



Thema:

Entwicklung eines webbasierten Tools zur Projektunterstützung

Studienarbeit

Arbeitsgruppe Wirtschaftsinformatik

Themensteller: Frank Hahnemann

Betreuer: Prof. Dr. Hans-Knud Arndt

Vorgelegt von: Frank Hahnemann

Abgabetermin: 19.10.05

Inhaltsverzeichnis

Inhaltsverzeichnis	II
Verzeichnis der Abkürzungen und Akronyme	III
Abbildungsverzeichnis	IV
Tabellenverzeichnis	V
1 Motivation.....	1
1.1 Ist-Zustand.....	1
1.2 Ziel der Arbeit	1
1.3 Aufbau der Arbeit.....	2
2 Problemdefinition - Anforderungen.....	3
2.1 Funktionale Anforderungen	3
2.2 Qualitative Anforderungen.....	4
2.3 Prozessbezogene Anforderungen	5
2.4 Systembezogene Anforderungen.....	5
3 Das Wiki-Konzept.....	6
3.1 Geschichte	6
3.2 Vorteile des Wiki-Konzepts	7
3.3 Nachteile und Probleme	8
3.4 Wiki-Engines im Vergleich.....	9
4 Umsetzung	10
4.1 Auswahl eines Wikis	10
4.2 Funktionsweise des Wikis in der vorhergesehenen Systemlandschaft	11
4.3 Customizing.....	12
4.4 Neuentwicklungen.....	13
4.4.1 Dynamische Editorfunktionalität über Vorlagen	13
4.4.2 Navigation.....	15
4.5 Weitere Komponenten.....	16
4.5.1 WYSIWYG-HTML Editor	17
4.5.2 Kalender	17
4.6 Vergleich Anforderungen – Endprodukt.....	18
5 Schlusswort.....	20
A Gegenüberstellung Wiki-Syntax und HTML	21
B Möglichkeiten bei der Erstellung von Vorlagen.....	22
Literaturverzeichnis	23

Verzeichnis der Abkürzungen und Akronyme

BSD	Berkeley Software Distribution
CSS	Cascading Style Sheets
GNU	GNU is not Unix
GUI	Graphical User Interface
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
PHP	PHP: Hypertext Preprocessor
SQL	Structured Query Language
WYSIWYG	What you see is what you get

Abbildungsverzeichnis

Abb. 4.1: Integration des Wikis in die vorhandene Systemlandschaft.....	11
Abb. 4.2: Vorlagengestützte Erstellung von internen Dokumenten.....	14
Abb. 4.3: Vorlagengestützte Erstellung von Übersichten	15
Abb. 4.4: Elemente der Navigation	16
Abb. 4.5: Der WYSIWYG-Editor	17

Tabellenverzeichnis

Tab. β.1: Vergleich ausgewählter Wiki-Engines.....	9
Tab. A.1: HTML vs. WikiSyntax.....	21
Tab. B.1: Beispiele für die Erstellung von Vorlagen	22

1 Motivation

1.1 Ist-Zustand

Die Übersicht in einer Struktur zu behalten, die über einen längeren Zeitraum gewachsen ist, kann selbst für Mitarbeiter, die diese Entwicklung miterlebt haben, schwierig sein. Ist diese Struktur eine Ablagestruktur für Dokumente¹, können Informationen verloren gehen. Dies geschieht, wenn nicht festgehalten wird, wie die Dokumente untereinander verknüpft sind oder wo sie in der Ablagestruktur zu finden sind. Für Mitarbeiter, die neu hinzukommen, ist eine Eingewöhnung in dieses Gerüst schwierig. Je nach Einsatzgebiet kann die dafür nötige Einarbeitungszeit variieren. Das Wissen über das „Wo steht was“ ist in einem Projekt jedoch unabdingbar, da Zeit restriktiv ist. Die Einarbeitungszeit und die Zeit für das Auffinden von Informationen sollten aus diesem Grund so kurz wie möglich sein.

Zentral angelegte Verzeichnisstrukturen für jedes Projekt können ein Schritt in die richtige Richtung sein. Einen schnellen Überblick lassen tief und sehr stark verzweigte Verzeichnisstrukturen nicht zu. Außerdem wird die Vernetzung der Dokumente durch eine hierarchische Struktur nicht ausreichend verdeutlicht.

1.2 Ziel der Arbeit

Ablagestrukturen wachsen mit der Zeit, es kommen immer mehr Dokumente hinzu, es entstehen Verzweigungen und Verfeinerungen. Wird dieses Netz nicht kartografisiert² ist es schwierig, gezielt Informationen, beziehungsweise komplette Dokumente, aus dieser Struktur zu holen und zu verwenden. Eine Software, die die Aufgabe eines Wegweisers übernimmt und zusätzliche Informationen speichert, die zuvor nicht digital verfügbar waren, kann dem Benutzer helfen, sich in diesem Netz zurechtzufinden. Ferner kann er einen Mehrwert aus den zusätzlichen Informationen, die dort abgelegt werden können, ziehen.

Mit der Vereinfachung der Suche nach Informationen kann zum einen Zeit eingespart werden, was den eng gesteckten Terminen in einem Projekt zugute kommt und zum anderen soll diese Software es ermöglichen, die Einarbeitungszeit in ein Projekt und dessen Unterprojekte zu verkürzen.

¹ Informationsträger, die neben Text auch multimediale Elemente enthalten können

² Es wird ein Index über die vorhandenen Dokumente und den enthaltenen Informationen angelegt.

Die zu entwickelnde Software soll für jeden Mitarbeiter frei zugänglich sein und der Benutzer soll das Recht dazu haben, über diese Software Dokumente zu veröffentlichen und andere zu bearbeiten. Die Einarbeitungszeit in diese Software wird minimiert, indem allgemein bekannte Bedienoberflächen und Arbeitsabläufe verwendet werden. Dies betrifft vor allem die Navigation und das Erstellen von Dokumenten.

1.3 Aufbau der Arbeit

Im Hauptteil der Arbeit wird zunächst eine klassische Problemdefinition aufgestellt. In dieser werden funktionale, system-, prozess- und qualitätsbezogene Anforderungen festgehalten. Für das bessere Verständnis wird vor dem vierten Kapitel, bei dem es um die Umsetzung dieser Anforderungen geht, das Wiki-Konzept, seine Geschichte und populäre Vertreter dieses Konzeptes vorgestellt. Im vierten Kapitel werden dann die Neuentwicklungen und Veränderungen an vorhandener Software erläutert, die notwendig waren um die gestellten Anforderungen umzusetzen.

Im abschließenden fünften Kapitel wird auf die zukünftigen Einsatzmöglichkeiten und eventuelle Weiterentwicklungen der Software eingegangen.

2 Problemdefinition - Anforderungen

„Die Problemdefinition (...) stellt den Auftrag zur Entwicklung eines Software-Produktes dar und beinhaltet die Anforderungen an das Produkt.“, [Dumke (2001), S.17]. Sie ist die erste Phase bei einer klassischen Software Entwicklung und wird in der Praxis auch als Lastenheft- bzw. Pflichtenhefterstellung bezeichnet [vgl. Dumke (2001), S.27]. Die gestellten Ansprüche werden in funktionale, qualitative, systembezogene und prozessbezogene Anforderungen unterteilt.

Die Anforderungen sollen so genau wie möglich definiert werden. Auf diese Weise kann zum einen der Fortschritt der Entwicklung bestimmt werden, zum anderen kann so verhindert werden, dass wichtige Eigenschaften bei dem späteren Einsatz der Software fehlen, da die Beschreibung dieser eventuell zu unpräzise war.

