

Fachhochschule St. Pölten

Erstellung einer webbasierten
Applikation zur zentralisierten
Verwaltung und Steuerung von
Webprojekten

Diplomarbeit

am Fachgebiet Telekommunikation und Medien
Institut für Medieninformatik
Fachhochschule St. Pölten

vorgelegt von

Wolfgang SPITZ, BSc.

Matrikelnummer: tm0710262062

Betreuer: Dipl.-Ing. (FH) Fritz GRABO
Zweitbegutachter: Dipl.-Ing. Grischa SCHMIEDL

Erklärung der Urheberschaft

Ich erkläre hiermit an Eides statt, dass ich die vorliegende Arbeit ohne Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe; die aus fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht. Die Arbeit wurde bisher in gleicher oder ähnlicher Form in keiner anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht.

Ort, Datum

Unterschrift

Kurzfassung

Durch die rasante Entwicklung der Webtechnologien und den daraus resultierenden Möglichkeiten ganze Applikationen und Inhalte im Web zu präsentieren, sowie dem Bedarf Informationen schnell und einfach adaptieren zu können, stiegen die technischen und organisatorischen Anforderungen an die zahlreichen Internetplattformen, Websites und Webapplikationen in den letzten Jahren. Um diesen Anforderungen gerecht zu werden, verwenden viele Unternehmen Web Content Management Systeme, die auf der einen Seite eine Arbeitserleichterung darstellen, allerdings auf der anderen Seite administrative Aufwände verursachen.

Diese Arbeit beschäftigt sich mit der Analyse dieser administrativen Aufwände innerhalb bestimmter Prozesse, bei der Verwaltung und Steuerung von Webprojekten, anhand konkreter Fallbeispiele einer New Media Agentur. Ziele dieser Arbeit sind wiederum, die Schwachstellen der Prozesse durch die Analyse aufzuzeigen, einen Lösungsansatz zu konzipieren, dieses Konzept in Form einer Softwarelösung als Prototyp umzusetzen und anschließend die Ergebnisse zu bewerten.

Abstract

Because of the fast-paced webtechnologies and the resultant opportunities to present whole applications and content in the web as well as the rising demand to adapt informations in a simple and fast manner, the technical and organisational requirements increased tremendously for lots of internet platforms, websites and web applications in the last years. To make the grade, lots of organisations are using web content management systems, which facilitate everyday routines on the one hand but cause administrative costs on the other hand.

This thesis deals with the analysis of administrative costs inside of certain processes at the administration and controlling of webprojects, on the basis of concrete examples of a new media agency. The goals of this thesis are to demonstrate the weaknesses of the processes, to design a solution, to develop a software solution as a prototype and finally to evaluate the results.

Vorwort

Diese Diplomarbeit ist im Zuge meines Masterstudiums an der Fachhochschule St.Pölten im Studiengang für Telekommunikation und Medien entstanden. Durch meine Affinität zu meinem gewählten Themenschwerpunkt Webtechnologien, während meines Studiums und durch mein persönliches Interesse an diesem Bereich, ist dies bereits die zweite Arbeit zu diesem Themenbereich. Dadurch zeichnet sich auch meine besondere Motivation bei der Umsetzung dieser Arbeit aus.

In dieser Diplomarbeit wird aus Gründen der besseren Lesbarkeit auf Paarformulierungen verzichtet und grundsätzlich die männliche bzw. geschlechtsneutrale Formulierung gewählt.

Danksagungen

In erster Linie möchte ich der Geschäftsführung und den Mitarbeitern der BULLDOGgroup GmbH recht herzlich danken, da ohne deren Unterstützung eine derartige Umsetzung dieser Arbeit nicht möglich gewesen wäre. Weiters danke ich meinem Diplomarbeitsbetreuer, der mir stets mit Rat und Tat zur Seite stand.

Zu guter Letzt danke ich besonders meiner Mutter und ihrem langjährigen Partner, die mich in all den langen Jahren meiner Ausbildung und auch während meines Studiums unterstützt und mir in allen wichtigen Lebenslagen stets zur Seite gestanden sind. Sowie danke ich meiner Partnerin, die mich geduldig und motivierend, vor allem bei der Umsetzung dieser Arbeit, unterstützt hat.

Danke!

*„Die Neugier steht immer an erster Stelle eines Problems,
das gelöst werden will.“ (Galileo Galilei, 1564-1642)*

Inhaltsverzeichnis

| | | |
|----------|---|-----------|
| 1 | Einleitung | 1 |
| 2 | Problemstellung | 3 |
| 2.1 | Problembeschreibung | 3 |
| 2.2 | Problemrelevanz | 4 |
| 2.3 | Problemfelder | 7 |
| 2.3.1 | Mangel an Zeit | 7 |
| 2.3.2 | Mangel an Arbeitskräften | 8 |
| 2.3.3 | Mangel an Know-How | 8 |
| 2.3.4 | Awareness des Entscheidungsträgers | 8 |
| 2.3.5 | Kosten | 8 |
| 2.4 | Beschreibung der New Media Agentur | 10 |
| 3 | Organisatorische Analyse | 11 |
| 3.1 | Erläuterung der Fallbeispiele | 11 |
| 3.2 | Analyse der Arbeitsweise an den konkreten Fallbeispielen | 11 |
| 3.2.1 | Redaktionelle Betreuung von Websites | 12 |
| 3.2.2 | Anlegen einer neuen Website inklusive vorab installierten WCMS | 14 |
| 3.3 | Ergebnisse und Interpretation der Analyse | 16 |
| 3.3.1 | Interpretation des Prozesses - redaktionelle Betreuung einer Website | 16 |
| 3.3.2 | Interpretation des Prozesses - Anlegen einer neuen Website mit vorab installierten WCMS | 18 |
| 4 | Entwicklung des Lösungsansatzes | 20 |
| 4.1 | Zentralisierung mehrerer Webprojekte | 20 |
| 4.2 | Steuerung von administrativen Tätigkeiten am Server | 23 |

Inhaltsverzeichnis

| | | |
|----------|--|-----------|
| 4.3 | Freie Erweiterbarkeit / Pluginfähigkeit | 25 |
| 5 | Praktische Umsetzung | 27 |
| 5.1 | Erläuterung der eingesetzten Technologien | 27 |
| 5.1.1 | Betriebssystem UBUNTU Server | 27 |
| 5.1.2 | Apache Webserver | 28 |
| 5.1.3 | Datenbanksystem MySQL | 29 |
| 5.1.4 | FTP-Server ProFTPD | 30 |
| 5.1.5 | PHP | 30 |
| 5.1.6 | Web Content Management System - TYPO3 | 31 |
| 5.2 | Erweiterbarkeit / Pluginfähigkeit | 33 |
| 5.3 | Beschreibung der Verwaltungsoberfläche | 36 |
| 5.4 | Zentralisierung der Redaktionsoberflächen mehrerer Webprojekte | 38 |
| 5.5 | Steuerung von administrativen Tätigkeiten am Server | 42 |
| 5.5.1 | Anlegen eines neuen Webspace | 43 |
| 5.5.2 | Anlegen eines neuen FTP-Zuganges | 45 |
| 5.5.3 | Anlegen einer neuen MySQL Datenbank | 47 |
| 5.6 | Beschreibung des Plugins „ <i>TYPO3</i> “ | 50 |
| 5.6.1 | Importieren einer bestehenden TYPO3-Installation | 50 |
| 5.6.2 | Sichern einer bestehenden TYPO3-Installation | 54 |
| 5.6.3 | Installieren eines neuen TYPO3-Projekts | 56 |
| 6 | Ergebnisse | 62 |
| 6.1 | Redaktionelle Betreuung von Websites | 62 |
| 6.2 | Anlegen einer neuen Website mit vorab installierten WCMS | 65 |
| 7 | Erweiterungsmöglichkeiten | 68 |
| 7.1 | zentralisierte Benutzerverwaltung | 68 |
| 7.2 | Post-Installationen von fehlender Software | 69 |
| 7.3 | Erweiterungen für das Plugin TYPO3 | 71 |
| 7.3.1 | Wahl von unterschiedlichen TYPO3-Versionen | 71 |
| 7.3.2 | Wahl der Kompatibilitätsversion durch den Benutzer | 72 |
| 7.3.3 | Zuweisung von Benutzern und Benutzergruppen beim Import-Modul | 73 |
| 7.4 | Durchführung von Modulen auf externen Serversystemen | 74 |

Inhaltsverzeichnis

| | |
|---|-------------|
| 7.5 Sichere Datenübertragen durch HTTPS | 75 |
| 7.6 Weitere mögliche technische Erweiterungen | 75 |
| 8 Conclusio | 77 |
| Abbildungsverzeichnis | XI |
| Tabellenverzeichnis | XIII |
| Verzeichnis der Listings | XIV |
| Literaturverzeichnis | XV |

Kapitel 1

Einleitung

Durch die rasante Entwicklung der Webtechnologien und den daraus resultierenden Möglichkeiten ganze Applikationen und Inhalte im Web zu präsentieren, stiegen ebenfalls die technischen und organisatorischen Anforderungen an die zahlreichen Internetplattformen, Websites und Webapplikationen in den letzten Jahren enorm. Durch diese Anforderungen und Präsentationsmöglichkeiten stehen sowohl Webentwickler, als auch Redakteure von Webprojekten neuen Herausforderungen gegenüber. Web Content Management Systeme (WCMS)¹ helfen Redakteuren und Webentwicklern diese technischen und organisatorischen Herausforderungen besser zu bewältigen. Durch diese Entwicklungen und Anforderungen nimmt der Einsatz von WCMS ständig zu.

Heutzutage verwenden „New Media Agenturen“² vermehrt WCMS, da diese die täglichen Wartungs- und Redaktionsarbeiten eines Webauftritts erheblich erleichtern. Durch den steigenden Einsatz dieser Systeme steigt allerdings auch die Zahl der Administrationsoberflächen und somit der administrative und organisatorische Aufwand in New Media Agenturen. Dieser steigende Mehraufwand bindet Arbeitskräfte und Arbeitszeit und verzögert die eigentlichen Tätigkeiten.

¹Ein Web Content Management System (WCMS) ist ein System zur Verwaltung der Inhalte von Websites bzw. ein Framework zur Konstruktion eines solchen(vgl. [KL06], S. 31).

²New Media Agenturen bieten Dienstleistungen im Umgang mit den neuen Medien, wie dem Internet, Mobile Kommunikation, etc., an.

Kapitel 1 Einleitung

Ziel dieser Arbeit ist es, diese eben beschriebenen Mehraufwände, anhand einer Fallstudie aus der Praxis zu analysieren, einen Lösungsansatz in Form einer Software zu finden und prototypisch umzusetzen. Dieser Lösungsansatz soll die administrativen und organisatorischen Mehraufwände minimieren und damit verbundene Arbeitsabläufe im Bereich des Webhostings³ optimieren.

³Unter Webhosting versteht man den zur Verfügung gestellten Speicherplatz und die darin befindlichen Daten, durch einen Internet Service Provider und die Überwachung durch diesen. (vgl. [Und01], S. 2)

Kapitel 2

Problemstellung

2.1 Problembeschreibung

Neben den Vorteilen, die Web Content Management Systeme mit sich bringen, entstehen aber parallel dazu, durch die Administrations- und Redaktionsoberflächen, sowie durch die damit verbundenen Arbeitsvorgänge im Bereich des Webhostings, administrative und organisatorische Tätigkeiten. Diese Tätigkeiten stellen Mehraufwände dar, die Arbeitskräfte und Arbeitszeit binden und somit die Durchführung der konstruktiven Tätigkeiten verzögern.

Die Ursache für einen Teil dieser Problematik liegt in der dezentral organisierten Administration von Websites und Webapplikationen vieler New Media Agenturen, wodurch es zu Verzögerungen bei der redaktionellen Betreuung von Webprojekten, sowie bei der Durchführung von Tätigkeiten rund um den Bereich des Webhostings kommt. Diese Tätigkeiten werden zwar durch diverse Softwaretools unterstützt, da aber meist eigene Serverbetreuer für das Webhosting verantwortlich sind, kommt es eben hier oft zu unerfreulichen Stehzeiten aufgrund unternehmensübergreifender Kommunikationswege.

Um diesen Problemen entgegen zu wirken, werden in den folgenden Kapiteln, an konkreten Fallbeispielen einer New Media Agentur, ganze Prozesse graphisch dargestellt und analysiert und gewisse Arbeitsschritte innerhalb der Prozesse genau beschrieben. Aus den Ergebnissen der Analyse dieser Prozesse wird ein Lösungsansatz in Form einer Software erarbeitet und prototypisch

Kapitel 2 Problemstellung

umgesetzt. Diese Software soll dabei helfen die administrativen und organisatorischen Aufwände zu minimieren und die damit verbundenen Tätigkeiten des Webhostings zu optimieren.

Auf Grund der soeben beschriebenen Problemstellung ergibt sich die folgende forschungsleitende Fragestellung dieser Arbeit: Welche organisatorischen und technischen Maßnahmen sind, für die Minimierung der administrativen und organisatorischen Mehraufwände, bei der Betreuung von Websites und Webapplikationen, sowie für die Optimierung der Tätigkeiten im Bereich des Webhostings, notwendig?

2.2 Problemrelevanz

Die folgenden Abbildungen bestätigen, die in Kapitel 2.1 beschriebenen Entwicklungen. Abbildung 2.1 zeigt die stark ansteigende Zahl von Hostnamen und aktiven Websites in den Jahren von November 1995 bis Februar 2009. Vor allem die steigende Anzahl von 216 Millionen aktiven Websites im November 2003 auf rund 800 Millionen aktiven Websites im Februar 2009, zeigt die stetig wachsende Bedeutung des World Wide Web.

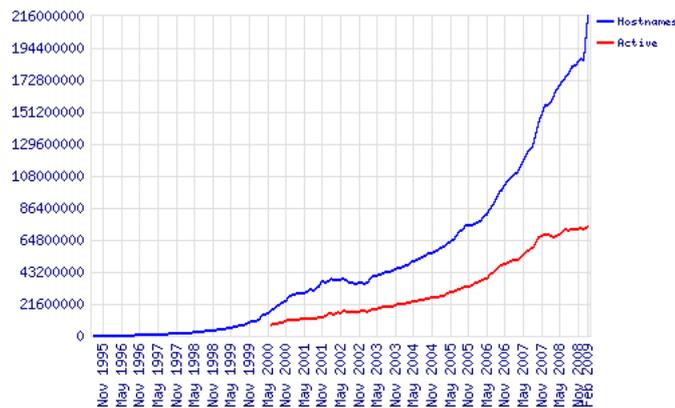


Abbildung 2.1: Anzahl Hostnamen und aktiver Websites von August 1995 bis Februar 2009 (vgl. [Net09])

Kapitel 2 Problemstellung

Abbildung 2.2 zeigt den steigenden prozentuellen Anteil der Unternehmen und der Haushalte mit einer Breitband-Internetverbindung in den Jahren 2001 bis 2006. Die grundlegenden Daten zu Abbildung 2.2 ergeben sich aus den hochgerechneten Ergebnissen der 6. Europäischen Erhebung über den IKT-Einsatz¹ und E-Commerce² 2005/2006 in Unternehmen und der 5. Europäischen Erhebung über den IKT-Einsatz in Haushalten 2006. Die beiden Erhebungen wurden von Statistik Austria mit einem europaweit einheitlichen Fragenprogramm, standardisierten Definitionen und einer einheitlichen Methodik in Form einer Stichprobenerhebung durchgeführt. Rund 3.300 Unternehmen mit mehr als 9 Beschäftigten (aus den Bereichen Sachgütererzeugung, Bauwesen, Handel, Beherbergungswesen, Verkehr und Nachrichtenübermittlung, Kredit- und Versicherungswesen, unternehmensbezogene Dienstleistungen und Film- und Videoherstellung) und ca. 5.300 Haushalte mit zumindest einem Haushaltsmitglied im Alter von 16 bis 74 Jahren haben an den Erhebungen teilgenommen (vgl. [Aus09]).

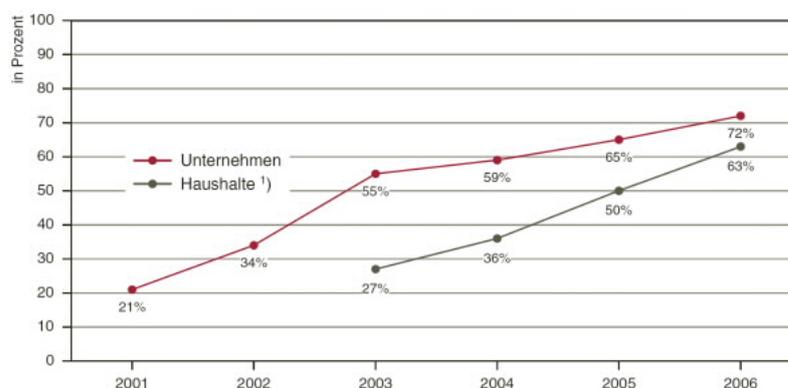


Abbildung 2.2: Prozentueller Anteil der Unternehmen und der Haushalte mit Breitbandverbindung 2001-2006 (vgl. [Aus09])

¹Die Abkürzung IKT steht für Informations- und Kommunikationstechnologie. Unter diesem Begriff sind alle Technologien aus den Bereichen Information und Kommunikation zusammengefasst (vgl. [Ser05], S.2ff).

²E-Commerce ist ein Konzept zur Nutzung von bestimmten Informations- und Kommunikationstechnologien zur elektronischen Integration und Verzahnung unterschiedlicher Wertschöpfungsketten oder unternehmensübergreifender Geschäftsprozesse und zum Management von Geschäftsbeziehungen (vgl. [Fri], Abruf 12.04.2009).

Kapitel 2 Problemstellung

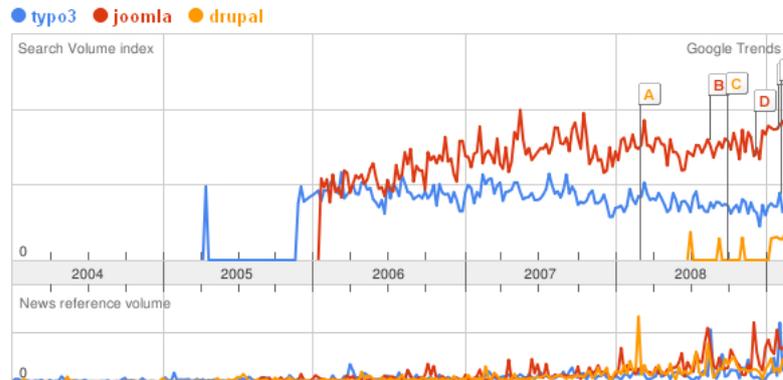


Abbildung 2.3: Google Trends Analyse der Suchanfragen zu Typo3, Joomla und Drupal ([Goo], Abruf 12.04.2009)

Abbildung 2.3 verdeutlicht die steigende Bedeutung und den steigenden Einsatz von Web Content Management Systemen. Mit Hilfe von Google Trends³ werden Suchanfragen über drei Open-Source WCMS in Österreich in einem Zeitraum vom Jahr 2004 bis zum Jahr 2009 graphisch dargestellt.

³Google Trends gibt Auskunft über die Verteilung bestimmter Suchanfragen über einen gewissen Zeitraum hinweg.

2.3 Problemfelder

Da der zu erarbeitende Lösungsansatz mit anderen Software-Projekten zu vergleichen ist, steht dieser ebenfalls ähnlichen organisatorischen Problemfeldern gegenüber. Grundsätzlich ist ein Software-Projekt, wie auch Projekte aus anderen Branchen, durch die folgenden Projekteigenschaften gekennzeichnet:

- inhaltliche Zielsetzung und Neuartigkeit
- zeitliche Zielsetzung
- beschränkte Ressourcen
- Komplexität

Darüber hinaus existiert allerdings ein entsprechend erhöhtes Risiko bei Software-Projekten hinsichtlich Technik, Zeitdruck, Organisation, Personal und Finanzierung (vgl. [Ber03], S. 2-6).

Besonders bei Software-Projekten, die für das eigene Unternehmen entwickelt werden sollen, treten verstärkt die folgenden Problemfelder bei der technischen Umsetzung auf:

- Mangel an Zeit
- Mangel an Arbeitskräften
- Mangel an Know-How
- Awareness des Entscheidungsträgers
- Kosten

2.3.1 Mangel an Zeit

Der Zeitmangel ergibt sich vermehrt durch ungenaue Spezifikationen und Pflichtenhefte zu Beginn eines Software-Projekts und durch das Vorziehen von Tätigkeiten für Kunden gegenüber internen Projekten.

2.3.2 Mangel an Arbeitskräften

Zu wenige Arbeitskräfte ergeben sich durch, die in anderen Projekten bereits eingesetzt und knapp bemessene Zahl an Projektmitgliedern, die für die Entwicklung zuständig sind. Dieses Problemfeld führt wiederum zu Verzögerungen im gesamten Projektablauf und beeinflusst wiederum den Zeitfaktor, da Arbeitskräfte fehlen, die das Projektcontrolling durchführen.

2.3.3 Mangel an Know-How

Der Mangel an Know-How ist ein wesentlicher Faktor, der vor allem die Problemfelder Zeit und Kosten beeinflusst. Dieses Problemfeld ist auf mangelnde Schulungen zurückzuführen, die wiederum aus Zeit- und Kostengründen vom Arbeitgeber nicht ermöglicht werden können.

2.3.4 Awareness des Entscheidungsträgers

Die Awareness des Entscheidungsträgers beeinflusst alle weiteren Problemfelder, da von ihm die Entscheidung abhängt, ob ein Software-Projekt umgesetzt wird oder nicht.

2.3.5 Kosten

Die Kosten spielen wie in jedem anderen Projekt die wichtigste Rolle. Allerdings sind sie in internen Projekten nur schwer ersichtlich und treten fälschlicherweise in den Hintergrund.

