

4. Evaluation

Inhalt

| | |
|---|----|
| 4.1. Beschreibung der Metriken | 37 |
| 4.2. Abbildung der Klassifikation auf den Informationsbedürfnisbaum | 38 |
| 4.3. Ergebnis der Evaluation | 42 |

In diesem Kapitel wird die Evaluierung des Papierprototypen (siehe Abschnitt 3.6) vorgestellt. Zur Evaluierung wurden Metriken zur Qualität von Open Source-Entwicklungsprozessen aus [Sch09] in das Metrik-Wiki eingeordnet. Die Metriken vermessen zum großen Teil Bugzilla Einträge, um Aussagen über die Qualität des Software-Projekts zu treffen. Es sollte vor allem evaluiert werden, wie gut sich Metriken in den Informationsbedürfnisbaum einsortieren lassen. Es wurden aber auch Beschreibungen der Metriken in dem Prototypen erstellt. Schaefer [Sch09] klassifiziert die Metriken aus zwei Sichten: der dokumentbezogenen Sicht und der gruppenbezogenen Sicht. In beiden Fällen werden die Metriken nach Qualitätsmerkmalen klassifiziert. Beide Sichten haben drei Hauptmerkmale, die sich in weitere Qualitätsmerkmale differenzieren. Im Folgenden wird erläutert, wie gut Metriken sich in dem Wiki beschreiben ließen. Danach wird beschrieben wie acht Qualitätsmerkmale in Informationsbedürfnisse übersetzt wurden und diese in einem Informationsbedürfnisbaum kategorisiert wurden. Am Ende dieses Kapitels wird ein Fazit zu der Evaluation gezogen.

4.1. Beschreibung der Metriken

Abbildung 4.1 zeigt die Beschreibung einer Metrik aus [Sch09] in dem Prototypen des Metrik-Wiki. Da es bei der Beschreibung der Metrik in dem Wiki wenig Beschränkungen gibt, war das Eintragen der Metriken relativ einfach. Die Metriken wurden durch die Punkte Definition, Beschreibung, Klassifikation (dokumentbezogene Sicht), Klassifikation (gruppenbezogene Sicht) und Bedeutung beschrieben. Bis auf die Klassifikation konnten die Punkte genauso in die Beschreibung der Metrik in dem Wiki übernommen werden. Auch bei der Definition der Eigenschaften der Metrik gab es keine Probleme. Die vermessene Entität, Skala, Reichweite und der Typ der Metrik gingen aus der Beschreibung der Metrik hervor, auch wenn sie dort nicht explizit angegeben wurden.

The Metric Wiki Advanced Search

Anzahl der gelösten Bugs ohne zugewiesenen Target Milestone
Edit History

Home

About

Add Measure

Last Changes

Open Requests

Anzahl der gelösten Bugs ohne zugewiesenen Target Milestone

Definition

Betrachtet wird der Anteil aller in einem Zeitintervall gelösten Fälle, die zum Zeitpunkt des Statusübergangs nach RESOLVED keinen gültigen Eintrag im Feld Target Milestone besaßen, sofern das Feld Resolution zu diesem Zeitpunkt den Wert FIXED besaß. Gültige Einträge sind solche, die einen tatsächlichen Milestone bezeichnen; Einträge wie Future gelten nicht als tatsächlicher Milestone. Zu berücksichtigen ist, wie bei allen Metriken die sich mit gelösten Bugs befassen, daß in den Eclipse-Projekten zeitweise die Resolutions 'LATER' bzw. 'REMIND' im Zusammenhang mit dem Zustand RESOLVED verwendet wurden, um das Aufschieben von Fehlern zu kennzeichnen, ohne daß diese tatsächlich behoben wurden. Zielvorgabe dieser Metrik ist eine Minimierung des Metrikwerts.

Beschreibung

Die Beschreibung des Eclipse-Prozesses [1] fordert eine Milestoneausgerichtete Entwicklung. Darüberhinaus wird eine größtmögliche Transparenz [2] gegenüber den Benutzern und Entwickler-Kollegen gefordert. Das Lösen von Bugs ohne die Information, in welchem Milestone der Fehler behoben ist oder sein wird verletzt beide Vorgaben.

