Informatikkaufleuten für Blinde und Sehbehinderte handelt, ist bei dem vom System generierten HTML-Quellcode auf dessen Barrierefreiheit zu achten.

#### **6.3 Überprüfung von Formulareingaben**

Durch systeminterne Überprüfung der vom Nutzer eingegebenen Formularinhalte ist auszuschließen, dass es durch (auch versehentliche) Verwendung von Script-Code zu Fehlern in der Anzeige oder in der Ausführbarkeit des Systems kommt.

#### **6.4 Browsereignung, Validität**

Die Bedienung des CMS muss mit allen gängigen Browsern möglich sein, auch darf sich das Layout des Webinformationsportals bei der Benutzung unterschiedlicher

Webbrowser nicht grundlegend unterscheiden. Dies ist vor allem durch Validität des zu generierenden HTML-Quellcodes zu gewährleisten.

### ***7. Erweiterbarkeit, Schnittstellen***

Die dem redaktionellen Nutzer zur Verfügung stehenden Programmmodule sollen gegebenenfalls um weitere Komponenten ergänzt werden können. Hierfür sind entsprechende Schnittstellen vorzuhalten, die die Einbindung neuer Programm Komponenten ohne relevante Anpassungsaufwand bei vorhandenen Programmmodulen ermöglichen.

### ***8. Zeitlicher Rahmen/Amortisation der Projektkosten bei betriebsinterner Nutzung***

Die Durchführung des Projektes darf einen zeitlichen Rahmen von 35 betrieblichen Arbeitsstunden nicht übersteigen.

Bestandteil der in Auftrag gegebenen Projektarbeit ist zudem eine Amortisationsrechnung bei (ausschließlich) firmeninterner Nutzung.

### ***9. Analyse und Empfehlung zum Softwarevertrieb***

Abschließend sollen der Geschäftsleitung der Marburger Bürobedarf GmbH in Betracht kommende Möglichkeiten zum Vertrieb der Software in Verbindung mit den jeweiligen kalkulatorischen Rechnungen zum Erreichen der Gewinnschwelle vorgestellt und eine diesbezügliche begründete Empfehlung ausgesprochen werden.

Marburg, den

---

Unterschrift Geschäftsleitung

---

Unterschrift Auftragnehmer

# **Pflichtenheft für den Projektauftrag "Web-CMS"**

Verfasst durch:

Norman Scheffler

Erstellt am 16. März 2006

Zusammenfassung:

In diesem Dokument werden die im Lastenheft niedergeschriebenen Anforderungen der Geschäftsleitung an ein Web-CMS präzisiert und mit technischen Festlegungen verknüpft. Aufgrund der Tatsache, dass die zu erstellenden Programmmodule funktional auf bereits während meines Praktikums bei der MarMed GmbH Cölbe programmierten Teilkomponenten zur Verwaltung von Webinhalten aufbauen, wird in diesem Pflichtenheft eindeutig zwischen den aktuellen Aufgaben der Projektarbeit und den damaligen Schritten zur Softwareerstellung unterschieden.

# Inhalt

1. Ausgangssituation .....	1
2. Zielbestimmung .....	1
2.1 Muss-Kriterien .....	1
2.2 Wunsch-Kriterien .....	1
2.3 Abgrenzungskriterien .....	1
3. Produktkomponenten .....	2
3.1 Neu zu erstellende Programmmodule .....	2
3.1.1 Nutzer-, Gruppen- und Rechteverwaltung .....	2
3.1.2 Beschränkung/Gewährung von Lese- und Schreibrechten .....	2
3.1.3 Anlegen, Verwalten und Einsatz von zusätzlichen Styleklassen .....	3
3.1.4 Festlegen von CMS-Grundeinstellungen .....	3
3.2 Überarbeitung von aus Praktikum stammenden Modulen .....	3
3.2.1 Anpassung des Terminkalenders .....	3
3.2.2 Schutz des Administrationsbereichs .....	4
3.2.3 Erweiterung des Editiermodus von Inhaltsseiten .....	4
3.3 (Nahezu) unveränderte Programmmodule .....	4
4. Serverumgebung, Kompatibilität und Wahl der Programmiersprachen .....	4
5. Qualitätsanforderungen .....	5
6. Erweiterbarkeit der Software .....	5
7. Amortisationsrechnung und Analyse zum Softwarevertrieb .....	5

## **1. Ausgangssituation**

Die Geschäftsleitung der Marburger Bürobedarf GmbH beauftragte mich mit der Erstellung eines webbasierten Content-Management-Systems für den firmeninternen Gebrauch und den Vertrieb an die Kunden der MBB. Dieser Projektauftrag erfolgte mit dem Wissen des Auftraggebers um die Existenz einer bereits während meines 16-wöchigen Betriebspraktikums bei der MarMed GmbH Cölbe programmierten betriebsspezifischen Lösung zur datenbankbasierten Verwaltung von Webinhalten.

## **2. Zielbestimmung**

Mit der zu entwickelnden Software sollen spätere Nutzer zeitsparend Inhalte im jeweiligen lokalen Netz publizieren und auf diese zugreifen können, ohne eigene Programmierkenntnisse aufweisen zu müssen.

In diesem Pflichtenheft erfolgt eine Konkretisierung der im Auftrag der Geschäftsleitung im Lastenheft festgehaltenen Anforderungen an ein Web-CMS in Bezug auf Funktionalität und Programmstruktur. Zudem wird in diesem Dokument eine Trennung der Anforderungen aus dem Lastenheft nach den folgenden Kriterien vorgenommen:

- Programmmodul komplett neu zu entwerfen und zu erstellen
- Aus Praktikumstätigkeit stammendes Programmmodul muss entsprechend geänderter Anforderungen neu konzipiert und programmiert werden.
- Aus Praktikumstätigkeit stammendes Programmmodul entspricht bereits ohne wesentliche Veränderungen den Anforderungen des Lastenhefts.

### **2.1 Muss-Kriterien**

Die Anwendung muss dem Nutzer ein Erstellen, Verändern und Strukturieren von Webinhalten ohne eigene Programmierkenntnisse ermöglichen. Dieses und die gesamte Administrations- und Grundkonfigurationstätigkeit muss zwingend an dem Nutzer gewährte Rechte gekoppelt sein. Nutzerrechte setzen zwingend einen Login-Bereich und eine Rechteverwaltung voraus.

Die gesamte Anwendung hat Qualitätsanforderungen bezüglich barrierefreier Webprogrammierung zu entsprechen.

Die Software muss sowohl auf Windows-, als auch auf Linux-Betriebssystemen betrieben werden können.

### **2.2 Wunsch-Kriterien**

Über die Grundanforderungen an eine Rechteverwaltung hinaus wird es als nützlich erachtet, dem Administrator des Web-CMS einen Überblick über sämtliche registrierte Nutzer und deren Gruppenzugehörigkeiten zu gewähren. Die Identifizierbarkeit des Verfassers eines Beitrages kann als wirksames Mittel zur Verhinderung von Spam-Beiträgen angesehen werden.

Die Entscheidung über eine Nutzung von Groupware-Funktionen (Terminkalender) obliegt den Verantwortlichen am jeweiligen Einsatzort der Software, wird aber als Programmmodul zur Verfügung gestellt

### **2.3 Abgrenzungskriterien**

Das Content-Management-System soll ausdrücklich nicht zum Versand von E-Mails gebraucht werden können. Dies würde zudem eine genaue Kenntnis des Ziel-

Betriebssystems bzw. –netzwerkes voraussetzen und widersprüche zudem dem Gedanken der zentralen Verwaltung und Ablage von Informationen.