2.1 Funktionale Anforderungen

Einer der Hauptaspekte bei der Entwicklung der Software zur Projektunterstützung ist die Umsetzung folgender Funktionalität. Die Software soll, wie in Kapitel 1.2 bereits erwähnt, ein Wegweiser für den Benutzer sein. Sie muss die vorhandene Ordnerstruktur verwenden, darf diese aber nicht verändern. Dank der Information, wo etwas in einer komplexen Ordnerstruktur steht, soll diese Software auf den Nutzer wie eine Karte wirken. Das gezielte Auffinden von Dokumenten und das Verdeutlichen der Zusammenhänge muss umgesetzt werden.

Die Zusammenhänge sollen durch Querverweise, so genannte Hyperlinks, verdeutlicht werden, welche die Navigation zu einem Dokument erlauben.

Zusätzliche Dokumente sollen über einen Editor erstellt, gespeichert und verändert werden können. Mit Hilfe dieser Funktionen soll jeder autorisierte Nutzer die Gelegenheit haben, Dokumente innerhalb der Software zu verändern. So kann er beispielsweise den Inhalt eines anderen Dokumentes kurz beschreiben oder Informationen über sich selbst anderen Nutzern zugänglich machen. In Bezug auf die Erstellung neuer Dokumente in diesem System sollen der Editor und die Software mit ihrem Ablagesystem flexibel sein.

Das Erstellen dieser zusätzlichen Dokumente muss über Vorlagen vereinfacht werden. Damit soll auch erreicht werden, dass gleichartige Textdokumente eine ähnliche

Struktur haben. Zusätzliche Vorlagen sollen über ein leicht verständliches System erstellt und eingepflegt werden.

Jeder Benutzer muss die Möglichkeit haben, die neu erstellten Dokumente zu editieren. Voraussetzung dafür ist, dass der Autor eine Editierung durch Dritte zugelassen hat.

Auf Grund dieser freien Editierbarkeit müssen Änderungen des Dokumentes bis zu einem bestimmten Punkt zurückverfolgbar sein. Auch soll festgehalten werden, von wem die Änderungen vorgenommen wurden und zu jedem Zeitpunkt zurückgenommen werden können.

Um die Software vor Zugriffen von nicht autorisierten Mitarbeitern zu schützen, sollen Zugriffsrechte vorhanden sein, die sich zum einem auf den Zugang zu der Software und zum anderen auf einzelne Dokumente bis hin zu Ordnerstrukturen erstrecken.

Die Suche nach Informationen innerhalb der Software muss über eine integrierte Suchfunktion realisiert werden. Einfache logische Verknüpfungen der Suchbegriffe sollen mit der Suchmaschine möglich sein, um präzisere Suchergebnisse zu erhalten.

2.2 Qualitative Anforderungen

Eine einfache Bedienbarkeit ist zwingend erforderlich, denn eine lange Einarbeitungszeit oder eventuelle Schulung des Personals für die Benutzung der Software darf nicht auftreten. Der Benutzer soll sich selbstständig zurechtfinden und die Kernfunktionen nach kürzester Zeit benutzen können.

Im Zuge einer einfachen Bedienbarkeit ist es nötig, dass der Editor zum Erstellen und Verändern von Textdokumenten das WYSIWYG-Prinzip³ umsetzt. Versierte Benutzer sollen auch einen erweiterten Modus benutzen können, der dieses Prinzip nicht erfüllt. Dieser kann also auf einer Auszeichnungssprache basieren.

Damit die Software an zukünftig auftretende Anforderungen angepasst werden kann, muss es möglich sein, neue Funktionen hinzuzufügen oder vorhandene Funktionen zu verändern. Ob es sich dabei um Module oder elementare Funktionen handelt, kann bei der Umsetzung festgelegt werden.

³ Wenn ein Editor dem WYSIWYG-Prinzip entspricht, dann erhält der Nutzer dieses Editors nach vollendeter Arbeit das Dokument in der Form, wie er es auch beim Bearbeiten gesehen hat.

Die Software soll frei definierbare Parameter beinhalten, mit denen das Design angepasst werden kann. Infolgedessen soll die Software auch in anderen Bereichen eingesetzt werden können.

2.3 Prozessbezogene Anforderungen

Aufgrund des schnellstmöglichen Einsatzes der Software steht nur eine kurze Entwicklungszeit von unter 5 Monaten zur Verfügung. Zudem kommt noch hinzu, dass dafür nur ein Entwickler freigestellt werden kann, um die Kosten niedrig zu halten.

Diese beiden Anforderungen verlangen einen frühen Prototyp, mit dem die Funktionen getestet und verändert werden können und der stetig bis zur Einführung weiterentwickelt wird.

2.4 Systembezogene Anforderungen

Die Software muss auf jedem Rechner lauffähig sein, um eine große Anzahl Nutzer zu erreichen. Eine separate Installation ist nicht zwingend notwendig. Aus diesem Grund sollte auf Webtechnologien zurückgegriffen werden, da auf jedem Arbeitsplatz ein Webbrowser installiert ist, der auch JavaScript⁴ unterstützt.

Die kurze Entwicklungszeit, die in den prozessbezogenen Anforderungen festgesetzt wurde, erlaubt es nicht, dass der einzige Entwickler sich in eine ihm unbekannt Programmiersprache einarbeiten muss. Nach einer Absprache mit ihm wird für die Entwicklung der Software PHP, HTML und JavaScript ausgewählt.

⁴ JavaScript ist eine Programmiersprache, die von einem Großteil der vorhandenen Internetbrowsern unterstützt wird

3 Das Wiki-Konzept

Das Konzept einer webbasierten Software, in der jeder Nutzer Inhalte erstellen, veröffentlichen und verändern kann, ist nicht neu. Es gibt im Internet eine große Anzahl so genannter Wikis, die diese Idee umsetzen. Auf dieses Konzept wird im Laufe dieses Kapitels näher eingegangen, da es auch den Kern der neu entwickelten Software ausmacht.

3.1 Geschichte

Die Idee von frei editierbaren Dokumenten, zu denen mehrere Nutzer Zugang haben und der Verknüpfung dieser Dokumente über Hyperlinks, ist älter als die Idee vom Internet selbst. Unter anderem hatte Tim Berners-Lee bei der Konzeption des World Wide Web vorgesehen, dass Inhalte, die dort veröffentlicht werden, auch verändert werden können. Die Editierbarkeit von Inhalten konnte auf Grund der verwendeten Webserver⁵ eine zeitlang nicht realisiert werden. Eine Umsetzung wurde erstmals durch Ward Cunningham mit der Entwicklung des ersten Wiki Wiki Webs erreicht.

Das Verbinden von Dokumenten über Hyperlinks wurde bereits 1945 von Vannevar Bush analysiert. „The human mind does not work that way. It operates by association. With one item in its grasp, it snaps instantly to the next that is suggested by the association of thoughts, in accordance with some intricate web of trails carried by the cells of the brain. (...) Yet the speed of action, the intricacy of trails, the detail of mental pictures, is awe-inspiring beyond all else in nature.“, [Bush (1945) S. 6]. Mit dieser Aussage stellte er den Sinn hierarchischer Anordnungen von Dokumenten in Frage. Dort erfolgt die Suche von Ebene zu Unterebene, bis das gewünschte Dokument am Ende der Suche eventuell gefunden wird und entspricht deswegen nicht der natürlichen Vorgehensweise. Bis zu einer Umsetzung seiner Idee sollten aber ein paar Jahre vergehen.

Douglas Engelbart entwickelte in einem Team das „Online-System“, welches sich aus Hard- und Software zusammensetzte und es erlaubte, Dokumente direkt zu bearbeiten, sogar im Mehrbenutzerbetrieb. Dieses System bot im Jahre 1968 Funktionalitäten, die eine lange Zeit brauchten, bis sie in heutigen Anwendungen umgesetzt und verfügbar waren.

⁵ Ein Server der über ein Netzwerk erreichbar ist und Anfragen über HTTP verarbeiten kann.

Anfang der 90er wurde der erste Webbrowser WorldWideWeb⁶ von Tim Berners-Lee entwickelt. Dieser ermöglichte einen Zugang zu dem von ihm entwickelten World Wide Web (kurz WWW).

