

# Value Driven Projects

## Ein Ansatz zur Umsetzung schlanker Projekte



**Es ist eine in der IT-Welt bestens dokumentierte Tatsache, daß ein bemerkenswert hoher Anteil an Entwicklungsprojekten scheitert.** Dieses Scheitern bedeutet in der Regel dramatische wirtschaftliche Schäden, die bei geschäftskritischen Lösungen um ein Vielfaches über die erhöhten Kosten der Projektabwicklung selbst, hinaus gehen. In Zeiten wirtschaftlichen Drucks werden Projekte vernünftiger Weise zunehmend nach deren Business Value selektiert. So steigt auch der Schaden, den krisenhafte oder scheiternde Projekte anrichten, dramatisch.

Obwohl die Suche nach dem „Stein der Weisen“ für erfolgreiche Entwicklungsprojekte seit jeher läuft, wächst vor dem geschilderten Hintergrund der letzten Jahre das Interesse an Abwicklungsformen, welche die Effizienz erhöhen und Risiken minimieren.

Beiträge dazu kommen aus allen beteiligten Disziplinen, vom klassischen Projektmanagement bis hin zum Software Engineering im engeren Sinne, also Entwicklungsmethodik- und Technologien. Teilweise wurden Disziplinen, die seit Jahrzehnten vernachlässigt waren, professionalisiert und in die Projektabwicklungen systematisch integriert, allen voran der Softwaretest.

Auf der Vorgehenseite wurde Vieles versucht, von genauer Standardisierung und Formalisierung des Projektmanagement- und Controlling bis hin zur Agilen Softwareentwicklung.

Während sich die Auftraggeber- und Fachbereichsseite häufig – auch gezwungen durch rechtliche Vorschriften – auf Ausschreibungs- und Pflichtenheftformalisten konzentrieren, um so ihr Risiko zu minimieren, setzt die Lieferseite häufig auf aktuelle technologische und vorgehens-technische „Ingenieurs-Heilsversprechen“. Im schlimmsten Fall zerfallen in solchen Konstellationen Projekte in einen Projektmanagementteil, der die fachliche und kaufmännische Steuerung des Projektes leisten soll und einen inhaltlichen Teil, der die technische Lösung herstellen soll. Nicht selten sind dabei Doppel-Projektmanagements anzutreffen, formalisiert oder informell: Ein „fachlicher Projektleiter“ und ein „technischer Projektleiter“.

Derartige komplexe Gebilde tendieren zu innerem Zerfall, die Frage des Ineinandergreifens der beiden Teile – kaufmännisch und technisch - bleibt häufig ungelöst und Synergien der einzelnen Disziplinen und Arbeitsschritte werden kaum genutzt. Im worst case befriedigt also ein „fachlicher Projektleiter“ die formalen Controllingbedürfnisse und läuft einem außer Kontrolle geratenen agilen Entwicklerteam, in dem in Personalunion sämtliche Rollen des Entwicklungsprozesses wahrgenommen werden, hinterher, das seinerseits den Einblick in die kaufmännischen Ziele und Zusammenhänge des ursprünglichen Auftrages und der geforderten Lösung längst verloren hat.

Es geht dabei nicht um den Wettstreit zwischen „Wasserfall“ und „Iteration“. Es geht um die Frage, ob während der gesamten Projektabwicklung die eigentliche letztlich IMMER kaufmännische Zielsetzung des Projektes im Fokus bleibt. Das gelingt am ehesten, indem Projekte möglichst schlank abgewickelt werden.

Die Kernidee dabei ist, daß jede am Geschehen beteiligte „Zelle“ mit dem Wissen um die angesprochene wirtschaftliche Zielsetzung des Projektes ausgestattet an die Arbeit geht. Für jede Entscheidung im Projekt muß die Frage nach dem Business-Value gestellt werden, was bis tief in Entwicklerentscheidungen hinein reicht. Mit vertretbarem Aufwand wird dies aber nur in „Lean Projects“ gelingen.

