

# **Zielgerichtete Anpassung eines WCMS an die Anforderungen einer Kommunikationsplattform - Eine Fallstudie im Bereich der Lehrerbildung**

**Bachelorarbeit**

**im Studiengang Software Systems Science  
in der Fakultät Wirtschaftsinformatik und Angewandte Informatik  
der  
Otto-Friedrich-Universität Bamberg**

vorgelegt von

**Felix Gellner**

angefertigt am

**Lehrstuhl für Medieninformatik  
Universität Bamberg**

**Prüfer: Prof. Dr. Andreas Henrich**

**Beginn der Arbeit: 09.04.2016  
Abgabe der Arbeit: 24.06.2016**

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Das WegE Projekt . . . . .	1
1.2	Zielsetzung und Vorgehensweise . . . . .	3
<b>2</b>	<b>Begriffsklärungen</b>	<b>4</b>
2.1	CMS und WCMS . . . . .	4
2.2	ECMS . . . . .	5
2.3	Web Portal . . . . .	5
<b>3</b>	<b>Projektanalyse und Spezifikation</b>	<b>6</b>
3.1	Der Prozess der Softwareentwicklung . . . . .	6
3.2	Project Blastoff . . . . .	7
3.3	Requirements Engineering . . . . .	7
3.4	Anforderungen an das WegE Projekt . . . . .	7
3.5	Die Wahl der passenden Technologie . . . . .	9
<b>4</b>	<b>Die Wahl des richtigen Content Management Systems</b>	<b>10</b>
4.1	Auswahlkriterien . . . . .	10
4.1.1	Technologie&Architektur . . . . .	11
4.1.2	Anbieterinformationen . . . . .	11
4.1.3	Content Management . . . . .	12
4.1.4	Content Delivery . . . . .	12
4.2	Die Vorauswahl dreier geeigneter Content Management Systeme . . . . .	12
4.3	Sharepoint . . . . .	13
4.4	LifeRay Portal . . . . .	13
4.5	Typo3 . . . . .	14
<b>5</b>	<b>Typo3 - Eine Übersicht</b>	<b>15</b>
5.1	Version und Hintergrund . . . . .	15
5.2	Installation . . . . .	15
5.3	Frontend Struktur . . . . .	16
5.4	Bedienung . . . . .	16
5.5	Funktionalität und Architektur von Typo3 . . . . .	17
5.5.1	Die Backend Module im Überblick . . . . .	17
5.5.2	Typoscript . . . . .	19
5.5.3	Erweiterungen/Extensions . . . . .	20
5.5.4	Templates erstellen . . . . .	20
5.6	Beispielinstallation . . . . .	22

<b>6</b>	<b>Anpassung eines Typo3 Systems an die gesammelten Requirements</b>	<b>25</b>
6.1	Rechtemanagement und Authentikation . . . . .	25
6.1.1	Möglichkeiten . . . . .	25
6.1.2	Implementation . . . . .	25
6.1.3	Evaluation . . . . .	26
6.2	Blog und News . . . . .	26
6.2.1	Alternativen . . . . .	27
6.2.2	Implementation . . . . .	27
6.2.3	Evaluation . . . . .	27
6.3	Forum . . . . .	28
6.3.1	Möglichkeiten . . . . .	28
6.3.2	Implementation . . . . .	29
6.3.3	Evaluation . . . . .	29
6.4	Anbindung an bestehende Systeme wie Opus . . . . .	30
6.4.1	Möglichkeiten . . . . .	30
6.4.2	Implementation . . . . .	30
6.4.3	Evaluation . . . . .	30
6.5	Teilen von wissenschaftlichen Arbeiten . . . . .	30
6.5.1	Möglichkeiten . . . . .	30
6.5.2	Implementation . . . . .	30
6.5.3	Evaluation . . . . .	30
6.6	Weitere nützliche Extensions . . . . .	30
6.7	Wartung und Sicherheit . . . . .	31
<b>7</b>	<b>Analyse des Mehrwerts der WegE Plattform</b>	<b>32</b>
7.1	Sichtung anderer Projekte der Qualitätsoffensive Lehrerbildung . . . . .	32
7.2	Alternativen und Verbesserungsvorschläge . . . . .	32
<b>8</b>	<b>Zusammenfassung</b>	<b>33</b>
8.1	Erkenntnisse . . . . .	33
8.2	Meine Meinung . . . . .	33
	<b>Abbildungsverzeichnis</b>	<b>34</b>
	<b>Tabellenverzeichnis</b>	<b>34</b>
	<b>Literaturverzeichnis</b>	<b>35</b>

# 1 Einleitung

Moderne Webanwendungen und Plattformen beinhalten heutzutage weitaus mehr als die Möglichkeit statische Informationen anzuzeigen. Mit dem Aufkommen des Web 2.0 hat das Internet einen gewaltigen Sprung gemacht und ein weiteres Mal die Welt verändert. Und wenn es auch für einige Leute immer noch Neuland ist, sind die Vorzüge des Internets inzwischen auch bei den traditionelleren Institutionen, wie dem deutschen Staat, mit hohem Ansehen vertreten. So kommt es, dass der Staat Deutschland 2013 ein massives Projekt zur Lehrerbildung ins Leben gerufen hat, das im Namen des WegE Projekts eine staatlich geförderte Online-Präsenz dazu gewinnt. Um solche Webpräsenzen umzusetzen verlässt man sich heutzutage auf ein Füllhorn von Technologien, welche versprechen, die Entwicklung zu erleichtern. Besonders der dynamische Charakter moderner Webseiten erfordert eine andere Herangehensweise als die ursprünglich statischen HTML-Seiten. Diese und noch viele andere Aufgaben übernehmen Web Content Management Systeme für uns. Ist man an dieser Erkenntnis angelangt, bedarf es nur noch der Entscheidung für eines der 1200 verfügbaren Content Management Systeme. Für diese Entscheidung sollte wiederum vorher klar sein, welche genauen Ziele ein Projekt verfolgt. Erst nach all diesen Schritten und einer groben Einarbeitung in die gewählten Technologien lässt sich eine fundierte Zeit- und Kostenabschätzung erzeugen, geschweige denn das Projekt umsetzen. Genau diese, und weitere Dinge, werden in der folgenden Arbeit in einem allgemeinen Licht, und im Bezug auf das WegE Projekt des Freistaat Bayerns, behandelt.

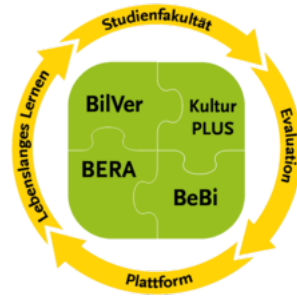
## 1.1 Das WegE Projekt



**Abbildung 1.1** — WegE Logo

Das WegE Projekt der Universität Bamberg ist ein Teil der 'Qualitätsoffensive Lehrerbildung'. WegE steht für 'Wegweisende Lehrerbildung' und hat als Hauptziel die Entwicklung reflexiver Kommunikationsprozesse, vor Allem in Bamberg. Die fachliche Zusammenarbeit zwischen Wissenschaftlern und Lehrern, sowie Lehramtsstudenten soll verbessert werden. Die Lehrerbildung an den fachlichen Stärken der Universität Bamberg wird profiliert, die Zusammenarbeit von Fachwissenschaften und Schulen wird gestärkt, die Fortbildungsangebote werden ausgebaut und das Gesamtprojekt wird für zukünftige Vorhaben aufwändig evaluiert. All dies führt zu dem simpleren, indirekten Ziel der Verbesserung der Schulbildung. Da sich das Projekt aktuell in Entwicklung befindet, können Ziele und Wortlaute sich zum Zeitpunkt der Veröffentlichung dieser Arbeit leicht verändert haben. [weg]

## 1 Einleitung



**Abbildung 1.2** — Die Bestandteile des WegE Projekts

Das WegE Projekt besteht im Wesentlichen aus vier einzelnen Projektvorhaben:

### *KulturPLUS*

Die Organisation KulturPLUS wird sich der Vernetzung geistes- und kulturwissenschaftlicher Perspektiven widmen. Dieses Vorhaben geht von den Fakultäten Geistes- und Kulturwissenschaften und Humanwissenschaften, genauer der Evangelischen Theologie, aus. Durch verschiedene Möglichkeiten soll Lehramtsstudenten die Kompetenz zum Umgang mit den Herausforderungen der kulturellen Vielfalt bezüglich sprachlicher, historischer, geographischer und religiöser Kontexte, beigebracht werden. Die konkrete Umsetzung dieser Ziele ist durch verschiedene Optionen vorgesehen. Darunter ein Wahlpflichtkurs namens *KulturPLUS-Modul*, die Überarbeitung von Schulpraktika, die Veröffentlichung fachwissenschaftlicher Ergebnisse auf der WegE Plattform und viele mehr. (vgl. <https://www.uni-bamberg.de/wege/kulturplus/>)

### *BilVer*

BilVer (BILDungswissenschaft im VERbund)[?] kümmert sich um die fallbezogene Vernetzung der bildungswissenschaftlichen Ausbildungsstelle. [?, ?, ?]<sup>1</sup>. Konkret soll der bildungswissenschaftliche Bereich der Lehramtsstudiengänge an der Universität Bamberg durch eine bessere Kommunikation zwischen Ausbildungsinhalte verschiedener Disziplinen, stärkere Berufsbezüge und innovativere Lehr- und Prüfungsformate, verbessert werden. (vgl. <https://www.uni-bamberg.de/wege/bilver/>)

### *BERA*

hat als Ziel den Aufbau eines Kompetenzzentrums in Bamberg. Dessen Aufgabe wird die Stärkung der beratungsbezogenen Professionsanteile im Studium und die Kooperation mit den Schulen der Region sein. Als zweiter Punkt wird ein Querschnittsmodul erstellt, welches Lehramtsstudenten Beratungskompetenz im schulischen Kontext lehren soll. Im Rahmen dieses Moduls werden im geplanten Beratungszentrum von BERA, praktische Veranstaltungen abgehalten, welche die Durchführung verschiedener Beratungsformate zeigen. (vgl. <https://www.uni-bamberg.de/wege/bera/>)

### *BeBi*

---

<sup>1</sup><https://www.uni-bamberg.de/wege/bilver/>

## 1 Einleitung

BeBi, kurz für Berufliche Bildung wird sich mit der Profilierung der Studiengänge Wirtschaftspädagogik und Berufliche Bildung/Fachrichtung Sozialpädagogik, beschäftigen. Im Studiengang Berufliche Bildung werden eine stärkere inhaltliche Vernetzung der beteiligten Fächer, angestrebt. In Folge dessen wird im beteiligten Fach Psychologie ein Wahlpflichtbereich erweitert. Dieser trägt den Namen Frühe Bildung und Entwicklung und soll mehr berufsbezogene, praktische Module enthalten. Im Studiengang Wirtschaftspädagogik wird ebenfalls eine Erweiterung des Lehrangebots durch das Erstellen neuer Module im sozialpädagogischen Bereich, angestrebt. (vgl. <https://www.uni-bamberg.de/wege/bebi/>)

Unterstützt wird das Projekt von vier weiteren Institutionen, eine davon, Strukturmaßnahme 3: Bildungs- und Internetplattform, hat die Funktion, die Kommunikation und Kooperation zwischen allen Projekten zu gewähren und das Engagement von außenstehenden Personen zu fördern. Im Mittelpunkt steht also die Internetplattform des Projekts WegE. Dies ist die Strukturmaßnahme, welche im idealen Fall von den Ergebnissen dieser Bachelorarbeit am Meisten profitiert. Die genauen Ziele der Strukturmaßnahme 3 werden in Kapitel 2 behandelt. (vgl. <https://www.uni-bamberg.de/wege/plattform/>)

In die Qualitätsoffensive der Lehrerbildung investiert der Bund bis 2023 insgesamt eine halbe Milliarde Euro bis zum Jahre 2023. Der Zeitraum für die Förderung des WegE Projekts ist weniger lang. Diese begann am 01.01.2016 und wird bis zum 30.09.2016 bestehen bleiben.