Als Erfinder des Wiki Wiki Web gilt Ward Cunningham. Er entwickelte Anfang 1995 das erste Wiki, das bis heute in Benutzung ist. Dies war auch das erste System, das all die zuvor aufgeführten Ideen umsetzen konnte und über das WWW mehreren Benutzern zugänglich gemacht wurde.

„Wiki wiki“ kommt aus dem Hawaiianischen und bedeutet „schnell“ und schnell können in einem Wiki Wiki Web (kurz Wiki) neue Dokumente mit Inhalt gefüllt und ohne Umwege veröffentlicht werden. Eine lange Zeit fristete dieses Konzept ein Nischendasein im WWW [vgl. Möller (2003a)]. Dank einer aktiven Gemeinschaft, die viele Dokumente in solchen Systemen veröffentlicht, vorhandene Dokumente bearbeitet und erweitert, gewannen diese Wikis an Umfang und Bedeutung. Auch populäre Suchmaschinen wie z. B. Google verweisen mittlerweile auf Dokumente in solchen Wikis. Das bekannteste Wiki dürfte die freie Enzyklopädie Wikipedia sein, ein Projekt der Wikimedia Foundation. Dort gibt es mittlerweile über 1,9 Millionen Artikel (Stand September 2005) in verschiedenen Sprachen, davon allein über 250.000 deutschsprachige Artikel. Diese Gemeinschaft erfreut sich eines enormen Zuwachses und hat bis jetzt über 50.000 registrierte Benutzer weltweit. Andere Wikis der Wikimedia Foundation erlauben freien Zugriff auf Wörterbücher, allgemeine Texte, verschiedene Multimedia Dateien, Neuigkeiten und ein Projekt zur Schaffung freier Lernmaterialien.

3.2 Vorteile des Wiki-Konzepts

Der größte Vorteil eines Wikis ist die freie Editierbarkeit von Dokumenten. Jeder hat die Möglichkeit seine eigenen Ideen zu einem Thema dem Dokument hinzuzufügen, Fehler zu beseitigen oder neue Dokumente zu erstellen. Infolgedessen entsteht aus der Zusammenarbeit von mehreren Benutzern ein Konsens zu einem Thema, da die Aussage in diesem so entstandenen Dokument nicht nur die Meinung einer Person repräsentiert.

Steht hinter dem Wiki eine aktive Gemeinschaft von Nutzern werden Fehler, Falschaussagen und böswillige Löschungen von Textpassagen oder kompletten Dokumenten schnell korrigiert. Denn die meisten Wiki-Engines⁷ besitzen eine

⁶ Später „Nexus“ um Verwechslungen zu vermeiden

⁷ Ein System das die Funktionalität eines Wikis zur Verfügung stellt und auf einem Webserver installiert wird.

Änderungsliste zu jedem Dokument, womit Veränderungen leicht rückgängig gemacht werden können.

Dank der Vernetzung der Dokumente durch Hyperlinks entsteht ein Netz und keine hierarchische Anordnung. So kommt der Nutzer des Wikis von einem Begriff in einem Dokument schnell zu einem Dokument, das sich mit diesem Begriff beschäftigt. Existiert zu einem Begriff, der eine Erklärung benötigt, kein eigenes Dokument, kann dieses sehr schnell angelegt werden.

Das Formatieren von Dokumenten über eine einfache Syntax ist eine Erleichterung bei der Erstellung und Bearbeitung von Dokumenten. Sie beschränkt sich auf die wenigen notwendigen Funktionen um einen Text zu formatieren und ist aus diesem Grund sehr viel einfacher anzuwenden als HTML – wofür eine längere Einarbeitungszeit notwendig ist. Eine solche Syntax wird in vielen Wiki-Engines verwendet, ein Standard existiert jedoch nicht. Ein tabellarischer Vergleich zwischen der Wiki-Syntax in Wikipedia und HTML, ist im Anhang A dieser Arbeit zu finden.

3.3 Nachteile und Probleme

Der Vorteil der Wikis ist auch einer der größten Nachteile und Kritikpunkte. Wie zuvor angesprochen müssen die Dokumente nicht korrekt sein. So können Fehler enthalten sein, sowohl bössartig als auch unbeabsichtigt eingefügt. Da keine redaktionelle Überprüfung erfolgt, bleiben solche Fehler bis jemand diese beseitigt hat. Eine Referenz auf ein Dokument innerhalb eines Wikis sollte aus diesem Grund kritisch betrachtet werden. Es gibt eine handvoll Regeln, die beachtet werden können, eine Garantie auf Korrektheit kann damit aber nicht gegeben werden. Allgemein wird die Glaubwürdigkeit der Dokumente in einem Wiki in Frage gestellt, da der Autor meistens unbekannt ist. So kann irgendjemand sie verfasst haben, ohne z. B. den wissenschaftlichen Hintergrund zu haben oder das ganze Dokument basiert absichtlich auf Falschaussagen. Umfangreiche Recherchen, um eine Aussage in einem Artikel oder Dokument zu bestätigen, bleiben da nicht aus.

Ein weiteres Problem sind die Benutzer des Wikis. Es benötigt insbesondere am Anfang sehr viel Arbeit, um das System mit verwertbaren Informationen zu füllen und so eine Akzeptanz bei anderen Nutzern zu erreichen. Ist die Gemeinschaft, die diese Wiki nutzt passiv, funktioniert das ganze System nicht, da ohne Veränderungen ein Wiki einer statischen Webseite entspricht. Es setzt somit auf die Zuarbeit seiner Nutzer, die mit und an den Dokumenten arbeiten.

Kritisiert wird aber auch die Wiki-Syntax, die erlernt werden muss und auf Grund ihrer Einfachheit nur begrenzte Möglichkeiten in der Gestaltung eines Dokumentes zulässt. Außerdem sind bis dato die WYSIWYG-Editoren zum Erstellen von Dokumenten nur in wenigen Wiki-Engines vorgesehen. Dieser Punkt wird jedoch kontrovers diskutiert und von aktiven Nutzern eines Wikis als Vorteil aufgefasst. Es lässt sich aber nicht bestreiten, dass die Hürde für neue Nutzer eigene Dokumente zu erstellen mittels eines einfach zu bedienenden Editors herabgesetzt werden kann, da das Erlernen einer speziellen Syntax wegfällt.

3.4 Wiki-Engines im Vergleich

Nachdem Ward Cunningham das erste Wiki entwickelt hatte und dafür die Programmiersprache Perl verwendete, folgten Adaptionen in andere Sprachen. Mit jeder Erweiterung oder Neuumsetzung des Wiki-Konzepts stieg auch die bereitgestellte Funktionalität. Mittlerweile gibt es über 100 verschiedene Wiki-Engines.

Tab. 3.1: Vergleich ausgewählter Wiki-Engines

	MediaWiki	PMWiki	TWiki	MoinMoin
Programmiersprache	PHP	PHP	Perl	Python
Datenspeicherung	Datenbank	Flaches Dateisystem ⁸	Flaches Dateisystem	Flaches Dateisystem
Lizenz	GPL ⁹	GPL	GPL	GPL
Suchmaschine	Inhalt und Titel	Inhalt	Inhalt, Titel, Metadaten und Backlinks	Inhalt, Titel, Metadaten und Backlinks
Internationalisierung	Ja	Ja	Ja	Ja
Erweiterbar/Anpassbar	Ja	Ja	Ja	Ja
GUI-Editor	Ja	Ja	Ja	Nein

Quelle: Vgl. DokuWiki (2005)

Anhand der Tabelle 3.1 kann festgestellt werden, dass sich die Wiki-Engines nicht maßgeblich unterscheiden. Der größte Unterschied ist bei der verwendeten Programmiersprache zu finden. Fehlende Funktionalitäten können in vielen Fällen über die vorhandenen Schnittstellen integriert werden.

