

Exploratives Testen mit Struktur – ein Widerspruch?

von Markus Gärtner

Vor einiger Zeit las ich von einer Konferenzpräsentation mit dem Title "Structured Testing vs. Exploratory Testing". Titel und Inhalt dieser Präsentation implizieren, dass Exploratives Testen und Strukturiertes Testen gegensätzliche Ansätze sind. Exploratives Testen als unstrukturiert zu bezeichnen, entspricht allerdings nicht meinem Verständnis von und meiner Erfahrung mit Explorativem Testen.

Als Schwimmtrainer habe ich früh schon eine Lektion gelernt: Wenn ich die Kinder und Jugendlichen koordinative Übungen wie beispielsweise „Rückenbeine mit senkrecht aus dem Wasser gehaltenen, gestreckten Armen“ durchführen lasse, dann beschweren sich diese zunächst einmal. Die Übung sei unmöglich und auf keinen Fall zu schaffen.

Im Rahmen meiner Schwimmtrainerausbildung hatte ich allerdings alle Übungen, die ich meinen Schützlingen auftrag, selbst mehrere Bahnen lang geschwommen. Dadurch wusste ich, wie schwer die jeweiligen Übungen waren, und ob sie für das Alter und die Fähigkeiten der Schwimmer angebracht sind. Außerdem wusste ich, dass diese Übungen in der Tat geschwommen werden können und konnte diese im Zweifelsfall auch vorschwimmen. Seitdem weiß ich, dass „das geht nicht“ oder „das ist unmöglich“ in den meisten Fällen „ich weiß nicht, wie das geht“ oder „ich weiß nicht, wie ich das möglich machen kann“ bedeutet.

Diese Lektion gilt auch für Struktur in Explorativem Testen. Wenn ich in einer meiner Schulungen den Satz höre, dass Exploratives Testen unstrukturiert ist, so weiß ich sofort, dass dem Teilnehmer eigentlich nur die Struktur im Explorativen Testen fehlt. Aber diese Struktur kann ich im Rahmen der Schulung vermitteln. Zu glauben, dass etwas möglich ist, ist schließlich der erste Schritt, um das Unmögliche möglich zu machen.

Zunächst einmal, was bedeutet das Wort "Struktur" oder auch "strukturiert"? Mein Wörterbuch sagt über Struktur

Struktur, die

1 räumlicher Aufbau, Konstruktion, Gerüst

2 innerer Aufbau, Gefüge aus voneinander abhängigen und sich gegenseitig bedingenden Teilen, Anordnung von Einzelteilen eines Ganzen

3 Oberflächenmuster eines textilen Stoffes oder von Papier

Beim Blick auf das Nomen wird der Aufbau oder auch die Anordnung von Einzelteilen eines Ganzen beschrieben. Doch was sind die Einzelteile im Software-Testen? Testen besteht aus Testdesign, Testdurchführung und einem Lernfaktor, der zukünftige Testfallermittlung in Form von Erfahrung beeinflusst. Traditioneller Softwaretest zerteilt diese drei Schritte sehr streng. Das bedeutet, dass Testdesign, Testausführung und Lernen von unterschiedlichen Personen durchgeführt werden können – zumindest was die ersten beiden Teile betrifft. Die größten Probleme entstehen dabei, wenn der Testdesigner und der Testausführer nicht voneinander lernen, weil sie beispielsweise an geographisch unterschiedlichen Orten arbeiten und sich auf Grund des Zeitzonenschieds nur selten abstimmen können.

Cem Kaner definierte in 1996 Exploratives Testen als eine Form des Software Testens, die die persönliche Freiheit und Verantwortung des einzelnen Testers hervorhebt, damit die Qualität ihrer oder seiner Arbeit optimiert werden kann, indem Testdesign, Testdurchführung, Testinterpretation und Test bezogenes Lernen als sich gegenseitig unterstützende Aktivitäten behandelt werden, und gleichzeitig während der Projektlaufzeit fortgeführt werden. Die Betonung liegt hier darauf, dass der einzelne Tester zwar die persönliche Freiheit genießt, bei seiner Arbeitsplanung, aber diese Freiheit auch mit der Verantwortung für seine Arbeit einhergeht. Verantwortung in diesem Zusammenhang bedeutet, dass der Tester auch dafür Sorge trägt, dass seine Arbeitsergebnisse nachgestellt werden können, indem beispielsweise Notizen zu ihren oder seinen Testaktivitäten vorliegen.

Beim Explorativen Testen sind also die drei Einzelteile des Testens per Definition miteinander verbunden. James Bach definiert Exploratives Testen als „simultanes Lernen, Testdesign und Testdurchführung“. [Bach2002] Das bedeutet, dass Exploratives Testen die drei Einzelteile des traditionellen Testens nicht vollständig voneinander trennt. Allerdings gibt es im Explorativen Testen wesentlich mehr Einzelteile als in einem „strukturierten“ Testansatz. Michael Bolton verweist derzeit auf 18 verschiedene Strukturen und Modelle [Bolton2011] – und ich gehe davon aus, dass noch mehr Bestandteile darauf lauern, entdeckt zu werden.

