



**Confluence
macht es
möglich**

Inhalt

Einleitung	4
Glossar verwalten	10
Medien verwalten	11
Diagramm mit Draw.io	12
Diagramm mit Enterprise Architect.....	13
Präsentation mit Powerpoint	16
Externe Links verwalten	18
Mit Jira integrieren	19
Baustein Epic verknüpfen	21
Baustein Case verknüpfen	22
Baustein Service verknüpfen	25
Änderungen kommunizieren	29
Änderungen zusammenfassen	30
Änderungen überprüfen	33
Kommentare moderieren	36
Kategorien von Kommentaren	36
Metrik für Seiten	39
Seitentitel mit Präfix versehen	41
Besseres Suchergebnis	42
Keine Konflikte	42
Verknüpfungen verwenden	44
Seiteninformationen	45

Querverweise bei Begriffen.....	47
Querverweise bei Technologien.....	48
Qualitätssicherung.....	48
Berechtigungen verwenden	52
Bausteine für alle lesbar	52
Bausteine inkrementell erstellen	53
Stichworte verwenden.....	56
Seitensuche	57
Stichwortsuche	58
Stichwortverzeichnis	59
Stichwordhistorie	60
Stichwordhierarchie	60
Stichwortbasierter Bericht	61
Seiten sicher löschen	65

Einleitung

Die Methode cards+ ist ein agiler Ansatz mit dem Ziel, Produktwissen strukturiert und mit hoher Qualität zu erfassen. Ohne Wiki wird die Umsetzung jedoch nicht gelingen.

[Die Methode cards+ auf einen Blick.](#)

Ein Wiki ist eine Web-site, deren Inhalte von den Besuchern nicht nur gelesen, sondern auch direkt im Web-browser bearbeitet und geändert werden können. Das Ziel ist, Erfahrung und Wissen gemeinschaftlich zu sammeln (kollektive Intelligenz) und in für die Ziel-gruppe verständlicher Form zu dokumentieren.

Confluence vom australischen Hersteller Atlassian ist so ein Wiki. Ein Confluence-Bereich ist sehr gut geeignet für den Aufbau einer Produktdokumentation. Confluence hat alle Funktionen, die notwendig sind für die Umsetzung der Methode cards+. Und mit dem Add-On-Konzept gibt es eine Vielzahl von Erweiterungsmöglichkeiten für das Wiki, z.B. Glossary, Scroll-Office, Gliffy, Draw.io oder Balsamiq.

[Confluence macht es möglich.](#)

Bereiche

Confluence verfolgt das Konzept der Bereiche (engl. spaces) und erfüllt damit eine Anforderung von cards+ an ein Wiki. Eine Best-Practice ist, dass jeder Bereich für einen bestimmten Zweck verwendet wird. Im konkreten Fall wird die Produktdokumentation zu einem einzigen Bereich zusammengefasst. Nur für das Glossar wird häufig ein separater Bereich angelegt, weil das Glossar auch für das Benutzerhandbuch, Betriebshandbuch oder allgemein für das ganze Projekt relevant ist.

Startseite

Die Startseite (engl. home page) eines Bereiches ist die Seite, mit der wir unsere Leser begrüßen. Darum enthält die Startseite auch

- einen Überblick über die Struktur der Seiten mit Glossar, Systembeschreibung, Systemstruktur, Architekturentwurf und Medienbibliothek,
- eine Stichwortwolke (engl. word cloud) mit den wichtigsten Begriffen in diesem Bereich,

- eine Suche beschränkt auf diesen Bereich und
- eine Liste der letzten Änderungen an den Seiten in diesem Bereich.

Die Startseite gibt dem Leser einen guten Einstieg in die Produktdokumentation. Durch die Anzeige der letzten Änderungen bekommt der Leser auch einen Überblick über den Schwerpunkt der Aktivitäten und Veränderungen in der Produktdokumentation. Optional können wir auch eine Anzeige von Beiträgen einrichten. Einen Beitrag verwenden wir dann für die Ankündigung eines wichtigen Ereignisses oder für die Dokumentation der Durchführung eines Reviews.

Seiten

Die Grundfunktion von einem Wiki ist es, Seiten (engl. pages) mit Inhalten bereitzustellen. Der Seiteneditor bietet einfache Grundfunktionen. Ein Autor kann aus sechs Überschriftenebenen, drei Absatzformaten (normal, für Code, für ein Zitat) und kombinierbaren Schriftstilen (**fett**, *kursiv*, unterstrichen, ~~durchgestrichen~~) wählen. Zusätzlich bietet Confluence mit

Makro noch weitere Gestaltungsmöglichkeiten, z.B. mehrspaltige Texte.

Textkommentare sind eine sehr elegante Möglichkeit, wie ein Leser Feedback direkt in der Seite zu einem bestimmten Wort oder Satz geben kann. Er muss die Textstelle nur markieren und seinen Kommentar hinzufügen.

Seitenkommentare werden zeitlich geordnet im Anschluss an die Seite gezeigt. Seitenkommentare können auch beantwortet werden und bilden so eine Diskussion.

Ein Stichwort/Schlagwort (engl. label, tag) ist ein Begriff, der einen zentralen Aspekt eines Textes repräsentiert. Es beschreibt den Inhalt in möglichst prägnanter Art und Weise, muss aber nicht im Titel oder im Text der Seite vorkommen.

Atlassian hat den englischen Begriff "label" in den deutschen Begriff "Stichwort" übersetzt. Das ist nicht ganz optimal. Stichworte sind in der üblichen Definition Teil des Textes oder Titels einer Seite. Schlagworte hingegen sind Zusatzinformationen aus einem kontrollierten Vokabular. Schlagworte müssen aber nicht zwingend im Text oder Titel der Seite vorkommen. Schlagwort trifft daher den Kern der Sache besser.

Bei einem Stichwort für eine Seite handelt es sich um festgelegtes, kontrolliertes Vokabular, das der Bearbeiter einer Seite aus dem Kontext des Projektes oder Produktes ableitet.

Die Seitenhierarchie bildet Strukturen innerhalb der Dokumentation mit Seiten und Unterseiten.

Die Seitenhistorie gibt eine Übersicht, welcher Autor zu welchem Zeitpunkt eine Änderung an einer Seite gemacht hat.

Seitenvorlagen

Seitenvorlagen (engl. templates) erfüllen eine weitere wichtige Anforderung von cards+. Sie sorgen für eine einheitliche Struktur der Seiten im Wiki. Ohne Vorlagen ist das praktisch unmöglich. Confluence

bietet zusätzlich in den Vorlagen die Möglichkeit, Hilfetexte für Autoren zu hinterlegen, mit Regeln, die wir im Projekt für die Inhalte der Abschnitte in den Bausteinen festlegen. Diese Texte verschwinden beim Eintippen des ersten Zeichens durch den Autor. Sehr angenehm.

Glossar verwalten

Die einfachste Lösung für ein Glossar in Confluence besteht aus Einzelseiten je Buchstabe im Alphabet als Unterseiten einer Glossar-Seite. Für einen Glossar-Eintrag gibt es eine Vorlage. Für einen neuen Glossareintrag muss der Autor in der richtigen Buchstaben-Seite des Glossars mit der Vorlage Term eine Unterseite mit dem Begriff als Titel und Stichwort anlegen und den Begriff erklären. Jede Glossareintrag-Seite kann in anderen Seiten einfach als Begriff verlinkt werden.

[Baustein Term](#)

Atlassian schlägt in seinem Blog eine ähnliche Lösung für ein Glossar vor. Glossary ist ein spezielles Add-On. Neben der einfacheren Erfassung von Glossareinträgen bietet Glossary ein Dashboard, einen optisch ansprechenden alphabetischen Überblick und weitere sehr nützliche Hilfestellungen für die Pflege und Nutzung des Glossars.

Medien verwalten

Eine Medienbibliothek ist ein Konzept, um Präsentationen, Fotografien (z.B. vom Flipchart oder Whiteboard), Abbildungen und Diagramme, die als Datei (engl. file) in verschiedenen Formaten vorliegen, an einem zentralen "Ort" in einer Medienseite in Confluence abzulegen.

[Baustein File](#)

Jeder Autor muss davon ausgehen, dass ein Bild aus dieser Medienbibliothek in mehr als einer Seite benutzt wird. Er wird entsprechend vorsichtig und abgestimmt mit anderen Autoren Änderungen vornehmen.

