



□ Rudolf Groetz

(rudolf.groetz@gmail.com)

beschäftigt sich seit 1989 mit IT, die damals noch EDV hieß. Aufgewachsen ist er in einer Welt mit Wasserfallprojekten, dem V-Modell und einer „Jetzt werfen wir mal die Software über die Mauer zu den Testern!“-Mentalität. Seit 2012 leitet der ISTQB Certified Full Advanced-Tester die QA-Abteilung bei JUMIO Inc. und versucht als Head of QA die richtige Dosierung zu finden, um Requirements Engineering und Softwaretests „agile“ erfolgreich zu machen.



□ Dr. Frank Simon

(frank.simon@bluecarat.de)

studierte Informatik und hat im Bereich der Qualitätssicherung großer Systeme promoviert. Nach über 10 Jahren im Beratungsgeschäft im Bereich Qualitätsmanagement und Testen leitet er heute bei der BLUECARAT das Business Development. Gleichzeitig ist er im Vorstand des German Testing Boards, Vorsitzender des BITKOM-Lenkungsausschuss Software und Leiter des Arbeitskreises Software Engineering.

Standardisiertes Generalwissen: agil, mobil und ISTQB-zertifiziert

Weltweit gibt es heute über 330.000 vom International Software Testing Qualification Board (ISTQB) zertifizierte Tester, davon alleine knapp 35.000 in Deutschland. Doch inwieweit passt das standardisierte Generalwissen in die heutige schnelllebige Zeit? Wie passt es in eine zunehmend fragmentierte Weiterbildungslandschaft, die insbesondere für das Requirements Engineering viele scheinbar alternative Trainingskurse aufzeigt? Im folgenden Gespräch zwischen Herrn Rudolf Groetz (RG), Head of Quality Assurance, Jumio Inc. – also der Anwenderseite – und Herrn Dr. Frank Simon (FS), stellvertretender Vorsitzender des German Testing Boards (GTB), wird dieser Frage nachgegangen.

FS: Das GTB/ISTQB hat ein sehr ganzheitliches Verständnis vom Testen. Das lässt sich einfach an drei Dimensionen festmachen:

- **Wann:** Getestet wird demnach während des gesamten Lebenszyklus eines IT-Systems, d. h. angefangen bei der Analyse, dem Requirements Engineering und der Konzeption bis zum Betrieb und dem Abschalten.
- **Was:** Getestet werden nicht nur die fertige Applikation, sondern ebenfalls alle relevanten Testobjekte wie Dokumentationen und der Quelltext.
- **Wogegen:** Getestet wird demnach nicht nur die Funktionalität, sondern ebenfalls alle nicht-funktionalen Anforderungen, wie Sicherheit, Performance und Wartbarkeit.

Dieses extrem breite Betätigungsfeld stellt enorme Anforderungen an den einzelnen Tester in der konkreten Projektsituation. Und es stellt natürlich extreme Anforderungen an Trainings, da dort der Spagat zwischen generalistischem Wissen, das

grundsätzlich relevant ist, und spezifischem Wissen, das nur für ausgewählte Projektsituationen bedeutsam ist, zu meistern ist. Meine erste Frage daher: „Wie breit ist Ihr Verständnis vom Testen in Ihrer Unternehmung aufgestellt? Fangen wir beim *Wann* an: Beginnt das Tätigkeitsfeld des Testers bei Ihnen auch in der Requirementsphase?“
 RG: Als Antwort ein ganz klares: Ja! Wir wissen alle, dass die kostengünstigsten Fehler die sind, die man nicht macht und somit auch nicht beheben muss. Früher nannte man es konstruktive Qualitätssicherung, heute nennt man es Shift-Left. Ich kann mich hier nur den agilen Gurus der Szene (Adcz, Crispin und Co) anschließen, die für spezielle Requirements Workshops plädieren, in denen alle Schlüsselpersonen (Stakeholder) beteiligt sind. Und die Tester sind nun mal Stakeholder. Aus diesem Grund ist ja ATDD (Acceptance Test Driven Development) entstanden und zeigt, wie es geht. Die einfachste, aber auch beste Form von Requirements sind Akzeptanzkriterien.