Eine Struktur im Explorativen Testen besteht darin, die Applikation unter verschiedenen Aspekten zu betrachten. Michael Kelly hat hierfür die FCC CUTS VIDS Touring Heuristik [Kelly1996] eingeführt. Die Buchstaben stehen für Features, Complexity, Claims, Configuration, User, Testability, Scenario, Variability, Interoperability, Data und Structure. Diese stehen für einzelne Touren, die ich als Tester durch die Applikation nehmen kann. Sie geben meiner Testtätigkeit damit Struktur – Einzelteile, die sich zu einem Ganzen zusammenfügen.

Der Test Heuristiken Spickzettel von Elisabeth Hendrickson [Hendrickson] bietet darüber hinaus Ideen für unterschiedlichste Testaspekte. Seien es verschiedene Hinweise zu Datentypen, Web-basierten Tests oder allgemeinen Heuristiken. Wenn ich in einer Testsession einmal nicht weiter weiß, dann bietet mir der Spickzettel Anregungen für weitere Ideen.

Eine weitere Struktur für Exploratives Testen bietet die HICCUPPS Eselsbrücke von James Bach und Michael Bolton.[Bolton2005] HICCUPPS steht für History, Image, Comparable Products, User Expectations, Product, Purpose und Standards & Statutes. Dahinter verbergen sich verschiedene Orakel – ein Prinzip oder Mechanismus, anhand dessen wir feststellen können, ob die Software im Sinne eines bestimmten Anwenders funktioniert. Orakel helfen dabei, Inkonsistenzen in einem Produkt festzustellen. Wenn sich beispielsweise von der Produktversion 2003 zur Produktversion 2007 von Microsoft Office die Benutzeroberfläche komplett ändert, dann ist sie inkonsistent zur gewohnten Arbeitsweise der Benutzer, der History in der obigen Liste.

Dass die Änderung in Microsoft Office – hoffentlich – mit voller Absicht vorgenommen wurde, verdeutlicht einen weiteren Aspekt von Orakeln: sie sind Heuristiken und können auch mal daneben liegen. Das bedeutet, dass ich als Tester mit Hilfe von Heuristiken weiterhin mein Gehirn benutzen sollte. Denn ein automatisierter Test kann nicht selbständig entscheiden, ob eine Änderung gewollt oder ungewollt ist. Dafür brauche ich immer noch menschlichen Gehirnschmalz – wie leider viel zu viele Teams immer wieder mit Erschrecken feststellen.

it-agile GmbH

Paul-Stritter-Weg 5
22297 Hamburg

Fon:+49 (0)40 88 173 - 300
Fax:+49 (0)40 88 173 - 333
E-Mail: info@it-agile.de
web: www.it-agile.de

Bankverbindung:
Hamburger Sparkasse
BLZ 200 505 50
Konto 1235/121355

Geschäftsführer:
Henning Wolf, Jens Coldewey
HRB:92261 Hamburg
Ust.-Id.: DE239483021

Die Heuristiken, Orakel und Touring Modelle, die Bolton bereitstellt, sind verschiedene Strukturen, die wir beim Explorativen Testen einsetzen. Zusammen mit der Struktur von Testnotizen und Debriefings nach einzelnen Testsessions behaupte ich, besitzt Exploratives Testen wesentlich mehr Struktur als so genanntes „strukturiertes Testen“.

Aber was ist mit wiederkehrenden Tätigkeiten wie beispielsweise Regressionstests? Testautomatisierung kann Exploratives Testen genau so unterstützen, wie Exploratives Testen auch Testautomatisierung über neue Tests informieren kann. Die beiden Ansätze bilden gerade in der Agilen Softwareentwicklung eine Symbiose, da sie sich sehr stark gegenseitig unterstützen. Beispielsweise kann ich mit Hilfe von Testautomatisierung damit beginnen, um die getesteten Werte herumzutesten. Wenn ich beispielsweise einen automatisierten Test für die Werte 5 und 10 habe, kann ich schnell einen Test mit den Werten 1, 11, 42 und MAXINT durchführen.

Michael Bolton und James Bach beschreiben Exploratives Testen als einen Testansatz, der sich mit anderen Techniken verbinden lässt. Dabei ist jede Form von Testen explorativ zu einem gewissen Grad, und gleichzeitig vorbestimmt oder gescrripted. Sich wiederholende Testtätigkeiten überlässt man dabei besser der Automatisierung und setzt den Faktor Mensch gezielt dort ein, wo er einen wesentlichen Beitrag liefern kann: beim Lernen und beim kreativen Designen von Testfällen auf Basis von bereits ausgeführten Testfällen. Denn genau dafür ist das menschliche Gehirn geschaffen.

Referenzen

[Bach2002] James Bach, *Exploratory Testing Explained*. 2002, <http://www.satisfice.com/articles/et-article.pdf>

[Bolton2011] Michael Bolton, *Resources for Exploratory Testing*, 2011, <http://www.developsense.com/resources.html#exploratory>

[Kelly1996] Michael Kelly, *FCC CUTS VIDS Touring Heuristics*, 1996, <http://www.michaeldkelly.com/archives/50>

[Bolton2005] Michael Bolton, *HICCUPPS Consistency Heuristics*, 2005 <http://www.developsense.com/articles/2005-01-TestingWithoutAMap.pdf>

[Hendrickson] Elisabeth Hendrickson, *Test Heuristics Cheat Sheet*, <http://testobsessed.com/wp-content/uploads/2011/04/testheuristicscheatsheetv1.pdf>, Deutsche Version unter http://www.it-agile.de/fileadmin/docs/Heuristic-Sheet_deutsch.pdf