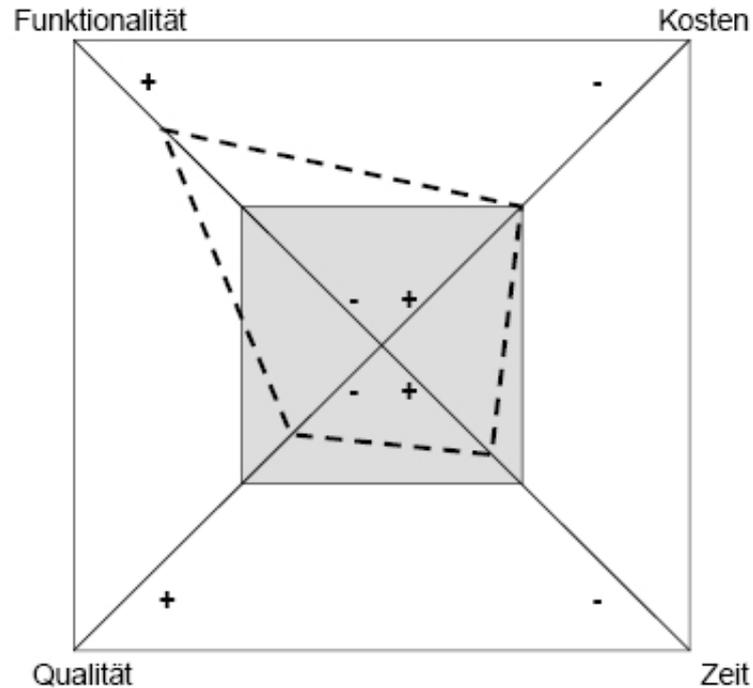


Abbildung 2.4: Einfluss und Auswirkungen von Zeit und Kosten auf die Qualität und Funktionalität in Software-Projekten (vgl [Ber03], S. 4)

All diese Problemfelder beeinflussen sich gegenseitig und verhindern ein Umdenken in der Arbeitsweise vieler New Media Agenturen. Abbildung 2.4 stellt den Einfluss der beiden wesentlichsten Problemfelder, Zeit und Kosten und deren Auswirkungen graphisch dar.

Neben den soeben beschriebenen Problemfeldern, die durch die Eigenentwicklung bei der technischen Umsetzung entstehen, ist speziell bei der Umsetzung der geplanten Softwarelösung, der technische Aufwand für alle Projektmitglieder als sehr hoch einzustufen. Deshalb ist es relevant für die Wirtschaft, die im Zuge dieser Diplomarbeit entwickelte Webapplikation, in weiterer Folge als Installationspaket (siehe Kapitel 7 - Erweiterungsmöglichkeiten) für spätere Implementierungen und Weiterentwicklungen anzubieten.

2.4 Beschreibung der New Media Agentur

Für diese Arbeit werden konkrete Fallbeispiele der New Media Agentur BULLDOGgroup GmbH⁴ beschrieben, graphisch dargestellt und analysiert. Die BULLDOGgroup GmbH hat Ihren Firmensitz in Eidenberg bei Linz in Österreich und bietet Dienstleistungen in den Bereichen Öffentlichkeitsarbeit, Werbung, Event und Incentive und dem Bereich der Informationstechnologien an.

Die BULLDOGgroup GmbH für Informationstechnologien besteht zur Zeit aus einem neunköpfigen Team. Dieses Team setzt sich aus drei Entwicklern, zwei Softwarearchitekten, zwei Redakteuren und zwei Grafikern zusammen, die in die Prozesse, die im folgenden Kapitel 3 beschrieben und analysiert werden, involviert sind.

⁴Nähere Informationen zur New Media Agentur BULLDOGgroup GmbH unter <http://www.bulldog-group.com>

Kapitel 3

Organisatorische Analyse

Die organisatorische Analyse beschreibt zwei Fallbeispiele für Prozesse in der New Media Agentur und stellt die Prozesse in Form von Ablaufdiagrammen graphisch dar. Anhand der Diagramme werden die Arbeitsschritte innerhalb der Prozesse analysiert.

3.1 Erläuterung der Fallbeispiele

In der IT-Abteilung der BULLDOGgroup GmbH werden zwei Prozesse besonders häufig abgewickelt. Auf der einen Seite stellt die redaktionelle Betreuung von Websites von Kunden einen dieser Prozesse in der New Media Agentur dar, auf der anderen Seite ist das Anlegen von neuen Websites und die Installation des dazugehörigen Web Content Management Systems, ein oft durchgeführter Arbeitsvorgang. Durch das wiederholte Auftreten dieser Prozesse, müssen einzelne analoge Arbeitsschritte innerhalb der Prozesse, immer wieder durchgeführt werden.

3.2 Analyse der Arbeitsweise an den konkreten Fallbeispielen

Kapitel 3.2.1 und 3.2.2 analysieren anhand von Ablaufdiagrammen, die in Kapitel 3.1 erwähnten Prozesse. Durch die Analyse wird versucht lohnende

Kandidaten für die Automatisierung von Arbeitsschritten an der momentanen Arbeitsweise bei diesen Prozessen aufzuzeigen.

3.2.1 Redaktionelle Betreuung von Websites

Die redaktionelle Betreuung von Websites stellt einen Prozess der Fallstudie dar. Abbildung 3.1 veranschaulicht die einzelnen Arbeitsschritte des Prozesses.

Der Prozess beginnt mit der Auftragserteilung für Wartungsarbeiten, wie beispielsweise der Aktualisierung von Inhalten einer Website, durch den Auftraggeber via E-Mail oder Telefon. Der Redakteur ruft nun die ihm bekannte Redaktionsoberfläche des Web Content Management Systems auf. Sind ihm, die zur Anmeldung erforderlichen Zugangsdaten bekannt, kann er sich am System anmelden und mit den Adaptierungsarbeiten beginnen.

Andernfalls muss der Redakteur die Zugangsdaten beim zuständigen Entwickler anfordern. Die Entwickler verwenden eine gemeinsame Excel-Liste¹ für die Verwaltung der Zugangsdaten für die Webprojekte der New Media Agentur. Nachdem der Entwickler nun diese Liste durchsucht hat, übermittelt er dem Redakteur die Zugangsdaten via E-Mail oder Telefon.

Nun kann sich der Redakteur an der Redaktionsoberfläche des Web Content Management Systems anmelden und mit seiner eigentlichen Arbeit beginnen. Nach Durchführung der in Auftrag gegebenen Adaptierungen kontrolliert der Redakteur am Frontend² seine Arbeiten. Die Kontrolle der Adaptierungen durch den Auftraggeber stellt den letzten Prozess- bzw. Arbeitsschritt innerhalb des Prozesses dar.

¹Microsoft Excel ist Tabellenkalkulationsprogramm, das seine Benutzer bei der Analyse, Freigabe und Verwaltung von Informationen unterstützt (vgl. [Mic09], Abruf 12.04.2009).

²Als Frontend des Webauftrittes wird die eigentliche Webpräsentation bezeichnet, also die Ansicht, die ein Besucher der Website sieht ([Sto04] Stöckl 2004, S.42)

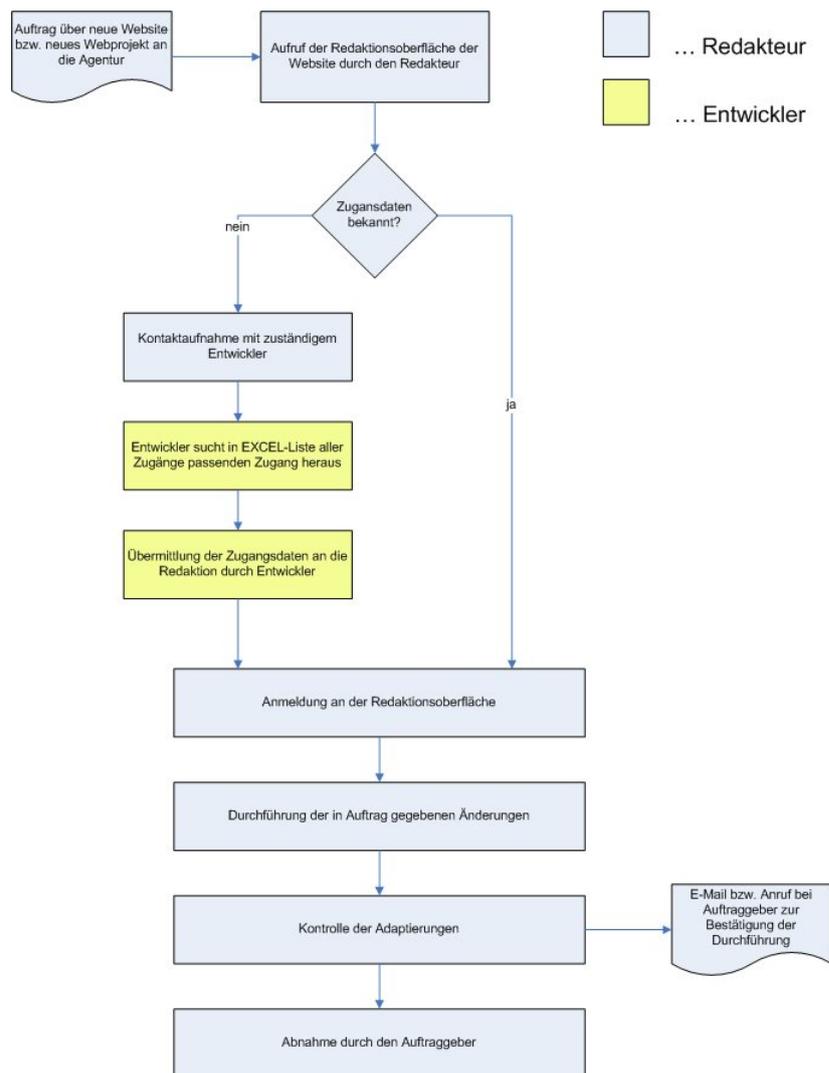


Abbildung 3.1: Prozessablaufdiagramm der redaktionellen Betreuung einer Website

3.2.2 Anlegen einer neuen Website inklusive vorab installierten WCMS

Das Anlegen einer neuen Website, die bereits ein Web Content Management System vorab installiert hat, stellt einen weiteren Prozess in der Fallstudie dieser Arbeit dar. Abbildung 3.2 veranschaulicht wiederum die einzelnen Arbeitsschritte des Prozesses.

Der Prozess wird durch die Auftragserteilung, vom Kunden an den Projektleiter, über die Erstellung einer neuen Website oder über ein anderes Webprojekt initialisiert. Der Projektleiter stellt sein Team zusammen und erteilt dem zuständigen Entwickler den Auftrag, für das neue Webprojekt, Speicherplatz auf dem Server zu reservieren und nach seinem Ermessen ein passendes Web Content Management System auszuwählen und zu installieren.

Der Entwickler erteilt dem zuständigen Serverbetreuer via E-Mail den Auftrag für das neue Webprojekt, entsprechenden Webspace, einen FTP-Zugang zu erstellen und eine Datenbank anzulegen. Die Zugangsdaten für den Webspace und die Datenbank werden nach Abschluss der Arbeiten des Serverbetreuers anschließend von diesem an den Entwickler wiederum via E-Mail übermittelt.

Nach Übermittlung der Zugangsdaten an den Entwickler, kann dieser nun die Installation des Web Content Management Systems manuell durchführen. Anschließend bestätigt der Entwickler dem Projektleiter die Durchführung und übergibt ihm ebenfalls alle notwendigen Zugangsdaten.

Kapitel 3 Organisatorische Analyse

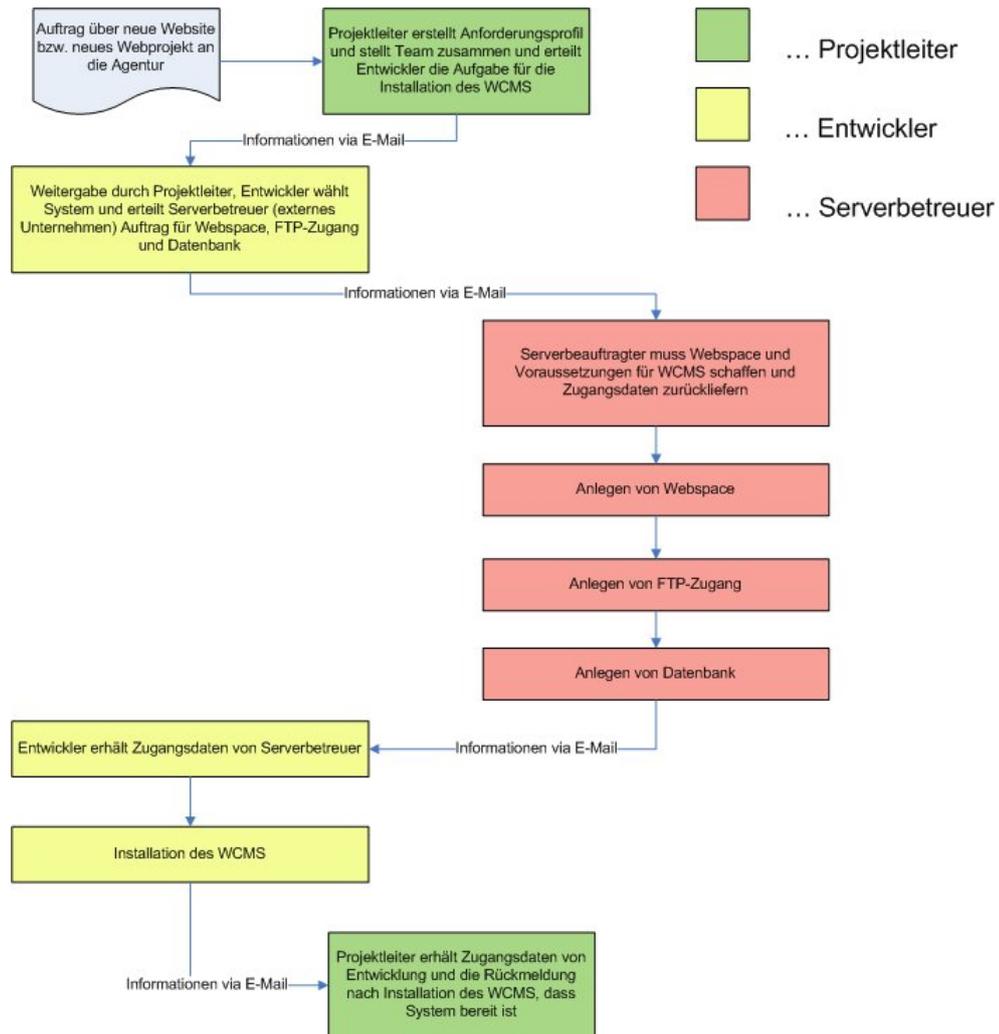


Abbildung 3.2: Prozessablaufdiagramm für das Anlegen einer neuen Website inklusive vorab installierten WCMS

3.3 Ergebnisse und Interpretation der Analyse

Die in Kapitel 3.2 beschriebenen Prozesse weisen deutliche Schwachstellen auf. Durch diese Schwachstellen verzögern sich einzelne Arbeitsschritte und somit der gesamte Prozess. Bei einigen Arbeitsschritten fallen verstärkt administrative Aufwände an, die ein hohes Optimierungspotenzial aufweisen.

3.3.1 Interpretation des Prozesses - redaktionelle Betreuung einer Website

Der Prozess „*die redaktionelle Betreuung einer Website*“ als exemplarisches Fallbeispiel der New Media Agentur beinhaltet zwei große Schwachstellen. Die erste Schwachstelle stellt den Einsatz eines weiteren Mitarbeiters dar, falls die erforderlichen Zugangsdaten für die Redaktionsoberfläche dem Redakteur nicht bekannt sind. Durch das Anfordern der Zugangsdaten via E-Mail können Verzögerungen auftreten, da E-Mails nicht sofort gelesen werden müssen bzw. auch irrtümlich gelöscht werden können. Die Zugangsdaten können auch telefonisch vom Redakteur angefordert werden, allerdings muss auch hier der Entwickler erreichbar sein und die erforderlichen Daten übermitteln. Aus diesen Gründen stellt der Einsatz von weiteren Personen bzw. Mitarbeitern in diesem Prozessschritt somit die erste Schwachstelle und eine deutliche Verzögerung des gesamten Prozesses dar.

Die zweite Schwachstelle im Prozessablauf tritt erst durch den wiederholten Vorgang des gesamten Prozesses auf. Durch die oft unvermeidbaren hohen Antwortzeiten bei der Kommunikation zwischen Redakteuren und Entwicklern beim Anmeldevorgang, entstehen in Summe, speziell bei diesen Arbeitsschritten, Verzögerungen, durch die der Redakteur wertvolle Zeit verliert, die dieser für Adaptierungen an Websites, nutzen hätte können.

Abbildung 3.3 zeigt die Anzahl und die Dauer der Anmeldevorgänge bei der redaktionellen Betreuung. Die zugrundeliegenden Daten wurden von den Redakteuren der BULLDOGgroup im Zeitraum vom 1. Februar 2009 bis zum 28. Februar 2009 erstellt, die jeden Anmeldevorgang in fünf unterschiedliche Kategorien einteilen mussten. Diese Kategorien beschreiben die Dauer eines

Kapitel 3 Organisatorische Analyse

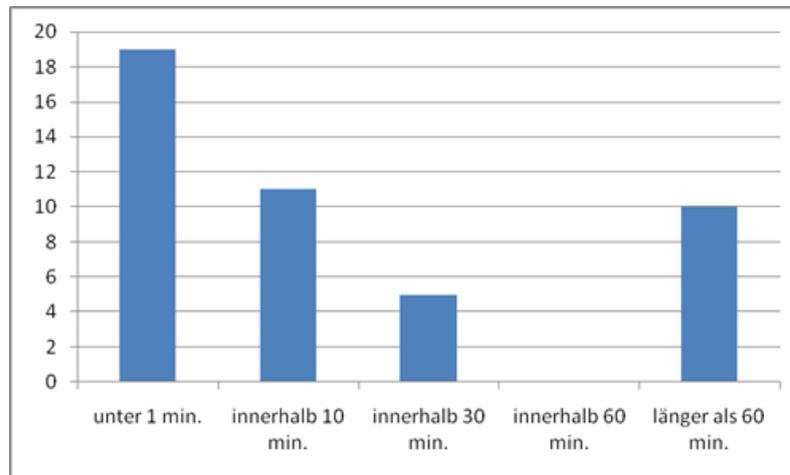


Abbildung 3.3: Anzahl und Dauer der Anmeldevorgänge für die redaktionelle Betreuung im Beobachtungszeitraum Februar 2009

Anmeldevorganges, entsprechen für eine spätere Bewertung (siehe Kapitel 6 - Ergebnisse) dem Schulnotensystem und lauten wie folgt:

- unter 1 Minute (Sehr gut)
- zwischen 1 - 10 Minuten (Gut)
- zwischen 10 - 30 Minuten (Befriedigend)
- zwischen 30 - 60 Minuten (Genügend)
- länger als 60 Minuten (Nicht Genügend)

Ein weiterer Nachteil, der sich aus der momentanen Arbeitsweise ergibt, stellt die Verwaltung der Zugangsdaten in Form einer Excel-Liste dar. Im Gegensatz dazu, würde eine zentralisierte, webbasierende Benutzerverwaltung ein erhebliches administratives Einsparungspotenzial darstellen. Diese Benutzerverwaltung könnte in weiterer Folge mit Hilfe von LDAP an ein globales Verzeichnis, wie zum Beispiel an ein Active Directory, angebunden werden und somit müssten Administratoren in Zukunft lediglich eine einzige Datenquelle verwalten (siehe Kapitel 7 - Erweiterungsmöglichkeiten). Allerdings führt die momentane Arbeitsweise dazu, dass viele Entwickler und Administratoren unterschiedliche Datenquellen und somit viele Zugangsdaten an verschiedenen

Speicherorten verwalten müssen. Dies führt zu erhöhten administrativen und auch organisatorischen Aufwänden.

3.3.2 Interpretation des Prozesses - Anlegen einer neuen Website mit vorab installierten WCMS

Der Prozess „*Anlegen einer neuen Website mit vorab installierten WCMS*“ beinhaltet eine große Schwachstelle. Diese Schwachstelle wird durch das Zusammenspiel von drei unterschiedlichen Verantwortlichen, die die einzelnen Arbeitsschritte innerhalb des Prozesses durchführen, und deren Verzögerungen bei der Kommunikation, verursacht.

Speziell bei der Kommunikation zwischen dem Entwickler und dem Serverbetreuer treten im konkreten Fallbeispiel der New Media Agentur vermehrt Verzögerungen auf, da der Serverbetreuer bei einem externen Unternehmen angestellt ist und dadurch bereits andere Kommunikationswege vorgegeben sind. Der Serverbetreuer wird vom Entwickler in der Regel per E-Mail beauftragt, einen neuen Webpace, einen neuen FTP-Zugang und eine neue Datenbank anzulegen. Der Serverbetreuer schafft die Voraussetzungen für den Entwickler, damit dieser dann in weiterer Folge das Web Content Management System installieren kann. Solange diese Basis vom Serverbetreuer allerdings nicht geschaffen wird, kann der zuständige Entwickler auch nicht mit seiner Arbeit fortfahren und es entstehen unerwünschte, aber vermeidbare Verzögerungen.

Abbildung 3.4 stellt die Anzahl der vom Entwickler in Auftrag gegebenen Installationsarbeiten an die externen Serverbetreuer und vor allem die benötigte Zeit der Serverbetreuer für die Durchführung dieser. Die für das Diagramm herangezogenen Daten entstanden durch die Aufzeichnungen der Entwickler im Zeitraum vom 1. Jänner 2009 bis zum 28. Februar 2009. Jeder, vom Entwickler in Auftrag gegebene Installationsvorgang und die dafür benötigte Dauer, wurde in eine der folgenden fünf Kategorien unterteilt:

- innerhalb 1 Stunde (Sehr gut)
- zwischen 1 - 12 Stunden (Gut)
- zwischen 12 - 24 Stunden (Befriedigend)

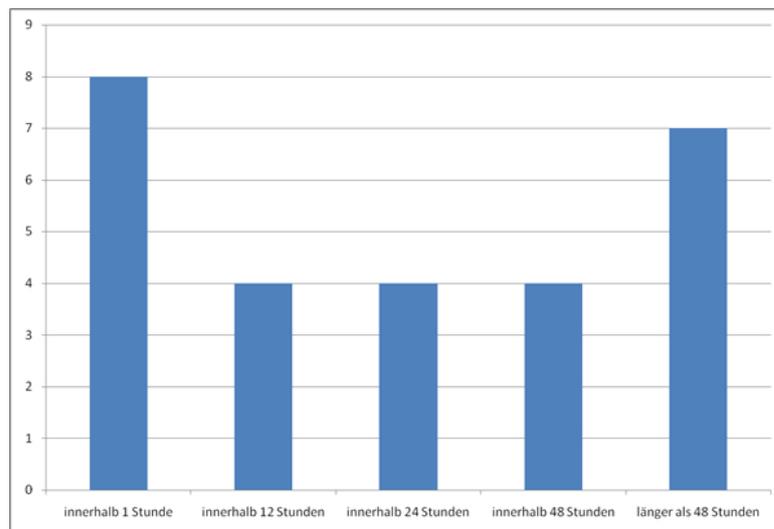


Abbildung 3.4: Anzahl der vom Entwickler in Auftrag gegebenen Arbeiten an die Serverbetreuer und die Dauer für die Durchführung durch diese

- zwischen 24 - 48 Stunden (Genügend)
- länger als 48 Stunden (Nicht Genügend)

Diese Kategorien entsprechen, wie in Kapitel 3.3.1, für eine spätere Bewertung, (siehe Kapitel 6 - Ergebnisse) dem Schulnotensystem.