Bedeutung

Ein ungenügender Metrikwert dieser Metrik kann auf mangelhafte Berücksichtigung der Prozessvorgaben hinsichtlich der Verwaltung von Milestones sowie der Schaffung einer größtmöglichen Transparenz hindeuten.

Referenzen

[1] GAMMA, ERICH und JOHN WIEGAND: The Eclipse Way - Processes that Adapt. EclipseCon, 2005.
[2] The Eclipse Development Process, Revision 2.4. http://www.eclipse.org/projects/dev_process/development_process.php, August 2008.

| Properties | |
|--------------------|--|
| Entity: | Bugzilla Einträge von Open-Source Eclipse Projekten |
| Scale: | intervall |
| Unit: | gelöste Fälle |
| Range: | 0 - inf |
| Type: | base |
| Information Needs: | Wie gut werden die Milestones eingehalten? Wie transparent ist der Funktionsumfang eines Milestone? Wie gut werden Anwender- und Entwicklergemeinde in den Entwicklungsprozess eingebunden? Wie ist die Transparenz? |

Abbildung 4.1.: Beschreibung der Metrik „Anzahl der gelösten Bugs ohne zugewiesenen Target Milestone“ in dem Metrik-Wiki

4.2. Abbildung der Klassifikation auf den Informationsbedürfnisbaum

In [Sch09] wurden die Metriken durch 14 (aus der dokumentbezogenen Sicht) beziehungsweise 7 (aus der gruppenbezogenen Sicht) Qualitätsmerkmale klassifiziert. Im Folgenden wird erläutert, wie acht dieser Qualitätsmerkmale auf Informationsbedürfnisse in dem Wiki abgebildet wurden. Da die Metriken nach Qualitätsmerkmalen geordnet wurden, stammen alle Informationsbedürfnisse von dem Informationsbedürfnis „*Wie ist die Qualität?*“ ab.

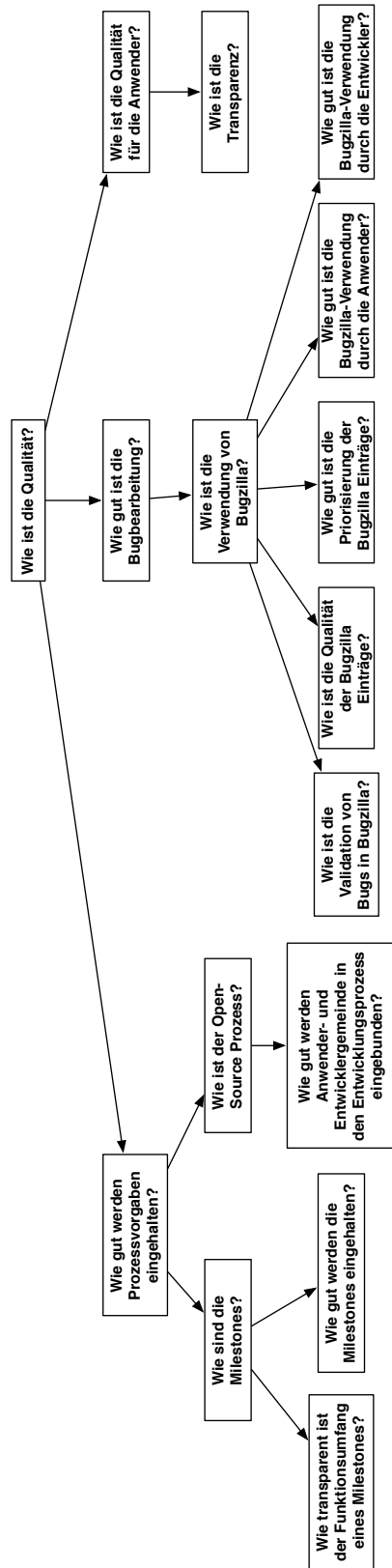


Abbildung 4.2.: Informationsbedürfnisbaum mit Informationsbedürfnissen aus der Evaluation

