

User Centered Design for SixSigma – systematische Quantifizierung in UCD-Projekten

Dirk Zimmermann
T-Mobile Deutschland GmbH
Landgrabenweg 151
53227 Bonn
dirk.zimmermann@t-mobile.de

Axel Schramm
T-Mobile Deutschland GmbH
Landgrabenweg 151
53227 Bonn
axel.schramm@t-mobile.de

Abstract

SixSigma-Methoden finden schon in vielen Projekten zur Prozessverbesserung einen erfolgreichen Einsatz. Im User Centered Design können sie ebenfalls an vielen Punkten eingesetzt werden, um Analyse, Konzeption und Evaluation zu unterstützen.

In diesem Beitrag werden die Grundzüge des Vorgehens im *Design for SixSigma* vorgestellt und auf den Bereich UCD übertragen. Ebenso berichten die Autoren von ihren Erfahrungen im Einsatz mit den Methoden in praktischen UCD-Projekten

Keywords

User Centered Design, Requirements, Quantifizierung, Messung, SixSigma

1.0 Einleitung

Im Rahmen von Prozess-Optimierungen hat die klassische SixSigma-Methode (Define, Measure, Analyze, Improve, Control: DMAIC) bisher in vielen Produktionsbereichen zu hohen Qualitätssteigerungen der Herstellungsprozesse geführt. Ebenso wurde SixSigma schon vielfach erfolgreich zur Optimierung von weiteren Geschäftsprozessen (z. B. Service oder Wartung) eingesetzt.

Im Hinblick auf die Ausgestaltung eines User Centered Design (UCD) Prozesses nach SixSigma Kriterien sind schon einige Experten, u.a. Nielsen (2003), zu dem Schluss gekommen, dass die DMAIC Methode zwar viele Tools bietet, die sinnvoll zur Messung und Optimierung eingesetzt werden können, die Methode jedoch im Sinne eines durchgängigen Prozesses weniger geeignet ist. Zur Analyse und Optimierung von Nutzerprozessen (dabei u. a. auch in Bezug auf den Software-Einsatz in diesen Prozessen) lassen sich jedoch viele DMAIC-Tools durchaus sinnvoll zur Messung und Optimierung einsetzen.

Im Gegensatz zu der prozessorientierten DMAIC-Methodik stellt das *Design for SixSigma* (DFSS) Modell einen An-

satz zur Verfügung, der weniger die Verbesserung suboptimaler Prozesse, als vielmehr die Neugestaltung von Produkten und Prozessen im Fokus hat. Mit den Phasen Define, Measure, Analyze, Design & Verify (DMADV) wird hierbei der Schwerpunkt auf die Erhebung und Quantifizierung von Anforderungen, sowie die systematische Entwicklung und Bewertung von Gestaltungsalternativen gelegt, um schließlich eine bestmögliche Lösung in Bezug auf die ermittelten Anforderungen zu entwickeln.

Die im DFSS eingesetzten Methoden zur Erhebung, Bewertung und Quantifizierung von Anforderungen und Lösungen weisen an einigen Stellen große Ähnlichkeit mit klassischen UCD-Methoden auf. Jedoch finden sich im DFSS-Bereich viele Tools, die in Bezug auf Systematik und Messbarkeit über die bekannten UCD-Methoden hinausgehen. Diese bieten dagegen oftmals eine größere Breite an Erhebungs- und Analyseverfahren, die auf vielfältige Nutzungskontexte anpassbar sind und eine kontextspezifische Analyse erlauben.

In diesem Beitrag werden die Grundlagen und Tools des DFSS vorgestellt und mit den bekannten Ansätzen des

UCD verglichen. Ziel des Beitrages ist, die wechselseitig vorhandenen Synergien aufzuzeigen, und praktische Anregungen zum Einsatz von DFSS in UCD-Projekten zu geben. Neben Methodenvorstellungen erfolgt dies anhand von Erfahrungen aus der Praxis, welche den Nutzen in Projekten illustrieren.

