

Integration statt Brückenbau

(Link zum Artikel: <http://www.it-republik.de/dotnet/artikel/2956>)

Collaboration-Plattform als Werkzeugbank

Text: Uta Kapp, Jean Pierre Berchez

Softwareentwicklung ist ein komplexer Kommunikationsprozess mit vielen einzelnen Aufgaben. Es gibt Fehlerdatenbanken. Aufgabenkataloge haben so verschiedene Namen wie Backlog-Listen oder Anforderungs-Tracker. Sourcecode wird verändert und in einem Archiv abgelegt. Es gibt Prioritäten und Termine und alle gemeinsam stehen in Bezug zu einem Release. Nur eine integrierte Collaboration-Plattform kann den Bezug zwischen allen Teilen herstellen und damit einen Gesamtüberblick für den aktuellen Entwicklungsstand ermöglichen.

Die Zusammenarbeit an einem Softwareprojekt stellt hohe Anforderungen an die Kommunikation. Eine Collaboration-Plattform verwaltet alle Informationen für ein Projekt und stellt Kommunikationswege bereit. Sie ist Software zur Unterstützung von Softwareentwicklung. Zentral auf einem Server stellt sie alle Funktionalitäten für die Teamzusammenarbeit zur Verfügung, sodass alle Teammitglieder von überall auf dem Globus auf dieses gemeinsame Repository zugreifen können. Eine Collaboration-Plattform unterstützt den gesamten Applikationslebenszyklus, indem sie die Tools, die für jede Phase des Projekts benötigt werden, in eine einzige Plattform integriert. Das erlaubt eine vereinfachte Verwaltung und eine zentrale Wissensdatenbank:

- Einmalige Anmeldung
- Suche über alle Tools hinweg
- Zentrales Rollenmanagement
- Verlinkte Dokumente
- Dashboards mit Berichten und Audits über alle Phasen hinweg.

Werkzeuge allein erzeugen keine Kommunikation. Ein erfolgreicher Kommunikationsprozess ist dann abgeschlossen, wenn die richtige Botschaft vom Sender bei dem Empfänger angekommen ist. Als guter Kommunikator erkennen wir bei einer verbalen Unterhaltung an der Reaktion des Zuhörers, ob er das Gesagte verstanden hat. Bei einem nicht verbalen Informationsfluss, wie auf einer Collaboration-Plattform, brauchen wir auch explizite Rückmeldeverfahren, um die Reaktion des Lesers bei Eintreffen der Botschaft zu erhalten. Das erfordert eine ganz neue Kommunikationskultur. Zusätzlich zur technischen Plattform muss diese Kompetenz in einem Team gezielt aufgebaut werden.

Wissensmanagement ist das zentrale Thema bei einem umfangreichen Softwareprojekt. Wissen muss geteilt, Dokumente abgelegt, Fehler gemeldet und Sourcecode committet werden. Benachrichtigungen an Teammitglieder sind bei Ereignissen, wie z. B. Versionsrelease oder Fehlermeldung, notwendig. Eine integrierte Plattform verwaltet Wiki, Dokumentenmanager, Tracker, Forum und Sourcecode-Archiv. Sie verbindet Informationen auf der Plattform miteinander und stellt sie zueinander in Beziehung. Um die Transparenz für alle Projektbeteiligten hoch zu halten, bekommt jeder Betroffene die Gelegenheit, sich in

Selbstbedienung alle Informationen zu beschaffen. Ein ausgefeiltes Sicherheits- und Rollenverteilungssystem sorgt für den reibungslosen Zugriff aller Mitglieder.

Wenn alle Informationen zu einem Projekt in digitaler Form vorliegen, dann ist es möglich die Metriken zu erstellen, mit dem ein Projektfortschritt messbar ist. Solche Metriken beantworten zum Beispiel die Frage, wie viele gemeldete Bugs pro Woche bearbeitet wurden. Eine andere Frage könnte sein, wie viel Programmieraufwand in einer Woche geleistet wurde. Automatisch bereitgestellte Reports beantworten diese Fragen. Die nahtlose Anbindung von Entwicklungsumgebungen (IDE), z. B. Visual Studio oder Eclipse, an eine integrierte Plattform ist wichtig, weil viele Informationen direkt bei der Kodierung der Software anfallen. Eine gute Anbindung mithilfe von Plug-ins sorgt dafür, dass Informationen von Programmierern mit wenig Aufwand bereitgestellt werden. Die Nachvollziehbarkeit (traceability) in Softwareprojekten wird mit solch einer integrierten Plattform sichergestellt. Welcher Bug wurde in welchem Code zu welchem Release repariert? Wie war die Konfiguration eines Release? Welche Artefakte sind Bestandteil des Builds? Ein Build lässt sich so, zu einem späteren Zeitpunkt, noch einmal exakt nachbauen.