4.2.1. Qualitätsmerkmale aus dokumentbezogener Sicht

Milestones First

Das Qualitätsmerkmal *Milestones First* gehört zu dem Obermerkmal *Prozesseinhaltung*. Es wurde den beiden Informationsbedürfnisse „*Wie gut werden die Meilensteine eingehalten?*“ und „*Wie transparent ist der Funktionsumfang eines Meilensteines?*“ zugeordnet. Beide Informationsbedürfnisse sind im Informationsbedürfnisbaum unter „*Wie ist die Qualität?*“ → „*Wie gut werden Prozessvorgaben eingehalten?*“ → „*Wie sind die Meilensteine?*“ eingeordnet. Da beide Informationsbedürfnisse Fragen zu Meilensteinen stellen, haben beide das Informationsbedürfnis „*Wie sind die Meilensteine?*“ als Elternknoten. Dieser hat als Elternknoten „*Wie gut werden Prozessvorgaben eingehalten?*“, da das Qualitätsmerkmal *Milestones First* das Obermerkmal *Prozesseinhaltung* hat. *Milestones First* wird in dem Wiki durch die zwei Informationsbedürfnisse „*Wie gut werden die Meilensteine eingehalten?*“ und „*Wie transparent ist der Funktionsumfang eines Meilensteines?*“ repräsentiert, da dieses Qualitätsmerkmal sich nach [Sch09] genau aus diesen Fragen ergibt.

Community Involvement

Das Qualitätsmerkmal *Community Involvement* wurde dem Informationsbedürfnis „*Wie gut werden Anwender- und Entwicklergemeinde in den Entwicklungsprozess eingebunden?*“ zugeordnet. Dieses Informationsbedürfnis wurde unter „*Wie ist die Qualität?*“ → „*Wie gut werden Prozessvorgaben eingehalten?*“ → „*Wie ist der Open-Source Prozess?*“ eingeordnet. „*Wie ist der Open-Source Prozess?*“ ist Elternknoten des Informationsbedürfnisses, da Einbindung von Anwender- und Entwicklergemeinde vor allem in Open-Source Projekten relevant ist. Da *Community Involvement*, genau wie *Milestones First* zum Obermerkmal *Prozesseinhaltung* gehören, stammt das Informationsbedürfnis auch von „*Wie gut werden Prozessvorgaben eingehalten?*“ ab.

Validierung von Bugzilla Fehlerberichten

Das Qualitätsmerkmal *Validierung* ist eine Verfeinerung des Merkmals *Bugzilla-Verwendung*. Es wurde auf das Informationsbedürfnis „*Wie ist die Validation von Bugs in Bugzilla?*“ abgebildet. Dieses ist im Informationsbedürfnisbaum unter „*Wie ist die Qualität?*“ → „*Wie gut ist die Bugbearbeitung?*“ → „*Wie ist die Verwendung von Bugzilla?*“ zu finden. Da das Qualitätsmerkmal ein Merkmal für Bugzilla-Datenbanken ist, stammt das Informationsbedürfnis von „*Wie ist die Verwendung von Bugzilla?*“ ab. Dieses ist wiederum Kind von „*Wie gut ist die Bugbearbeitung?*“, da es dieses Informationsbedürfnis weiter verfeinert.

Qualität der Bugzilla Fehlerberichte

Das Qualitätsmerkmal *Qualität der Einträge*, eine Verfeinerung des Merkmals *Bugzilla-Verwendung*, ergibt sich aus dem Informationsbedürfnis „*Wie ist die Qualität der Bugzilla Einträge?*“. Dieses befindet sich im Informationsbedürfnisbaum unter „*Wie ist die Qualität*“ → „*Wie gut ist die Bugbearbeitung?*“ → „*Wie ist die Verwendung von Bugzilla?*“. Da das Informationsbedürfnis Bugzilla-Datenbanken betrifft, ist es Kind von „*Wie ist die Verwendung von Bugzilla?*“.

Priorisierung von Bugzilla Fehlerberichten

Das Qualitätsmerkmal *Priorisierung* besitzt als übergeordnetes Merkmal *Bugzilla-Verwendung*. Es wird durch das Informationsbedürfnis „*Wie gut ist die Priorisierung der Bugzilla Einträge?*“ repräsentiert. Dieses wurde unter „*Wie ist die Qualität*“ → „*Wie gut ist die Bugbearbeitung?*“ → „*Wie ist die Verwendung von Bugzilla?*“ eingeordnet, da es Bugzilla spezifisch ist.

