

Wie können Ansätze aus Lean Six Sigma beim Steuern von IT-Projekten helfen? Ein Erfahrungsbericht.

Kristina Birn¹

Abstract: Prozessmanagement beschäftigt sich mit der Steuerung und Verbesserung von Geschäftsprozessen. Mit Lean Six Sigma wird die Brücke zwischen Prozessoptimierung und Lean Management geschlagen. Auch wenn die Prozessoptimierung ihre Wurzeln in der produzierenden Industrie hatte, wurden mittlerweile Ansätze für alle Geschäftsbereiche von Unternehmen und Organisationen entwickelt. Aber betrifft Prozessoptimierung nicht eigentlich nur die Linie in produzierenden Unternehmen? Wie könnte diese Methodik bei der Umsetzung von Projekten helfen? Anhand von Praxisbeispielen wird gezeigt, wie Ansätze und Werkzeuge aus Lean Six Sigma bzw. Prozessmanagement allgemein beim Management von IT-Projekten unterstützen können.

Keywords: Projektmanagement, Prozessmanagement, Lean Six Sigma, Digitalisierung, IT-Projekte, Geschäftsprozesse.

1 Einleitung

Six Sigma ist ein Managementsystem zur nachhaltigen Verbesserung von Prozessen und der Ergebnisqualität. Lean (Management) ist eine Denkweise, die eine vollkommene Realisierung des Kundenwertes zum Ziel hat. Beide haben ihren Ursprung in der produzierenden Industrie, Six Sigma wurde 1987 von Motorola in den USA entwickelt, die Basis für Lean Management lieferte das Toyota-Produktionssystem, das in den 1990er Jahren weltweit für Furore sorgte.

Die Verbindung beider Gedankengebäude zu Lean Six Sigma (oder auch Six Sigma+Lean) Anfang dieses Jahrtausends war naheliegend: in beiden Fällen geht es um die wirtschaftliche Erfüllung von Kundenbedürfnissen als ultimativem Qualitätsmaßstab - Six Sigma steht als Qualitäts-Kennzahl für eine Einhaltung der vom Kunden gesetzten Spezifikationsgrenzen in 999.996,6 von 1 Million Fällen.

Längst hat Lean Six Sigma die Begrenzung auf die produzierende Industrie überwunden und wird als allgemein gültiges Konzept für das Qualitätsmanagement aller Geschäftsprozesse eines Unternehmens betrachtet: „Six Sigma is for the entire organization, not solely for manufacturing or operations. Every process needs to be addressed. [...] Six Sigma is about quality – but not just in terms of the final product.“ [Wh03, S. 36 und 41] Um dieses Argument noch weiter zu treiben: Lean Six Sigma funktioniert prinzipiell auch in reinen Dienstleistungsunternehmen. Denn Verschwendung findet man fast überall.

Optimierte Prozesse gehen einher mit minimaler Verschwendung, stabilisierte Arbeitsabläufe führen zu geringen Qualitätsschwankungen beim Prozess-Ergebnis. Das ständige Streben nach Verbesserung aus der Lean-Welt passt zum DMAIC-Zyklus von Six Sigma mit seinen Phasen Define-Measure-Analyze-Improve-Control, noch verstärkt durch die

¹ Dr. Birn IT-Prozesse, Grünwälderstr. 10-14, 79098 Freiburg, elektropost@kbirn.de

zusätzliche Phase „Reinforce“ [PE16] im Sinne eines kontinuierlichen Verbesserungsprozesses.

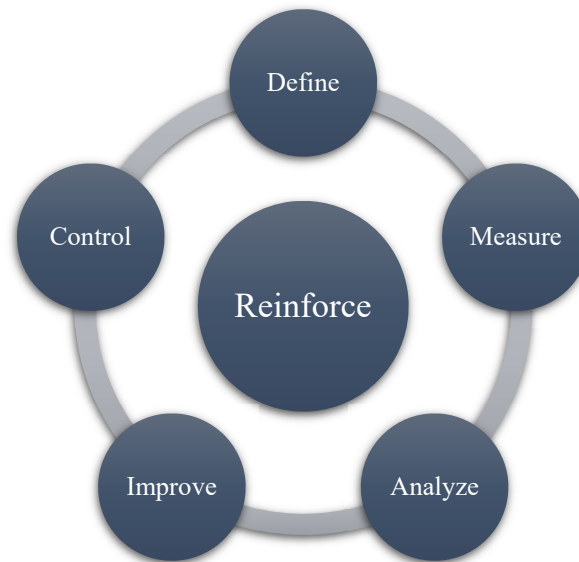


Abb. 1: Der DMAIC-Zyklus (erweitert um das ‚R‘ für Reinforce)