⁸ Dokumente werden als Textdateien in einem oder mehreren Ordnern abgelegt

⁹ GNU General Public License - Lizenz für freie Software, die es dem Nutzer ermöglicht, diese für jeden Zweck zu benutzen, frei zu kopieren, zu verändern, die veränderte Software wieder frei zu verbreiten. Genaueres dazu ist unter <http://www.fsf.org/> zu finden.

4 Umsetzung

Die gestellten Anforderungen an das Endprodukt und die kurze Zeit, die für die Erstellung der Software vorgesehen wurde, erlauben keinen umfangreichen und zeitintensiven Entstehungsprozess. Da eine Eigenentwicklung eines solchen Wikis sehr langwierig sein kann und nicht abschätzbare Risiken mit sich bringt, wurde eine fertige Wiki-Engine verwendet und den gestellten Anforderungen angepasst. Aus diesem Grund war innerhalb kürzester Zeit ein Prototyp fertig gestellt. Die Basisfunktionalität war dank der Wiki-Engine sofort gegeben.

4.1 Auswahl eines Wikis

Die Auswahl wurde allein durch die verwendete Programmiersprache der Wiki-Engine eingeschränkt. Es wurde das in Kapitel 3.4 erwähnte PMWiki ausgewählt. Dieses Wiki bietet von Anfang an eine große Funktionalität und ist über so genannte Rezepte¹⁰ leicht erweiterbar. Dank dieser Rezepte können grundlegende Verhaltensweisen des Wikis verändert werden. So kann die Datenhaltung, anders als in Tab. 3.1 angegeben, über eine Datenbank mittels eines entsprechenden Rezeptes realisiert werden.

Ein weiterer Grund für die Wahl des PMWikis war die einfache Strukturierung und ein großes Angebot an verschiedenen Erweiterungen. Diese können sehr einfach eingebunden und aktiviert werden. Vorteilhaft ist bei dieser Wiki-Engine die standardmäßige Verwendung eines flachen Dateisystems, was das Arbeiten mit den Dokumenten vereinfacht. Die Verwendung einer Datenbank war nicht vorgesehen, denn dafür hätten zusätzliche Ressourcen zur Verfügung gestellt werden müssen.

Das PMWiki erlaubt bei seinen Dokumenten eine sehr einfache Gruppierung. Diese erfolgt über den Dateinamen. So kann über ein Dokument, das einen Aspekt eines anderen Dokumentes genauer erklärt, durch einen Punkt zwischen den Ursprungsdokumentennamen und dem Namen des neuen Dokumentes eine Gruppierung erzeugt werden. Auf diese Weise lässt sich eine einheitliche Namensgebung für wiederkehrende Dokumente erzeugen. So sind z. B. Dokumentnamen wie ProjektA.Pflichtenheft, ProjektA.Projekthandbuch, ProjektB.Pflichtenheft usw. denkbar. Infolgedessen wird eine Strukturierung der Dokumente ermöglicht. Darüber hinaus können so auch verschiedene Kategorien angelegt werden (Projekte, Übersichten, Personal usw.).

¹⁰ Diese Rezepte sind unter www.pmwiki.org verfügbar. Diese müssen in ein dafür vorhergesehenes Verzeichnis kopiert werden und über die PMWiki Konfigurationsdatei eingebunden werden.

Weiterhin wird durch diese Wiki-Engine eine Suchfunktion zur Verfügung gestellt, die der gestellten Anforderung entspricht. Eine globale Änderungsliste, in der festgehalten wird, wer ein Dokument innerhalb des Wikis verändert hat, ist vorhanden. Außerdem existiert eine Änderungsliste zu jedem Dokument, mit der vorgenommene Änderungen verfolgt und zurückgenommen werden können.

4.2 Funktionsweise des Wikis in der vorhergesehenen Systemlandschaft

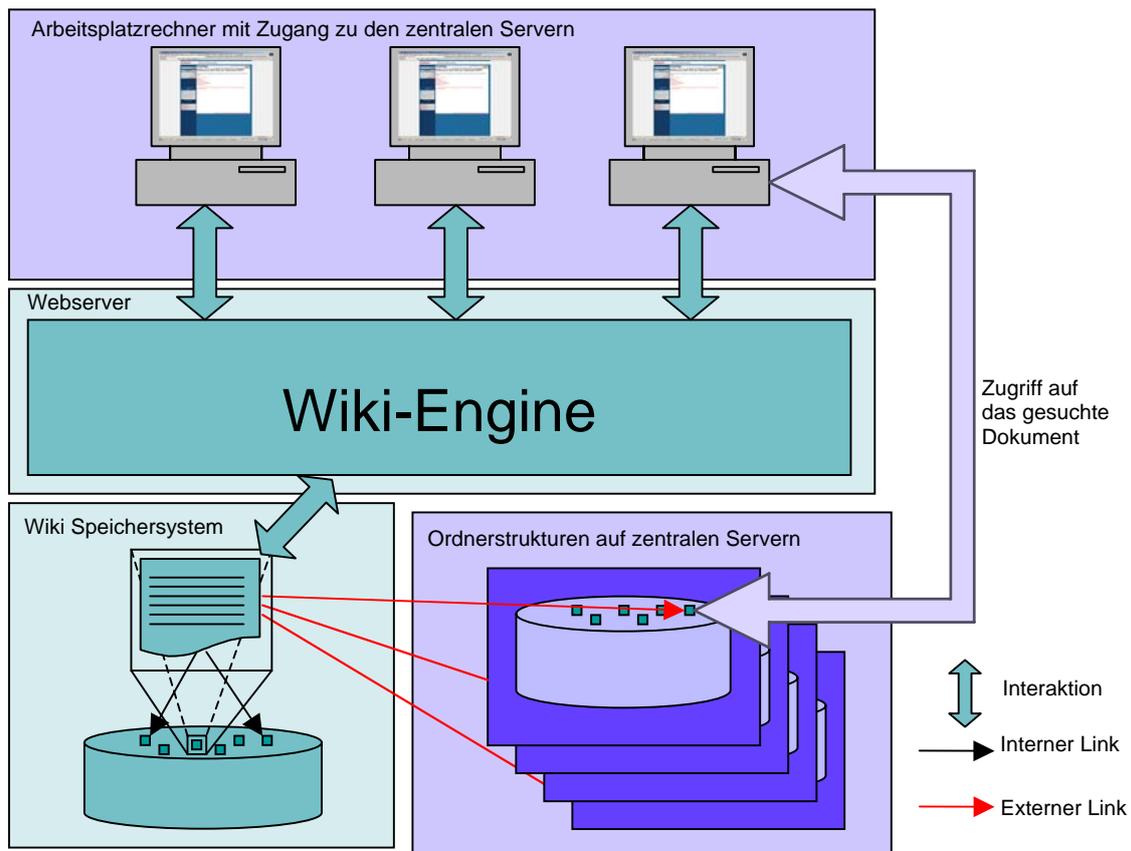


Abb. 4.1: Integration des Wikis in die vorhandene Systemlandschaft

In den funktionalen Anforderungen wurde festgelegt, dass die vorhandenen Ordnerstrukturen nicht verändert werden dürfen. Dies wird über das in Abb. 4.1 dargestellte Gerüst erreicht. Weiterhin sind dort auch die alte Struktur und der direkte Zugriff auf die Dokumente verdeutlicht. Wie schwierig sich dieser gestalten kann, wurde am Anfang der Arbeit in der Ist-Analyse aufgezeigt.