Wiki

Ein Projektwiki ist die Wissensdatenbank eines Projekts. Hier kann unbürokratisch Wissen von Entwicklern, Benutzern, Testern und Projektmanagern bereitgestellt werden. Hier werden auch Reports abgelegt. Es ist eine lebendige Dokumentation eines Projekts. Wenn die Seiten in einem Wiki als Gemeinschaftsarbeit betrieben werden, dann wird daraus eher ein Biotop, das sich selbst reguliert, als ein Verwaltungstool. Informationen werden nur bei Bedarf eingestellt und nicht auf Vorrat produziert. Ein gut geführtes Wiki hat wertvollere Informationen für ein Team zu bieten als die beste Hochglanzdokumentation. Natürlich setzt das Disziplin und Verantwortungsbereitschaft im Team voraus.

Dokumentenmanagement

Ein integriertes Dokumentenmanagement steht für den Austausch von Dokumenten zur Verfügung. Das sind z. B. downloadbare Pakete oder Dokumentationen.

Fehlermeldungssysteme (Bug Tracker)

Bug-, Anforderungs- und Änderungs-Tracker sind Teil eines jeden Softwareentwicklungsprozesses. Sie verfolgen die Listen mit den Aufgaben und Arbeitsschritten eines Projekts. Bug Tracker erlauben eine direkte Fehlerrückmeldung der Benutzer an das Entwicklerteam. Hier kann auch nachvollzogen werden, in welcher Sourcecode-Datei ein Bug oder eine Anforderung realisiert und implementiert wurde.

Sourcecode Configuration Management

Ein Sourcecode-Archiv, das nahtlos in eine Collaboration-Plattform integriert ist, hat sowohl den Bezug zu Releases, Meilensteinen als auch zu konkreten Build-Versionen. So ist zu jedem Zeitpunkt die Nachvollziehbarkeit jeder veröffentlichten Release gegeben. Nur eine bestimmte Konfiguration aus Versionen eines Sourcecodes, Bibliotheken und anderen Artefakten erzeugt eine lauffähige Version.

Versionsreleases und Build-Management

Kontinuierliche Integration aller hinzukommenden Features kann nur einfach und automatisch erfolgen, wenn die Builds automatisch täglich erstellt werden und mit automatischen Tests ablaufen. Eine Collaboration-Plattform steuert diesen Prozess, liefert alle Artefakte und speichert das Ergebnis.

Metriken

Metriken geben einen Einblick in den Projektstand und -fortschritt. Wenn ein gleich bleibendes Tempo in der Entwicklung angestrebt ist, dann muss das auch messbar sein. Das ist nur möglich, wenn alle Daten digital vorliegen. Integrierte Plattformen sind die beste Voraussetzung für solche Metriken.

Uta Kapp arbeitet als freiberufliche IT-Beraterin und systemischer Coach. Mit einer Kombination aus Fachberatung und Prozessberatung für Softwareprojekte, hilft sie Entwicklungsteams bei der Bewältigung der ständig steigenden Komplexität. Hier kommen die agile Softwareentwicklungsmethode Scrum und Organisationsaufstellungen zum Einsatz. Der Einsatz von Collaboration-Plattformen ist ein weiterer Schwerpunkt.

Jean Pierre Berchez ist Geschäftsführer der [HLSC UG](#) und beschäftigt sich seit mehr als 15 Jahren mit den Themen Projektmanagement, Software Engineering und objektorientierte Softwareentwicklung. In den letzten Jahren liegt sein Interesse insbesondere auf den Themengebieten agile Entwicklung mit Schwerpunkt Scrum sowie "Collaborative" Software Development. Neben seiner Tätigkeit als Geschäftsführer ist Jean Pierre Berchez auch als Lehrbeauftragter an den BAs Stuttgart und Heidenheim sowie an der Hochschule Liechtenstein für die Themen "Anforderungsmanagement", "Scrum" und "Collaborative Software Development" tätig. Er organisiert unter anderem den Community-Event "[Scrum-Day](#)" in Deutschland.

Original Artikel auf:

<http://it-republik.de/dotnet/artikel/Integration-statt-Brueckenbau-2957.html>

andere Artikel dieser Serie

- [Kollaborieren mit Plattform](#)
- [Integrationen und Schnittstellen – "Wie es Euch gefällt"](#)
- [Bughunting](#)
- [Was macht ein Open Source-Projekt erfolgreich?](#)
- [Metriken, Statistiken, Dashboards](#)
- [Kommunikation zwischen Entwicklern, Benutzern und Management](#)
- [Wikinomics](#)
- [Kollabieren oder Kollaborieren](#)
- ["Projekt Gold" oder: Der Weg führt zum Ziel](#)