**Was sind wesentliche Charakteristika von „Lean Projects“:**

**Eine streng am Business-Value orientierte Auswahl an Anforderungen und Features.**

Egal ob vorab detailliert spezifiziert oder iterativ identifiziert und entwickelt, der Ausgangspunkt jeder schlanken Projektabwicklung ist Konzentration auf das Wesentliche (eigentlich „das Wertschöpfende“) bei der Festlegung der Anforderungen und geforderten Features.

**Eine intelligente Portionierung in beherrschbare Teilaufgaben**

Diese Kompetenz stellt nach wie vor eine Königsdisziplin im Projektmanagement und auch in Software-Architektur und Design dar. Wesentlich im Zusammenhang mit „Lean Projects“ ist es, die Frage der Beherrschbarkeit (Planbarkeit, Steuerbarkeit) für jede Teilaufgabe gezielt zu stellen und das Ergebnis der Teilung erst zu akzeptieren, wenn diese Frage positiv beantwortet werden kann.

**Eine Technologie-Auswahl, die der Problemstellung angemessen ist.**

Nicht selten steht die technische Komplexität aufgrund von Architekturvorgaben oder Technologiepräferenzen in keiner passenden Relation zum wirtschaftlichen Beitrag der zu entwickelnden Applikation.



„State of the Art“ ist dabei nicht selten ein verlockender Irrweg, es wird mit Zukunftssicherheiten argumentiert, die aufgrund der raschen Technologiefortschritte niemals ausgeschöpft werden können (weil rasche Innovation auch rasches Veraltern bedeutet und somit „state of the art“ Systeme auch rasch zu legacy systems werden!). Durchschaubarkeit und Verwertbarkeit von unspezifischem Informatikwissen in der Wartung der Applikationen ist weit entscheidender als technologisches „State of the art“ für das in der Zukunft das Know-how fehlen wird.

**Ein kompaktes Team.**

Für jede Teilaufgabe ist ein Minimum an Personal einzusetzen, ohne unvereinbare Rollen in Personalunion zu belassen. Was in dieser Frage möglich ist und was nicht hängt naturgemäß stark von den handelnden Personen ab, es gibt aber Tendenzen: In den seltensten Fällen sind z. B.: die Rollen „Anforderer“ und „Umsetzer“ vereinbar ohne die Wirtschaftlichkeit des Projektes zu gefährden. Ähnliches gilt zwischen Analyse und Test, Entwicklung und Test, um nur Beispiele zu nennen.

**Eine straffe Menge an Vorgehensregeln.**

Ein exzellentes Projektteam tritt an eine neue Aufgabe mit einem klaren Plan heran, in welchen Vorgehensschritten die Lösung spezifiziert, realisiert, getestet und ausgeliefert werden wird. Ein guter Richtwert für Regel-Mengen ist dabei: Was nicht im Kopf behalten werden kann, ist zu viel.

**Die Erschließung von Synergien über den Entwicklungszyklus.**

Viele Aufgaben im Entwicklungszyklus können vom Aufwand her minimiert werden, indem davor liegende Arbeitsschritte so ausgeführt werden, daß sie spätere direkt „speisen“. Prominentestes Beispiel an dieser Stelle: Synergien zwischen Requirements-Engineering und Testen (Ausrichten der Anforderungsspezifikation am späteren Test, was sowohl für präzisere Spezifikationen sorgt als auch die Testaufwände drastisch reduzieren kann).

Alle diese Charakteristika dienen der ursprünglichen Zielsetzung, die „kaufmännische Frage“ allgegenwärtig zu halten, mit anderen Worten also das Projekt beherrschbar zu halten. Daß Ressourcenlimitierung – also „Schlankheit“ mit anderen Worten – ein funktionierender Rahmen zur Herstellung komplexer, optimal angepasster und effizienter Lösungen ist, ist uns in der Evolution in ausreichender Menge bewiesen worden.

Der Ansatz „Lean Projects“ folgt dieser „simplen“ Strategie und erhöht, kompetent umgesetzt, die Erfolgchancen für Entwicklungsprojekte.



Verfasser **Ernst Lieber** ist Leiter des Geschäftsfelds Softwareentwicklung bei ANECON Software Design und Beratung G.m.b.H.

