

Use Case Points Aufwandsschätzung auf Basis unterschiedlicher Spezifikations-Formate

Stephan Frohnhoff, Karsten Kehler

sd&m AG, Berliner Str. 76, D-63065 Offenbach

frohnhoff@sdm.de

Zusammenfassung:

Die Use Case Point Methode (UCP-Methode) erlaubt in Software-Entwicklungsprojekten zu einem frühen Zeitpunkt eine einfache Schätzung von zu erwartenden Aufwänden. Basis für eine solche Schätzung sind in der industriellen Praxis häufig Grob-Spezifikationen unterschiedlichen Formats und unterschiedlicher Granularität. Entscheidend für den Erfolg der UCP-Methode und die Vergleichbarkeit der Ergebnisse ist vor allem, ob und wie es gelingt, die vorliegende Spezifikation auf Use Cases im Sinne der UCP-Methode abzubilden. Hierzu fasst der Artikel einen Vorschlag für einen Leitfaden zusammen, der aus der Spezifikations-Praxis des Softwarehauses sd&m abgeleitet wurde.

Schlüsselbegriffe

Projektschätzung, Top-Down-Schätzung, Aufwandsschätzung, Use Case Points, UCP, Spezifikation

1 Einführung

Die Use Case Point Methode (UCP-Methode) wurde bereits ausführlich in [1] beschrieben. Es handelt sich dabei um eine Top-Down-Schätzmethode, welche die funktionalen Anforderungen aus der Spezifikation in abzählbare Einheiten (Use Cases) klassifiziert und diesen gemäß ihrer geschätzten Komplexität Punkte (Use Case Points) zuordnet. Der geschätzte Projektaufwand ist dann proportional zu diesen Use Case Points. Ein wesentlicher Kritikpunkt an dieser Methode ist, dass Use Cases in unterschiedlicher Granularität beschrieben werden können und dies unmittelbar Einfluss auf das UCP Schätzergebnis haben könnte.

Auf Basis der in [1] beschriebenen Projekte wurde nun ein Leitfaden zum "Schneiden" von Use Cases für die Abschätzung des funktionalen Umfangs mittels der UCP Methode entwickelt. Dieses umfangreiche Dokument wird als Preprint [2] veröffentlicht werden. Der folgende Artikel enthält eine Zusammenfassung der wichtigsten Erkenntnisse. Ziel ist nicht, ein einheitliches Vorgehen bei der Beschreibung von Use Cases zu definieren.

Bezugs-Domäne sind Software-Entwicklungsprojekte für betriebliche Informationssysteme, welche vom Software-Haus sd&m AG als Dienstleister für Kunden unterschiedlicher Branchen in individueller Auftragsfertigung abgewickelt wurden. Die Aufwandsschätzungen setzen dabei auf unterschiedlichen Grobspezifikationen des Auftraggebers auf und folgen keinem einheitlichen Format; oft liegen UML-ähnliche Beschreibungen oder rein textuelle Spezifikationen vor.

2 Grundlegende Begriffe

Im Folgenden wird ausschließlich der Begriff *Use Case* verwendet, er gilt aber synonym auch für Anwendungsfall. Zur einheitlichen Definition der Komplexitätsstufe wird ein *Use Case* durch die Zahl seiner *Szenarien*, *Schritte* und *Dialoge* in einfach, mittel und komplex bewertet, eine Definition der Begriffe folgt in den nächsten Abschnitten:

Einfach: maximal 3 Szenarien, Schritte und Dialoge => 5 Punkte

Mittel: maximal 7 Szenarien, Schritte und Dialoge => 10 Punkte

Komplex: mindestens 8 Szenarien, Schritte oder Dialoge => 15 Punkte

Als ungewichtete UCP-Maßzahl (d.h. ohne Berücksichtigung der Komplexitätsfaktoren) wird dann die Punkte-Summe über alle so gewichteten Use Cases gebildet. Für eine detaillierte Darstellung wird auf [1] verwiesen.