### 3. Produktkomponenten

#### 3.1 Neu zu erstellende Programmmodule

##### 3.1.1 Nutzer-, Gruppen- und Rechteverwaltung

Voraussetzung für die Verwirklichung dieses Pakets von zusammengehörigen Programmmodulen ist eine Datenbankstruktur, in der Nutzer der Software mit relevanten persönlichen Daten abgespeichert werden können. Zudem müssen diese Nutzer in Gruppen zu Einheiten mit identischen Rechten zusammengefasst werden können.

Über zu programmierende Weboberflächen muss der spätere Administrator des CMS

- neue Nutzer mit persönlichen Daten anlegen können, Nutzerdaten korrigieren sowie eingetragene Nutzer aus dem System löschen können;
- neue Gruppen mit zugehörigen Daten anlegen, Gruppendaten ändern sowie angelegte Gruppen aus dem System löschen können;
- in eine zu bestimmende Gruppe Mitglieder aufnehmen und hieraus wieder entfernen können.

Die Anmeldung des späteren Nutzers am System erfolgt über einen zu programmierenden Login-Bereich in Form eines Web-Formulars, in dem zur Eingabe eines Benutzernamens und eines Passworts aufgefordert wird. Die Programmfunktionen, die sich an einen solchen Login-Vorgang knüpfen sollen, sind schematisch im in der Anlage D dargestellten Programmablaufplan abgebildet.

##### 3.1.2 Beschränkung/Gewährung von Lese- und Schreibrechten

Voraussetzung für einen seitenbezogenen Schutz vor Lese- und Schreibzugriffen ist die Möglichkeit, zu einem Web-Dokument die für den entsprechenden Zugriff berechtigten Gruppen festlegen zu können. Diese Bestimmung wird wieder durch jeweils zu programmierende Webformulare zu realisieren sein, mittels derer die verknüpften Informationen zu einer Seite und den Zugriffsberechtigten in Datenbanktabellen abzulegen sein werden.

Die folgende konsolidierte Entscheidungstabelle bildet das Regelwerk für den Schreib-Zugriff auf eine Web-Seite im CMS ab:

	Beschreibung	R1	R2	R3	R4	R5
<b>Bedingung 1:</b>	Nutzer ist angemeldet	J	J	J	J	N
<b>Bedingung 2:</b>	Nutzer ist Administrator	J	N	N	N	-
<b>Bedingung 3:</b>	Schreibrecht der Seite ist auf Gruppe(n) beschränkt	-	N	J	J	-
<b>Bedingung 4:</b>	Nutzer gehört Gruppe(n) mit Schreibrecht an	-	-	J	N	-
<b>Aktion 1:</b>	Nutzer wird Schreibrecht gewährt; Anzeige der Seite im Editiermodus					
<b>Aktion 2:</b>	Anzeige Hinweis auf fehlendes Schreibrecht; Abbruch des Aufrufs des Editiermodus.					

Auch das Regelwerk für den Lese-Zugriff auf eine Webseite im CMS ist nachfolgend in einer konsolidierten Entscheidungstabelle dargestellt:

	Beschreibung	R1	R2	R3	R4	R5
<b>Bedingung 1:</b>	Nutzer ist angemeldet	J	J	J	N	-
<b>Bedingung 2:</b>	Nutzer ist Administrator	J	N	N	-	-
<b>Bedingung 3:</b>	Leserechte der Seite sind auf Gruppe(n) eingeschränkt	J	J	J	J	N
<b>Bedingung 4:</b>	Nutzer gehört Gruppe(n) mit Leserecht an	-	J	N	-	-
<b>Aktion 1:</b>	Nutzer wird Leserecht gewährt, angefragte Seite wird angezeigt.					
<b>Aktion 2:</b>	Anzeige Hinweis auf fehlendes Leserecht, Abbruch des Seitenaufrufs					

Erteilt werden die dargestellten Rechte an einem Webdokument durch deren Bearbeiter.

### 3.1.3 Anlegen, Verwalten und Einsatz von zusätzlichen Style-Klassen

Der spätere Administrator des Web-CMS soll den Redakteuren des Intranets neben den in der zentralen Stylesheet-Datei definierten Layout-Klassen noch weitere elementbezogene Formatierungsmöglichkeiten zur Verfügung stellen können. Hierfür müssen in einer anzulegenden Datenbanktabelle Eigenschaften wie der Name und der CSS-Code abgelegt werden können und über ein zu erstellendes Web-Formular diese Klassen angelegt, verändert oder auch wieder gelöscht werden können.

### 3.1.4 Festlegen von CMS-Grundeinstellungen

Der spätere Administrator des CMS soll den firmenspezifischen Namen, die Startseite und das grundlegende Navigationslayout festlegen können. Diese Informationen sollen wiederum in einer Datenbanktabelle gehalten und über ein zu erstellendes Webformular verwaltet werden können.

## 3.2 Überarbeitung von aus Praktikum stammenden Modulen

### 3.2.1 Anpassung des Terminkalenders

Große Teile des aus der Praktikumstätigkeit stammenden Programmmoduls zum Anlegen und Verwalten von Terminen müssen für die neu zu schaffende Rechteverwaltungssituation geändert werden. Informationen über den Ersteller eines Termins und die von einem Termin betroffenen Gruppen oder Einzelpersonen sind in der Datenbankstruktur mit dem dazugehörigen Termin zu verknüpfen. Die Weboberflächen zur Anzeige und zum Anlegen von Einträgen sind dahingehend zu verändern, dass nicht nur allgemeine, sondern auch gruppen- und personenspezifische Termine angesetzt und angezeigt werden können.

Es ist zudem zu realisieren, dass nur der angemeldete Nutzer Termine eintragen darf, um diesen als Ersteller identifizieren zu können. Für den nicht angemeldeten Nutzer sollen nur Termine „allgemeiner Natur“ angezeigt werden, für den angemeldeten Nutzer alle allgemeinen Termine, diejenigen Einträge, die für Gruppen bestimmt sind, denen auch er angehört, sowie explizit für ihn selbst bestimmte Termine. Löschar soll ein Termin nur für den Ersteller oder für denjenigen sein, für den der Eintrag persönlich bestimmt ist.

Nahezu unverändert bleiben das Layout der Weboberflächen zur Anzeige/zum Eintragen sowie die in einer Datenbanktabelle abzulegenden Teilkomponenten von Terminen (Betreff, Termintext, Datum etc.).

### **3.2.2 Schutz des Administrationsbereichs**

Die bisher sämtlich in einem durch eine HTACCESS-Datei passwortgeschützten Ordner abgelegten Programm-Scripte für den administrativen Bereich des Content-Management-Systems werden mit einer Funktion versehen, die bei einem Zugriff auf diese Dateien überprüft, ob der zugreifende Nutzer Inhaber von Administrationsrechten ist. Diese Eigenschaft wird nach dem Login in einer Session-Variablen gespeichert werden und macht den bisherigen (als recht unsicher zu bezeichnenden) Schutz vor unberechtigtem Zugriff überflüssig.

