

EXTREME PROGRAMMING (XP)

— FÜR SCRUM — MASTER UND PRODUCT OWNER

SCRUM-IMPLEMENTATION MIT XP-PRAKTIKEN
EFFIZIENTER GESTALTEN



BILL WALKER

INHALT

Vorwort

Die XP-Grundlagen

Warum XP?

XP-Werte

Einfachheit

Kommunikation

Feedback

Respekt

Mut

XP-Regeln

XP-Praktiken

Teamverständnis entwickeln (Sit together)

Funktionsübergreifende Teams (Whole Teams)

Transparente Arbeitsplätze (Informative Workspace)

Committete Arbeit (Energized Work)

Paarprogrammierung (Pair Programming)

User Storys (Storys)

Wöchentliche Zyklen (Weekly Cycles)

Quartalsweise Zyklen (Quarterly Cycle)

Leerlauf (Slack)

Zehn-Minuten-Entwicklung (Ten-Minute Build)

Kontinuierliche Integration (Continuous Integration)

Testgetriebene Entwicklung (Test-First Programming)

Inkrementelles Design (Incremental Design)

Real Customer Involvement

Incremental Deployment

Team Continuity
Shrinking Teams
Root-Cause-Analyse
Shared Code
Single Code Base
Daily Deployment
Negotiated Scope Contract
Den Kunden ins Team integrieren
Das Planning Game

XP-Rollen

Tester
Architekten
Anforderungsmanager (Requirements Engineers,
Interaction Designers)
Programmierer
Projektleiter
Product Manager
HR
Technische Dokumentation
Benutzer
XP-Coach
Führungskräfte

XP-Events

Iteration Planning
Daily Standup Meeting

XP-Artefakte

Story Cards
 INVEST-Pattern
Task Cards
Der "War Room" (Artefakt?)

XP einführen

XP skalieren

Als Scrum-Master in der Scrum-Guide-Falle

XP und Scrum kombinieren

Unterschiede von Scrum und XP

Scrum durch XP ergänzen

Der Scrum-Master

Der Product Owner

Entwickler-Team/Developer:innen

Sprint

Sprint Planning

Daily-Scrum

Sprint Review

Product Backlog

Sprint Backlog

Increment

XP-Praktiken

Literaturliste

Vorwort

Als Product Owner oder Scrum-Master haben Sie womöglich keine eigenen Erfahrungen als Entwickler und das ist vollkommen in Ordnung. Es ist keine Anforderung an Ihre Rolle, dass Sie Software schreiben können oder irgendwelche Entwicklungsplattformen bedienen. Dafür gibt es in Scrum eine spezielle Rolle, deren Inhaber im Allgemeinen über viel Erfahrung in diesen Fragen verfügen.

Warum also sollte es für einen Product Owner oder Scrum-Master überhaupt von Interesse und Bedeutung sein, sich mit Entwicklungsmethoden und sogar einem ganz anderen Framework wie XP - Extreme Programming - auseinanderzusetzen? Dazu mag womöglich ein Teil aus einem Interview einen Anhaltspunkt geben: Jeff Sutherland, einer der Väter von Scrum, antwortete in einem Interview von PC Week Russia:

“How are the traditional and extreme programming methods combined in practice?”

It’s not traditional practice. SCRUM is a framework for teams using engineering practices outlined by the extreme programming. The first SCRUM team implemented all the extreme programming practices before extreme programming existed. In fact Kent Beck, who started extreme programming, asked me to send him all the information I had on SCRUM, and he had everything about SCRUM before he started working on extreme programming.