Das neu hinzugekommene Wiki legt die Dokumente, die mittels der Wiki-Engine erstellt werden können, in einem eigenen Speichersystem ab. In einem solchen Dokument existieren 2 verschiedene Arten von Hyperlinks. Da wären zum einen die

Querverweise auf andere Dokumente innerhalb des Wiki-Speichersystems (interne Dokumente) und zum anderen die Querverweise auf Dokumente, die auf anderen Speichersystemen hinterlegt sind (externe Dokumente). Genau das soll die Wegweiserfunktionalität sein, die dem Nutzer ein schnelleres Auffinden von Dokumenten auf mehreren verteilten Systemen erlaubt. Diese Schlüsselfunktion wird mit Hilfe der Wiki-Engine bereitgestellt, denn die verwendete Wiki-Syntax des PMWikis erlaubt in der Grundeinstellung das Hinterlegen von so genannten externen Links. Damit sind Querverweise gemeint, die auf Dokumente zeigen, die sich nicht im Speichersystem des Wikis befinden und können somit für externe Dokumente verwendet werden.

Ein weiterer Vorteil dieses Systems ist, dass der Benutzer nicht wissen muss, wie die Ablagestruktur im Endeffekt aussieht. Deshalb ist es sehr einfach für ihn über das Wiki zu seinem Ziel zu kommen. Das explizite Wissen, auf welchem Server und in welcher verzweigten Ordnerstruktur das Dokument abgelegt wurde, ist nicht mehr nötig. Wenn er regelmäßig auf ein bestimmtes Dokument zugreifen muss, kann er, dank der Darstellung des Wikis im Browser, einen Favoriten¹¹ anlegen, um noch schneller auf das Dokument zugreifen zu können.

Über einen Browser werden auch die internen Dokumente dargestellt. Zudem können sämtliche Funktionalitäten des Wikis von einem nicht textbasierten Browser neuester Generation dargestellt werden. Es ist nicht nötig, zusätzliche Software für die Benutzung des Wikis zu installieren. Die Seite, die der Nutzer sieht, wird von der Wiki-Engine vorbereitet und als HTML-Datei an den Browser weitergegeben. Die internen Dokumente werden, wenn der Nutzer ein solches Dokument ansehen oder bearbeiten möchte, durch die Wiki-Engine interpretiert und für die Ausgabe so vorbereitet¹², dass der Nutzer die gewünschte Aktion ausführen kann.

4.3 Customizing

Über das Customizing des PMWikis kann vor allem die Darstellung der Seiten verändert werden. So gibt es ein zentrales Verzeichnis, in dem verschiedene visuelle Grundschemas abgelegt werden können. Für das Verändern der Vorlage reichen elementare HTML- und CSS-Kenntnisse aus, um das Erscheinungsbild des Wikis komplett zu verändern und den Vorgaben anzupassen.

¹¹ Ein Favorit ist in einem Webbrowser ein Hyperlink der in einer Liste (Favoritenliste) abgelegt werden kann. Er erlaubt die schnelle Navigation zu dem Ziel dieses Hyperlinks. Moderne Webbrowser erlauben das Anlegen eines solchen Favorites.

¹² Mehr dazu im Kapitel 4.4.1

Im Laufe der Entwicklung wurde die Darstellung der angezeigten Seite durch eine Neuordnung verschiedener Elemente übersichtlicher gestaltet. Damit soll erreicht werden, dass der Nutzer ohne Anweisungen das findet, was er braucht und nicht durch die Darstellung der Seite verwirrt wird. Aus diesem Grund war auch eine Neuentwicklung der Navigation nötig, da die vorhandene sehr statisch ist. Dies ist in einem Wiki, in dem sehr viele Dokumente hinterlegt werden sollen, kein guter Ansatz.

Um den Zugang zu den internen Dokumenten einzuschränken, wurde ein vorhandener Passwortschutz aktiviert. Dieser gestattet es, einzelne Dokumente innerhalb des Wikis vor Zugriffen zu sperren. Der Lese- und Schreibschutz oder nur Letzterer kann vom Autor des Dokumentes festgelegt werden. Der generelle Zugriff auf das Wiki kann zusätzlich über einen zugangsbeschränkten Webserver, auf dem die Wiki-Engine installiert wird, eingerichtet werden. Der Zugriff auf die Dokumente, die sich außerhalb des Wikis befinden, wird über die vorhandene Systemstruktur geregelt. Kein Nutzer erhält durch das Wiki Zugang zu Dokumenten, auf die er zuvor auch nicht zugreifen konnte. Das Wiki stellt lediglich die Pfadangabe, die sich hinter so einem externen Link verbirgt, zur Verfügung.

4.4 Neuentwicklungen

Für die Umsetzung der Anforderungen, die nicht durch das ausgewählte Wiki abgedeckt werden können, muss dieses durch zusätzliche Komponenten, die die benötigten Funktionalitäten bereitstellen, erweitert werden. Während die Navigation vorhandene Einstiegspunkte in die Wiki-Engine nutzt, wurde die Ablaufsteuerung, die aktiviert wird, wenn ein Nutzer einen Text verändern möchte, an die neuen Anforderungen angepasst.

4.4.1 Dynamische Editorfunktionalität über Vorlagen

Ein wichtiger Punkt bei der Entwicklung war die Programmierung eines Editors, mit dem der Nutzer des Wikis vorlagengestützt mit Hilfe eines Formulars neue Dokumente erstellen kann. Auf diese Weise soll eine einfache Bedienung des Wikis, die unter anderem durch Anpassungen am Design der Seitendarstellung verbessert wurde, erreicht werden.

Dieser Editor funktioniert nach dem folgenden Prinzip und stellt damit dem Nutzer eine formularbasierte Seite zur Verfügung, in der er einzelne Eingabefelder einfach modifizieren kann. Möchte der Nutzer ein internes Dokument verändern, klickt er dafür

einen entsprechenden Link an. Danach überprüft die veränderte Ablaufsteuerung, um welches Dokument es sich handelt. Wurde dieses Dokument über eine Vorlage erstellt, wird dies erkannt und die Daten dementsprechend ausgelesen und für die Ausgabe auf dem Bildschirm vorbereitet. Das bedeutet auch, dass die Formularfelder der Vorlage mit den vorgegebenen Werten aus dem Dokument ausgefüllt werden. Handelt es sich um ein Dokument, das nicht mit einer Vorlage erstellt wurde, wird dieses über den WYSIWYG-Editor angezeigt und kann darüber verändert werden.

Wenn der Nutzer ein von Grund auf neues Dokument erstellt, wird ihm eine Seite präsentiert, in der er auswählen kann, welche Vorlage er für die Erstellung benutzen möchte. Je nach Auswahl wird so ein Gruppenname festgelegt, der dem Dokumentennamen automatisch vorangestellt wird. Danach kann der Nutzer die einzelnen Felder verändern und das Dokument speichern.

Bevor das interne Dokument gespeichert werden kann, muss es wiederum dafür vorbereitet werden. Alle gemachten Eingaben werden ausgewertet und in die Art umgewandelt, wie sie später für die Ausgabe notwendig ist. Dies wird wieder über die Vorlage realisiert. Im nächsten Schritt wird das so erzeugte Dokument an die Wiki-Engine weitergegeben, die dann das interne Dokument speichert. Daraus folgt außerdem, dass bei einer Veränderung der Vorlage die damit erstellten Dokumente ihre alte Form beibehalten.

Wie das Ganze aussehen kann, wird in Abb. 4.2 gezeigt. Dort ist auf der linken Hälfte der formulargestützte Editor und auf der anderen Hälfte die so erzeugte Seite zu sehen.



Abb. 4.2: Vorlagengestützte Erstellung von internen Dokumenten

Es wurde außerdem ein spezieller Editor programmiert, der den Nutzer dazu befähigt, Übersichten zu erstellen. Das Aussehen dieser lässt sich über eine Vorlage verändern, verfolgt jedoch ein anderes Prinzip. Während bei den zuvor erwähnten Vorlagen eine

HTML-Datei als solche dient, wird bei den Übersichten eine CSS-Datei verwendet. In der Abb. 4.3 lässt sich die Funktionsweise anhand eines Beispiels erkennen.

