



□ Michael Hüttermann

[E-Mail: [michael@huettermann.net](mailto:michael@huettermann.net)]

ist Dipl.-Wirt.-Inf. (Java Champion, SCJA, SCJP, SCJD, SCWCD). Er ist freiberuflicher Entwickler, Architekt, Berater, Coach, Autor und Dozent für Java/JEE, ALM/SCM und agile Softwareentwicklung. Er war verantwortlicher Stage-Producer für den Tooling-Track der Agile 2009, ist Treiber der Java User Group Köln, java.net JUGs Community Leader, Mitglied der Agile Alliance, im Board der JetBrains Academy, Committer von FEST sowie Java-Contributor. Er schreibt Bücher und spricht auf Konferenzen. Folgen Sie ihm auf Twitter: <http://twitter.com/huettermann>.

## objektspektrum themenspecial: agility

### Fast 10 Jahre Agile Manifest: Agilität ist Mainstream!

Vor fast zehn Jahren wurde das Agile Manifest mit seinen vier Wertepaaren und 12 Prinzipien formuliert. Neben den dort propagierten Werten und Prinzipien existieren agile Strategien, die nicht mehr nur ausschließlich im Werkzeugkasten des Agilisten einen Platz gefunden haben. Agile Vorgehensmodelle und Rahmenwerke sind mittlerweile recht verbreitet, und auch reichhaltige Vorgehensmodelle nutzen immer häufiger agile Ansätze und adaptieren agile Strategien. Es gehört nun fast schon zur Etikette, den „Agilitätsknopf“ gut sichtbar am Revers zu tragen. Agilität ist zum Mainstream geworden. Doch nicht jeder, der von sich behauptet, agil zu arbeiten, tut das auch. Gleichwohl lässt allein dieses Bekenntnis schon erhoffen, dass Software hier mit höherer Erfolgsaussicht erstellt wird.

Selbstwahrnehmung und Reflexion gehören zu den essenziellen Aspekten der agilen Softwareentwicklung. **Henning Wolf** und **Arne Roock** diskutieren in diesem Themenspecial die Rolle von „Feedback“ für die erfolgreiche Softwareentwicklung.

Feedback ist natürlich auch in großen Projekten nötig. Eine Arbeit in großen, verteilten Teams setzt zudem einen Prozess voraus, der skaliert. Bekannt ist schon länger, dass agile Ansätze auch (oder gerade?) in großen Projekten helfen können, existente Defekte sichtbar zu machen und zu adressieren. **Werner Schoepe** zeigt, ob und wie Agilität skaliert, und stellt Lösungsansätze und Werkzeuge vor. Auch **Clemens Reijnen** und **Stefan Mieth** thematisieren Werkzeuge. Sie illustrieren in ihren Beiträ-

gen Prozessvorlagen zur werkzeugunterstützten Implementierung von Scrum. Ebenso handelt der Erfahrungsbericht von **Lars Roith**, „Distributed Scrum“, von eben diesem, wohl bekanntesten agilen Rahmenwerk.

Eine suboptimal an Projektphasen, Rollen oder Werkzeugen ausgerichtete Vorgehensweise kann zu Silos führen – unabhängig davon, in welchem Vorgehensmodell man unterwegs ist. Häufig wird eine Ausdehnung der agilen Herangehensweise auf möglichst alle Phasen angestrebt, um diese gleichzeitig eng zu verweben. Als Ergebnis verspricht ein Application Lifecycle Management (ALM) mehr Kundennähe, bessere Reaktionszeiten auf sich ändernde Anforderung sowie eine

erhöhte Wahrscheinlichkeit, die Software in guter Qualität bereitzustellen. Werkzeugagnostische Strategien sind dabei der erste Schritt, deren Implementierung mit konkreten Werkzeugen ein nächster. Wie ein Ansatz aussehen kann, zeigt Ihnen **Jürgen Pilz** in seinem Beitrag.

Ein ALM, das suggeriert bereits der Begriff, tritt dem häufig zu kurzen Schritt entgegen, Projektherausforderungen primär auf die eigentliche Softwarekonstruktion (Entwicklung) zu projizieren. Aber auch andere Projektphasen haben entscheidende Relevanz und beeinflussen die Wahrscheinlichkeit des Projekterfolgs signifikant. Eine solche Phase ist das Anforderungsmanagement. Ein (agiles) Projekt steht und fällt mit einem soliden

Anforderungsmanagement, auch wenn agile Vorgehen nicht immer explizite Aussagen über diese Phase, deren Rollen (Verantwortlichkeiten), Produkte (Ergebnistypen) und Aktivitäten treffen. Dieses Online-Themenspecial widmet sich auch Ansätzen, die das Anforderungsmanagement noch intensiver in die „agile Softwareentwicklung“ miteinbeziehen. **Alexander Merten**, **Robert Stevenson** und **Sabine Wildgruber** übertragen Methoden des Anforderungsmanagements auf das agile Terrain. Die Brücke schlägt auch **Peter Maurer** in seinem Beitrag über Best Practices für die Anforderungsmodellierer in agilen Projekten. Schließlich erläutert **Dr. Claudia Hirschmann**, wie es möglich ist, in der

durch besondere gesetzliche Vorgaben geprägten Medizintechnik, einen agilen Umgang mit Anforderungen einzuschlagen.

Es ist längst kein Geheimnis mehr, dass sich reichhaltige und agile Vorgehensmodelle nicht gegenseitig ausschließen müssen. Viele Projekte haben positive Erfahrungen damit gemacht, die eigentliche Konstruktion der Software „agil“ zu gestalten und diese in eine traditionelle, umschließende Wirtsstruktur einzubetten. Früher häufig tituliert als „im Geheimen agil“, muss dies mittlerweile keine Schattenswelt mehr sein. Eine Koexistenz ist salonfähig geworden und kann zu sehr positiven Ergebnissen führen. **Karol Frühauf**, **Thomas Haas** und **Oliver C.**

**Hoeffleur** gehen in ihrem Beitrag der Frage nach, ob wirklich die ganze Organisation agil arbeiten muss. Die Brücke, auf eine andere Weise, schlägt auch **Dr. Ernst Ambichl**. Er beschreibt Ansätze, mittels derer Altsysteme, die ohne Testschnittstellen vorliegen, testbar gestaltet werden können.

Dieses Themenspecial liefert Ihnen einen breit gefächerten Themenquerschnitt. Es bietet sowohl Beiträge zu eher traditionellen, fast zeitlosen Fragestellungen der agilen Softwareentwicklung als auch Beiträge zu sehr aktuellen Themen.

Ich wünsche Ihnen viel Spaß bei der Lektüre!

*Michael Hüttermann*