### **3.2.3 Erweiterung des Editiermodus von Inhaltsseiten**

Die Datenbankstruktur für das Anlegen/Verwalten von Seiteninhalten wird für die jeweiligen spezifischen Layout-Möglichkeiten zu ergänzen sein. Zudem ist das bisherige Webformular mit all seinen Unterformularen zum Anlegen von Texten, Überschriften, Tabellen etc. um Ausklapplisten für zusätzliche elementbezogene Style-Zuweisungen zu erweitern. Jeder Speichervorgang wird den aktuellen Nutzer mit dem Datum im entsprechenden Datensatz der Inhaltsseite festhalten, wodurch den Transparenz-Anforderungen aus dem Lastenheft genüge getan wird. Aus dem Webformular zum Öffnen einer Seite im "Editiermodus" heraus muss dann auch festzulegen sein, welche Gruppen Lese- und Schreibrechte an der betreffenden Webseite genießen sollen.

## ***3.3 (Nahezu) unveränderte Programmmodule***

Im Folgenden werden nur kurz die Komponenten der bisherigen Software genannt, die zur Erfüllung der Anforderungen aus dem Lastenheft zum aktuellen Projektauftrag nicht oder nicht wesentlich verändert werden müssen:

- Softwareteile zum Anlegen, Löschen, Verschieben von Navigationskategorien und -unterkategorien.
- Module zum Erstellen von Tabellen, Linklisten und Bildbereichen
- Module zum automatischen Referenzieren von Dateien mittels eines "Web-Explorers"

## ***4. Serverumgebung, Kompatibilität und Wahl der Programmiersprachen***

Die im Lastenheft definierten Anforderungen an eine Eignung des CMS für verschiedene Serverbetriebssysteme sind dann als erfüllt anzusehen, wenn - wie bei den bereits aus meiner Praktikumstätigkeit stammenden Programmmodulen auch - als Programmiersprache PHP gewählt wird. PHP-Scriptdateien sind in Verbindung mit einer entsprechenden Webserversoftware auf allen gängigen Betriebssystemen ausführbar, auch der bereits im Praktikum verwandte Interbase-Datenbankserver ist sowohl auf Windows-, als auch auf Linux-Betriebssystemen einsetzbar.

## **5. Qualitätsanforderungen**

Die im Lastenheft zu dieser Projektarbeit gestellten Anforderungen an das Web-CMS bezüglich Bedienkomfort, Barrierefreiheit, Browsereignung/Validität und Zuverlässigkeit werde ich in einer zu erstellenden umfassenden Checkliste konkretisieren. Anhand eben dieser Liste sind sämtliche Komponenten der Software zu überprüfen und gegebenenfalls Änderungen (auch an bereits vorhandenen) Programmbestandteilen vorzunehmen.

## **6. Erweiterbarkeit der Software**

Schnittstellen zur Ergänzung des CMS um beispielsweise weitere Layout-Möglichkeiten wie Aufzählungslisten oder DIV-Blöcke sind in Bezug auf die Datenbankstruktur des CMS bereits geschaffen, der Quellcode der PHP-Scriptdateien wird derart zu strukturieren sein, dass das Einbinden solcher neuen Komponenten lediglich eine Ergänzung, nicht jedoch eine Abänderung von bestehenden Scriptanweisungen erfordert.

## **7. Amortisationsrechnung und Analyse zum Softwarevertrieb**

Für die Berechnung der Amortisation bei betriebsinterner Nutzung der zu Erstellenden Software sind von der Geschäftsleitung und den Kollegen Informationen über den bisherigen Publikationsaufwand im Intranet und Angaben über durchschnittliche monatliche Einkommen einzuholen.

Zur Erstellung einer Vertriebsempfehlung für das CMS wird eine Marktrecherche nach Konkurrenzprodukten und –preisen sowie zu den rechtlichen Aspekten der Softwarelizenzierung erforderlich.

Marburg, den

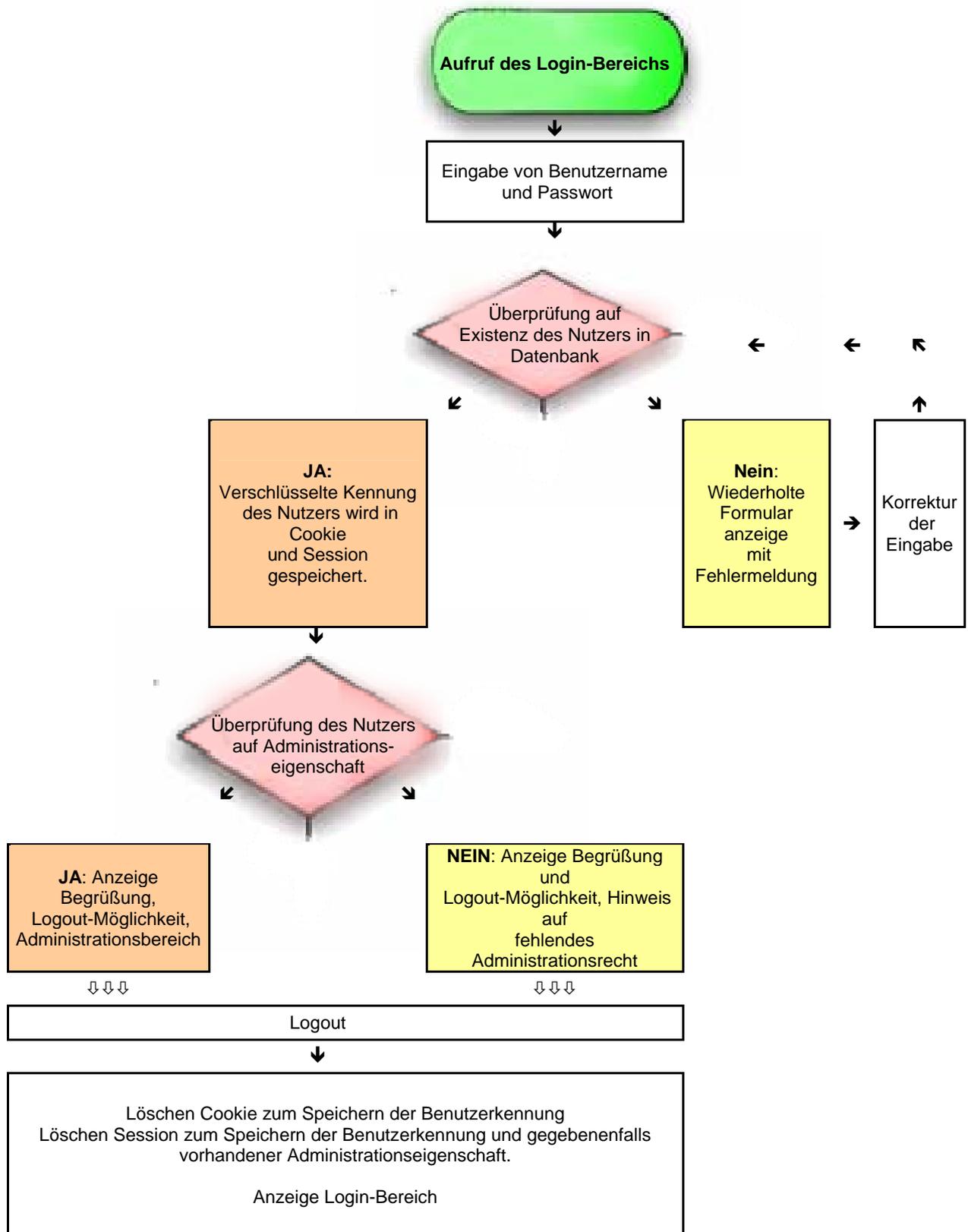
---

Unterschrift Geschäftsleitung

---

Unterschrift Auftragnehmer

Anlage D: Programmablaufplan „User-Login“ (nach DIN 66001)