## 1.2 Zielsetzung und Vorgehensweise

Das grundlegende Ziel dieser Arbeit wird aus dem konkreten Projekt der wegweisenden Lehrerbildung motiviert, wird jedoch auch einen allgemeineren Ton anschlagen, um so eine Relevanz für technisch ähnliche Projekte zu erreichen. In den folgenden Kapiteln werden zunächst die Anforderungen an das System behandelt. Diese Anforderungen wurden hauptsächlich in mehreren Meetings mit den Vertretern aller beteiligten Teilprojekte erhoben. Der Prozess wird in Kapitel 2 behandelt. Anschließend werden die Möglichkeiten der technischen Umsetzung verglichen, Konkret wird der Vergleich verschiedener Content Management Systeme hinsichtlich der Anforderungen gezogen. Im Hauptteil wird die technische Umsetzung der einzelnen, gesammelten Anforderungen im Bezug auf ein Content Management System, genau geprüft, getestet und anhand sinnvoller Kriterien evaluiert. Die technische Umsetzung einzelner Anforderungen geschieht meist mit Erweiterungen zu einem CMS. Hierzu wird die Eignung bereits bestehender Erweiterungen untersucht und passende Erweiterungen zu Testzwecken implementiert. Die Evaluation passender Erweiterungen geschieht vor Allem anhand der Kriterien Zeitaufwand, Stabilität und Wartungsaufwand. In einem kleinen Exkurs werden zum Ende hin die angestrebten Features der WegE Plattform auf ihren Mehrwert untersucht und alternative Optionen durch den Vergleich mit anderen Projekten der Qualitätsoffensive Lehrerbildung verglichen. Zu guter Letzt werden die interessantesten Ergebnisse zusammengefasst.

## 2 Begriffsklärungen

Zum vollen Verständnis der Arbeit werden grundlegende Kenntnisse des Software Engineering und der Web Technologien vorausgesetzt. Weitere essentielle Begrifflichkeiten und Technologien werden nun erklärt.

### 2.1 CMS und WCMS

Ein Content Management System (Abk. CMS), oder auf Deutsch Inhaltsverwaltungssystem, ist eine Software, die bei der Erstellung, Pflege und Planung von Content helfen kann. Vor Allem dann, wenn mehrere Leute an einem Projekt zusammenarbeiten. Heutzutage trifft man solche CMS zumeist im Web, woraus sich der Begriff Web Content Management System, WCMS, ergibt. Diese ermöglichen konkret die Erstellung und Bearbeitung multimedialer Inhalte auf Webseiten ohne Programmierkenntnisse. So kann beispielsweise ein Journalist ohne viel Mühe News auf einer Webseite veröffentlichen. Content Management Systeme umfassen meist folgende Features:

- Möglichkeit, unterschiedliche Rollen und Verantwortlichkeiten an verschiedene Nutzer und Content-Kategorien/Typen zu vergeben
- Identifizieren der möglichen Nutzer und ihrer Rollen
- Definition der Verarbeitungsprozesse als Workflow
- Erstellung und Verwaltung von Templates
- Semantisches Ordnen von Inhalten
- Veröffentlichung von Content

Obwohl WCMS der präzisere Begriff ist, werden diese aufgrund ihrer Verbreitung oft mit dem Oberbegriff CMS betitelt und auch in dieser Arbeit synonym verwendet. Fast immer gliedern sich Content Management Systeme in ein Backend und ein Frontend. Im Backend können sich nur bestimmte Nutzer, wie Administratoren und Autoren einloggen um hier die Seite und deren Inhalte zu verwalten. Dafür ist kein extra Programm nötig, das Backend lässt sich bequem durch den Browser erreichen. Das Frontend ist die Webseite die öffentlich zugänglich ist und jene Inhalte für die Besucher der Webseite präsentiert. Die Liste an bestehender CMS Software ist sehr lang. Die meistverwendeten CMS sind momentan Wordpress, Drupal und Joomla.

### 2.2 ECMS

Enterprise-Content-Management erweitert die Funktionalität eines CMS auf die Ebene einer kompletten Organisation. Das CMS und die resultierende Website sind also eine Komponente eines ECMS. Dabei können diese Systeme je nach Unternehmen sehr unterschiedliche Funktionen übernehmen. Im Wesentlichen helfen sie dabei die Arbeit und Zusammenarbeit innerhalb einer Organisation zu vereinfachen. Konkrete Funktionen ähneln denen des CMS oft sehr, wie das Verwalten von Dateien, beschränken sich jedoch oft auf ein internes Netzwerk von Mitarbeitern.

### 2.3 Web Portal

Ein Web Portal ist eine Webseite, die Informationen aus verschiedenen Quellen uniform bündelt. Einzelne Quellen, auch Portlets genannt, bekommen meist einen Bereich der Website-Oberfläche zugewiesen. Ein typisches Merkmal ist die Anpassung der Portlets durch Drag und Drop. Sehr oft sieht man Web Portale im Intranet von Organisationen als zentrale Anlaufstelle mit personalisierten Inhalten, einheitlichem Design und einem einzigen Login. Erst beim Transfer bestehender Anwendungen in ein Web Portal kommen deutliche Nachteile zum Vorschein, da dieser Transfer sehr komplex werden kann.

Sowohl WCMS als auch Web Portale erstellen Webseiten. Web Portale sind dabei mehr auf die Applikationen fokussiert und WCMS mehr auf den Content.



## 3 Projektanalyse und Spezifikation

*“Those who fail to plan, plan to fail”- Winston Churchill*

Die Planungsphase eines Projekts ist sehr wichtig und erspart, sofern richtig durchgeführt, eine Menge Arbeit. Diese Arbeitersparnis wächst exponentiell mit der Größe des Projekts und trifft somit auch besonders auf ein Projekt wie das der WegE Plattform zu. Der folgende Abschnitt wird mit Hilfe des klassischen Software Engineering das Projekt analysieren. Hier geht es vor Allem um das Erkennen und die Spezifikation der Anforderungen an die Plattform, denn diese beeinflussen natürlich direkt die technische Anpassung, die in Kapitel 6 vorgenommen wird.

### 3.1 Der Prozess der Softwareentwicklung

Der Grund warum Software Engineering existiert, ist die wachsende Komplexität von besagter Software. Um diese Komplexität zu bewältigen verwendet man Software Process Models. [PR10] liefert eine generische Unterteilung eines Software Process Model Frameworks in fünf Bereiche: Kommunikation, Planung, Modellierung, Konstruktion und Einsatz. Diese fünf Bereiche können verschieden angeordnet sein. Bei einer linearen Anordnung wird ein Bereich nach dem anderen abgearbeitet. Eine iterative Anordnung erlaubt es einen oder mehrere Bereiche zu wiederholen, bevor man voranschreitet. Eine parallele Anordnung erlaubt mehrere Aktivitäten zur selben Zeit. In der letzten beschriebenen Anordnung, dem evolutionären Einsatz dieses Frameworks werden die fünf Punkte zyklisch angeordnet.

**Software life cycle** Der generische Softwareentwicklungsprozess, oder auch Systems Development Life cycle genannt, besteht aus fünf Schritten:

1. Specification (Analyse)
2. Architecture & design (Entwurf)
3. Implementation (Implementierung)
4. Testing (Test)
5. Maintenance (Wartung)

Diese Definition ist schon lange nicht mehr die einzige Ansicht auf den Prozess der Softwareentwicklung, doch auch viele neuere Definitionen ähneln dieser stark. Jeder Bereich des Life Cycles ist für sich ein eigenes Fachgebiet und kann je nach Größe und Art des Projekts vom Aufwand

### 3 Projektanalyse und Spezifikation

stark variieren. Dennoch finden sich immer diese fünf Teile in einem zyklischen Ablauf wieder, weshalb es sich als eine gute Basis zur Orientierung innerhalb eines Projekts anbietet. In dieser Bachelorarbeit steht vor Allem die Analyse und der Entwurf im Vordergrund. Zwar werde Ich parallel zu dieser Arbeit eine exemplarische Webseite implementieren, aber nicht die eigentliche WegE Plattform umsetzen. So spielt sich diese Arbeit aus der Sicht des WegE Projekts vor Allem im zweiten Teil ab. Die Wartung spielt auch in dem Sinne eine gewisse Rolle, als dass der Aufwand dafür bei der Implementation möglichst klein gehalten werden soll.

Da nun klar ist, dass vor Allem die Analyse und der Entwurf des WegE Projekts in dieser Arbeit eine Rolle spielen, können diese Teile nun vertieft werden.

## 3.2 Project Blastoff

In der Analyse-Phase geht es vor Allem darum, zu verstehen was gebaut werden soll. Dabei hilft das sehr nützliche Konzept des Project Blastoff.

Im Project Blastoff wird die Realisierbarkeit überprüft, diverse Infos gesammelt und letztendlich entschieden ob das Projekt begonnen werden soll. Konkrete Outputs des Project Blastoff sind: Risikoanalyse, Kostenanalyse, mögliche Technologien, Terminologiesammlung, Stakeholdersammlung, Ziele, Einschränkungen, Umfang/ Eckpfeiler des Projekts.