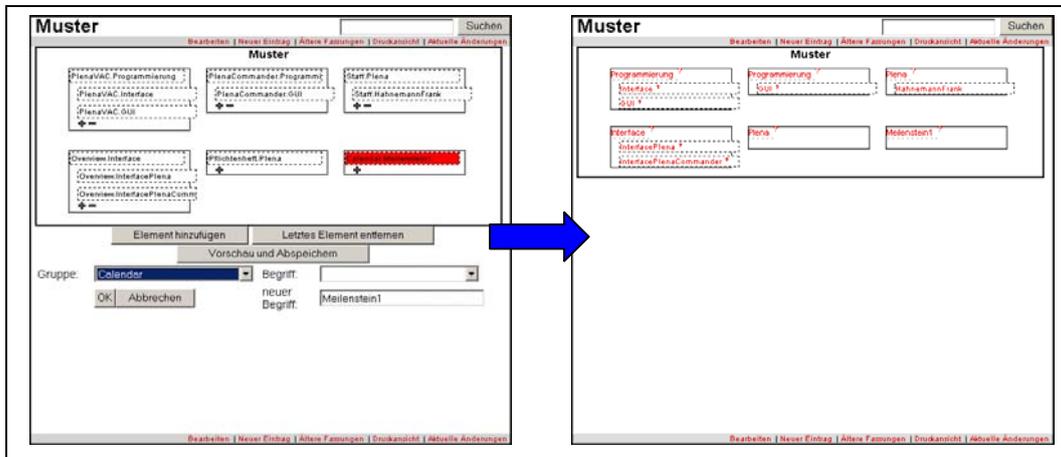


Abb. 4.3: Vorlagengestützte Erstellung von Übersichten

Innerhalb der Vorlage führen verschiedene Schlüsselwörter zu unterschiedlichen Ergebnissen bei der Anzeige. So sind unter anderem Auswahlmensüs, Textfelder und Tabellen möglich. Die Syntax erlaubt ein umfangreiches Funktionsspektrum bei der Erstellung neuer Vorlagen. Zur besseren Verdeutlichung sind einige Beispiele im Anhang B aufgeführt.

4.4.2 Navigation

Die Navigation in einem Wiki muss angesichts des sich ständig erweiternden Inhalts auch dynamisch sein. Infolgedessen wurden dort zwei permanent angezeigte Elemente in der Navigationsleiste eingeführt. Ein Element ist eine Liste mit den internen Dokumenten, die zuletzt verändert wurden. Das andere Element ist die Navigation. Dort sind alle internen Dokumente aufgeführt, auf die das aktuell angezeigte Dokument per Hyperlink verweist. Außerdem sind auch alle internen Dokumente aufgeführt, die auf das aktuell angezeigte Dokument verweisen, so genannte Backlinks. Dadurch kann der Nutzer sich komfortabel in diesem Netz von Dokumenten bewegen und gleichzeitig die Zusammenhänge zwischen diesen erkennen. In der Abb. 4.4 werden diese Elemente noch einmal veranschaulicht. Dieses Netz wird beim Laden der Seite und nach erfolgten Änderungen im Datenbestand des Wikis neu aufgebaut. Die Informationen dafür werden aus den internen Dokumenten ausgelesen und in einer Tabelle gespeichert. Dies erfolgt per PHP. Um die Navigation dynamisch zu halten und Ladezeiten beim Navigieren zu vermeiden, wird diese Tabelle an den JavaScript-Teil der Navigation

weitergegeben¹³. Dadurch wird eine schnelle Bedienung der Elemente in der Navigation ermöglicht.

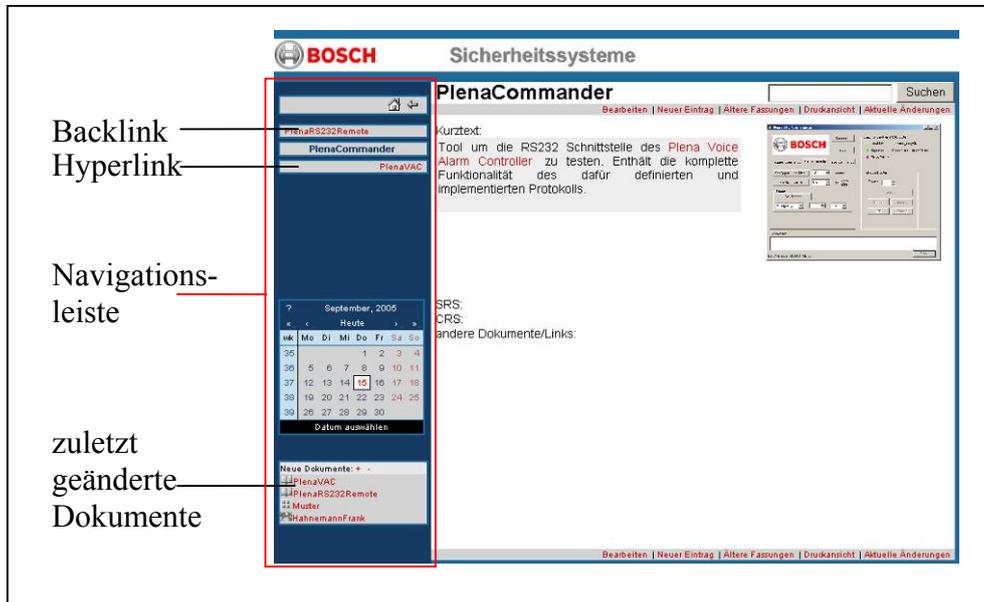


Abb. 4.4: Elemente der Navigation

Die Navigationsleiste bietet neben diesen zwei Elementen noch die Möglichkeit, auf einfache Art und Weise zusätzliche Module automatisch hinzuzufügen. Diese Module müssen mit PHP programmiert werden und in einem speziell dafür angelegten Verzeichnis gespeichert werden. Danach wird dieses Modul automatisch in der Navigationsleiste aufgenommen und kann sofort eingesetzt werden.

4.5 Weitere Komponenten

Um die qualitativen Anforderungen an den Editor umzusetzen, wurden weitere vorgefertigte Komponenten verwendet. Für den WYSIWYG-Editor musste jedoch zunächst ein PMWiki Rezept (EnableHTML) installiert werden. Zweck dieser Erweiterung ist es neben der Wiki-Syntax, HTML bei der Erstellung von internen Dokumenten zuzulassen. Das ist für den WYSIWYG-Editor und für die in Kapitel 4.4.1 vorgestellten Funktionalitäten notwendig.

¹³ Da der Webserver die PHP Befehle interpretiert und dem Browser eine fertige HTML-Seite sendet, ist es auch möglich, in dieser generierten HTML-Seite JavaScript einzubauen und so z. B. eine Tabelle, die in diesem Script verwendet wird, mit Werten zu füllen, die z. B. aus einer in PHP erstellten Tabelle stammen. Das JavaScript läuft dann nur auf dem Rechner des Nutzers, ist also unabhängig vom Webserver.

4.5.1 WYSIWYG-HTML Editor

Der HTMLArea genannte Editor, basierend auf JavaScript, konnte ohne weitere Probleme in die vorhandene Landschaft integriert werden. Nur wenige Anpassungen waren notwendig. Der Editor unterliegt der BSD-Lizenz, die der GPL ähnelt. Dadurch ist es möglich, den Quellcode und somit die Funktionalität der Software zu erweitern. So müssen mit diesem Editor nun auch interne Links erstellt werden. Dies wurde über eine zusätzliche JavaScript-Funktion realisiert. Weiterhin wurden verschiedene Funktionen, die für das Erstellen von internen Dokumenten unnötig sind, deaktiviert. Der Vorteil dieses Editors ist, dass er sich wie jedes Textbearbeitungsprogramm benutzen lässt und die typischen Bedienungselemente verwendet.