# Checkliste zur Qualitätssicherung

## Übersicht

1. Checkliste für Webstandards .....	1
1.1 Qualität des Codes .....	1
1.2 Grad der Trennung von Inhalt und Layout .....	2
1.3 Zugänglichkeit (Accessibility) für Nutzer .....	2
1.4 Zugänglichkeit (Accessibility) für Geräte/Maschinen .....	3
1.5 Grundlegende Benutzbarkeit (Usability) .....	4
1.6 Seitenverwaltung .....	5
2. Checkliste für CMS-Funktionen .....	5
2.1 Test der Web-Formulare .....	5
2.2 Test der Datenbankzugriffe und -funktionen .....	6
2.3 Test von Rechtezuweisungen und -beschränkungen .....	6

## 1. Checkliste für Webstandards

### 1.1 Qualität des Codes

- Hat die Seite einen korrekten Doctype?  
*Anmerkung: Jede Seite bezieht ihren Head-Bereich aus einem Funktionsaufruf mit dem Parameter Seitentitel. Der Doctype und der Zeichensatz werden automatisch eingebunden.*
  
- Hat die Seite eine Zeichensatzcodierung?  
*Anmerkung: s.o.*
  
- Verwendet die Seite valides (X)HTML?  
*Anmerkung: Alle testhalber im CMS angelegten Seiten wurden auf ihre Validität mittels des Online-Validators unter <http://validator.w3.org> positiv geprüft. Hierfür wurde der Service von [www.dyndns.org](http://www.dyndns.org) für das Zuweisen einer dynamischen IP-Adresse genutzt.*
  
- Verwendet die Seite valides CSS?  
*Anmerkung: Die Validität des CSS wurde ebenfalls über den Online-Service des W3C positiv geprüft.*

- Verzichtet die Seite auf unnötige Klassen oder IDs?  
*Anmerkung: Jedes Element bezieht sein Standard-Layout aus einer zentralen CSS-Datei, spezielle Layout-Klassen können jedoch vom jeweiligen Redakteur zugewiesen werden.*
- Ist der Code gut strukturiert?  
*Anmerkung: Der automatisch erzeugte HTML-Quellcode ist optisch nicht strukturiert, ein Eingriff in diesen Quellcode ist ohnehin nicht möglich. Die diesen Quellcode erzeugenden PHP-Scripte dagegen sind ausreichend strukturiert.*
- Überprüfung auf tote Links:  
*Anmerkung: Die CMS-Grundstruktur weist keine toten Links auf, das Funktionieren weiterer Links verantwortet der jeweilige Redakteur von Inhaltsseiten. Ein automatisches Referenzieren von Dateien beugt jedoch solchen Fehlern vor.*
- Überprüfung der Größe/Geschwindigkeit der Seite:  
*Anmerkung: In der Grundstruktur des CMS sind keine ladezeitintensiven Grafiken enthalten, auch dieser Punkt obliegt im Einsatz dem jeweiligen Redakteur einer Inhaltsseite.*
- Überprüfung auf Javascript-Fehler:  
*Anmerkung: Bis auf den Bereich der dynamischen Navigation wurde im CMS nahezu gänzlich auf Javascript verzichtet. Dem Redakteur einer Seite ist es nicht möglich, Scriptcode über Eingabefelder einzufügen.*

## 1.2 Grad der Trennung von Inhalt und Layout

- Verwendet die Seite für die Gestaltung überall CSS (Schriftart, Farbe, Abstände, Ränder etc.)?  
*Anmerkung: Layout-Angaben erfolgen ausschließlich über Cascading Stylesheets.*

## 1.3 Zugänglichkeit (Accessibility) für Nutzer

- Ist das ALT-Attribut für alle beschreibenden Bilder sinnvoll gesetzt?  
*Anmerkung: Die in der CMS-Grundstruktur enthaltenen Grafiken sind jeweils mit aussagekräftigem Alt-Text versehen, wieder obliegt es im Einsatz des Systems dem Redakteur einer Seite, beim Einbinden von Grafiken auf diesen Punkt zu achten.*
- Verwendet die Seite relative Einheiten für die Schriftgröße (statt absoluter Einheiten)?  
*Anmerkung: Zusätzliche, vom jeweiligen Administrator angelegte, Style-Klassen obliegen nicht der Kontrolle des Programmierers. Das Grundlayout der Software verwendet relative Schriftgrößen.*

- Wird das Layout beim Vergrößern der Schrift gestört?  
*Anmerkung: Bei extremer Vergrößerung zeigen sich unter Umständen ungewünschte Scrollbalken.*
- Nutzt die Seite zugängliche Formulare?  
*Anmerkung: Jedes Formularelement ist sowohl mit Maus, als auch mit der Tastatur erreichbar. Alle relevanten Schalter sind eindeutig beschriftet und besitzen einen Shortcut. Formularelemente, die kein eigenes Label besitzen sind mit einem Title-Attribut versehen.*
- Nutzt die Seite zugängliche Tabellen?  
*Anmerkung: In der CMS-Grundstruktur sind die verwandten Tabellen klar strukturiert, die später von den Redakteuren möglicherweise einzufügenden Tabellen erlauben keine Verschachtelungen.*
- Haben die Farben ausreichend Helligkeit/Kontrast?  
*Anmerkung: Positiv fremdgetestet.*
- Werden wichtige Informationen nur mit Farben gekennzeichnet?  
*Anmerkung: Hinweise zu den Bedienelementen des CMS sind logisch verfasst und jeweils optisch eindeutig zu den von ihnen beschriebenen Objekten platziert. Die Deklaration wichtiger Informationen obliegt über die CMS-Grundstruktur hinaus wiederum den jeweiligen Redakteuren. Eine Vielzahl von Style-Klassen steht potentiell zur Verfügung.*
- Sind alle Linktexte beschreibend (für blinde Nutzer)?  
*Anmerkung: Zutreffend für die CMS-Grundstruktur, beim Neuerstellen von Links oder Linklisten hat wiederum der Redakteur hierauf zu achten.*

#### **1.4 Zugänglichkeit (Accessibility) für Geräte/Maschinen**

- Funktioniert die Seite in neuen und alten Browsern vernünftig?  
*Anmerkung: Es werden keinerlei HTML-Tags oder -Attribute verwandt, die von älteren Browsern nicht interpretiert werden können. Browser, die nicht CSS-fähig sind, werden das vorgesehene Layout jedoch nicht transportieren können.*
- Ist der Inhalt der Seite auch ohne CSS zugänglich?  
*Anmerkung: Der jeweilige Inhalt des Web-CMS wird in jedem Fall auch ohne CSS sichtbar sein.*
- Ist der Inhalt der Seite auch ohne Bilder zugänglich?  
*Anmerkung: Es wurden in der CMS-Grundstruktur keinerlei grafische Links oder Schalter verwandt. Der spätere Inhalt obliegt auch diesbezüglich dem jeweiligen Redakteur.*

