

Requirements Engineering in IT-Ökosystemen mit Hilfe von Archetypen

Leif Singer, Eric Knauss, Kurt Schneider

Fachgebiet Software Engineering
Leibniz Universität Hannover
Welfengarten 1, 30167 Hannover
{leif.singer|eric.knauss|kurt.schneider}@inf.uni-hannover.de

IT-Ökosysteme sind Systeme aus IT-getriebenen Geräten, Subsystemen und Personen, die als *Akteure* miteinander mittels der *Infrastruktur* des IT-Ökosystems interagieren. Hierbei genießt jeder Akteur des IT-Ökosystems eine beschränkte Autonomie. Er kann zeitweilig ohne direkte Weisung Entscheidungen fällen, ist aber der Kontrolle einer übergeordneten Instanz unterworfen. Als Beispiel kann man sich eine „SmartCity“ vorstellen, in der Verwaltung, Bürger, Unternehmen und Infrastruktur (wie bspw. Ampelanlagen) mit Hilfe von vernetzten Geräten interagieren und Dienste in Anspruch nehmen.

In diesen Systemen ist ein breites Spektrum von Beteiligten vertreten, deren Interessen bei der Software-Entwicklung in derartigen Systemen identifiziert und gegeneinander abgewogen werden müssen. Es ist oftmals unklar, welche Eigenschaften der Beteiligten zu erwarten sind und wie sich diese auf die geplanten Requirements-Engineering-Aktivitäten auswirken werden. Diese Unsicherheit erschwert die Erhebung und Validierung von Anforderungen in solch einer Umgebung. Um diesem Problem zu begegnen, schlägt dieses Positionspapier eine Klassifizierung der möglichen Beteiligten in Archetypen vor, die die Chancen und Risiken dieser Beteiligten im Umgang mit Anforderungen durch Überzeichnung herausstellen. Mit diesem Hilfsmittel können sich Requirements Engineers ein schnelles erstes Bild der in IT-Ökosystemen zu erwartenden Herausforderungen machen. Auf Basis dieser Einschätzung können dann spezialisierte Herangehensweisen und Werkzeuge identifiziert werden, die die jeweiligen zu erwartenden Risiken der Archetypen abschwächen und ihre Vorzüge nutzen können. Dies ermöglicht ein effizienteres und effektiveres Requirements Engineering im Rahmen der besonderen Anforderungen von IT-Ökosystemen.

Ein Beispiel für einen solchen Archetyp ist der *Unwissentliche Nutzer*, der sich seinen Interaktionen mit dem IT-Ökosystem nicht bewusst ist und dementsprechend auch keinerlei Interesse an oder Zeit für Requirements-Aktivitäten hat. Ein Autofahrer etwa, dessen Navigationsgerät sich mit der Verkehrsleitzentrale abstimmt, sollte kein Interesse daran haben müssen, Anforderungen aktiv mitzuteilen. Die Überzeichnung in den vorgeschlagenen Archetypen versucht, Requirements Engineers diese Eigenschaften bewusster zu machen, damit diese besser berücksichtigt werden können – etwa indem das Navigationssystem nach einer Fahrt mittels dreier verschiedener Schaltflächen in Form von Gesichtern (*fröhlich*, *neutral*, *traurig*) um eine leichtgewichtige Bewertung bittet.