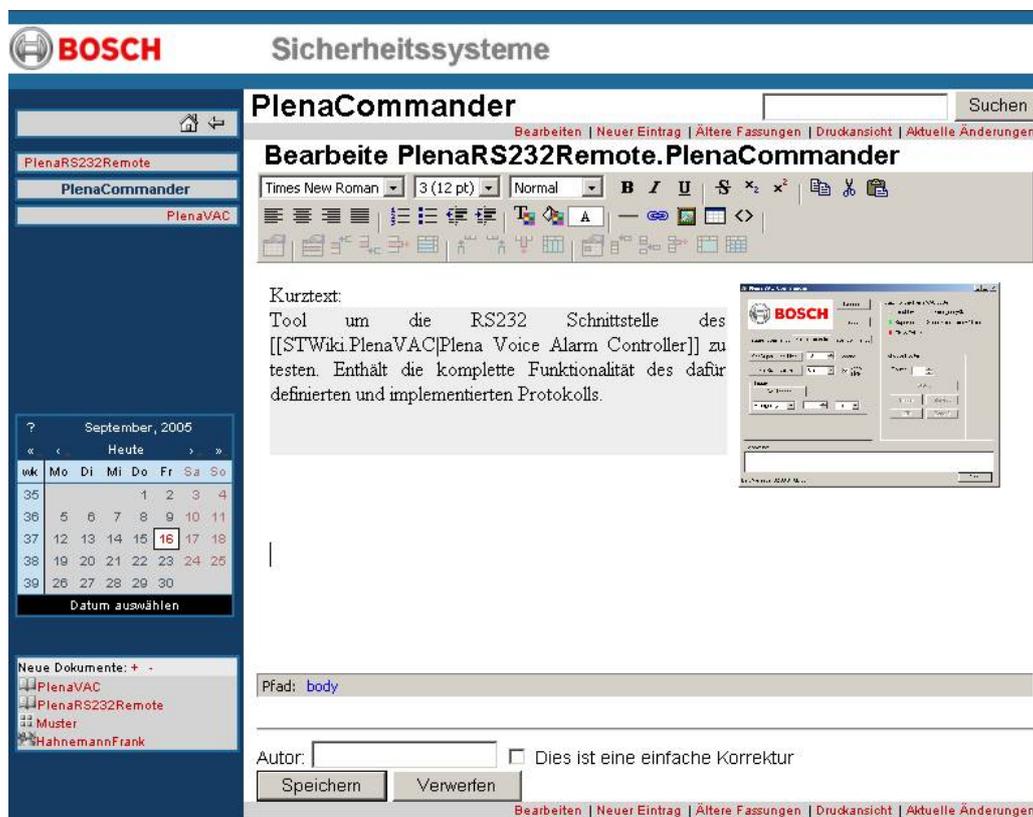


Abb. 4.5: Der WYSIWYG-Editor

4.5.2 Kalender

Das Kalendermodul, das unter anderem in der Abb. 4.5 zu sehen ist, soll verdeutlichen, wie einfach zusätzliche Module der Navigationsleiste hinzugefügt werden können und hatte zunächst einen experimentellen Status. „The DHTML Calendar“ ist eine weitere Software, die über die GPL lizenziert ist und zahlreiche Anpassungsmöglichkeiten bietet.

Dieser Kalender erlaubt es dem Benutzer, zu jedem Tag Dokumente abzulegen. Dies war nach einer kleinen Modifikation möglich. Existiert ein solches Dokument, wird der Tag in diesem Modul in einer anderen Farbe angezeigt. So eignet sich dieses Modul dafür, über diese Oberfläche auf wichtige Projekttermine hinzuweisen und gleichzeitig in dem hinterlegten Dokument auf andere interne und externe Dokumente zu verweisen. Diese können dann wiederum über einen einfachen Mausklick erreicht werden.

Die Liste, mit den zuletzt geänderten Dokumenten, wurde infolgedessen erweitert. So existiert ein zusätzlicher Teil für Kalenderdokumente innerhalb dieses Moduls. Mit dieser Erweiterung werden dem Nutzer die nächsten Termine angezeigt.

4.6 Vergleich Anforderungen – Endprodukt

Ein Großteil der funktionalen Anforderungen wurde durch die Verwendung einer vorgefertigten Lösung umgesetzt. Hier sollen diese noch einmal kurz aufgezählt werden. Das Wiki-Konzept erlaubt die Umsetzung der geforderten Wegweiserfunktionalität, die das Auffinden von Dokumenten auf verschiedenen Servern in komplexen Ordnerstrukturen erleichtert. Weiterhin können Querverweise angelegt werden. Wie das im Einzelnen aussieht ist im Kapitel 4.2 nachzulesen. Dass jeder Nutzer die internen Dokumente über einen Editor, der im Webbrowser läuft, bearbeiten kann, ist eine Hauptfunktion in einem Wiki. Genauso kann in einem Wiki eine gemachte Änderung rückgängig gemacht werden. Neben diesen Funktionen bietet jedes Wiki eine Suchfunktion, die in der Anforderungsanalyse festgelegt wurde. Die ausgewählte Wiki-Engine erlaubt auch die verlangten logischen Verknüpfungen von Suchbegriffen.

Die restlichen Anforderungen werden unter anderem durch den Funktionsumfang des PMWikis abgedeckt. Der vorhandene Editor wurde ersetzt. Das WYSIWYG-Prinzip wird durch die Integration eines fertigen Editors realisiert. Neue Dokumente können über Vorlagen mit einem speziell dafür entwickelten Editor erstellt werden. Eine einheitliche Strukturierung von wiederkehrenden Dokumenten ist möglich.

Sowohl die prozess- als auch die systembezogenen Anforderungen werden erfüllt. So wird keine andere Programmiersprache verwendet und dieses Wiki ist auf jedem Rechner mit installiertem Webbrowser lauffähig. Es muss nur auf einen vorhandenen Webserver installiert werden, der von jedem Arbeitsplatz über das Netzwerk erreichbar ist. Die Entwicklungszeit wurde eingehalten und einzelne Funktionen konnten nach zwei Wochen an einem Prototyp getestet werden, so zum Beispiel der Passwortschutz, um den Zugriff einzuschränken. Letzteres wird dadurch unterstützt, dass nicht jeder

Zugang zu jedem Server und somit zu jeder Ordnerstruktur hat. Das wird und kann durch das Wiki auch nicht umgangen werden.

Nur schwierig können die qualitativen Anforderungen getestet werden. Es ist möglich, die Oberfläche zu verändern. Verschiedene Parameter können über eine zentrale Datei verändert werden. Die verwendete Sprache kann ebenfalls über eine zentral gelegene Sprachdatei umgestellt werden. Wie gut die Bedienbarkeit dieses Wikis ist und ob das Personal, das es benutzen soll, keinerlei Schulung benötigt, muss in einem Test untersucht werden. Bei der Entwicklung wurde immer wieder darauf geachtet, dass sich das Wiki wie eine normale Webseite bedienen lässt. Nur das der Nutzer die zusätzliche Option hat, die angezeigte Seite sofort zu verändern.

Zusammenfassend kann gesagt werden, dass die fertige Software so gut wie jeden Punkt, der in den Anforderungen festgelegt wurde, erfüllen kann. Lediglich die Bedienbarkeit sollte in einem späteren Test untersucht werden.

5 Schlusswort

Mit Hilfe eines Wikis kann innerhalb kürzester Zeit eine Wissensplattform aufgebaut werden. Das Potential, was eine solche Plattform besitzt, muss jedoch durch die Nutzer umgesetzt werden, denn ein Wiki verwaltet sich nicht von selbst und legt nicht selbstständig Inhalte an. Es bedarf der Zuarbeit von außen. Es müssen dafür Leute motiviert werden, die dieses System nutzen und somit wachsen lassen. Ist diese schwierige Anfangsphase geschafft, kann ein solches System die Arbeit in einem Unternehmen und somit auch in Projekten, mit den in Kapitel 1.1 gemachten Voraussetzungen, erleichtern.

