



[Künstler Burgy Zapp](#)

von Paul Conte

Lernen Sie von erfahrenen Software Projektmanagern

Mit der Vielzahl neuer Technologien und geschäftlicher Anforderungen - Web- und mobile Anwendungen, neue Sprachen wie Java, PHP und Ajax sowie SOA (Service Oriented Architecture) um nur ein paar zu nennen - hat die Komplexität von Anwendungssoftware deutlich zugenommen, was die ohnehin schon herausfordernden Projektmanagement-Aufgaben zusätzlich erschwert. Nach grundsätzlichen Überlegungen in Teil 1 dieses Artikels (NEWSolutions Märzausgabe 2007) beschreibt der Autor die Fokussierung in Teil 2 (NEWSolutions Juniausgabe 2007). Im heutigen dritten Teil werden Risikominimierung, Terminaussagen und weitere Aspekte beleuchtet.

Die erfolgreiche Handhabung eines Softwareentwicklungsprojekts erfordert eine effektive Handhabung technischer, finanzieller, personeller, „politischer“ und gelegentlich auch gesetzlicher Herausforderungen. Es gibt allerdings nicht viele Projektmanager, die sich als Experten in all diesen Bereichen bezeichnen können. Das primäre Geheimnis einer erfolgreichen Projektabwicklung liegt in der Anwendung von Praktiken, die stabil laufenden Code erzeugen sowie in der Vermeidung wesentlicher Risiken und dem Lernen aus Erfahrungen. Mit diesen Praktiken wollen wir uns näher befassen, um ein paar weitere Geheimnisse des Software Projektmanagements aufzudecken.

Risikominimierung statt Ergebnismaximierung

Viele Projektleiter, die Programmierer sind oder waren (aber durchaus auch andere Projektleiter), sehen ihr Gesamtziel in der Maximierung des Gesamtergebnisses nach dem Motto: Liefere das maximale Ergebnis in der kürzesten Zeit mit den geringsten Kosten. Das aber ist definitiv die falsche Strategie!

Viele Projekte scheitern, weil sie sich in großem Umfang aufblähen. Es werden entweder die

falschen Dinge erstellt oder das Projekt verzögert sich gröblich oder überschreitet das geplante Budget erheblich und die angestellten Schätzungen gelangen im Verlauf des Projekts niemals in die Nähe der Realität. Projekte scheitern nicht, weil eine nützliche, aber eigentlich nicht wesentliche Funktion nicht erstellt wurde oder weil die Benutzeroberfläche vielleicht etwas klobig geraten ist. Sie scheitern ebenso nicht, weil der Zeitplan und das Budget sich in einer Reihe revidierter Schätzungen im Verlauf des Projektfortschritts in moderatem Umfang erweitert haben.

Anstatt sich der Illusion hinzugeben, alles sei akkurat zu schätzen, perfekt zu planen und es würden keine wesentlichen Probleme auftreten, sollte man die Risiken erkennen und darauf vorbereitet sein, indem man:

- für jedes Risiko eventuelle Lösungsmöglichkeiten einplant
- Projektaktionen vorsieht, um die Wahrscheinlichkeit und Auswirkungen eines Risikos zu begrenzen
- Aktionen vorsieht, um die Wahrscheinlichkeit und Auswirkungen eines Risikos fortlaufend zu verfolgen

So ist beispielsweise der Verlust eines Entwicklers, der eine Schlüsselrolle im Projekt einnimmt, in vielen Projekten ein immer wieder auftretendes Risiko. Dem kann man z. B. entgegenwirken, indem man für eine breit gestaffelte Ausbildung im Team sorgt oder ein Bonusprogramm für den erfolgreichen Abschluss eines Projekts aufsetzt. Beide Aktionen mögen als Verschwendung von Zeit und/oder Geld gesehen werden, wenn der seit langem tätige, zuverlässige Entwickler das Unternehmen nicht verlässt. Zielt man aber auf ein optimales Ergebnis, sind hier weder Zeit noch Geld wirklich verschwendet. Auf lange Sicht werden Projektmanager, die suboptimale Ziele mit substantiell geringeren Risiken setzen, deutlich seltener scheitern. Und in den Augen des Unternehmens gelten solche Projektmanager als erheblich erfolgreicher als jene Projektmanager, die einen Mix aus „optimalen“ und gescheiterten Projekten vorzuweisen haben.