Ein weiteres Ziel der Medienbibliothek ist die Unabhängigkeit von Werkzeugen zur Bearbeitung bei der Darstellung für die Leser. Die Vorschau in der Medienseite ist immer eine PNG-Datei. Dadurch ist die Anzeige einer eingebetteten Medienseite immer gleich, unabhängig vom Werkzeug. Dieses Vorgehen erleichtert auch den Export einer Produktdokumentation, z.B. mit einem der Scroll-Exporter.

Diagramm mit Draw.io

Draw.io ist eine kostenpflichtige Erweiterung (engl. addon) für Confluence. Es eignet sich hervorragend zum Zeichnen von einfachen Diagrammen aller Arten. Das Werkzeug bietet zahlreiche Diagrammtypen, die oft in der Software-Entwicklung zum Einsatz kommen. Für die Darstellung in Diagrammen bietet draw.io eine ganze Reihe von Symbolen für

- Unified Modelling Language,
- Enterprise Integration Patterns oder
- Business Process Modeling Language.

Ebenfalls interessant sind Schablonen für Mockups
Die schrittweise Anleitung richtet sich an Autoren, die in Confluence mit draw.io ein Diagramm direkt im Wiki erstellen, pflegen und zentral verfügbar machen wollen, entsprechend dem Konzept der Medienbibliothek. Das Vorgehen basiert auf der Nutzung des Confluence-Addon von Draw.io.

Der Autor legt eine neue, leere Medienseite an. Die übergeordnete Seite ist immer die Startseite der Medienbibliothek. Der Titel der Medienseite beginnt immer mit "File", gefolgt von einer knappen, aber eindeutigen Beschreibung. Jede Medienseite hat mindestens die Stichworte „file“ und „draw-io“.

Mit dem Aufruf der Funktion "Draw.io Diagramm" im Einfügen-Menü des Confluence-Editors legt der Autor eine neue Draw.io-Datei an. Beim Beenden des Editors von Draw.io wird sowohl eine XML- als auch eine PNG-Datei im Anhang der Medienseite gespeichert. Beim ersten Speichervorgang gibt der Autor den Titel der Medienseite als Dateiname für beide Dateien an.

Diagramm mit Enterprise Architect

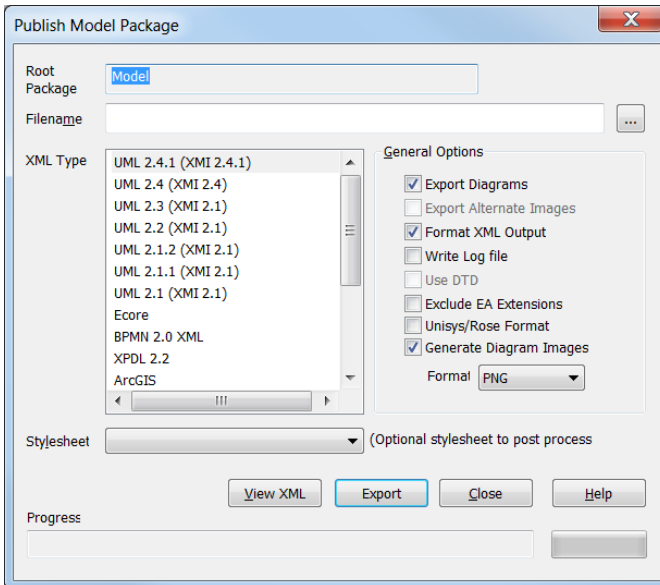
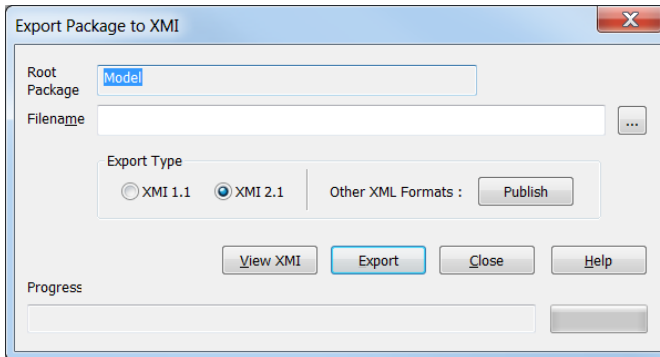
Enterprise Architect von Sparx Systems (kurz EA) ist ein Werkzeug für Analyse und Design mit der Unified Modelling Language (kurz UML). UML ist eine Sprache zur Beschreibung technischer Systeme und hilft konsistente und nachvollziehbare Modelle des Sys-

tems zu erstellen. EA stellt dabei eine benutzerfreundliche und flexible Modellierungsumgebung zur Verfügung. EA wird immer dann eingesetzt, wenn UML-Diagramme nicht mehr trivial sind.

Die schrittweise Anleitung richtet sich an Autoren, die ein komplexes UML-Diagramm im EA erstellen und pflegen, aber im Wiki veröffentlichen und zentral verfügbar machen wollen, entsprechend dem Konzept der Medienbibliothek. Das Vorgehen benötigt kein spezielles Confluence-Addon.

Der Autor legt eine neue, leere Medienseite an. Die übergeordnete Seite ist immer die Startseite der Medienbibliothek. Der Titel der Medienseite beginnt immer mit "File", gefolgt von einer knappen, aber eindeutigen Beschreibung. Jede Medienseite hat mindestens die Stichworte „file“ und „ea“.

Im Anhang der Medienseite wird eine PNG-Dateien des Diagramms als Vorschaubild eingefügt. EA hat dafür eine bestens geeignete Export-Funktion.



Export im Format PNG

Das Schöne an dieser Export-Funktion ist, dass der EA für jedes Bild eine eindeutige Bezeichnung vergibt, die sich nicht mehr ändert. Dieser Bezeichner ist gleichzeitig der Name der exportierten PNG-Datei. Diese vom EA erzeugte PNG-Datei fügt der Autor in den Anhang der Medienseite ein. Damit schlägt er automatisch eine Brücke zwischen EA und Wiki.

Präsentation mit Powerpoint

Powerpoint ist ein sehr beliebtes Werkzeug im Projektmanagement. Auch in der Anforderungsanalyse wird Powerpoint gerne verwendet.

Die schrittweise Anleitung richtet sich an Autoren, die in Confluence eine mit Powerpoint erstellte Präsentation im Wiki zentral verfügbar machen wollen, entsprechend dem Konzept der Medienbibliothek. Das Vorgehen benötigt kein spezielles Confluence-Addon.

Der Autor legt eine neue, leere Medienseite an. Die übergeordnete Seite ist immer die Startseite der Me-

dienbibliothek. Der Titel der Medienseite beginnt immer mit "File", gefolgt von einer knappen, aber eindeutigen Beschreibung. Jede Medienseite hat mindestens die Stichworte „file“ und „powerpoint“.

Im Anhang der Medienseite wird sowohl die Powerpoint-Datei (das Original für weitere Bearbeitung) und eine PNG-Dateien einer Folie als Vorschaubild eingefügt. Der Name der beiden Dateien ist identisch mit dem Titel der Medienseite.

Externe Links verwalten

Eine Linksammlung ist ein Konzept, um dokumentierte Informationen außerhalb von Confluence nutzbar zu machen. Nutzbar machen bedeutet, dass für jedes externe Dokument ein Linkseite als Platzhalter im Wiki angelegt wird. Besonders bei dokumentierten Informationen, die in unterschiedlichen Versionen gültig sein können, ist das ein entscheidender Vorteil. Gilt eine neue Version des externen Dokumentes, muss nur die Linkseite geändert werden.

[Baustein Link](#)

Mit Jira integrieren

Confluence ist ein ausgereiftes Produkt mit vielen Funktionen, die man für ein Wiki braucht. Jira wird vom Hersteller als Werkzeug zur Vorgangs- und Projektverfolgung für agile Teams bezeichnet. Die beiden Produkte der australischen Firma Atlassian bilden ein sehr leistungsfähiges Paar. Hier gilt der (verkürzte) Satz von Aristoteles: Das Ganze ist mehr als die Summe seiner Teile.

