Quantifizierung nicht-funktionaler Anforderungen

JURISTISCHES IT-PROJEKTMANAGEMENT WS1617
DOZENT: DR. FRANK SARRE
LMU MÜNCHEN
ZHENHAO LI

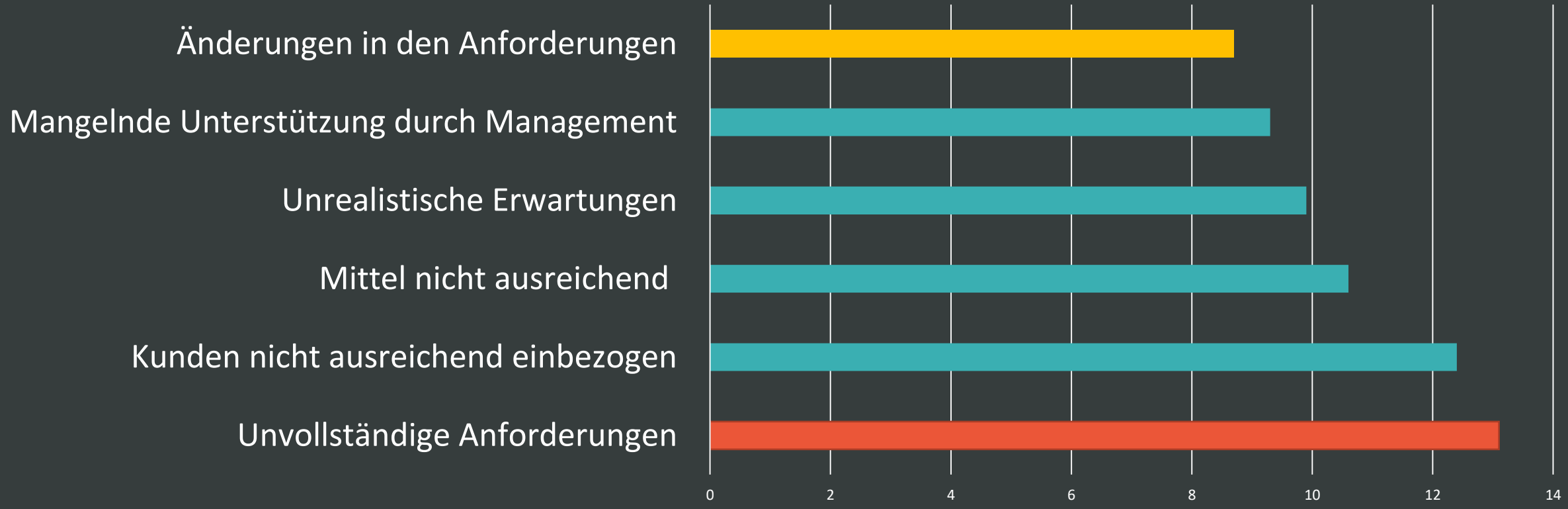
Agenda

- Einordnung des Themas
- Motivation
- Quantifizierung
- Nicht-funktionale Anforderungen
- Anforderungsanalyse
- Maße und Metriken
- Fazit

Einordnung des Themas

Motivation

Gründe woran IT-Projekte scheitern [1]



[1] Standish Group : *The CHAOS Report.*, 1994.

Zu lösende Probleme

- „Typische Situation: chaotische Projektdurchführung => mäßiger Qualität“ [2]
- „IT-Verträge und Pflichtenhefte sind oft von mäßiger Qualität, insbesondere unvollständig, zu grob und zu wenig praxistauglich. „ [2]
- „Softwaresysteme sind nur sehr aufwendig bezüglich ihrer Qualität zu beurteilen!“ [2]

[2] Sarre, Frank: Juristisches IT-Projektmanagement Skript am 25.10.2016

Werkvertrag, Abnahme, Mittlerer Ausführungsstandard

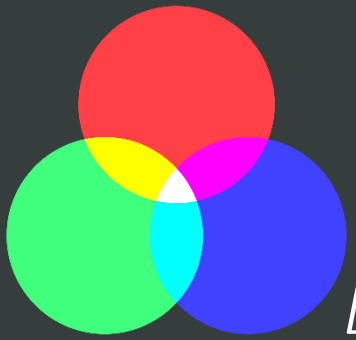
- Die meisten IT-Projekte: Individual Entwicklung bzw. Weiterentwicklung
- => Werkvertrag § 631 BGB
  - => AG muss die hergestellte Software abnehmen § 640 BGB
  - => AN will die Vergütung dafür bekommen

- Pflichtenheft nicht vorhanden bzw. nicht vollständig
  - *„Ist ein Ergebnis geschuldet, das unter Berücksichtigung des vertraglichen Zwecks des Programms dem Stand der Technik bei einem mittleren Ausführungsstandard entspricht“ (vgl. BGH, 16.12.2003 - X ZR 129/01) [3]*
  - => sehr offen formuliert, schwer zu beurteilen, aufwendig für beide Vertragsparteien

Quantifizierung

- Maßnahmen => Eigenschaften eines Objektes messbar machen
- Beispiel:
- Aussage: „Das Auto ist rot“

[4]



[5]

- Wie rot ist das Auto? Wie kann man dieses Rot des Autos reproduzieren?