Noch einmal, ja, bei uns sind die Tester bereits in der Requirements Engineering

(RE)-Phase involviert. Aus diesem Grund lege ich viel Wert auf eine solide RE-Ausbildung. Jeder Test-Engineer im Jumio QA-Team ist mindestens Certified Professional Requirements Engineer und ISTQB Certified Tester auf Foundation Level. Nachdem Model Based Testing (MBT) bei uns Einzug hält, wird auch der RE Advanced Level – Requirements Modeling – spruchreif.

FS: Damit kommen auf den Tester aber weitere Skill-Anforderungen zu. Es genügt eben nicht mehr, sich durch Anwendungen zu klicken und zu versuchen, etwas zum Absturz zu bringen. Der Tester muss also nicht mehr nur Äquivalenzklassen, Testdaten und Testreports kennen, sondern auch ganz früh im Lebenszyklus mithelfen. Muß ein Tester damit auch vollständig das Requirements Engineering abdecken?

RG: In ihren Büchern [Adz09] [Cri11], die meiner Meinung nach in keiner Tester-Bibliothek fehlen sollten, plädieren Adcz und Crispin für mehr direkte Kommunikation beim RE. Crispin nennt es „the power of three“, Adcz nennt es die „3-Amigos“. Egal, wie auch der Name ist, es

geht darum, dass die Leute an einem Tisch sitzen und miteinander reden. Der Tisch kann auch virtuell sein (Skype, Google Hangout). Wichtig ist, dass die verschiedenen Rollen nicht im kleinen Kämmerchen vor sich hinbrüten, sondern gemeinsam an der Spezifikation arbeiten und sich durch Fragen gegenseitig befruchten.

Es gibt nun mal verschiedene Blickwinkel auf ein Requirement und die sollten von allen erörtert werden. Und den weitesten Winkel haben meiner Meinung nach die Tester. Die Tester haben das meiste Wissen über ein System. Warum also soll ich dieses Wissen nicht schon früh – Shift-Left – anzapfen? Mein Herz hüpfert jedes Mal vor Freude, wenn sich Product Owner bei den Testern erkundigen, wie denn das eine oder das andere Feature jetzt wirklich funktioniert.

In den meisten Fällen ist es nicht möglich, einen eigenen Requirements Engineer im Team unterzubringen. Diese Fälle schreien danach, dass ein Tester diese Rolle übernimmt. Es gibt Synergien, die genutzt werden können. Requirements Engineers und Tester verwenden zum Teil die gleichen Analysetechniken – Entscheidungstabellen, Statediagrams, Use-Case-Modelle.

FS: Da dieses Testen, so wie Sie es beschreiben, sehr tief in die Branche und die Technik eintaucht, stellt sich natürlich die Frage, ob das ISTQB-Schema, das „nur“ die drei Ausbaustufen Foundation, Advanced und Expert besitzt, diesen Anforderungen gerecht werden kann. Alle Kern-Trainingsmodule sind explizit unabhängig von einer Branche und einer Technologie. Wir glauben aus gutem Grund, weil wir eben der festen Überzeugung sind, dass die Grundlagen guten Testens universelle Gültigkeit besitzen. Wie bewährt sich dieses Denken in der Praxis?

RG: Der Besuch eines dieser Trainingsmodule schafft einen Mehrwert für die Person, die es besucht. Und noch mehr wird es Wert, wenn die Person die Prüfung ablegt und „sich“ zertifizieren lässt. Leider gibt es aber auch andere Meinungen. Aber mal der Reihe nach.