- Funktioniert die Seite auch in Textbrowsern wie Lynx?  
*Anmerkung: Positiv fremdgetestet.*
- Lässt sich die Seite vernünftig ausdrucken?  
*Anmerkung: Die für einen eventuellen Ausdruck in Frage kommenden Seiten wurden mit einem speziellen Print-Layout in Form einer CSS-Datei versehen.*
- Funktioniert die Seite auf Handhelds?  
*Anmerkung: Test mangels Verfügbarkeit nicht möglich.*
- Besitzt die Seite ausführliche Meta-Informationen?  
*Anmerkung: In diesem Stadium der Softwareentwicklung ist es noch nicht möglich, auf die Meta-Informationen der vom CMS generierten HTML-Dokumente zuzugreifen.*
- Funktioniert die Seite in verschiedenen Fenstergrößen des Browsers?  
*Anmerkung: Bei zu extremer Verkleinerung des Browserfensters erscheint der Inhalt nicht mehr schlüssig.*

### 1.5 Grundlegende Benutzbarkeit (Usability)

- Gibt es eine klare visuelle Gliederung?  
*Anmerkung: Gerade dies ist der Vorteil der geschaffenen Gliederungsmöglichkeiten in der Navigation.*
- Sind die Überschriften unterschiedlicher Ordnung leicht unterscheidbar?  
*Anmerkung: Im Standard-Layout unterscheiden sich Überschriften nicht nur durch ihre Größe, sondern auch durch ihre Hintergrundfarbe.*
- Besitzt die Seite eine leicht verständliche Navigation?  
*Anmerkung: Es besteht zumindest die Möglichkeit für den Nutzer, die Navigation sehr logisch zu konzipieren.*
- Ist die Navigation auf allen Seiten durchgängig?  
*Anmerkung: Der Navigationsbereich steht dem Nutzer permanent im selben Layout zur Verfügung. Dies trifft dann nicht zu, wenn sich vom Redakteur erzeugte Verweise zu anderen Dokumenten im neuen Fenster öffnen sollen.*
- Wird eine konsistente und passende Sprache/Formulierung verwendet?  
*Anmerkung: Für die CMS-Bedienelemente trifft dies zu, der spätere Inhalt liegt in der Verantwortung des Redakteurs.*
- Gibt es eine Sitemap und ist diese leicht zu finden?  
*Anmerkung: Hierauf wurde bewusst verzichtet, dies widerspräche dem Grundkonzept der geforderten Rechteverwaltung.*

- Ist bei größeren Seiten eine Suchfunktion vorhanden?  
*Anmerkung: Dieses Feature ist neben weiteren Komponenten in einer Folgeversion geplant.*
- Gibt es auf jeder Seite einen Link zur Startseite?  
*Anmerkung: Der Navigationsbereich des CMS sieht dies vor, dies kann aber vom jeweiligen Administrator auch anders entschieden werden.*
- Sind alle Links unterstrichen?  
*Anmerkung: Im Standard-Layout trifft dies zu, auch diese Eigenschaft ist elementbezogen änderbar.*
- Sind schon besuchte Links entsprechend farblich gekennzeichnet?  
*Anmerkung: Im Standard-Layout ist eine farbliche Hervorhebung von besuchten oder aktiven Links vorgesehen.*

## 1.6 Seitenverwaltung

- Hat die Seite eine aussagefähige und hilfreiche 404-Fehlerseite, die überall funktioniert?  
*Anmerkung: Mittels einer HTACCESS-Datei im Wurzelverzeichnis des Webservers wird für den Fall des Nichtvorhandenseins einer aufgerufenen Seite auf eine selbst erstellte Fehlerseite mit enthaltenen Anweisungen weitergeleitet.*
- Hat die Seite lesbare URLs?  
*Anmerkung: Die im Webbrowser einzugebenden Adressen hängen vom Namen des Zielsystemrechners ab.*
- Hat die Seite ein Favicon?  
*Anmerkung: Im Wurzelverzeichnis des Webservers wurde ein Favicon als Software-Logo abgelegt. Dieses Logo wird beim Aufruf der Software in der Adresszeile des Browsers aufgerufen.*

## 2. Checkliste für CMS-Funktionen

### 2.1 Test der Web-Formulare

- Werden leere Formulareingaben vom System abgefangen/toleriert?  
*Anmerkung: diejenigen leer gelassenen Formularfelder, die zwingend eine Eingabe des Users erfordern, werden nicht, oder mit vom System zugewiesenen Default-Werten verarbeitet.*
- Toleriert das System unzulässige Formularwerte/Code-Eingaben?  
*Anmerkung: In den Fällen, in denen User-Eingaben in Datenbanktabellen gespeichert werden sollen und nicht nur als Vergleich zur Authentisierung*

*dienen, wird der eingegebene Text mit Stringfunktionen auf unzulässige Zeichen überprüft und diese gegebenenfalls ersetzt oder eliminiert.*

- Sind auszuwählende Formulareinträge jeweils sinnvoll vorselektiert?  
*Anmerkung: Insbesondere Ausklapplisten für Datum/Uhrzeit sind mit aktuellen Werten voreingetragen. Prinzipiell haben alle Ausklapplisten, Radio-Buttons etc. sinnvolle Vorselektierungen.*

## **2.2 Test der Datenbankzugriffe und -funktionen**

- Kann es durch Nutzung des Users zu Datenbankfehlern kommen?  
*Anmerkung: Fehler werden in den Scriptdateien abgefangen und dem User rückgemeldet. (Beispiel: Verstoß gegen Löschrregeln).*
- Erhält der Nutzer sinnvolle Fehlermeldungen bei Funktionsstörungen?  
*Anmerkung: In Betracht kommen hierfür lediglich ein Ausfall des Webservers oder des Datenbankservers. Ersteres wird über den Webbrowser mit einer Nichtverfügbarkeit der angeforderten Seite angezeigt, für den letzteren Fall generiert die Software eine entsprechende Hinweismeldung.*

## **2.3 Test von Rechtezuweisungen und -beschränkungen**

- Sind Scriptdateien für den Administrationsbereich auch ohne Berechtigung zugänglich?  
*Anmerkung: Jede einzelne PHP-Datei überprüft bei ihrem Aufruf eine Session-Variable auf ein Administrationsrecht des Users. Ein Fehlen dieser Eigenschaft führt trotz Aufrufs der Datei über die Adresszeile des Webbrowsers zum Abbruch des Seitenaufbaus und zu einer Fehlermeldung.*
- Lassen sich sonstige Lese- oder Schreibbeschränkungen vom Nicht-Berechtigten umgehen?  
*Anmerkung: Mehrmalige erfolglose Fremdtests*

Anlage F: Amortisationsrechnung bei (ausschließlich) firmeninterner Nutzung des Web-CMS

<b>Ausbilder:</b>		<b>Auszubildende:</b>	
Anzahl:	4	Anzahl:	19
*) Durchschnittlicher jährlicher Bruttoverdienst:	27.000,00 €	*) Durchschnittlicher jährlicher Bruttoverdienst:	6.600,00 €
*) + 25 % Arbeitgeberanteil für Sozialabgaben	33.750,00 €	*) + 25 % Arbeitgeberanteil für Sozialabgaben	8.250,00 €
*) + 15 % Handlungskostenzuschlag	38.812,50 €	*) + 15 % Handlungskostenzuschlag	9.487,50 €
*) Anzahl Arbeitstage/Jahr abzüglich 29 Urlaubstage	223	*) Anzahl Arbeitstage/Jahr abzüglich 26 Urlaubstage	226
*) Durchschnittliche wöchentliche betriebliche Arbeitszeit	28:17 h	Durchschnittliche wöchentliche betriebliche Arbeitszeit (=38:30 h Wochenarbeitszeit - 14:00 h Berufschulzeit)	24:30 h
Durchschnittliche tägliche betriebliche Arbeitszeit	05:39 h	Durchschnittliche tägliche betriebliche Arbeitszeit	04:54 h
Durchschnittliche Kosten pro Arbeitsstunde	30,77 €	Durchschnittliche Kosten pro Arbeitsstunde	8,57 €

