

WikiWikiWeb
Proseminar Website-Management-Systeme

Markus Müller
m_muell@informatik.uni-kl.de

29. April 2004

Zusammenfassung

In der heutigen Zeit sind manche Websites zu einer solchen Größe herangewachsen, dass eine manuelle Verwaltung schwer bzw. gar nicht mehr möglich ist. Was liegt da näher als eine solche Website mit einem Softwaresystem zu verwalten? Das WikiWikiWeb ist ein solches Softwaresystem, wenn es auch einen etwas ungewöhnlichen Ansatz verfolgt.

Inhaltsverzeichnis

1	Begriffsklärungen	
2	Einleitung	
3	Das Konzept des WikiWikiWeb	
3.1	Hintergrund	2
3.2	Informationsfluss	2
3.3	Innere Struktur	2
4	„Killer-Features“	3
4.1	Versions-Management	3
4.2	Pflege & Wartung	3
4.3	Instant Publishing	3
4.4	Linkkonsistenz	3
5	Fehlende Features	4
5.1	Content-Life-Cycle	4
5.2	Statische Realisierung	4
6	Funktioniert das Konzept?	4
6.1	Benutzerverwaltung	4
6.2	Community	5
7	Implementierungen	5
7.1	Verbreitung einzelner Wikis	5
7.2	Programmiersprachen	5
7.3	Betriebssysteme	5
7.4	Datenspeicherung	6
7.5	Links	6
7.6	HeyHeyWickie	6
7.6.1	Merkmale	6
7.6.2	Verarbeitung einer Anfrage	6
8	Bewertung	6
8.1	Bewertung anhand des Schemas vom Vortrag „Grundlagen CMS“	6
8.1.1	Workflow	6
8.1.2	Freigabeverfahren	6
8.1.3	Versionskontrolle	7
8.1.4	Unterstützte Datenbanken	7
8.1.5	Betriebssystem	7
8.1.6	Suchmaschinen	7
8.2	Eigene Bewertung	7
8.2.1	Vorteile	7
8.2.2	Nachteile	7
9	Weiterführende Literatur	7

1 Begriffsklärungen

1	Seite Eine „Seite“ ist eine einzelne Webseite innerhalb eines Webangebots.
1	Website Eine „Website“ ist eine Ansammlung von Webseiten die zusammen ein Webangebot bilden.
2	
2	Wiki Die Begriffe „Wiki“, „WikiWikiWeb“, „Wiki-Engine“ sind im Rahmen dieses Proseminars synonym verwendet. „Wiki“ ist eine Kurzform von „WikiWikiWeb“ und „Wiki-Engine“ bezeichnet die einem Wiki zugrunde liegende Programmlogik. Da eine Wiki-Engine überwiegend nur in einem Wiki eingesetzt wird, kann man in vereinfachender Weise ein Wiki mit der zugehörigen Engine identifizieren.

2 Einleitung

4	Das World Wide Web hat sich in den letzten Jahren relativ stark verändert. So werden auf heutigen Websites Inhalte bereit gestellt, die sich wesentlich von denen unterscheiden, die noch vor ein paar Jahren dort anzutreffen waren. Hauptsächlich geändert haben sich dabei die Präsentation des Inhalts wie auch die Menge der Informationen.
5	War es früher die Aufgabe des Webmasters, sich um das Erstellen von Seiten zu kümmern, ist diese Tätigkeit heute bei größeren Websites von einer Person aufgrund des Umfangs an Seiten nicht mehr realisierbar.
6	Auch werden heute teilweise besondere Ansprüche an das Design und Layout von Seiten einer Webpräsenz gestellt, so dass sich diese Anforderungen ohne geeignete Softwaresysteme nicht mehr praktikabel erfüllen lassen.
6	An diesem Punkt treten spezielle Softwaresysteme zum Verwalten von Websites in Erscheinung, die sogenannten Website-Management-Systeme. Aufgabe dieser Systeme ist es, die nötigen Arbeiten zum Erstellen und Pflegen von Seiten einer Webpräsenz so weit wie möglich zu automatisieren und die restliche Arbeit möglichst sinnvoll auf eine oder mehrere Arbeitskräfte aufzuteilen. Die Aufgabe des Webmasters, der sich ursprünglich selbst um die komplette Verwaltung der Site gekümmert hat, tritt damit mehr und

mehr in den Hintergrund und beschränkt sich heute fast ausschließlich auf die technischen Aspekte wie z.B. Wartung des Webserver und des Content-Management-Systems, mit welchem die Autoren, Grafiker, Designer die einzelnen Seiten letztendlich erstellen.