2.0 SixSigma Grundlagen

SixSigma hat zum einen zum Ziel, durch systematische Abbildung und Analyse von Abläufen zu erkennen, an welchen Stellen es Prozessschwächen und Fehlerursachen gibt. Zum anderen soll durch die Festlegung, Messung und kontinuierliche Überprüfung von Kennzahlen ein systematisches Qualitätsmanagement in die Prozesse implementiert werden.

Im Rahmen der SixSigma-Methodik werden drei verschiedene Projektarten unterschieden. Zum einen sind dies klassische Aufgaben des kontinuierlichen Prozessmanagements, d. h. der Verantwortung eines oder mehrerer Prozesse, im Rahmen derer die Messung, Überwachung und Steuerung von existierenden Prozessen liegt. Hier werden vor allem etablierte Kennzahlen zu laufenden Prozessen erhoben, mit definierten Zielwerten verglichen, und bei Be-

darf mit vordefinierten Maßnahmen ge-
gengesteuert.

Zum anderen können Prozesse, in denen größere Probleme festgestellt werden, im Rahmen eines SixSigma-Prozessverbesserungs-Projektes analysiert und optimiert werden. Diese Methode wird, analog zu den durchlaufenen Projektphasen, DMAIC genannt: Define, Measure, Analyze, Improve, Control. Innerhalb dieser Phasen wird der Prozess exakt definiert, mit einem Set an Mess-, Beschreibungs- und Analysemethoden durchleuchtet, und schließlich mit Hilfe von Kreativ-Techniken zur Erzeugung von Lösungsideen so verändert, dass er die definierte Zielsetzung an Output- und Prozessqualität erreicht.

Die dritte Variante des Einsatzes von SixSigma-Methoden greift dann, wenn es um die Neugestaltung eines Prozesses oder Produktes geht, also nicht auf etwas Existierendes aufgesetzt werden kann. Die Grundlage für die Neugestaltung liefert eine systematische Analyse der Anforderungen an den Prozess oder das Produkt, die in Folgeschritten in eine passende Lösung weiterentwickelt werden. Dieser Ansatz wird wiederum analog zu den durchlaufenen Phasen DMADV genannt: Define, Measure, Analyze, Design, Verify. Parallel dazu hat sich aber auch der Begriff *Design for SixSigma* etabliert, der hier im Weiteren zur Bezeichnung dieser Herangehensweise genutzt wird.

3.0 Design for SixSigma & User Centered Design

Im DFSS-Ansatz werden die Anforderungen und Bedürfnisse der Kunden in den Mittelpunkt der Analyse gestellt. Kunden sind hierbei sowohl tatsächliche Kunden, die ein Produkt kaufen oder einen Prozess nutzen, wie auch andere Stakeholder, deren Anforderungen für die Ausgestaltung der Lösung relevant sind.

Nachdem der Scope für die zu entwickelnde Lösung in der Define-Phase festgelegt wird, erfolgt somit als erster Schritt eine Erhebung und Analyse von Kundenaussagen durch Beobachtung und Befragung. Aus diesen Ergebnissen werden im Weiteren Nutzeranforderungen und Qualitätskriterien für die Lösung abgeleitet. Die zentralen Kriterien werden priorisiert und in einer Design Scorecard gesammelt, gegen sie werden die im Weiteren entwickelten Lösungen jeweils bewertet. Zur Entwicklung eines High Level Designs werden sowohl Kreativtechniken eingesetzt, um neue Lösungsansätze zu entwickeln, als auch Evaluationsmethoden, um die Erfüllung von Anforderungen zu prüfen. Durch die Entwicklung und Implementierung von Kennzahlen wird die Basis geschaffen, dass auch nach der Umsetzung (d. h. im Roll-Out und in der Nutzungsphase) die Erfüllung der Anforderungen überprüft und sichergestellt werden kann.