Durchschnittliche Kosten /Arbeitsstunde aller 27 Mitarbeiter	10,59 €
Projektgesamtkosten	327,12 €
(bei durchschnittlich zu veranschlagenden Arbeitskosten) Break-Even-Point nach Zeitersparnis von	30:53 h
Amortisation des Projekts nach einem Jahr bei einer monatlichen Zeitersparnis von	00:05 h/Mitarbeiter

\*) Angaben von Herrn Rainer Datzer,  
Geschäftsführer der Marburger Bürobedarf GmbH

## **Analyse der Möglichkeiten zum Softwarevertrieb**

Neben der MBB-internen Nutzung soll das im Rahmen meiner Projektarbeit kompletierte Web-CMS auch an die Kunden des Unternehmens im Ring der Übungsfirmen vertrieben werden. Nachfolgend erläutere ich 3 in Betracht kommende Vertriebswege mit den notwendigen Kalkulationsrechnungen.

### **1. Verkauf von Software-Lizenzen**

Die erste in Betracht kommende Möglichkeit des Softwarevertriebs ist der Weg der Lizenzveräußerung.

Erwerber der Software bekämen nach Zahlung einer festzulegenden Lizenzgebühr ein zeitlich unbefristetes Nutzungsrecht am Web-CMS durch den Lizenzgeber (hier die MBB GmbH) eingeräumt.

In der Praxis wäre eine Vertragsabwicklung derart vorstellbar, dass nach Zustandekommen des Vertrages und nach Zahlungseingang dem interessierten Kunden die Software inklusive des Software-Lizenzvertrages durch den Versand einer CD-ROM oder E-Mail zur Verfügung gestellt und seitens der MBB dem Kunden ein uneingeschränktes und zeitlich unbefristetes Nutzungsrecht eingeräumt werden würde.

Angenommener Preis der Softwarelizenz	10,00 EUR
enthaltene 16 % MwSt.	1,38 EUR
Handlungskostenzuschlag i. H. v. 30%	1,99 EUR
(fiktiver) Selbstkostenpreis	6,63 EUR
Softwareentwicklungskosten (Projektgesamtkosten + CD-ROM-Layout, Entwurf Lizenzvertrag usw.)	400,00 EUR
Anzahl Lizenzveräußerungen zur Erreichung der Gewinnschwelle	61

### **2. Vertrieb zur kostenfreien Nutzung durch den Endverbraucher und Lizenzierung über die • GNU General Public License (GPL)**

Bei Wahl dieser Vertriebsmöglichkeit würde einem jedem Nutzer ein uneingeschränktes Nutzungsrecht an der Software eingeräumt werden. Eine hierfür vorausgesetzte Offenlegung des Quellcodes lässt sich insbesondere bei PHP-Scriptdateien ohnehin nicht verhindern. Das CMS könnte auf der Firmen-Webseite zum freien Download angeboten werden, wobei kein weiterer Vertriebsaufwand erforderlich wäre.

Einzigste Möglichkeit der Gewinnerzielung bliebe bei dieser Lösung der Weg über einen kostenpflichtigen Support.

Supportpauschale (angenommener durchschnittlicher Zeitaufwand 1h):	25,00 EUR
- 16% MwSt.	3,45 EUR
- 40%iger Supportkostenzuschlag (Telefon, Fax etc.)	3,43 EUR
- durchschnittliche Kosten einer betrieblichen Arbeitsstunde von Auszubildenden	8,57 EUR
=Gewinn/Supportstunde	9,55 EUR
Softwareentwicklungskosten (Projektkosten + Bereitstellen zum Download etc.)	400,00 EUR
Anzahl Supportstunden zum Erreichen der Gewinnschwelle:	42

### 3. Kostenfreier Vertrieb der Software mit integriertem MBB-Produktkatalog

Dieser Vorschlag zum Softwarevertrieb baut in Bezug auf Gewährung von Nutzungslizenzen auf Vorschlag 2 auf, verzichtet jedoch auf einen kostenpflichtigen Support als potentielle Einnahmequelle. Stattdessen wird die Software in der Form zum Download angeboten, dass nach Installation dem Nutzer ein Web-CMS zur Verfügung steht, in dem sich bereits Informationen über die MBB und deren Produktpalette befinden.

Ziel dieser Vertriebsform ist der hiermit verbundene Informations- und Werbeeffekt und eine zu erwartende Umsatzsteigerung der MBB GmbH.

Vertrieb 3	
Produktentwicklungskosten (=Projektkosten + MBB-Produktkatalog + Bereitstellung zum Download)	500,00 EUR
*) durchschnittlicher Umsatz-Gewinnanteil	13%
notwendige Umsatzsteigerung der MBB zur Amortisierung der kostenlosen Software:	3.846,15 EUR

\*) Angabe durch Herrn Rainer Datzer, Geschäftsführer der MBB GmbH

## Vertriebsempfehlung

Die hohe Anzahl von Anbietern auch kostenfreier CMS-Produkte lässt einen wirtschaftlichen Erfolg von entgeltlichen Lizenzvergaben nicht vermuten. In Anbetracht des doch sehr geringen Installations- und Konfigurationsaufwandes ist eine erhebliche Inanspruchnahme von entgeltlichen Supportleistungen ebenfalls als unwahrscheinlich anzusehen. Zudem wäre bei dieser Lösung das Problem der Fakturierung von Forderungen aus diesen Serviceverträgen nicht zu unterschätzen.

Schlussendlich empfehle ich der Geschäftsleitung der Marburger Bürobedarf GmbH einen Vertrieb der Software auf die im Vorschlag 3 dargestellte Art und Weise.

Mit sehr geringem Aufwand, nämlich lediglich durch das Bereitstellen der Software zum kostenfreien Download, könnte sich die MBB gegenüber ihren Mitbewerbern erhebliche Vorteile im Bereich der Produktinformation verschaffen. Am Programm interessierte Kunden bräuchten zudem dann nicht mehr mit Katalogmaterial versorgt werden, was einen weiteren positiven Kosteneffekt zur Folge hätte.

# Installationsanleitung für das Web-CMS

## *Kurzgliederung*

1. Systemvoraussetzungen .....	1
2. Softwarevoraussetzungen .....	1
2.1 Webserver .....	1
2.2. Datenbankserver.....	1
2.3 Konfigurationsarbeit .....	2
3. Installation des CMS.....	2
3.1 Die CMS-Datenbank .....	2
3.2 Die CMS-Scriptdateien .....	2

## **1. Systemvoraussetzungen**

Die Anwendung, das heißt sowohl die CMS-Scriptdateien, als auch die Datenbankserversoftware können sowohl auf Windows-Betriebssystemen (ab Windows 2000), als auch auf einem Linux-Betriebssystem installiert werden.