Copy. Paste. Done. So schnell kann man die URL eines Vorganges oder eines Filters in Jira in einer Confluence-Seite einfügen. Es ist so einfach, sagt Produktmanager Terrence Caldwell.

blogs.atlassian.com

Jira ist ein sehr leistungsstarkes Werkzeug. Wir können mit Jira sehr gut die Methode Scrum unterstützen, mit Vorgängen für das Backlog vom Typ Epic, Story und Sprint. Das Backlog-Management profitiert von den Möglichkeiten zur Abbildung von Abhängig-

keiten von Stories untereinander, von Story zu Ersteller (der Produktverantwortliche) bzw. Bearbeiter (jemand aus dem Team) und zum Produktwissen in Confluence. Die Einschätzung der Komplexität des Teams aus Refinement- und Planning-Meeting sichert der Produktverantwortliche in der Story, als Grundlage für seine Planung. Der aktuelle Fortschritt wird mit einem Burn-Down-Diagramm dargestellt. Jira hat noch viele weitere Funktionen für die "Prozesssicht", für die Sicht auf den Weg des Scrum-Teams zum richtigen Produkt. Jira beantwortet die Frage, wie wir unser Ziel erreichen.

Confluence hingegen ist ein Werkzeug für die "Produktsicht", die Sicht auf den Zustand des Produktes. Die Methode cards+ versetzt uns in die Lage, eine Produktdokumentation inkrementell zu erstellen. Ein Case wird in der Anforderungsanalyse angelegt, mit der abstrakten Beschreibung des Problems und konkreten Beispielen. Erst in Zusammenarbeit mit dem Team wird die Lösung skizziert. Mit der Übernahme

des Case als Story in den Sprint beginnt die Umsetzung. Am Ende eines Sprints gibt es immer eine neue Version des Produktes (das Produktinkrement), bestehend aus Code, Test und Dokumentation. Der aktuelle Leistungsumfang des Produktinkrements wird durch die Bausteine der Systembeschreibung sichtbar. Art, Umfang und Struktur der Software wird durch weitere Bausteine angemessen beschrieben.

Baustein Epic verknüpfen

Der Baustein Epic ist das Artefakt in unserer Systembeschreibung, mit dem wir ein Ergebnis der Analyse einer Anforderung sichern. Er gibt einer Anforderung einen eindeutigen Namen und enthält die grobe Ausarbeitung der fachlichen Problemstellung. Er aggregiert Vorgänge, Anwendungsfälle, Sonderfälle und Fehlersituation zu einem sinnvollen Ganzen.

Ein Epic in Jira lässt sich ganz leicht mit seinem "Zwilling" in Confluence verknüpfen. Ganz pragmatisch erreichen wir dadurch eine hohe Wiedererkennung der Anforderungen in Jira und Confluence. Mit

[Baustein Epic](#)

einem Klick kann der Nutzer zwischen den beiden Werkzeugen wechseln.

Baustein Case verknüpfen

Der Baustein Case ist das Artefakt in unserer Systembeschreibung, mit dem wir ein konkretes fachliches Problem aus der Analyse einer Anforderung sichern. Er gibt einem Vorgang, Anwendungsfall, einer besonderen Situation oder einem bekannten und akzeptierten Fehler einen aussagekräftigen Titel. Er kann aber auch eine Situation bezeichnen, für die es keine Lösung gibt (z.B. wegen technischer Hürden oder schlicht zu teuer).

Für die Realisierung eines Case, d.h. die Transformation in Code und Test, benötigt das Team eine Story in Jira. Der Titel für Case und Story ist gleich. Wie beim Epic ergibt sich auch hier automatisch eine hohe Wiedererkennung. In der Beschreibung der Story ergänzt der Produktverantwortlicher kurzfristig benötigtes detailreiches Wissen (Beispiele, Szenarien, Testfälle). Spätestens in einem Refinement-

[Baustein Case](#)

Meeting entwickelt das Team zusammen mit dem Produktverantwortlicher die Lösungsidee und dokumentiert sie im Case als Folge von Essenzschritten, in der die Rolle von Services und Events beschrieben wird. Akzeptanzkriterien schließlich sorgen dafür, dass Produktverantwortlicher, Entwickler und Tester das gleiche Verständnis von Umfang und Ziel der Story haben.

Eine Story ist im Normalfall mit der Confluence-Seite des umzusetzenden Cases verbunden. Mit einem Klick auf den Link bekommt der Entwickler alle Informationen aus der Anforderungsanalyse, die für die Story relevant sind. Er wechselt von Jira zu Confluence. Er wechselt einfach die Perspektive. Auch die in der Lösungsidee erwähnten Confluence-Seiten für Services und Events werden mit der Story verknüpft. Alle für Entwickler wissenswerten Besonderheiten einer Komponente oder notwendigen Spezifikationen sind "nur einen Klick" entfernt. Diese Zusammenstellung der wichtigsten Confluence-Seiten

in der Story hat den enormen Vorteil, dass jeder Entwickler sicher und leicht an alle notwendigen Informationen kommt. Der Wechsel von Jira in Confluence und wieder zurück ist völlig transparent.

Gerade bei neuen Produkten gibt es immer wieder die Situation, in der erst die Voraussetzungen innerhalb der Software-Architektur geschaffen werden müssen, bevor eine Case umgesetzt werden kann. Dazu erstellt der Produktverantwortlicher häufig eine sogenannte Enabler-Story im Backlog. In manchen Fällen wird für die Umsetzung eines Qualitätsmerkmals (z.B. Logging, Monitoring, Sicherheit) oder eine Optimierung im Zusammenhang mit einem Case eine separate Story angelegt. Die Gründe dafür sind vielfältig: Der Sprint ist voll, die Umsetzung erfordert Spezialkenntnisse im Team oder das Qualitätsmerkmal ist viel niedriger priorisiert als die fachliche Funktion.

Jede dieser technisch motivierten Stories hat wenig bis keinen wirksamen fachlichen Mehrwert. Trotzdem wird jede Story mit dem Case verknüpft, für den sie die Voraussetzungen oder Verbesserungen

schaft. Damit bekommen wir in Confluence eine wunderbare Übersicht über alle Stories, die für einen Case relevant sind. Mit dieser zusätzlichen zum Backlog in Jira verfügbaren Sicht auf Stories fällt es dem Produktverantwortlichen leichter, Konflikte zwischen Stories zu erkennen. Mit einem Blick sieht er den aktuellen Status und Bearbeiter der für einen Case geplanten Stories.

Baustein Service verknüpfen

Der Baustein Service ist das Artefakt der Systemstruktur, mit dem wir einen Dienst unserer Anwendung so beschreiben, wie wir ihn realisiert haben. Im Steckbrief des Bausteins stehen grundlegende Informationen über die gewählten Technologien.

In der Regel schreiben wir Stories, um Fähigkeiten des Produktes zu ergänzen, zu ändern oder eventuell sogar zu entfernen. Es gibt aber auch Qualitätsmerkmale, die keinen direkten Bezug zu einem Case haben. Die Aufarbeitung von technischen Schulden

[Baustein Service](#)

zählt dazu. Gleiches gilt für den Fall, dass es eine wesentliche Änderung in der Software-Architektur gibt. Das kann eine geplante Aktualisierung einer Bibliothek oder Middleware sein, der Einsatz neuer Features der Programmiersprache oder der Austausch eines Infrastrukturproduktes (z.B. Datenbank-Server, Messaging-Broker, Streaming-Cluster). Nicht unerwähnt bleibt die grundlegende Optimierung eines Dienstes.

Ein daraus resultierendes Refactoring im Code mit einer ausreichenden Komplexität und dem damit verbundenen Risiko nimmt ein Produktverantwortlicher gerne als Story in das Backlog auf. Die Story wird mit den Confluence-Seiten der Services verknüpft, die von der Story betroffen sind. Nach Abschluss des Refactorings kann das Team lückenlos die notwendigen Änderungen an der Dokumentation vornehmen. Der Produktverantwortlicher kann in der Zwischenzeit erkennen, welche Services gerade technisch "renoviert" werden. Er wird einen Case verschieben,

wenn der ebenfalls Änderungen an einem der betroffenen Services erfordert. Hier schafft Confluence ohne große Mühe eine Sicht, die der Produktverantwortlicher im Backlog in Jira nur sehr schwer nachbilden kann.

Fazit

Der Produktverantwortlicher zieht große Vorteile für sein Backlog-Management aus der Kombination von Jira und Confluence. Mit Jira hat er den Prozess der agilen Software-Entwicklung mit Scrum im Griff. In Confluence sammeln er inkrementell das dafür nötige Produktwissen. Confluence zeigt ihm in jeder Seite die Verknüpfungen zu Stories in Jira an. Dadurch behält er die Übersicht über jeden Case oder Service.