3 Das Konzept des WikiWiki-Web

3.1 Hintergrund

Ein solches Softwaresystem zur Erstellung und Verwaltung von Inhalten einer Website ist das WikiWikiWeb. Die Idee für ein solches Konzept zur Verwaltung geht zurück auf Ward Cunningham. Dieser hatte in den späten 80er Jahren erstmals die Vorstellung von einem System nach diesem Schema.

Er wollte ein System schaffen, das einfach und schnell zu bedienen ist und eine geringe Einarbeitungszeit benötigt. Aus diesem Grund heißt das WikiWikiWeb auch WikiWikiWeb: „wiki“ ist hawaiisch und bedeutet soviel wie „schnell“. Ferner wollte Cunningham bewusst auf eine hierarchische Struktur verzichten. Er bevorzugte einen „flachen“ Namensraum für alle Seiten innerhalb des Systems. Ein weiterer wichtiger Punkt für ihn war die Offenheit. Sprich: jeder sollte prinzipiell dazu in der Lage sein, Seiten zu verändern. Außerdem sollten diese Änderungen genau protokolliert werden, damit die Entstehungsgeschichte einer Seite erhalten bleibt.

Bei WikiWikiWeb steht die Verwaltung von textueller Information im Vordergrund.

3.2 Informationsfluss

In einem gängigen¹ Content-Management-System ist es in erster Linie so, dass ein Autor oder aber ein Team von Autoren Inhalte bereitstellt. Diese stehen dann wiederum einer Leserschaft zur Verfügung. Wie der Name „Leserschaft“ schon andeutet, können diese Personen nur lesend auf die Artikel der Redaktion zugreifen. Die einzige Interaktionsmöglichkeiten seitens der Leserschaft gibt es

¹Generell ist es schwer, von einem „gängigen“, „normalen“ oder aber „herkömmlichen“ System zu sprechen, einfach aufgrund der Tatsache, dass es so etwas nicht gibt. In diesem Zusammenhang meine ich mit „gängigen“ Systemen Systeme, die in diesem Proseminar behandelt wurden.

eventuell in Form eines Forums, in dem über Artikel diskutiert werden kann oder per E-Mail. Eine direkte Möglichkeit des Lesers, aktiv die bereitgestellten Inhalte zu bearbeiten, gibt es nicht.

Bei dem WikiWikiWeb ist dieser Sachverhalt grundlegend anders. Denn im Gegensatz zu einem herkömmlichen System unterscheidet das WikiWikiWeb zunächst einmal nicht zwischen Autor und Leser. Aus der Sicht des Wiki gibt es nur einen Typ von Anwender, den Benutzer.

Dies bedeutet, dass die Besucher einer Website selbst Inhalte erstellen, bereits vorhandene erweitern und fehlerhafte korrigieren können. Insgesamt wird damit die Seite „lebendiger“ weil es nicht mehr nur eine Einweg-Kommunikation von der Redaktion zu den Lesern gibt, sondern auch die Leser untereinander Informationen bereitstellen können. Konsequenterweise kann man jetzt auch nicht mehr von *einer* Redaktion und damit von Autoren reden. Denn Autor ist im Prinzip jeder.

3.3 Innere Struktur

Wenn man sich verschiedene Content-Management-Systeme ansieht, so organisieren sie die Seiten, die sie verwalten in einer hierarchischen Struktur. Dies bedeutet, dass es in der Regel eine Hauptseite gibt, die sich in der Hierarchie an oberster Stelle befindet bzw. wenn man sich die Struktur als einen Baum mit Knoten vorstellt, dann stellt die Hauptseite den Wurzelknoten dar. Dieser Wurzelknoten wiederum besitzt weitere Unterknoten, die ihrerseits wieder Unterknoten haben können. Den Schluß der Kette bilden die einzelnen Seiten, welche bei dem Baum die Blätter darstellen.

Innerhalb eines Wikis findet man eine grundlegend andere Struktur vor. Es existiert keine baumartige Anordnung der einzelnen Seiten. Das WikiWikiWeb ist im wahrsten Sinne des Wortes ein „Web“: Die Seiten bilden ein Netz, welches, je mehr Seiten es gibt, immer dichter wird. Es gibt also nur eine Ebene im Gegensatz zu den anderen Systemen.