Durch die Analyse dieses Prozesses ist eindeutig hervorgegangen, dass das Anlegen einer neuen Website mit vorab installierten Web Content Management System bei weitem mehr Optimierungspotenzial besitzt als der Prozess für die redaktionelle Betreuung einer Website. Dies ist eine Schlussfolgerung aus den längeren Verzögerungen durch die unternehmensübergreifende Kommunikation und die höheren Durchführungszeiten der Tätigkeiten an sich bei letzterem Prozess.

Kapitel 4

Entwicklung des Lösungsansatzes

Anhand der gewonnenen Erkenntnisse aus Kapitel 3.3, der Analyse der exemplarischen Prozesse der New Media Agentur, wurde der folgende Lösungsansatz erarbeitet und in weiterer Folge in Form einer Softwarelösung prototypenhaft umgesetzt. Der Lösungsansatz dieser Arbeit unterteilt sich in drei Abschnitte, die in den folgenden Abschnitten erläutert werden:

- Zentralisierung mehrerer Webprojekte
- Steuerung von administrativen Tätigkeiten am Server
- Freie Erweiterbarkeit / Pluginfähigkeit

4.1 Zentralisierung mehrerer Webprojekte

Eine technisch einfache Variante, um die beiden Schwachstellen, die in Kapitel 3.3.1 für den Prozess „*redaktionelle Betreuung einer Website*“ beschrieben wurden, zu schließen, wäre, die Excel-Liste der Entwickler, die die Zugangsdaten zu den einzelnen Redaktionsoberflächen enthält, den Redakteuren auf einem Server zur Verfügung zu stellen. Dies ist allerdings aus organisatorischer Sicht, auf Grund einer Entscheidung der Geschäftsführung, sowie aus Gründen der Sicherheit, da ein Redakteur nur für einen Teil der Webprojekte zuständig ist, am konkreten Fallbeispiel der BULLDOGgroup GmbH, nicht möglich.

Kapitel 4 Entwicklung des Lösungsansatzes

Um nun die beiden beschriebenen Schwachstellen des Prozesses zu schließen und den gesamten Prozessablauf zu optimieren, muss eine übergeordnete Verwaltungsschicht eingezogen werden, die sämtliche Redaktionssysteme zusammenführt und somit zentralisiert. Der Redakteur wird in weiterer Folge nur mehr aufgefordert, sich an der übergeordneten Verwaltungsoberfläche anzumelden und kann automatisch auf alle Redaktionsoberflächen, für deren Websites er zuständig ist, zugreifen. Durch diesen Lösungsansatz muss sich jeder Redakteur nur mehr seine persönlichen Zugangsdaten zur Verwaltungsoberfläche merken und es werden beide Schwachstellen des Prozesses umgangen.

Abbildung 4.1 zeigt den schematischen Aufbau des ersten Abschnittes des Lösungsansatzes, für das Zusammenführen mehrerer Webprojekte in die übergeordnete Verwaltungsschicht. Wie das Schema zeigt wird ein eigenes Modul namens „*Autologin*“ auf die zusammengeführten bzw. importierten Web Content Management Systeme aufgesetzt. Dieses Modul stellt die Schnittstelle zwischen der zentralisierten Verwaltungsschicht und den einzelnen WCMS dar. Weiters zeigt die Abbildung, dass der Anmeldevorgang des Redakteurs nun über die Verwaltungsschicht geleitet wird, die dann den Redakteur an die entsprechende Redaktionsoberfläche weiterleitet.

Der schematische Aufbau aus Abbildung 4.1 zeigt aus Gründen der Übersichtlichkeit, dass alle Web Content Management Systeme sich auf dem selben Server, wie auch die übergeordnete Verwaltungsschicht, befinden. Dies ist aber keine zwingende Maßnahme sondern nur eine Möglichkeit, da die unter der Verwaltungsschicht liegenden WCMS auch auf anderen externen Servern installiert sein können.

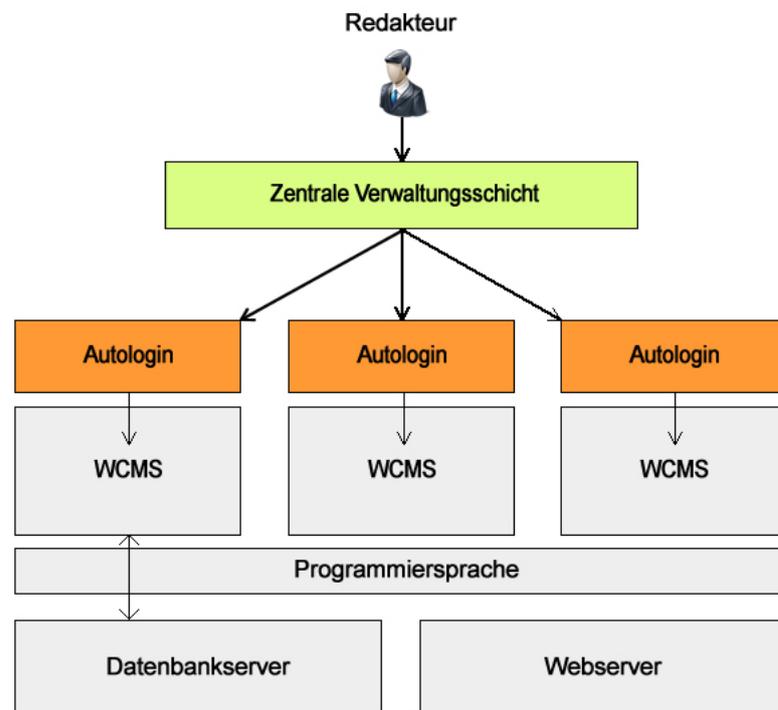


Abbildung 4.1: Schematischer Aufbau für das Zusammenführen mehrerer Webprojekte

4.2 Steuerung von administrativen Tätigkeiten am Server

Durch die in Kapitel 3.3.2 aufgezeigten Schwachstellen des Prozesses „Anlegen einer neuen Website mit vorab installierten WCMS“, die durch die Verzögerungen bei der Kommunikation zwischen Entwickler und Serverbetreuer, sowie durch die Durchführung der aufgetragenen Arbeiten durch den Serverbetreuer, entstehen, zielt dieser Abschnitt des Lösungsansatzes auf die Übernahme der Steuerung von administrativen Tätigkeiten am Server ab. Dieser Teil konzentriert sich speziell, durch das konkrete Fallbeispiel der New Media Agentur auf das Steuern der folgenden Tätigkeiten:

- Anlegen eines neuen Webspace auf dem Webserver
- Anlegen eines neuen FTP-Zuganges
- Anlegen einer neuen Datenbank
- Installieren eines Web Content Management Systems

Abbildung 4.2 zeigt den schematischen Aufbau der Verwaltungsschicht für die Steuerung von administrativen Tätigkeiten am Server und das Zusammenspiel mit den dafür benötigten Installationen auf dem Server. Jede Tätigkeit stellt in der Terminologie des Lösungsansatzes ein Modul dar. Jedes dieser Module muss zuerst eine Verbindung zum entsprechenden Server herstellen können. Anschließend muss wiederum jedes Modul unterschiedliche Voraussetzungen („*Prerequisite Checks*“) auf dem darunterliegenden System überprüfen, bevor es mit dem eigentlichen Prozess beginnen kann. Erst wenn das Modul eine gültige Verbindung hergestellt hat und alle Voraussetzungen erfolgreich überprüft wurden, kann der jeweilige Prozess des Moduls durchgeführt werden. Falls keine Verbindung hergestellt werden kann oder eine der Überprüfungen einen Fehler an die Verwaltungsschicht meldet wird der Prozess nicht durchgeführt, sondern der gesamte Prozess mit einer Fehlermeldung abgebrochen.

Weiters zeigt Abbildung 4.2, dass die Verwaltungsschicht und die darin enthaltenen Module für die Durchführung der Prozesse nicht auf einen lokalen

Kapitel 4 Entwicklung des Lösungsansatzes

Server beschränkt sind, sondern diese auch auf externe Server anwendbar sind (siehe Kapitel 7 - Erweiterungsmöglichkeiten).

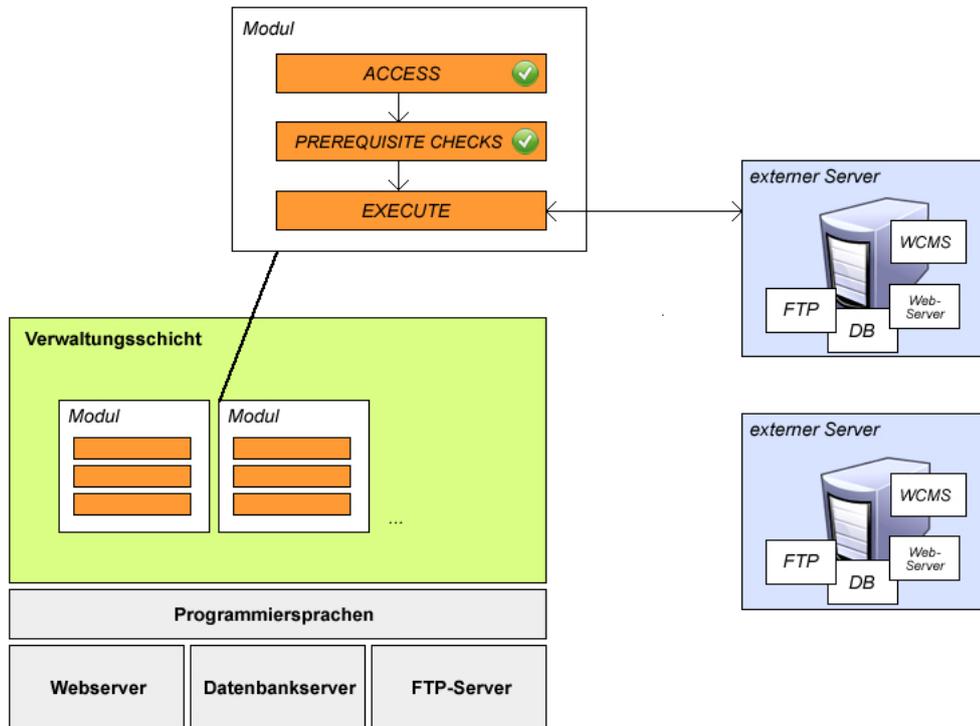


Abbildung 4.2: Schematischer Aufbau für die Steuerung von administrativen Tätigkeiten am Server über die Verwaltungsschicht

| Plattform | Anzahl an WCMS | WCMS Beispiele |
|---------------|----------------|-------------------------------|
| Java | 9 | OpenCms, Fedora, dotCMS, ... |
| Perl | 7 | Scoop, Slash, WebGUI, ... |
| PHP | 42 | TYPO3, Joomla, WordPress, ... |
| Python | 5 | Django, MoinMoin, Plone, ... |
| Ruby on Rails | 2 | Radiant, Typo |

Tabelle 4.1: Auszug über die Vielzahl an Open Source und freien WCMS ([web], Abruf 20.03.2009)

4.3 Freie Erweiterbarkeit / Pluginfähigkeit

Durch die Vielzahl von Web Content Management Systemen, die heutzutage am Markt angeboten werden, ist es unabdingbar für eine zentrale Verwaltungsschicht von Webprojekten, die Möglichkeit zu bieten, unterschiedlichste Systeme in Form von Plugins in die Verwaltungsschicht integrieren zu können. Durch den Einsatz von Plugins ist das System frei erweiterbar. In der Terminologie des Lösungsansatzes ist jedes Plugin als eigenständig anzusehen und enthält die in Kapitel 4.2 beschriebene Form von Modulen. Die Anzahl und Funktionalität der Module ist variabel und wird durch das Plugin beschrieben.

Tabelle 4.1 zeigt einen Auszug über die Vielzahl an Open Source und freien Web Content Management Systemen, die am Markt angeboten werden. Dabei ist auffallend, dass Web Content Management Systeme vermehrt mit der Programmiersprache PHP entwickelt worden sind.

Abbildung 4.3 veranschaulicht den gesamten Lösungsansatz unter der Berücksichtigung aller drei funktionaler Abschnitte.

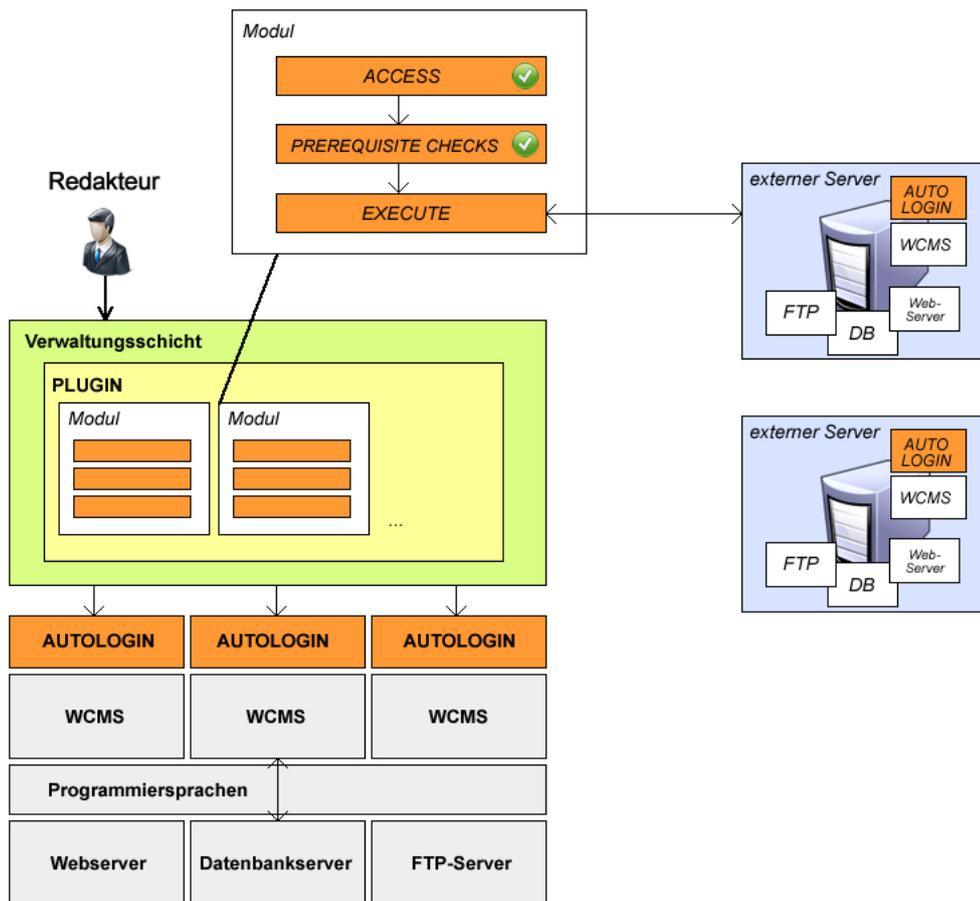


Abbildung 4.3: Schematischer Aufbau des gesamten Lösungsansatzes

Kapitel 5

Praktische Umsetzung

Dieses Kapitel beschreibt die prototypische Softwarelösung des bereits erarbeiteten Lösungsansatzes (siehe Kapitel 4). Zu Beginn werden die Teststellung und die dafür eingesetzten Technologien beschrieben und in weiterer Folge die technischen Aspekte der einzelnen Abschnitte des Lösungsansatzes erläutert.

Grundsätzlich ist die Teststellung als Prototyp für die exemplarischen Fallbeispiele der New Media Agentur BULLDOGgroup GmbH, die in Kapitel 3 und Kapitel 4 beschrieben wurden, anzusehen. Der Prototyp wird als Softwarelösung, in Form einer Webapplikation, technisch umgesetzt.

Als Testumgebung wurde ein eigener Server in einer VMware¹ Umgebung aufgesetzt. Die folgenden Kapitel beschreiben die eingesetzten Technologien, die am Server in der VMware Umgebung installiert und konfiguriert wurden.

5.1 Erläuterung der eingesetzten Technologien

5.1.1 Betriebssystem UBUNTU Server

Der Server wurde mit dem Betriebssystem Ubuntu 7.10 Server, Gutsy Gibbon, installiert. Die Ubuntu Server Edition, basiert auf der soliden Debian²

¹VMware ist eine Virtualisierungsplattform und ist die am häufigsten eingesetzte Software für die Optimierung und das Management von virtualisierten IT-Umgebungen (vgl. [Inc09c], Abruf 12.04.2009).

²Debian ist ein freies Betriebssystem aus der UNIX-Familie und steht unter GNU Public License (vgl. [itPII09], Abruf 12.04.2009).

Kapitel 5 Praktische Umsetzung

Linux-Distribution, und verfolgt laut Entwickler ebenfalls das Ziel zuverlässiger Leistung und Sicherheit. Ubuntu verändert den Unternehmensmarkt für Serverinstallationen durch sein Angebot an freier Software, die stabil, sicher und durch deren Entwickler unterstützt und weiterentwickelt wird. Seit dem Markteintritt verwenden mittlerweile hunderte von Organisationen weltweit das Server-Betriebssystem (vgl. [Ubu09]).

5.1.2 Apache Webserver

Der Apache Webserver ist ein Produkt der Apache Software Foundation³ und der meist verwendete Webserver weltweit. Abbildung 5.1 zeigt eine Übersicht über den Einsatz von verschiedenen Webservern, wobei der Apache Webserver mit rund 46% im März 2009 eine klare Führungsposition gegenüber seinen Konkurrenten in Anspruch nimmt.

Weiters zeigt die Abbildung die Entwicklung über den Einsatz der unterschiedlichen Webserver beginnend im Dezember 1995 bis März 2009. Tabelle 5.1 zeigt zu den prozentualen Werten aus Abbildung 5.1 die absoluten Zahlen über den Einsatz der einzelnen Webserver-Anbieter über den gesamten Markt aller aktiver Websites im Internet im März 2009.

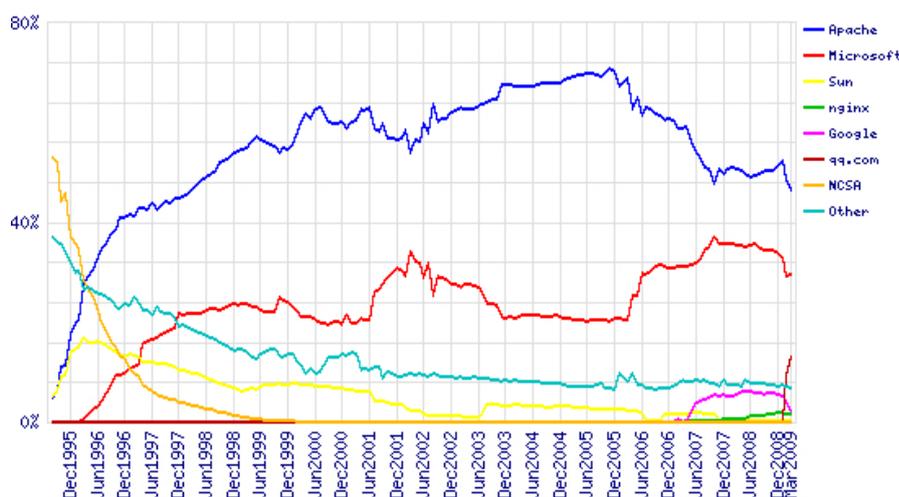


Abbildung 5.1: Marktübersicht der eingesetzten Webserver für alle aktiven Websites im gesamten Internet ([Net09])

³Nähere Informationen zur Apache Software Foundation unter <http://www.apache.org>

| Developer | Februar 2009 | Prozent | März 2009 | Prozent | Änderung |
|-----------|--------------|---------|-----------|---------|----------|
| Apache | 104796820 | 48,59% | 104178852 | 46,35% | -2,24% |
| Microsoft | 62935449 | 29,18% | 66229250 | 29,47% | +0,29% |
| qq.com | 20021763 | 9,28% | 28905129 | 12,86% | +3,58% |
| Google | 8157546 | 3,78% | 5403930 | 2,40% | -1,38% |
| nginx | 3447596 | 1,60% | 3838784 | 1,71% | +0,11% |

Tabelle 5.1: Absolute Zahlen der Marktübersicht der eingesetzten Webserver für alle aktiven Websites im gesamten Internet ([Net09])

Der für die Softwarelösung installierte Server in der VMware Umgebung verwendet den Apache Webserver in der Version 2.2.

5.1.3 Datenbanksystem MySQL

MySQL ist ein quelloffenes, relationales SQL-Datenbankmanagementsystem und wurde von 1995 bis 2008 von MySQL AB und seit der Übernahme durch Sun Microsystems⁴ im Jahr 2008 auch von diesem Unternehmen entwickelt, vertrieben und unterstützt. MySQL selbst ist mit einer GNU Public License (GPL) ausgestattet.

MySQL gilt bei seinen Anwendern als schneller, zuverlässiger und einfach bedienbarer Datenbankserver. Eine Studie von JoinVision⁵ zeigt, wie in Abbildung 5.2 graphisch dargestellt, dass IT-Fachkräfte mit fast 30% am häufigsten MySQL als Datenbanksystem einsetzen (vgl. [Inc09a], Abruf 09.04.2009).

Weiters arbeitet der MySQL-Server sowohl als Client-Server Version als auch in eingebetteten Systemen. Die MySQL-Datenbanksoftware als Client-Server System, bestehend aus einem Multi-Thread-SQL-Server, unterstützt verschiedene Backends sowie diverse Clientprogramme und -bibliotheken und Verwaltungswerkzeuge und kann mittels vieler verschiedener Programmierschnittstellen (API⁶) angesprochen werden (vgl. [Inc09b], Abruf 12.04.2009).

⁴Nähere Informationen zu Sun Microsystems unter <http://www.sun.com>

⁵Nähere Informationen zu JoinVision unter <http://www.joinvision.com>

⁶API steht für Application Programming Interface und beinhaltet alle Routinen, Datenstrukturen, Objekte und Protokolle für eine Applikation.

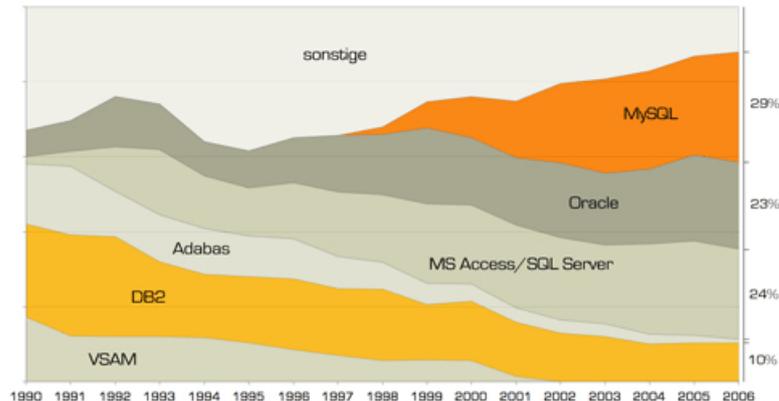


Abbildung 5.2: Studie von JoinVision über den Einsatz von MySQL ([Inc09a], Abruf 09.04.2009)

Die Softwarelösung verwendet das Datenbankmanagementsystem MySQL in der Version 5.0.45.

