

## Ist Testen rentabel?

(©2014 Peter Hartauer)  
info@peterhartauer.de

Mittlerweile ist es hinreichend bekannt, dass Testen wichtig für die Software ist. Wenn wir nun aber gefragt werden, was wir bereit sind für den Test auszugeben, dann werden wir feststellen, dass wir nur unzureichend wissen, was der eigentliche Mehrwert des Testens ist.

Im Folgenden sehen wir uns nun vorliegende Mehrwertberechnungen an:

- Ohne Tests
- Mit manuellen Tests
- Mit automatisierten Tests
- Mit vorausgehender Verifikation und Validierung
- Mit Weitergabe von Informationen an den Support
- Mit Risikoreduktion
- Mit Ermittlung von Informationen zur Projektsteuerung

Hierbei betrachten wir die vier Werte des Testens:

- Es werden Fehler gefunden, die behoben werden: es werden Schäden verhindert.
- Es werden Fehler gefunden, die nicht behoben werden: aber sie sind bekannt.
- Tests führen zu einer Risikoreduktion.
- Tests führen das Projekt zum Erfolg durch präzise, zeitnahe und glaubwürdige Informationen.

Es werden auch die Kosten der Nicht-Qualität beachtet:

Qualität = Konformität + Nicht-Konformität

Diese Kosten umfassen Konformitätsprüfung (Entdeckung von Anomalien) und unterstützende Maßnahmen der Qualitätssicherung (Vermeidung von Anomalien). Die Kosten der Nicht-Konformität belaufen sich auf die Kosten für die Korrektur von Mängeln, Fehlernachtests, Betreuung von unzufriedenen Kunden, Schäden am Image des Unternehmens, usw.

Wir betrachten hierbei das Return on Investment: Investitionen / Kosten

Ziehen wir nun ein fiktives Beispiel heran:

Konformitätskosten:

- Manuelle Tests
- Automatisierte Tests
- Statische Prüfungen

Nicht-Konformitätskosten:

- Fehlerbehebung

Annahmen:

- Alle drei Monate wird eine neue Version erstellt.
- In jeder Version sind 1000 Anomalien enthalten.

Korrekturkosten:

- 1 €, während der Codereviews
- 10 €, wenn der Programmierer testet
- 100 €, wenn das Testteam testet
- 1000 €, wenn der Kunde den Fehler findet

## Die erste Berechnung: ohne Tests

### Annahmen:

- 1000 Anomalien sind im Produkt.
- 250 Anomalien werden durch die Programmierer gefunden.
- 750 Anomalien werden durch die Nutzer entdeckt.

### Mehrwert des Tests:

- KEINER! Aber man sieht Potential.
- Kosten der Qualität über einer 3/4 Millionen Euro
- Unzufriedene Kunden

### Verbesserungsmöglichkeiten:

- Einführung von formalen, manuellen Tests durch ein Testteam

Ohne Tests	
Test	
Personal	0 €
Infrastruktur	0 €
Gesamtinvestition	0 €
Entwicklung	
Anzahl der gefundenen Anomalien	250
Kosten der Fehlerkorrektur	2.500 €
Test	
Anzahl der gefundenen Anomalien	0
Kosten der Fehlerkorrektur	0 €
Nutzer	
Anzahl der gefundenen Anomalien	750
Kosten der Fehlerkorrektur	750.000 €
Kosten für die Qualität	
Konformität	0 €
Nicht-Konformität	752.500 €
Gesamtkosten für die Qualität	752.500 €
Return on Investment	Nicht verfügbar

## Die zweite Berechnung: mit manuellen Tests

### Annahmen:

- 1000 Anomalien sind im Produkt
- 250 Anomalien werden durch die Programmierer gefunden.
- 350 Anomalien werden durch die Tester gefunden.
- 400 Anomalien werden durch die Nutzer entdeckt.