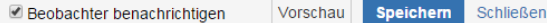
- Er sieht, ob ein Case schon im Backlog berücksichtigt wurde.
- Er sieht, welcher Service in welchem Case relevant ist.

- Er sieht, in welchen Cases ein Service verwendet wird.
- Er kann die Vollständigkeit des Backlog einschätzen.
- Er bekommt mit einem Klick den aktuellen Status in Jira.
- Er sieht den aktuellen Bearbeiter im Team.
- Er kann Konflikte bei Stories vor der Umsetzung lösen.
- Er erkennt Duplikate im Backlog.

Beide Werkzeuge sind web-basiert und können ganz leicht in jeder Arbeitsumgebung oder auch unterwegs genutzt werden. Damit sind sie Werkzeuge für das ganze Team. Confluence und Jira passen sehr gut zusammen. Durch die sehr gute Integration der beiden Produkte behält das Team gleichzeitig Überblick über den Entwicklungsprozess und die Produktkredimente. Es gibt ausreichend viele gute Gründe, Jira und Confluence zusammen mit den agilen Methoden Scrum und cards+ einzusetzen.

Änderungen kommunizieren

Ein Autor hat in Confluence verschiedene Ansätze, um ihre Änderungen an seine Leser zu kommunizieren. Jede Änderung wird automatisch an die Beobachter einer Seite gemeldet, wenn er den Haken bei “Beobachter benachrichtigen” (“Notify Watcher”) setzt.

A screenshot of a Confluence interface showing a row of four buttons. The first button is a checkbox labeled 'Beobachter benachrichtigen' with a checked mark. The second button is 'Vorschau', the third is 'Speichern' (highlighted in blue), and the fourth is 'Schließen'.

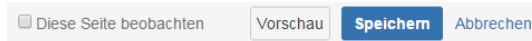
Beobachter benachrichtigen

Jeder Leser kann sich selbst zum Beobachter einer Seite machen, indem er in der Seite die Schaltfläche “Beobachten” (“Watch”) klickt.

Mit der Funktion “Teilen” (“Share”) informiert ein Autor aktiv einen fixen Leserkreis (Nutzer oder Nutzergruppen) über Änderungen an einer bestimmten Seite. Zusammen mit einem erläuternden Hinweis des Autors zu den Änderungen schickt Confluence automatische eine Mail an jeden adressierten Leser.

Änderungen zusammenfassen

Confluence ist mitunter sehr “gesprächig”, verschickt eine Mail für jede geänderte Seite, die ein Nutzer beobachtet oder jeden Seitenkommentar, den ein anderer Nutzer beantwortet. Leider wird ein Nutzer in Confluence sehr schnell zum Beobachter einer Seite. Eine kleine Verbesserung eines Rechtschreibfehler, ein kurzes Feedback zum Inhalt, und schon aktiviert Confluence die Beobachtung dieser Seite, wenn es der Nutzer nicht abhakt.



Seite nicht beobachten

Eine weitere Verbesserung schafft die Funktion “Täglicher E-Mail-Bericht” (“Subscribe to daily updates”).

Das ist eine Funktion, die jeder Nutzer in seinen persönlichen Einstellungen aktivieren kann.

E-Mail-Einstellungen

Automatisch abonnieren

Zukünftige Änderungen von Seiten und Blogbeiträgen, die Sie erstellen, bearbeiten oder kommentieren, werden automatisch abonniert.

Tägliche Aktualisierungen abonnieren

Sie erhalten einen täglichen E-Mail-Bericht, der alle Änderungen zusammenfasst, die Sie anzeigen dürfen.

E-Mail-Einstellungen

Confluence schickt dem Nutzer jeden Tag (in der Regel kurz nach Mitternacht) eine einzige E-Mail mit einer Zusammenfassung der Änderungen und Kommentare aller Bereiche des letzten Tages.

Confluence-Änderungen in den letzten 24 Stunden

24.11.2016 • eMundo Wiki

CARDS+

- Systemstruktur 17:25
Bearbeiter [Robert Bruckbauer](#) • [Änderungen anzeigen](#)
- Architekturentwurf 17:24
Bearbeiter [Robert Bruckbauer](#) • [Änderungen anzeigen](#)
- Spec Dokumentation mit CARDS+ 09:57
Bearbeiter [Robert Bruckbauer](#) • [Änderungen anzeigen](#)
- Image CARDS+ auf einen Blick 09:52
Erstellt von [Robert Bruckbauer](#)
- Image Überblick Topic 09:53
Erstellt von [Robert Bruckbauer](#)
- Systembeschreibung 09:55
Bearbeiter [Robert Bruckbauer](#) • [Änderungen anzeigen](#)

E-Mail-Bericht

Leser können nun selbst entscheiden, wann sie sich über die Änderung des letzten Tages informieren. Autoren bekommen täglich einen Überblick über Änderungen anderer Autoren in seinem Bereich.

Änderungen überprüfen

Ein einfacher Ansatz, wie ein Autor eine zielgerichtete Überprüfung einer Änderung an einer oder mehreren von ihm bearbeiteten Seiten organisiert, besteht aus einer Kombination folgender Funktionen von Confluence:


- Beitrag (engl. post)
- Aufgabe (engl. task)
- Nutzererwähnung (engl. mention)

Der Autor kann die zu prüfenden Seiten beschränken. Gerade bei ganz neuen Themen oder bei großer Unsicherheit bei den Änderungen vermeiden Lesebeschränkungen für die Zeit der Überprüfung unnötige Diskussionen.

Mit einem Review-Beitrag formuliert ein Autor den Wunsch nach einem Review. Der Beitrag enthält eine kurze Erklärung/Motivation und eine Liste der zu prüfenden Seiten – natürlich als Verknüpfung.

Abhängig von der Projektkultur sucht sich der Autor bei nächster Gelegenheit (z.B. Daily, Jour-Fixe) andere Autoren, die seine Seiten prüfen. Jene Kollegen, die sich für die Prüfung der Dokumentation bereit erklären, ergänzt der Autor im Review-Beitrag. Für jeden Reviewer legt er eine Review-Aufgabe (engl. task) mit Nutzererwähnung (engl. mention) an. Er kann sogar einen Termin vergeben.

Jeder Reviewer wird automatisch durch Confluence über eine Review-Aufgabe benachrichtigt. Die Aufgabe ist mit dem Review-Beitrag verknüpft und wird automatisch in der persönlichen Aufgabenliste des Reviewers angezeigt.

Mir zugewiesen	Erstellt von mir	Unvollständig	Abgeschlossen
Beschreibung	Fälligkeitsdatum ▾	Aufgabe wird angezeigt auf	
<input type="checkbox"/> @Robert Bruckbauer bitte bis zum  30.11.2016	30.11.2016	Review Medienbibliothek	
<input type="checkbox"/> @Robert Bruckbauer		Review Anpassung an neue Version von CARDS+	

[Aufgabenbericht mit eigenen Einstellungen erstellen](#)

Für die Prüfung selbst nutzt der Reviewer die Kommentarfunktionen. Wenn der Reviewer seine Prüfung erledigt hat, hakt er seine persönliche Aufgabe ab.

Der Autor bekommt kontinuierliches Feedback der Reviewer. Er erkennt selbst anhand der Aufgabenliste im Review-Beitrag, wann alle Prüfungen erledigt wurden. Dann beginnt der Autor mit der Abarbeitung der Kommentare. Er nutzt dabei die Idee der moderierten Kommentare. Ziel des Autors ist, am Ende alle Kommentare zu klären und in den Seiten zu löschen.

Zum Abschluss des Reviews wird der Review-Beitrag einfach gelöscht. Eventuelle Lesebeschränkungen der geprüften Seiten kann der Autor jetzt entfernen.

Leser können sich jederzeit in der Beitragsübersicht des Bereiches über die aktuellen Reviews informieren. Sie bekommen so wertvolle Hinweise auf kommende Veränderungen in der Dokumentation und können die Kommentare in den von einem Review betroffenen Seiten besser einschätzen.