Die Zertifizierung ist ein Beweis dafür, dass strukturiertes Wissen und Können vorhanden ist. Damit sind Grundlagen gelegt, um sich auch weiteres Fachwissen selbst anzueignen. Tester sind häufig Quereinsteiger in das Thema. Ein Bewerber, der ISTQB Foundation Level zertifiziert ist, belegt, dass er das kleine 1x1 des Testens beherrscht. Ich weiß, dass wir die

gleiche Sprache sprechen. Ich weiß, dass wir unter Äquivalenzklassen und risikobasiertem Testen das gleiche verstehen.

Der ISTQB Advanced Level ist das große 1x1. Ich weiß, dass der Bewerber Äquivalenzklassen zum Ableiten von Tests anwenden kann. Ich weiß, dass er einen Systemtest risikobasiert planen kann, u. v. m. Es gibt das schöne Sprichwort: „Ein Bild sagt mehr als tausend Worte!“. Für mich ist das ISTQB-Zertifikat das Bild in einem Bewerbungsschreiben.

Jetzt werden die Zweifler aber aufschreien und sagen: „Ja, das Zertifikat ist schön und gut, aber das sagt doch nichts über die Praxis aus!“ Dem halte ich aber die vom ISTQB geforderte Voraussetzung entgegen, die erfüllt sein muss, um in den Advanced Level zu gelangen: Nämlich den Nachweis von mindestens 24 Monaten Praxis im Softwaretest.

FS: Damit tragen Sie natürlich bei mir „Eulen nach Athen“. Aber Sie kennen sicher auch gegenteilige Meinungen, gerade aus der agilen Denke, wonach Testen einer der breit verfügbaren Skills der sogenannten „T-Shaped-Skills“ sein sollte. Das Breitenwissen (horizontaler T-Strich) und das Expertenwissen (vertikaler Strich) belegen hier die Fähigkeiten einer Person. Vielleicht bedarf es damit dedizierter Tester, vielleicht sogar mit einem ISTQB-Zertifikat, gar nicht mehr?

RG: Ja, ich kenne diese Meinung. Und ich kann es nicht mehr hören: „Wir testen agil, also brauche ich kein ISTQB! Es reicht Test-DNA“. Das eine sind Äpfel, das andere Birnen. Übertragen wir diese DNA-Denkweise mal auf den American Football Sport, den Teamsport par Excellence.

Nehmen wir einmal an, wir haben zwei Running Backs (Angreifer für Laufspielzüge). Nehmen wir weiter an, beide sind 25 Jahre alt, 1,75m groß, 90kg schwer. Beide sind gleich schnell, beide sind extrem wendig, also beide haben die perfekte Football-DNA. Einer davon hat die klassische College-Football-Ausbildung – Freshman, Sophomores, Junior, Senior – und Football von der Pike auf gelernt (Technik, Taktik, Spielzüge, Teampsychologie,... – er ist sozusagen ISTQB Advanced Level Football-Spieler). Nehmen wir weiter an, der andere ist Quereinsteiger, also Footballautodidakt, Freshman-Level.

Jetzt die Gretchenfrage: Sie sind Headcoach des Teams und müssen sich für den alles entscheidenden Spielzug für einen der beiden entscheiden. Welchen würde Sie

einsetzen? Also? Certified Football-Spieler oder Football-Autodidakt? Ich weiß zumindest, wen ich wählen würde.

Bezogen auf Agilität und Testen: Mir gefällt die Definition von Brandes [SQM12]: „Agiles Testen bedeutet, vorhandene und bewährte Testtechniken so im agilen Entwicklungsprozess zu verankern, dass sie die Ziele des agilen Vorgehens unterstützen, nämlich:

- schnelles Feedback,
- hoher Automatisierungsgrad,
- geringer Management-Overhead,
- enge Zusammenarbeit im Team“.

Was spricht da gegen ISTQB-Module? Bewährte Testtechniken klingt doch sehr verdächtig nach ISTQB, oder? Und auch Testautomatisierung wird durch ISTQB abgedeckt. ISTQB passt also hervorragend auch in agile oder mobile Teams. Trotzdem ist es schon richtig: Das Generalwissen des ISTQB reicht heute in vielen Fällen nicht mehr. Es gibt sowohl branchenspezifische als auch technikspezifische Parameter, die heute zusätzlich von Nöten sind, um professionell zu testen.

