

4 Methoden und Werkzeuge

4.1 Use-Case-basiertes Vorgehen zur Auswahl von Social Software

Sebastian Behrendt, Matheus Hauder, Marin Zec, Sascha Roth, Alexander Richter

4.1.1 Ziel und Motivation

Wie im Beitrag von Ehms/Richter erläutert, zeichnet sich Social Software insbesondere durch ihre Nutzungsoffenheit aus: Die Software selbst lässt die Art und Weise der späteren Nutzung größtenteils offen und ist nicht an ein typisches Nutzungsszenario gebunden. Hier unterscheidet sich Social Software auch bei der Softwareauswahl von traditionellen betrieblichen Anwendungssystemen (wie z. B. im Bereich Enterprise Resource Planning (ERP)), denen bereits in der Entwicklungsphase klare Strukturen, Prozesse und vorher vorgegebene bzw. definierte Nutzungsszenarien zugrunde liegen. Eine Bestandserfassungsmaske in einem ERP-System bspw. dient eben nur genau dem Zweck, Waren zu erfassen. In der Regel lässt sich sogar eine Funktion (z. B. manuelle Wareneingangskontrolle) genau einem konkreten Geschäftsprozess (z. B. Einkauf) zuordnen. Bei Social Software gibt es dagegen in der Regel verschiedene Funktionen zur Unterstützung derselben Arbeitspraktik und dieselbe Funktion kann meist zur Unterstützung unterschiedlicher Arbeitspraktiken eingesetzt werden. Das Freitextfeld in einem Microblog bspw. kann u. a. dazu verwendet werden, Probleme zu lösen, Ideen zu sammeln oder transparent zu kommunizieren. Aus diesem Grund stellt die Auswahl von Corporate Social Software die beteiligten Personen vor eine besondere Herausforderung, denn Nutzungsoffenheit führt dazu, dass klassische Software-Auswahl und -Einführungsprozesse nicht funktionieren. Insbesondere trifft dies auf die Anforderungsanalyse und Plattformauswahl zu. Grund dafür ist, dass man hier nicht funktionszentriert vorgehen kann, denn ein und derselbe Zweck sind meist mit komplett unterschiedlichen Funktionen erreichbar. Aus diesem Grund wurde ein Vorgehen auf höherer Ebene – siehe aperto-Ebenenmodell – auf der Basis von Use-Cases entwickelt, welches im vorliegenden Beitrag näher erläutert wird.

4.1.2 Anforderungen an Software im Unternehmensumfeld

Grundlage jeder Werkzeugauswahl ist die Ableitung von Unternehmensanforderungen. Um diese Anforderungen an Softwareprodukte zu strukturieren, existieren verschiedene Modelle. Dazu zählen etwa FURPS+ (Grady et al. 1987) und die internationale Norm ISO/IEC 25000, die wir im Folgenden kurz erläutern. FURPS steht als Akronym für die vermeintlich wichtigsten Kriterien für Softwarequalität. Das „+“ wurde im Laufe der Zeit hinzugefügt, um weitere Anforderungsaspekte, bspw. in Bezug auf Design, Implementierung, Schnittstellen oder physikalische Bedingungen, zu berücksichtigen.