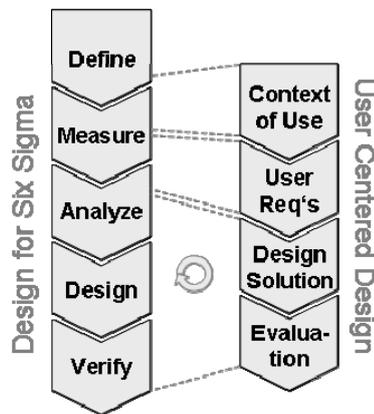


Abbildung 1: DFSS und UCD im Vergleich

Wenn man den Ansatz des DFSS mit dem klassischen Vorgehen im UCD vergleicht, springen einige Ähnlichkeiten ins Auge, etwa die Fokussierung auf Kunden- bzw. Nutzeranforderungen, das Entwickeln und Evaluieren von Prototypen, sowie die Überprüfung der Anforderungserfüllung zum Ende der Entwicklung. Bei genauerer Be-

trachtung finden sich allerdings auch einige Unterschiede, etwa in der etwas allgemeineren Ausrichtung auf „Kunden“ im DFSS und der spezifischen Fokussierung auf „Nutzer“ im UCD. Um zu verstehen, wie sich die Methoden jeweils ergänzen bzw. an welchen Stellen sie schon relativ ähnlich sind, haben die Autoren im Rahmen von UCD Projekten verschiedene Methoden des DFSS eingesetzt. Im Folgenden werden die Erfahrungen mit den jeweiligen Methoden in den Phasen Define, Measure und Analyze eines DFSS-Projektes aufgeführt.

4.0 Define

Die erste Phase in jedem Projekt ist die Projektdefinition. Hierbei ist das Hauptziel, im Projektteam ein gemeinsames Verständnis über das Projektziel, die zu erreichenden Ergebnisse und den generellen Ablauf und die beteiligten Ressourcen zu schaffen. Dies sind zwar eher klassische Projektmanagement Aktivitäten, für ein zielgerichtetes Arbeiten allerdings außerordentlich hilfreich. Das zentrale Dokument in der Define-Phase ist der Project Charter.

Im Project Charter werden alle wichtigen Informationen zum Ziel und Zweck des Projektes zusammengestellt. Er ist ein „lebendes Dokument“, welches je nach Erkenntnisfortschritt angepasst werden kann, und wird zu jedem Gate Review validiert und ggf. aktualisiert. Ein Project Charter besteht aus mehreren Hauptelementen:

Der Business Case beschreibt das Geschäftsumfeld und stellt die Ausgangssituation dar. Er erläutert, warum es sich lohnt das Projekt durchzuführen, warum die Umsetzung jetzt erfolgen soll, sowie welche Konsequenzen folgen könnten, wenn es nicht durchgeführt wird.

Zunächst wird hierbei das Problem bzw. die Produktidee dargestellt. Grundsätzlich gilt, dass hier keine Ursachenbe-

schreibung oder Formulierung von Lösungen erfolgt, sondern ausschließlich eine Beschreibung des Ist-Zustandes. Auch das angestrebte Ziel sollte klar formuliert werden, wobei ebenfalls keine Lösungen, sondern ausschließliche der Soll-Zustand beschrieben wird. Problem- und Zielformulierung müssen aufeinander abgestimmt sein. Die Zieldarstellung kann bei sauberer Problemformulierung unmittelbar abgeleitet werden. Wichtig ist, dass Probleme und Ziele nach der SMART Regel formuliert werden, also spezifisch, messbar, abgestimmt, realistisch und terminiert sind.

Die Projektabgrenzung dient in der Erfahrung der Autoren der abgestimmten Fokussierung auf bestimmte Inhalte in einem Entwicklungsprojekt. Dabei wird mit Hilfe eines Projektrahmens definiert, welche Anteile im Projekt-Scope liegen, welche außerhalb sind, und zu welchen derzeit noch keine Entscheidung getroffen werden kann. Gerade in Bezug auf Ziele für und Erwartungen an die UCD-Tätigkeiten hat sich dies als eine geeignete Visualisierung bewährt, um im Projekt eine gemeinsame Sicht auf den UCD Anteil herzustellen.

