
Geleitwort

Als wir das Manifest für Agile Softwareentwicklung verfassten, handelte es sich bei den Personen, die uns unterstützten, um eine kleine Minderheit, die versuchte, eine Branche zu verändern. Heute, 20 Jahre später, ist »agil« im Mainstream angekommen. Aber ich schreibe »agil« nicht ohne Grund in Anführungszeichen – viele Leute sagen, dass sie agile Softwareentwicklung betreiben, und die meisten glauben das auch wirklich. Doch ihre Handlungen haben wenig Ähnlichkeit mit der gemeinsamen Vision, die wir vor zwei Jahrzehnten teilten.

Die Wahrheit ist, dass eine agile Arbeitsweise ein Netz miteinander verbundener Praktiken erfordert, die sowohl das Management als auch die technische Ausführung der Softwareentwicklungsarbeit umfassen. Viele dieser Praktiken, insbesondere die technischen, sind nicht gut verstanden oder werden nicht allgemein gelehrt. Infolgedessen haben zu viele eine verfälschte Vorstellung davon, wie effektiv die Entwicklung von Softwareprodukten sein kann.

James Shore war einer der frühen Pioniere auf dem Weg zu Extreme Programming, einer zentralen Säule der agilen Bewegung. Seine erste Auflage dieses Buches war einer meiner Favoriten: ein Handbuch für Teams, das ihnen das nötige Wissen vermittelte, um einen agilen Prozess richtig durchzuführen. Später arbeitete er mit Diana Larsen zusammen, um das Agile Fluency Model zu entwickeln – ein Modell, das ihre Erfahrungen festhält, auf welche verschiedenen Arten Menschen ihre Fähigkeiten im Umgang mit agilen Ansätzen entwickeln können. In diesem Modell bietet eine einfache Anwendung von Projektmanagementtechniken, die oft als grundlegender Scrum-Ansatz bezeichnet wird, einen gewissen Wert, indem sie sich auf die Kundenbedürfnisse konzentriert. Es fehlen aber die technischen Fähigkeiten, die Sie benötigen, um die hohe Produktivität und Zuverlässigkeit auszuschöpfen, die viele Teams erreichen.

Diese Sichtweise bestimmt zu Recht die Struktur dieses Buches, das den Schwerpunkt darauf legt, wie Sie sich auf die Wertstiftung konzentrieren und wie Sie diesen Wert zuverlässig liefern. Ausrichtung an der Wertstiftung heißt, dass Sie die Bedeutung einer starken Teamarbeit verstehen, Fähigkeiten zur adaptiven Planung entwickeln und eng mit den Kunden und Benutzerinnen der entstehenden Software zusammenarbeiten. Zuverlässig liefern konzentriert sich auf

wesentliche technische Praktiken für Tests, Refactoring, Entwurf und kollaborative Entwicklung. Dabei wird die oft kontraintuitive Vorstellung berücksichtigt, dass die Erstellung von Software mit hoher interner Qualität die Kosten senkt und die Geschwindigkeit der Codebereitstellung erhöht. Kombiniert mit einer DevOps-Kultur und Continuous Delivery ermöglicht dies eine hohe Frequenz, Funktionen schnell in Produktion zu bringen, was wiederum den Teams die Möglichkeit gibt, mehr darüber zu erfahren, was wertstiftend ist, indem sie beobachten, wie die Software in der Praxis genutzt wird.

Ich hatte das Glück, vor 20 Jahren bei Thoughtworks ein Zuhause zu finden, wo unsere Teams diese Art von Fähigkeiten nutzen, um unsere Kunden bei der Entwicklung neuer Softwareprodukte zu unterstützen und alte Traditionen abzulösen. Wie James haben auch wir festgestellt, dass Extreme Programming eine solide Grundlage für unsere Arbeit darstellt, und wir haben diese Techniken in den letzten zwei Jahrzehnten mit großem Erfolg angewandt. Ich freue mich daher sehr darüber, dass James ein weiteres Jahrzehnt seiner Coaching-Erfahrung in die Überarbeitung seines Buches eingebracht hat. Das Ergebnis ist eine solide Grundlage für das Erlernen dieser Fähigkeiten, die uns so sehr geholfen haben. Wie alles, was sich lohnt, wird es Zeit brauchen, und auf dem Weg dorthin wird es Frustrationen geben. Aber dieser Leitfaden kann Ihnen auf dieser Reise helfen – weg von öden Zeremonien hin zu der Kraft, die wir empfunden haben, als James und ich diese Techniken vor all den Jahren zum ersten Mal angewandt haben.

– Martin Fowler
Chief Scientist, Thoughtworks