Kommentare moderieren

Die Produktdokumentation ist ein Ergebnis des ganzen Teams. Der Produktverantwortliche, unterstützt von Analysten und IT-Experten, beginnt mit dem fachlichen Entwurf. Zusammen mit dem Team ergänzt er die Lösung im Rahmen des für die Software gewählten Architekturstils. Jeder Analyst, Entwickler und Tester im Team hat Anteil an einer guten Dokumentation. Er korrigiert Fehler, kommentiert Unklarheiten oder macht Vorschläge für Verbesserungen, mit Kommentaren direkt im Wiki.

Confluence bietet zwei Wege, wie Leser Kommentare hinterlassen können: Textkommentare direkt in der Seite und zeitliche geordnete Seitenkommentare.

Kategorien von Kommentaren

Offener Punkt

Offene Punkte und andere Unklarheiten versucht der Autor in einem persönlichen Gespräch mit dem Ersteller eines Kommentars zu klären. Ein Kommentar kann aber auch im Daily angesprochen werden, um

eine Klärung zusammen mit dem Team herbeizuführen. Der Rahmen für so eine Klärung kann das regelmäßige Refinement-Meeting sein. Alternativ kann ein Workshop organisiert werden, wenn zur Klärung interessierte Parteien oder andere Experten erforderlich sind. Das Ergebnis der Klärung nutzt der Autor, um den Inhalt einer Seite zu verbessern.

Fehler

Es ist im ureigensten Interesse eines Autors, dass Fehler in der Dokumentation schnell gefunden werden. Das Feedback des Teams ist dabei der effizienteste Weg. Analysten, Entwickler und Tester haben unterschiedliche Sichten auf das Produkt. Ein Analyst kann die Nutzersicht einnehmen, kennt die Prozesse, und erkennt so Widersprüche. Ein Entwickler erkennt einen Fehler in der Dokumentation, weil er weiß, wie das Produkt programmiert wurde. Ein Tester kennt einen Testfall, der im Widerspruch zur Dokumentation steht. Am Ende eines Sprints darf eine

Produktdokumentation nur das beschreiben, was das Produktinkrement leistet.

Feedback

Eine schöne Funktion von Confluence ist die Benachrichtigung des Autors einer Seite, wenn ein Leser einen neuen Kommentar geschrieben hat. Der Autor verliert nie den Überblick über die Kommentare. Er entscheidet aber selbst über den Zeitpunkt, zu dem er einen Kommentar bearbeitet. Die gleiche Funktion informiert den Leser, der den Kommentar erstellt hat, wenn er gelöscht bzw. gelöst wurde. Benachrichtigungen sind automatisch erzeugtes Feedback.

Ein für alle Autoren und Leser sichtbares Feedback gibt das Makro "Zuletzt aktualisiert" ("Activity Stream"). Mit dem Makro werden alle Änderungen und Kommentare von Seiten an einer zentralen Stelle im Wiki platziert werden. Durch die Einschränkung auf den Bereich (engl. space) der Produktdokumentation sehen

Metrik für Seiten

Eine Seite im Wiki, die keine Kommentare enthält, ist eine gute Seite. Andererseits ist eine Seite im Wiki mit Kommentaren schlicht und einfach nicht fertig. Kommentare sind Hinweise auf Unklarheiten, offene Punkte oder Fehler. Sehr viele Kommentare aus einer von mehreren Lesern kontrovers geführten Diskussion sind ein klares Indiz für den Autor, eine Seite noch einmal intensiver zu betrachten. Kommentare sind eine Metrik für die Qualität.

Fazit

Kommentare unterstützen die Kommunikation im Team. Sie machen Feedback einzelner Personen für alle sichtbar. Sie lösen eine Diskussion aus, wenn es kontroverse Kommentare gibt. Sie sind die einzig verfügbare messbare Größe für die Qualität einer Seite. Darum ist es besonders wichtig, dass Autoren in allen Fällen Kommentare löschen, wenn sie bearbeitet oder geklärt sind. Der Ersteller des Kommentars wird von Confluence automatisch informiert. Er kann die

Änderungen an der Seite überprüfen. Im besten Fall ist nun alles gut. Ist er aber nicht zufrieden, kann er dies erneut als Kommentar in der Seite vermerken.

Seitentitel mit Präfix versehen

Die Methode `cards+` definiert Bausteine, die sich in Struktur und Definition des Inhaltes unterscheiden. Das Präfix ist der Name des Bausteins. Die Verwendung eines Präfixes im Titel jeder Seite ist hilfreich für Leser, weil sie mit einer gewissen Erwartungshaltung den Inhalt einer Seite lesen können.

Die einzige Ausnahme von dieser Regel bilden Glosseinträge. Damit können die Begriffe aus dem Glossar ganz einfach in jedem Text verwendet werden. Ein Präfix würde hier ganz massiv stören.

Es ist ein großer Vorteil, wenn wir Leser haben, die konstruktives, zielgerichtetes Feedback geben können. Durch das Präfix erkennt ein Leser sofort, um welchen Baustein es sich handelt. Er kann anhand der Definition des Bausteins beurteilen, ob der Inhalt angemessen ist. Durch die bekannte Struktur des Bausteins ist er auch in der Lage, die Vollständigkeit einer Seite zu beurteilen.

Besseres Suchergebnis

Eine Produktdokumentation wird unter anderem in einem Wiki realisiert, weil es dort vielfältige Suchfunktionen gibt.

Ein Suchergebnis kann abhängig vom gesuchten Begriff viele Treffer haben, trotz einer Einschränkung der Suche auf den Bereich (engl. space) der Produktdokumentation. Durch das Präfix im Seitentitel ist aber immer klar, um welchen Baustein es sich bei den Seiten im Suchergebnis handelt. Das ist Erwartungsmanagement im Kleinen.

Keine Konflikte

Confluence hat die Einschränkung, dass ein Seitentitel in einem Bereich (engl. space) nur einmal verwendet werden kann. Das ist in vielen Fällen unkritisch. Es gibt aber eine Reihe von Bausteinen, die ohne Präfix durchaus Potential für Konflikte im Seitentitel haben.

Nehmen wir als Beispiel die Bausteine Topic und Domain. Nutzt der Produktverantwortliche und sein

Team den Modellierungsansatz “Domain Driven Design”, dann passiert es sehr häufig, dass ein Topic (fachlicher Kontext) und die entsprechende Domain (technischer Kontext) den gleichen Namen haben. Jetzt könnte die Lösung sein, nur für den technischen Kontext das Präfix “Domain” zu verwenden. Ähnlich verhält es sich bei den Bausteinen Entity und Event. Mit dem Baustein Entity wollen wir ein Verständnismodell für ein wichtiges Informationsobjekt schaffen. Gleichzeitig kann so ein Informationsobjekt Teil einer nachrichten-orientierten Schnittstelle sein. Es liegt also nahe, den Bausteinen Entity und Event den gleichen Namen zu geben. Das geht aber ohne Präfix in Confluence nicht.

Fazit

Das Präfix ist sicherlich eine Besonderheit in einem Wiki. Das Präfix ist eine Empfehlung aus der Praxis. Sie ist sehr einfach umzusetzen. cards+ funktioniert auch ohne diese Regel. Aber nicht so gut.

Verknüpfungen verwenden

Ein Wiki lebt von der Dynamik in den Inhalten. Leser erwarten von den Autoren, dass Informationen im Wiki optimal aufbereitet sind. Zusammenhänge sind wichtig, gleiches Wissen wird an unterschiedlichen Stellen vermittelt. Die große Gefahr bei einem Wiki ist das Kopieren von Inhalten. Genau wie eine Papierkopie ist eine kopierte Information falsch oder veraltet, sobald sich die originale Information ändert.

Confluence bietet eine Reihe von Funktionen, mit denen Seiten miteinander verknüpft werden. Aber was noch viel wichtiger ist: Confluence bietet auch eine Funktion, um den Überblick über die Verknüpfungen zu behalten.




Eine Seite mit einer anderen zu verlinken ist der einfachste Fall einer Verknüpfung (engl. link) in Confluence. Eine Verknüpfung wird in der Leseansicht der Seite farblich hervorgehoben. Durch Klick auf den Text wird der Leser auf die andere Seite geleitet.

Das Makro "Seiten einschließen" ("Include Page") ist eine spezielle Variante einer Verknüpfung von Seiten in Confluence. Ohne zusätzlichen Klick sieht ein Leser einen Text oder ein Bild in verschiedenen Seiten. Der Autor muss den Inhalt aber nur an einer einzigen Stelle pflegen. Die Funktion bildet u.a. die Grundlage für das Konzept der Medienbibliothek.