2.1 Use Case

Wir legen folgende Definition zugrunde: Ein Use Case spezifiziert exakt eine Aktivität der systemunterstützenden Aktivitäten eines Geschäftsprozesses. Er beschreibt das Verhalten und die Interaktion eines Systems als Reaktion auf die zielgerichtete Anfrage oder Aktion eines Akteurs. Die Beschreibung des Use Cases erklärt sowohl das extern sichtbare als auch das detaillierte interne Systemverhalten in der Sprache und aus der Sicht der Anwender. Mit einem Use Case wird durch das System ein für den Anwender sinnvoller Dienst erbracht oder ein benutzbares Ergebnis erreicht.

Ganz entscheidend ist dabei der Hinweis, dass mit einem Use Case immer ein bestimmtes **fachliches Ziel** erreicht werden soll. Ohne das Erreichen dieses Zieles werden wir in der Regel nur von Anwendungsfunktionen oder Schritten eines Use Cases sprechen. Beispiele dafür sind alle funktional orientierten Beschreibungen oder auch dialogorientierte Beschreibungen. So ist z.B. das Verwenden eines Suchdialoges häufig kein eigener abgeschlossener Use Case, da das eigentliche fachliche Ziel mit dem Finden und Anzeigen der Treffermenge noch nicht erreicht ist, weil z.B. mit einem bestimmten Datensatz nach dem Suchen weitere fachliche Schritte durchgeführt werden.

2.2 Szenarien

Ein **Szenario** eines Use Cases ist der fachliche Ablauf von Aktionen der Aktoren und des Systems zum fehlerfreien Erfolgsfall, der das fachliche Ziel des Use Cases ist. Jeder Use Case hat in der Regel genau ein solches Erfolgszenario.

Neben dem Erfolgszenario können **Alternativszenarien** spezifiziert werden. Diese beschreiben separat fachliche Spezialfälle, z.B. beim Vorliegen besonderer Randbedingungen und Ausnahmefälle, auch diese beenden den Use Case somit fachlich erfolgreich.

2.3 Schritte

Entscheidend für die Gewichtung eines Use Cases ist die Anzahl der für die Erreichung des fachlichen Zieles erforderlichen Schritte. Auch dafür gibt es leider keine allgemein anerkannte Definition, häufig wird von Aktionen, Operationen bzw. Folge von Operationen oder Aktivitäten gesprochen. Systemseitige Aktionen werden auch als Anwendungsfunktionen bezeichnet, insbesondere wenn die Beschreibung bereits eine gewisse funktionale Dekomposition der Use Cases enthält.

Ein **Schritt** im Ablauf eines Use Cases ist ein in sich geschlossener fachlicher Teil des Use Cases, der vom folgenden Schritt und davorliegenden Schritt eindeutig getrennt ist durch z.B. den Wechsel des Aktors oder der verarbeitenden "Schicht" (z.B. Eingabe im Dialog durch den Nutzer => Verarbeitung der Eingabe am Server => Anzeige des Ergebnisses an der Oberfläche), durch Erzeugen eines definierten (Zwischen-) Ergebnisses (z.B. Erzeugen von Druckdokumenten) oder durch Aufspalten eines neuen Szenarios.

Entscheidend für die Use Case Modellierung ist hierbei immer die Betrachtung der obersten Ebene der Zerlegung in Schritte aus Sicht des Use Cases. Jede weitere hierarchische Verfeinerung ist für die Schätzung nach UCP dann nicht mehr relevant. Werden also z.B. Aktivitäten in einer Beschreibung eines Use Cases genannt, die wiederum aus Operationen bestehen, so werden hier nur die Aktivitäten gezählt. Bei der Zählung berücksichtigt werden müssen

- die Anzahl aller Schritte in allen Szenarien; es handelt sich also um eine Aufsummierung über alle Szenarien, wobei Schritte, die in mehreren Szenarien verwendet werden, nur einmal in die Zählung eingehen
- gleichwertig sowohl Benutzeraktionen (-schritte) als auch systemseitige Aktionen

Typische **Beispiele** für Schritte sind die Eingabe eines oder mehrerer Werte in einen Dialog (ohne dass dazwischen ein Server-Roundtrip erfolgt), Aufrufe von Anwendungsfunktionen, Server-Transaktionen, Durchführung fachlicher Prüfun-

gen, Erzeugung von Ausgaben. Triviale Auswahlvorgänge durch einen Aktor aus einer Anzeige heraus werden nicht als eigener fachlicher Schritt gezählt (z.B. „Adresse auswählen“), es können aber je nach Auswahl neue Szenarien entstehen.