Auf der Installations-CD befinden sich neben den selbst erstellten Programm-Dateien auch die Installationsdateien für einen Interbase-Datenbankserver und – falls auf dem Zielsystem noch nicht vorhanden – die Installationsdateien für einen Apache-Webserver.

Diese Installationsdateien liegen jeweils in einer Windows- und einer Linux-Version vor.

## **2. Softwarevoraussetzungen**

### **2.1 Webserver**

Als Webserversoftware ist ein Apache-Webserver zu installieren. Vorkonfigurierte Programmpakete befinden sich auf der Installations-CD. So Sie noch keinen Apache-Webserver installiert haben, wählen Sie aus dem Ordner "Apache" die für Ihr Betriebssystem geeignete Version und installieren Sie diese auf dem von Ihnen gewählten Zielsystem.

### **2.2. Datenbankserver**

Als Datenbankserversoftware ist die frei verfügbare Version Interbase 6 von Borland bzw. die von der Open-Source-Gemeinschaft weiterentwickelte Version mit dem Namen Firebird zu installieren. Erstgenanntes Programm ist auf der Installations-CD enthalten.

Wählen Sie aus dem Ordner "Interbase" die für Ihr Betriebssystem geeignete Version und installieren Sie diese vollständig auf Ihrem Zielsystem.

Betreiben Sie Datenbank- und Webserver auf getrennten Systemen, so muss der Interbase-Client (als auswählbare Komponente in der Datenbankserversoftware enthalten) zusätzlich auf dem Webserverrechner installiert werden.

### 2.3 Konfigurationsarbeit

Als einzige Konfigurationsarbeit ist eine Änderung in der php.ini im Apache-Programmordner bezüglich der Möglichkeit des Zugriffs auf einen Interbase-Datenbankserver durch den Webserver vorzunehmen.

Konkret: Das auskommentierende Semikolon am Beginn der in der php.ini aufzusuchenden Anweisungszeile

```
";extension=php_interbase.dll"
```

ist zu entfernen. (Apache/bin/php.ini)

## 3. Installation des CMS

Stellen Sie sicher, dass sowohl Datenbankserver als auch Webserver gestartet sind.

### 3.1 Die CMS-Datenbank

Die CMS-Datenbankstruktur muss nicht vom Systemadministrator selbst geschaffen werden. Es genügt, die auf der CD enthaltene Datei "cms.gdb" in den Datenbankordner zu kopieren. Hiermit stehen dem Web-CMS zur einfacheren Weiterarbeit bereits einige Grundkonfigurationen zur Verfügung (z.B. Eintrag des Terminkalenders in der Navigation).

Ein komplett "leeres" Informationsportal kann wie unter 3.2 beschrieben erzeugt werden.

### 3.2 Die CMS-Scriptdateien

Entpacken sie die Datei "cms\_scripts.zip" in Root-Verzeichnis ihres Webservers (normalerweise im Ordner HTDOCS des Ordners Apache)

Um die Datenbankinstallation komplett automatisiert ablaufen zu lassen, rufen Sie über einen Browser folgende Adresse auf, um die Installationsscripte des CMS zu starten:

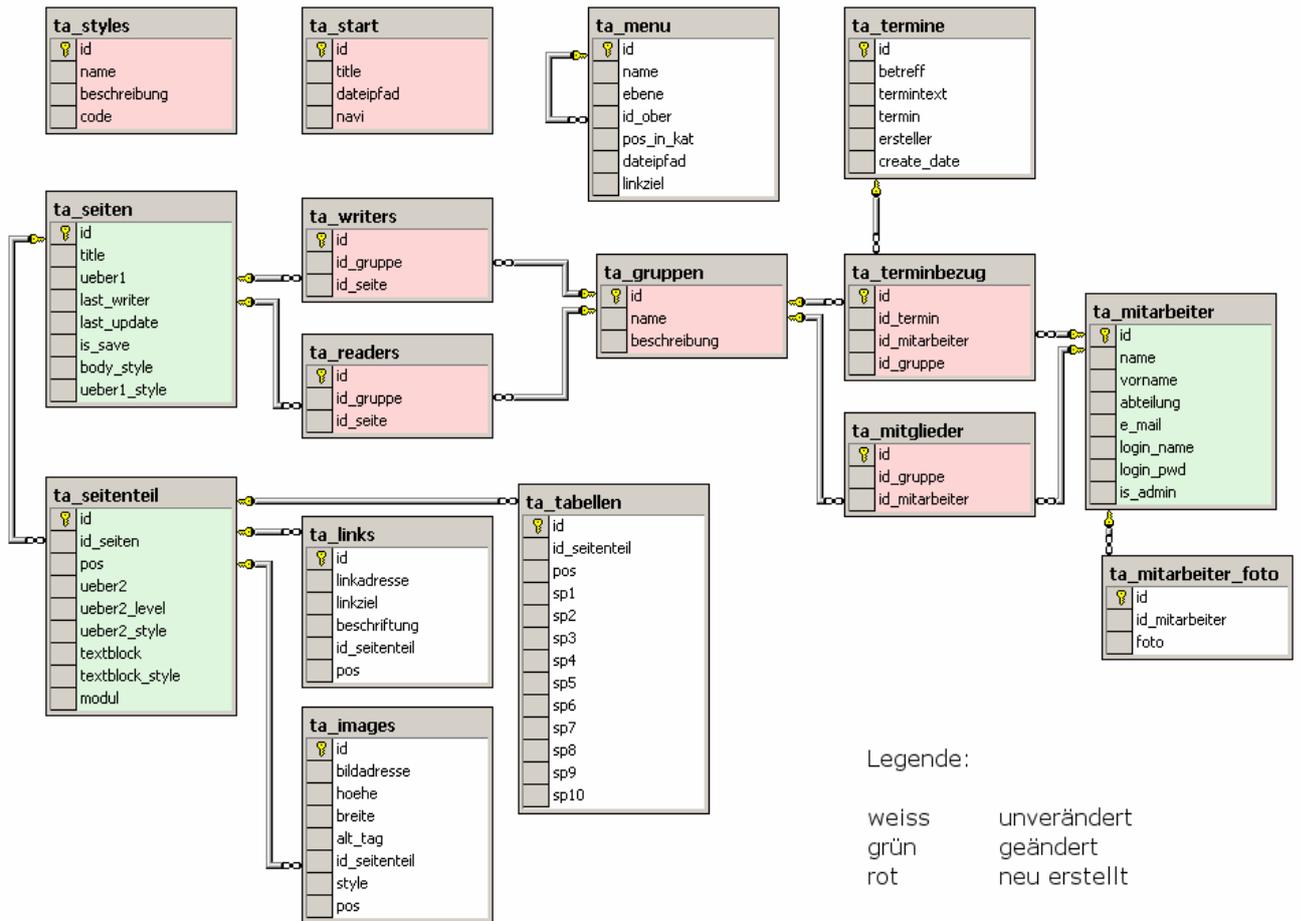
<http://rechnername/install.php> (Passen Sie den Rechnernamen entsprechend an, für den Fall, dass diese Adresse vom Webserverrechner direkt aufgerufen werden soll, kann "rechnername" durch "localhost" ersetzt werden.)