Ein wichtiges Fundament für eine Prozessoptimierung nach Lean Six Sigma ist eine belastbare Statistik: Lean Six Sigma hat zum Ziel, dass die im Rahmen eines Prozesses produzierten Ergebnisse stabil innerhalb definierter Toleranzgrenzen liegen. Sowohl der aktuelle Ausgangspunkt als auch das angestrebte Ziel werden in klar messbare Kennzahlensysteme gefasst. Auf diese Weise ersetzt man die nahezu unvermeidlichen Fehleinschätzung des Zustands „nach Gefühl“ durch eine transparente und einigermaßen objektive Analyse der Situation [siehe hierzu auch Mi19]. Das ist bei Prozessverbesserungsprojekten wichtig, da hier häufig in langjährig eingeübte Handlungsweisen eingegriffen wird (was in der Umstellungsphase für die Betroffenen umständlich bis schmerzhaft ist) und daher Ausgangsbasis und gewünschter Zielzustand möglichst frei von willkürlichen Annahmen dargestellt werden sollten.

Genau hier liegt aber auch nach meiner Erfahrung eine Herausforderung bei der Anwendung von Lean Six Sigma im Rahmen von Digitalisierungsprojekten im Dienstleistungsbereich oder bei Unterstützungsprozessen, da sich die Projektziele hier einer quantitativen Bewertung entziehen, und zwar sowohl hinsichtlich der Messung der Ausgangssituation insgesamt als auch der Prozessstabilität im Speziellen. Das ist insbesondere bei der Softwareentwicklung maßgeblich: „Processes for software development are necessary, but their output is hardly predictable with standard statistical process control.“ [Fe19, S. 3]. Fehlmann hat für die Prozesskontrolle in der Softwareentwicklung bereits gezeigt, dass eine Reihe von Lean SixSigma-Werkzeugen wie das *Ishikawa-Diagramm*, die *Deming Process Chain* und *Quality Function Deployment (QFD)* mit gewissen Anpassungen die Ergebnisqualität steigern können [Fe19].

Der vorliegende Beitrag konzentriert sich (ergänzend zur Software-Entwicklung i.e.S.) auf das „Drumherum“ eines IT-Projekts, die Veränderung von Geschäftsprozessen durch den Einsatz von IT (also das Erfassen der Anforderungen, die Auswahl der entsprechenden Werkzeuge und deren Einsatz in den entsprechenden Prozessen) und das Projektmanagement. Auch hier steht man vor der Aufgabe, das gewünschte Projektziel bzw. den zu erreichenden Ziel-Zustand in geeignete Kennzahlen zu gießen. Wenn man nicht von vornherein mit Umsatzsteigerung oder (noch schlimmer) Kostensenkung argumentieren will, muss man meist etwas tiefer graben. Der Aufwand lohnt sich aber, weil die Identifikation der Projekt-Stakeholder mit den Prozessveränderungen besser funktioniert, wenn man ihnen ein etwas persönlicheres Ziel bietet als das Wohl des ganzen Unternehmens.

Lean Six Sigma bietet eine Reihe von Werkzeugen und Verfahrensweisen, die dabei unterstützen, Prozessveränderungsprojekte so effektiv wie möglich und dabei nur so schmerzhaft wie nötig umzusetzen. Die standardisierten Projektphasen, -rollen und -ergebnisse helfen dabei, beim Projektmanagement selbst Verschwendung zu vermeiden und sich auf das Projektziel zu konzentrieren [Ad12].