Eine Besonderheit bei den Schritten stellen fachlich sehr komplexe Abläufe dar, die nach obigen Kriterien in einem einzigen Schritt abgearbeitet werden. Diese würden mit der genannten Definition potentiell unterschätzt werden. Dazu zählen z.B. in einem einzigen Schritt ausgeführte Prüfungen, danach komplexe Berechnungen und evtl. noch die Erzeugung/Veränderung einer größeren Anzahl von Entitäten. In diese Kategorie fallen ebenso umfangreiche Regelwerke oder auch Funktions- und Entscheidungsmatrizen. In solchen Fällen schlagen wir eine Vergleichsgewichtung gegenüber anderen ("klassischen") Use Cases aus dem Projekt vor mit dem Ziel einer Zerlegung in einzelne kleinere Schritte, die fachlich abgrenzbare Teilergebnisse produzieren.

2.4 Dialoge

Als weiterer Parameter geht die Anzahl der unterschiedlichen Dialoge eines Use Cases in die Schätzung ein. Unter einem **Dialog** wird für die UCP Schätzung in einem erweiterten Sinn jede Interaktionsschnittstelle verstanden. Dialoge werden wie folgt gezählt:

- Jeder Reiter bzw. jede Maske eines Dialoges (mit signifikanten fachlichen Unterschieden) wird als eigener Dialog gezählt,
- Jeder Frame einer Webseite (mit signifikanten Steuerelementen) wird als eigener Dialog gezählt,
- Triviale Pop-Up-Meldungen, Bestätigungen und Menüs werden nicht gezählt

Neben den klassischen Dialogen werden aber auch noch folgende Interaktionsschnittstellen zu den Dialogen hinzu gezählt:

- Komplexe Schnittstellen, die Nachbarsystemen (zusätzlich zur Verarbeitung) zur Verfügung gestellt werden
- Druckstücke bzw. sonstige speziell erzeugte formatierte Ausgaben

3 Erfahrungen für den "Schnitt" von Use Cases

In den folgenden Abschnitten werden Regeln für das "Schneiden" und damit Strukturieren von Use Cases für die Anwendung der UCP-Methode gegeben. Ziel ist, eine normierte und für unabhängige Schätzer einheitliche Anwendbarkeit der UCP-Methode sicherzustellen. Die Regeln wurden aus Interviews mit insgesamt ca. 20 erfahrenen Projektleitern und Chef-Architekten verschiedener Projekte ext-

rahiert. Diese Erfahrungen sind dann anhand von Schätzungen mit Spezifikationen verschiedenster Form (Use Cases, Dialoge, textuelle Beschreibungen und Mischvarianten) verifiziert worden. Eine umfangreiche statistische Überprüfung dieser Regeln insbesondere auf die damit zu erzielende Reproduzierbarkeit der Use Case Point Zählung steht noch aus.

3.1 Granularität

Das Finden des richtigen Schnittes der Use Cases ist für die Schätzung ein ganz entscheidender Punkt. Ein exaktes Maß für eine ausgewogene Use Case Beschreibung kann aber nicht angegeben werden, daher hier einige Richtwerte.

Ein Zeichen von zu komplexen (oder auch zu einfachen) Use Cases kann die Größe der betreffenden Beschreibungen im Spezifikationsdokument liefern. Wir haben verschiedene Kriterien, um an der Beschreibung festzustellen, ob der Umfang eines Szenarios zu groß gewählt ist:

- Die textuelle Beschreibung eines Szenarios umfasst mehr als eine DIN-A4-Seite **oder**
- Ein Szenario enthält mehr als 12 Schritte **oder**
- Ein Use Case beinhaltet mehr als sieben Verzweigungen, Szenarien. In diesem Fall sind die einzelnen Szenarien von der Gewichtung her eher als eigene Use Cases zu betrachten.

Übersteigt ein Use Case diese Werte wesentlich, ist er für die Schätzung zu umfangreich und sollte zerlegt werden. Die angegebenen Werte sind Erfahrungswerte aus Projekten und sollten, wie alle angegebenen Grenzwerte, im konkreten Fall nicht blind befolgt, sondern anhand eigener Erfahrungen oder anderer Use Cases überprüft werden.