Seiteninformationen














Confluence zeigt in den Seiteninformationen (engl. page information) eine ganze Reihe von Metadaten zu jeder Seite. Dazu zählen neben dem Seitentitel und der URL (normal und kurz) die Seitenhierarchie, ein Verzeichnis für ankommende und abgehende Verknüpfungen, eine Liste von Stichworten und die Änderungshistorie.

Abgehenden Verknüpfungen (engl. outgoing links) zeigen einem Autor, welche anderen Seiten in der vorliegenden Seite relevant sind. Confluence unterscheidet externe und interne Verknüpfungen.

Abgehende Links	
Externe Links (2)	
	cardsplus.info/cards
	cardsplus.info/cards-auf-einen-blick/
CARDS+ (1)	 Produktdokumentation

Abgehende Links

Ankommende Verknüpfungen (engl. incoming links) wiederum zeigen, wo die vorliegende Seite eine Rolle spielt, also entweder im Text verlinkt oder als Seite eingebettet wurde.

Ankommende Links	
CARDS+ (13)	
	Case Architekturentscheidung dokumentieren
	Case Aufgabenbereich dokumentieren
	Spec Dokumentation mit CARDS+
	Decision Vorgehen bei der Systembeschreibung
	Case Dienst dokumentieren
	Case Komponente dokumentieren
	Case Nachricht dokumentieren
	Case Fachklasse dokumentieren
	Case User-Story vorbereiten
	Topic Produktdokumentation
	Decision Vorgehen beim Architekturentwurf
	Case Nicht-funktionalen Aspekt dokumentieren
	Case Anforderung dokumentieren

Ankommende Links

Der Autor bekommt durch die Seiteninformationen einen Eindruck von der “Größe” der Seite. Je mehr

Verknüpfungen eine Seite zu anderen Seiten hat, desto komplexer ist in der Regel ihr Inhalt. Und umso sorgfältiger muss ein Autor seine Änderungen gestalten.

Querverweise bei Begriffen

Begriffe im Glossar helfen uns bei der Etablierung einer gemeinsamen Sprache. Das Glossar besteht aus vielen einzelnen Seiten je Begriff, gruppiert nach dem Anfangsbuchstaben. Ein Begriff kann ganz leicht als Verknüpfung in jedem Text in den verschiedenen Bausteinen der Systembeschreibung und Systemstruktur verwendet werden. Diese Verknüpfungen helfen uns, Begriffe konsequent und konsistent zu benutzen. Konsequent, weil das Glossar unser Vokabular definiert. Konsistent, weil der Begriff nur im Glossar erklärt wird, im Text aber ohne weitere Erklärung einfach eingesetzt wird. Umgekehrt bündelt jede Seite im Glossar durch seine Seiteninformation andere Seiten im Wiki, die mit dem beschriebenen Begriff irgendwie zu tun haben.

Querverweise bei Technologien

Der Baustein Service verwenden wir, um Diensten eines IT-Systems einen Namen zu geben und Wissen über die Implementierung zu bündeln, das durch Code und Test nicht ausreichend dokumentiert wird. Dazu zählt auch das Wissen über die verwendeten Technologien, als einfache Liste. Gibt es eine Seite für jede Technologie, besteht diese Liste aus Verknüpfungen. Ein Entwickler kann sich mit einem Klick über die im Dienst verwendete Technologie informieren. In den Seiteninformationen der Technologieseite bilden die eingehenden Verknüpfungen ganz automatisch einen Katalog der Dienste, die eine bestimmte Technologie nutzen.

Qualitätssicherung

Confluence aktualisiert den Text von Verknüpfungen zu anderen Seiten, wenn es eine interne Verknüpfung ohne vom Seitentitel abweichendem Text ist. Bei der Eingabe einer Verknüpfung muss der Autor unbe-

dingt darauf achten, dass im Eingabefeld Verknüpfungstext nichts steht. Im Text wird trotzdem der Titel der Seite als klickbarer Link angezeigt.

Produktdokumentation CARDS+ Suchen

Überschrift	Bereich	Letzte Änderung
 Produktdokumentation	CARDS+	vor 13 Minuten
 Topic Produktdokumentation	CARDS+	vor etwa 13 Stunden
 Decision Werkzeug für die Produktdokumentation	CARDS+	09.01.2017
 CARDS+	CARDS+	vor 11 Minuten
 Spec Dokumentation mit CARDS+	CARDS+	vor etwa 13 Stunden
 ...	CARDS+	01.12.2016

Verknüpfungsstando CARDS+ Agil dokumentieren mit ... Glossar P Produktdokumentatic

Verknüpfungstext

Verknüpfung herstellen

Leider kopieren Autoren Verknüpfungen gerne eine Verknüpfung aus der Leseansicht einer anderen Seite oder aus dem Webbrowser in den Text ein (engl. copy & paste). Die Verknüpfung sieht optisch genau gleich aus wie eine interne Verknüpfung, ein

Klick leitet den Leser auch auf die richtige Seite. Bei dieser externen Verknüpfung wird aber der Text von Confluence nicht automatisch angepasst, wenn der Seitentitel der verlinkten Seite geändert wird.

Fazit

Verknüpfungen sind eine schöne Sache für Leser. Durch die Seiteninformation behalten Autoren den Überblick über Querverweise und Verknüpfungen in ihren Seiten.

Die Seiteninformationen sind ein Werkzeug zur Qualitätssicherung. Es ist bewährte Praxis, dass ein Autor zum Abschluss der Bearbeitung einer Seite einen kurzen Blick auf die Seiteninformation wirft. Er prüft jede externe Verknüpfung. Wenn es sich in Wirklichkeit um eine interne Verknüpfung auf eine Seite im falschen Format handelt, wandelt er sie um. Diese kleine Aufmerksamkeit kostet den Autor nur wenig Zeit. Sie hilft aber, die Qualität der Inhalte dramatisch zu erhöhen.

Leider bietet Confluence die Seiteninformationen nur Funktion in der Leseansicht der Seite. Es fehlt die Möglichkeit, die Seiteninformationen auch für die Gestaltung von Inhalten zu nutzen. Aber wie so oft im Confluence-Ökosystem gibt es Erweiterungen (engl. addons), die diese Lücke schließen.

Berechtigungen verwenden

Confluence unterscheidet drei Arten von Berechtigungen: Globale Berechtigungen, bereichsbezogene Berechtigungen und Seitenberechtigungen. Berechtigungen gelten in der Regel für ganze Nutzergruppen. Im Einzelfall werden Seitenberechtigungen auch an Nutzer vergeben. Berechtigungen können erteilt, aber nicht explizit entzogen werden. Ein einzelner Nutzer kann darum niemals ausgeschlossen werden, wenn er Mitglied einer Nutzergruppe ist, die berechtigt ist.

Das Berechtigungskonzept von Confluence ist trotz seiner Einfachheit durchdacht und für viele Anwendungsfälle sehr gut geeignet.

Bausteine für alle lesbar

Ein Wiki soll Transparenz schaffen. Dafür ist es aber notwendig, dass alle interessierten Parteien die Seiten zumindest lesen können. Das Bearbeiten einer Seite sollte aber für Projektfremde Personen ausgeschlossen werden.

[Sebastian Höhne](#) gibt in seinem Blog einen guten Überblick.

Confluence unterscheidet Seitenberechtigungen für Lesen und Bearbeiten einer Seite. Jede Beschränkung einer Seite wird hierarchisch an deren Unterseiten vererbt. Ein Nutzer muss immer sowohl vererbte als auch explizit für die Seite gesetzte Berechtigungen haben, um die Seite sehen zu können.

Seiten mit Lesebeschränkung können über ihre Verknüpfung (z.B. ein JIRA-Link) von allen Nutzern aufgerufen werden. Nicht berechnigte Nutzer sehen, dass es die Seite gibt, aber nicht deren Inhalt.

Bausteine inkrementell erstellen

Die Methode cards+ unterstützt inkrementelles Dokumentieren mit einer ganzen Reihe aufeinander aufbauender Bausteine. Autoren im Projekt- und Produktmanagement, in der Anforderungsanalyse und bei Realisierung, Test, Einführung und Betrieb nutzen diese maßgeschneiderten Bausteine zur Sicherung ihrer Ergebnisse und Erkenntnisse.