Im Folgenden wird auf Basis eigener Praxiserfahrungen anhand der fünf DMAIC-Phasen exemplarisch skizziert, wie sich der Werkzeugkasten von Lean Six-Sigma [Ge05] auf IT-Projekte bzw. die Digitalisierung von Geschäftsprozessen übertragen lässt. Bei IT-Projekten ist es je nach Vorgehensweise und Agilität der Entwicklung erforderlich, die Phasen IMPROVE und CONTROL um IT-spezifische Detailphasen und/oder Rückkopplungsschleifen zu erweitern (siehe hierzu beispielsweise den Vorschlag in [Ad12]). Die sechste Reinforce-Phase erfordert keine zusätzlichen Werkzeuge (sondern eher eine konsequente Anwendung der bereits vorhandenen), daher wird sie aus den nachfolgenden Betrachtungen ausgeklammert.

Im Folgenden wird ein Fall betrachtet, in dem es um den Prozess „Eintritt neuer Mitarbeiter ins Unternehmen“ geht. Bei der betrachteten Firma müssen mehrere Abteilungen (Personalabteilung, Fachabteilung des neuen Mitarbeiters, hauseigene IT, Controlling etc.) dafür Aufgaben erfüllen, die teilweise voneinander abhängen. Im Status Quo wurden diese Aufgaben mehr oder weniger erfolgreich über einen Laufzettel in Papierform abgebildet, in dem die erforderlichen Aufgaben aufgelistet waren, in den aber auch wichtige Informationen für nachfolgende Schritte händisch eingetragen wurden.

Der Prozess war für die Firma unbefriedigend, weil hochqualifizierte (und entsprechend teure) neue Mitarbeiter regelmäßig die ersten Arbeitstage damit zubrachten, organisatorische Details zu klären bzw. auf Arbeitsmaterialien zu warten, oder im Verlauf der ersten Wochen ihnen zugeteilte Aufgaben nicht erledigen konnten, weil zentrale Befugnisse nicht rechtzeitig erteilt worden waren. Die Zufriedenheit aller Beteiligten mit diesem Prozess war entsprechend niedrig.

Die Aufgabe bestand darin, gemeinsam mit dem IT-Dienstleister den Workflow für den Eintritt neuer Mitarbeiter so aufzubereiten, dass er IT-gestützt ablaufen konnte. Außerdem sollten erforderliche Informationen automatisch aus der Datenbank des Unternehmens gezogen bzw. dort an entsprechender Stelle hinterlegt und damit zentral verfügbar gemacht werden.

2 Lean Six Sigma in IT-Projekten

2.1 DEFINE-Phase: Wo liegt das Problem?

In der DEFINE-Phase findet der erfahrene Projektmanager einen größeren Teil seiner bewährten Werkzeuge wieder wie Big Picture, Steckbrief, PSP, Netzplan, Risiko- und Stakeholdermatrizen, RACI-Matrix und Kommunikationsplan. Die zentrale Bedeutung des Kundenwunsches zeigt sich in der Integration von auch aus dem Anforderungsmanagement bekannten Elementen wie der Kundenbedürfnistabelle und des Kano-Modells.

Der Projektsteckbrief – bzw. die Project Charter – ist bei IT-Projekten essenziell wichtig. Gerade hier ist das Risiko, dass sich der Projektauftrag bzw. die resultierenden Anforderungen im Lauf des Projekts verändern, sehr hoch. Umso wichtiger ist es, den Status Quo am Anfang sauber zu erfassen: Wozu machen wir das Projekt? Welcher Nutzen wird erwartet? Was ist am heutigen Zustand nicht zufriedenstellend? Was wird von dem Projekt erwartet? Und genauso wichtig: Was nicht? Selbst wenn sich der ‚Feature Creep‘ möglicherweise nicht verhindern lässt, so kann man ihn dadurch zumindest transparenter gestalten. Die unbedingte Orientierung am Kundenwert hilft hier dabei, potenzielle Verschwendung und nur scheinbar nützliche Anforderungen zu identifizieren und auszuschließen.

Im ersten Schritt ging es in unserem Fallbeispiel darum, die Project Charter / den Projektsteckbrief anzulegen, mit den Teilaspekten

- Ausgangssituation, Problem, Ziel (SMART formuliert²) und erwarteten Nutzen formulieren;
- Projektrahmen abstecken (IN/OUT-Rahmen) – beispielsweise war der Einstellungsprozess nicht Teil des Projekts;
- Stakeholder identifizieren (neuer Mitarbeiter, beteiligte Abteilungen des Unternehmens);
- geplanten Zeitrahmen mit Meilensteinen definieren.