### Mehrwert des Tests:

- Reduktion der Kosten für die Qualität im etwa 33 %
- Return on Investment bei 350 %
- Etwa 40 % weniger Fehler beim Nutzer

### Verbesserungsmöglichkeiten:

- Einführung von automatisierten Tests

Mit manuellen Tests	
Test	
Personal	60.000 €
Infrastruktur	10.000 €
Gesamtinvestition	70.000 €
Entwicklung	
Anzahl der gefundenen Anomalien	250
Kosten der Fehlerkorrektur	2.500 €
Test	
Anzahl der gefundenen Anomalien	350
Kosten der Fehlerkorrektur	35.000 €
Nutzer	
Anzahl der gefundenen Anomalien	400
Kosten der Fehlerkorrektur	400.000 €
Kosten für die Qualität	
Konformität	70.000 €
Nicht-Konformität	437.500 €
Gesamtkosten für die Qualität	507.500 €
Return on Investment	350 %

## Die dritte Berechnung: mit manuellen und automatisierten Tests

### Annahmen:

- 40% der Fehler werden durch den automatisierten Test entdeckt.
- Die Investitionen von 150.000€ für die Automatisierung amortisieren sich über 3 Jahre (12 Releases).

### Mehrwert des Tests:

- Halbierung der Kosten für die Qualität
- Return on Investment bei 445 %
- Etwa 66 % weniger Fehler beim Nutzer

### Verbesserungsmöglichkeiten:

- Präventive Maßnahmen zur Fehlervermeidung einführen.

Mit manuellen und automatisierten Tests	
<b>Test</b>	
Personal	60.000 €
Infrastruktur	10.000 €
Werkzeuge (Abschreibung)	12.500 €
Gesamtinvestition	82.500 €
<b>Entwicklung</b>	
Anzahl der gefundenen Anomalien	250
Kosten der Fehlerkorrektur	2.500 €
<b>Test</b>	
Anzahl der gefundenen Anomalien	500
Kosten der Fehlerkorrektur	50.000 €
<b>Nutzer</b>	
Anzahl der gefundenen Anomalien	250
Kosten der Fehlerkorrektur	250.000 €
<b>Kosten für die Qualität</b>	
Konformität	82.500 €
Nicht-Konformität	302.500 €
Gesamtkosten für die Qualität	385.000 €
Return on Investment	445 %

Die vierte Berechnung: mit manuellen, automatisierten und statischen Tests

Annahmen:

- Tester prüfen die Anforderungen und die Entwürfe.
- Dadurch vermeiden sie den Einbau von 150 Anomalien.
- Diese werden sofort behoben und kosten dadurch 1 €.

Mehrwert des Tests:

- Reduktion der Kosten für Qualität um etwa 66 %
- Return on Investment bei 627 %
- Etwa 90 % weniger Fehler beim Nutzer

Mit manuellen, automatisierten und statischen Tests	
<b>Test</b>	
Personal	60.000 €
Infrastruktur	10.000 €
Werkzeuge (Abschreibung)	12.500 €
Gesamtinvestition	82.500 €
<b>Entwicklung</b>	
Anzahl der gefundenen Anomalien	250
Kosten der Fehlerkorrektur	2.500 €
<b>Test</b>	
Anzahl der gefundenen Anomalien	500
Anzahl der gefundenen Anomalien (statisch)	150
Kosten der Fehlerkorrektur	50.150 €
<b>Nutzer</b>	
Anzahl der gefundenen Anomalien	100
Kosten der Fehlerkorrektur	100.000 €
<b>Kosten für die Qualität</b>	
Konformität	82.500 €
Nicht-Konformität	152.650 €
Gesamtkosten für die Qualität	235.150 €
Return on Investment	627 %

Nicht behobene Fehler?

Warum sollte es so etwas in einer Software geben?