5.1.4 FTP-Server ProFTPD

ProFTPD ist einer der weit verbreitetsten FTP-Server für UNIX Plattformen, sowie MAC OSX und Windows. Der ProFTPD FTP-Server zeichnet sich durch eine sehr einfache Konfiguration in einer globalen Konfigurationsdatei und durch einen modularen Aufbau aus. Durch ProFTPD ist es möglich, verschiedenen Benutzern die Möglichkeit einzuräumen, Daten mit Hilfe des File Transfer Protocols (FTP) auf den Server, in ihr zugeteiltes Verzeichnis, hochzuladen (vgl. [web09b], Abruf 09.04.2009).

5.1.5 PHP

PHP steht für die Abkürzung „*PHP Hypertext Preprocessor*“ und ist eine serverseitige Programmiersprache. Serverseitig deshalb, weil PHP-Skripte durch den Server interpretiert werden und deren Ausgabe erst an den Client übermittelt wird. PHP ermöglicht Entwicklern die Erzeugung von dynamischen Internetseiten, mit denen Webapplikationen erstellt werden können. Im Unterschied zu statischen Internetseiten kann sich der Inhalt als Folge von Aktionen

des Betrachters oder aufgrund neuer Basisinformationen wie zum Beispiel aus Datenbanken ändern.

PHP unterstützt zwar die Zusammenarbeit mit vielen verschiedenen Datenbanksystemen, allerdings setzen 93% der PHP-Entwickler MySQL Datenbanken ein, wie eine Umfrage aus dem Jahr 2003 berichtet. Weiters bietet PHP im Vergleich zu anderen Programmiersprachen einige Vorteile. Aus der bereits erwähnten Umfrage, an der sich mehr als 10.000 Entwickler beteiligt haben, ergeben sich als wichtigste Gründe für die Nutzung von PHP:

- Der Schwerpunkt der Sprache liegt bei der Entwicklung von Internetanwendungen
- PHP ermöglicht die einfache Entwicklung von Programmen
- PHP unterstützt verschiedene Plattformen
- PHP lässt sich leicht in Apache, den am häufigsten eingesetzten Webserver, integrieren
- PHP hat eine gute Performance

(vgl. [The04] S.15f)

Die Softwarelösung wird in der PHP Version 5.2.3 programmiert.

5.1.6 Web Content Management System - TYPO3

TYPO3 ist ein System zur Erstellung und Verwaltung von Webprojekten. Es bietet eine Vielzahl von Entwicklungs- und Erweiterungsmöglichkeiten. Dem Leistungsumfang nach gehört TYPO3 in die Klasse der Enterprise-Lösungen, obwohl es als Open-Source-Software kostenlos verfügbar ist.

TYPO3 ist ein Web Content Management System, das sowohl der Entwicklung als auch der Pflege von Webprojekten dient. So profitiert der Redakteur einerseits von den komfortablen Varianten, Inhalte einer Website, auch zeitgesteuert, zu veröffentlichen und zu verwalten. Den Entwickler einer Website unterstützt das System andererseits mit Designvorlagen und Templates bei der

Kapitel 5 Praktische Umsetzung

Gestaltung einer Website, bei der Einrichtung von Navigation und Menüstruktur und bei der Vergabe von Berechtigungen und Zugriffsmöglichkeiten für Redakteure.

Weiters kann das WCMS TYPO3 als Entwicklungsplattform für Webapplikationen genutzt werden, die in eine Website über einen eigenen Erweiterungsmechanismus eingebunden werden können. TYPO3 hat mittlerweile eine breite Community, die bereits eine Vielzahl von Erweiterungsmodulen (Anwendungen) erstellt und veröffentlicht hat, die in eigene Webanwendungen integriert werden können.

TYPO3 ist in PHP entwickelt, verwendet MySQL als Datenbanksystem und ist als Open-Source-Software unter der GPL (Gnu Public License) veröffentlicht. Weiters gilt dieses WCMS als plattformunabhängige Serverapplikation (vgl. [Sto04] S.15).

Für die Softwarelösung wird TYPO3 verwendet, weil es ein sehr mächtiges Web Content Management System ist, hinter der eine große Community in den letzten Jahren entstanden ist. Weiters setzt die BULLDOGgroup GmbH in 90% aller Websiteentwicklungen dieses Web Content Management System ein. Speziell für die technische Umsetzung des Lösungsansatzes wird das WCMS TYPO3 in der Version 4.2.6 verwendet.

5.2 Erweiterbarkeit / Pluginfähigkeit

Wie in Kapitel 4.3 beschrieben ist es, durch die vielen unterschiedlichen Web Content Management Systeme, die am Markt angeboten werden, für die Softwarelösung unerlässlich, frei erweiterbar in Form von Plugins zu sein. Die Softwarelösung besteht grundsätzlich aus einem Satz von Plugins. Ein Plugin wiederum beschreibt ein Objekt bzw. eine Klasse die unterschiedliche Funktionen (Module) enthält. Erst durch die Programmierung der Funktionen, erhält ein Plugin seine Funktionalitäten.

Abbildung 5.3 zeigt den Aufbau der Ordner- und Dateistruktur, die für Erweiterungen unbedingt eingehalten werden muss. Unter dem Root-Verzeichnis der Applikation befindet sich der Ordner „*plugins*“. Die darin befindlichen Unterordner beschreiben genau ein Plugin, wie im konkreten Beispiel in Abbildung 5.3 das Plugin „*typo3*“.

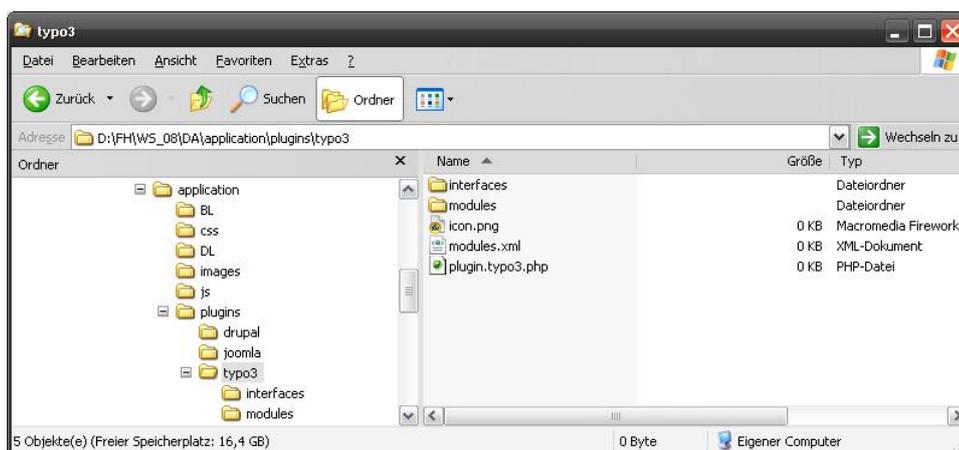


Abbildung 5.3: Aufbau der Ordner- und Dateistruktur der Softwarelösung

Weiters befindet sich im Ordner „*plugins*“ eine XML-Datei⁷ namens „*plugins.xml*“. Diese Datei beschreibt alle aktiven und inaktiven Plugins der Verwaltungsoberfläche, wobei nur aktive Plugins geladen werden. Listing 5.1 zeigt den Aufbau der Datei *plugins.xml*. Jedes Tag `<plugin>` besitzt folgende Attribute:

⁷XML ermöglicht den Austausch von strukturierten Inhalten, unter der Verwendung von Tags, die eine bestimmte Funktion erfüllen, über das Web (vgl. [CW01], S.4).

Kapitel 5 Praktische Umsetzung

- *active* [true|false], gibt an ob ein Plugin aktiv ist und somit geladen werden soll oder nicht
- *classpath*, gibt den genauen Unterordner im Ordner „*plugins*“ an, in dem sich die Pluginklasse und die Module befinden
- *classname*, gibt den genauen Namen inklusive der Dateierweiterung des Plugins an

Der Text, der sich innerhalb des Tags `<plugin></plugin>` befindet, dient als Beschreibung für die Navigationsstruktur der Verwaltungsoberfläche.

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <!ELEMENT APPLICATION (PLUGIN) *>
3 <!ELEMENT PLUGIN (#PCDATA) *>
4 <!ATTLIST PLUGIN active (true|false) #REQUIRED>
5 <!ATTLIST PLUGIN classpath CDATA #REQUIRED>
6 <!ATTLIST PLUGIN classname CDATA #REQUIRED>
7
8 <APPLICATION>
9 <PLUGIN active="true" classpath="typo3" classname="plugin.typo3.
   php">TYPO3</PLUGIN>
10 <PLUGIN active="false" classpath="joomla" classname="plugin.
   joomla.php">JOOMLA</PLUGIN>
11 <PLUGIN active="false" classpath="drupal" classname="plugin.
   drupal.php">DRUPAL</PLUGIN>
12 </APPLICATION>
```

Listing 5.1: Datei *plugins.xml*, beschreibt die aktiven und inaktiven Plugins der Verwaltungsoberfläche

Im Unterordner bzw. im Plugin „*typo3*“ befinden sich wiederum die Ordner „*modules*“ und „*interfaces*“. In ersteren befinden sich alle ausprogrammierten Funktionen (Module), die in der Plugin-Klasse enthalten sind. Der Ordner „*interfaces*“ enthält alle Interfaces, die das Plugin implementieren und ausprogrammieren kann.

Weiters enthält der Plugin-Ordner die XML-Datei „*modules.xml*“, die ähnlich aufgebaut ist wie die Datei „*plugins.xml*“, allerdings alle zu ladenden und aktiven Module enthält. Listing 5.2 zeigt den Aufbau der „*modules.xml*“. Jedes Tag `<module>` enthält die folgenden Attribute:

Kapitel 5 Praktische Umsetzung

- *active* [true|false], gibt an ob ein Modul aktiv ist und somit geladen werden soll oder nicht
- *filename*, gibt den genauen Namen der Datei im Unterordner *modules* an, die die Funktionalität des Modules enthält

Der Text, der sich innerhalb des Tags `<module></module>` befindet, dient als Beschreibung der Funktionalitäten des Plugins in der Navigationsstruktur der Verwaltungsoberfläche.

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <!ELEMENT PLUGIN(MODULE)*>
3   <!ELEMENT MODULE(#PCDATA)*>
4     <!ATTLIST MODULE active(true|false) #REQUIRED>
5     <!ATTLIST MODULE filename CDATA #REQUIRED>
6
7 <PLUGIN>
8   <MODULE active="true" filename="import_t3.php">import TYPO3</
9     MODULE>
10  <MODULE active="true" filename="backup_t3.php">backup TYPO3</
11    MODULE>
12  <MODULE active="true" filename="install_t3.php">install TYPO3</
13    MODULE>
14 </PLUGIN>
```

Listing 5.2: Datei *modules.xml*, beschreibt die aktiven und inaktiven Module eines Plugins

Neben den bereits beschriebenen Dateien und Unterordnern enthält der jeweilige Plugin-Ordner eine Datei namens „*icon.png*“. Diese Datei hat keine besondere Funktion, sondern wird, falls sie existiert, beim Erstellen der Navigationsstruktur, durch die Verwaltungsoberfläche, vor dem Namen des entsprechenden Plugins geladen. Abbildung 5.4 zeigt einen Screenshot der Verwaltungsoberfläche.

Eine genaue Beschreibung der Verwaltungsoberfläche folgt in Kapitel 5.3.

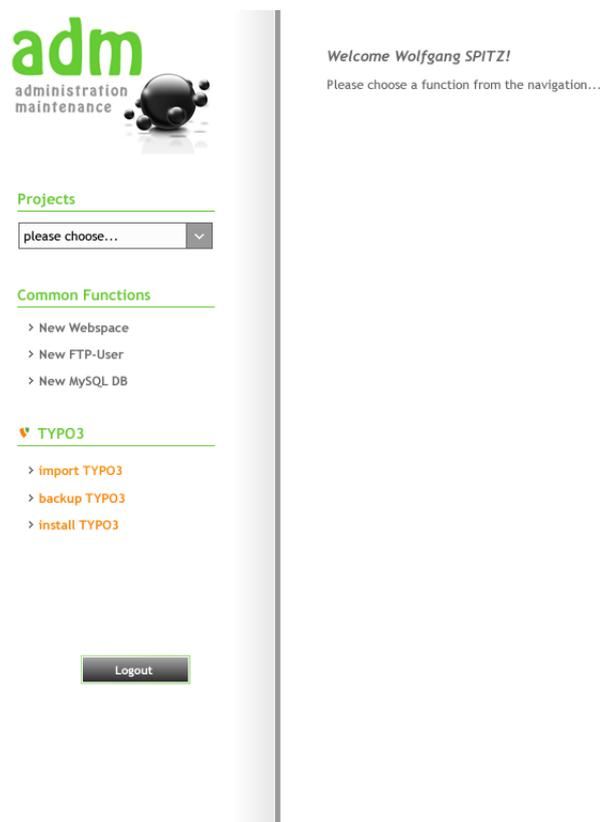


Abbildung 5.4: Screenshot der Verwaltungsoberfläche

5.3 Beschreibung der Verwaltungsoberfläche

Die Verwaltungsoberfläche stellt, den in Kapitel 4 beschriebenen Lösungsansatz dar, der in weiterer Folge auch „ADM“⁸ genannt wird. Abbildung 5.3 zeigt die grafische Darstellung der Verwaltungsoberfläche, wobei diese grundsätzlich in zwei Bereiche unterteilt ist. Der linke Bereich stellt die Navigationsstruktur und der rechte Teil den Inhaltsbereich dar. Die Navigationsstruktur beinhaltet alle zur Laufzeit geladenen Plugins und deren aktive Funktionalitäten. Neben den zur Verfügung gestellten Plugins, die wie in Kapitel 5.2 beschrieben, in die Verwaltungsoberfläche durch die entsprechenden XML-Dateien geladen werden, existieren globale Funktionalitäten, die dem Benutzer zur Verfügung stehen.

Die erste dieser „globalen“ Funktionalitäten stellt die Auswahl eines Webpro-

⁸ADM steht für Administration & Maintenance und bezeichnet die Verwaltungsoberfläche

Kapitel 5 Praktische Umsetzung

jekts über eine Auswahlliste dar. Durch die Wahl eines Webprojekts wird die entsprechende Redaktionsoberfläche im Inhaltsbereich der Verwaltungsoberfläche geladen. Der Vorteil hierbei ist, dass der Benutzer sich nicht mehr wie bisher an der Redaktionsoberfläche anmelden muss, sondern direkt in das Backend⁹ des Web Content Management Systems gelangt. Dieser Vorgang wird in Kapitel 5.4 genauer beschrieben.

Der zweite Bereich der globalen Funktionalitäten stellt allgemeine administrative Tätigkeiten am Server dar, die häufig durchgeführt werden müssen und nicht von einem entsprechendem Web Content Management System abhängig sind. Diese Tätigkeiten stehen allerdings nur Administratoren zur Verfügung und lauten wie folgt:

- Anlegen eines neuen Webspace auf dem Server
- Anlegen eines neuen FTP-Zugangs für einen entsprechenden Server- Benutzer
- Anlegen einer neuen MySQL Datenbank für einen entsprechenden Server- Benutzer

Diese administrativen Tätigkeiten werden in Kapitel 5.5 genauer beschrieben.

⁹Das Backend wird als Wartungs- oder Pflegebereich des Web Content Management Systems bezeichnet (vgl. [Sto04], S.42). In dieser Arbeit wird es auch, abhängig vom Benutzer, als Redaktions- oder Administrationsoberfläche bezeichnet

5.4 Zentralisierung der Redaktionsoberflächen mehrerer Webprojekte

Eine wesentliche Funktion des Lösungsansatzes stellt die Zentralisierung der Redaktionsoberflächen mehrerer Webprojekte dar. Dadurch kann der bereits an der Verwaltungsoberfläche angemeldete Benutzer, schnell und einfach zu seiner Redaktionsoberfläche, durch die Auswahl des entsprechenden Webprojekts aus der Auswahlliste, navigieren. Der Vorteil hierbei ist, dass er sich nicht mehr erneut an der Redaktionsoberfläche des jeweiligen WCMS anmelden muss, sondern gleich direkt im Backend landet.

Diese Auswahlliste ist in Abbildung 5.4 dargestellt und enthält alle zuvor importierten Webprojekte. Die Funktion des Importierens muss wiederum das entsprechende Plugin zur Verfügung stellen, in dem es das mitgelieferte Modul „*import_t3.php*“, wie im konkreten Beispiel im Unterordner „*typo3*“ in Abbildung 5.3 dargestellt, ausprogrammiert. Erst durch diese Funktion, können einzelne Webprojekte dem Benutzer über die Auswahlliste bereitgestellt werden.

Es ist unbedingt notwendig, dass die Funktion des Importierens das jeweilige Plugin übernimmt, da das Importieren an sich vom jeweiligen Web Content Management System und dessen Anmeldevorgang abhängig ist. Obwohl sich doch viele WCMS in ihren Funktionalitäten ähneln, sind sie doch programmtechnisch völlig verschieden und unterschiedlich aufgebaut. Dies betrifft durchaus alle Funktionen, die in weiterer Folge durch die Verwaltungsoberfläche dem Benutzer zur Verfügung gestellt werden.

Die Funktion des Importierens einer bestehenden TYPO3-Installation wird in Kapitel 5.6.1 genauer erläutert. Dieses Kapitel beschreibt die Vorgehensweise, sobald ein Benutzer ein Webprojekt aus der Liste ausgewählt hat. Diese Funktionalität ist allerdings immer von der Art und Weise des Anmeldevorganges des Web Content Management Systems abhängig. Im konkreten Fallbeispiel ist die Vorgehensweise bei der Anmeldung an einem TYPO3-WCMS wie in Abbildung 5.4 in Form eines Ablaufdiagrammes beschrieben.

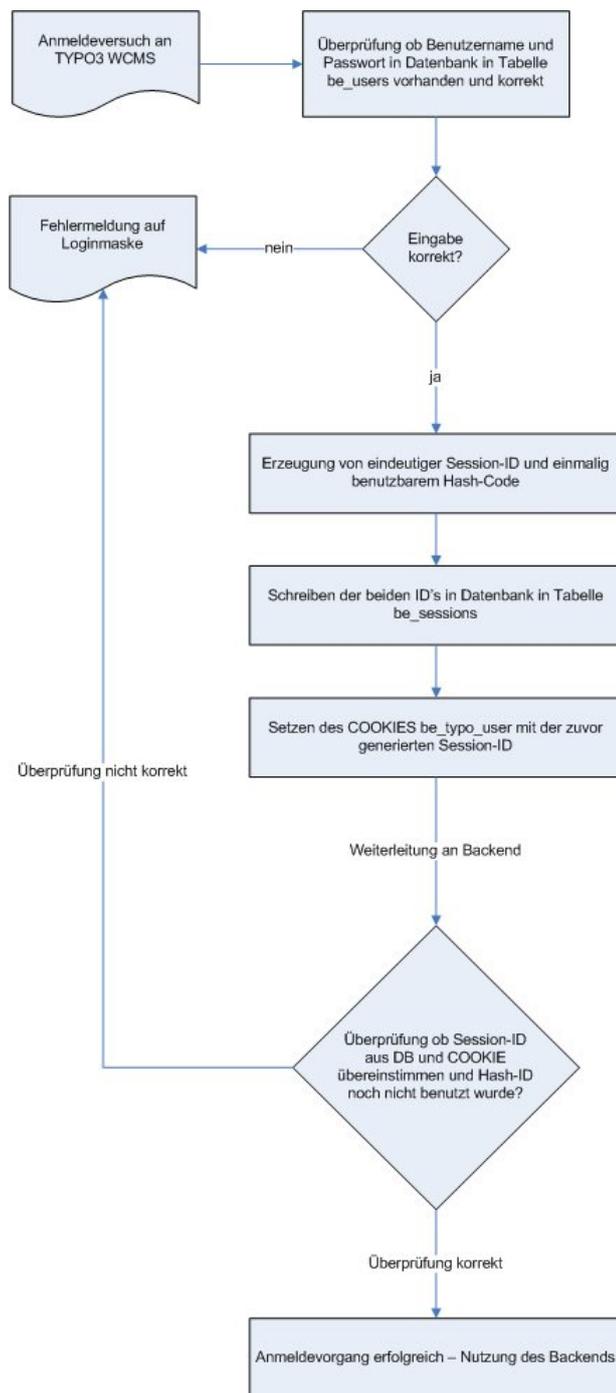


Abbildung 5.5: Ablaufdiagramm des Anmeldevorgangs bei einem TYPO3-Projekt

Abbildung 5.5 zeigt, dass zu Beginn beim Anmeldeversuch an einer TYPO3 Redaktionsoberfläche, das System die eingegebenen Daten, Benutzername und

Kapitel 5 Praktische Umsetzung

Passwort, mit den Daten in der Tabelle *be_users* der Systemdatenbank verifiziert. Stimmen die Daten überein, wird eine eindeutige Session-ID und ein achtstelliger Hashcode erzeugt und in der Tabelle *be_sessions* der Systemdatenbank abgespeichert. Weiters wird ein Cookie namens *be_typo_user* mit der gleichen Session-ID erstellt.

Anschließend überprüft ein Skript ob der Inhalt des Cookies *be_typo_user* und der Wert der übergebenen Session-ID übereinstimmen und ob der zuvor erzeugte Hashcode auch noch nicht für einen Loginversuch verwendet wurde. Falls diese beiden Überprüfungen zutreffen gilt der Anmeldevorgang als erfolgreich und der angemeldete Benutzer wird zur Redaktionsoberfläche weitergeleitet. Anderenfalls gilt der Anmeldeversuch als fehlgeschlagen und der Benutzer wird zurück zur Anmeldeseite geleitet und es wird vom System eine konkrete Fehlermeldung ausgegeben.