Projektziel war, dass ab Einführung des Workflow-Tools am ersten Arbeitstag eines neuen Mitarbeiters alle erforderlichen Vorarbeiten vollständig abgeschlossen waren.

Im vorliegenden Fall war das (Papier-)Formular für den Eintritt der Mitarbeiter der Ausgangspunkt. Aus den vorhandenen Informationen wurden alle bekannten Aufgaben (noch ohne Details zu den Aufgaben) sowie die jeweiligen Abhängigkeiten abgeleitet und als erste Diskussionsgrundlage in einem Flussdiagramm erfasst.

² Spezifisch, Messbar, Akzeptabel, Realistisch, Terminiert

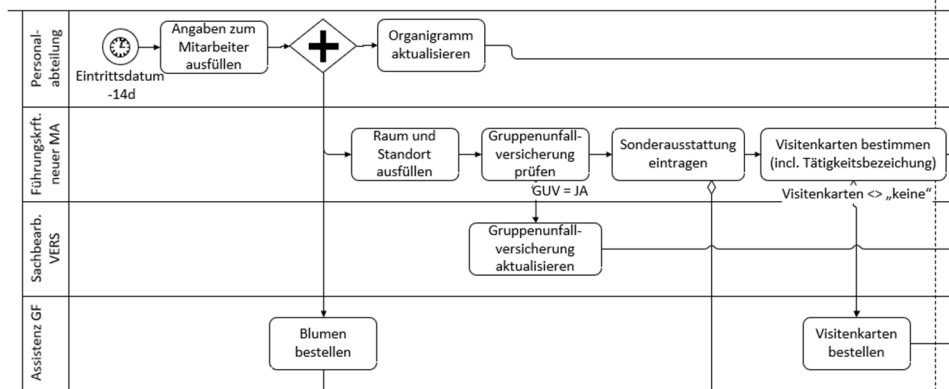


Abb. 2: Flussdiagramm Eintrittsprozess (Auszug)

Beim Erfassen der Anforderungen helfen CTC-/CTB-Matrizen. *Critical to Customer* (CTC) leitet aus Kundenmeinungen (*Voice of Customer*, VoC) eindeutige und messbare Kundenanforderungen ab – wobei Kunde hier sehr weit gefasst werden kann: Bei der Digitalisierung von Geschäftsprozessen gibt es häufig keinen direkten Bezug zu Kundenaufträgen oder -anforderungen, hier können die Angestellten der direkt oder unmittelbar betroffenen Abteilungen des eigenen Unternehmens als Kunden betrachtet werden. Mittels Quality Function Deployment (QFD) können die Kundenanforderungen in Spezifikationen überführt werden.

Zusätzlich werden aus den Stimmen aus dem Business (*Voice of Business*, VoB) die messbaren Anforderungen des eigenen Unternehmens *Critical to Business* (CTB) abgeleitet. Bei der Digitalisierung von Geschäftsprozessen ist hier neben der generellen strategischen Linie des Unternehmens (z.B. hinsichtlich Preisbewusstsein und Qualitätsanspruch) auch von zentraler Bedeutung, was sich die Geschäftsleitung konkret von einer Verbesserung der Geschäftsprozesse verspricht.

Bei Digitalisierungsprojekten steht man häufig vor der Herausforderung, aus relativ schwammigen Aussagen wie „die Zuständigkeiten sind unklar“ oder „das Formular ist total unübersichtlich“ messbare Spezifikationsgrenzen zu ziehen. Klar quantifizierte Ziele sind für den Projekterfolg aber unerlässlich. Dieser Schritt ist daher zwar mühsam, aber wichtig und führt häufig dazu, dass Teile der Analyse-Phase vorgezogen werden müssen, um zum wahren Bedürfnis von Kunden oder Business zu finden. Der Aufwand lohnt sich aber, um auch später bei der Entwicklung bzw. dem Customizing der entsprechenden Software eine bessere Vorstellung vom erwarteten Nutzen zu haben. Außerdem deutet sich bereits in dieser Phase häufig an, dass manche beobachteten Probleme möglicherweise gar nicht durch den Einsatz von Software gelöst werden können, sondern organisatorische Anpassungen erfordern.