Einer der Gründe kann natürlich eine reine Kostenfrage sein. Wenn die Behebung eines Fehlers mehr kosten würde, als wenn man diesen in der Software belassen würde, dann ist eine Behebung nicht wirtschaftlich. Hierbei ist aber zu beachten, dass eine wirtschaftliche Betrachtung nur dann möglich ist, wenn man auch die potentiellen Fehlerkosten kennt oder zumindest abschätzen kann.

Ein anderer Grund kann sein, dass die Behebung Folgen hat, die man noch nicht abschätzen kann. Seit es durch eine sehr komplexe Software, die ein großes Potential an Seiteneffekten (ripple effect) hat, oder dass man selbst nichts darüber weiß, wie die Software selbst aufgebaut ist.

Der Vorteil, den man aber trotz noch vorhandener Fehler hat, ist, dass man die Anwender durch einen Eintrag in der Dokumentation darauf hinweisen kann, wie man diese Fehler vermeidet und, wenn er schon aufgetreten ist, wie man ihn behebt oder den Schaden gering hält. Oftmals sind diese Hinweise in einem Abschnitt enthalten, den man als „Bekannte Fehler“ („Known Issues“) kennt.

Doch nicht nur der Kunde kann einen Vorteil daraus ziehen. Auch der Support wird durch die Bekanntmachung von Fehlern nicht mit unliebsamen Überraschungen konfrontiert. Zu guter Letzt kann man außerdem versuchen, dass man diese Fehler maskiert, bis eine geeignete Lösung gefunden wird.

Der Mehrwert dieser drei Optionen liegt klar auf der Hand: Der Fehler ist kein unkalkulierbares Risiko mehr und im Projekt kann eine bessere Planung durchgeführt werden.

Doch auch eine Kostenberechnung ist hier möglich! Man kann auch einen Mehrwert bei bekannten, aber nicht behobenen Fehlern berechnen. Diese machen sich in Support bemerkbar. Auch hier müssen wir mit hypothetischen Daten rechnen. In der Praxis empfiehlt es sich Rückmeldungen aus dem Helpdesk einzuholen um konkrete Daten zu erhalten und eine realistische, für Sie zutreffende Berechnung zu erhalten.

Unsere Annahme ist nun, dass ein Anruf beim Helpdesk im Durchschnitt 15 Minuten kürzer ist als bei einem unbekanntem Fehler. Dies liegt daran, dass der Helpdesk nicht aufwändige Befragungen durchführen muss, da sowohl Symptome als auch Unterstützungsvorschläge schon bereit liegen.

Für jede Anomalie treffen im ersten Jahr etwa 5 Anrufe pro Monat ein. Der Stundenlohn eines Servicemitarbeiters im Helpdesk wird mit 40€ pro Stunde veranschlagt. Somit können wir folgende Berechnung aufstellen:

$$0,25h * 5 * 12 * 40€/h = 600€$$

Somit können wir pro bekanntem Fehler im Jahr 600 € sparen.

In dieser Berechnung sind die nicht getätigten Anrufe der Benutzer, welche die Fehler in der Anleitung nachgelesen haben, noch gar nicht mit eingerechnet. Dadurch verbessert sich diese Quote. Doch diese Zahlen sind nicht immer einfach zu ermitteln. Hierbei ist es notwendig, dass Daten aus dem Support gesammelt und regelmäßig ausgewertet werden.

Hierbei sind aber die Kosten, die beim Anwender durch die Verzögerungen im Arbeitsablauf entstehen, noch gar nicht berücksichtigt. Im Nebeneffekt kann eine offene und ehrliche Kommunikation zwischen Kunde und Lieferant, bei der auch noch offene Fehler mitgeteilt werden, dazu führen, dass das Vertrauen der Kunden steigt und auch Phasen, in denen es durch Zeit- oder Budgetdruck normalerweise zu starken Spannungen kommen kann, eine einfachere Lösung der Situation möglich ist.

