



[Editorial]



Liebe Testcommunity,
Die Sommerzeit ist vorbei, und die Uhren wieder umgestellt, ich hoffe, auch alle Softwareanwendungen haben die Umstellung gut verkraftet.

Die Abende bieten sich jetzt immer mehr zum Lesen interessanter Lektüre an, und wir hoffen, mit dieser Ausgabe wieder dazu beitragen zu können.

Wir haben wieder Fachartikel zu einer bunten Palette an Themen zusammengestellt (Seiten 6 - 10, und ab 20), weiters Rückschau und Vorschau zu diversen Events (Seiten 11 - 15).

Besonders begrüßen wir einen neuen Goldpartner des ATB - QACube, eine Vorstellung finden Sie auf Seite 5.

Nehmen Sie auch bis 15.12.2015 an der Umfrage zu Entwicklungen und Trends bei Test und Qualitätssicherung teil, näheres auf Seite 10.

Wie immer gibt es die Informationen zu Seminarterminen und Prüfungsterminen unserer Partner, sowie zu Konferenzen zum Thema Softwaretest und Qualitätssicherung.

Last, but not least möchte ich auf den alljährlichen ATB Charity Punsch aufmerksam machen - neben einem geselligen Beisammensein und Vernetzungsmöglichkeit kommen die Spenden einem guten Zweck zugute - kommt zahlreich und unterhaltet Euch gut!

Viel Spaß beim Lesen und Gut Test!

◀(Karl Kemminger)

**[Ankündiger zum ATB Charity Punsch:
Motto „Vernetzen. Spaß haben und HELFEN“]**



Wann?



Vorraussichtlich
Donnerstag 10.Dez 2015
ab 18:00-21:00 Uhr

Wo?

Weihnachtsmarkt Altes AKH,
beim Standl des Unibräu
1090 Wien
Alser Straße 4/ Hof1



Angebot für Firmen!

Spenden und Werben!
Spender mit Logo werden auf all unseren Werbeplattformen wie dem Newsletter, der Website und bei Konferenzauftritten genannt.

Wie funktioniert?

Ab einer Spende von EUR 100,- sind Sie dabei !

◀(Helmut Pichler)

Bereits fix dabei (alfab.)



[Inhalt]

• Goldpartner	Seite 2 - 4
• Prüfungstermine	Seite 4
• Vorstellung neuer Goldpartner	Seite 5
• Erfahrung im Test	Seite 6 - 7
• Testen eines Services via API	Seite 8 - 10
• Umfrage Softwaretest	Seite 10
• Neues aus dem ASQF e.V.	Seite 11 - 12
• ATB Expertentreff	Seite 13
• Rückschau Accelerate	Seite 14
• DACH Meeting	Seite 15
• Trainingsprovider Seminare	Seite 16 - 17
• Impressum	Seite 17
• Partner Veranstaltungen	Seite 18
• Konferenzen, Silberpartner	Seite 19
• Visual Studio Code	Seite 20 - 24



PODIUMSDISKUSSION

SQS Quality Brunch

24. November 2015 ab 8.30 Uhr, Hotel Regina, Wien



sqs.com

Herausforderung Digitalisierung

Neue Ansprüche an die Softwarequalität

ANMELDUNG

MEHR INFO

Am Podium sitzen:

- Univ. Dozent Harry M. Sneed
- Dipl.-Ing. Andreas Nehfort
- Mag. Ewald Pichler
- Mag. (FH) Johannes Kreiner

Moderation: Dr. Mohsen Ekssir

Die Teilnahme ist kostenfrei.

Sichern Sie sich jetzt Ihren Platz! Weitere Informationen zur Veranstaltung und die Möglichkeit zur Anmeldung finden Sie [hier](#).

Bezahlte Einschaltung



ANECON

Qualität hautnah

Erfolgsmodelle flexibler Test Service Center

Produktivität und Qualität durch Test Service Center in Unternehmen - wir diskutieren über die Herausforderungen und Erfolge:

- > Erfahrungsbericht Hutchison Drei
- > Hochkarätige Podiumsdiskussion
- > Informativer Fachvortrag

22. ANECON Expertenfrühstück

Termin: 11.11.2015, 08.30 Uhr

Ort: Albert Hall, 1080 Wien

Programm: www.anecon.com/veranstaltungen

> Jetzt kostenlos anmelden!