Diese Tatsache wirkt sich insbesondere auf den Namensraum und die Semantik des Seitentitels aus. Wenn man sich in einem gängigen CMS² z.B. eine Seite mit dem Titel „Inhalt“ ansieht, so muss man

²Content-Management-System

unterscheiden, auf welcher Ebene man sich gerade befindet. „Inhalt“ auf der Hauptseite hat eine völlig andere Bedeutung wie „Inhalt“ auf einer untergeordneten Ebene. Obwohl beide Seiten gleich heißen, so erhält man bei dem „Inhalt“ in einer untergeordneten Ebene eben nur den Inhalt der jeweiligen Ebene und nicht, wie beim „Inhalt“ auf der Hauptseite, den Inhalt aller Ebenen. Trotz des vermeintlich gleichen Titels haben beide Seiten eine völlig andere Bedeutung. In einem Wiki gibt es, bedingt durch die innere Struktur, nur eine Ebene im Namensraum und von daher besitzt ein Titel auch nur eine einzige Bedeutung. Dies war eines der Ziele von Cunningham bei der Entwicklung.

Dadurch aber, dass es keine festgelegte Struktur in der Anordnung der Seiten innerhalb eines Wikis gibt, existiert auch nicht notwendigerweise eine spezielle Startseite. Jede Seite eines Wikis kann stattdessen eine Startseite sein. In der Praxis realisiert man den Einstieg in ein Wiki aber dennoch über eine spezielle Startseite, in der man eine kurze Einleitung und Überblick zu dem Thema gibt, mit dem sich das Wiki beschäftigt. Dies ist aber lediglich eine Konvention und das Wiki selbst stellt dafür keine besondere Funktionalität bereit.

4 „Killer-Features“

4.1 Versions-Management

Das Versions-Management bei WikiWikiWeb gehört mit zu dem besten, was unter CM-Systemen zu finden ist.

Es war eine der Designentscheidungen von Ward Cunningham, dass die Entstehungsgeschichte einer Seite nachvollziehbar sein soll. Da das speichern *jeder* Änderung einer Seite seit dem Zeitpunkt der Erstellung extrem viel Ressourcen beanspruchen würde, geht man in der Regel dazu über, entweder die Änderungen in einem gewissen Zeitrahmen, etwa alle Veränderungen des letzten Monats, oder aber nur eine bestimmte Anzahl an Änderungen zu speichern.

Anhand dieses Archivs ist es möglich, mehr oder weniger detailliert die letzten Änderungen einer Seite zu verfolgen.

4.2 Pflege & Wartung

Bei dem WikiWikiWeb verteilen sich die Kosten für Pflege und Wartung signifikant anders als bei anderen Content-Management-Systemen. Zum Einen sind die Leser in der Lage, die Informationen ständig anzupassen. Dadurch werden nicht nur eventuelle Fehler korrigiert, sondern auch die Aktualität der bereitgestellten Informationen ist größer. Zum Anderen können die Benutzer fehlende Informationen relativ einfach ergänzen.

Dadurch spart der Betreiber nicht nur die Kosten für die Pflege, sondern auch die Kosten für die Erstellung von neuen Inhalten.

Die einzigen Kosten, die ein Betreiber selbst übernehmen muss sind die Kosten für die Wartung des Systems an sich auf der technischen Ebene, sprich die Serveradministration und eventuelle Kosten für die benötigte Bandbreite.

4.3 Instant Publishing

Wie bereits im Unterpunkt „Pflege & Wartung“ angesprochen, können Benutzer selbst ohne große Schwierigkeiten Inhalte selbst veröffentlichen. Eines der Ziele von W. Cunningham ist es, dass jeder das System ohne größere Vorkenntnisse benutzen kann.

Dadurch, dass es relativ einfach ist, eigene Inhalte zu veröffentlichen, und das ganze System dynamisch realisiert ist, werden Änderungen und neue Seiten sofort verfügbar.

4.4 Linkkonsistenz

Gerade bei WikiWikiWeb ist dieses Thema besonders interessant, weil es im Prinzip keine toten Links gibt. Wird auf eine Seite innerhalb des Wikis ein Link gesetzt und diese Seite existiert nicht, dann erstellt das Wiki in der Regel automatisch eine entsprechende Seite und bittet den Benutzer, auf dieser Seite den entsprechenden Inhalt zu hinterlegen.