Lesebeschränkungen lassen sich sehr gut für eine schrittweise Erstellung einer Confluence-Seite für so einen Baustein verwenden.

Die Bausteine von cards+ sind so gestaltet, dass jede Seite innerhalb eines Prozessschrittes abgeschlossen werden kann. Trotzdem wird so eine Seite nicht immer in einem Rutsch geschrieben. Die Fertigstellung der Seite erfordert häufig Klärungen durch den Autor, die sich auch über einen längeren Zeitraum erstrecken können. Eine Lesebeschränkung der Seite ausschließlich auf den Autor stellt sicher, dass kein Leser einen unvollständigen, nicht abgestimmten oder möglicherweise sogar fehlerhaften Stand zu Gesicht bekommt. Der Autor kann in Ruhe und ohne Einschränkung im Wiki seine Arbeit abschließen.

Jeder Baustein hat eine nachvollziehbare Struktur. Darum kann jede Seite entsprechend den Vorgaben durch den Baustein geprüft werden. Der Autor erweitert im zweiten Schritt die Lesebeschränkung auf jene Personen, die eine Seite prüfen werden. Mit

Kommentaren in der Seite können die Prüfer ihr Feedback transparent für alle Beteiligten geben. Auch diese Kommentare sind nur für die berechtigten Personen sichtbar und können vom Autor schrittweise abgearbeitet werden.

Nach Abschluss aller Prüfungen und der ggfs. notwendigen Korrekturen gibt der Autor die Seite für alle Leser frei. Er tut dies, indem er die Lesebeschränkungen entfernt. Die weitere Bearbeitung der Seite ist ab jetzt eine Aufgabe des ganzen Teams.

Fazit

Dieser sehr einfache Workflow lässt sich bereits mit Bordmitteln von Confluence realisieren. Er gibt einem Autor die volle Kontrolle über den Zeitpunkt, wann Prüfer und Leser einen Baustein im Wiki sehen können. Mit diesem Workflow stellt ein Autor sicher, dass nur geprüfte Inhalte im Wiki für alle Leser sichtbar sind.

Stichworte verwenden

Eine wachsende Menge an Seiten in unserem Wiki führt dazu, dass die Baumstruktur in Confluence unübersichtlich wird. Stichworte, richtig verwendet, sind sie ein sehr leistungsfähiges Mittel, um Seiten im Wiki schnell und sicher zu finden, zusätzlich zur Navigation in der Baumstruktur. Stichworte stellen neben Kommentaren und dem Gefällt-Mir-Status sehr nützliche Zusatzinformationen einer Seite im Wiki dar. Stichworte können praktisch in beliebiger Anzahl verwendet werden.

Ein Stichwort/Schlagwort (engl. label, tag)

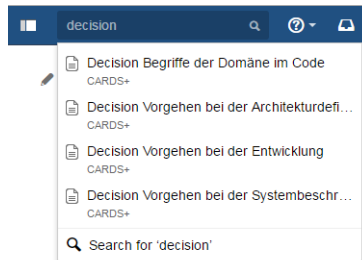
- ist ein Begriff, der einen zentralen Aspekt eines Textes repräsentiert.
- beschreibt den Inhalt in möglichst prägnanter Art und Weise.
- muss nicht im Titel oder im Text der Seite vorkommen.

Bei einem Stichwort handelt es sich um festgelegtes, kontrolliertes Vokabular, das der Bearbeiter einer

Seite aus dem Kontext des Projektes oder Produktes ableitet. Wichtig ist aber, dass Stichworte einheitlich benutzt werden. Genaugenommen nutzen wir die Funktion der Stichworte für Schlagworte. Unser Vokabular für Schlagworte ist das Glossar. Nur so können wir die Möglichkeiten nutzen, die Confluence mit einer Vielzahl von Funktionen und Makros bietet.

Seitensuche

Suchen ist bereits Teil unserer Kultur. Nicht ohne Grund ist 'googeln' der Begriff für eine Suche oder Recherche im Internet. Was liegt also näher, als diese Idee auch in unserem Wiki umzusetzen.



Suche nach dem Begriff „decision“

Bei der Suche nutzen wir den Umstand, dass wir am Beginn des Titels der Seiten in unserem Wiki immer den Namen des Bausteins angeben. In diesem Beispiel suchen wir Seiten vom Typ 'Decision'. Die inkrementelle Suche kann durch Eingabe weiterer Buchstaben immer weiter verfeinert werden. Der im Suchergebnis angezeigte Name des Bereichs ist eine zusätzliche Hilfestellung.


Wir können so unser Wiki für Recherchen im Zusammenhang mit unserem Produkt verwenden. Aber wie bei einer Suche im Internet sind die Suchergebnisse einer Seitensuche oft zu groß oder zu wenig exakt. Auch hier bietet Confluence eine Lösung in Form des Makros "Seitenhierarchiesuche" ("Page Tree Search"). Gibt man dort einen Text im Eingabefeld ein, so wird in dieser und allen untergeordneten Seiten in diesem Bereich gesucht.


Stichwortsuche


Die Suche nach Stichworten ist wesentlich exakter als die Suche nach Seiten oder die Volltextsuche.


This list shows content tagged with the following label: [decision](#)


To add a label to the list of required labels, choose '+ labelname' from Related Labels.

 [Medien > EAID_345B37F7_AF5F_4e14_BFAC_88051C3E6081.png](#)
23.11.2016 • Robert Bruckbauer
[image](#) [ea](#) [spec](#) [decision](#) [subsystem](#)

 [Decision Begriffe der Domäne im Code](#)
19.05.2016 • Robert Bruckbauer
[decision](#) [dokumentation](#)

 [Decision Vorgehen bei der Architekturdefinition](#)
04.05.2016 • Robert Bruckbauer
[decision](#) [cards+](#) [dokumentation](#)

 [Decision Vorgehen bei der Systembeschreibung](#)
04.05.2016 • Robert Bruckbauer
[decision](#) [cards+](#) [dokumentation](#)

 [Decision Vorgehen bei der Entwicklung](#)
04.05.2016 • Robert Bruckbauer
[decision](#) [scrum](#) [dokumentation](#)

Related Labels

[+ page](#) [+ sem](#) [+ test](#)
[+ image](#) [+ cards+](#) [+ ta](#)
[+ frontend](#) [+ web](#) [+ spec](#)
[+ scrum](#) [+ dokumentation](#)
[+ testing](#) [+ datennutzung](#)
[+ subsystem](#)
[+ merkmalsvektor](#) [+ ea](#)
[+ mobile](#) [+ app](#)

Suche nach dem Stichwort „decision“

In dem Beispiel wird die Suche generell auf Seiten mit dem Stichwort 'decision' eingeschränkt. Jede Eingabe eines weiteren Stichwortes verfeinert das Suchergebnis.

Stichwortverzeichnis

Aus den verwendeten Stichworten erzeugt Confluence mit dem Makro "Populäre Stichworte" ("Popular Labels") ein alphabetisch geordnetes Stichwortverzeichnis.

Das Verzeichnis gibt es auch in Form einer Wolke (engl. tag cloud, heat map), in der die Größe der Stichworte proportional zu der Häufigkeit der Verwendung ist. Die Anzahl der angezeigten Stichworte ist beschränkt. Der Grenzwert ist aber konfigurierbar. Ein Klick auf ein Stichwort im Verzeichnis startet sofort eine entsprechende Stichwortsuche.

Stichworthistorie

Als Stichworthistorie bezeichnen wir die Auflistung der Stichworte auf Seiten, die zuletzt geändert wurden. Confluence bietet das Makro “Zuletzt verwendete Stichwörter” (“Recently Used Labels”), das eine Liste der zuletzt genutzten Stichwörter erzeugt. Die Größe der Liste kann konfiguriert werden. Die Liste ist eine echte Untermenge des Stichwortverzeichnisses. Die Stichworthistorie gibt Aufschluss über den aktuellen Schwerpunkt der Änderungen im Wiki.

Stichworthierarchie

Eine Auflistung aller Stichworte von anderen Seiten, die mindestens ein gemeinsames Stichwort mit der

aktuellen Seite haben, bilden eine Stichworthierarchie. Confluence bietet dafür das Makro “Zugehörige Stichwörter” (“Related Labels”). So eine Hierarchie gibt Aufschluss über Zusammenhänge, die auf den ersten Blick oft nicht erkennbar sind. Synonyme bei Stichworten können in dieser Liste gut erkannt werden.