Im Workshop mit Vertretern der beteiligten Abteilungen wurden daraus die Anforderungen an den Eintrittsprozess abgeleitet. Der Einstieg war, wie gut das derzeitige Formular inhaltlich bereits die anstehenden Aufgaben abbildete bzw. an welcher Stelle es konkret im aktuellen Prozess ‚hakte‘. Es kann hilfreich sein, wenn man als Moderator des Verfahrens eigene Vorschläge für mögliche wahre Bedürfnisse anbietet, um die Reaktion der Teilnehmer zu testen bzw. um die erste Hemmschwelle zu überbrücken. Das war in diesem

Fall nicht nötig, die Stimmen von Kunde und Business (VoC/VoB) wurden von Teilnehmern schnell formuliert, wie beispielsweise:

- „Ich möchte meine Aufgaben so früh wie möglich zugeteilt bekommen.“
- „Ich muss an so viel denken, da fällt das schon mal hinten runter.“
- „Die Angaben auf dem Zettel stimmen meistens nicht, ich muss regelmäßig nachfragen.“
- „Es wäre toll, wenn die Blumen direkt am ersten Tag auf dem Schreibtisch stehen.“
- „Ich weiß nie, was schon erledigt ist.“

Diese Angaben wurden in einer CTC/CTB-Matrix geordnet und priorisiert. Das zugrundeliegende wahre Bedürfnis wurde daraus abgeleitet (z. B. der Wunsch nach einer fehlerfreien und pünktlichen Bearbeitung) und anschließend die messbare Anforderung daraus abgeleitet, wie z.B.

- „Alle Prozessbeteiligten erhalten vom System ihre Aufgaben unter Berücksichtigung der Abhängigkeiten zum frühestmöglichen Zeitpunkt zugeteilt. Die Information darüber erfolgt in Echtzeit sowohl im Workflow-Tool selbst als auch zusätzlich per E-Mail.“

2.2 MEASURE-Phase: Wo stehen wir heute?

In jedem Fall muss es möglich sein, die Kunden- und Business-Anforderungen in IT-Projekten in eindeutige User-Stories zu überführen. Wenn das nicht gelingt, ist die Anforderung noch unklar. Nicht immer gelingt allerdings die Definition von Output-Messgrößen. Hier muss man sich häufig damit begnügen, dass nach erfolgreichen technischen und funktionalen Tests die Abnahme der Features durch den Anforderer das einzige (eher binäre) Entscheidungskriterium ist.

Ich hatte bislang Schwierigkeiten, die statistischen Verfahren, die Lean Six Sigma für diese Phase bietet, auf die Anforderungen aus IT-Projekten zu übertragen. Bei den nicht-funktionalen Eigenschaften im Rahmen von z.B. Stress-Tests ist es einfacher, da sich Zugriffszeiten, Übertragungsraten etc. besser messen lassen. Aber insbesondere die Kundenanforderungen entziehen sich häufig einer kardinalen Messung. Bearbeitungs- und Durchlaufzeit, Klickwege und der Erfolg von Suchanfragen lassen sich noch messen, aber ob der Abbruch eines Vorgangs auf die Eigenschaften der Software oder auf eine spontane Umpriorisierung durch den Anwender zurückzuführen ist, lässt sich durch reine Messung meist schwer nachweisen. Und für eine systematische und vor allem statisch belastbare Nutzerbefragung und -analyse fehlen meist Zeit und Geld.

Im vorliegenden Fall einigte man sich auf die folgenden kardinalen Messkriterien:

- Bei wieviel Prozent der neu eingestellten Mitarbeiter waren am ersten Tag nicht alle Aufgaben erledigt?

- Wie häufig bekommt man während der Vorbereitung ein ausgefülltes Formular mit Rückfragen zurück bzw. wie oft geht ein Vorgang im ersten Anlauf durch (*Right First Time*, RFT)?
- Wie häufig werden Felder zweckentfremdet, weil es für wichtige Informationen keine passende Stelle zu geben scheint?

Die meisten der definierten Kundenbedürfnisse liefen demgegenüber darauf hinaus, dass die in der DEFINE-Phase definierte Anforderung aus Sicht des Anforderers korrekt erfüllt wird (Beispiel s.o.).