Ebenso sind sehr kleine Use Cases, die sich vor allem in einer geringen Anzahl von Schritten, Szenarien und Dialogen zeigen, ein Anzeichen für einen potentiell falschen Schnitt. Zeichen dafür sind:

- Die Beschreibung eines Use Cases umfasst nur wenige Zeilen (Vorsicht, in einer Grobspezifikation kann dieses Kriterium sehr häufig zutreffen, wenn in einem Satz viele Schritte einfach nur aufgezählt werden, in diesem Fall kann der Schnitt trotzdem korrekt sein)
- Der Use Case enthält keinen Dialog **und**
- Der Use Case hat nur ein Szenario **und**
- Der Use Case hat nur einen oder zwei Schritte

Hier sollte noch einmal überlegt werden, ob es sich eventuell nicht eher um eine Anwendungsfunktion oder einen Teilschritt eines übergeordneten Use Cases handelt (z.B. "Suchen").

Wichtig ist aber auch, dass grundsätzlich „Ausreißer“ erlaubt sind. Auch hier eine hilfreiche Maßzahl: Es sollten sich ca. 80%-90% aller vorhandenen Use Cases in diesen Grenzen bewegen. Die restlichen sind Ausnahmen, die entweder sehr groß oder sehr klein sind, vom Schnitt her aber zu den anderen "passen". Insgesamt ist eine ausgewogene Verteilung von kleinen, mittleren oder großen Use Cases ein Zeichen für einen "guten" Schnitt. Für die Wiederholbarkeit und Konsistenz einer Schätzung ist entscheidend, dass die Kriterien zum Finden von Use Cases und -schritten innerhalb der einzelnen Schätzung und für verschiedene Schätzungen einheitlich sind. Es empfiehlt sich daher, im Verlaufe einer UCP Schätzung immer wieder den "Schnitt" von Use Cases und -schritten zu vergleichen.

3.2 Berücksichtigung "versteckter" Funktionalität

Häufig kommt es vor, dass neben der reinen fachlichen Funktionalität in gesonderten Kapiteln einer Spezifikation oder eines Lastenheftes weitere "Nebenfunktionalität" beschrieben ist oder diese auf jeden Fall benötigt wird. Dazu gehören z.B. Bereinigungsläufe, Migrationsprogramme, administrative Funktionalität, Verwaltungs- und Konfigurationsoberflächen.

Diese Funktionalität muss selbstverständlich auch berücksichtigt werden. In diesem Fall könnten das z.B. die folgenden Use Case sein: „*Nächtliche Bereinigung der Kontodifferenzen*“, „*Migration der Adressdaten*“, „*Konfiguration der Metadaten*“. Entscheidend ist die vollständige Abdeckung der spezifizierten (und implizit angenommenen) Anwendungsfunktionalität mittels der erfassten Use Cases.

3.3 Aufruf anderer Use Cases

Bei der Verwendung von Use Cases zur Spezifikation einer Anwendung ist auch die Möglichkeit vorgesehen, durch eine hierarchische Ordnung eine Aufrufstruktur von Use Cases zu verwenden. Verboten im Sinne einer guten und lesbaren Spezifikation sind Verzweigungen auf einzelne Szenarien oder Schritte eines anderen Use Cases. Der aufrufende Use Case sorgt dafür, dass die Vorbedingung des aufgerufenen Use Cases erfüllt ist und der gesamte (Unter-) Use Case durchlaufen wird. Nach vollständigem Durchlaufen des aufgerufenen Use Cases muss dessen Ergebnis gelten. Anschließend setzt der aufrufende Use Case seine Bearbeitung fort.

Für die UCP Methode ist der Umgang mit solchen aufzurufenden Use Cases relativ einfach folgendermaßen geregelt:

- Der Aufruf des untergeordneten Use Cases und die Schaffung der Vorbedingungen für diesen zählt als ein Schritt im aufrufenden Use Case
- Die Szenarien, Schritte und Dialoge des aufgerufenen Use Cases werden nur dort und nicht im aufrufenden Use Case gezählt (und damit über die gesamte Anwendung hinweg betrachtet auch nur einmal)

Eine extensive Verwendung von Aufrufstrukturen von Use Cases in einer Spezifikation kann ein Hinweis auf einen noch nicht passenden "Schnitt" der Use Cases für die Schätzung sein (muss es aber nicht!).