Wenn man das ganze von einer anderen Perspektive her sieht, dann bedeutet dies, dass man zuerst einen Link setzen muss, wenn man eine neue Seite erstellen möchte. Ohne den Link, kann die Seite nicht erreicht werden und es können folglich auch keine Inhalte zu diesem Thema hinterlegt werden.

Dieses Vorgehen steht im krassen Gegensatz zur Vorgehensweise bei anderen CM-Systemen. Dort ist es üblich, zuerst neue Inhalte zu erstellen und erst dann die entsprechenden Links zu setzen, bzw. das CMS setzt automatisch Links, sobald eine neue Seite erstellt wurde.

5 Fehlende Features

5.1 Content-Life-Cycle

Andere Content-Management-Systeme bieten teilweise einen recht feingranularen Content-Life-Cycle. Dies bedeutet, dass wenn ein Autor einen Artikel erstellt hat, er diesen an einen Kontrolleur zur Durchsicht schicken muss. Dieser begutachtet den Artikel und kann ihn entweder veröffentlichen oder zurück zum Autor schicken mit der Bitte, ihn noch einmal zu überarbeiten. Ferner bieten manche CM-Systeme die Möglichkeit, ältere Artikel automatisch zu archivieren.

Im Gegensatz dazu bietet das WikiWikiWeb nur einen stark verkürzten Content-Life-Cycle. Eines der Prinzipien des Wikis ist das „Instant Publishing“. Dies bedeutet, dass alle Artikel, die erstellt oder verändert werden, sofort veröffentlicht werden, ohne dass sie zuerst von einem Kontrolleur begutachtet werden müssen.

Eine Archivierung würde bei WikiWikiWeb nur bedingt Sinn machen. Eines der Ziele ist es, das „Web“ aus verschiedenen Seiten immer feinmaschiger zu machen. Würden jetzt einzelne Seiten archiviert werden, so würden Löcher im Netz entstehen, und damit wäre eine solche Funktionalität nicht im Sinne des Erfinders.

5.2 Statische Realisierung

Ein Wiki lässt sich nur sinnvoll dynamisch realisieren, weil sich die Inhalte relativ schnell ändern können. Eine statische Implementierung würde von daher bedeuten, dass eine relativ große Anzahl von Seiten bei jeder Änderung neu erstellt und auf einen Webserver geladen werden müssten. Dies ist aus drei Gründen nicht sehr effizient. Erstens müssen *alle* Seiten neu generiert werden, die sich ändern, gleichgültig ob sie jemals von jemandem angesehen werden. Zweitens erzeugt der Dateitransfer selbst Kosten, die man ursprünglich durch die sta-

tische Realisierung einsparen wollte. Drittens gibt es durch diesen Ansatz eine mehr oder weniger große Verzögerung, bis die aktualisierten Inhalte zur Verfügung stehen.

Daher ist eine dynamische Realisierung auf jeden Fall einer statischen vorzuziehen, auch wenn die dynamische vielleicht aus Sicht des Webservers zunächst als die teurere Alternative erscheint.

6 Funktioniert das Konzept?

6.1 Benutzerverwaltung

Während andere Content-Management-Systeme eine teilweise recht feingranulare Rechteverwaltung für die Benutzer bereitstellen, verfolgt das WikiWikiWeb auch hier einen komplett anderen Ansatz. In einem herkömmlichen CMS werden die Benutzerrechte in der Regel mittels spezieller Gruppen- und/oder Rollenverteilung verwaltet. Eine ähnliche Verwaltungsstruktur fehlt dem WikiWikiWeb fast vollständig. Von der ursprünglichen Idee her gibt es praktisch keine restriktive Benutzerverwaltung. Dies bedeutet, dass man sich zwar als Benutzer durchaus bei einem Wiki anmelden kann, dies aber jedoch nicht zwingend erforderlich ist, um Änderungen an den Seiten selbst vorzunehmen. Einziger Vorteil einer Anmeldung ist, dass sich ein Benutzer einen bestimmten Benutzernamen registrieren lassen kann, welcher dann im Changelog³ einer Seite angezeigt wird.

Mittlerweile gibt es aber auch Implementierungen, die es nur zuvor angemeldeten Benutzern erlauben, Änderungen durchzuführen. Dies entspricht aber nicht mehr der ursprünglichen Idee Cunninghams.