Der Multi-Generations-Plan legt dann die vereinbarten Projekthinhalte auf eine Zeitleiste und beschreibt, welche Ergebnisse im ersten, zweiten und dritten Projektteil erreicht werden sollen. In der Regel werden Projekte über mehrere Iterationen hinweg entwickelt. Damit ist auch für die UCD-Aktivitäten eine phasenweise Zieldefinition sinnvoll.

Mit Hilfe der RACI-Matrix wird in einer Tabelle zu jedem Projekt-Teilschritt definiert, welchen Anteil die verschiedenen Projektrollen daran haben. Diese reichen von (R)esponsible (=zuständig für die Erledigung) über (A)ccountable (=verantwortlich für das Ergebnis) und (C)onsulted (=mit in die Erledigung eingebunden) bis hin zu (I)nformed (=über Fortschritt und Ergebnis infor-

miert). Hiermit wird im Projekt sichergestellt, dass die relevanten Projektbeteiligten jeweils angemessen in die Aktivitäten eingebunden bzw. darüber informiert sind.

Weitere Tools sind die Stakeholder-Analyse, mit deren Hilfe kontinuierlich betrachtet wird, wie einzelne Projektbeteiligte zu dem Projekt stehen, und wie ihre Meinung ggf. beeinflusst werden kann, sowie der Kommunikationsplan, der schon zu Beginn des Projektes die Kommunikation von Standard- und außergewöhnlichen Ereignissen definiert, z. B. bei Projektproblemen oder –verzug.

Was auf den ersten Blick selbstverständlich wirkt, bekommt durch den Einsatz formeller Methoden in der Erfahrung der Autoren eine größere Verbindlichkeit, als sie üblicherweise in UCD-Projekten vorhanden ist. Hiermit verbunden ist auch ein offizielles Commitment zu den Zielen sowie von benötigten Ressourcen zur Durchführung, das im Weiteren individuelle Anfragen und Absprachen obsolet macht oder sehr vereinfacht. Ebenso werden die UCD-Aktivitäten als eigenständiger Beitrag zum Projekt sichtbar, und der erreichbare Mehrwert wird schon zu Beginn spezifiziert.

5.0 Measure

Im Rahmen der Measure-Phase werden die Kunden und Nutzer identifiziert und entsprechend möglicher Unterscheidungskriterien segmentiert. Zu den jeweiligen Gruppen werden dann Anforderungen erhoben, gesammelt, sortiert & priorisiert. Im Rahmen von DFSS ist es ein zentrales Ziel, zu den einzelnen Anforderungen messbare Qualitätskriterien (Critical to Quality, CTQ) und daraus abgeleitete konkrete Messgrößen zu definieren und schon in der Anforderungsanalyse Zielwerte bzw. einen akzeptablen Va-

riationsbereich zu definieren. Im Folgenden werden die zur Erhebung und Analyse jeweils eingesetzten Methoden des DFSS vorgestellt

5.1 Erhebung von Daten

Im UCD existieren eine Vielzahl von Methoden, um Nutzungs-Kontext und –Anforderungen zu erheben, so z.B. die Kontextanalyse (z. B. nach DA-Tech, 2009) oder die Contextual Inquiry (Beyer & Holtzblatt, 1997). Ihnen gemein ist, dass durch Beobachtung und Befragung von Nutzern in deren jeweiligem Arbeitsumfeld ein möglichst realistisches Bild der Organisation, Aufgabenstruktur, Aufgabenbearbeitung und anderer Kontextfaktoren gezeichnet werden kann.