Mit Weitergabe von Informationen an den Support	
<b>Test</b>	
Personal	60.000 €
Infrastruktur	10.000 €
Werkzeuge (Abschreibung)	12.500 €
Gesamtinvestition	82.500 €
<b>Entwicklung</b>	
Anzahl der gefundenen Anomalien	250
Kosten der Fehlerkorrektur	2.500 €
<b>Test</b>	
Anzahl der gefundenen Anomalien	500
Anzahl der gefundenen Anomalien (statisch)	150
Kosten der Fehlerkorrektur	50.150 €
Nicht behobene Fehler	650
<b>Nutzer</b>	
Anzahl der gefundenen Anomalien	100
Kosten der Fehlerkorrektur	100.000 €
Einsparungen in Support	32.500 €
<b>Kosten für die Qualität</b>	
Konformität	82.500 €
Nicht-Konformität	152.650 €
Gesamtkosten für die Qualität	235.150 €
Return on Investment	666 %

Die Einsparungen entstehen mit einem relativ geringen Aufwand. Es sind keinerlei zusätzliche Investitionen notwendig.

Durch hypothetische 650 bekannte, aber nicht korrigierte Fehler, kann folgende Berechnung aufgestellt werden:

$$650 * 50€ \text{ (Aus der Berechnung } 0,25h * 5 * 40€/h) = 32.500€$$

Um solche Daten zu erheben ist es notwendig, dass das Testprojekt Daten aus Metriken erhält, die im Support gesammelt werden. Nur durch die Weitergabe der Daten ist es



dem Test möglich Prozessverbesserungen nicht nur anzustoßen, sondern auch über deren Erfolg zu berichten.

Eine essentielle Grundlage für die Erhebung solcher Metriken ist natürlich, dass hierbei weder personenbezogene Daten erhoben werden, noch dass der Aufwand diese Daten zu erheben den Nutzen egalisiert. Es ist also von Vorteil, wenn die Metriken durch ein Werkzeug erhoben werden können. Das entlastet die Mitarbeiter, die dadurch wieder sich ihrer eigentlichen Aufgabe widmen können und ermöglicht es zudem eine gewisse Objektivität in die Daten zu bringen.

## Der Test als Versicherung?

Als Tester und als Testmanager sind Sie defacto Risikoexperten und müssen mit existierenden Produktrisiken umgehen können. Sie können also Schadenshöhe und Eintrittswahrscheinlichkeit definieren und somit eine qualitative, wenn nicht quantitative Aussage zu den potentiellen Schadenshöhen treffen.

Außerhalb der beruflichen Praxis haben Sie auch Risiken, mit denen Sie umgehen müssen. Dies können Schäden am Eigentum, Krankheiten oder andere Dinge des alltäglichen Lebens sein. Was ist das am häufigsten eingeleitete Mittel um mit diesen Risiken zurechtzukommen? Richtig, eine Versicherung.

Eine Versicherung übernimmt auch Risiken, die wir nicht selbst tragen wollen. Dafür verlangt diese eine regelmäßige Prämie, welche an das Risiko angepasst ist. So unterscheiden sich KFZ-Versicherungen nach Modell, Fahrer, Fahrpraxis, Region, ja sogar Lackfarben spielen eine Rolle.

Somit kann man erkennen, dass der Test wie eine Versicherung wirkt. Das Risiko, dass wir im Schadensfall viel Geld für einen Fehler ausgeben müssen wird durch den Test behandelt. Er versucht eine bessere Einschätzung des Fehlers zu ermöglichen, ihn zu entdecken und die Möglichkeit der Behebung sicherzustellen. Somit ist der Test als minimierende Maßnahme für das Produktrisiko wiederum wirtschaftlich berechenbar. Der Wert des Tests ist das eingesparte potentielle Schadensausmaß bei Eintritt multipliziert mit der Eintrittswahrscheinlichkeit.