Durch die Verwendung der Verwaltungsoberfläche und des zuvor durchgeführten Importvorganges (siehe Kapitel 5.6.1) wird dieser Anmeldevorgang nun verändert. Der Benutzer wählt wie bereits eingangs erwähnt ein Webprojekt aus der Auswahlliste im ADM aus. Sobald er eine Auswahl getroffen hat, wird in die Tabelle *adm_webprojects_login* der ADM Datenbank ein neuer Eintrag generiert. Dieser Datensatz enthält die folgenden Attribute und Eigenschaften:

- *webid*, eine eindeutige ID für jeden Datensatz in der Tabelle (Primärschlüssel der Tabelle)
- *userid*, verweist auf den entsprechenden Benutzer in der Tabelle *be_users*, der gerade die Auswahl getroffen hat (Fremdschlüssel)
- *domain*, Domain des Webprojekts
- *ses_id*, eindeutige Session-ID
- *hash_id*, eindeutige Hash-ID
- *logged_in* [0|1] - falls *logged_in* gleich 0 ist, wurde noch kein Loginversuch durchgeführt, anderenfalls hat *logged_in* den Wert 1

Kapitel 5 Praktische Umsetzung

- *tstamp* - aktueller Timestamp, wann die Auswahl vom Benutzer durchgeführt wurde

Anschließend wird der Benutzer allerdings nicht an das Login-Skript des Webprojekts, sondern an ein, durch das ADM angelegte Skript am System des WCMS, weitergeleitet. Dieses Skript erhält die zuvor erzeugte Session- und Hash-ID und die eindeutige ID des Datensatzes aus der Tabelle *adm_ webprojects_ login*. Nun wird überprüft ob die Session- und Hash-ID's ident sind und ob auch noch kein Loginversuch mit diesen eindeutigen Nummern durchgeführt wurde, also *logged_in* den Wert 0 besitzt.

In dies der Fall verhält sich das Skript nun ähnlich dem Anmeldevorgang des WCMS TYPO3. Zuerst wird das Cookie *be_typo_user* mit der Session-ID erstellt und anschließend ein neuer Datensatz mit den eindeutigen IDs in der Tabelle *be_sessions* erzeugt. Der Benutzer ist dem TYPO3 System bereits aus dem Importvorgang bekannt und wird somit an das Backend des WCMS weitergeleitet.

Sollte allerdings bereits ein Anmeldeversuch für dieses Webprojekt mit den übergebenen IDs durchgeführt worden sein, wird der Benutzer zur Verwaltungsoberfläche zurückgeleitet und eine entsprechende Fehlermeldung ausgegeben.

5.5 Steuerung von administrativen Tätigkeiten am Server

Neben der Funktion direkt zur Redaktionsoberfläche eines importierten Webprojekts zu gelangen stellt die Softwarelösung automatisierte Prozesse für die Durchführung von administrativen Tätigkeiten am Server zur Verfügung. Diese Prozesse sind nicht vom jeweiligen Web Content Management System abhängig, sondern können dem Benutzer auch WCMS-übergreifend nützlich sein. Die Funktionen dieser Prozesse stellen in der Terminologie, wie in Kapitel 4 beschrieben, einzelne Module dar. Da diese Module aber eben unabhängig sind, sind diese im allgemeinen Plugin „*common*“ ausprogrammiert. Das „*common*“-*Plugin* hat den gleichen Aufbau wie alle anderen Plugins, wie in Kapitel 5.2 beschrieben.

Grundsätzlich benötigt der Administrator und somit der ausführende Benutzer dieser Module „*root-Rechte*“¹⁰ für das Serversystem, auf dem der automatisierte Prozess durchgeführt werden soll. Ansonsten kann von der Administrationsoberfläche nicht garantiert werden, dass ein Modul ordnungsgemäß durchgeführt werden kann. Somit verlangt das System vor der Ausführung eines Moduls nach der Eingabe des Root-Passwortes. Diese Eingabe ist für jedes Modul verpflichtend und läuft unabhängig von der eigentlichen Modul-Funktionalität analog ab. Aus diesem Grund wird bei der jeweiligen Erläuterung der Module auf die Beschreibung der Passwordeingabe verzichtet.

Da der Benutzer „*root*“ auf UNIX-Systemen allerdings jener ist, der die uneingeschränkte Kontrolle über das zugrunde liegende System besitzt, wird in Kapitel 7 - Erweiterungsmöglichkeiten - eine sichere Kommunikation mittels dem HTTPS-Protokoll¹¹ für die Softwarelösung erläutert.

In den folgenden Unterkapiteln werden nicht alle Quellcodes dezidiert angeführt, sondern lediglich relevante Skripts für die Ausführung der Module be-

¹⁰Der Benutzer „*root*“, auch „*Superuser*“ genannt, ist jener Benutzer, der in UNIX-Betriebssystemen uneingeschränkte Rechte besitzt.

¹¹HTTPS steht für HyperText Transfer Protocol Secure und ist ein sicheres Protokoll, um Daten verschlüsselt zwischen Browsern und Webservern zu übertragen (vgl. [Buc09], Abruf 12.04.2009).

schrieben. Alle restlichen Skripte, die vor allem für die Überprüfung der Voraussetzungen vor der Ausführung der Module verantwortlich sind, können der beiliegenden CD-ROM entnommen werden.

5.5.1 Anlegen eines neuen Webspace

Einer dieser automatisierten Prozesse ist das Anlegen eines neuen Webspace auf einem Server. Abbildung 5.6. zeigt einen Screenshot der Eingabemaske für diesen Prozess.

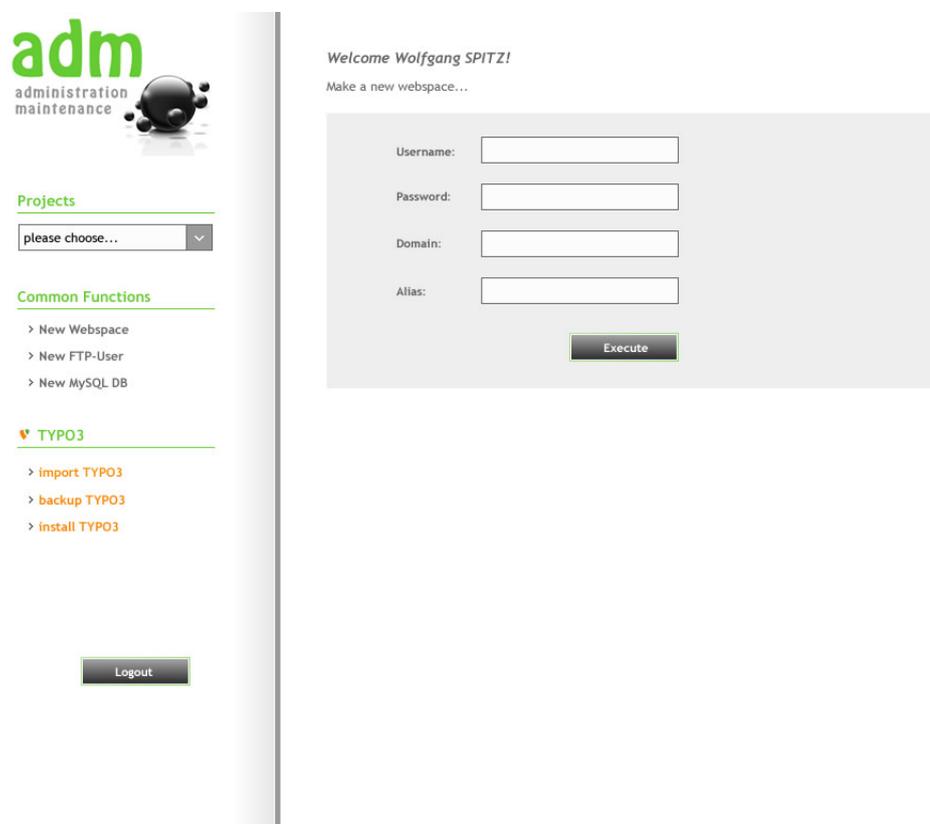


Abbildung 5.6: Screenshot der Eingabemaske für das Anlegen eines neuen Webspace

Die Eingabemaske verlangt zuerst nach einem Benutzernamen, wobei bereits der nächste freie Webbenutzer vom System vorgeschlagen wird. Webbenutzer werden auf dem System in der Regel mit dem Begriff „web“ und einer fortlaufenden Nummer angelegt. Allerdings ist es dem Benutzer erlaubt jeden anderen

Kapitel 5 Praktische Umsetzung

freien Namen zu verwenden. Als nächsten Schritt muss der Benutzer für den neuen Weospace sein Passwort vergeben. Die letzten Eingaben verlangen nach einer Domain, einem Domain-Alias und der Serveradresse (IP-Adresse) unter welcher der Benutzer seinen Weospace im Internet erreichen kann. Durch das Drücken des „Execute“-Buttons wird das Skript ausgeführt.

Die Abfolge der einzelnen Programmschritte innerhalb des Skripts (Moduls) werden, so wie in Abbildung 4.2 schematisch dargestellt, eingehalten. Das heißt zuerst werden die folgenden Voraussetzungen für das Durchführen der eigentlichen Funktionen überprüft:

- Existiert der angegebene Benutzer noch nicht auf dem System?
- Ist genügend Speicherplatz wie gewünscht vorhanden?
- Existiert die angegebene Domain noch nicht auf dem System?

Erst wenn alle diese Voraussetzung erfüllt sind, führt das Modul das folgende Shellskript¹² auf dem Server aus:

```
1  if [ $# != 5 ]
2  then
3      echo "error usage: adm_create_webpace.sh web password
4      domain alias ipaddress"
5  else
6      mkdir /var/www/$1
7      mkdir /var/www/$1/html
8      /etc/adm/adm_useradd.sh $1 $2
9      chown $1:www-data /var/www/$1/html
10     chmod 0750 /var/www/$1/html
11
12     # make virtual host entry for apache
13     echo "<VirtualHost $5>" > /etc/apache2/adm_vhosts/$1.conf
14     echo "  ServerName $3" >> /etc/apache2/adm_vhosts/$1.conf
15     echo "  ServerAlias $4" >> /etc/apache2/adm_vhosts/$1.conf
16     echo "  DocumentRoot /var/www/$1/html" >> /etc/apache2/adm
    _vhosts/$1.conf
17     echo "</VirtualHost>" >> /etc/apache2/adm_vhosts/$1.conf
```

¹²Die UNIX-Shell-Skript Programmierung dient zum größten Teil dazu, Tätigkeiten und Abläufe von Systemadministratoren zu vereinfachen und zu automatisieren (vgl. [Dit03], S.12).

```
17 /etc/init.d/apache2 force-reload
18 fi
```

Listing 5.3: Shellskript für das Anlegen eines neuen Webspace

Zuerst wird überprüft, ob dem Shellskript auch wirklich alle fünf Parameter ordnungsgemäß übergeben wurden. Ist dies nicht der Fall wird das Skript abgebrochen und eine entsprechende Fehlermeldung ausgegeben. Bei korrekter Übergabe der Parameter wird in Zeile fünf und sechs zuerst die Ordnerstruktur für den Webspace angelegt. Anschließend wird ein neuer Benutzer mit dem dazugehörigen Passwort am System angelegt. Neben der Änderung der Besitzer und der Rechte der Dateien innerhalb des Webspace wird durch eine neue Konfigurationsdatei ein neuer virtueller Host für den Apache Webserver angelegt. Dies ist notwendig, damit die Website unter der gewünschten Domain und Domain-Alias im Internet erreichbar ist. Als letzten Schritt des Skripts wird die Konfiguration des Apache Webservers neu geladen.

5.5.2 Anlegen eines neuen FTP-Zuganges

Der zweite Prozess ist das Anlegen eines neuen FTP-Zuganges für einen bestehenden Benutzer des Systems. Abbildung 5.7 zeigt die Eingabemaske für diesen Prozess.

Der Prozess erfordert als einzige Eingabe einen existierenden Benutzer des Systems. Diese Benutzer werden in einer Auswahlliste in der Verwaltungsoberfläche angezeigt, um Eingabefehler auf Seiten des Benutzers zu vermeiden. Allerdings muss auch bei diesem Prozess zuerst die Vorraussetzung überprüft werden, ob der angegebene Benutzer auch wirklich auf dem System existiert. Ist dies der Fall wird das Shellskript ausgeführt und der Benutzer zusätzlich in die Gruppe der so genannten „*ftpuser*“ hinzugefügt. Der folgende Quellcode zeigt diesen Vorgang:

```
1 if [ $# != 1 ]
2 then
3     echo "error usage: adm_addftpuser.sh user"
4 else
5     usermod -a -G ftpuser $1
```

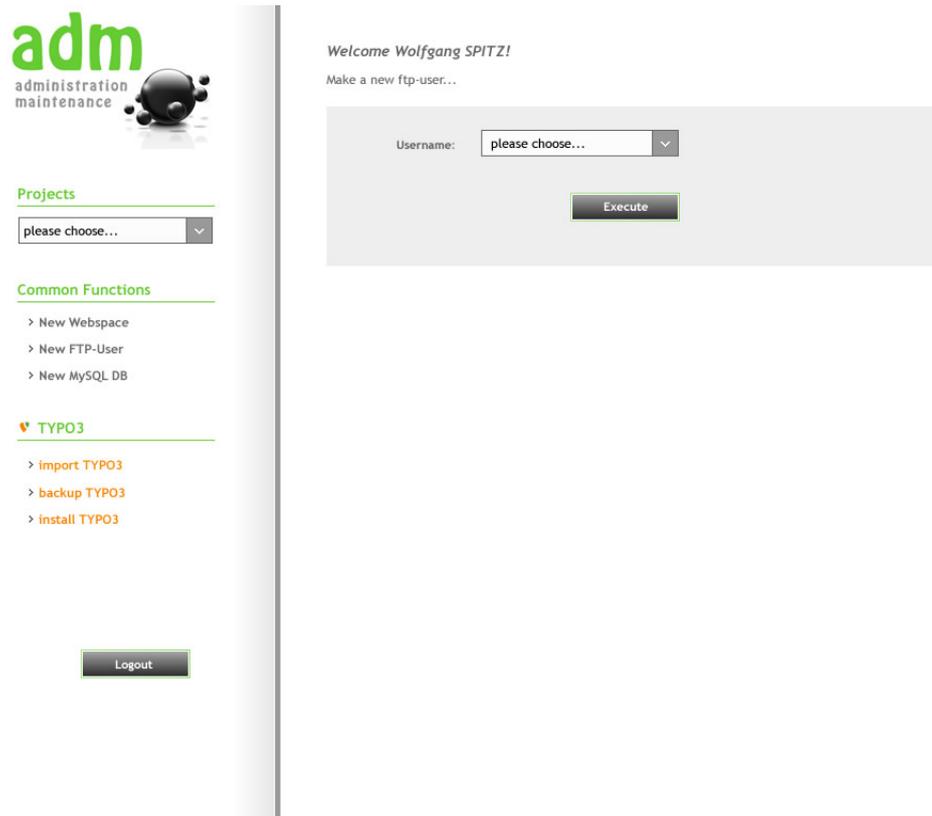


Abbildung 5.7: Screenshot der Eingabemaske für das Anlegen eines neuen FTP-Zuganges

6 fi

Listing 5.4: Shellskript für das Anlegen eines neuen FTP-Zuganges

Da der Server ProFTPD so konfiguriert wurde und ausschließlich dem Benutzer root und allen anderen Benutzern, die Mitglied der Gruppe „*ftpuser*“ sind, Zugriff zum Server via FTP gewähren, ist dies ein vergleichsweise unkomplizierter Vorgang.

5.5.3 Anlegen einer neuen MySQL Datenbank

Der dritte und letzte allgemeine Serverprozess der Softwarelösung ist das Anlegen einer neuen MySQL Datenbank für einen bestehenden Benutzer des Systems. Abbildung 5.8 zeigt die Eingabemaske für diesen Prozess:

The screenshot shows the 'adm' administration interface. On the left is a sidebar with the 'adm' logo (administration maintenance) and a navigation menu. The main content area displays a welcome message for 'Wolfgang SPITZ!' and a form titled 'Make a new MySQL-database...'. The form contains three input fields: 'Username:', 'Password:', and 'DB-Name:'. Below the fields is an 'Execute' button. The sidebar menu includes sections for 'Projects' (with a dropdown), 'Common Functions' (with links for 'New Webpace', 'New FTP-User', and 'New MySQL DB'), and 'TYPO3' (with links for 'import TYPO3', 'backup TYPO3', and 'install TYPO3'). A 'Logout' button is located at the bottom of the sidebar.

Abbildung 5.8: Screenshot der Eingabemaske für das Anlegen einer neuen MySQL Datenbank

Kapitel 5 Praktische Umsetzung

Wie bei den beiden vorigen Prozessen müssen auch hier zuerst die folgenden Voraussetzungen überprüft werden:

- Existiert der Benutzer schon auf dem System?
- Existiert der Benutzer noch nicht auf dem Datenbankserver?
- Existiert die Datenbank noch nicht auf dem Datenbankserver?

Falls diese Voraussetzungen erfüllt worden sind wird in weiterer Folge das folgende Shellskript ausgeführt:

```
1  if [ $# != 5 ]
2  then
3      echo "error usage: adm_create_db.sh host user mysqlpw
4          dbname userdbpw"
5  else
6      /etc/adm/adm_create_sql_script.sh $2 $4 $1 $5
7      mysql -h $1 -u root -p$3 < /etc/adm/mysql/$2.sql
8  fi
```

Listing 5.5: Shellskript für das Anlegen einer neuen MySQL Datenbank

Wie bereits bei allen bisher beschriebenen Shellskripts, wird auch beim Anlegen einer neuen MySQL Datenbank zuerst überprüft, ob die Anzahl der angegebenen Parameter korrekt ist. Neben dem Datenbankserver muss das Skript den Benutzernamen des Systems und der neu anzulegenden Datenbank erhalten. Weiters benötigt es das Passwort des Datenbankservers, sowie den Namen der neu anzulegenden Datenbank und das dazugehörige Passwort, welches der Benutzer benötigt um in weiterer Folge auf die Datenbank zugreifen zu können.

Sind diese fünf Parameter ordnungsgemäß an das Skript übergeben worden, wird ein weiteres Skript aufgerufen, welches, die in Zeile sechs auszuführenden SQL-Statements, in eine externe Datei schreibt. Anschließend werden diese SQL-Statements, die in Listing 5.6 beschrieben sind, ausgeführt.

```
1  if [ $# != 4 ]
2  then
3      echo "error usage: adm_create_sql_script.sh user db host
4          userpw";
```

```
4 else
5     touch /etc/adm/mysql/$1.sql
6     echo "create user $1;" > /etc/adm/mysql/$1.sql
7     echo "drop database if exists $2;" >> /etc/adm/mysql/$1.
      sql
8     echo "create database $2;" >> /etc/adm/mysql/$1.sql
9     echo "grant CREATE,INSERT,DELETE,UPDATE,SELECT on $2.* to
      $1@$3;" >> /etc/adm/mysql/$1.sql
10    echo "set password for $1 = password('$4');" >> /etc/adm/
      mysql/$1.sql
11    echo "set password for $1@$3 = password('$4');" >> /etc/
      adm/mysql/$1.sql
12    echo "flush privileges;" >> /etc/adm/mysql/$1.sql
13    echo "exit" >> /etc/adm/mysql/$1.sql
14 fi
```

Listing 5.6: Shellskript für das Erstellen des auszuführenden SQL-Skripts

Das Shellskript „*adm_create_sql_script.sh*“ erhält die Parameter aus dem Skript, welches es aufgerufen hat, übergeben. Das Skript erzeugt für den Benutzer eine SQL-Datei, die die SQL-Statements enthält, die zuerst einen neuen Datenbankbenutzer erstellt. Anschließend wird die Datenbank angelegt, die Funktionen, die innerhalb der Datenbank durchgeführt werden dürfen, festgelegt und zu guter Letzt das Zugriffspasswort für den neuen Benutzer der Datenbank gesetzt.

5.6 Beschreibung des Plugins „*TYPO3*“

Dieses Kapitel beschreibt das eigens für diese Softwarelösung entwickelte Plugin TYPO3. Dieses Plugin enthält die folgenden Module bzw. Funktionalitäten:

- Importieren einer bestehenden TYPO3-Installation
- Backup einer bestehenden TYPO3-Installation
- Installieren eines neuen TYPO3-Projekts

5.6.1 Importieren einer bestehenden TYPO3-Installation

Das Modul „*Importieren einer bestehenden TYPO3-Installation*“ muss ausgeführt werden, um auf die Redaktionsoberfläche des Web Content Management Systems in weiterer Folge direkt über die Auswahlliste der Verwaltungsoberfläche zugreifen zu können. Erst nachdem dieser Importvorgang erfolgreich ausgeführt wurde, kann das Webprojekt aus der Auswahlliste ausgewählt werden.

Wie bei allen anderen Modulen, muss auch dieses zuvor die Voraussetzungen für das Ausführen der Funktion überprüfen. Erst wenn die folgenden Voraussetzungen gegeben sind, kann das Modul ordnungsgemäß ausgeführt werden.

- Wurde die angegebene TYPO3-Installation noch nicht importiert?
- Existiert die angegebene TYPO3-Installation auf dem angegebenen System?
- Existiert ein gültiger Zugriff zur MySQL-Datenbank der TYPO3-Installation?

Abbildung 5.9 zeigt die Eingabemaske dieses Moduls innerhalb des Plugins TYPO3. Neben der Serveradresse (externe IP-Adresse oder Localhost) muss eine gültige Verbindung, inklusive Datenbankhost (externe IP-Adresse oder

Kapitel 5 Praktische Umsetzung

Localhost), -name, -benutzer und -passwort, zur Datenbank der TYPO3-Installation angegeben werden. Diese Verbindung ist essentiell für den Importvorgang, da es ansonsten nicht möglich ist, den in Kapitel 5.4 beschriebenen Anmeldevorgang erfolgreich durchzuführen.