Im DFSS wird zur Analyse ebenso eine Vielzahl von Methoden eingesetzt. Diese reichen von klassischen Dokumenten-recherchen über Interviews und Fokusgruppen bis hin zu weitreichenden Umfragen. Sobald sich eine Erhebung um den eigentlichen Ort des Geschehens (japanisch: Gemba) dreht, also direkt vor Ort durchgeführt wird, wird im DFSS von einer Gemba-Studie gesprochen. Diese bildet in der Regel das zentrale Element einer Anforderungserhebung. Das Ergebnis dieser Erhebungen wird im DFSS „Voice of the Customer“ genannt. Es bildet gemeinsam mit dem ebenfalls zu erhebenden „Voice of the Business“, also den wirtschaftlichen oder technischen Anforderungen, die Basis für die weitergehende Analyse.

Im direkten Vergleich ähneln sich die beiden Herangehensweisen sehr, allerdings finden sich auf der UCD-Seite methodisch vielfältigere Ansätze zur Erhebung von Nutzungsanforderungen, nicht zuletzt weil SixSigma sich methodisch nicht auf den Nutzungsaspekt konzentriert, das UCD aber schon.

Viele der UCD Erhebungsmethoden (z. B. teilnehmende Beobachtung, Kontextinterviews) lassen sich jedoch ohne

Probleme in Gemba-Studien und als Quelle für die nachfolgende Analyse mit DFSS-Methoden nutzen.

5.2 Auswertung & Ableitungen

Die Analyse der "Voice of the Customer" (VOC) erfolgt im DFSS in mehreren Schritten. Im ersten werden hierbei aus den verschiedenen VOC die eigentlichen User Needs abgeleitet, d. h. zu formulierten Wünschen wird hinterfragt, was die zugrundeliegende Anforderung ist. Hier erfolgt bereits eine erste Strukturierung der Anforderungen, indem die verschiedenen Anforderungsquellen übereinandergelegt und in eine Übersicht gebracht werden, die sogenannte User Needs Table.

Im Folgeschritt werden die Anforderungen in eine hierarchische Struktur gebracht, die verschiedene Ebenen abbildet und somit zusammengehörige oder einander unter- oder übergeordnete Anforderungen beschreibt. Hierzu wird üblicherweise gemeinsam mit Affinity-Diagrammen gearbeitet, die so weit diskutiert und entwickelt werden, bis ein konsistentes Abbild der User Needs entsteht. Basierend hierauf wird entschieden, auf welcher Ebene in der Hierarchie im Folgenden priorisiert und geplant wird. Dies hat den Vorteil, dass in der Priorisierung jeweils vergleichbare Anforderungen gegeneinander bewertet werden, und man nicht auf verschiedenen Ebenen zu bewerten hat.

Die erste Priorisierung findet über eine Kano-Analyse statt, in der die Anforderungen in eine von drei Klassen eingeteilt werden:

- a) Basisanforderungen oder Dissatisfier, deren Erfüllung nicht unbedingt zur Nutzerzufriedenheit beiträgt, deren Abwesenheit jedoch zu Unzufriedenheit führen würde.

- b) Leistungsanforderungen, die sowohl bei Erfüllung zur Zufriedenheit beitragen, wie auch bei Fehlen zur Unzufriedenheit.
- c) Begeisterungsanforderungen oder Satisfier, deren Fehlen die Nutzer nicht unzufrieden macht, deren Erfüllung jedoch positiven Einfluss auf die Zufriedenheit hat.

Die Analyse erfolgt, indem zu jeder Anforderung sowohl eine Einschätzung zu einem Produkt, was diese erfüllt abgefragt wird, wie auch zu einem Produkt, was die Anforderung nicht erfüllt. Aus der Kombination beider Antworten lässt sich dann die Anforderungsart ableiten.

Die Basisanforderungen bilden den Grundstock der Muss-Anforderungen, da sie in der Regel die zwingend notwendigen Produkteigenschaften beschreiben. Daher werden sie in der weiteren Priorisierung nicht mehr mit bewertet, sondern jeweils nur mit aufgeführt. Zu den Leistungs- und Begeisterungsanforderungen wird im Folgenden ein weiteres Tool eingesetzt, mit dem eine Rangreihe der Priorität der Anforderungen erstellt werden kann. Dieses Tool nennt sich Analytisch-Hierarchischer Prozess (AHP) und basiert auf einem Paarvergleich aller betrachteten Anforderungen. Aus den paarweisen Bewertungen wird mit dem Tool dann ein relatives Gewicht der einzelnen Anforderungen berechnet, das zur Bildung der Prioritäts-Rangfolge genutzt wird.