## 3.3 Requirements Engineering

Requirements Engineering behandelt den Prozess des Dokumentierens, Verwaltens und Findens von Requirements. Auch diese Subdisziplin des Software Engineering teilt sich in weitere Bereiche auf. Wichtig ist hierbei vor Allem das Finden und Prüfen der Requirements. Des Weiteren werden noch die Art des Aufschreibens und die Verwaltung von Requirements als wichtige Teile des Requirements Engineering angesehen.

**Requirement** Ein Requirement wird nach Standard des IEEE Konsortiums als eine Kondition oder Fähigkeit, die für einen User zur Lösung eines Problems notwendig ist, beschrieben. Ausnahmen sind vertraglich zu erfüllende Requirements.

Auf weitere Details wird nun nicht näher eingegangen. Dieses Grundwissen soll im Folgenden dazu dienen die Requirements des WegE Projekts organisiert aufzuschreiben.

## 3.4 Anforderungen an das WegE Projekt

Für die Erhebung der Anforderungen an ein System gibt es verschiedene Möglichkeiten.

*TODO*

### 3 Projektanalyse und Spezifikation

Die nun aufgezählten Requirements wurden noch nicht auf ihre Machbarkeit geprüft, das ist auch Teil dieser Arbeit. Deshalb werden sich nicht alle folgenden Requirements auf der fertigen Plattform wiederfinden. Dies ist außerdem keine vollständige Liste der Requirements an das WegE Projekt, sondern eine Aufzählung der Requirements mit einer Relevanz in Bezug auf die technische Umsetzung.

**Statische Informationen** Als Erstes soll das WegE Projekt über das Internet erreichbar sein, um so Informationen über kommende Termine, Forschungsergebnisse und das Projekt selbst zu verbreiten.

**Newsfeed** Die Plattform soll einen Newsfeed beinhalten, welche Besucher der Seite über neueste Informationen aufklärt.

**Blogs** Einige Einrichtungen der WegE Plattformen werden neben den allgemeinen News einen Blog führen.

**Redakteure** Es muss möglich sein, dass Redakteure aus den einzelnen Abteilungen ohne Programmierkenntnisse Inhalte auf der Plattform einstellen können. Da sich das WegE Projekt aus mehreren Abteilungen zusammensetzt, muss es möglich sein, dass mehrere Redakteure Neuigkeiten auf der Plattform veröffentlichen können. Hier wird bereits die Notwendigkeit eines CMS deutlich. Als nicht-funktionales Requirement schließt sich hier noch, extra hervorgehoben, die Bedienfreundlichkeit für die Redakteure an.

**Forum** Um den Kontakt mit Interessierten Nutzern aufzubauen, soll möglicherweise ein eigenes Forum eingerichtet werden. Auf diesen soll es nur eingeloggten Usern möglich sein, zu diskutieren. Sollte das Forum eine gewissen Größe erreichen, ist es wichtig Moderatoren mit mehr Rechten ernennen zu können, die Struktur und Inhalt des Forums wahren.

**Registrierung** Um an Diskussionen im Forum teilzunehmen sollen verschiedene Möglichkeiten zur Anmeldung angeboten werden. Zuerst ein Registrierungsformular für die Öffentlichkeit und zusätzlich eine Möglichkeit zur Anmeldung mittels Kennnummer der Universität Bamberg.

**Forschungsergebnisse teilen** Nutzer sollen auf der Plattform neue Forschungsergebnisse als Dateidownload erhalten können.

**Mail Formular** Als weitere Möglichkeit zur Kontaktaufnahme soll ein Kontaktformular bereitstehen, das Mails an Ansprechpartner des WegE Projekts sendet.

**Kalender** Ein interaktiver Kalender soll kommende Termine anzeigen können. Zusätzlich soll ein Nutzer hier direkt einen freien Termin für einen Beratungstermin für sich belegen können.

**Search Engine Optimization** Sehr oft spielen Suchmaschinen bei der Popularität einer Website eine Rolle, weshalb SEO ein häufig genanntes Requirement ist. Die Website soll möglichst hoch von Suchmaschinen geranked werden und es soll möglich sein Besucherdaten einzusehen.

### 3.5 Die Wahl der passenden Technologie

Nachdem man sich ein Bild des geplanten Projekts und dessen Anforderungen gemacht hat, gilt es sich für die richtigen Tools/Technologien zu entscheiden. Hierfür gibt es keinen einheitlichen Blueprint und es muss selbst nach bestimmten Kriterien abgewägt werden, welche auch eigens gewichtet werden müssen. Auswahlkriterien für die Wahl der richtigen Technologie können folgende sein:

- Vorwissen des Teams
- Budget für Lizenzen u.ä.
- Entwicklungszeit/Deadline
- Technische Aspekte

Die technischen Aspekte sind sehr zahlreich und werden im nächsten Kapitel vertieft.

## 4 Die Wahl des richtigen Content Management Systems

Dieses Kapitel beschäftigt sich mit der Wahl der passenden Technologie - spezieller noch, des passenden WCMS - für ein geplantes Projekt. Diese Aufgabe wird im Kontext des Software Engineering im Project Blastoff erledigt, der Teil des ersten Schrittes, der Projektanalyse, ist. Anschließend wird eine Vorauswahl von drei CMS grob auf die Eignung für dieses Projekt überprüft. Diese drei CMS sind Sharepoint, LifeRay und Typo3.

### 4.1 Auswahlkriterien

Die Wahl des richtigen CMS kann aufgrund der Fülle von möglichen Optionen sehr schwierig sein. Es gilt das beste Tool für den Job zu finden und hierzu müssen sehr viele Variablen beachtet, gegeneinander abgewägt und gegebenenfalls Kompromisse eingegangen werden. Hierbei spielt die Kompetenz und Vorkenntnis eines Entwicklerteams eine große Rolle, wodurch sich eine solche Entscheidung schwer generalisieren lässt. Grundsätzlich lassen sich Auswahlkriterien in vier Punkte untergliedern: Technologie&Architektur, Content Management, Content delivery und Anbieter. RÜPING[Rue10] bietet zu jedem dieser vier Punkte eine ausführliche Checkliste mit wichtigen Fragen, die einem bei der Entscheidung helfen.

**CmsMatrix** Zur Beantwortung der Checklisten, kann die Webseite CMSMatrix<sup>1</sup> eine Hilfe sein. Sie listet über 1000 verschiedene Systeme, die sich einzeln auswählen und anhand sehr vieler Kriterien vergleichen lassen. Die Kriterien untergliedern sich dabei in verschiedene Kategorien:

- Systemanforderungen: Hier können Dinge wie Programmiersprache, Serveranforderungen und Ähnliches verglichen werden. Für das WegE Projekt ist natürlich wichtig, dass das System im Rechenzentrum der Universität läuft.
- Sicherheit: Immer wenn externe User sich in ein System einloggen können stellt sich schnell die Frage nach Sicherheit. In dieser Kategorie lassen sich sicherheitsspezifische Aspekte, wie Art der Authentifizierung oder SSL Möglichkeiten, vergleichen. Dies spielt im WegE Projekt zwar eine große Rolle, sollte jedoch kein großer Entscheider werden, da der Punkt Sicherheit von den meisten Systemen ausreichend abgedeckt ist.
- Support: In diesem Unterpunkt lässt sich vor Allem gut erkennen, wie leicht ein Entwickler eines CMS an Hilfe kommt. Es werden verschiedene Möglichkeiten an Hilfe zu gelangen,

---

<sup>1</sup><http://www.cmsmatrix.org/>

## 4 Die Wahl des richtigen Content Management Systems

verglichen. Besonders wichtig für einen einzelnen Entwickler ist hier ein Punkt zum Vergleich der Entwickler-Communities und Foren.

- Bedienbarkeit: Hier lässt sich die Existenz von Features, die einem das Bedienen des CMS erleichtern, anzeigen. Dazu gehören zum Beispiel ein WYSIWIG-Editor, eine Undo Möglichkeit oder auch ein Spell Checker.
- Performance: Hier wird anhand von Ladezeiten und Caching Möglichkeiten die Performance verglichen.
- Management: In diesem Punkt lässt sich schnell erkennen, wie angenehm die Arbeit eines Administrators im jeweiligen CMS sein wird. Es wird verglichen wie ein CMS mit Templating umgeht, wie es Mehrsprachigkeit umsetzt, ob Crons leicht umzusetzen und weitere ähnliche Aspekte.
- Interoperabilität: Dieser Unterpunkt listet die Möglichkeiten eines CMS zur Interoperabilität. Dies bedeutet konkret die Möglichkeit von Datenaustausch durch XHTML oder RSS. Desweiteren listet es auch den Support von HTML5 und UTF8, welche beide auf der WegE Plattform von Vorteil wären.
- Flexibilität: Hier lassen sich vom CMS eingebaute Möglichkeiten zur Nutzung von mehrsprachigem Content, der Eingabe von Metadaten oder der Möglichkeit des URL Rewriting vergleichen.
- Applikationsumfang: Fast jede mögliche Funktion einer Webseite existiert unter dem Punkt Applikationsumfang als Auswahlkriterium um schnell herauszufinden ob ein CMS diese Funktion bereits anbietet, oder als Addon zur Verfügung steht. Einige exemplarische Funktionen sind ein Blog, ein Mail-Formular oder eine Kalenderfunktion.
- Commerce: Sucht man ein CMS für einen Online Shop lassen sich hier wichtige Funktionen wie ein Shopping Cart, vergleichen.

Mit Hilfe von CmsMatrix und der offiziellen Seite des jeweiligen CMS sollten sich alle folgenden Fragen beantworten.

### 4.1.1 Technologie&Architektur

Die Fragen der Checkliste drehen sich hier um die unterliegende Technologie, die Systemarchitektur, verfügbare Systemkomponenten, Systemvoraussetzungen und andere nicht-funktionale Eigenschaften des Systems. Es wird nach der grundlegenden Technologie/Programmiersprache und dem Datenformat gefragt. Es wird gefragt, wie Content und Layout separiert sind und ob eine API zum Content existiert. Allgemeiner wird untersucht, welche großen Komponenten das System ausmachen und wie gut dessen Performance ist. Zuletzt werden Fragen nach Anforderungen an Betriebssystem, Datenbank, Hardware und Web Server, gestellt

#### 4.1.2 Anbieterinformationen

Neben technischen Details sind auch Information über den Vertrieb des CMS und den Anbieter dessen interessant. Besonders wichtig ist das Lizenzmodell und ob es sich um ein Open Source Tool handelt. Falls es sich um ein proprietäres Tool handelt, muss ein genauerer Blick auf das Lizenzierungsmodell geworfen und dessen Kosten ermittelt werden. Diese hängen oft mit der Anzahl von Nutzern zusammen. Weitere wichtige Faktoren sind der offizielle Support und die Stärke der Community um das Tool. Ein starker Support mindert eventuelle Risiken bei der Entwicklung. Eine gute Community erleichtert den Einstieg in ein CMS und kann gegebenenfalls als Ersatz für den Support dienen. Es ist sicherlich eine gute Idee sich an die populäreren Optionen zu halten, da man hier mit dem besten Support und der aktivsten Community rechnen kann, was Einstieg, Support und Lernkurve positiv beeinflusst. Vor Allem als sehr kleines oder Ein-Mann-Team lohnt sich ein Blick auf die CMS mit den stärksten Communities.