Das hier erweiterte Wiki kann dank der Anpassungsmöglichkeiten nicht nur als Tool zur Projektunterstützung eingesetzt werden. Ein Einsatz als Wissensspeicher, Kommunikationsplattform oder auch als einfache Intranet- oder Internetseite ist denkbar. Die Einsatzmöglichkeiten sind sehr breit gefächert.

Was in der endgültigen Fassung fehlt (da es auch keine Anforderung an die Software war), ist ein Editor, um Vorlagen zu erstellen. Momentan ist das Erstellen solcher nur mit Hilfe von HTML-Kenntnissen realisierbar, auch wenn das mit ein paar Beispielen nicht schwierig ist. Bei einer Portierung in andere Bereiche und Bedarf von anderen Vorlagen ist die Erstellung eines solchen Editors sinnvoll. Dieser ist auf Grund der einfachen Syntax der Vorlagen in kurzer Zeit entwickelt.

Abschließend kann gesagt werden, dass ein Wiki eine Möglichkeit ist, schnell Dokumente anzulegen, sie anderen zugänglich zu machen und letztendlich den Teamgeist fördern. Wikipedia ist dafür ein gutes Beispiel. Dort arbeiten mehrere tausend Menschen an einer freien Enzyklopädie und was als kleines Projekt begann, hat mittlerweile einen erstaunlichen Umfang erreicht.

Anhang

A Gegenüberstellung Wiki-Syntax und HTML

Für diesen Vergleich wird die Syntax von Wikipedia verwendet, da es sich dabei um das bekannteste Wiki weltweit handelt.

Tab. A.1: HTML vs. WikiSyntax

	HTML	Wikipedia
Textformatierung		
Überschrift	<code><h1>Text</h1></code> <code><h2>Text</h2></code>	<code>==Text==</code> <code>===Text===</code>
Aufzählung	<code></code> <code>Punkt 1</code> <code>Punkt 2</code> <code>Punkt 3</code> <code></code>	<code>* Punkt 1</code> <code>* Punkt 2</code> <code>* Punkt 3</code>
Fett	<code>fett</code>	<code>'''fett'''</code>
Kursiv	<code><i>kursiv</i></code>	<code>''kursiv''</code>
Links (bei den ersten 3 Varianten wird link.html als Ziel angenommen)		
Variante 1	<code></code> Link <code></code>	<code>[[Link]]</code>
Variante 2	<code></code> Linkskurve <code></code>	<code>[[Link]]skurve</code>
Variante 3	<code></code> Alternativtext <code></code>	<code>[[Link Alternativtext]]</code>
Variante 4	<code></code> <code>http://wikipedia.org</code>	<code>http://wikipedia.org</code>
Variante 5	<code></code> klick <code></code>	<code>[http://wikipedia.org klick]</code>
Verschiedenes		
Bilder	<code></code> <code></code>	<code>[[Bild:beispiel.png]]</code>
Tabellen (im Beispiel wird eine Tabelle mit 2 Spalten und 2 Zeilen erzeugt)	<code><table><tr></code> <code><td>eins</td><td>zwei</td></code> <code></tr><tr></code> <code><td>drei</td><td>vier</td></code> <code></table></code>	<code>{ </code> <code> eins zwei</code> <code> -</code> <code> drei vier</code> <code> }</code>

Quelle: Vgl. Wikipedia (2005a); Wikipedia (2005b); Wikipedia (2005c)

B Möglichkeiten bei der Erstellung von Vorlagen

Die Vorlagen sind HTML-Dateien, die in einem speziellen Ordner abgelegt werden. In der nachfolgenden Tabelle werden jeweils ein Codebeispiel und das damit erzielte Ergebnis präsentiert. Innerhalb eines HTML-Elementes ist ein Bezeichner (`id`) enthalten. Dieser ist notwendig, damit der Editor das entsprechende Formularfeld vor dem schließenden Teil des HTML-Elements zeichnet. Damit können die Daten, die in dem Dokument gespeichert werden, bei einer erneuten Bearbeitung innerhalb einer solchen Vorlage, wieder gefunden werden.

Tab. B.1: Beispiele für die Erstellung von Vorlagen

<pre><tr><td>Name</td> <td id="name"></td></tr> <tr><td>Vorname</td> <td id="vorname"></td></tr></pre>	
<pre>Kurztext:
 <div id="kurztext" id2="textarea/cols='43'/rows='10'"></pre>	
<pre><tr><td>Projekt</td></tr> <td id="projekt0" <select name="projekt0"> <option value="Projekt."></option> </select></td></pre>	

Neben dem notwendigen Bezeichner zum Erzwingen von Eingabefeldern, kann noch ein Zweiter angegeben werden, um eine HTML-Textarea zu initiieren. Dies ist im zweiten Beispiel zu sehen (`id2="textarea/cols='43'/rows='10'"`). Im dritten Beispiel wird die Möglichkeit demonstriert, eine HTML-ChoiceBox zu erzeugen. Dafür muss in dem HTML-Element `<option>` das Attribut `value` mit einem Wert gefüllt werden. Dieser kann ein Feld oder auch ein Teil eines Dateinamens sein. Die Werte werden automatisch eingetragen. Im letzten Beispiel wird außerdem eine Liste erzeugt, in der zusätzliche Eingabefelder über einen entsprechenden Knopf hinzugefügt und entfernt werden können. Diese Liste wird automatisch zur Verfügung gestellt, wenn in dem Bezeichner eines HTML-Elements am Ende eine null steht (z. B. `projekt0`).

Allgemein ist dieser Editor und die Möglichkeiten zur Erstellung von Vorlagen sehr variabel. Die aufgeführten Beispiele geben nur einen Teil der möglichen Fälle wieder.

Literaturverzeichnis

Bush, Vannevar (1945): The Atlantic Monthly; Juli 1945; As We May Think; Volume 176, No. 1; 101-108

DokuWiki (2005): wiki:compare[DokuWiki]. <http://wiki.splitbrain.org/wiki:compare>. 29. September 2005.

Dumke, Reiner (2001): Software Engineering - Eine Einführung für Informatiker und Ingenieure: Systeme, Erfahrungen, Methoden, Tools. 3. Aufl., Vieweg Verlag Braunschweig/Wiesbaden, 2001

Möller, Erik (2003a): Das Wiki-Prinzip. <http://www.heise.de/bin/tp/issue/r4/dl-artikel2.cgi?mode=html&artikelnr=14736>. 29. September 2005.

Möller, Erik (2003b): Alle gegen Brockhaus <http://www.heise.de/bin/tp/issue/r4/dl-artikel2.cgi?mode=html&artikelnr=14802>. 29. September 2005.

Möller, Erik (2003c): Diderots Traumtagebuch. <http://www.heise.de/bin/tp/issue/r4/dl-artikel2.cgi?mode=html&artikelnr=14855>. 29. September 2005.

Möller, Erik (2003d): Diesen Artikel bearbeiten. <http://www.heise.de/bin/tp/issue/r4/dl-artikel2.cgi?mode=html&artikelnr=14903>. 29. September 2005.

PHP (2005): PHP:Hypertext Preprocessor. <http://www.php.net/>. 29. September 2005.

Theis, Thomas (2000): PHP4 – Webserver-Programmierung für Einsteiger. Galileo Computing Bonn, 2000

Wikipedia (2005a): Hilfe:Textgestaltung. <http://de.wikipedia.org/wiki/Hilfe:Textgestaltung>. 29. September 2005

Wikipedia (2005b): Hilfe:Links. <http://de.wikipedia.org/wiki/Hilfe:Links>. 29. September 2005

Wikipedia (2005c): Hilfe:Tabellen. <http://de.wikipedia.org/wiki/Hilfe:Tabellen>. 29. September 2005.

Abschließende Erklärung

Ich versichere hiermit, dass ich die vorliegende Studienarbeit selbstständig, ohne unzulässige 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.

Unterhaching, den 19. Oktober 2005