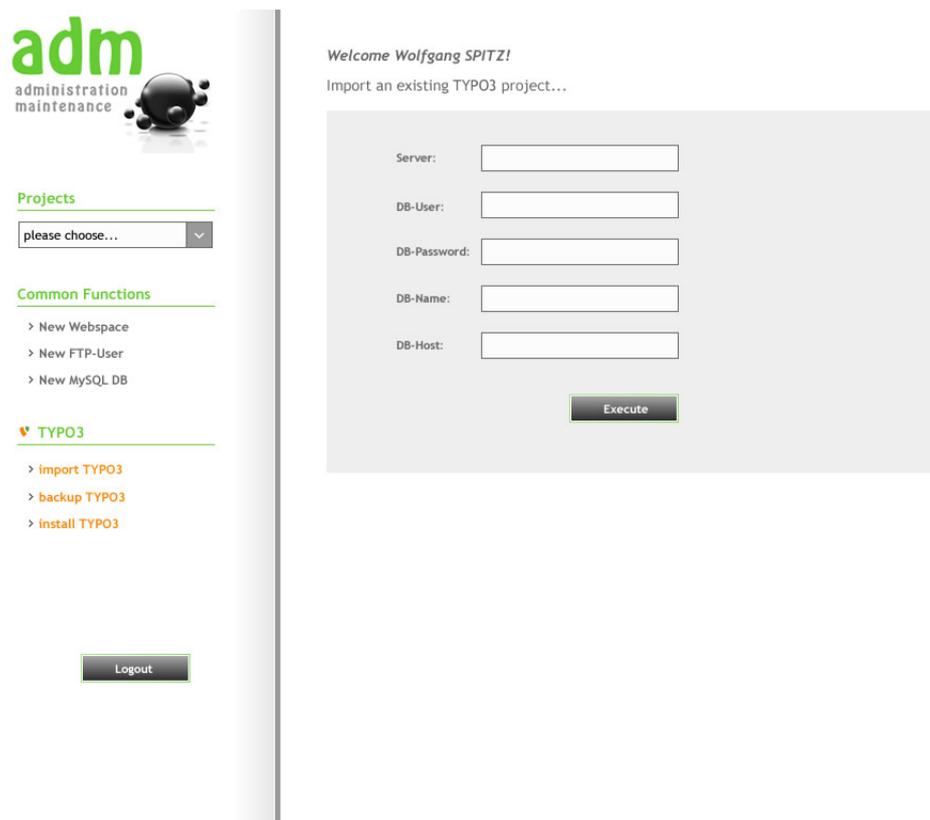


Abbildung 5.9: Screenshot der Eingabemaske für das Importieren einer bestehenden TYPO3-Installation

Nach erfolgreicher Überprüfung der Voraussetzungen wird durch das Modul das folgende Skript in den TYPO3-Installationsordner kopiert. Dieses Skript wird dann durch Auswahl des Webprojekts aus der Auswahlliste in der Verwaltungsoberfläche anstatt des standardmäßigen Anmeldeskripts von TYPO3 ausgeführt.

```
1 <?php
2     include("dbconfig.inc.php");
3     include("functions.inc.php");
4
5     $session_id = $_GET['sid'];
```

Kapitel 5 Praktische Umsetzung

```
6     $hash_code = $_GET['hid'];
7     $username = $_GET['username'];
8     $website = $_GET['website'];
9
10    if(checkIds($session_id,$hash_code)
11    {
12        $session_name = "be_typo_user";
13        $session_ip_lock = "127.0.0.1";
14        $userid = getUserid($username);
15        $tstamp = time();
16        $session_backend_userid = 0;
17
18        // sets the cookie
19        setcookie("be_typo_user", $session_id);
20
21        // write session and user to database in table be_sessions
22        $query = "insert into be_sessions(ses_id, ses_name, ses_
                iplock, ses_hashlock, ses_userid, ses_tstamp, ses_
                backuserid) values('".$session_id."','".$session_name
                ."', '".$session_ip_lock."', $hash_code, $userid, $tstamp,
                $session_backend_userid)";
23        $res = mysql_query($query) or die(mysql_error());
24
25        mysql_free_result($res);
26
27        header("Location: http://".$website."/typo3/backend.php);
28    }
29    else
30    {
31        die("Session is not valid");
32    }
33    ?>
```

Listing 5.7: PHP Skript für den neuen Anmeldevorgang einer importierten TYPO3-Installation

Neben diesem Skript werden noch zwei weitere Skripts in das TYPO3-Projekt kopiert. Die Datei „*dbconfig.inc.php*“ erstellt die Verbindung zur angegebenen TYPO3-Datenbank her und die Datei „*functions.inc.php*“ enthält, die für das oben beschriebene Skript notwendigen Funktionen. Beide Dateien werden zu Beginn des Skripts inkludiert.

Kapitel 5 Praktische Umsetzung

Zuerst werden die übergebenen GET-Parameter, genauer die bereits erzeugte Session- und Hash-ID, sowie der Benutzername und der Name des TYPO3-Projekts, gespeichert. Anschließend wird in der Methode „*checkIds()*“ überprüft, ob die beiden IDs mit der in der ADM-Datenbank befindlichen Identifikationsnummern übereinstimmt und für diese IDs noch kein Anmeldeversuch durchgeführt worden ist. Ist dies der Fall liefert die Methode *true* zurück, ansonsten wird das Skript mit einer entsprechenden Fehlermeldung beendet.

Als nächstes werden die restlichen Attribute, die für den Eintrag in der Tabelle *be_sessions* notwendig sind, erzeugt. Anschließend wird das Cookie „*be_typo_user*“ mit dem Wert der Session-Id gesetzt und der benötigte Datensatz in die Tabelle *be_sessions* geschrieben. Nun ist die Anmeldung am TYPO3-System erfolgreich durchgeführt worden und der letzte Befehl leitet den Benutzer direkt in das TYPO3-Backend.

5.6.2 Sichern einer bestehenden TYPO3-Installation

Durch die Funktion, eine bestehende TYPO3-Installation zu sichern, ist es möglich auf einfache und schnelle Art und Weise das gesamte Webprojekt inklusive einer aktuellen Datenbanksicherung lokal zu speichern. Abbildung 5.10 zeigt einen Screenshot der Eingabemaske für das Sichern einer bestehenden TYPO3-Installation.

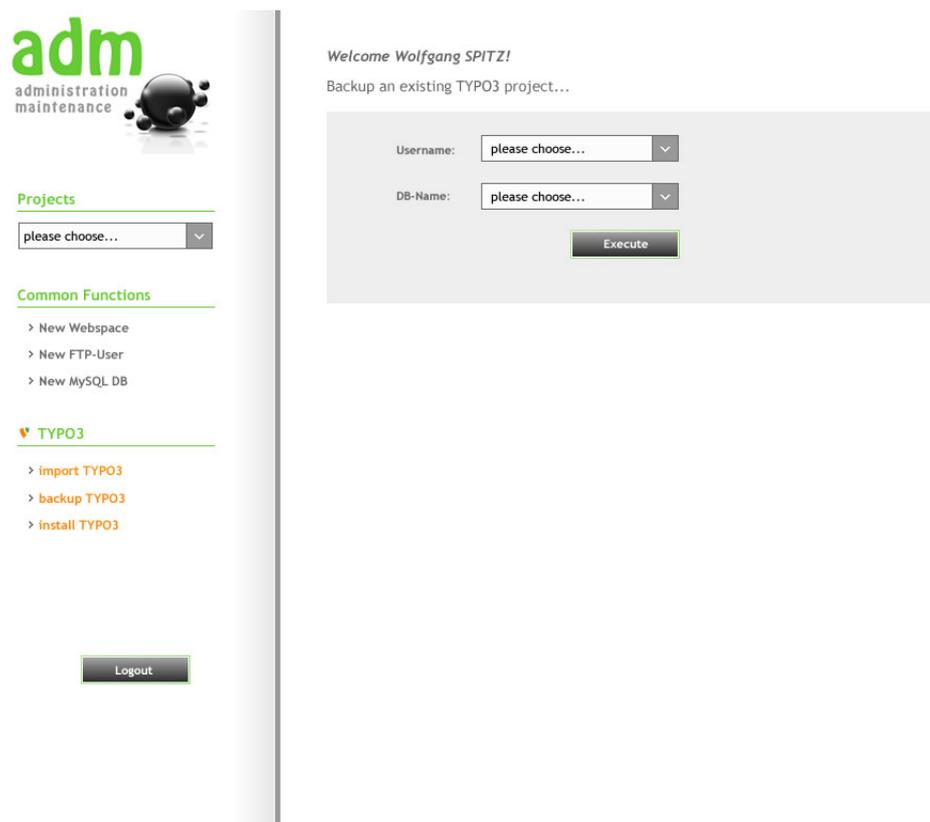


Abbildung 5.10: Screenshot der Eingabemaske für das Sichern einer bestehenden TYPO3-Installation

Die in Abbildung 5.10 dargestellte Eingabemaske enthält zwei Auswahllisten. Die erste besteht aus allen, dem System bekannten, Benutzern und die zweite aus allen dem System bekannten Datenbanken. Dadurch werden Eingabefehler von der Benutzerseite her vermieden. Nichts desto trotz müssen für die Ausführung dieses Moduls die folgenden Voraussetzungen überprüft werden:

- Existiert noch kein Backup für diesen Benutzer?

Kapitel 5 Praktische Umsetzung

- Existiert der angegebene Benutzer auf dem System?
- Existiert die angegebene TYPO3-Installation?
- Existiert die angegebene Datenbank?

Bei erfolgreicher Überprüfung werden alle Source-Dateien, sowie alle installierten Extensions und der Ordner „*fileadmin*“ in ein Archiv gepackt. Zusätzlich wird eine Sicherung der Datenbank (Datenbank-Dump) erstellt und anschließend dem Benutzer der Verwaltungsoberfläche zum Download angeboten. Diesen Ablauf zeigt das folgende Shellskript:

```
1  if [ $# != 3 ]
2  then
3      echo "error usage: adm_backup_typo3.sh user dbname
4          dbpassword"
5  else
6      cd /var/www/$1/
7      mkdir backup
8
9      cd /var/www/$1/backup
10     tstamp=`date +%d.%m.%Y`
11     mysqldump -p$3 $2 > /var/www/$1/backup/$2_$tstamp.dump
12     tar -czvf backup_t3_$1.tar.gz /var/www/$1/html/*
13
14     chown $1:www-data /var/www/$1/backup/*
15     chmod 0775 /var/www/$1/backup/*
16 fi
```

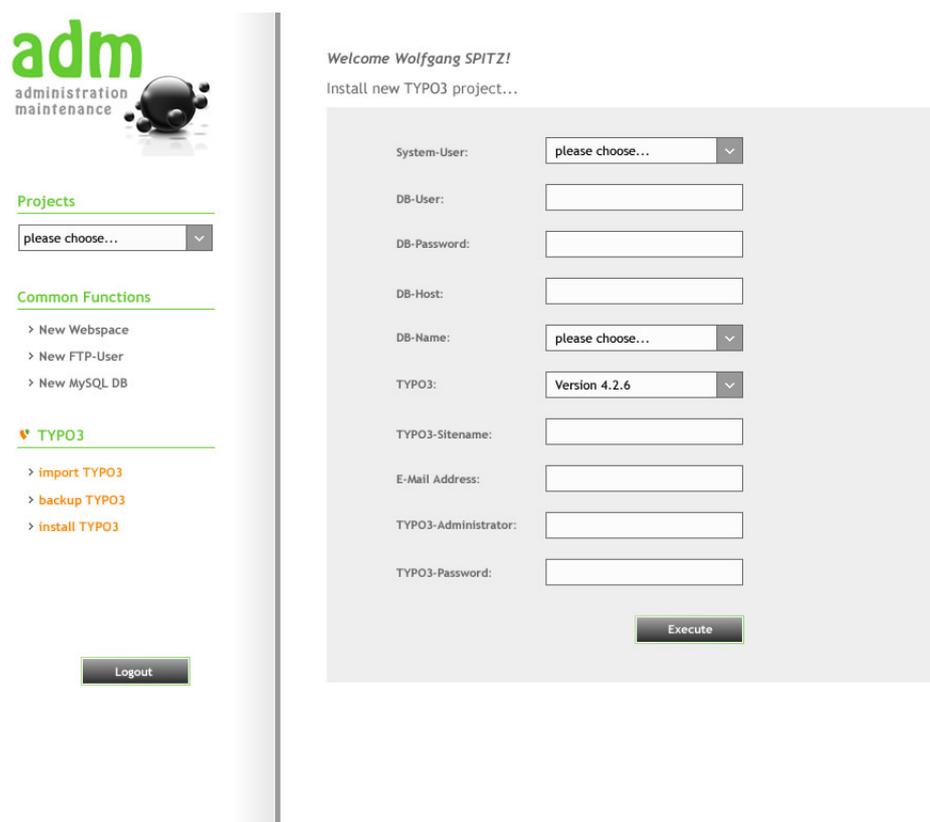
Listing 5.8: Shellskript für das Sichern einer bestehenden TYPO3-Installation

Wie im Listing gezeigt wird in Zeile zehn der MySQL-Dump durch den Befehl „*mysqldump*“ initialisiert, der wiederum in eine externe Datei im Unterordner „*backup*“ des Webverzeichnis des angegebenen Benutzers erstellt wird. Anschließend wird in Zeile elf die gesamte TYPO3-Installation durch den Befehl „*tar*“ in ein Archiv gepackt und ebenfalls im Unterordner „*backup*“ hinzugefügt. Zu guter Letzt wird noch der Besitzer und die Gruppe der Dateien an den angegebenen Benutzer angepasst und die Rechte entsprechend geändert.

5.6.3 Installieren eines neuen TYPO3-Projekts

Das letzte, für die Softwarelösung, entwickelte Modul ist das Installieren eines neuen TYPO3-Projekts. Durch diese Funktion ist es dem Benutzer des Systems möglich ein TYPO3-WCMS in der angegebenen Version herunterzuladen, zu installieren, zu konfigurieren und die Systemdatenbank zu importieren. Das heißt der Benutzer erhält ein fertiges Basissystem, auf das er seine Website aufbauen kann und muss sich nicht um Installationen und/oder Konfigurationen, die das zu grundlegende System betreffen, kümmern.

Um dieses Modul ausführen zu können müssen einige Eingaben vom Benutzer getätigt werden. Abbildung 5.11 zeigt einen Screenshot der Eingabemaske mit allen Eingabefeldern für das Installieren eines neuen TYPO3-Projekts.



The screenshot displays the 'adm administration maintenance' interface. On the left, there is a sidebar with a 'Logout' button and a 'Projects' section containing a dropdown menu. Below that, 'Common Functions' includes links for 'New WebSpace', 'New FTP-User', and 'New MySQL DB'. The 'TYPO3' section is expanded, showing links for 'import TYPO3', 'backup TYPO3', and 'install TYPO3'. The main content area shows a welcome message for 'Wolfgang SPITZ!' and the title 'Install new TYPO3 project...'. The form contains the following fields:

- System-User: dropdown menu (selected: please choose...)
- DB-User: text input field
- DB-Password: text input field
- DB-Host: text input field
- DB-Name: dropdown menu (selected: please choose...)
- TYPO3: dropdown menu (selected: Version 4.2.6)
- TYPO3-Sitename: text input field
- E-Mail Address: text input field
- TYPO3-Administrator: text input field
- TYPO3-Password: text input field

An 'Execute' button is located at the bottom right of the form.

Abbildung 5.11: Screenshot der Eingabemaske für das Installieren eines neuen TYPO3-Projekts

Bei der ersten Eingabe muss der Benutzer den Systembenutzer auswählen,

Kapitel 5 Praktische Umsetzung

für den das TYPO3-Projekt in seinem bestehenden Webverzeichnis installiert wird. Als nächsten Punkt muss er eine gültige DB-Verbindung angeben, die aus Benutzernamen, Passwort, Datenbankhost und Datenbanknamen besteht. Anschließend wählt er eine TYPO3-Version aus, die in weiterer Folge vom Repository heruntergeladen werden soll. Als letzte Eingaben wird nach dem Namen des TYPO3-Projekts, einer E-Mail Adresse, die in das TYPO3-System für Benachrichtigungen eingetragen wird, sowie nach den Zugangsdaten für einen TYPO3-Administrator, verlangt.

Aus den Parametern der Eingabemaske ergeben sich auch ein Großteil der Voraussetzungen, die erfüllt werden müssen, bevor das Modul ausgeführt wird und lauten wie folgt:

- Existiert der Benutzer auf dem System?
- Existiert eine gültige Datenbankverbindung?
- Sind mindestens 200 MB Speicherplatz auf dem Server vorhanden?

Erst nach erfolgreicher Überprüfung aller Voraussetzungen wird das Modul und die folgenden drei Shellskripts in dieser Reihenfolge von der Verwaltungsoberfläche ausgeführt.

```
1 if [ $# != 2 ]
2 then
3     echo "error usage: adm_install_typo3 user t3version"
4 else
5     # go to specified webfolder
6     cd /var/www/$1/html
7
8     # download typo3_src
9     wget http://prdownloads.sourceforge.net/typo3/typo3_src
        -$2.tar.gz?download
10
11     if [ -f typo3_src-$2.tar.gz ]
12     then
13         # download dummy
14         wget http://prdownloads.sourceforge.net/typo3/
            dummy-$2.tar.gz?download
15         if [ -f dummy-$2.tar.gz ]
16         then
```

```
17         # entpacken typo3_src
18         tar xfvz typo3_src-$2.tar.gz
19         # entpacken typo3_dummy
20         tar xfvz dummy-$2.tar.gz
21         # delete archives
22         rm typo3_src-$2.tar.gz
23         rm dummy-$2.tar.gz
24
25         # typo3_dummy eine ebene heraus moven
26         cd dummy-$2
27         mv * ../
28         cd ..
29         rm dummy-$2 -r
30         # ln -s fuer typo3_src dateien
31         rm typo3_src
32         ln -s typo3_src-4.2.6 typo3_src
33         # chown on all files
34         chown $1:www-data * -R
35         # chmod on all files
36         chmod 0775 * -R
37         exit 0
38     else
39         echo "download error of typo3-dummy-$2
40             package from specified url"
41         rm /var/www/$1/html/showfiles.php*
42         exit 2
43     fi
44 else
45     echo "download error of typo3_src-$2.tar.gz from
46         specified url"
47     rm /var/www/$1/html/showfiles.php*
48     exit 1
49 fi
```

Listing 5.9: Shellskript für das Installieren eines neuen TYPO3-Projekts

Dem Skript „*adm_install_typo3*“ werden die Parameter *Systembenutzer* und *TYPO3-Version* ordnungsgemäß übergeben. Anschließend wechselt das Skript in das Webverzeichnis des Systembenutzers und lädt die TYPO3-Source Dateien in Form von Archiven herunter. Nach dem Entpacken der Archive werden

Kapitel 5 Praktische Umsetzung

die Source-Dateien, für spätere Updates, durch symbolische Links¹³ erweitert. Die letzten beiden Befehle verändern wiederum den Besitzer und die Gruppe, sowie die Rechte aller Dateien innerhalb des Webverzeichnis.

Das nächste Shellskript behandelt die Konfiguration des installierten TYPO3-Systems. Da TYPO3 zur Gänze in PHP geschrieben ist, basiert auch die Konfigurationsdatei „*localconf.php*“ auf dieser Programmiersprache. Diese Konfigurationsdatei wird normalerweise durch Änderungen im TYPO3-Installationstool erstellt bzw. verändert. Durch die Manipulation dieser Datei, kann die Verwaltungsoberfläche bereits vorab gewünschte Einstellungen treffen.

```
1 if [ $# != 6 ]
2 then
3     echo "error usage: adm_configure_typo3 user dbpassword
4         dbhost dbname t3sitename email"
5 else
6     cd /var/www/$1/html/typo3conf
7     touch ENABLE_INSTALL_TOOL
8
9     rm temp*
10    rm localconf.php
11    echo "<?php" > /var/www/$1/html/typo3conf/localconf.php
12    echo "\$TYPO3_CONF_VARS['SYS']['sitename'] = 'New Typo3
13        Site';" >> /var/www/$1/html/typo3conf/localconf.php
14    echo "\$TYPO3_CONF_VARS['BE']['installToolPassword'] = '
15        bacb98acf97e0b6112b1d1b650b84971';" >> /var/www/$1/
        html/typo3conf/localconf.php
16    echo "\$TYPO3_CONF_VARS['EXT']['extList'] = 'tsconfig_help
17        ,context_help,extra_page_cm_options,impexp,sys_note,
18        tstemplate,tstemplate_ceditor,tstemplate_info,
19        tstemplate_objbrowser,tstemplate_analyzer,func_wizards
20        ,wizard_crpages,wizard_sortpages,lowlevel,install,
21        belog,beuser,aboutmodules,setup,taskcenter,info_
22        pagetsconfig,viewpage,rtehtmlarea,css_styled_content,t
23        3skin';" >> /var/www/$1/html/typo3conf/localconf.php
24    echo "\$typo_db_extTableDef_script = 'extTables.php';" >>
25        /var/www/$1/html/typo3conf/localconf.php
```

¹³Das Arbeiten mit symbolischen Links bei einer Typo3-Installation wird in verschiedenen Literaturen einschlägig beschrieben (vgl. [web09c], Abruf 10.04.2009).

Kapitel 5 Praktische Umsetzung

```
16 echo "\$typo_db_username = '$1';" >> /var/www/$1/html/typo
    3conf/localconf.php
17 echo "\$typo_db_password = '$2';" >> /var/www/$1/html/typo
    3conf/localconf.php
18 echo "\$typo_db_host = '$3';" >> /var/www/$1/html/typo3
    conf/localconf.php
19 echo "\$TYPO3_CONF_VARS['SYS']['encryptionKey'] = '5a643c
    84284fe7770b8697392869e332';" >> /var/www/$1/html/typo3
    3conf/localconf.php
20 echo "\$typo_db = '$4';" >> /var/www/$1/html/typo3conf/
    localconf.php
21 echo "\$TYPO3_CONF_VARS['SYS']['sitename'] = '$5';" >> /
    var/www/$1/html/typo3conf/localconf.php
22 [...]
23 echo "?>" >> /var/www/$1/html/typo3conf/localconf.php
24
25 chown $1:www-data /var/www/$1/html/typo3conf/localconf.php
26 chmod 0775 /var/www/$1/html/typo3conf/localconf.php
27 fi
```

Listing 5.10: Shellskript für das Konfigurieren einer TYPO3-Installation

Zu Beginn des Skripts wird durch das Anlegen der Datei „*ENABLE_ INSTALL_ TOOL*“ im Verzeichnis „*/typo3conf*“ das Installationstool von TYPO3 aktiviert. Dies ist für spätere Veränderungen am System in der Entwicklungsphase notwendig. Anschließend werden durch das Erstellen einer neuen „*localconf.php*“ die einzelnen Einstellungen getroffen.

Das dritte und letzte auszuführende Shellskript, importiert die Systemtabellen, die von TYPO benötigt werden. Unter anderem die Tabelle *be_users*, die alle Systembenutzer von TYPO3 enthält.

```
1 if [ $# != 4 ]
2 then
3     echo "error usage: adm_create_db.sh user dbpw dbname
        dbhost"
4 else
5     mysql -h $4 -u $1 -p$2 -D $3 < /etc/adm/mysql/t3_import.
        sql
```

⁶ fi

Listing 5.11: Shellskript für das Importieren der Systemdatenbank für eine TYPO3-Installation

Als letzten Schritt des Moduls, wird ein Administrator mit den angegebenen Zugangsdaten in die Tabelle *be_users* des TYPO3-Projekts importiert. Somit kann der Administrator bereits zum Backend der TYPO3-Installation navigieren und sich mit seinen Zugangsdaten anmelden. Um an dieser Stelle Relevanz und das Zusammenspiel aller Module hervorzuheben, sollte der Administrator nun das Import-Modul des TYPO3-Plugins ausführen und das soeben installierte Projekt auch importieren. Dadurch kann er sein neu installiertes Basissystem über die Auswahlliste der Verwaltungsoberfläche auswählen und kann mit dem Erstellen des Frontends beginnen.