#### 4.1.3 Content Management

Unter Content Management fragt Rüping [Rue10] wie gut ein CMS seine Grundfunktionalität erledigt. Die Checkliste untersucht die Modellierung des Content, welche Typen von Content möglich sind, wie mit Links umgegangen wird und wie Content importiert/expoertiert werden kann. Darüber hinaus wird der Editor Client auf Kriterien wie Anpassungsmöglichkeiten, Validierungsmechanismen, Spell Checker und Ergonomie, untersucht. Zuletzt wird nach dem Workflow gefragt und wie dieser angepasst werden kann.

#### 4.1.4 Content Delivery

Im letzten Punkt geht es um den technischen Zustellungsprozess von Content an die Nutzer. Es wird nach der Möglichkeit eigene Komponenten zu integrieren und Support von SOA gefragt. Eine Möglichkeit eigene Komponenten, wie zum Beispiel Extensions, in die Umgebung zu integrieren wird untersucht. Es wird untersucht ob sich eigene Controller-Logik implementieren lässt, was eine Anpassung des Routing von bestimmten Seiten erlauben würde. Es wird gefragt ob das CMS eine Suchengine oder eine Personalisierungsengine bietet. Mit der Personalisierungsengine ist etwas, wie die Oberfläche eines Web Portals gemeint. Bei den Suchengines ist interessant welche Dateiarnten gesucht werden können und wie schnell die Suche ist.

### 4.2 Die Vorauswahl dreier geeigneter Content Management Systeme

Die Vorauswahl von geeigneten Content Management Systemen für das WegE Projekt wurde durch mehrere Faktoren beeinflusst. Die größte Rolle spielt dabei zweifelsohne das Rechenzentrum der Universität Bamberg, die dort schon bestehenden Systeme und die vorhandene Expertise zur langfristigen Betreuung des WegE Projekts. Anders formuliert, die Vorkenntnisse des Teams. Vor Allem mit Typo3 hat das Rechenzentrum Erfahrung, Sharepoint wird seit kurzem genutzt. Die sehr populären Optionen Joomla! und Drupal wären durchaus geeignet für

das Projekt, bieten jedoch zu wenig Mehrwert zu Typo3 um die die Zeit zur Einarbeitung zu rechtfertigen. Die Größe und Ambiguität des Projekts schließt außerdem das populärste aller CMS Wordpress, aus, welches sich eher für kleinere Projekte eignet. Sollte die Vorkenntnis eines Teams sich jedoch nur auf Wordpress beschränken, ist es trotzdem durchaus eine mögliche Wahl. Somit landete die Vorauswahl bei Sharepoint, Typo3 und LifeRay, welche nun vorgestellt und geprüft werden.

### 4.3 Sharepoint



**Abbildung 4.1** — SharePoint Logo

Sharepoint<sup>2</sup> ist ein Produkt von Microsoft und wurde hauptsächlich für die Verwendung innerhalb von Unternehmen entwickelt. Da das ganze System ausschließlich im Browser läuft gibt es keine Einschränkung auf das Betriebssystem Windows, wie man vielleicht denken könnte, auch wenn das Standarddesign hierfür optimiert wurde. Es dient als gemeinsamer Ort zum Speichern, Strukturieren und Freigeben von Dateien und Informationen. Dafür verwendet das CMS einige spezielle Sharepoint-Technologien:

- SharePoint Online lässt ein Unternehmen einen Cloud-Dienst von Microsoft verwenden, anstatt einen eigenen Server aufzusetzen.
- SharePoint Foundation ist die grundlegende Technologie für die SharePoint Webseiten, welche in verschiedenster Form auftreten können. Am prominentesten natürlich Seiten zur Zusammenarbeit an Listen, Daten und Dokumenten, aber außerdem noch Blogs und Wikis. Foundation ist eine kostenlose Edition von SharePoint.
- SharePoint Server ist die kostenpflichtige Edition und bietet selbstverständlich weit mehr Features als SharePoint Foundation. Diese sind zum Beispiel Enterprise Content Management, Business Intelligence, unternehmensweite Suche, persönliche Websites und ein Newsfeed. Die Kosten hierfür lassen sich schwer abschätzen, da dabei eine Reihe von Variablen eine Rolle spielen. Die reinen Kosten für die Lizenzierung belaufen sich zum Stand der Arbeit auf 4,20 Euro bis 8,40 Euro pro Benutzer pro Monat.
- SharePoint Designer erlaubt es leicht Layouts, Workflows und Verbindungen zu externen Datenquellen anzupassen.

### 4.4 LifeRay Portal

Das Hauptprodukt von Liferay heißt Liferay Portal<sup>3</sup> und hat den Fokus auf Businesslösungen und Geschäftsprozessen. Es werden zwei Varianten angeboten, LifeRay Portal CE ist lizenzkos-

<sup>2</sup><https://support.office.com/de-de/article/Was-ist-SharePoint-97b915e6-651b-43b2-827d-fb25777f446f>

<sup>3</sup><https://www.liferay.com/products/liferay-portal>





**Abbildung 4.2** — Liferay Logo

tenfrei und Open Source. Liferay Portal EE ist die kommerzielle Variante und bietet zusätzlich zur freien Version offiziellen Support und Langzeitunterstützung. Liferay Portal unterteilt sich in drei Teile:

- Liferay Portal: Die Kernfunktionalitäten sind die Authentifizierung von Nutzern mit Single Sign-on, Personalisierung von Seiten durch Drag und Drop, automatischer Dateiupload mit WebDAV (WEB-based Distributed Authoring and Versioning), Suchen und Taggen von Web-Inhalten und Mehrsprachigkeit.
- Liferay CMS: Zusätzlich wird Liferay CMS geboten, das auf Liferay Portal aufbaut und typische CMS Aufgaben übernimmt.
- Liferay Collaboration: Als letztes steht das Angebot Liferay Collaboration zur Verfügung. Hier finden sich Komponenten, wie Foren, Wikis, Blogs Kalender und Ähnliches.

Nach dem ersten Eindruck ist Liferay eine durchaus geeignete Wahl für das WegE Projekt, alle Anforderungen lassen sich hiermit umsetzen.

## 4.5 Typo3



**Abbildung 4.3** — Typo3 Logo

Typo3 ist ein System, mit dem das Rechenzentrum der Universität Bamberg bereits sehr vertraut ist, da die offizielle Seite der Universität auch mit Typo3 betrieben wird. Es ist Open Source und frei von Lizenzkosten. Des Weiteren zeigt CMSMatrix, dass alle gesammelten Anforderungen an die WegE Plattform entweder vorhanden sind oder als kostenloses Addon zur Verfügung stehen. Als Sahnehäubchen kommt hinzu, dass die zukünftigen Redakteure der WegE Plattform bereits Erfahrung mit dem Backend von Typo3 haben und somit die Einarbeitung in ein neues System wegfällt.

Ein Blick in die Gliederung verrät dem aufmerksamen Leser, dass die Wahl des CMS auf Typo3 fiel und diesem CMS die nächsten zwei Kapitel gewidmet sind. Somit wird alles Weitere zu Typo3 dort ausführlich erklärt.

## 5 Typo3 - Eine Übersicht

Die Wahl des CMS fiel nun durch die ausreichende Recherche im vorigen Kapitel auf Typo3 und wird als Grundlage für alle technischen Aspekte dienen, weshalb Typo3 in dieser Bachelorarbeit einen essentiellen Teil ausmacht. In diesem Kapitel werden zunächst alle nötigen Grundlagen rund um Typo3 geklärt, um im anschließenden Kapitel die Anpassung eines Typo3 Systems an die zuvor gesammelten Anforderungen informiert vornehmen zu können.

Die meisten hier präsentierten Informationen entspringen den offiziellen Typo3 Docs<sup>1</sup> [typ] und einem Lernkurs des Rheinwerk Verlages zur neuen Typo3 Version 7[Wag16]. Spezifische Verweise werden natürlich extra angegeben.

### 5.1 Version und Hintergrund

Typo3 ist ein Open Source CMS, ursprünglich entwickelt von Kasper Skårhøj und basiert auf der Programmiersprache PHP. Die Entwicklung begann bereits 1997 als Nebenprodukt von Skårhøjs Tätigkeit als Webdesigner. Der Name Typo3 entstand, nachdem Skårhøj durch einen Schreibfehler (engl. Typo) die Arbeit mehrerer Tage löschte. Die Zahl 3 ist der damals aktuellen Version geschuldet, wurde jedoch weiterhin beibehalten. Das System ist Open Source unter der GPL-Lizenz. Bei der Popularität von Open Source Web Content Management Systemen belegt Typo3, je nach Quelle, meist den fünften Platz<sup>2</sup>. Typo3 umfasst eine Fülle an Versionen und spaltete sich 2015 sogar in zwei verschiedene Kernentwicklungsteams. Eines das weiter an Typo3 CMS arbeitet und eines, das an Typo3 Neos arbeitet. Diese Bachelorarbeit basiert auf der neuesten Version von Typo3 CMS 7.6 LTS. Version 8 wird jedoch auch noch 2016 veröffentlicht.

### 5.2 Installation

Die Installation und Erstellung einer Typo3 Website gestaltet sich relativ umfangreich. Je nach Vorhaben muss mit komplexen Einstellungen am serverseitigen System experimentiert werden. Zum Zweck der Entwicklung bietet es sich jedoch an das Typo3 System zunächst lokal zu installieren. Der hierfür nötige Server kann entweder durch ein Programm wie Wamp oder Xampp, welche eine fertige Apache-Distribution liefern, bereitgestellt werden oder durch das Installieren eines Servers auf einer virtuellen Maschine. Die vorgenommenen Implementationen in Kapitel 4 wurden mit Wamp<sup>3</sup> realisiert.

---

<sup>1</sup><https://docs.typo3.org/>

<sup>2</sup><http://trends.builtwith.com/cms/open-source> Mai 2016: Platz 5

<sup>3</sup><http://www.wampserver.com>

## 5.3 Frontend Struktur

Das Frontend ist von Typo3 in bestimmte Bereiche unterteilt:

- Header: Zeigt Inhalte ganz oben auf der Website, wie ein Logo
- Menü: Beinhaltet das Hauptmenü
- Content: Unterteilt sich in drei Teile, Links, Mitte und Rechts
- Footer: Zeigt Inhalte am Ende der Webseite

Wo genau diese Bereiche auftauchen lässt sich mittels eines Templates definieren, das mit Typoscript beschrieben wird. Im Backend finden sich genau diese Teile wieder und lassen sich dort mit Content befüllen.