3.4 Umgang mit Batch-Funktionalität

Häufig werden serverseitig komplexe Funktionen oder auch Batch-Abläufe, die ohne Interaktion mit anderen Abläufen oder Aktoren durchgeführt werden, nicht als klassische Use Cases spezifiziert. Trotzdem handelt es sich oft um Use Cases, sie haben einen definierten Start mit Vorbedingungen (das ist häufiger sogar einfacher zu bestimmen als bei anderen Use Cases), es gibt Ablaufszenarien und ein bestimmtes fachliches Ziel.

Die Herausforderung besteht in diesen Fällen eher darin, den Ablauf für die Schätzung in einzelne Schritte der richtigen Größe zu zerlegen. Im Prinzip gelten hier alle Hinweise über die Schritte wie auch bei anderen Use Cases.

- Wenn sehr komplex: Zerlegung in Unter Use Cases entsprechend fachlicher Blöcke
- Schritte können ebenfalls gefunden werden anhand fachlicher Blöcke, dem Zugriff auf Schnittstellen oder anhand der verarbeiteten Entitäten
- Ein einzelner Schritt kann hier ggf. eine größere Mächtigkeit (=abgedeckte Funktionalität) haben als ein Schritt eines Nicht-Batch Use Cases, da es in der Regel keinen Aktoreneingriff und damit keine komplexe Schnittstelle oder z.B. einen Schichtenwechsel innerhalb der Anwendungsarchitektur geben wird
- Generell gilt, dass Schritte innerhalb eines Use Cases, über alle Uses Cases einer Schätzung hinweg und (idealerweise) über alle UCP Schätzungen hinweg etwa gleich groß sein sollten.

Ist für einen Batch Use Case eine solche Zerlegung gefunden worden, wird er genauso gezählt wie ein "normaler" Use Case, d.h. es werden Szenarien, Schritte und Dialoge (hier wahrscheinlich eher in der Form von Druckstücken oder komplexen Ausgabedateien) erfasst.

4 Bewertung von Use Case Beschreibungsformen für die UCP Methode

Für die Dokumentation von Use Cases haben sich in der industriellen Praxis unterschiedliche Formen bewährt und durchgesetzt. Diese wurden in zahlreichen Projekten untersucht und die Erfahrungen können hier nur zusammengefasst werden. Sehr gut geeignet sind Spezifikationen, die vollständig in Form von Ablaufdiagrammen oder Aktivitätendiagrammen (UML) vorliegen.

Bild 1 zeigt beispielhaft als Ablaufdiagramm den Use Case „Adresse anlegen“. Die verschiedenen Szenarien, Dialoge und Schritte sind als Legenden farblich unterschieden und nummeriert. Man beachte, dass Anzeigen (Dialoge) ebenfalls als Schritt zu zählen sind, siehe z.B. *Vorschlagliste*: 2. Dialog = 4. Schritt. Die Aufspaltung bei einer Alternative führt zu neuen Szenarien entsprechend der Anzahl der neuen Verzweigungen (keine Kombinationen mit vorherigen Szenarien). Triviale Fehlerszenarien können als eigene Szenarien mit Schritten dargestellt sein, sollten aber nicht als solche gezählt werden.