Man kann sich jetzt natürlich die berechtigte Frage stellen, ob ein solch offener Ansatz nicht dazu führen könnte, dass Cracker⁴ oder Skript-Kiddies⁵ eine solche Website unbrauchbar machen, indem sie gezielt falsche Informationen verbreiten.

Allerdings verfolgen solche Personen meistens das Ziel, sich in ihrem Umfeld durch „große“ Leis-

³Ein „Changelog“ protokolliert, wie der Name schon sagt, alle Änderungen.

⁴Ein „Cracker“ dringt in fremde Systeme ein mit dem Ziel andere zu schädigen.

⁵„Skript-Kiddies“ tun im Prinzip das gleiche wie Cracker, jedoch tun sie dies ohne eigene geistige Leistung, sie benutzen Programme oder Skripte anderer.

tungen Anerkennung zu verschaffen, wie z.B. das Eindringen in fremde Rechner. Da in einem Wiki aber jeder ohne große Vorkenntnisse, was ja eines der Designprinzipien war, Seiten verändern kann, stellt eine solche Aktion keine „große“ Leistung dar und somit ist die Motivation bei dem entsprechenden Personenkreis eher gering.

Auf der anderen Seite existiert die Möglichkeit mittels des Versions-Management die letzten Änderungen einer Seite zu verfolgen und somit sind eventuelle Schäden an einzelnen Seiten leicht reparierbar.

Insgesamt macht es somit wesentlich mehr Aufwand, Seiten gezielt zu verfälschen als die verfälschten Seiten wieder zu berichtigen.

6.2 Community

Ein ganz zentraler Punkt ist die Community, denn ohne sie würde ein Wiki nicht funktionieren. Es „lebt“ gewissermaßen davon, dass es Leute gibt, die sich die Seiten ansehen und auch selbst Inhalte ergänzen.

Dies wird vor allem durch das offene Konzept gefördert. Je leichter es für den Benutzer ist, sich selbst einzubringen, umso eher wird er es auch letztendlich tun.

7 Implementierungen

Ward Cunningham hatte nicht nur die Idee für das WikiWikiWeb, sondern er hat seine Idee auch in die Realität umgesetzt und selbst ein Wiki geschrieben. Dieses „Ur“-Wiki findet heute kaum noch Verbreitung. Man findet es heute nur noch bei Cunningham selbst, auf der Website des „Portland Pattern Repository“⁶.

Heute gibt es eine Unmenge von verschiedenen Implementierungen mit teilweise stark unterschiedlicher Funktionalität.

7.1 Verbreitung einzelner Wikis

Die am weitesten verbreitetsten Wikis sind gegenwärtig laut c't Ausgabe 25/2003 das „MediaWiki“ mit fast 69 Prozent, gefolgt von „UseModeWiki“ (16 Prozent). Weitere Wikis mit signifikanter

⁶<http://c2.com/ppr/>

Verbreitung sind „MoinMoin“, „PhpWiki“, „TWiki“.

c't hat auch einige Messungen⁷ bzgl. der Performance einiger Wikis durchgeführt. Relativ schwach hat dabei das „UseModeWiki“ bei der Volltextsuche abgeschnitten, welche bei ihm mit 45 Sekunden mehr als viermal so lange dauert als bei „MoinMoin“. Schneller war dagegen „PhpWiki“, das nicht einmal drei Sekunden dafür benötigte.

Alle Performancemessungen wurden bei Wikis durchgeführt, die nicht auf eine externe Datenbank bei der Seitenspeicherung zurückgreifen, sondern die Datenverwaltung in Eigenregie vornehmen. Ein Benchmark von Wikis mit DB-Anbindung ist insofern nicht sinnvoll, als dass man hier in erster Linie eine Performancebetrachtung der Datenbank und nicht des Wikis machen würde.

7.2 Programmiersprachen

Es gibt Implementierungen für alle⁸ Programmiersprachen. Darunter finden sich auch solch exotische wie beispielsweise „Awk“, „Bash“, „CommonLISP“ oder „Ruby“.

7.3 Betriebssysteme

Für nahezu alle Betriebssysteme gibt es eine oder mehrere Implementierungen des WikiWikiWebs. Egal ob es sich um Unix-Derivate wie „NetBSD“, „FreeBSD“, „OpenBSD“, „Mac OS X“ oder „Linux“ handelt: in den meisten Fällen gibt es einen passenden Interpreter der jeweiligen Sprache in der ein Wiki geschrieben ist. Auch für proprietäre⁹ Systeme wie „Windows“ gibt es Implementierungen.