2.3 ANALYZE-Phase: Woran liegt's denn?

Aus der Analyze-Phase haben es mir für IT-Projekte insbesondere alle Werkzeuge ange-tan, die sich mit dem „Warum“ beschäftigen. Es ist erstaunlich, zu welche spannenden Antworten man über die „5-Why“ (eine Methode, die jedes dreijährige Kind mühelos beherrscht) in einem Workshop kommt.

Außerdem nutze ich für die Analyse von IT-gestützten Prozessen die Darstellung in Prozessdiagrammen, weil die Erfahrung gezeigt hat, dass diese Darstellungsform von Anwendern und Entwicklern auch ohne spezielles Fachwissen gleichermaßen gut verstanden wird. Es lässt sich anschaulich eine Fülle von Informationen einschließlich Schwachstellen komprimiert darstellen, fehlende Ausprägungen oder Über-Definitionen sind visuell einfach erkennbar.

Und schließlich ist auch das Ergebnis einer Prozessoptimierung über Flussdiagramme sehr anschaulich darstellbar, wie das nachfolgende Beispiel zeigt. Hier wurde beispielsweise durch die Einführung von SAP die Spritzgussplanung bei einem Automobilzulieferer von einem hochgradig händischen Prozess in eine vollautomatisierte Vorplanung überführt. Auf den ersten Blick ist beim Vergleich der beiden Prozessdiagramme bereits zu sehen, dass händische Aufgaben (weiß) durch automatisierte Aufgaben (blau) ersetzt wurden und damit eine potenzielle Fehlerquelle (rote Anmerkung) aufgelöst werden konnte:

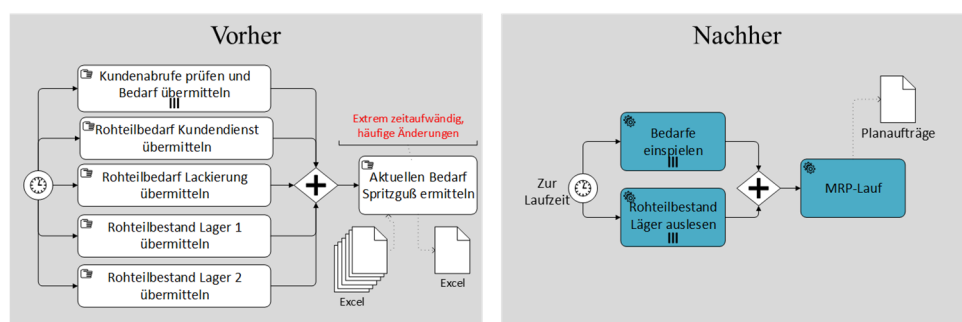


Abb. 3: Visuelle Darstellung einer Prozess-Automatisierung

Die Prozessdokumentation und -analyse ist kein Selbstzweck, sondern findet immer in direkter Anlehnung an die Fragestellung statt. Ansonsten ist die Gefahr groß, dass man sich im liebevollen Modellieren größerer „Tapeten“ verliert und dabei das eigentliche Pro-

jektziel aus den Augen verliert. Im vorliegenden Fall wurde das vorab erstellte Flussdiagramm mit den Teilnehmern durchgesprochen und um potenzielle Fehlerquellen ergänzt. Ein weiterer zentraler Punkt dieses Schrittes war die genaue Untersuchung der verfügbaren Dateninformationen, um herauszufinden, welche Details aus der Datenbank gezogen werden könnten. So wurde beispielsweise diskutiert, ob die Zugangsberechtigungen eindeutig aus der Bereichszugehörigkeit der neuen Mitarbeiter abgeleitet werden könnten oder ob aus der Zuordnung zu einem Team automatisch eine Kostenstellen-Nummer abgeleitet werden kann.

Auf dieser Basis wurde dann von den IT-Dienstleistern ein Vorschlag vorbereitet, welche Teilschritte wie weit automatisierbar sein würden und was das für die Datenpflege bedeutet. Individuell zu programmierende Features wurden mit entsprechenden Implementierungskosten versehen.

2.4 IMPROVE-Phase: Wie soll die Lösung aussehen?

Bei der Entwicklung der Lösungen sind Kreativitätstechniken wie Brainstorming, Brainwriting nützlich – insbesondere Anti-Brainstorming kann helfen, einen Schritt weiterzukommen: Es fällt häufig einfacher, die Maßnahmen zu benennen, die alles noch viel schlimmer machen würden, und anschließend daraus dann die Verbesserungsmaßnahmen zu erkennen.

