

Webbasiert und kollaborativ: Ein Requirements Editor auf Basis von ReqIF

Dennis van der Vlugt¹, Erik Kamsties² und Axel Terfloth¹

¹itemis AG, Am Brambusch 15, 44536 Lünen

²Fachhochschule Dortmund, Emil-Figge-Strasse 42, 44227 Dortmund

Motivation. Verteilte Softwareprojekte stehen vor den verschiedensten Herausforderungen. Dieser Beitrag untersucht im Bereich des Requirements Engineerings die Herausforderungen, die sich im *Datenaustausch* und der *Kollaboration* der Stakeholder ergeben und stellt ein Werkzeug vor, das diese Herausforderungen adressiert.

Eine kürzlich publizierte Übersicht über Requirements Management (RM) Werkzeuge zählt 111 Produkte¹. Diese Vielfalt ist u.a. durch unterschiedliche Vorgehensmodelle (agil vs. traditionell) begründet. Die Folge ist, dass in größeren Projekten oft unterschiedliche Werkzeuge zum Einsatz kommen. Das Requirements Interchange Format (ReqIF) gewährleistet, dass Anforderungen zwischen mehreren in einem Projekt beteiligten Partnern mit unterschiedlichen Werkzeugen verlustfrei ausgetauscht werden können. Beispielsweise unterstützt das kommerzielle Werkzeug DOORS, aber auch viele Open Source Werkzeuge wie ProR dieses Format.

Eine weitere Herausforderung ist die Festlegung der Zusammenarbeit der beteiligten Partner [3], u.a.:

- Sollen sie über aktuelle Inhalte und Stände informiert bleiben?
- Sollen von ihnen Zustimmungen oder Kommentare eingeholt werden?
- Sollen sie die an sie kommunizierten Inhalte ändern dürfen?
- Müssen die geänderten Daten wieder in andere Systeme zurück überführt werden können?

Fragestellung. Wir gehen in diesem Beitrag der Frage nach, wie ein webbasiertes Werkzeug zum Editieren von ReqIF aussehen kann und welche kollaborativen Features es benötigt.

Stand der Technik. Viele Werkzeuge, die im RE verwendet werden, bieten Möglichkeiten der Kollaboration. Eine erste grobe Einordnung kann in einer sogenannten *Raum-Zeit-Matrix* [1], S. 220, erfolgen, die aus zwei Dimensionen gebildet wird. (1) *Raum*: Können sich die Teammitglieder, die das Werk-

zeug benutzen, am gleichen Ort oder an unterschiedlichen Orten aufhalten? (2) *Zeit*: Kann Kommunikation zur gleichen Zeit (synchron) oder zu unterschiedlichen Zeiten (asynchron) stattfinden?

Es geben sich vier Kategorien. Ein Vertreter der Kategorie 1 (gleicher Ort, synchron) ist CREWSpace [2]. Die meisten datenbankbasierten RE Werkzeuge wie z.B. DOORS lassen sich in Kategorie 4 (verschiedene Zeiten, verschiedene Orte) einordnen. Eine weitergehende Unterscheidung kann anhand der unterstützten Methoden erfolgen, z.B. erlaubt es FlexiSketch Skizzen modellbasiert in Form von Freihand-Skizzen kollaborativ zu modellieren [4].

Lösungsansatz. Zentrales Ziel ist es, nicht ein Werkzeug *in Konkurrenz* zu den etablierten Werkzeugen zu positionieren, sondern als *Ergänzung* zur einer existierenden RM-Werkzeuglandschaft. Aus diesem Grund verwendet der Editor als *zentrales Datenmodell ReqIF*, anstatt ein eigenes Datenmodell zu nutzen und einen ReqIF-Import/-Export anzubieten, wie es gängige RM-Werkzeuge tun.

Hauptzielgruppe des Werkzeugs sind Stakeholder, die im RE-Prozess partizipieren, in dem sie z.B. Anforderungen editieren und kommentieren oder Attribute modifizieren und für die daher die Lizenzkosten für ein vollwertiges RM-Tool zu hoch wären.

Das Werkzeug unterstützt nach der Raum-Zeit-Matrix die Kategorien 2 und 4, d.h. (1) unabhängig vom Ort sollen Benutzer gemeinsam an einem ReqIF Dokument arbeiten können, (2) unabhängig von der Zeit sollen Benutzer die Möglichkeit haben, auf das ReqIF Dokument zuzugreifen um Ergänzungen etc. vorzunehmen und (3) unabhängig von Zeit und Ort über den Inhalt einer spezifizierten Anforderung zu diskutieren, indem sie Anforderungen kommentieren oder Textnachrichten austauschen.

Dem Benutzer soll der Zugang zu dem Werkzeug erleichtert werden, indem es als Webanwendung realisiert wird, so dass Installation und Konfiguration entfällt.

Aus diesen Überlegungen zum Ziel des Werkzeugs und der Zielgruppe ergibt sich auch eine deutlich reduzierte Feature-Menge (gegenüber einem herkömmli-

¹<http://makingofsoftware.com/resources/list-of-rm-tools>



Abbildung 1: Screenshot des ReqIF-Editors

chen RM-Werkzeug), die sich am ReqIF-Datenmodell orientiert:

1. Grundfunktionen des ReqIF-Editors:
 - (a) ReqIF Dokument öffnen, navigieren, bearbeiten, speichern, exportieren (z.B. nach JSON)
 - (b) ReqIF Dokument filtern (nach SpecObjects, nach Spezifikationen)
 - (c) SpecObject hinzufügen, löschen, Attribute editieren
 - (d) Spezifikation reorganisieren, löschen
2. Kollaborative Funktionen des ReqIF-Editors:
 - (a) Benutzer registrieren, anmelden
 - (b) ReqIF Dokument von der Cloud aus öffnen, in die Cloud persistieren
 - (c) ReqIF Dokument für andere Benutzer freigeben, freigegebenes ReqIF Dokument öffnen
 - (d) ReqIF Dokument versionieren, versionierte ReqIF Dokumente öffnen
 - (e) **Synchrones Editieren**
 - (f) **Mitteilungen versenden**
 - (g) **Anforderung kommentieren**
 - (h) Änderungshistorie einer Anforderung öffnen, Anforderung aus der Änderungshistorie wiederherstellen

Die Abbildung 1 zeigt den Editor mit einem ReqIF-Dokument, das die Anforderungen an den Editor enthält. Links ist die Navigation dargestellt, zentral die Anforderungen und Attribute, rechts die Funktionen zur Kollaboration.

Der Vortrag wird vertieft auf kollaboratives RE und die Konzeption entsprechender Features für einen ReqIF-Requirements-Editor eingehen.

Literatur

- [1] Jan Leimeister. *Collaboration Engineering - IT-gestützte Zusammenarbeitsprozesse systematisch entwickeln und durchführen*. Springer, 2014.
- [2] Rainer Lutz, Sascha Schäfer, and Stephan Diehl. Using mobile devices for collaborative requirements engineering. In *IEEE/ACM International Conference on Automated Software Engineering*, pages 298–301. ACM, 2012.
- [3] Andreas Lux. Requirements Interchange Format (ReqIF) - kollaborative Anforderungsanalyse in verteilten Projekten.
- [4] D. Wüest, N. Seyff, and M. Glinz. Sketching and notation creation with flexisketch team: Evaluating a new means for collaborative requirements elicitation. In *2015 IEEE 23rd International Requirements Engineering Conference (RE)*, pages 186–195, 2015.