Sagen Sie niemals „10 Wochen“, wenn die richtige Antwort „Ich weiß es nicht“ wäre

Wie schätzen Sie die Dauer eines Projekts? Sie überlegen, wie lange es gedauert hat, etwas Ähnliches zu erstellen und versuchen herauszufinden, was in dem neuen Projekt zu tun ist, richtig? Was ist aber, wenn Sie vor Ihrer ersten ILE RPG CGI Web-Anwendung stehen und zuvor noch nie etwas Vergleichbares abgewickelt haben? Der beste Weg ist in einem solchen Fall, sich entsprechende Erfahrung zu verschaffen, indem man ein Pilotprojekt durchführt. Versuchen Sie, für das Pilotprojekt einen Funktionsumfang zu identifizieren, von dem Sie annehmen, dass sie ihn in ca. 5 Prozent der Zeit abwickeln können, die Sie in etwa für das Gesamtprojekt schätzen. Es sollte ein Projekt sein, das die entscheidenden „Unbekannten“ abdeckt, aber natürlich nicht die Hälfte der Zeit in Anspruch nimmt, die für die gesamte Produktionsversion aufgewendet werden muss. Konzentrieren Sie sich überdies nicht ausschließlich auf die Codierung. Die größten Unbekannten sind möglicherweise, eine Übereinstimmung mit den Endbenutzern bezüglich der Definitionsanforderungen zu erzielen.

Ihr Unternehmen wird möglicherweise die Notwendigkeit für ein Pilotprojekt in Frage stellen und auf einer groben Schätzung bestehen, bevor ein Pilotprojekt überhaupt in Erwägung gezogen wird. Bedenken Sie dabei sorgfältig, dass aus bisher noch nicht gänzlich erforschten genetischen Ursachen das menschliche Gehirn erhebliche Schwierigkeiten hat, die Begriffe „grob“ und „Schätzung“ zu behalten. Die Menschen werden unausweichlich behalten, dass Sie „6 Monate“ oder was auch immer gesagt haben und werden sich einer auf dem Pilotprojekt basierenden, signifikant längeren Schätzung gegenüber äußerst unaufgeschlossen verhalten. Zeigen Sie sich hart und bleiben Sie der Linie treu, die sagt: „Wir sind nicht in der Lage, eine Schätzung abzugeben, ohne über eine gewisse Erfahrung zu verfügen, auf der die Schätzung basieren kann.“

Nutzen Sie die Vorteile verfügbarer Tools

Es ist annähernd ausgeschlossen, zu viel Geld für gute Tools für ihr Entwicklungsteam auszugeben. Das ist darin begründet, dass Tools - verglichen mit der Verschwendung von Entwicklungszeit für Aufgaben, die sich mit Hilfe von guten Tools und Automatisierung schneller erledigen lassen - relativ billig sind.

Ein Vorbehalt ist allerdings, dass Tools ihre Hebelwirkung nur in einer Entwicklungskultur entfalten können, die sich durch rasche Akzeptanz besserer Verfahrensweisen und Tools auszeichnet. Ein Unternehmen, in dem die Entwickler sich nicht über die Handcodierung jeder einzelnen Zeile hinausbewegen, weil sie „SEU-Experten“ sind, wird nicht in der Lage sein, Vorteile aus guten Tools zu ziehen, gleichgültig, wie gut oder erschwinglich sie sind.

Befindet man sich in einer solchen Situation, wird es möglicherweise nicht gelingen, die Entwicklungskultur für das nächste in Angriff zu nehmende Projekt gänzlich zu verändern. Die Softwareentwicklung ist heute durch Technologiewechsel von in rascher Folge explosionsartig wachsendem Umfang charakterisiert. Daher sollte man intensiv daran arbeiten, seine Entwicklungsteams neu auszurichten und fortwährend praktikable, neue Tools zu identifizieren und einzusetzen, die die Entwicklung vereinfachen und beschleunigen können.

This content is available for purchase. Please select from available options.

- [7 Euro/Monat NEWSabo digital - sofort zugreifen.](#)
- [13,5 Euro/Monat NEWSabo plus inklusive 5x Login & Print-Ausgabe - sofort zugreifen.](#)

[Login & Purchase](#)