

## Kapitel 2

# Das Agile Manifest

Branchen wie etwa die militärische oder pharmazeutische Software-Entwicklung, die Verantwortung für Leib und Leben tragen, können nicht auf ausführliche Dokumentation verzichten. Das verlangen zu Recht die zuständigen Behörden. Es handelt sich jedoch um eng abgegrenzte Felder. In der Mehrzahl der Software-Projekte jedoch sollte die Entwicklung funktionierender Software im Mittelpunkt stehen, was Kritiker wie Ward Cunningham, Martin Fowler, Alistair Cockburn oder Kent Beck betonten. Dies führte 2001 zur Verkündung des „Manifesto for Agile Software Development“, des Agilen Manifests.

Das Agile Manifest ist eine Reaktion der Entwickler auf die sog. „schwergewichtigen“ Prozess- bzw. Vorgehensmodelle, wie den Unified Process (UP) (Jacobson et al. 1999) von Rational (jetzt IBM) oder das V-Modell,<sup>14</sup> das Vorgehensmodell der deutschen Bundesbehörden. Obwohl beide Ansätze sehr unterschiedlich sind – UP beleuchtet den iterativ-inkrementellen Software-Entwicklungsprozess, während das V-Modell die Sicht des Kunden auf das Software-Projekt in den Vordergrund stellt – gelten beide als sehr „dokumentenlastig“, was zumindest bei „Anfängern“ erheblichen zusätzlichen Aufwand bedeutet. Nun muss man wissen, dass Software-Entwickler ihren Beruf wählen, um Software-Lösungen zu designen und zu implementieren. Allzu viel Dokumentation ist ihnen eher lästig. Es braucht üblicherweise große Überzeugungsarbeit hinsichtlich der Notwendigkeit einer aufwändigen Dokumentation, wie der Autor aus eigener Erfahrung weiß.

Das mittlerweile weltweit unterstützte Agile Manifest (Beck et al. 2001) zeigt einen Ausweg aus dieser Problematik. Es definiert vier Leitsätze für eine erfolgreiche und effiziente Software-Entwicklung:

- Die individuellen Beteiligten und ihre Interaktion sind wichtiger als Prozesse und Werkzeuge.
- Die Ablieferung lauffähiger Software ist wichtiger als umfassende Dokumentation.
- Die Zusammenarbeit mit dem Kunden ist wichtiger als Vertragsverhandlungen.

---

<sup>14</sup>Zum aktuellen V-Modell XT siehe (Höhn u. Höppner 2008) oder (Friedrich et al. 2009), zum „alten“ V-Modell 97 siehe (Dröschel u. Wiemers 2000).

- Auf Änderungen der Anforderungen einzugehen ist wichtiger als die sture Verfolgung eines Plans.

Die agilen Leitsätze liest man sinnvollerweise „von rechts nach links“:

- Prozesse und Werkzeuge sind wichtig, aber wichtiger sind im Zweifelsfall die am Projekt Beteiligten und ihre Interaktionen.
- Dokumentation (auch eine umfassende) ist wichtig, aber im Zweifelsfall hat die Ablieferung einer lauffähigen Software Priorität.
- Verträge sind wichtig, aber im Zweifelsfall ist die Zusammenarbeit mit dem Kunden wichtiger.
- Pläne sind wichtig, aber wenn es erforderlich ist, müssen die Anforderungen eben im Sinne des Kunden geändert (also angepasst) werden.

Besonders interessant ist die Interpretation des zweiten Leitsatzes: *Dokumentation (auch eine umfassende) ist wichtig, aber im Zweifelsfall hat die Ablieferung einer lauffähigen Software Priorität.* Dies heißt eben nicht, dass in agilen Projekten nicht dokumentiert werden muss. Es wird lediglich etwas über die Priorität der Dokumentation gesagt. Agile Projekte brauchen also eine besondere, flexible Art der Dokumentation. Statische Dokumente werden der Situation nicht gerecht. Die Lösung sind oft Wikis oder kommerzielle Dokumenten-Management-Produkte, die eine Versionierung und damit eine leichte Änderbarkeit der Dokumente ermöglichen. Mittels einer elektronischen Unterschrift lässt sich auch das Freigabeverfahren der Dokumente vereinfachen. Somit ergibt sich die Möglichkeit, agile Modelle auch im sog. „regulierten Umfeld“<sup>15</sup> einzusetzen.

Zu Verträgen und Plänen sei gesagt, dass bei Änderungen und Anpassungen, insbesondere in späten Phasen, natürlich das Wohlwollen beider Parteien, des Kunden und der Entwickler, gefragt ist: Die Entwickler müssen wissen und sich dessen auch sicher sein, dass der Kunde sie mit seinem Änderungswunsch nicht einfach nur „ärgern“ will, sondern in der Regel ein vitales Interesse hat. Der Kunde muss sich seinerseits im Klaren darüber sein, dass sowohl Spezifikations- wie auch und Planänderungen einen erheblichen Mehraufwand bedingen, der auch bezahlt sein will. Insbesondere bei Festpreisangeboten muss hier ggf. nachverhandelt bzw. der Funktionsumfang im Rahmen der Iterationsplanung neu definiert werden. Wichtig ist bei alledem, dass niemals die Qualität des Software-Produkts unter solchen Änderungen leidet. Dies muss durch vorher definierte Testverfahren (insbesondere auch durch automatisiertes Testen) ausgeschlossen werden.

Das Agile Manifest stellt mit seinen vier Leitsätzen eine Art Randbedingungen der leichtgewichtigen Software-Entwicklung dar. Doch als Durchführungshilfen in einem konkreten Prozess sind sie weniger geeignet. Die diversen Prozessmodelle, wie die in der Folge diskutierten, haben die Aufgabe, die agilen Leitsätze umzusetzen.

---

<sup>15</sup> Von einem regulierten Umfeld spricht man bei von Behörden kontrollierten Projektumgebungen.

## Literatur

Beck K et al. (2001) [www.agilemanifesto.org](http://www.agilemanifesto.org). Zugriff Oktober 2009

Dröschel W, Wiemers M (2000) Das V-Modell 97. München: Oldenburg

Friedrich J, Hammerschall U, Kuhrmann M, Sihling M (2009) Das V-Modell XT, New York, NY: Springer

Höhn R, Höppner S (2008) Das V-Modell XT, Berlin: Springer

Jacobson I, Booch G, Rumbaugh R (1999) Unified Software Development Process. Reading, MA: Addison-Wesley