4.2.2. Qualitätsmerkmale aus gruppenbezogener Sicht

Transparenz

Transparenz ist ein Qualitätsmerkmal für Anwender. Es wurde dem Informationsbedürfnis „*Wie ist die Transparenz für die Anwender?*“ zugeordnet. Dieses ist unter „*Wie ist die Qualität*“ → „*Wie ist die Qualität für die Anwender?*“ zu finden.

Das Informationsbedürfnis ist vor allem ein Informationsbedürfnis für Open-Source-Projekte, da in Open-Source-Projekten die Anwender in den Entwicklungsprozess mit einbezogen werden. Deshalb könnte ein Informationsbedürfnis „*Wie ist die Qualität für Anwender bei Open-Source Projekten?*“ eingeführt werden. Dieses würde Kind von „*Wie ist die Qualität für die Anwender?*“ sein und selber „*Wie ist die Transparenz für die Anwender?*“ als Kind besitzen.

Bugzilla-Verwendung durch die Anwender

Das Qualitätsmerkmal *Bugzilla-Verwendung durch die Anwender* wurde auf das Informationsbedürfnis „*Wie gut ist die Bugzilla-Verwendung durch die Anwender?*“ abgebildet. Dieses lässt sich unter „*Wie ist die Qualität*“ → „*Wie gut ist die Bugbearbeitung?*“ → „*Wie ist die Verwendung von Bugzilla?*“ finden. „*Wie gut ist die Bugzilla-Verwendung durch die Anwender?*“ ließ sich gut in den Informationsbedürfnisbaum einordnen und ist ein gutes Beispiel dafür, wie die Informationsbedürfnisse immer spezifischer werden, je tiefer sie im Baum sind.

Bugzilla-Verwendung durch die Entwickler

Analog zu *Bugzilla-Verwendung durch die Anwender* wurde das Qualitätsmerkmal *Bugzilla-Verwendung durch die Entwickler* auf das Informationsbedürfnis „*Wie gut ist die Bugzilla-Verwendung durch die Entwickler?*“ abgebildet. Dieses befindet sich im Informationsbedürfnisbaum unter „*Wie ist die Qualität?*“ → „*Wie gut ist die Bugbearbeitung?*“ → „*Wie ist die Verwendung von Bugzilla?*“.

4.3. Ergebnis der Evaluation

Die Beschreibung von Metriken in dem Metrik-Wiki stellte keine Probleme dar. Wie in Abschnitt 4.1 beschrieben, war es dank wenig Einschränkungen bei der Beschreibung von Metriken einfach, Metriken in das Wiki einzutragen.

Die Evaluation hat gezeigt, dass sich Qualitätsmerkmale relativ gut in Informationsbedürfnisse überführen lassen. Allerdings lassen sich Informationsbedürfnisse nicht immer eindeutig in den Informationsbedürfnisbaum einordnen. In [Sch09] wurden die Metriken aus zwei Sichten klassifiziert. In der Evaluation wurden für Merkmale aus beiden Sichten entsprechende Informationsbedürfnisse erstellt. Manche Informationsbedürfnisse, die aus gruppenbezogenen Qualitätsmerkmalen entstanden sind, könnten auch Elternknoten von Informationsbedürfnissen aus dokumentbezogener Sicht sein. So könnte zum Beispiel das Informationsbedürfnis „*Wie ist die Validation von Bugs in Bugzilla?*“ auch Kind von „*Wie gut ist die Bugzilla-Verwendung durch die Entwickler?*“ sein.

Es wurde schon vor der Evaluation damit gerechnet, dass sich Informationsbedürfnisse nicht eindeutig in den Informationsbedürfnisbaum einordnen lassen. Dies ist allerdings ein allgemeines Problem von Kategorienbäumen. Es wird davon ausgegangen, dass die Nutzer des Wikis den Baum nach und nach selbst optimieren. Ein Informationsbedürfnisbaum scheint die beste Möglichkeit zu sein Metriken zu kategorisieren, da komplexere Kategorisierungen wie Graphen zu kompliziert sind. Kategorienbäume sind eine weit verbreitete Methode der Kategorisierung und werden daher von den Benutzern vermutlich gut akzeptiert.