Während fast alle Wikis einen Webserver voraussetzen, gibt es auch einige Implementierungen, welche ohne Webserver auskommen und komplett eigenständig lauffähig sind.

Dies ist vor allem für Betriebssysteme wie Windows interessant, da sie in der Regel ohne Webserver ausgeliefert werden.

⁷Gemessen wurde die Dauer einer Volltextsuche auf etwa 10 000 Seiten mit insgesamt 80 Megabyte

⁸„Alle“ meint hier „fast alle“. Auch wenn es eine unüberschaubar grosse Anzahl von Wikis für viele Programmiersprachen gibt, so habe ich keine Informationen darüber, ob es wirklich für *jede* Sprache eine Implementierung gibt.

⁹entspricht nicht offenen Standards

7.4 Datenspeicherung

Es existieren unterschiedliche Ansätze zur Speicherung des Content innerhalb eines Wikis. Die Wikis lassen sich grob in zwei Kategorien einordnen: Zum Einen gibt es Wikis, welche zur Datenspeicherung auf eine externe Datenbank zurückgreifen, zum Anderen gibt es Wikis, welche sich um die Verwaltung komplett selbst kümmern und den Content in Dateien innerhalb eines Dateisystems ablegen.

7.5 Links

Ein weiteres Unterscheidungsmerkmal von Wikis ist die Art und Weise der Erstellung von Links. Hier gibt es keinen einheitlichen Standard.

In der Implementierung Cunninghams wurden Links dadurch erzeugt, dass man das betreffende Wort, welches den Link repräsentieren sollte, im „CamelCase“ geschrieben hat. Dies bedeutet Großschreibung in der Mitte des Worts.

Da solche Wörter jedoch recht schlecht in Suchmaschinen indiziert werden können und im Allgemeinen schlechter lesbar sind, werden bei aktuelleren Wikis die Links mittels speziellen wikispezifischen Metazeichen erzeugt.

7.6 HeyHeyWickie

Als Beispiel für eine Implementierung von WikiWikiWeb habe ich mir das HeyHeyWickie ausgesucht. Wegen seiner Überschaubarkeit ist es ideal, um einige grundlegende Eigenschaften im Rahmen dieses Proseminars zu erörtern.

7.6.1 Merkmale

Das HeyHeyWickie zeichnet sich dadurch aus, dass es zum Einen nur 3919 Bytes groß ist, zum Anderen bietet es eine gute Schnittstelle für eigene Erweiterungen.

Es ist in Python geschrieben und benutzt sowohl den Template Prozessor EmPy als auch Docutils für die Konvertierung der einzelnen Seiten nach HTML für die Darstellung im Webbrowser.

7.6.2 Verarbeitung einer Anfrage

Wenn eine Seite angefordert wird, dann sendet der Browser eine entsprechende Anfrage an den

Webserver, der diese mittels CGI¹⁰ an die Wiki-Engine weiterleitet. Je nachdem, welche Seite angefordert wurde, lädt das Wiki jetzt die entsprechende Datei aus dem Dateisystem und beginnt damit, den Inhalt entsprechend aufzubereiten.

Als erstes beginnt der Template Prozessor EmPy mit seiner Arbeit. Er führt zunächst den in die Datei eingebetteten Python-Code aus, sofern vorhanden. Anschließend wird der Datenstrom von EmPy nach ReStructuredText konvertiert. Damit ist EmPy mit seiner Bearbeitung fertig.

Als nächstes kommen die Docutils zum Einsatz. Sie erzeugen aus dem ReStructuredText das fertige HTML-Dokument, welches dann auch zurück zum Browser geschickt wird.

8 Bewertung

8.1 Bewertung anhand des Schemas vom Vortrag „Grundlagen CMS“

In diesem Abschnitt soll das WikiWikiWeb anhand des Grundlagenvortrags „Grundlagen des CMS“ bewertet werden. Als Bewertungsmaßstab habe ich mich für das Schulnotensystem entschieden.

8.1.1 Workflow

Einen Workflow als solches gibt es bei WikiWikiWeb nicht. Es würde auch dem Prinzip des „Instant Publishing“ schaden, wenn alle Änderungen zuerst von einem Kontrolleur durchgesehen werden müssten.

Von daher bekommt das WikiWikiWeb hier die **Note 6** weil hier eine solche Funktionalität komplett fehlt.