## 5.4 Bedienung

Typo3 generiert letztendlich eine Website. Diese lässt sich jedoch durch das Typo3 Backend entscheidend verändern. Die Bedienung geschieht fast immer durch dieses Backend, welches von der Startseite aus durch das Anhängen von `/typo3` in der Adresszeile zu erreichen ist. Hier werden im Grunde drei Aufgaben erledigt. Templates erstellen um das Design der Website anzupassen, Typo3 Konfigurationen vornehmen und die Erstellung von Inhalten.

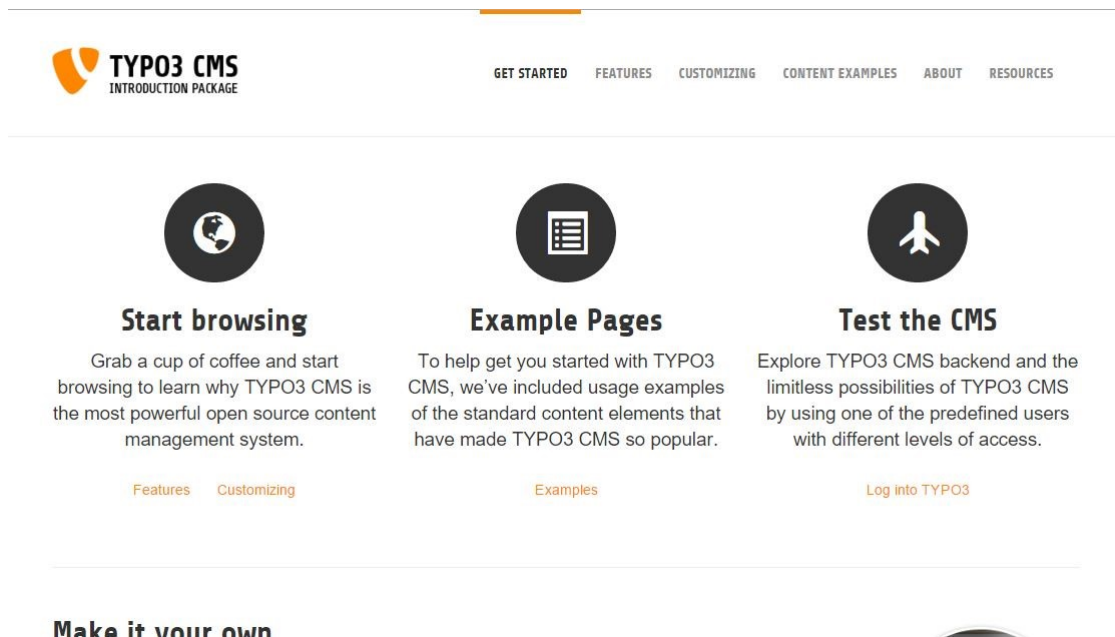


Abbildung 5.1 — Das Frontend gleicht einer normalen Website.

Das größte visuelle Merkmal der neuen Typo3 Version 7.6 LTS ist das neue, klarere Design des Backend von Typo3. So wird es für Anwender ohne Programmierkenntnisse einfacher sich

## 5 Typo3 - Eine Übersicht

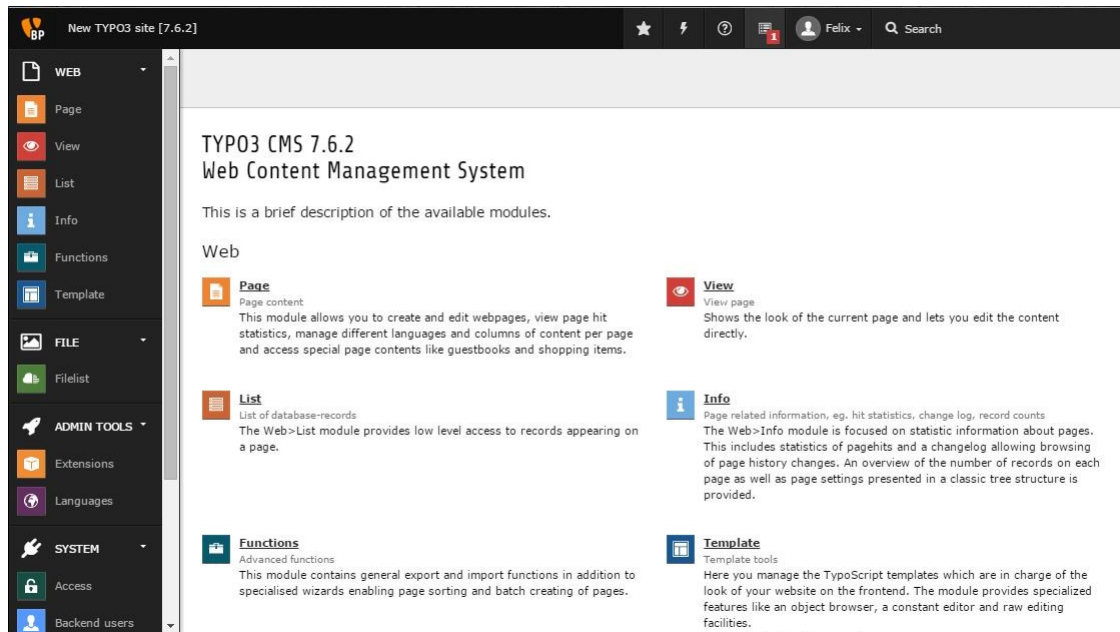


Abbildung 5.2 — Das Typo3 Backend

zurechtzufinden und Inhalte zu erstellen. Dies geschieht hauptsächlich über einen integrierten WYSIWYG-Editor.

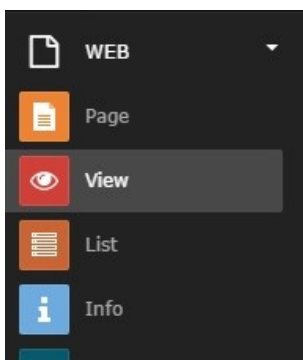
Dennoch enthält die Backend sehr viele Optionen, versteckte Knöpfe und Eigenheiten, die einer Erklärung bedürfen.

### 5.5 Funktionalität und Architektur von Typo3

Es handelt sich an dieser Stelle zwar um eine wissenschaftliche Arbeit und kein Handbuch, doch es ist ein wesentlicher Bestandteil dessen, die Typo3 Umgebung zu verstehen um analysieren zu können, wie sie am geschicktesten auf die gesammelten Anforderungen anzupassen ist. Daher folgt nun eine komprimierte Übersicht über den Umgang und die Entwicklung von Typo3 Websites und den Workflow beim Erstellen dieser.

#### 5.5.1 Die Backend Module im Überblick

Das Backend sieht seit dem Update deutlich einladender aus, die Bedienung gestaltet sich jedoch immer noch anspruchsvoller als bei vergleichbaren Konkurrenten. Deshalb findet sich hier zunächst eine Liste mit Erklärungen zur Navigationsleiste im Backend:



## 5 Typo3 - Eine Übersicht

- **Page:** Hier lassen sich die Einzelseiten der Webanwendung auf einem hohen Level erstellen und verwalten. Auch wenn noch keine Seite angelegt wurde, sieht man im Hierarchiebaum bereits einen Eintrag. Dieser stellt sozusagen die Wurzel aller zukünftigen Seiten dar, zeigt selbst jedoch keine Seite an.
- **View:** Hier ist es möglich, das Frontend zu begutachten, ohne das Backend zu verlassen.
- **List:** In diesem Menüpunkt findet sich eine Übersicht der Elemente, die dem Seitenbaum zugeordnet sind. Das können Webseiten oder auch Komponenten von Plugins sein.
- **Info:** Hier lassen sich lediglich Informationen, wie zum Beispiel Erstellungsdatum, zu einzelnen Komponenten anzeigen.
- **Functions:** Dieser Bereich beherbergt einige nützliche Hilfsfunktionen, um schneller Änderungen, wie Seitensortierung, durchzuführen.
- **Template:** Hier werden alle Templates der Seite verwaltet.
- **Filelist:** Hier werden alle Dateien des sogenannten Filemount verwaltet. Meist handelt es sich um Dateien, die Redakteure in Einträge integrieren. Aber auch schnelle Änderungen an CSS-Dateien lassen sich hier durch einen integrierten Texteditor vornehmen.
- **Extensions:** In diesem Bereich werden alle installierten Erweiterungen gelistet. Man hat dort die Möglichkeit diese zu aktivieren und deaktivieren. Zu bemerken ist, dass das nicht mit allen Erweiterungen möglich ist, da einige Grundfunktionalitäten von Typo3 auch als Erweiterungen gelistet sind und diese lassen sich nicht manipulieren. Darüber hinaus lassen sich hier neue Erweiterungen suchen und herunterladen.
- **Languages:** Dieser Punkt hilft bei der Verwaltung von mehrsprachigen Seiten. Auch das Backend lässt sich in mehreren Sprachen anzeigen.
- **Access:**
- **Backend users:** Hier lassen sich die Nutzer des Backend verwalten. Es lassen sich neue Backend User anlegen und auch spezifische Rechte dieser User konfigurieren. Typische Rollen sind Administrator/-in und Redakteur/-in.
- **Install:** Hier lassen sich durch das Install-Tool einige gravierende Einstellungen vornehmen, wie das Zurücksetzen von Passwörtern und ähnlichen Interaktionen mit der Datenbank. Außerdem

## 5 Typo3 - Eine Übersicht

zeigt diese Unterseite die Konfigurationsdetails der Umgebung an.

- Log: In diesem Bereich findet sich ein Protokoll über alle Nutzeraktionen und aufkommende Fehler.
- DB check: Hier lassen sich Statistiken über die Einträge der Datenbank anzeigen und Suchen in der Datenbank direkt durchs Backend durchführen.
- Configuration: In diesem Punkt findet sich eine recht unübersichtliche Liste von Konfigurationsvariablen.
- Reports: Der Report liefert eine Liste von Nachrichten über den Lauf der Webseite und mögliche Probleme.