So when I got together in 1995 with Ken Schwaber to talk about moving SCRUM out into the industry, Ken felt we should just present SCRUM as the team framework: the product order, the SCRUM master, the team, the scrum meetings (how they work together), and the SCRUM artifacts (how you track and manage a SCRUM project). And he felt that if we did that, SCRUM could be implemented very quickly in two days and over time, people could improve the engineering practices using the continuous improvement that is embedded in SCRUM. And when they looked at removing the impediments that were blocking their progress, many of them would be engineering problems and they could look to extreme programming practices to help them with that.”¹

Wenn wir heute Extreme Programming genauer betrachten, so machen wir nichts anderes, als dass wir uns mit einem anderen Blickwinkel auf Scrum zubewegen. Wir fokussieren die Blickrichtung von Entwicklern und das nicht vonseiten irgendwelcher völlig fremder Ansätze und Techniken, sondern aus einem Blickwinkel, der seit seinen Anfängen mit Scrum interagiert hat und sich aus parallelen Quellen entwickelt und weiterentwickelt hat.

Die Auseinandersetzung mit XP ist damit nicht nur eine Möglichkeit, mehr über Erfolgselemente für Entwickler zu erfahren, sondern mehr über die Erfolgsfaktoren von Scrum an sich - als würde man einen lieben Freund bitten, einem von außen seine Eindrücke zu unserer Handlungsweise und unserem Tun zu geben, um darauf basierend bessere Entscheidungen zu treffen und neue Ideen zu gewinnen, wie bestehende Prozesse und Vorgehensweisen sich weiterentwickeln können.

Sie werden im Verlauf des nachfolgenden Buches feststellen, dass vieles von dem, was wir in XP betrachten, sehr nahe an oder in manchen Fällen gar deckungsgleich mit Elementen von Scrum ist, anderes wiederum das Potential hat, unser bestehendes Scrum zu optimieren oder zu ergänzen.

Aus meiner Praxis bin ich überzeugt, dass es für jeden Scrum-Master und jeden Product Owner absolut essentiell ist, so viel wie möglich über das Denken und die Herangehensweise von Entwicklern zu verstehen, um eine gemeinsame Kommunikationsebene zu schaffen und damit sowohl bessere Produkte als auch erfolgreichere Prozesse zu ermöglichen. Hilfreich wird dabei unter anderem auch der Umstand sein, dass manche Herangehensweisen und Vorstellungen – trotz der Nähe zu Scrum – in XP anders angegangen werden. Das kann zusätzliche Impulse zum besseren Verständnis von Scrum und zu einem zielführenden Einsatz des Frameworks bieten.

Viel Erfolg damit!

¹ Übersetzung:

“Wie werden die traditionellen und extremen Programmiermethoden in der Praxis kombiniert?”

Es ist keine traditionelle Praxis. SCRUM ist ein Framework für Teams, die technische Praktiken anwenden, die in der extremen Programmierung beschrieben sind. Das erste SCRUM-Team implementierte alle extremen Programmierpraktiken, bevor es extreme Programmierung gab. Tatsächlich bat mich Kent Beck, der mit extremer Programmierung begann, ihm alle Informationen zu senden, die ich über SCRUM hatte, und er hatte alles über SCRUM, bevor er anfang, an extremer Programmierung zu arbeiten. Als ich 1995 mit Ken Schwaber zusammenkam, um über die Einführung von SCRUM in der Branche zu sprechen, war Ken der Ansicht, dass wir SCRUM nur als Team-Framework präsentieren sollten: die Produktbestellung, den SCRUM-Master, das Team, die Scrum-Meetings (wie Sie zusammen funktionieren) und die SCRUM-Artefakte (wie Sie ein SCRUM-Projekt verfolgen und verwalten). Und er war der

Meinung, dass SCRUM in zwei Tagen sehr schnell implementiert werden könnte und die Mitarbeiter im Laufe der Zeit die Konstruktionspraktiken verbessern könnten, indem sie die in SCRUM eingebettete kontinuierliche Verbesserung nutzen. Und wenn sie sich bemühten, die Hindernisse zu beseitigen, die ihren Fortschritt blockierten, waren viele dieser Hindernisse technische Probleme, und sie konnten sich auf extreme Programmierpraktiken verlassen, um ihnen dabei zu helfen.“