



Allein auf weiter Flut die Zeit mit Testen verschwenden – gerade Anwender sehen darin oft keinen Sinn. Dabei können sie von der Erfahrung profitieren.

Gesteigerte Effizienz durch fehlertoleranz im Software-Testing

Jan Overbeck*

Geht es um Software-Tests, driften die Gemüter schnell auseinander. Die Angst, Fehler gemacht zu haben auf Seiten der Entwickler, trifft auf Manager, die Fehlerfreiheit erwarten. Konflikte sind vorprogrammiert.

Seit G. Myers 1978 *The Art of Software Testing* geschrieben hat, haben wir viel über Software-Test gelesen und gehört. Unter Autoren zu diesem Thema ist es beliebt, dem Leser zur Einleitung dramatische Auswirkungen eines Software-Fehlers zu schildern (etwa das Gepäck-Chaos in Heathrow), um dann im Hauptteil zu zeigen, wie man mit verbesserten Test-Methoden oder -Werkzeugen diesen Fehler wahrscheinlich gefunden hätte. Alle diese Beiträge zum Stand der Technik im Software-Test sind wichtig und

richtig – es hat sich eine Menge entwickelt seit 1978. Aus der täglichen Praxis des Software-Tests ist jedoch ein Aspekt erfolgskritisch, der vielfach zu wenig beleuchtet wird. Der Faktor Mensch. Wird in einem Projekt oder in einer Organisation Software-Testen eingeführt, so löst das bei Menschen in unterschiedlichen Rollen verschiedene – meist destruktive – Reaktionen aus. Die resultierenden Haltungen gegenüber dem Test lassen sich jedoch in konstruktive Haltungen umwandeln. Das Ergebnis ist eine höhere Effizienz im gesamten Projekt, nicht

nur im Testen. Erst dann hat Software-Test den Beitrag zum Projekterfolg geleistet, den man sich erwarten darf. Je nach Rolle reagieren die einzelnen Parteien unterschiedlich und brauchen differenzierte Lösungsansätze.

Entwickler:
«Testen stellt mich bloß.»

Die Aufgabe von Testen ist es, Fehler zu finden. Andererseits mag es niemand, wenn seine Fehler gefunden und aufgezeigt werden. Obwohl im Testen die Fehler von vielen Beteiligten gefunden werden – was der Fachbereich nicht ausdrücken kann, was der Analytiker falsch verstanden hat, was der Chefarchitekt übersehen hat, was das letzte Projekt falsch in die Schnittstelle eingebaut hat – zeigt sich, dass Entwickler spontan zu der beschriebenen Reaktion neigen. Idealsituation aber ist, wenn niemand fehlerfrei arbeiten muss, da der Betreffende dann weder innovativ noch effizient sein kann. Eine fehlertolerante Arbeitsumgebung erlaubt hohe Produktivität, in der die Rolle des Software-Tests darin besteht, teure Umwege zu vermeiden. Ist diese Sicht erst im gesamten Team verankert, empfinden sich alle Beteiligten im selben Boot – jeder in seiner spezifischen Rolle.

In Projekten ab einer gewissen Komplexität (in der Regel in Projekten mit mehr als zwei Beteiligten) ist es unumgänglich, die fachlichen Know-how-Träger in den Test zu