Falls Sie noch immer am Zweifeln sind, ob jetzt der Test wirklich sinnvoll ist eine einfach Frage:

Würden Sie Ihre Krankenversicherung kündigen, nur weil Sie jetzt gesund sind?

Kehren wir nun zu unserem hypothetischen Beispiel zurück...

Wir nehmen an, dass es folgende Qualitätsfaktoren, basierend auf vorangegangenen und ähnlichen Projekten, gibt:

Performanzprobleme: durchschnittliche Kosten: 100.000€ bei 10% Wahrscheinlichkeit

Funktionale Probleme: durchschnittliche Kosten: 5.000€ bei 50% Wahrscheinlichkeit

Sicherheitsprobleme: durchschnittliche Kosten: 250.000€ bei 5% Wahrscheinlichkeit

Andere Probleme: durchschnittliche Kosten: 10.000€ bei 10% Wahrscheinlichkeit

In unserem Beispiel müssten wir diese Kosten also mit folgenden Prämien versichern:

$$100.000€ * 0,1 + 5.000€ * 0,5 + 250.000€ * 0,05 + 10.000€ * 0,1 = 26.000€$$

Doch Vorsicht! Diese Berechnung ist mit Fallstricken durchsetzt. Sie liefert zwar adäquate Werte um sich einen Überblick über die Risiken zu verschaffen, aber sie bezieht sich nur auf den Mittelwert bei einer kleinen Anzahl von Projekten als Basis. Einige Projekte können auch kleinere Risiken selbst übernehmen, während bei anderen Projekten dies nicht möglich ist. Des Weiteren wird das Testteam die Schäden bei nicht entdeckten Fehlern niemals dem Projektteam ersetzen!

Die sechste Berechnung: mit Risikoreduktion

Annahmen:

- Keine zusätzlichen Investitionen

Mehrwert des Tests:

- Äquivalent von 26.000€ „Versicherung“
- Return on Investment bei 698 %
- Tests liefern Informationen für ein einfacheres Produktmanagement

Mit Weitergabe von Informationen an den Support	
<b>Test</b>	
Personal	60.000 €
Infrastruktur	10.000 €
Werkzeuge (Abschreibung)	12.500 €
Gesamtinvestition	82.500 €
<b>Entwicklung</b>	
Anzahl der gefundenen Anomalien	250
Kosten der Fehlerkorrektur	2.500 €
<b>Test</b>	
Anzahl der gefundenen Anomalien	500
Anzahl der gefundenen Anomalien (statisch)	150
Kosten der Fehlerkorrektur	50.150 €
Nicht behobene Fehler	650
<b>Nutzer</b>	
Anzahl der gefundenen Anomalien	100
Kosten der Fehlerkorrektur	100.000 €
Einsparungen in Support	32.500 €
Als „Versicherung“	26.000 €
<b>Kosten für die Qualität</b>	
Konformität	82.500 €
Nicht-Konformität	152.650 €
Gesamtkosten für die Qualität	235.150 €
Return on Investment	698 %

Ein Hauptgrund für das Scheitern von Projekten ist oftmals ein Mangel an Informationen aus Metriken. Diese Metriken helfen den Managern aussagekräftige Daten zu ermitteln, die eine Steuerung des (Test-)Projekts ermöglichen.

Typische Metriken sind:

Fehlermetriken (offen, geschlossen, Status, Behebungszeit, Ursache, Quelle, Ort,...)

Testfallmetriken (bestanden, nicht bestanden, nicht durchgeführt, blockiert,...)

Überdeckungsmetriken (Risiken, Anforderungen, Code,...)

Fortschrittsmetriken (Testfortschritt, Budgetverbrauch,...)

Risikometriken (Behandelt, verringert, offen,...)

Durch eine konsequente Ermittlung von Metriken ist es möglich im Rahmen der Projektüberwachung das Risiko des Scheiterns eines Projekts von 40% auf 20% zu verringern.