Bei der Priorisierung der Lösungen helfen beispielsweise Portfolio-Diagramme wie die Aufwand-Nutzen-Matrix – hier können anhand von frei wählbaren Kriterien-Dimensionen (wie beispielsweise Häufigkeit der Ausführung und Anwender-Zufriedenheit oder eben Aufwand und Nutzen) die einzelnen Anforderungen oder Features untereinander abgewogen werden, woraus man eine Rangreihung und damit Priorisierung der Lösungsvorschläge erhält. Das funktioniert sowohl für einzelne Bereiche (um mehrere Lösungen derselben Anforderung gegeneinander abzuwägen) als auch für den gesamten Projektumfang über alle Anforderungen (ein gutes Instrument im Umgang mit dem ‚Feature Creep‘).

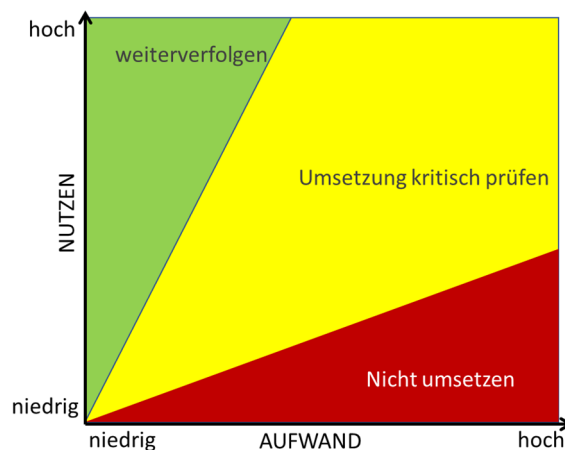


Abb. 4: Aufwand-Nutzen-Matrix [nach Lu14]

Im vorliegenden Fall wurde vor allem die Frage der Benachrichtigung und Eskalation lebhaft diskutiert. Bei der Automatisierung von Workflows muss immer individuell geklärt werden, welche Informationen Holschuld und welche Bringschuld sind. Es ist verführerisch, vom System über alles benachrichtigt werden zu können. Die Benutzer übersehen dabei häufig die Gefahr, dass wichtige und dringende Informationen in der Benachrichtigungsflut untergehen. Hier prallen gelegentlich auch unterschiedliche Persönlichkeiten aufeinander. Gleiches gilt für die Eskalation bei Nichterledigen: Wie oft und wann wird die zuständige Person erinnert? Ab wann geht eine Information an fachliche Vorgesetzte?

Ein zweiter in jedem Automatisierungsprojekt abzuwägender Aspekt ist das mögliche Vorbefüllen von Formularen mit Werten. Das spart einerseits Zeit und vermeidet Fehler, andererseits erfordert es eine hohe Disziplin bei der Stammdatenpflege und erlaubt keine Sonderfälle. Auch hier wurden die verschiedenen Möglichkeiten diskutiert und unter Beachtung der jeweiligen Implementierungskosten gegeneinander abgewogen.

Die Ergebnisse flossen in das Angebot des IT-Dienstleisters ein

2.5 CONTROL-Phase: Nach der Verbesserung ist vor der Verbesserung

So langweilig das für die Meisten ist: eine gute Dokumentation ist wichtig! Das hilft nicht nur im aktuellen Projekt, sondern auch in der sich anschließenden Betriebsphase. Ein Dashboard, das in der Post-Release-Phase Fehlermeldungen und Bugfixing-Zeiten anschaulich darstellt (am besten integriert in das eigene Ticketsystem), dient der allgemeinen Übersicht über den Stand. Das nützt allerdings alles nichts ohne ein designiertes Team aus entsprechend geschulten Mitarbeitern, das sich auch um die Behebung der identifizierten Probleme kümmert. Gerade in IT-Projekten ist der Übergang vom Projekt in den Betrieb oft fließend, daher müssen neben den Abnahmekriterien für das Projekt auch die Bedingungen der Betriebsphase (wie z.B. Service-Level-Agreements) und die Vorgehensweise bei weiteren Systemänderungen frühzeitig definiert und schriftlich festgelegt werden.