Folgen Sie den Anweisungen der Weboberfläche. Wenn Sie wie unter 3.1. beschrieben eine vorkonfigurierte Datenbankstruktur verwenden, so genügt zum Aufruf des webbasierten Informationsportals die Adresse des Webserverrechners in Ihrem Browser. Beispiel: <http://rechnername>

Nach Aufruf des Links "C-Manager" im Navigationsbereich Ihres Intranets werden Sie zur Eingabe von Benutzernamen und Passwort aufgefordert. Ihnen steht beim Erstzugriff ein Benutzer mit dem Login-Namen "admin" und dem Passwort "admin" zur Verfügung. Aus Gründen der Sicherheit sollten diese Daten in der aufgerufenen Weboberfläche zunächst geändert werden. Die Konfigurationsmöglichkeiten des CMS werden Ihnen nun angezeigt und sind Schritt für Schritt dokumentiert.

**Achtung:** In dem Fall, dass Datenbank- und Webserversoftware auf verschiedenen Rechnern im Netz betrieben werden, ist in der Datei "cms\_dbopen.php" der Eintrag "\$rechner=`localhost`;" auf den entsprechenden Namen des Datenbankserverrechners zu ändern.

# Anlage I: CMS-Datenbankdiagramm



## Anlage J: Quellcode Weboberfläche zur Verwaltung von Gruppenmitgliedschaften

```
1 <?php
2 include_once("../includes/session.php");
3 include_once("../includes/head.php");
4 include_once("../includes/buttons.php");
5 include_once("../includes/cms_dbopen.php");
6 // Einbindung v. Funktion z. Überprüfung des Administrationsrechts, f.
7 // Dateikopf, f. Buttonfunktionen u. f. Datenbankverbindung
8
9
10 //Erzeugt HTML-Kopf mit Title, CSS und Standardüberschrift
11 echo dateikopf("Gruppenzugehörigkeiten ändern")."<body>".baul();
12
13 //erzeugt Variablen mit Werten aus abgesandten Formularen
14 extract($_POST);
15
16 // 1. Formularverarbeitung Entfernen/Hinzufügen
17 //
18 if(isset($plus)) // Wurde "Hinzufügen" gewählt
19
20     // Fügt alle gewählten Mitarbeiter mit ID und Gruppen-ID in Verknüpfungstabelle ein
21     for($i=0;$i<count($id_mitarbeiter);$i++)
22         @ibase_query("insert into ta_mitglieder(id_mitarbeiter, id_gruppe)
23             values ($id_mitarbeiter[$i], $id_gruppe)");
24
25 else if(isset($minus)) // Wurde "Entfernen" gewählt
26     // Löscht alle Verknüpfungen von Gruppen-ID zu den IDs der zu löschenden Mitarbeiter.
27     for($i=0;$i<count($id_mitarbeiter_in);$i++)
28         @ibase_query("Delete from ta_mitglieder
29             where id_mitarbeiter = $id_mitarbeiter_in[$i]
30             and id_Gruppe = $id_gruppe");
31
32
33 // 2. Anzeige auswählbarer Gruppen
34
35 $mit=0; //Schaltvariable überhaupt Gruppen da?
36
37 // wenn keine Gruppe per Formular abgesandt wurde Zuweisung Default-Wert
38 if(!isset($id_gruppe)) $id_gruppe=0;
39
40 // Datenbankabfrage nach allen angelegten Gruppen mit Beschreibung
41 $erg=ibase_query("select id, name, beschreibung
42     from ta_gruppen
43     order by name");
44
45 //Formular sendet Daten an sich selbst
46 echo "<form action=\"\$PHP_SELF\" method=\"POST\" name=\"formular\"><p>";
47
48 // Start Anzeige vorhandener Gruppen mit Radiobutton,
49 // ist keine Gruppen-ID an das Formular übergeben worden,
50 // wird 1. Gruppe vorselektiert, sonst die der übergebenen Gruppen-ID
51 // wenn Keine Gruppen-ID per Formular übergeben wurde, dann wird
52 //alphabetisch erste Gruppe zur Anzeige der Mitgliedschaften verwandt,
53 //in jedem Fall werden ID, Name und Beschreibung der selektierten
54 //Gruppe für spätere Anzeige der Mitgliedschaften gespeichert.
55 while($row=ibase_fetch_row($erg))
56 {
57     if(!$mit) $mit=1; // Ja, es gibt Gruppen.
58 }
```

## Anlage J: Quellcode Weboberfläche zur Verwaltung von Gruppenmitgliedschaften

```
59     echo "<input type=\"radio\" name=\"id_gruppe\" value=\"\$row[0]\"";
60     if((\$id_gruppe==\$row[0]) || (!$mit && !$id_gruppe))
61     {
62         $id_gruppe=$row[0];
63         $gruppenname=$row[1];
64         $gruppenbeschreibung=$row[2];
65         echo " checked";
66     }
67     echo " onClick=\"this.checked=true; window.document.formular.submit();\""; //
Sofortversand
68     echo " \$row[1]"; //Anzeige Gruppennamen
69     echo $row[2] ? " ($row[2])<br>" : "<br>"; //wenn Gruppenbeschreibung belegt, dann Anzeige in Klammern.
70 }
71
72 echo $mit ? weiter("Gruppe wählen", "W", "")
73     . " " . action("Zurück zum Administrationsbereich", "Z", "index.php")
74     . " " . action("Überblick über die Gruppenzugehörigkeiten", "Ü",
"../content/mitarbeiterfotos/gruppen.php")
75     . "</p></form>"
76     : "Bisher sind keine Gruppen angelegt!</p></form>";
77
78
79 // 3. Anzeige der (Nicht-)Mitgliedschaften zur ausgewählten Gruppe
80 if(isset($id_gruppe)) // Gibt es überhaupt Gruppen?
81 {
82     echo "<h2>Zugehörigkeiten in Gruppe \"\$gruppenname\"";
83     echo $gruppenbeschreibung? " ($gruppenbeschreibung)</h2>": "</h2>";
84
85     //Abfrage der Nicht-Mitglieder
86     $erg=ibase_query("select id, vorname, name
87         from ta_mitarbeiter
88         where id not in ( select id_mitarbeiter
89             from ta_mitglieder
90             where id_gruppe=$id_gruppe)
91         order by vorname");
92
93     //Abfrage der Mitglieder
94     $erg2=ibase_query("select m.id, m.vorname, m.name
95         from ta_mitarbeiter m
96         inner join ta_mitglieder mg
97         on m.id=mg.id_mitarbeiter
98         where mg.id_gruppe = $id_gruppe
99         order by m.vorname");
100
101     //Anzeige Formular zur Mitgliederverwaltung
102
103     // Formular wird an aufrufende Seite gesandt
104     // Gruppen-ID wird "versteckt" mit Formular versandt.
105     echo "<form action=\"\$PHP_SELF\" method=\"POST\">";
106     echo "<input type=\"hidden\" name=\"id_gruppe\" value=\"\$id_gruppe\">";
107     echo "<table border=\"1\" width=\"100%\"><tr>";
108     echo "<td align=\"left\">";
109
110     // Ausgabe Nichtmitglieder in Mehrfach-Auswahlliste
111     echo "<h2>Nichtmitglieder:</h2><br><br>";
112     echo "<select class=\"vollnamen\" name=\"id_mitarbeiter[]\" size=\"5\" multiple>";
113     while($row=ibase_fetch_row($erg))
114         echo "<option value=\"\$row[0]\">$row[1] $row[2]</option>";
```