8.1.2 Freigabeverfahren

Eine der Designentscheidungen bei dem WikiWikiWeb war es, dass sich Informationen extrem leicht online stellen lassen sollten. Diese Anforderung wurde auch so umgesetzt.

Deshalb hat das WikiWikiWeb hier die **Note 3** verdient. Auf der einen Seite ist es zwar einfach, Informationen zu veröffentlichen, auf der anderen

¹⁰Common Gateway Interface

Seite aber macht sich der fehlende Workflow insofern bemerkbar, als dass das Freigabeverfahren im Bezug auf die anderen WCM-Systeme zwar einfach, aber dennoch unvollständig umgesetzt wurde.

8.1.3 Versionskontrolle

Dieses Feature setzt das WikiWikiWeb so gut um wie kaum ein anderes WCMS. Die Entstehungsgeschichte einer Seite lässt sich recht detailliert verfolgen und Änderungen lassen sich notfalls rückgängig machen. Die **Note 1** ist daher durchaus gerechtfertigt in dieser Kategorie.

8.1.4 Unterstützte Datenbanken

Mittlerweile gibt es Wiki-Engines, die (fast) jede Datenbank unterstützen. Aber eine Datenbank ist nicht zwingend erforderlich, weil es durchaus Wikis gibt, welche den Datenbestand in Form von Dateien ablegen. Deshalb ist die **Note 1** hier vertretbar.

8.1.5 Betriebssystem

Es existieren Implementierungen für nahezu alle verfügbaren Betriebssysteme, sowohl für die verschiedenen Unix-Systeme, als auch für proprietäre Betriebssysteme wie beispielsweise Windows.

Dies bedeutet auch hier die **Note 1** für WikiWikiWeb.

8.1.6 Suchmaschinen

WikiWikiWebs unterstützen in der Regel kein automatisches Eintragen neuer Seiten in eine Suchmaschine. Jedoch gibt es teilweise recht gute wiki-eigene Suchfunktionen. Innerhalb eines Wiki erübrigt sich allerdings meistens eine spezielle Suche da die Seiten selbst miteinander entsprechend sinnvoll verlinkt sind. Deshalb erhält das WikiWikiWeb in dieser Kategorie die **Note 3** weil es auf der einen Seite nicht aktiv mit Suchmaschinen zusammenarbeitet, aber die Seiten untereinander sinnvoll vernetzt und sich meistens eine Suche entweder erübrigt oder sich die bestehenden Suchmaschinen innerhalb eines Wikis zurechtfinden.

8.2 Eigene Bewertung

8.2.1 Vorteile

Ein großer Vorteil des WikiWikiWeb ist die Möglichkeit, relativ einfach Informationen zu veröffentlichen. Einfach in zweierlei Hinsicht:

Einerseits ist der Vorgang an sich recht einfach gehalten, so dass man ohne große Vorkenntnisse sofort aktiv werden kann.

Andererseits sind relativ wenig Schritte nötig, um etwas zu publizieren. Es ist daher nicht nur einfach, sondern auch noch recht schnell.

Der komplett offene Ansatz kann ebenfalls überzeugen. Denn gerade die Offenheit fördert das Gefühl für die Community. Ohne die Gemeinschaft kann ein Wiki nicht existieren.

8.2.2 Nachteile

Der komplett offene Ansatz eines Wikis macht es aber auch für manche Bereiche ungeeignet. Denn dadurch, dass jeder Seiten bearbeiten kann, stellt eine Seite praktisch immer nur die Meinung von demjenigen dar, der sie zuletzt bearbeitet hat.

Natürlich gibt es auch Versuche, ein etwas restriktiveres Benutzermanagement zu integrieren, jedoch wirkt sich solch ein halboffener Ansatz nicht förderlich auf die Gemeinschaft aus.

9 Weiterführende Literatur

Übersicht über die unterschiedlichen Wikis

<http://c2.com/cgi/wiki?WikiEngines>

Wikipedia <http://www.wikipedia.org>

HeyHeyWiki <http://python.net/~gherman/HeyHeyWickie.html>

Artikel über HeyHeyWiki http://www.linux-magazine.com/issue/35/EmPy_Wiki.pdf

EmPy <http://www.alcyone.com/pyos/empy>

ReST Spezifikation

<http://docutils.sf.net/spec/rst/reStructuredText.html>

Portland Pattern Repository

<http://c2.com/ppr>