Dies erfolgt also nach dem Ausspruch vom Tom deMarco:

„You can't control what you don't measure“

Hier noch eine kleine Warnung:

Damit diese Metriken funktionieren, müssen diese natürlich an Ihr spezielles Projekt angepasst werden. Je nach Projektgröße können die Werte nämlich variieren. So nennt Caspers Jones in seinem Buch „Estimating Software Costs“ Prozentsätze von 2% für das Scheitern bei kleinen Projekten bis zu 85% für sehr große Projekten.

Wenn wir also nun annehmen, dass gute Tests und eine Überwachung durch Metriken gut die Hälfte der zu erwartenden Reduktion des Risikos ausmachen, dann können wir eine 10%ige Wertsteigerung im Projekt annehmen.

Somit werfen wir einen letzten Blick auf unsere Return on Investment-Berechnung.

Hierfür nehmen wir ein mittelständisches Projekt. Des Weiteren kommen die 10% Kosteneinsparungen aus guten Tests und Metriken hinzu. Für die Berechnung erkennen wir außerdem nur die reinen Projektkosten an. Verluste wie Marktverlust, Auftragsverlust, usw. werden nicht mit eingerechnet.

In unserem Beispiel sind die reinen Projektkosten nur die Budgets von Entwicklung und Test. Somit können wir folgende Rechnung aufstellen:

$(\text{Budget Entwicklung} + \text{Budget Test}) * \text{Einsparungen} = \text{Wert der Informationen}$

$(82.500€ + 247.500€) * 0,1 = 33.000€$

## Die letzte Betrachtung: Mit Ermittlung von Informationen zur Projektsteuerung

### Annahmen:

- Die Testergebnisse müssen glaubwürdig, zuverlässig und aktuell sein um eine gute Informationsquelle darzustellen

### Mehrwert des Tests:

- Das Risiko des Scheiterns wird um 10% reduziert (Wertschöpfung: 33.000€)
- Return on Investment bei 738 %

### Verbesserungsmöglichkeiten:

- Überwachung durch den Einsatz von Werkzeugen

Mit Weitergabe von Informationen an den Support	
<b>Test</b>	
Personal	60.000 €
Infrastruktur	10.000 €
Werkzeuge (Abschreibung)	12.500 €
Gesamtinvestition	82.500 €
<b>Entwicklung</b>	
Anzahl der gefundenen Anomalien	250
Kosten der Fehlerkorrektur	2.500 €
<b>Test</b>	
Anzahl der gefundenen Anomalien	500
Anzahl der gefundenen Anomalien (statisch)	150
Kosten der Fehlerkorrektur	50.150 €
Nicht behobene Fehler	650
<b>Nutzer</b>	
Anzahl der gefundenen Anomalien	100
Kosten der Fehlerkorrektur	100.000 €
Einsparungen in Support	32.500 €
Als „Versicherung“	26.000 €
Mehrwert Projektüberwachung	33.000 €
<b>Kosten für die Qualität</b>	
Konformität	82.500 €
Nicht-Konformität	152.650 €
Gesamtkosten für die Qualität	235.150 €
Return on Investment	738 %

## Zusammenfassung:

Durch diese, wenn auch hypothetischen Berechnungen, können wir erkennen, dass Testen einen messbaren Mehrwert liefert. Dieser Mehrwert lässt sich kaufmännisch berechnen und somit hat auch der Test seine wirtschaftliche Daseinsberechtigung. Tests sind in jeder Hinsicht profitabel. Dabei ist es vollkommen egal, ob die Tests automatisiert sind oder nicht. Wichtig ist es Abweichungen frühzeitig zu erkennen. Es ist hierbei egal, ob der Fehler behoben wird oder nicht. Tests liefern Hilfestellungen für den Support und auch für die Projektüberwachung. Tests sind eine Versicherung, die uns dabei hilft Kosten zu sparen.

Tests sind also sinnvoll...

...und gewinnbringend!