FS: Ein guter Punkt. Der ISTQB hat für diesen Zweck Erweiterungsmodule, sogenannte Extensions, vorgesehen, die das horizontale Generalwissen um technik- oder branchenspezifische vertikale Spezialkurse erweitern. So wird es noch in diesem Jahr eine Extension zum Foundation Level namens „Agile Testing“ geben. Der Syllabus ist bereits in Englisch verfügbar. Es geht dort nicht um andere oder neue Testtechniken, sondern darum, wie die etablierten Methoden effizient in die besondere Technik der agilen Projektdurchführung eingeführt werden können.

Der Kerninhalt der Agile Testing Extension liegt daher darin, zu erläutern, wie das bestehende ISTQB-Know-how in agilen Projekten zum Einsatz kommen kann. Von daher an dieser Stelle die letzte Frage: Welche weiteren Ideen hätten Sie für solche Extensions in Zukunft? Wo sehen Sie die Möglichkeit, das generalistische ISTQB-Wissen weiter in vertikalen Durchstichen anzuwenden?

RG: Diese „Extensions“ hören sich sexy an. Ich kann mich mit dem CAT (Certified Agile Tester) nicht so richtig anfreunden, weil er sehr Scrum-lastig ist. Ich würde mir auch „Extensions“ für einige neue technische Bereiche wünschen, vor allem für den Bereich mobile Applikationen. In diesem

Bereich gibt es durchaus kontextspezifische nicht-funktionale Anforderungen, die adressiert werden sollten. Welche Testansätze gibt es im physischen Kontext? Welche im Kommunikationskontext? Wie gehe ich vor, um für verschiedene Mobilfunknetze oder verschiedene Signalstärken Tests zu designen?

Aber auch beim Testen von mobilen Applikationen ist das generalistische ISTQB-Wissen von Vorteil – Stichwort: Ableiten einer Devicematrix (Hardware, Betriebssystem, Bildschirmgröße...) nach der Pairwise-Methode. Da wäre durchaus Platz für einen Extensions-Syllabus. Auch eine „Performance-Extension“ hätte eine Daseinsgrundlage: Wie definiere ich technisch Lastmodelle, wo setze ich die richtigen Messklammern und Messpunkte? Wie setze ich einen Performance-Test technisch

auf? Wo positioniere ich meine Lastgeneratoren? Solches Wissen bekomme ich aktuell nur in teuren Onsite-Seminaren. Und natürlich der Klassiker: Security. Was sind heute technisch machbare Tests? Wie z. B. definiere ich einen Angriffsvektor? Welche Tools sind heute verfügbar?

Ich denke, dass es großen Bedarf für solche Extensions gibt. Das entspricht vielleicht nicht 100 % Ihrem Anspruch auf generalistischem Wissen, aber die Praxis wird heute immer anspruchsvoller. Gerade am Anfang beim Requirements Engineering ist es wichtig, zusätzlich einiges technisches und branchenspezifisches Know-how zu haben. Die Extensions scheinen mir da eine gute Möglichkeit. Aber beeilen Sie sich, die Mitbewerber schlafen nicht! Wenn es funktioniert ... die Tester-Community wird es Ihnen danken. ■

Referenzen

[Adz09] Gojko Adzic, Bridging the Communication Gap, Neuri Limited, 2009, ISBN: 978-0955683619.

[Cri11] Lisa Crispin, Janet Gregory, Agile Testing: A Practical Guide for Testers and Agile Teams, Addison-Wesley, 2011, ISBN: 978-0321534460.

[SQM12] SQ-Magazin, Ausgabe 23/Juni 2012.

[Sim10] Frank Simon, Daniel Simon, Qualitätsrisikomanagement: Ganzheitliche Projektsteuerung, Logos-Verlag, Berlin, 2010.