### 5.5.2 Typoscript

Typoscript ist die Skriptsprache von Typo3. Es handelt sich dabei nicht um eine richtige Programmiersprache, was sich auch schon vom Namen her vermuten lassen könnte. Das zeigt sich auch in der Art und Weise wie Typoscript von Typo3 verarbeitet wird, nämlich als sehr langes PHP Array und einem internen Parser. Typoscript ist mehr dafür ausgelegt dem Ersteller der Webseite eine Möglichkeit zu geben, zu bestimmen, wie Typo3 mit bestimmten Objekten, wie zum Beispiel Bildern, Menüs oder Templates, umgehen soll. Es ist so also hauptsächlich für die spezifische Ausgabe von Content zuständig. In Typoscript wird immer mit Objekten gearbeitet, denen man bestimmte Eigenschaften zusprechen kann.

Eine Typoscript Datei, die in jedem Template enthalten ist, ist die setup.ts. Aus dieser ist nun ein kleiner Schnipsel zu sehen.

```
page = PAGE
page{
  includeCSS.style = fileadmin/Template_WegE/bootstrap/css/style.css

  10 = FLUIDTEMPLATE
  10{
    file = fileadmin/Template_WegE/index.html
    layoutRootPath = fileadmin/Template_WegE/layouts/
    partialRootPath = fileadmin/Template_WegE/partials/

    variables{
      siteName = TEXT
      siteName.value = WegE Fallstudie
    }
  }
}
```

}

Zu Beginn wird eine Variable mit dem Namen `page` gesetzt, welche die ganze Seite präsentiert. Diese bekommt in Zeile 3 ein neues Stylesheet zugewiesen, das dann automatisch eingebunden wird. Danach wird eine Variable zum Fluidtemplate. Die Zahl 10 präsentiert dabei einen Platz im PAGE Array, das danach aus Typoscript generiert wird. Hier ist die Reihenfolge oft wichtig. Es ist üblich Anweisungen mit in Zehnerschritten zu nummerieren, sodass man später eventuell vergessene Anweisungen noch zwischen zwei Zahlen einfügen kann. Für das Fluidtemplate werden nun die Pfade zu den Layouts und Partialen definiert. Layouts und Partialen sind Einzelteile eines Templates. Genauer dazu im Verlauf dieses Kapitels. Darunter wird noch mittels Typoscript der Name der Seite definiert, welcher später an vielen Stellen verwendet werden kann.

### 5.5.3 Erweiterungen/Extensions

Ein großer Vorteil und Grund für die Wahl von Typo3 als CMS sind die Erweiterungen oder Extensions. Inzwischen existieren tausende von Erweiterungen für Typo3. Allerdings muss dazugesagt werden, dass ein großer Teil dieser veraltet ist, nicht mehr gepflegt wird oder gar überhaupt nicht auf der neuen Version lauffähig gemacht werden kann. Das liegt auch daran, dass viele der Extensions von freien Programmierern in ihrer Freizeit entwickelt werden. So gilt es etwas vorsichtig bei der Auswahl passender Extensions zu sein. Zu finden sind diese Extensions auf der offiziellen Seite unter [typo3.org/extensions/repository](http://typo3.org/extensions/repository). Wichtig ist es hier auf die Kompatibilität zu achten, die dort angezeigt wird. Auch findet sich hier meist eine Dokumentation zur Einrichtung und weiteren Infos zur jeweiligen Erweiterung. Die Erweiterungen der eigenen Typo3 Anwendung lassen sich durch den Menüpunkt *Erweiterungen* im Backend verwalten.

### 5.5.4 Templates erstellen

Beim Erstellen einer Typo3 Website bekommt der Entwickler für gewöhnlich ein Frontend Design der Seite geliefert und hat dann die Aufgabe dieses Design in Typo3 zu integrieren. Das geschieht durch das Schreiben von Typoskript, das Installieren von Extensions, der Anpassung des Backend und die Entwicklung von Templates. Der Begriff Template ist bei Typo3 mehrfach belegt, deshalb vorerst eine Unterscheidung:

- HTML-Template: Die HTML-Vorlage, welche meist ein Designer liefert.
- TypoScript-Template: Alle Typoscript-Anweisungen, die Ausgabe von Menüs und Ähnlichem steuern.
- Fluidtemplate: Mit Typo3 lassen sich im HTML-Code Fluid-Funktionen verwenden. Diese erlauben Variablen, Bedingungen und das Aufteilen des HTML-Codes in Partialen.

Ein Typo3 Fluidtemplate besteht also aus einem Layout, das Dinge wie head und header enthalten kann, einem Template, das den Inhalt repräsentiert und Partialen, die mehrmals auf einer Seite vorkommen können.

## 5 Typo3 - Eine Übersicht

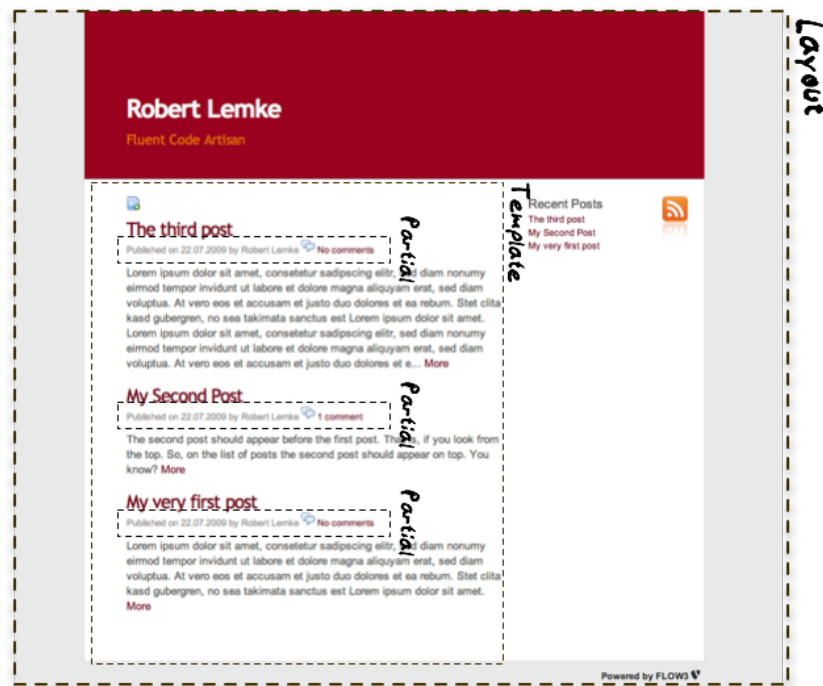


Abbildung 5.4 — Typo3 Template Komponenten

Im Folgenden geht es vor Allem um das Fluidtemplate, welches aus dem HTML-Template erzeugt wird. Für das Fluidtemplate ist es heutzutage üblich eine Template-Extension zu erstellen. Das bedeutet, dass das Template in einzelnen Dateien auffindbar ist, statt in der Datenbank, was die Versionierung und Weitergabe eines Templates erheblich vereinfacht. Da die Teamarbeit durch Dienste wie Git äußerst populär geworden ist, wird diese Methode der Template-Entwicklung bevorzugt.

Zu Beginn einer neuen Template-Extension hilft eine Typo3 Extension weiter. Diese findet sich unter dem Namen *extension\_builder*. Dieser Builder erzeugt im Backend einen neuen Menüpunkt unter dem sich im Unterpunkt Domain Modeling eine neue Template-Extension erstellen lässt. Diese taucht dann auch in der Extension-Liste auf.

Für die Arbeit mit den Dateien des erstellten Templates wird ein externer Editor benötigt. Im besten Fall sollte dieser auch mit Typoscript umgehen können, was die Wahl deutlich einschränkt. Zwei gute Optionen sind hier die IDEs UltraEdit und Webstorm.

Da der Fokus dieser Arbeit mehr auf der Umsetzung funktionaler Anforderungen liegt, wird an dieser Stelle nicht weiter auf die Möglichkeiten zum designen von Typo3 Webseiten eingegangen. Für die beispielhafte Umsetzung der Anforderungen wird ein einfaches Bootstrap Gerüst verwendet.



## 5.6 Beispielinstallation

In diesem Unterpunkt wird der tatsächliche Installationsprozess der Testumgebung, inklusive einem Blick auf den Code, dokumentiert. Da sich diese Installationsanleitung sehr praktisch gestaltet, sollten die Versionen der genutzten Software beachtet werden, da hier schnell Änderungen auftreten können. Die Testumgebung läuft mit Typo3 7.6 LTS und Wamp 3.0.0 mit PHP 5.6. Wichtig hierbei ist vor Allem die Typo3 Version. Es wird mit Bootstrap Version 3 gearbeitet.

### Typo 3 Installationsschritte

1. Typo3 7.6 LTS herunterladen.
2. Wamp oder vergleichbares Programm herunterladen und installieren.
3. Den Typo3 Ordner in den **www**-Ordner von Wamp (htdocs bei XAMPP) verschieben
4. Nach dem Starten von Wamp zu `localhost/typo3Ordner` navigieren.
5. Es erscheint ein Installationsguide und wahrscheinlich eine Reihe von Fehlern, die zu beheben sind
6. Die `php.ini` öffnen
  - Die Variable `memory_limit` erhöhen auf 64 oder höher
  - Die Variable `upload_max_filesize` auf mindestens 10MB erhöhen
  - Die Variable `max_execution_time` auf 240 setzen
7. Die OpenSSL Extension muss als Systemvariable gesetzt werden. Unter Systemvariablen die Variable mit dem Namen `OPENSSL_CONF` und dem Wert des Pfades zur `openssl.cnf` angeben. Danach eventuell Computer neu starten.
8. Die PHP Extension Fileinfo geht nicht. In der `php.ini` das Semikolon vor `extension=php_fileinfo.dll` entfernen.
9. Windows Apache Thread Stack Size Fehler. Dieser kann in der `httpd.conf` Datei angepasst werden. Hier folgenden Codeschnipsel ans Ende der Datei kopieren:  
`<IfModule mpm_winnt_module> ThreadStackSize 8388608 </IfModule>`
10. Alle Dienste von Wamp neu starten und zum Installationstool von Typo3 zurückkehren. Im nächsten Schritt wird eine Datenbank benötigt. Will man diese selbst anlegen hilft das Tool `phpmyadmin` unter `localhost/phpmyadmin`. Hier lässt sich mit einem Klick eine leere Datenbank anlegen.
11. Typo3 verlangt beim Anlegen der Datenbank auch den Port. Dieser lässt sich in der Konsole mit dem Befehl `netstat -a -o` auslesen. Hierzu vergleicht man die angezeigte PID mit dem Wamp Prozess im Taskmanager. Für gewöhnlich ist der Port eine Zahl um die 3000.
12. Nun lässt sich das Backend, durch das Anhängen von `/typo3` in der Adresszeile, aufrufen.