Anhand dieser Liste wird sodann entschieden, bis zu welcher Priorität Anforderungen umgesetzt werden. Die resultierenden Anforderungen bilden gemeinsam mit den Basisanforderungen die Grundlage für die weiteren Schritte.

Der erste Folgeschritt ist die Definition von messbaren Qualitätskriterien für jede priorisierte Anforderung, diese werden „Critical to Quality“, also CTQ genannt. Für jedes dieser CTQs werden dann eine oder mehrere Messgrößen definiert, anhand derer der Grad der Erfüllung bewertet werden kann.

Als letzter Schritt der Measure-Phase werden zu allen CTQs erwartete Zielwerte bzw. Zielkorridore definiert. Diese ermittelt man z. B. durch Benchmarking mit Mitbewerbern oder Ist-Analysen bei Bestandssystemen, aber auch durch Erfahrungswerte von Nutzern und Experten.

Im Vergleich mit der Anforderungsanalyse im UCD finden sich im DFSS einige Schritte, die so im UCD nicht explizit durchlaufen werden. Anforderungen werden nicht nur erhoben, strukturiert und aufgelistet, sondern mit Stakeholdern gemeinsam analysiert & bewertet, in Relation zueinander gesetzt und abgewogen. Durch eine Betrachtung und gemeinsame Bewertung im Projektteam kann in der Erfahrung der Autoren schon frühzeitig im Projekt eine gemeinsame Sicht und somit ein grundsätzliches Commitment der Projektpartner erreicht werden. Ebenso wird gemeinsam ein klarer Zielkorridor für die erwartete Systemleistung definiert, anhand dessen im weiteren z. B. Prototypen bewertet und verglichen, oder aber auch die gesamte Systemleistung bewertet werden kann. Der zusätzliche Aufwand für die Strukturierung, Gruppierung und Quantifizierung hat sich in der Projekterfahrung gerade in Bezug auf Verbindlichkeit und Commitment rentiert.

6.0 Analyze

Während der Analyze-Phase werden die zuvor erhobenen Anforderungen systematisch in Lösungsalternativen umgesetzt, welche dann anhand der festgelegten CTQs bewertet werden.

Im ersten Schritt wird hierbei eine Design Scorecard erstellt. In ihr werden die nach AHP ausgewählten Anforderungen aufgelistet, und um die jeweiligen Messgrößen, sowie festgelegte Zielkorridore für alle CTQs ergänzt. Die Erstellung der Design Scorecard erfordert in der Regel erneut die Einbeziehung von Stakeholdern und Nutzern, vor allem für die Festlegung der Zielkorridore.

Der Vorteil dieses Schrittes liegt darin, dass schon vor Beginn des Entwurfes mögliche Konflikte und Widersprüche in der erwarteten Systemleistung aufgezeigt und gelöst werden können.

Mit Hilfe von Kreativitätstechniken werden danach Lösungsansätze für das System entwickelt. Hierbei wird zunächst der Lösungsraum umrissen, d. h. die Dimensionen definiert, anhand derer sich mögliche Lösungsalternativen unterscheiden können. Danach werden im Projektteam verschiedene Lösungsalternativen, z. B. mit Hilfe einer Morphologischen Box entwickelt, in der eine Alternative jeweils einen Lösungsansatz anhand der Ausprägung auf den Dimensionen beschreibt. Für diese Alternativen kann dann im Folgenden mit Hilfe der Pugh-Matrix eine erste Abschätzung erfolgen, welche Art der Umsetzung am ehesten geeignet ist, möglichst viele Anforderungen zu erfüllen.