Bezahlte Einschaltung



KOMMENDE SEMINARTERMINE

ISTQB® Certified Tester Foundation Level - Extension: Agile Tester

24. - 25.11.2015 - [Linz](#), [Wien](#), [München](#)

Model Based Testing Hands On

01. - 02.12.2015 - [Linz](#), [Wien](#), [Graz](#)

GUI-Testautomatisierung in Theorie und Praxis

02. - 03.12.2015 - [Linz](#), [Wien](#), [Graz](#)

Professionelles Requirements Engineering und Management

09. - 11.12.2015 - [Linz](#), [Wien](#), [Graz](#)

Scrum Master Professional

09. - 10.12.2015 - [Linz](#), [Wien](#), [Graz](#)

ISAQB® Certified Professional for Software Architecture - Foundation Level (CPSA-FL)

14. - 17.12.2015 - [Linz](#), [Wien](#), [Graz](#), [Lustenau](#), [München](#)

Buchen Sie Seminare frühzeitig und nehmen Sie den Frühbucherrabatt von bis zu 10% in Anspruch!

Weitere Informationen zu unseren Seminaren finden Sie auf unserer [Homepage](#) oder in unserem [Seminarprogramm](#)



If you can't see IT, you can't fix IT!

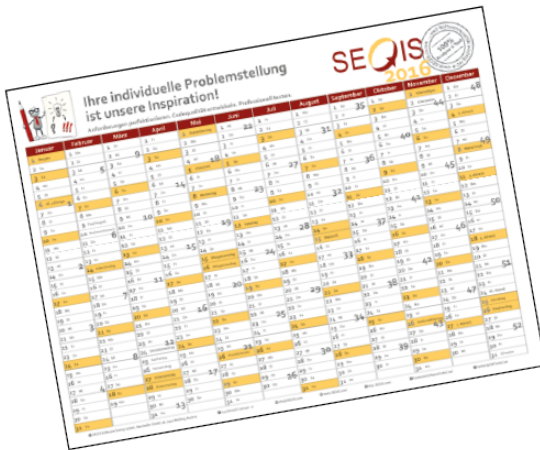
Get the Cube!

QACube drives competitive advantage with predictive **visual analytics & dashboards** across projects in popular tools such as Tricentis Tosca, HP ALM, JIRA, Rally, Excel ...

- Automates Data Consolidation & Aggregation
- Provides Visibility to all Stakeholders
- Enables Transparency through the Right KPIs

www.qacube.com

Sie planen bereits das Jahr 2016? Ihnen fehlt jedoch das passende Tool?



Wir schenken Ihnen einen Jahresplaner
für Ihr erfolgreiches Jahr 2016!

- ✓ Perfekter Überblick für das gesamte Team
- ✓ Alle Termine gesammelt an einem Ort
- ✓ Für Ihre Zeit-, Urlaubs- und Projektplanung
- ✓ Übersichtlich und kompakt (100 x 70 cm)

Planen Sie Ihr Jahr mit dem SEQIS Wandkalender!
Bestellen Sie Ihr kostenloses Exemplar per Mail an
marketing@SEQIS.com – senden Sie uns einfach
Ihre Kontaktdaten.



www.SEQIS.com

aktuell. innovativ. praxisbezogen.

[Facebook/SoftwareTestIsCool](https://www.facebook.com/SoftwareTestIsCool/)

Bezahlte Einschaltung

[Öffentliche ISTQB® Prüfungstermine, Certible]

Öffentliche Prüfungen finden zu folgenden Terminen statt:

16.11.2015 13:00 Klagenfurt
19.11.2015 09:00 Wien
25.11.2015 15:00 Innsbruck
01.12.2015 16:00 Linz
09.12.2015 10:00 Wien
14.12.2015 13:00 Graz
15.12.2015 15:00 Salzburg
21.12.2015 10:00 Wien
31.12.2015 16:00 Wien

Anmeldung und weitere Termine unter
[https://www.certible.com/de/
Kalender/](https://www.certible.com/de/Kalender/)

◀ (Maria-Therese Teichmann)

{Certible}
international certification body

[Öffentliche ISTQB® Prüfungstermine, iSQI]

Öffentliche Prüfungen in Wien finden
zu folgenden Terminen statt:

20.11.2015
29.01.2016
19.02.2016
30.03.2016
29.04.2016

Anmeldung und weitere Termine
finden Sie auf
[https://www.isqi.org/de/
kalenderuebersicht.html](https://www.isqi.org/de/kalenderuebersicht.html)



[Redakteurinnen und Redakteure gesucht]

Haben Sie einen außergewöhnlichen
Bug gefunden? Kennen Sie ein Tool,
von dem die Testercommunity unbe-
dingt wissen sollte? Haben Sie ein
Buch gelesen, das andere auch lesen
sollten?

Ja? Dann schreiben Sie an den ATB-
Insider. Wir suchen noch Redakteu-
rinnen und Redakteure, die Spaß am
Schreiben haben.