## 5 Typo3 - Eine Übersicht

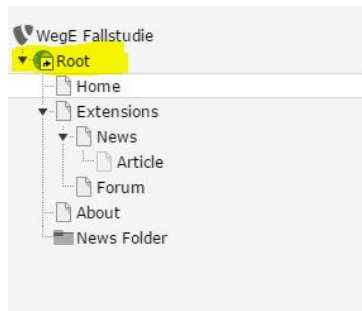


Abbildung 5.5 — Aufbau Website-Baum

**Template-Erstellung Schritte** Der Vorteil eines CMS ist die Erstellung von dynamischem Content. Um das zu bewerkstelligen braucht es ein Template zum Anzeigen dieser Inhalte. Für die Testumgebung wurde ein rudimentäres Template basierend auf dem Bootstrap Framework installiert. Das Bootstrap Framework sorgt ohne viel Arbeit für ein besseres Aussehen der Webseite und Responsivität, d.h. eine automatische Anpassung an alle Bildschirmgrößen.

1. Bootstrap 3 herunterladen.
2. Bootstrap Starter Template herunterladen durch das Kopieren des Quellcodes (F12). Diese vorerst als `index.html` abspeichern.
3. In der `index.html` alle href-Links an die neue Umgebung anpassen. Der vollständige Code findet sich im Anhang.
4. Einen neuen Ordner im Verzeichnis fileadmin anlegen. Dieser hält das Template.
5. Hier die `index.html` und den bootstrap Ordner ablegen. Dazu drei neue Ordner anlegen mit den Namen `layouts`, `partials` und `ts`.
6. Im Backend unter dem Menüpunkt Template muss das Bootstrap Template im Root der Webseite angelegt werden. Sollte dieser nicht existieren, kann dieser unter **Page** angelegt werden.
7. Im Root-Template unter **Includes** müssen zwei Dinge eingefügt werden, `css_styled_content` und `fluid_styled_content`. Die Reihenfolge ist dabei wichtig. Sollte `fluid_styled_content` nicht existieren, muss dieses als Extension installiert werden.
8. Unter **General** werden nun zwei externe Typoscript Dateien für die Konstanten und das Setup verlinkt. So erhält man mehr Flexibilität bei der Arbeit mit Versionierungen und Teamarbeit.

Unter **Constants** wird Folgendes geschrieben: `<INCLUDE_TYPOSCRIPT: source=FF-ILE:fileadmin/Template_WegE/ts/constants.ts`■

Unter **Setup**: `<INCLUDE_TYPOSCRIPT: source= FF-ILE:fileadmin/Template_WegE/ts/setup.ts`■

Die zwei verlinkten Dateien müssen nun auch im ts-Ordner erstellt werden.

## 5 Typo3 - Eine Übersicht

9. Die setup.ts benötigt nun einiges an Code, der aus der index.html übertragen wird. Diese enthält dann nur noch den Code zwischen den body-tags. Die fertige Datei ist im Anhang zu finden.
10. Die Teile aus der index.html, die nun wegfallen, werden in der setup.ts angelegt. Dafür ist ein PAGE Objekt nötig, welches alle referenzierten Dateien, wie bootstrap.css, einbindet. Die komplette Datei ist im Anhang zu finden. Die Inhalte, die das Template der Testumgebung dynamisch erstellt werden sollen beschränken sich auf das Menü und den Seitentitel. Nachdem diese Datei angelegt ist, wird das Menü automatisch mit den erstellten Seiten aus dem Typo3 Backend gefüllt.

Nach dem Anlegen eines rudimentären, aber voll funktionsfähigem Template, kann mit dem Erstellen von Content, beziehungsweise dem Installieren und Testen von Extensions, begonnen werden.

## 6 Anpassung eines Typo3 Systems an die gesammelten Requirements

Nach dem Erstellen einer simplen Typo3 Umgebung ist das Ziel dieses Teils die Untersuchung verschiedener Extensions, die zur Umsetzung der WegE Requirements genutzt werden können. Eine passende Extension erfüllt dabei klar die Anforderungen, ist effizient in der Nutzung und Wartung und lässt sich möglichst leicht implementieren.

### 6.1 Rechtemanagement und Authentikation

Die Erstellung und das Management der Usergruppen, die für das WegE Projekt geplant sind, könnten sich als eine der schwierigsten Aufgaben entpuppen. Zum einen gibt es eine Menge Redakteure, für viele verschiedene Sektionen der Webseite. Dann soll es die Möglichkeit geben sich mittels Kennnummer der Universität einzuloggen und zum Schluss soll jeder interessierte User sich einen Account anlegen können, um zum Beispiel im Forum mitzuwirken. Daraus resultiert ein kompliziertes Rechtemanagement.

Wichtig hierbei die Unterscheidung von Backend Usern und Frontend Usern. Das Erstellen von Backend Usern lässt sich bequem über das Backend von Typo3 erledigen. Ein Administrator legt dafür einfach einen neuen Nutzer an und bestimmt die jeweiligen Rechte dieses Nutzers. Das werden zumeist Redakteure sein und da sich die Zahl dieser in Grenzen halten sollte, ist es möglich diese alle manuell anzulegen.

#### 6.1.1 Möglichkeiten

Braucht eine Webseite nur Administratoren und einige Redakteure, also nur Backend User, so lässt sich das nur mit Typo3 allein bewerkstelligen. Dafür legt ein Administrator per Hand im Backend manuell neue Zugänge an. Das funktioniert nicht mit Frontend Usern. Die momentan vorherrschende Methode zur Verwaltung von Frontend Usern ist die Extension `sr_feuser_registration`. Sie befindet sich in den meist heruntergeladenen Extensions, wurde bereits 2005 das erste mal veröffentlicht und wird in Version 4.0 angeboten.

#### 6.1.2 Implementation

Die Implementation dieser Extension verläuft, in Berücksichtigung der Größe des Vorhabens, sehr schnell ab.

## 6 Anpassung eines Typo3 Systems an die gesammelten Requirements

WegE Fallstudie Home Extensions About

**Please complete the information for your membership.**

Please complete all required fields \*

Preferred username \*

Password \*

Repeat Password \*

Title: ☒ None ☐ Mr ☐ Ms

First name \*

Last name \*

Status:

Date of birth (dd-mm-yyyy)

Title

Company

Address

City

County/province

Country

Postal code

Telephone

Abbildung 6.1 — Das erzeugte Registrierungsformular

### 6.1.3 Evaluation

Die Arbeit mit der `sr_feuser_registration` Extension stellte sich als sehr angenehm heraus. Das mag mitunter daran liegen, dass sie sich seit über 10 Jahren in Entwicklung befindet und in einem sehr ausgereiften Stadium befindet. Die Extension bietet so gut wie jede erdenkliche Funktion rund um das Thema Frontend User als Menüpunkt im Backend, sodass man schnell und ohne Anleitung ein funktionierendes Registrierungsformular aufsetzen kann. Es nutzt die bereits vorhandenen Funktionen von Typo3, womit es sich leicht definieren lässt, welche Seiten nur für registrierte User gedacht sind. Sind Frontend User auf einer Typo3 Seite nötig ist diese Extension eine optimale Lösung. Darüber hinaus gibt es viele Extensions, welche die Funktion von `sr_feuser_registration` erweitern.

## 6.2 Blog und News

Die Anzeige von News oder einem fortlaufenden Blog findet auf immer mehr Webseiten Relevanz. So ist es wenig verwunderlich, dass die bekannteste News-Extension von Typo3 auch eine der meistverwendeten Extensions überhaupt ist. Die populärste Wahl scheint dabei die Extension mit dem Namen `news` zu sein.

### 6.2.1 Alternativen

Eine weitere Extension, die für die Anzeige eines Blogs geeignet ist, heißt `typo3_blog`. Die Extension steht der vorgestellten `news` Extension in Nichts nach, jedoch bietet die `news` Extension zwei Dinge in einem, `news` und `Blog` Funktionalität und deckt somit zwei mögliche Anforderungen der WegE Plattform ab. Somit minimiert die Wahl der `news` Extension theoretisch Installationsaufwand.

### 6.2.2 Implementation

Zu Beginn eine kleine Warnung: Eine Extension mit dem Namen `tt_news` ist nicht mehr lauffähig und sollte nicht mit der Extension `news` verwechselt werden.

Nach dem Installieren der Extension findet sich im Menü des Typo3 Backend ein neuer Unterpunkt. Hier lassen sich neue News anlegen und verwalten. Vorerst müssen jedoch einige Seiten zur Anzeige der News angelegt werden. Im Page Menü erstellt man dazu eine Seite zum Anzeigen der Übersicht aller News. Hier werden kleine Vorschauen der News in einer Liste angezeigt. Dazu muss des Weiteren eine Unterseite zur News Übersichtsseite angelegt werden. Hier wird eine volle News angezeigt, sofern der User auf der Übersichtsseite auf den Auszug einer News klickt. Als letztes wird noch ein Ordner benötigt, in dem alle News gespeichert werden. Dieser wird im Webseitenbaum angelegt.

Beim Erstellen von Content auf der News Übersichtsseite und einem Klick auf Normal->Content, findet sich oben im Menü der Reiter Plugins. Hier taucht auch das News System auf. Dieses muss noch ein zweites Mal auf der Detailseite hinzugefügt werden.

Um diese Funktionalität der Detailseite zu bewerkstelligen müssen nach dem Erstellen dieser Seiten im Plugin einige Angaben gemacht werden. Beim Editieren des Plugins auf der News Übersichtsseite unter dem Reiter Plugin muss die List view ausgewählt sein. Diese erzeugt die gewünschte Übersicht aller News. Im Unterreiter Additional wird angegeben welche Seite für die Detailansicht verantwortlich sein soll. Im Plugin der Detailseite ist für die Detailansicht Detail view zu wählen und eine Referenz auf die Übersichtsseite zu setzen.

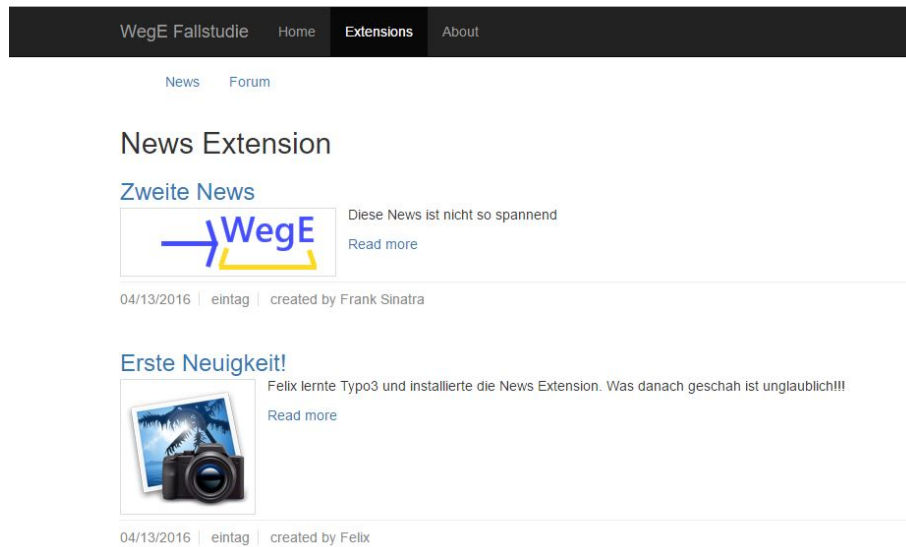
Unter dem News Menüpunkt lassen sich nun neue Artikel mit Tags, Bildern, Kategorien, Autor etc. erstellen, die bereits, wie gewollt, angezeigt werden.