Kapitel 6

Ergebnisse

In Kapitel 3 wurden die folgenden zwei konkreten Fallbeispiele der „New Media Agentur“ BULLDOGgroup GmbH beschrieben und auf ihre Schwachstellen hin analysiert:

- Redaktionelle Betreuung von Websites
- Anlegen einer neuen Website mit vorab installierten WCMS

Diese beiden Szenarien wurden an Hand der Aufzeichnungen von Redakteuren und Entwicklern hinsichtlich der damaligen Arbeitsweise bewertet und in Abbildung 3.3 und Abbildung 3.4 graphisch dargestellt. Die folgenden Abschnitte beschreiben die Veränderungen in diesen beiden Szenarien durch den Einsatz der prototypischen Softwarelösung, die in Kapitel 4 konzipiert und technisch umgesetzt wurde.

6.1 Redaktionelle Betreuung von Websites

Durch die Analyse der redaktionellen Betreuung von Websites und der Interpretation wurde festgestellt, dass die Schwachstelle des Prozesses im Anmeldevorgang durch die Redakteure liegt. Genauer gesagt sind in vielen Fällen die Zugangsdaten für den Anmeldevorgang nicht vorhanden und somit kommt es zu Verzögerung bei der Kommunikation mit den Entwicklern, die diese Zugangsdaten verwalten.

Kapitel 6 Ergebnisse

Um dies zu umgehen wurde der Lösungsansatz für die Zentralisierung von Webprojekten entwickelt (siehe Kapitel 4.1). Dieser Lösungsansatz benötigt aus technischer Sicht einen Administrator, der in der Verwaltungsoberfläche ausgewählte Webprojekte importiert, die dann in weiterer Folge in der Auswahlliste im ADM zur Verfügung stehen (siehe Kapitel 5.4 und 5.6.1). Durch den Einsatz der Verwaltungsoberfläche ist der Anmeldevorgang am Backend von Redaktionsoberflächen überflüssig und der Prozess wurde wie in Abbildung 6.1 graphisch dargestellt optimiert.

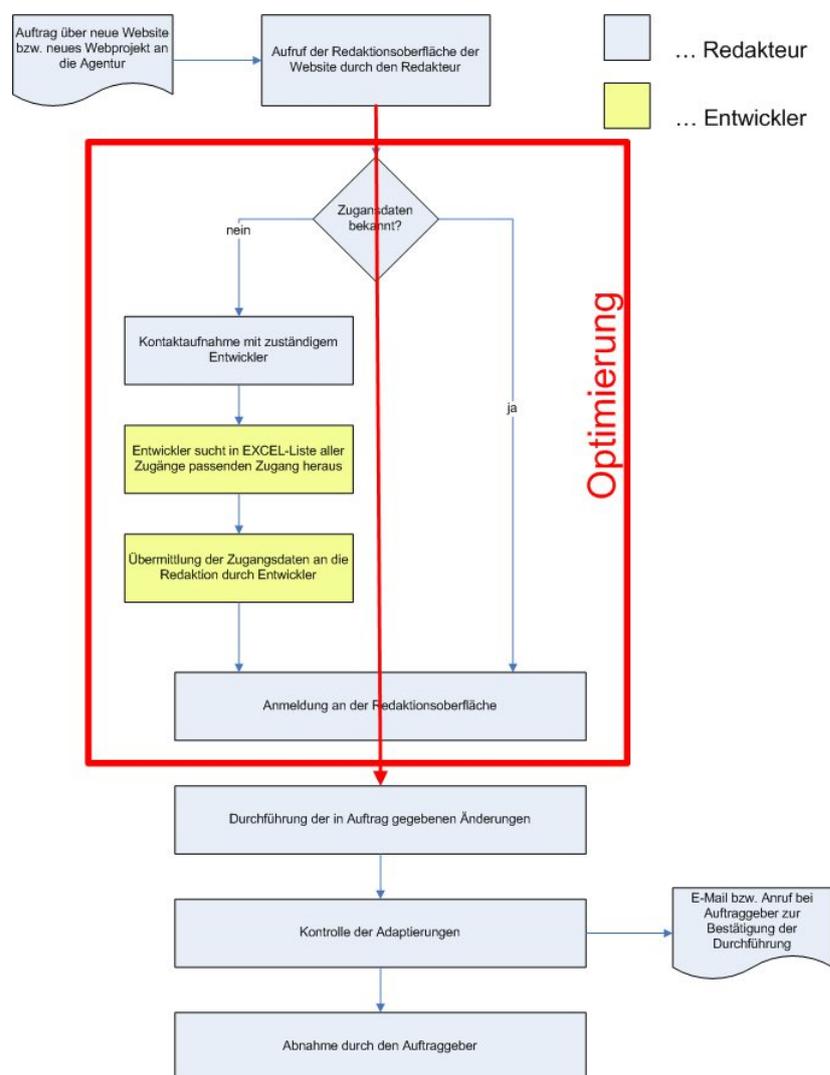


Abbildung 6.1: Ablaufdiagramm des optimierten Prozesses für die redaktionelle Betreuung einer Website

Kapitel 6 Ergebnisse

Der rote Kasten in Abbildung 6.1 zeigt, die durch die Optimierung wegfallenden Arbeitsschritte. Durch diese Optimierung benötigt der Redakteur keine Zugangsdaten mehr, sondern wählt einfach aus den einzelnen Webprojekten aus und gelangt automatisch in die gewünschte Redaktionsoberfläche des Web Content Management Systems.

Um diese Optimierung in Zahlen zu bewerten, zeigt Abbildung 6.2 die Änderungen der in Kapitel 3.3.1 durchgeführten Prozessbewertung. Durch den veränderten Anmeldevorgang fallen die Verzögerungen, die bisher durch die Kommunikation zwischen Redakteur und Entwickler entstanden sind, automatisch weg.

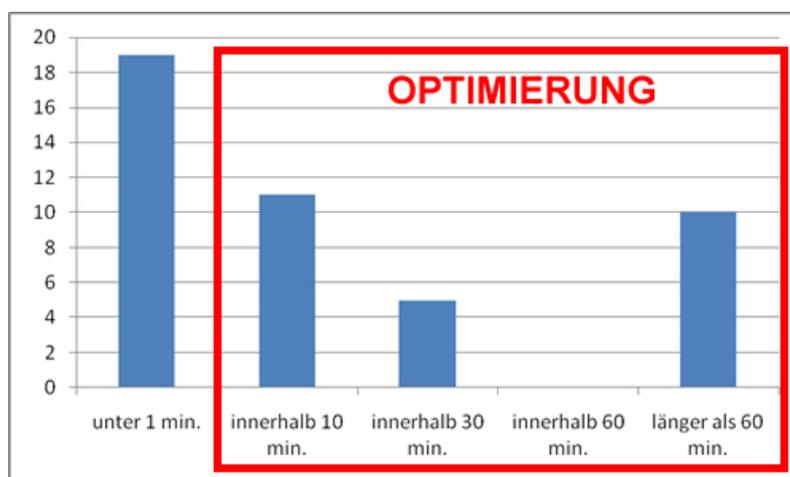


Abbildung 6.2: Prozessbewertung der redaktionellen Betreuung einer Website nach der Optimierung

Abbildung 6.2 zeigt, die, durch den Einsatz der Softwarelösung, wegfallenden Verzögerungen auf Grund der umgangenen Kommunikationswege zwischen Redakteuren und Entwicklern.

6.2 Anlegen einer neuen Website mit vorab installierten WCMS

Durch die Prozessanalyse des Anlegens einer neuen Website mit vorab installierten Web Content Management System und der Interpretation wurde festgestellt, dass die Schwachstellen des Prozesses in der unternehmensübergreifenden Kommunikation und der Durchführung der Tätigkeiten durch Dritte und den daraus resultierenden Verzögerungen liegen.

Um diese Schwachstellen zu schließen wurde der Lösungsansatz für die Steuerung von administrativen Tätigkeiten am Server durch die Softwarelösung entwickelt. Bestimmte Prozesse werden somit nicht mehr von externen Unternehmen, sondern von den eigenen Entwicklern durchgeführt. Wie an dem konkreten Fallbeispiel der New Media Agentur BULLDOGgroup GmbH gezeigt wurde, fallen das Anlegen von neuem Webspace, das Anlegen einer neuen MySQL Datenbank sowie das Anlegen eines neuen FTP-Zuganges nicht mehr in die Zuständigkeit des externen Unternehmens. Diese Prozesse können nun durch die Verwaltungsoberfläche durchgeführt werden. Darüber hinaus bietet die Softwarelösung ADM (siehe Kapitel 5) noch das Automatisierte Installieren und Konfigurieren des WCMS TYPO3 an.

Abbildung 6.3 zeigt, angedeutet durch den roten Kasten und die roten Pfeile, die Optimierung der Kommunikationswege und somit die Beseitigung der Schwachstellen und den daraus resultierenden Verzögerungen.

Um diese Optimierung wiederum in Zahlen zu bewerten, zeigt Abbildung 6.4 die Änderungen der in Kapitel 3.3.2 durchgeführten Prozessbewertung. Durch die veränderten Kommunikationswege und die veränderte Zuständigkeit für die Durchführung der einzelnen Arbeitsschritte innerhalb des Prozesses, wurde der gesamte Prozess hinsichtlich der Dauer, beginnend mit der Auftragserteilung bis hin zur erfolgreichen Durchführung, optimiert.

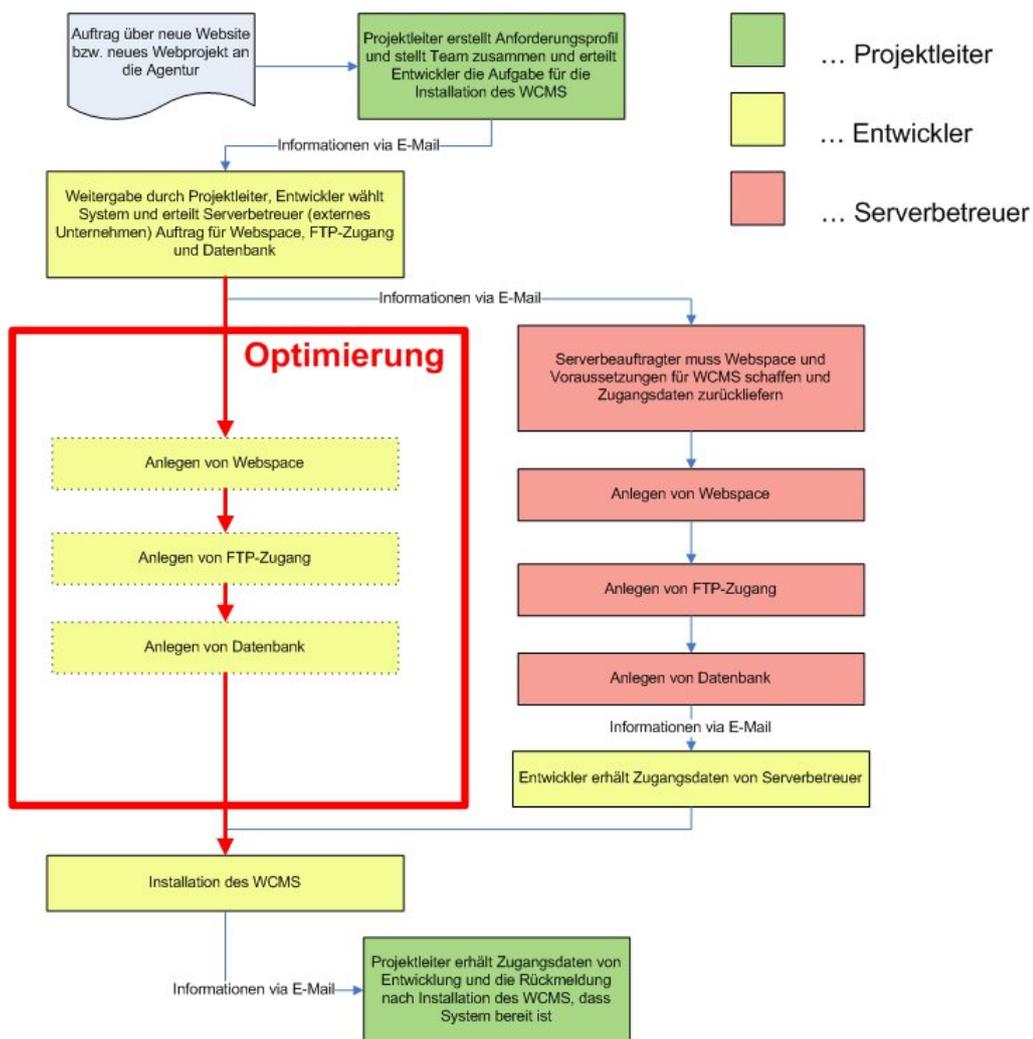


Abbildung 6.3: Ablaufdiagramm des optimierten Prozesses für das Anlegen einer Website mit vorab installierten WCMS

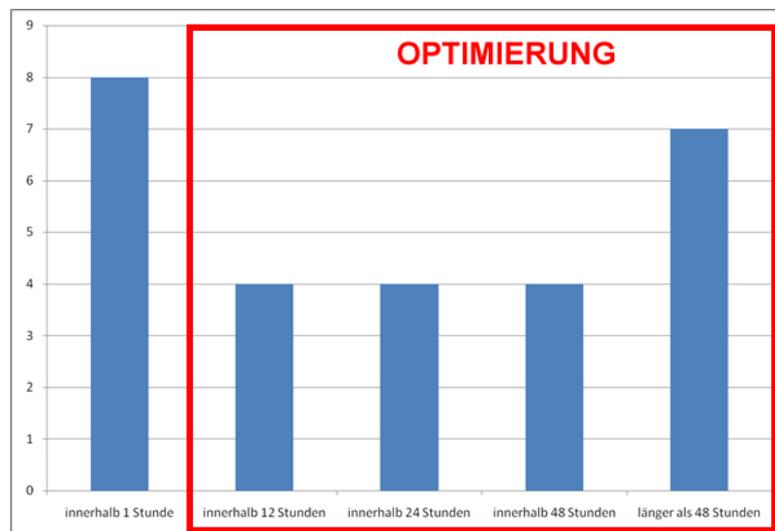


Abbildung 6.4: Prozessbewertung des Anlegens einer neuen Website mit vorab installierten WCMS nach der Optimierung

Kapitel 7

Erweiterungsmöglichkeiten

Da diese Arbeit, den in Kapitel 4 entwickelten Lösungsansatz in Form einer Softwarelösung als Prototyp technisch umgesetzt hat, beschreibt dieses Kapitel sinnvolle Erweiterungsmöglichkeiten für die entwickelte Verwaltungsoberfläche. Allerdings werden mögliche Plugins, die durch die freie Erweiterbarkeit in die Verwaltungsoberfläche integriert werden können, nicht berücksichtigt, da dies zu einer endlosen Liste von möglichen Systemen führen würde.

7.1 zentralisierte Benutzerverwaltung

Die momentane Verwaltungsoberfläche beinhaltet eine Benutzerverwaltung, wobei hier lediglich zwischen Administratoren und Redakteuren unterschieden wird. Diese Benutzerverwaltung ist allerdings aus organisatorischer Sicht und aus Sicht des Unternehmens völlig eigenständig. Um nun das Ziel der Minimierung von administrativen Aufwänden weiterhin konsequent zu verfolgen, wäre eine unternehmensweite zentralisierte Benutzerverwaltung von großem Vorteil.

Als Kernverwaltung könnte beispielsweise ein Active Directory¹ unter Microsoft Windows oder andere LDAP² unterstützende Verwaltungssysteme fun-

¹Active Directory ist ein Verzeichnisdienst der Microsoft Cooperation und stellt eine gemeinsame einheitliche Datenbank dar, die Informationen von Benutzern, Rechnern, Druckern und Anwendungen bereitstellt (vgl. [aM09], Abruf 12.04.2009).

²LDAP steht für Lightweight Directory Access Protocol und ist ein TCP/IP-basiertes Directory-Zugangsprotokoll. Durch den Einsatz von LDAP können unterschiedlichste Informationen aus Verzeichnisdiensten ausgelesen werden (vgl. [JL03], S.5ff).

gieren. Die Verwaltungsoberfläche würde Benutzer nun nicht mehr anhand der eigenen Systemdatenbank, sondern mit Hilfe, der von PHP zur Verfügung gestellten LDAP-Funktionen, gegen die LDAP unterstützende Benutzerverwaltung authentifizieren. Dadurch könnte man unterschiedlichste Eigenschaften von Benutzern, wie das persönliche Profil oder beispielsweise die Zugehörigkeit von Gruppen und deren Gruppenrichtlinien aus der Verwaltung auslesen und in die Verwaltungsoberfläche des ADM übernehmen.

Somit wäre für IT-Administratoren die Möglichkeit gegeben, sämtliche Zugänge über eine einzige Verwaltung zu steuern. Diese Zugänge übertragen sich in weiterer Folge, durch den Einsatz des ADM, auf die Administrations- und Redaktionsoberflächen der importierten Webprojekte. Diese Erweiterung stellt somit einen sinnvollen weiteren Schritt für die Minimierung von administrativen Aufwänden dar.

7.2 Post-Installationen von fehlender Software

Eine zusätzliche Erweiterung stellt die Möglichkeit dar, fehlende Software, die für das Ausführen eines Moduls benötigt wird, nachträglich zu installieren. Beispielsweise beim Anlegen einer neuen MySQL Datenbank, wie in Kapitel 5.5.3 beschrieben, erkennt das Modul durch die Überprüfung der Voraussetzungen, dass der MySQL-Datenbankserver nicht am System installiert ist.

Bei der momentanen Lösung bricht das System automatisch mit einer konkreten Fehlermeldung ab. Eine bessere Variante wäre es die fehlende Software nachträglich noch installieren zu können. Das heißt das System erkennt automatisch welche Software für die Ausführung des Moduls fehlt und bietet an, diese zu installieren. Abbildung 7.1 zeigt den schematischen Aufbau des Moduls, um dies auch technisch umsetzen zu können.

Neben den bisher bekannten Bestandteilen der Module innerhalb der Plugins, ist nun ein weiterer Baustein hinzugekommen. Dieser Baustein nennt sich „Dependencies“ und enthält alle Softwarenamen und Installationswege, diese zu installieren, von denen das jeweilige Modul abhängig ist. Der Ablauf eines Moduls bleibt allerdings ident. Zuerst werden wiederum die Voraussetzungen

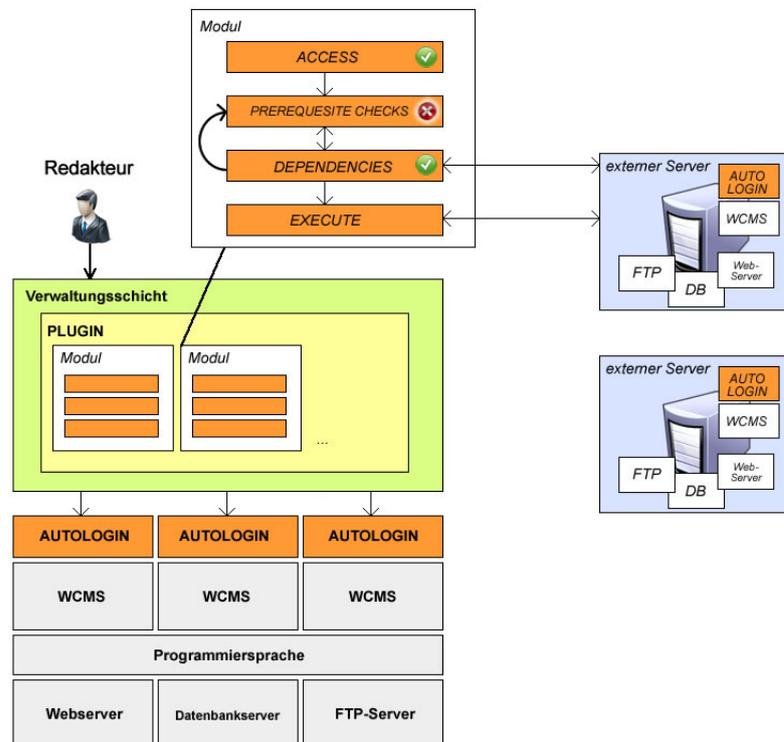


Abbildung 7.1: Schematischer Aufbau der Erweiterung von Modulen für Post-Installationen

für das Ausführen des Moduls überprüft. Schlagen diese Voraussetzungen fehl, bricht nun das Modul nicht ab sondern der Baustein der „*Dependencies*“ wird aktiv und bietet die fehlende Software zur Installation an. Anschließend werden erneut alle Voraussetzungen überprüft. Dieser Vorgang wiederholt sich so lange bis die Überprüfung erfolgreich war und das Modul ausgeführt werden kann.

Aus technischer Sicht enthält der entsprechende Plugin-Ordner, einen Unterordner „*dependencies*“, der die Klasse „*Dependencies*“ enthält. Diese enthält wiederum die Funktionen für die Post-Installationen. Geladen werden diese Dependencies über die XML-Datei „*modules.xml*“. Somit besitzt jedes Modul seine eigenen Abhängigkeiten und kann diese, falls erforderlich, getrennt voneinander durchführen.

7.3 Erweiterungen für das Plugin TYPO3

Neben den bisher genannten Erweiterungen werden in den folgenden Unterkapiteln, drei nützliche und wertvolle Erweiterungen speziell für das Plugin TYPO3 erläutert, das konkret für die Softwarelösung entwickelt wurde.

7.3.1 Wahl von unterschiedlichen TYPO3-Versionen

In der momentanen Version der Verwaltungsoberfläche ADM, wird bei den Modulen des TYPO3-Plugins nur die Version 4.2.6 unterstützt. Das heißt, dass beim Ausführen der Module in Verbindung mit älteren oder gar neueren TYPO3-Versionen Probleme auftreten können. Speziell bei älteren Versionen $< 4.2.x^3$ können durch die zahlreichen Veränderungen ab dieser Version, vermehrt Probleme auftreten.

Um diesen Problemen entgegen zu wirken, sollte es dem Benutzer möglich sein zwischen alten und aktuellen Versionen des WCMS TYPO3 auszuwählen. Die Problematik stellt allerdings nicht die Auswahl im Frontend dar, sondern die Adaptierung der einzelnen Module mit Berücksichtigung der Versionsnummer von TYPO3. Hierzu gibt es grundsätzlich zwei unterschiedliche Lösungsansätze:

1. Es existiert eine Pluginklasse, die wiederum eine Vielzahl von Modulen enthält. Oftmals gleiche Module, die sich nur durch die TYPO3-Versionsnummer und die dadurch entstehenden Abweichungen in der Programmierung unterscheiden.
2. Es existieren innerhalb des Plugin-Ordners, viele Plugins, die sich nur durch die TYPO3-Versionsnummer unterscheiden.