Stichwortbasierter Bericht

Eine immer wiederkehrende Kritik ist die unübersichtliche oder unflexible Darstellung der Baumstruktur im Wiki. Viele Kollegen wünschen sich zur Verbesserung zusätzliche Unterseiten zur Gruppierung der Seiten. Leider ist es aber so, dass die Auslöser für diesen Wunsch sehr unterschiedlich sind. Eine Baumstruktur im Wiki kann aber nur ein einziges Ziel verfolgen und niemals diese Vielzahl von teilweise widersprüchlichen Wünschen erfüllen. Gerade Unterseiten für die Gruppierung wiederholen sich häufig. Wenn aber im Wiki jeder Seitentitel nur einmal verwendet werden kann, führt das zu Kunstgriffen wie

Nummerierung oder anderen schwer nachvollziehbaren Schreibweisen für die Unterseiten.

Eine Lösung für dieses Dilemma ist ein stichwortbasierter Bericht. Confluence stellt dazu einige Makros (z.B. "Content By Label", "Page Properties Report") zur Verfügung, die Seitentitel inklusive konfigurierbarer Zusatzinformationen (z.B. Name des Bereiches, Stichworte der Seite) 'live' auflistet, wenn bestimmte Stichworte enthalten sind. Ein Klick auf einen gelisteten Seitentitel öffnet direkt die Seite, während ein Klick auf eines der Stichworte eine entsprechende Stichwortsuche startet.

Stichwortbasierte Berichte sind ein sehr nützliches Werkzeug für die tägliche Arbeit der Autoren an den Inhalten im Wiki, weil der Bericht die Menge der zu bearbeitenden Seiten auf eine überschaubare Größe reduziert. Und Berichte geben den Nutzern die Möglichkeit, einen kleinen Teil im Wiki an ihre ganz persönlichen Bedürfnisse anzupassen – ein nicht zu unterschätzender Vorteil!

Fazit

Die Motivation für die Autoren, Stichworte zu verwenden, ergibt sich aus den vielfältigen Möglichkeiten, die Confluence für Recherchen bietet. Voraussetzung für zufriedenstellende Ergebnisse ist aber eine konsequente Pflege der Stichworte. Dabei helfen eine Reihe von einfachen Regeln.

- Verwende Stichworte nur im Singular.
- Vermeide Stichworte aus zusammengesetzten Worten, ersetze sie besser durch gebräuchliche Abkürzungen.
- Füge standardisierte Stichworte, die das Artefakt beschreiben, bereits durch die Seitenvorlage beim Anlegen einer Seite ein.
- Leite Stichworte bereits beim Anlegen der Seite aus dem Titel ab.
- Verwende Fachausdrücke und Begriffe aus dem Glossar im Text als Stichworte.
- Verwende die Kurzbezeichnung der Anforderung als Stichwort.

- Verwende die Kurzbezeichnung des Arbeitspaketes als Stichwort.

Diese Regeln können weitestgehend formal in Reviews überprüft werden. Durch Training und Disziplin können wir eine gute Qualität bei den Stichworten erhalten. Die Groß- und Kleinschreibung von Stichworten spielt keine Rolle. Trotzdem empfiehlt es sich, Stichworte ausschließlich in Kleinbuchstaben zu schreiben.

Das Stichwortverzeichnis als Wolke gibt einen sehr guten optischen Überblick über die wichtigsten Begriffe im Wiki. Es ist gleichzeitig auch eine Metrik für die fachliche Qualität der Inhalte. Die Begriffe sollten den interessierten Parteien oder anderen Lesern bekannt sein.

Das Stichwortverzeichnis als Liste hilft vor allem dabei, Duplikate aufgrund leicht abweichender Schreibweise schnell und sicher zu erkennen.

Seiten sicher löschen

Eine Produktdokumentation wird in Confluence über die gesamte Lebenszeit der Software gepflegt. Das kann bei einer erfolgreichen Software eine lange Zeit sein. Hygiene ist darum sehr wichtig. Wie Entwickler toten Code identifizieren und entfernen, so muss jeder Autor darauf achten, Inhalte zu entfernen, die nicht mehr gültig sind. Zur Erhaltung der Qualität muss er manchmal ganze Seiten löschen.

Die folgenden Standardfunktionen von Confluence werden zu einem einfachen Workflow für sicheres Löschen kombiniert.

- **Seite verschieben** – Eine Seite wird in Confluence verschoben, indem die übergeordnete Seite geändert wird. Ihr Inhalt bleibt unverändert.
- **Verwaiste Seiten** – Eine Seite ist verweist (engl. orphaned), wenn sie keine übergeordnete Seite hat. Sie ist für normale Nutzer nicht mehr sichtbar.
- **Seiten schützen** – Eine Seite kann beschränkt (engl. restricted) werden. Sie ist dann nur mehr für berechnigte Leser sichtbar.

Im Workflow für sicheres Löschen von Seiten nutzen wir die Möglichkeit, eine Seite "unsichtbar" zu machen. Verwaiste Seiten haben keine übergeordnete Seite und werden aus diesem Grund in der normalen Ansicht von Confluence nicht mehr angezeigt. Sie sind nicht mehr sichtbar. Zusätzlich nutzen wir Seitenbeschränkungen, um die für das Löschen vorgesehenen Seiten vor unbeabsichtigter und unerwünschter Veränderung zu schützen.

Zur Vorbereitung für das sichere Löschen muss eine sogenannte Bitte-Löschen-Seite angelegt, versteckt und beschränkt werden. Der Titel kann frei gewählt werden.

Die Bitte-Löschen-Seite wird vor den Leser versteckt, indem sie zu einer verwaisten Seite gemacht wird.

Geben Sie Bereich und Überschrift der neuen übergeordneten Seite und der untergeordneten Seiten für diese Seite an.


Neuer Bereich:



Neue übergeordnete

Seite:
Wenn Sie beginnen, einen Seitentitel einzugeben, wird eine Liste mit Vorschlägen angezeigt.

Bitte-Löschen-Seite verstecken

Die Bitte-Löschen-Seite wird auf die Gärtner beschränkt, damit sie kein Autor mehr verändern kann.

 Bearbeiten eingeschränkt ▾ Alle können die Seite anzeigen, nur einige können sie bearbeiten.

<input type="text" value="Geben Sie einen Benutzernamen bzw."/>	<input type="button" value="Kann bearbeiten."/>	<input type="button" value="Hinzufügen"/>
 Jeder	<input type="button" value="Kann anzeigen"/>	
 Robert Bruckbauer	<input type="button" value="Kann bearbeiten."/>	

Bitte-Löschen-Seite beschränken

Die nun unsichtbare und geschützte Bitte-Löschen-Seite kann nun verwendet werden, um dort alle zum Löschen vorgesehene Seiten als Unterseiten zu sammeln.

Sicheres Löschen bedeutet, eine Seite als Unterseite der Bitte-Löschen-Seite zu verschieben. Das bedeutet, die fraglichen Seiten werden nur vorläufig gelöscht, weil sie in der Bitte-Löschen-Seite weiterhin existieren.

Nur über die Verwaltungsfunktionen des Bereiches kann ein Gärtner die nun verwaisten zum Löschen vorgesehenen Seiten sehen. Für eine bestimmte, bei Lesern und Autoren bekannte Aufbewahrungszeit

(z.B. bis zum Abschluss des nächsten Sprints) lässt der Gärtner die Seiten dort liegen. Nach Ablauf der Frist löscht er die Unterseiten endgültig.

Fazit

Der beschriebene Workflow ist sehr einfach und kann mit Standardfunktionen von Confluence realisiert werden. Durch die Vererbung der Berechtigungen auf alle Unterseiten ist die zum Löschen vorgesehene Seite für keinen Leser sichtbar oder lesbar und für keinen Autor bearbeitbar. Greift ein Autor oder Leser über eine Verknüpfung auf eine vorläufig gelöschte Seite zu, schickt Confluence eine E-Mail an den letzten Autor der Seite mit der Bitte um Zugriff auf die Seite. Durch die E-Mail-Benachrichtigung erkennt der Autor der vorläufig gelöschten Seite veraltete Verknüpfungen und kann andere Autoren und Leser gezielt informieren, dass die Seite veraltet und demnächst gelöscht wird.