Dieser Ansatz erlaubt, schon sehr früh im Konzeptions- und Gestaltungsprozess die Vielzahl möglicher Lösungsalternativen einzuschränken, was die folgenden Schritte der Lösungsbeschreibung zielgerichteter ablaufen lässt.

Eine oder mehrere ausgewählte Lösungsansätze werden daraufhin prototypisch in ein High Level Design überführt. Die Konzeption wird dabei dadurch erleichtert, dass es schon klar umrissene Eigenschaften aus der Morphologischen

Box gibt, sowie die Beschreibung der erwarteten Systemleistung aus der Design Scorecard.

Die weiteren Schritte ähneln denen der Konzeption und Gestaltung aus dem UCD sehr. Auch hier erfolgen ein iteratives Prototyping und die formative Evaluation mit Nutzern und Stakeholdern. Gerade der Evaluationsanteil wird allerdings auch hier durch die zuvor erhobenen und definierten Anforderungen und Qualitätskriterien unterstützt, da die relevanten Messgrößen schon im Vorfeld definiert und im Projektteam abgestimmt sind.

Das Ergebnis der Analyse-Phase ist dann ein definiertes High Level Design, das die größtmögliche Umsetzung der Anforderungen gewährleistet, sowie in entscheidenden Kriterien auch schon in Bezug auf Umsetzungsgrad evaluiert wurde.

Aus UCD-Sicht ist der Hauptnutzen von DFSS in dieser Phase dadurch gegeben, dass ein systematischer Schritt von der Anforderungsanalyse zur Konzeption und Gestaltung erfolgen kann, und schon sehr früh in der Konzeptionsphase ein Bewertungsrahmen für Lösungen vorliegt. Darüber hinaus wird die kreative Arbeit durch eine klare Beschreibung der erwarteten Systemeigenschaften unterstützt.

7.0 Design & Verify

In den Phasen Design und Verify erfolgen die eigentliche Umsetzung der Lösung sowie der Übergang in den Betrieb und die kontinuierliche Qualitätssicherung. Auch hier dienen die erhobenen Anforderungen und Kriterien als Rahmen für z. B. Evaluation und Testing.

Zusätzlich wird im Rahmen der Verify-Phase allerdings auch noch ein Plan

entwickelt, wie die Systemleistung im Betrieb kontinuierlich überprüft und ggf. korrigiert werden kann. Hierdurch wird der Grundgedanke des SixSigma unterstützt, die Qualität eines Produktes oder Prozesses nicht nur einmal zu definieren, sondern kontinuierlich überprüfen und bei Bedarf reagieren zu können.

8.0 Resümee und Ausblick

Anhand der drei Phasen Define, Measure und Analyze wurden die möglichen Ansatzpunkte zur Integration von DFSS und UCD vorgestellt, sowie Erfahrungen mit dem Einsatz von DFSS in UCD-Projekten geschildert. Die Ansätze sind in den Augen der Autoren ideal geeignet, um sich gegenseitig zu ergänzen und somit das UCD um Komponenten des Qualitätsmanagements zu erweitern. Der Aspekt der frühzeitigen Quantifizierung von Anforderungen sowie die systematische Fortführung in Konzeption und Design stellen hier den Hauptnutzen dar.

Die Nutzung von DFSS in den Folgephasen Design und Verify befindet sich derzeit in der Pilotierung, Erfahrungen dazu sind als Thema für weitere Publikationen in Planung.

9.0 Literatur

Beyer, H. and Holtzblatt, K. (1997). Contextual Design : A Customer-Centered Approach to Systems Designs. Morgan Kaufmann.

DATech (2009). Leitfaden Usability v1.2 <http://www.datech.de/share/files/Leitfaden-Usability.pdf>

Nielsen, Jacob (2003): Two Sigma, Usability and Quality Assurance, Alertbox 24.11.2003, <http://www.useit.com/alertbox/20031124.html>

Staudter, C., Mollenhauer, J.-P., Meran, R., Roenpage, O., von Hugo, C., and Hamalides, A. (2008). Design for Six Sigma and Lean-Toolset. Springer.