Bei Digitalisierungsprojekten (wie bei den meisten IT-Projekten) steht man vor der Herausforderung, dass sich die wahre Nützlichkeit des Projekts häufig erst nach einer gewissen Einschwingphase einstellt. In der Regel ändern sich dabei nicht nur Prozesse und Werkzeuge, sondern auch Regeln, Arbeitsanweisungen und Begleitdokumente. Das erfordert insbesondere von langjährigen Mitarbeitern, gewohnte Verhaltensweisen umzustellen – ein Umstellvorgang, der nicht über Nacht funktioniert. Die Gesamtdauer der Umstellphase bemisst sich aus der Dauer der individuellen Umstellungen, den kritischen Pfad bestimmen die Langsamsten. Daher ist es sinnvoll, gleich mit Übergabe des Projekts in den Betrieb eine Evaluationsperiode und eine anschließende Bewertung des bisher Umgesetzten zu vereinbaren. Häufig zeigt sich erst im operativen Betrieb, ob Formulare und Abläufe den Arbeitsalltag angemessen widerspiegeln. Wichtig ist aber, dass die Prozessbeteiligten sich in der Evaluationsphase konsequent an den vereinbarten Ablauf halten und bei Fehlern genau dokumentieren, wann und warum der geänderte Prozess nicht funktioniert.

Auch im vorliegenden Projekt stellte sich heraus, dass das angepeilte Projektziel (alle Aufgaben sind zum ersten Arbeitstag erledigt), nicht in jedem Fall gewährleistet war. Wie sich herausstellte, lag das in allen beobachteten Fällen an einer nicht konsequent gelebten Stellvertreter-Regelung. In diesem Fall waren also sowohl der definierte Prozess als auch das

gemeinsam entwickelte Formular für den gewünschten Zweck ausreichend, die erforderlichen organisatorischen Maßnahmen hinkten aber hinterher.

3 Zusammenfassung

Lean Six Sigma kommt zwar aus der produzierenden Industrie, bietet aber auch für IT-Projekte viele spannende Ansätze, die dabei helfen können, IT-Projekte erfolgreicher umzusetzen. Wenn man sich hierbei ein gewisses ‚Cherry-Picking‘ gestattet, kann man aus Lean Six Sigma ein schlankes Toolset ableiten, das einen standardisierten Rahmen für IT-Projekte bietet: „Es ist nicht das Ziel, alle Techniken in jedem Projekt einzusetzen“ [Ad12]. Vorschläge und Ansätze hierzu finden sich beispielsweise in [Lu14] und [Ge05].

Literaturverzeichnis

- [Ad12] Adam, M.; Waldner, H.: Erfolgreiche IT-Projekte mit Lean Six Sigma, https://www.fhkufstein.ac.at/content/download/3524238/11483487/file/Adam_Waldner_IT-Projekte%20mit%20LSS_Symposium%202012.pdf, Stand: 25.07.2019.
- [Fe19] Fehlmann, T.: Statistical Process Control for Software Development – Six Sigma for Software revisited, <http://www.e-p-o.com/downloads/sixsigmarevisited.pdf>, Stand: 25.07.2019.
- [Ge05] George, M. L. et al: The Lean Six Sigma Pocket Toolbook. McGraw-Hill, 2005.
- [Lu14] Lunau, S. (Hrsg.); Meran, L et al.: Six Sigma + Lean Toolset. Springer Gabler, 2014 (5. Aufl).
- [Mi19] Mikusz, M.; Herzwurm, G: Adaption, Umsetzung, Grenzen und Nutzen von Six Sigma in der Softwareentwicklung, Universität Stuttgart, https://www.researchgate.net/publication/221278571_Adaption_Umsetzung_Grenzen_und_Nutzen_von_Six_Sigma_in_der_Softwareentwicklung, Stand: 25.07.2019
- [PE16] PEEC: Lean Six Sigma Green Belt / Black Belt. Zertifizierungs-Trainingsunterlagen der Productive Engineering Europe Company, 2016.
- [Wh03] Wheat, B. et al: Leaning into Six Sigma. A Parable of the Journey to Six Sigma and Lean Enterprise. McGraw-Hill, 2003.