Die Extension kommt mit einem eigenen Design daher, was für eine seriöse Anwendung natürlich angepasst werden müsste. Für diese Testumgebung bleibt es jedoch beim Standard-Design.

### 6.2.3 Evaluation

Die `news` Extension bietet alle Funktionalitäten, die man für die Anzeige von News jemals brauchen wird. Die Implementation geht schnell, wenn auch nicht unbedingt beim ersten Mal. Alles in Allem ist diese Extension eine exzellente Wahl.

## 6 Anpassung eines Typo3 Systems an die gesammelten Requirements



**Abbildung 6.2** — Das Ergebnis der news Extension

### 6.3 Forum

Die Anforderung an ein Forum ist mit Sicherheit eines der aufwändigsten Teile dieses Projekts. Das bedeutet auch, dass eine hochwertige Extension hier eine große Zeitersparnis bedeuten kann. Ein Forum soll bestimmten Nutzern erlauben neue Themen anzulegen. Diese Themen werden in Listen angeordnet und das Klicken eines Themas führt zu einer Unterseite, in der User dieses Thema diskutieren können. Der Content besteht dabei nicht nur aus Text, sondern auch aus Bildern und eventuell aus Dateianhängen. Dazu soll es Funktionalität geben, all diesen Content editieren und löschen zu können und dabei diese Funktionalität sinnvoll nur bestimmten Nutzern zu geben. So soll zum Beispiel nur der Ersteller ein Thema wieder löschen können. Erreicht ein Forum eine bestimmte Größe ist es wünschenswert eine weitere Usergruppe definieren zu können, die ein Forum moderieren und somit die Rechte besitzen jegliche Anpassungen vorzunehmen.

#### 6.3.1 Möglichkeiten

Es gibt eigentlich nur eine Extension, die für Typo3 7.6 in Frage kommt und die heißt `typo3_forum`. Sollte diese aus irgendwelchen Gründen als Möglichkeit ausscheiden kann man auf ein externes Forum Plugin zurückgreifen, das auf einer niedrigeren Ebene integriert wird. Ein etabliertes Beispiel heißt `phpBB` und bietet genau diese Funktionalität. Tatsächlich bietet `phpBB` mehr Anpassungsmöglichkeiten als die `typo3_forum` Extension, ist aber auch mit etwas mehr Arbeit verbunden, da man aus der Typo3 Umgebung ausbricht.

## 6 Anpassung eines Typo3 Systems an die gesammelten Requirements



**Abbildung 6.3** — Das Ergebnis der typo3\_forum Extension

### 6.3.2 Implementation

Die Implementation der grundlegendsten Forum-Funktionalität fällt moderat aus. Nach der Installation wird ein Sysfolder benötigt und ein spezieller Forum-Record angelegt. In diesem lassen sich die Rechte für jegliche Funktionen genau anpassen. Beispielsweise welche Usergruppen in der Lage sind Antworten zu posten. Außerdem wird das Root Template mit der Extension erweitert. Zuletzt kommt man nicht drum herum etwas Typoscript zu schreiben. Es handelt sich hierbei jedoch nur um einen kleinen Schnipsel, der die Orte zur Anzeige des Forums und des Speichers über die PID definieren. Zusätzlich ist zu beachten, dass MySQL nicht im strict mode sein darf. Das lässt sich in der my.ini abschalten.

### 6.3.3 Evaluation

Grundsätzlich bietet dieses Forum alle Funktionen, die für ein vernünftiges WegE Forum nötig sind. Leider existieren zu diesem Zeitpunkt noch einige schwer nachvollziehbare Bugs. Ein Beispiel hierfür sind die Buttons zum Erstellen eines neuen Forumeintrages, die nicht immer angezeigt werden, wenn sie es sollten. Diese lassen sich durch individuelle Anpassungen soweit vermeiden, das bedeutet wiederum mehr Aufwand. Da diese Extension jedoch noch sehr neu ist und von einem professionellen Team betreut wird, ist es gut möglich, dass sich dieser Fakt bereits in wenigen Monaten ändert.



## **6.4 Anbindung an bestehende Systeme wie Opus**

### **6.4.1 Möglichkeiten**

### **6.4.2 Implementation**

### **6.4.3 Evaluation**

## **6.5 Teilen von wissenschaftlichen Arbeiten**

### **6.5.1 Möglichkeiten**

### **6.5.2 Implementation**

### **6.5.3 Evaluation**

## **6.6 Weitere nützliche Extensions**

- RealURL - hübsche urls
- MetaSEO Enhancements - page seo
- Grid Elements - Spalten für den WYSIWYG
- Mask - Eigene Contentelemente machen

### **YAG - Yet another Gallery**

Will man auf einer Webseite sehr viele Bilder anzeigen, ist es ratsam diese ein wenig zu strukturieren. Das geschieht meist durch die Aufteilung in Galerien und Alben. Dieses sollen dem Betrachter natürlich auch ansprechend präsentiert werden. Beide dieser Anforderungen werden von der YAG Extension erfüllt. Die Installation ist sehr einfach, zu beachten ist jedoch, dass eine nahtlose Integration nicht trivial ist und einige Anpassungen vorgenommen werden um eine ansprechende Ausgabe zu bekommen.

### **Grid Elements**

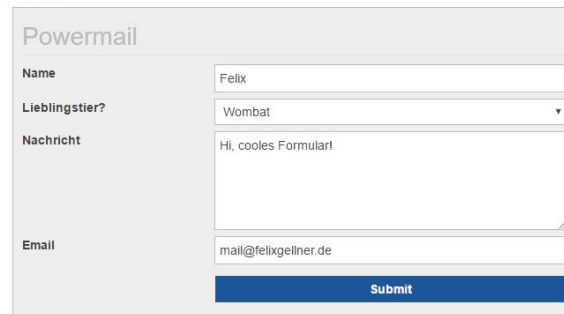
### **Powermail**

Eine sehr einfach klassifizierbare Extension ist Powermail. Sie erfüllt den Zweck ein Formular anzuzeigen, durch das ein Nutzer der Seite eine Nachricht senden kann ohne in ein externes Mailingprogramm zu wechseln. Dies findet im WegE Projekt als Anlaufstelle eine Verwendung. Die Installation gestaltet sich sehr einfach und alle Aspekte eines solchen Formulars lassen sich be-

## 6 Anpassung eines Typo3 Systems an die gesammelten Requirements

Dieses Plugin funktioniert sendet natürlich keine Mails auf einer lokalen Umgebung.

Mail



The screenshot shows a web form titled "Mail" with a "Powermail" header. The form contains four input fields: "Name" with the value "Felix", "Lieblingstier?" with a dropdown menu showing "Wombat", "Nachricht" with the text "Hi, cooles Formular!", and "Email" with the value "mail@felixgellner.de". A blue "Submit" button is located at the bottom right of the form.

**Abbildung 6.4** — Ein Formular erstellt mit Powermail

quem über das Backend anpassen. Das Formular wird als Plugin in die Seite integriert, bekommt ein erstelltes Formular zugewiesen, erhält einen Include im Template und schon ist es betriebsfähig. Die einzige etwas speziellere Anforderung an diese Extension, die das WegE Projekt haben könnte, wäre die Möglichkeit unterschiedlicher Empfängeradressen in unterschiedlichen Formularen. Auch das ist kein Problem. Die Wahl auf Powermail fiel aufgrund der Popularität und Stabilität. Andere ähnliche Extensions unterscheiden sich jedoch kaum.

### 6.7 Wartung und Sicherheit

## **7 Analyse des Mehrwerts der WegE Plattform**

### **7.1 Sichtung anderer Projekte der Qualitätsoffensive Lehrerbildung**

### **7.2 Alternativen und Verbesserungsvorschläge**

## **8 Zusammenfassung**

### **8.1 Erkenntnisse**

Die Weltherrschaft zu erringen, ist nicht so einfach, wie viele sich das vorstellen...

### **8.2 Meine Meinung**

# Abbildungsverzeichnis

1.1	WegE Logo . . . . .	1
1.2	Die Bestandteile des WegE Projekts . . . . .	2
4.1	SharePoint Logo . . . . .	13
4.2	LifeRay Logo . . . . .	13
4.3	Typo3 Logo . . . . .	14
5.1	Das Frontend gleicht einer normalen Website. . . . .	16
5.2	Das Typo3 Backend . . . . .	17
5.3	Typo3 Backend Menü . . . . .	17
5.4	Typo3 Template Teile . . . . .	21
5.5	Aufbau Website-Baum . . . . .	23
6.1	sr_feuser_register Extension . . . . .	26
6.2	News Extension . . . . .	28
6.3	News Extension . . . . .	29
6.4	Powermail Extension . . . . .	31

## **Tabellenverzeichnis**

# Literaturverzeichnis

- [PR10] PRESSMAN, Roger S. ; ROGER, Pressman: *Software Engineering: A Practitioner's Approach* -. 7. Aufl. New York : McGraw-Hill Education, 2010. – ISBN 978-0-073-37597-7
- [Rue10] RUEPING, Andreas: *Where Code and Content Meet - Design Patterns for Web Content Management and Delivery, Personalisation and User Participation*. 1. New York : John Wiley & Sons, 2010. – ISBN 978-0-470-97094-2
- [typ] *Typo3 Documentation*. [www.docs.typo3.org](http://www.docs.typo3.org). – Accessed June 12 2016
- [Wag16] WAGNER, Wolfgang: *Typo3 CMS 7 - Das umfassende Training*. feb 2016. – DVD Training
- [weg] *Wege: Wegweisende Lehrerbildung*. [www.uni-bamberg.de/wege](http://www.uni-bamberg.de/wege). – Accessed June 12 2016

Ich erkläre hiermit gemäß § 27 Abs. 2 APO, dass ich die vorstehende Diplomarbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Bamberg, 31.05.2005