Der erste Lösungsansatz besteht auf das Konzept, dass pro Web Content Management System eine einzige Plugin-Klasse existiert. Dies führt zu einer Vervielfachung der Module innerhalb der Klasse. Der zweite Lösungsansatz adaptiert das Konzept und erstellt für jedes WCMS und jede Versionsnummer eine eigene Plugin-Klasse. Im ersten Lösungsansatz muss somit jedem

³Nähere Informationen zur TYPO3 Version > 4.2 unter <http://www.typo3.org>

Modul die Versionsnummer von TYPO3 zugewiesen werden. Das heißt die Tags `<module>` erweitern sich um ein Attribut „*version*“. Wobei im zweiten Lösungsansatz die Versionsnummer dem Plugin in der Datei „*plugins.xml*“ zugewiesen werden muss.

Dazu ist weiters zu empfehlen, die Verwendung von Wildcards bei Versionsnummern zu berücksichtigen. Dies wiederum setzt voraus, dass die Versionsnummern in einer chronologischen und numerischen Reihe aufeinander folgen. Das heißt beispielsweise, dass ein Modul mit der Versionsnummer „4.2*“ alle untergeordneten Versionsnummern wie „4.2.1“ - „4.2.n“ beinhaltet.

Ob nun der erste oder der zweite Lösungsansatz für eine verbesserte Übersicht sorgt hängt vom Verständnis des Entwicklers ab. Letztendlich unterscheiden sich die beiden Lösungsansätze in der Beschreibung der jeweiligen XML-Datei.

7.3.2 Wahl der Kompatibilitätsversion durch den Benutzer

Bei dieser Erweiterung handelt es sich um die Konfiguration des Web Content Management Systems TYPO3. Durch das Installationsmodul von TYPO3 werden bereits viele Eigenschaften für die Schaffung des Basissystems vorkonfiguriert. Allerdings verhält sich das System immer noch nach der alten TYPO3-Version 3.8. Um dies zu umgehen muss der Administrator im Installationstool den „*Update Analyzer*“ ausführen, der dann das System auf die Version 4.2.y updated. Erst dann verhält sich das Backend in bestimmten Bereichen auch nach der neuesten Version 4.2.x.

Bei diesem Update-Vorgang werden einzelne TYPO3-System-Tabellen adaptiert bzw. neu erstellt. Um diesen Vorgang nun zu automatisieren müsste das Installationsmodul dahingehend erweitert werden, dass die für den Update-Vorgang relevanten SQL-Statements auf der TYPO3-Systemdatenbank durchgeführt werden. Anschließend muss die TYPO3-Konfigurationsdatei „*local-conf.php*“ um den folgenden Eintrag ergänzt werden:

```
1 echo "\$TYPO3_CONF_VARS['SYS']['compat_version'] = '$2';" >> /  
var/www/$1/html/typo3conf/localconf.php
```

Listing 7.1: Shellskript für die Adaptierung der TYPO3-Konfigurationsdatei localconf.php

Die im Listing stehenden Parameter enthalten in *\$1* den *Systembenutzer* und in *\$2* die *Kompatibilitätsversion*. Durch „»“ wird die Datei nicht neu erstellt sondern der Eintrag am Ende der Datei angehängt.

7.3.3 Zuweisung von Benutzern und Benutzergruppen beim Import-Modul

In der momentanen Softwarelösung, kann der Administrator bestehende Webprojekte für seinen eigenen Zugang importieren. Das heißt ihm stehen dann in weiterer Folge, die von ihm importierten WCMS-Administrations- und -Redaktionsoberflächen, durch die Auswahl eines Webprojekts aus der Liste im ADM zur Verfügung. Allerdings sollte vor allem den Redakteuren, denen die Importfunktion nicht zur Verfügung steht, zumindest die Funktion, direkt in den Redaktionsbereich seiner ihm zugeteilten Webprojekte zu gelangen, zur Verfügung stehen.

Um dies zu bewerkstelligen benötigt der Administrator beim Importvorgang die Möglichkeit, das zu importierende Webprojekt für bestimmte Benutzer (Redakteuren und Nicht-Redakteuren), die in der Benutzerverwaltung der Verwaltungsoberfläche enthalten sind, zuzuordnen. Diese Zuordnungen müssten in einer Tabelle in der Systemdatenbank gespeichert werden, wobei diese zumindest die ID (Identifikationsnummer) des Webprojekts und die ID des Benutzers als Datensatz speichern muss. Durch diese beiden IDs kann das System nach erfolgter Anmeldung am System feststellen, welche Webprojekte dem angemeldeten Benutzer zur Verfügung stehen und somit in die Auswahlliste geladen werden.

7.4 Durchführung von Modulen auf externen Serversystemen

Bereits in Kapitel 4 wurde das Ausführen von Modulen auf externen Serversystemen konzipiert. Der Prototyp, seine Plugins und deren Module wurden allerdings nur für lokale Aktivitäten technisch umgesetzt. Das heißt alle Serverprozesse greifen auf den lokalen Server in der VMware Umgebung zu. Allerdings ist der Lösungsansatz für das Ausführen von Modulen auf externen Serversystemen nicht besonders kompliziert.

Wie in Abbildung 7.1 schematisch dargestellt, beinhaltet jedes Modul einen Baustein namens „*Access*“. Dieser Bauteil ist im Lösungsansatz zwar konzipiert, aber technisch nicht umgesetzt worden. Für die Ausführung von Modulen auf externen Servern muss also dieser zusätzliche Teil programmiert werden. Um dies zu bewerkstelligen muss sich ein Skript zu Beginn mit dem externen Server und den übergebenen Zugangsdaten verbinden.

Bei erfolgreicher Verbindung müssen die restlichen drei Bausteine des Modules, so fern das Modul zum ersten mal auf dem externen Server ausgeführt wird, „*PREREQUISITE CHECKS*“, „*DEPENDENCIES*“ und „*EXECUTE*“, auf den externen Server kopiert werden. Anschließend muss das Skript mit Hilfe einer Secure Shell (SSH)⁴ auf dem externen Server angestoßen werden, unter der Bedingung, dass wiederum die Voraussetzungen für das Ausführen des Moduls gegeben sind und alle Abhängigkeiten des Moduls auf dem Server vorhanden sind.

Die größte Einschränkung stellt allerdings das zugrundeliegende Betriebssystem dar. In der momentanen Lösung wird Ubuntu Gutsy Gibbon in der Serverversion 7.10 verwendet. Um die Funktionalität der Module aufrecht zu erhalten, müssen alle weiteren externen Server als Voraussetzung ebenfalls dieses Betriebssystem verwenden.

⁴SSH, die Secure Shell ist ein weit verbreiteter, softwarebasierter Ansatz der Netzwerksicherheit, durch den Daten vor der Übertragung automatisch verschlüsselt werden (vgl. [?], S.2f).

7.5 Sichere Datenübertragen durch HTTPS

Vor allem durch, die in Kapitel 5 beschriebene, für die Modul-Ausführung notwendige Eingabe des „*Root-Passworts*“, ist es wichtig die Datenübertragung abhörsicher und verschlüsselt zu übertragen. Dies ermöglicht der Einsatz des HTTPS-Protokolls unter der Verwendung von Server-Zertifikaten. Dadurch werden die vom Benutzer eingegebenen Daten nicht mehr unverschlüsselt übertragen und bietet somit einen zusätzlichen Schutz vor Angreifern für die Verwaltungsoberfläche.

7.6 Weitere mögliche technische Erweiterungen

Neben den bereits beschriebenen Erweiterungen gibt es viele sinnvolle und mögliche technischen Erweiterungen für das System. Im letzten Unterkapitel werden einige weitere technischen Erweiterungsmöglichkeiten aufgezählt:

- Erweiterungen der Plugins für die Web Content Management Systeme JOOMLA, Wordpress und DRUPAL
- Erweiterungen der Module für andere Softwareinstallationen, wie beispielsweise vsftpd⁵
- Erweiterungen der Module für andere Datenbanksysteme, wie beispielsweise PostgreSQL⁶ oder ORACLE⁷
- Erweiterungen der Module für andere Betriebssysteme, wie beispielsweise DEBIAN, CentOS⁸ oder FreeBSD⁹
- Erweiterung, um die gesamte Softwarelösung als Installationspaket für weitere Entwicklungen und für Implementierungen anzubieten

⁵vsftpd ist ein sicherer FTP-Server, der sich speziell durch seine hohe Geschwindigkeit auszeichnet. vsftpd steht unter der GPL Lizenz für UNIX-Systeme (vgl. [web09a], Abruf 11.04.2009).

⁶Nähere Informationen zu PostgreSQL unter <http://www.postgresql.org/>

⁷Nähere Informationen zu ORACLE unter <http://www.oracle.com>

⁸Nähere Informationen zu CentOS unter <http://www.centos.org>

⁹Nähere Informationen zu FreeBSD unter <http://www.freebsd.org>

Kapitel 7 Erweiterungsmöglichkeiten

Die soeben aufgezählten Punkte, stellen mit Sicherheit nicht alle möglichen Erweiterungen dar. Wie bereits eingangs erwähnt ist das System grundsätzlich frei erweiterbar.

Kapitel 8

Conclusio

Ziel dieser Arbeit war es die notwendigen Maßnahmen aus organisatorischer und technischer Sicht für die Minimierung der administrativen Aufwände bei der Verwaltung und Steuerung von Webprojekten zu erarbeiten. Durch die organisatorische Analyse (siehe Kapitel 3) wurden zwei konkrete Prozesse am Fallbeispiel der New Media Agentur BULLDOGgroup GmbH beschrieben, analysiert und deren Schwachstellen aufgezeigt. Auf Grund dieser Schwachstellen wurde ein Lösungsansatz entwickelt (siehe Kapitel 4).

Der Lösungsansatz gliedert sich in die folgenden drei Teilbereiche:

- Zentralisierung mehrerer Webprojekte, als Lösungsansatz für die Schwachstelle bei der Kommunikation zwischen Redakteur und Entwickler beim Anmeldevorgang für den Prozess der redaktionellen Betreuung einer Website.
- Steuerung von administrativen Tätigkeiten am Server, als Lösungsansatz für die Schwachstellen bei der unternehmensübergreifenden Kommunikation und der Durchführung der Tätigkeiten durch Dritte und den daraus resultierenden Verzögerungen für den Prozess des Anlegens einer neuen Website mit vorab installiertem Web Content Management System.
- Freie Erweiterbarkeit in Form von Plugins, für eine spätere Weiterentwicklung und um die Vielzahl von unterschiedlichen Web Content Management Systemen, die am Markt angeboten werden, in die Lösung integrieren zu können.

Kapitel 8 Conclusio

Dieser Lösungsansatz wurde in Form einer Softwarelösung als Prototyp umgesetzt und anschließend bewertet. Die technische Umsetzung (siehe Kapitel 5) hat gezeigt, dass durch Entwicklung dieser drei Teilbereiche eine Basis für eine frei erweiterbare Softwarelösung geschaffen wurde, die die administrative Aufwände aus den beiden konkreten Prozessen des Fallbeispiels minimiert.

Allerdings ist dem hinzuzufügen, dass der Prozess für das Anlegen einer neuen Website mit vorab installierten WCMS bei weitem mehr Optimierungspotenzial besitzt als der Prozess für die redaktionelle Betreuung einer Website. Dies lässt sich aus den längeren Verzögerungen durch die unternehmensübergreifende Kommunikation und die längere Dauer bei der Durchführung der Tätigkeiten an sich, schlussfolgern.

Nichts desto trotz hat der Prozess für die redaktionelle Betreuung einer Website durchaus das Recht auf Optimierung, nicht nur durch die aufgezeigte Schwachstelle, sondern auch aus Sicht der Usability der Redakteure und aus Sicht der firmeninternen IT, die für die Verwaltung der Benutzer zuständig ist. Durch die Zentralisierung der Webprojekte wurde eine einzige Benutzerverwaltung geschaffen, die auf alle importierten Webprojekte angewendet wird. Dadurch wurde der Verwaltungsaufwand für die Benutzerzugänge drastisch minimiert. Weiters ist es, wie bereits erwähnt, eine enorme Erleichterung aus Sicht der Usability, wenn der Redakteur direkt in die Redaktionsoberfläche durch die einfache Auswahl eines Webprojekts gelangt, ohne sich jedes mal erneut mit seinen Zugangsdaten anmelden zu müssen.

Alles in allem, wurde durch diese Arbeit eine Basisplattform geschaffen, die gewisse administrative Aufwände innerhalb von Prozessen in der Verwaltung und Steuerung von Webprojekten und im Bereich des Webhostings minimiert. Durch die in Kapitel 7 beschriebenen Erweiterungen kann diese Plattform noch mehr administrative Aufwände, durch die Optimierung von bestimmten Prozessen, minimieren. Durch die Entwicklung der Erweiterbarkeit in Form von Plugins kann nahezu jedes Web Content Management System in das System integriert werden und somit beweist dieses System seine Orientierung in die Zukunft. Diese Plattform stellt eine noch nicht angebotene Lösung dar, die dadurch für den Markt und dessen Wirtschaft relevant sein kann.

Abbildungsverzeichnis

| | | |
|-----|--|----|
| 2.1 | Anzahl Hostnamen und aktiver Websites von August 1995 bis Februar 2009 (vgl. [Net09]) | 4 |
| 2.2 | Prozentueller Anteil der Unternehmen und der Haushalte mit Breitbandverbindung 2001-2006 (vgl. [Aus09]) | 5 |
| 2.3 | Google Trends Analyse der Suchanfragen zu Typo3, Joomla und Drupal ([Goo], Abruf 12.04.2009) | 6 |
| 2.4 | Einfluss und Auswirkungen von Zeit und Kosten auf die Qualität und Funktionalität in Software-Projekten (vgl [Ber03], S. 4) | 9 |
| 3.1 | Prozessablaufdiagramm der redaktionellen Betreuung einer Website | 13 |
| 3.2 | Prozessablaufdiagramm für das Anlegen einer neuen Website inklusive vorab installierten WCMS | 15 |
| 3.3 | Anzahl und Dauer der Anmeldevorgänge für die redaktionelle Betreuung im Beobachtungszeitraum Februar 2009 | 17 |
| 3.4 | Anzahl der vom Entwickler in Auftrag gegebenen Arbeiten an die Serverbetreuer und die Dauer für die Durchführung durch diese | 19 |
| 4.1 | Schematischer Aufbau für das Zusammenführen mehrerer Webprojekte | 22 |
| 4.2 | Schematischer Aufbau für die Steuerung von administrativen Tätigkeiten am Server über die Verwaltungsschicht | 24 |
| 4.3 | Schematischer Aufbau des gesamten Lösungsansatzes | 26 |
| 5.1 | Marktübersicht der eingesetzten Webserver für alle aktiven Websites im gesamten Internet ([Net09]) | 28 |

Abbildungsverzeichnis

| | | |
|------|---|----|
| 5.2 | Studie von JoinVision über den Einsatz von MySQL ([Inc09a], Abruf 09.04.2009) | 30 |
| 5.3 | Aufbau der Ordner- und Dateistruktur der Softwarelösung . . . | 33 |
| 5.4 | Screenshot der Verwaltungsoberfläche | 36 |
| 5.5 | Ablaufdiagramm des Anmeldevorgangs bei einem TYPO3-Projekt | 39 |
| 5.6 | Screenshot der Eingabemaske für das Anlegen eines neuen Web-space | 43 |
| 5.7 | Screenshot der Eingabemaske für das Anlegen eines neuen FTP-Zuganges | 46 |
| 5.8 | Screenshot der Eingabemaske für das Anlegen einer neuen MySQL Datenbank | 47 |
| 5.9 | Screenshot der Eingabemaske für das Importieren einer bestehenden TYPO3-Installation | 51 |
| 5.10 | Screenshot der Eingabemaske für das Sichern einer bestehenden TYPO3-Installation | 54 |
| 5.11 | Screenshot der Eingabemaske für das Installieren eines neuen TYPO3-Projekts | 56 |
| 6.1 | Ablaufdiagramm des optimierten Prozesses für die redaktionelle Betreuung einer Website | 63 |
| 6.2 | Prozessbewertung der redaktionellen Betreuung einer Website nach der Optimierung | 64 |
| 6.3 | Ablaufdiagramm des optimierten Prozesses für das Anlegen einer Website mit vorab installierten WCMS | 66 |
| 6.4 | Prozessbewertung des Anlegens einer neuen Website mit vorab installierten WCMS nach der Optimierung | 67 |
| 7.1 | Schematischer Aufbau der Erweiterung von Modulen für Post-Installationen | 70 |

Tabellenverzeichnis

| | |
|--|----|
| 4.1 Auszug über die Vielzahl an Open Source und freien WCMS ([web], Abruf 20.03.2009) | 25 |
| 5.1 Absolute Zahlen der Marktübersicht der eingesetzten Webservers für alle aktiven Websites im gesamten Internet ([Net09]) | 29 |

Verzeichnis der Listings

| | | |
|------|---|----|
| 5.1 | Datei <i>plugins.xml</i> , beschreibt die aktiven und inaktiven Plugins der Verwaltungsoberfläche | 34 |
| 5.2 | Datei <i>modules.xml</i> , beschreibt die aktiven und inaktiven Module eines Plugins | 35 |
| 5.3 | Shellskript für das Anlegen eines neuen Webspace | 44 |
| 5.4 | Shellskript für das Anlegen eines neuen FTP-Zuganges | 45 |
| 5.5 | Shellskript für das Anlegen einer neuen MySQL Datenbank | 48 |
| 5.6 | Shellskript für das Erstellen des auszuführenden SQL-Skripts | 48 |
| 5.7 | PHP Skript für den neuen Anmeldevorgang einer importierten TYPO3-Installation | 51 |
| 5.8 | Shellskript für das Sichern einer bestehenden TYPO3-Installation | 55 |
| 5.9 | Shellskript für das Installieren eines neuen TYPO3-Projekts | 57 |
| 5.10 | Shellskript für das Konfigurieren einer TYPO3-Installation | 59 |
| 5.11 | Shellskript für das Importieren der Systemdatenbank für eine TYPO3-Installation | 60 |
| 7.1 | Shellskript für die Adaptierung der TYPO3-Konfigurationsdatei <i>localconf.php</i> | 73 |

Literaturverzeichnis

- [aM09] Johann Wolfgang Göthe Universität Frankfurt am Main. Active Directory. <http://www.rz.uni-frankfurt.de/services/ads/index.html>, Jänner 2009. Abruf 13.04.2009.
- [Aus09] Statistik Austria. Prozentueller Anteil der Unternehmen und der Haushalte mit Breitbandverbindung 2001-2006. http://www.statistik.at/dynamic/web_de/statistiken/informationsgesellschaft/ikt-einsatz_in_haushalten/015944, März 2009. Abruf 13.04.2009.
- [Ber03] Dipl.-Ing. Johannes Bergsmann. Software Projektmanagement - Wieso scheitern Software-Projekte? <http://www.software-quality-lab.at/swql/uploads/media/Software-Projektmanagement-20030914.pdf>, 2003. Abruf 19.03.2009.
- [Buc09] DATACOM Buchverlag. IT Wissen - HTTPS. <http://www.itwissen.info/definition/lexikon/secure-HTTP-S-HTTP.html>, 2009. Abruf 12.04.2009.
- [CW01] Linda Burman Chuck White, Liam Quin. *XML*. SYBEX, 2001.
- [Dit03] Patrick Ditchen. *Shell-Skript Programmierung*. mitp, 2003.
- [Fri] W. Frisch. Information Retrieval & E-Commerce. <http://www.univie.ac.at/frisch/isegov/aushaengUniWien/VOEinfuehrungWFR1.pdf>. Abruf 12.04.2009.
- [Goo] Google. Google Trends Analyse. <http://www.google.at/trends?q=typo3,joomla,drupal&geo=aut&sa=N>. Abruf 12.04.2009.

Literaturverzeichnis

- [Inc09a] Sun Microsystems Inc. Marktanteil von MySQL. <http://www.mysql.de/why-mysql/marketshare/>, 2009. Abruf 09.04.2009.
- [Inc09b] Sun Microsystems Inc. MySQL 5.1 Referenzhandbuch - Was ist MySQL? <http://dev.mysql.com/doc/refman/5.1/de/what-is.html>, 2009. 12.04.2009.
- [Inc09c] VMware Inc. VMware - Entdecken Sie die Virtualisierung. http://info.vmware.com/content/GLP_DE_virt_LP1Buy?urlcode=PaidSearch_Google_EMEA-Central_German_VI_General_VMware_Search_DE_virt_LP1Buy&src=PaidSearch_Google_EMEA-Central_German_VI_General_VMware_Search_DE_virt_LP1Buy&ossrc=PaidSearch_Google_EMEA-Central_, 2009. Abruf 12.04.2009.
- [itPII09] Software in the Public Interest Inc. Debian - das universelle Betriebssystem. <http://www.debian.org>, 2009. Abruf 12.04.2009.
- [JL03] Dieter Klünter Jochen Laser. *LDAP verstehen - OpenLDAP einsetzen*. dpunkt.verlag, 2003.
- [KL06] Andrea Ertel Prakash Dussoye Elmar Hinz Kai Laborenz, Thomas Wendt. *TYPO3 4.0 - Das Handbuch für Entwickler*. Galileo Press, 2006.
- [Mic09] Microsoft. Häufig gestellte Fragen zu Microsoft Office Excel 2007. <http://office.microsoft.com/de-at/excel/HA101650601031.aspx>, 2009. Abruf 12.04.2009.
- [Net09] Netcraft. Februar 2009 Web Server Survey, März 2009.
- [Ser05] Dr. Georg Serentschy. IKT - Masterplan. http://www.bmvit.gv.at/telekommunikation/bekanntmachung/downloads/IKT_Masterplan.pdf, 2005. Abruf 12.04.2009.
- [Sto04] Andreas Stoeckl. *Content Management mit TYPO3*. Galileo Press, 2004.
- [The04] Thomas Theis. *Einstieg in PHP 5*. Galileo Press, 2004.

Literaturverzeichnis

- [Ubu09] Canonical Ltd. Ubuntu. Ubuntu Server Edition. <http://www.ubuntu.com/products/whatisubuntu/serveredition>, März 2009. Abruf 13.04.2009.
- [Und01] Kim Underwood. Web Hosting: An Overview. <http://www.marcusball.com/opinion/CS3Opinion/90803.pdf>, April 2001. Abruf 03.03.2009.
- [web] List of Content Management Systems. http://en.wikipedia.org/wiki/List_of_Content_Management_Systems. Abruf 13.04.2009.
- [web09a] About vsftpd. <http://vsftpd.beasts.org/>, April 2009. 11.04.2009.
- [web09b] ProFTPD Features & Platforms. <http://www.proftpd.org/features.html>, 2009. Abruf 09.04.2009.
- [web09c] TYPO3 Installation Basics. http://wiki.typo3.org/index.php/Typo3_Installation_Basics, 2009. Abruf 10.04.2009.