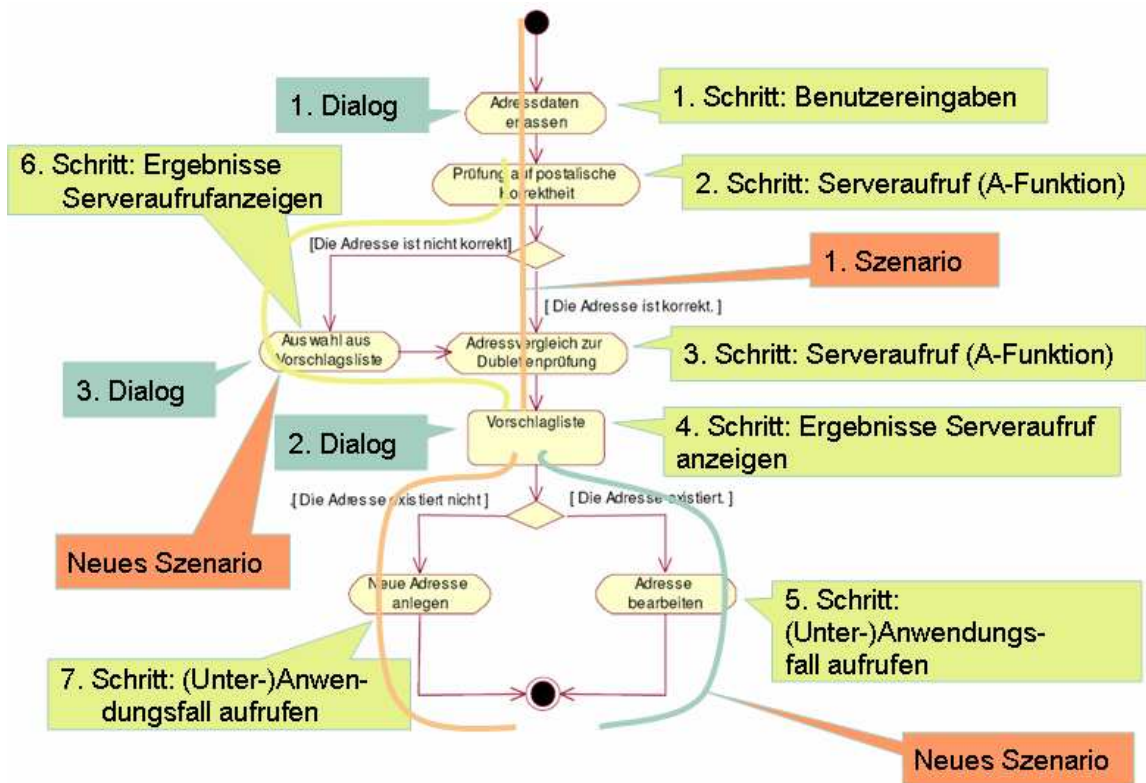


Bild 1: UCP-Zählung am Beispiel eines Ablaufdiagrammes

Wir ermitteln damit 3 Szenarien, 3 Dialoge und 7 Schritte. Gemäß Kapitel 2 folgt daraus eine mittlere Komplexität, d.h. dieser Use Case ist ungewichtet mit 10 Punkten zu bewerten.

Die UCP-Zählung wurde hier am Beispiel eines Ablaufdiagrammes exemplarisch aufgezeigt. Analog wurden weitere Spezifikationsarten und deren „Zählung“ in der UCP Methode behandelt. Eine detaillierte Darstellung würde den Rahmen

dieses Artikels sprengen und erfolgt in [2]. Das Ergebnis dieser Analyse ist aber in der folgenden Tabelle 1 zusammengefasst.

| Spezifikationsform | Eignung für UCP Methode | Vorteile | Risiken |
|---|---|---|--|
| Aktivitäten-diagramm oder Ablaufdiagramm | sehr gut | Direkte Übernahme von Use Cases und Zählung der Szenarien und Schritte; einfach und schnell | Gefahr der Vernachlässigung der Überprüfung gegen textuelle Beschreibung; Aktivitätendiagramme können zu fein sein |
| Textuelle bzw. tabellarische Beschreibung | sehr gut | Aufzählung kann direkt übernommen werden, Szenarien und Schritte sind bereits beschrieben | |
| Grobe (textuelle) Beschreibung | gut | Aufzählung kann direkt übernommen werden, Szenarien und Schritte sind grob beschrieben | Unterschätzung durch zu wenig Informationen über die Szenarien/Schritte; Unterschätzung durch vergessene ganze Use Cases |
| Dialogbeschreibung | gut | Use Cases und Szenarien sind relativ einfach zu finden, gute Abdeckung der Anwendung | Verdeckte Schritte, Anwendungslogik ist nicht erfasst, daher immer textuelle Ergänzungen beachten |
| Sequenzdiagramm | gut, wenn alle Use Cases erfasst sind, sonst mittel | Schnitt der Use Cases bereits vorgegeben; Schritte und Szenarien einfach ersichtlich | Gefahr zu feiner Granularität; in der Regel keine vollständige Abdeckung der Anwendung; Überprüfung der textuellen Ergänzung empfohlen |
| Funktionale Beschreibung | je nach Umfang gut | Beschreibung der Schritte und Funktionen, gute Abdeckung der geforderten Funktionalität der Anwendung | Komposition zu Szenarien und Use Cases u.U. aufwändig, Gefahr in Funktionen und damit zu detailliert zu schätzen |
| Zustandsdiagramm | mittel | | anderer "Schnitt" durch die Anwendung; Übersehen von Use Cases; nur im Zusammenhang mit textuellen Ergänzungen verwendbar! |
| CRUD Diagramm (Create/Read/Update/Delete) | mittel | Die meisten Use Cases sind bereits gefunden | Schritt müssen erst gefunden werden; Übersehen von Use Cases; nur im Zusammenhang mit textuellen Ergänzungen empfohlen |
| Geschäftsprozesse als Grobspezifikation | mittel, je nach Grad der Detaillierung | Aufzählung der Use Cases kann direkt übernommen oder einfach gefunden werden, guter Überblick der Anwendung | Unterschätzung durch zu wenig Informationen über die Szenarien/Schritte |