- Maßnahme: Farbmodelle wie RGB, z.B. RGB (255, 0, 0) für ein bestimmtes Rot
- => Eindeutig, **reproduzierbar**

Nicht-funktionale Anforderungen (NFA)

- Funktionale Anforderungen (FA):
  - Was das System leistet => die Funktionalitäten

- Nicht-funktionale Anforderungen:
  - In welche Qualität das System etwas leistet
  - Keine eindeutige Auflistung, aber gewisse Standards, z.B. ISO IEC 25010 [6]
  - Aber auch kulturelle, politische und rechtliche Anforderungen

Nicht-funktionale Anforderungen

[7]

[7] http://3.bp.blogspot.com/-vGQi96wpQnc/UcN-b_drUHI/AAAAAAAABgg/dbN6DST2Udk/s1600/ISO-25010-2.jpg

NFA– antiproportionale Abhängigkeiten

- Sicherheit vs. Benutzbarkeit
  - 2 Schritte Authentifizierung
  - Lange Passwörter aus mehreren Buchstaben, Zahlen und Sonderzeichen
  - Jede Woche ein neues Passwort
- Effizienz vs. Wartbarkeit
  - Eingebetteter Assembler Code in einer höheren Programmiersprache Umgebung
- => Priorisierung der Anforderungen ebenfalls sehr wichtig
  - Sollte auch vertraglich festgehalten werden, damit die Umsetzung machbar ist

Anforderungsanalyse

- Anforderungen sammeln und definieren
- Wichtiger Schritt:
  - Ermittlung der Anforderungen verschiedener Stakeholdern
- Herausforderungen:
  - AG stammt aus einem anderen Fachbereich => eigene Fachsprache
  - Fokus meist auf gewünschte Funktionalitäten
  - => Nicht-funktionalen Anforderungen nicht explizit erwähnt

Implizite Anforderungen

- AG: „Die zu entwickelnde Anwendung wird von Tausenden Nutzer gleichzeitig verwendet“
- => Skalierbarkeit könnte wichtig sein

- AG: „Zusätzlich zum Desktop Programm planen wir auch in naher Zukunft die Einführung mobiler Apps für unsere Kunden“
- => Erweiterbarkeit sowie Übertragbarkeit könnten wichtig sein

- Daher: stets Nachfragen und vertraglich festlegen, um Streitfälle vorzubeugen

Maße

- AG muss die hergestellte Software abnehmen § 640 BGB
- => Überprüfung auf Vertragsmäßigkeit
- Nur möglich, wenn quantitativ konkret definiert

- => Maße: messbare Größe

- Neben Qualitätsanforderungen auch organisatorische oder technische Anforderungen

Maße - Beispiele

- Technische Anforderungen (z.B. aus organisatorischen Gründen wie Lizenzen):
  - Das System muss mit der Programmiersprache Java 6 entwickelt werden und muss auf dem IBM WebSphere Application Server 7 laufen.
  - Die Daten des Systems müssen auf dem Datenbankmanagementsystem Oracle 11g gespeichert werden.
- Ergonomische Anforderungen (Benutzbarkeit):
  - Dialoge müssen gem. EN ISO 9241-110 gestaltet werden.
- Effizienz:
  - Das System muss jede Anfrage des Benutzers innerhalb von 5 Sekunden ausführen. Der Speicherbedarf darf 16 GB nicht übersteigen.

Maße

- Wartbarkeit
- Erweiterbarkeit
- etc…

- Welche messbare Größe besitzen diese?

- => Metriken

Metriken

Die IEEE Standard 1061 definiert Softwaremetrik wie folgt:

- *„Eine Softwarequalitätsmetrik ist eine Funktion, die eine Software-Einheit in einen Zahlenwert abbildet, welcher als Erfüllungsgrad einer Qualitätseigenschaft der Software-Einheit interpretierbar ist. “*

  – IEEE Standard 1061, 1998 [8]

- Software in Zahlenwert
- => Bewerten und vergleichen

Metriken - Beispiele

- komplexe Eigenschaften lassen sich zu gewissem Grad ebenfalls messen
- Wartbarkeit (Testbarkeit):
  - McCabe-Metrik: [9]
  - Anzahl der IF Abfragen, je höher desto schlechter wartbar
- Erweiterbarkeit, Übertragbarkeit:
  - Schichtenarchitektur
  - Kopplung, Kohäsion bewerten (Datenflüsse zählen)

Fazit

- Nicht vollständige Anforderungen ist der häufigste Grund, woran IT-Projekte scheitern.
- Insbesondere werden nicht-funktionale Anforderungen oft vergessen
- => Gute Anforderungsanalyse ist sehr wichtig für den Erfolg eines IT-Projekts
- Alle Anforderungen priorisieren und quantitativ in Vertragsbestandteilen wie Pflichtenheft festlegen
- Bei komplexen Eigenschaften Metriken anwenden
- => AN kann besser planen und vertragsgemäß umsetzen
- => AG kann besser überprüfen bei der Abnahme

Quellenverzeichnis

- [1] Standish Group : *The CHAOS Report.*, 1994.
- [2] Sarre, Frank: Juristisches IT-Projektmanagement Skript am 25.10.2016
- [3] Bundesgerichtshof: Urteil vom 16.12.2003 - X ZR 129/01
- [4] http://www.iconsplace.com/icons/preview/red/car-256.png
- [5] http://static1.squarespace.com/static/54dc7e81e4b09b3ddced5bcc/t/54fdbccae4b0297e0214aa37/1425915082361/Icon_RGB-01.png
- [6] ISO/IEC 25010:2011: Systems and software Quality Requirements and Evaluation (SQuaRE) -- System and software quality models., 2011.
- [7] http://3.bp.blogspot.com/-vGQi96wpQnc/UcN-b_drUHI/AAAAAAAABgg/dbN6DST2Udk/s1600/ISO-25010-2.jpg
- [8] IEEE 1061: Standard for a Software Quality Metrics Methodology., 1998.
- [9] MacCabe, Thomas J.: Structured testing. IEEE Computer Society Press, 1983.