Tabelle 1: Bewertung unterschiedlicher Spezifikationsformen für die UCP Methode

5 Fazit und Ausblick

Mit Hilfe des skizzierten Leitfadens können UCP Schätzungen auf Basis von Spezifikationen unterschiedlicher Form und Granularität durchgeführt werden. Der Leitfaden bildet für große Organisationen das Fundament, damit unterschiedliche Schätzer anhand klarer Regeln einheitlich schätzen und somit Vergleichbarkeit erzielen. Der Aufwand für die Einführung des hier beschriebenen Vorgehens in einer Organisation wie sd&m beschränkt sich dabei für erfahrene Software-

Ingenieure auf das Studium des Leitfadens. Die Methodik ist leicht auf andere Softwarehäuser übertragbar.

Entscheidend für den Erfolg der Schätzung und die Belastbarkeit des Ergebnisses ist vor allem, ob und wie es gelingt, die vorliegende Spezifikationsform auf die hier beschriebenen Use Cases abzubilden. Im Idealfall entspricht die Spezifikation den aufgeführten Anforderungen, so dass nur noch eine quantitative Bewertung (Gewichtung + Zählung) erforderlich ist.

Die Schätzung erfordert keine zusätzliche Detaillierung der Use Cases, sie kann und sollte auf einem rein fachlichen Level erfolgen. Damit kann eine Grobspezifikation mit bloßer Aufzählung und kurzer Benennung dieser Komponenten sogar eine bessere (weil schneller zu erfassende) Basis für eine UCP Schätzung als eine detaillierte Beschreibung sein. Generell sind Ablaufdiagramme, Aktivitätendiagramme und textuelle bzw. tabellarische Beschreibungen sehr gut geeignet. Gute Erfahrungen wurden aber auch mit Dialogbeschreibungen für betriebliche Informationssysteme gemacht.

Wenn Schwierigkeiten beim Finden der Use Cases oder ihrer Bewertung auftreten, deutet dies i.d.R. darauf hin, dass nicht genügend Informationen zur Abschätzung des funktionalen Umfangs vorhanden ist und in diesen Fällen die UCP Methode keine zuverlässigen Werte liefern kann.

Die verschiedenen Spezifikationsformen wurden für den vorliegenden Leitfaden auf Basis von jeweils zwei bis drei Experteneinschätzungen bewertet. In einem nächsten Schritt sollte die Reproduzierbarkeit der unterschiedlichen UCP Zählungen unter Anwendung dieses Leitfadens in Abhängigkeit von der Spezifikationsform untersucht werden, in dem je Form eine größere Gruppe von Schätzern jeweils unabhängig voneinander eine UCP Bewertung durchführt. Eine geringe statistische Varianz der Use Case Points wäre dann ein Maß für die Güte der Eignung für die UCP Methode.

Literaturhinweise

1. Frohnhoff, S.; Jung, V.; Engels, G.: "Use Case Points in der industriellen Praxis" In "Applied Software Measurement - Proceedings of the International Workshop on Software Metrics and DASMA Software Metrik Kongress", Abran, A. et al. Eds. Shaker Verlag, 2006, pp. 511-526.
2. Frohnhoff, S.; Kehler, K., Dumke, R.: „Leitfaden zum Finden von Anwendungsfällen für die Use Case Points Methode“, Preprint, Fakultät für Informatik, Universität Magdeburg, 2007