Einsendungen an:
newsletter@austriantestingboard.at

◀ (Karl Kemminger)

[Why Have Few Achieved High Levels of QA Maturity?]

von Britta Steele



The Head of QA of one of the largest Swiss Banks once said “without a past there is no future” meaning that you can’t take the leap in maturity to get from good to great maturity without first building the baseline and then moving up the maturity curve. Still, having the right information at your fingertips can help enterprises to avoid false starts or making the wrong decisions.

Sometimes, it is necessary to go off track from the initial plan as unforeseen changes such as legislative, technology or business changes take place. If resources are tight, seeing the quickest way to get back on track or where to allocate the critical resources can make the difference in attaining the software delivery KPIs.

Organizations that have the flexibility to stick close to their software delivery plans while making the optimized use of their resources are most likely to succeed.

Best of Breed in technology can result in Worst of Breed in decision-making as so many different software tools, each with their own proprietary reporting tools result in a type of “tunnel vision” where important key indicators for decision-making across projects are easily missed.

Software delivery resources can be optimized by 58% with better utilization resulting from Cross Border Reporting and Visualization.

QA Visual Analytics

QACube helps drive competitive advantage with predictive QA visual analytics and dashboards across popular ALM/software testing tools such as: Tosca Testsuite, HP ALM, JIRA, Rally, Excel... so you can deliver great software on time and with less risk.

Agile software delivery teams and managers can easily visualize their results across technologies, systems, processes for status updates, potential bottlenecks and early warning signs.

Banks, financial services, insurance companies, telcos and others can achieve visibility and transparency for faster time-to-market with lower risk.

At every level, QACube provides the ability to drill up, or drill down, see the root source of the problem, identify the critical defects vs. progress made, and prioritize testing based on the risks, with full visibility and transparency.

◀ (Britta Steele)



[Die Autorin]

Britta K. Steele

Chief Marketing Officer, QACube

Britta Steele joined QACube in September, 2014 as the Chief Marketing Officer. Britta brings over 15 years of executive marketing, partnerships and management experience to her role at QACube for ALM visual analytics and dashboards. QACube delivers visibility and transparency across the Application Lifecycle including software testing. At the same time, she is accountable for marketing strategy for Sixsentic AG, a Zurich-based Managed Services Software Testing company.

www.qacube.com



[Erfahrung im Test: Ja lernen wir denn nie?]

von Martin Klonk

Erfahrung im Test heißt nicht immer, dass wir wirklich als Organisation daraus lernen. Oft endet es damit, dass man kollektiv die Fehler immer wieder macht. Bleiben wir tatsächlich blind? Das Test Service Center könnte ein Modell sein, um als Organisation im Test wieder lernfähig zu werden.



Abb. 1 Trotz Erfahrung blind. Organisationen lernen nicht von allein.

Warum Erfahrung im Test nicht Lernen ist und was Tucholsky damit zu tun hat.

„Erfahrung heißt gar nichts, man kann seine Sache auch 35 Jahre schlecht machen“.

So treffend kritisierte Kurt Tucholsky die Weltanschauung seiner Zeitgenossen zwischen den beiden Weltkriegen. Ich denke, die Menschen haben dann doch etwas gelernt aus der Erfahrung der damaligen Zeit, aber welchen Preis haben sie dafür bezahlen müssen! Tucholskys Zitat spricht hier eine bittere Erkenntnis aus, die auch auf meinen simplen Alltag in der IT zutrifft.

Lernen wir aus den Erkenntnissen, die wir Jahr für Jahr in den Projekten gewinnen? Wir haben ein Ergebnis erzielt. Aber war es auch gut? Hat es die Erwartungen erfüllt? Haben wir effizient gewirtschaftet?

Vielleicht liegt es daran, dass ich als Testspezialist offenbar eher in jene Situationen geholt werde, in denen es brenzlich ist, denn es braucht viel guten Willen, auch die letzten drei Fragen mit einem Ja zu beantworten. Irgendwie beschleicht mich der Eindruck, als müsse ich mit jedem Projektteam aufs Neue die Vorgehensweise erfinden. Gespräche mit Anderen in der Tester-Community zeigen mir, dass ich mit diesem Eindruck bei weitem nicht der Einzige bin.

Natürlich macht jeder von uns seine eigenen Erfahrungen und optimiert seine Fähigkeiten, lernt etwas dazu. Aber die Gruppe als solches erscheint da resistent.

Immer wieder wird der Test in der Praxis ignoriert, trotz beherzter Lippenbekenntnisse. Immer wieder stürzen seine Erkenntnisse ganze Projekte überraschend in eine gestresste Endphase. Erfahrung heißt also noch nichts, so lange die Gruppe an IT-Experten eines Unternehmens nicht gewillt ist zu lernen.

Ich bin überzeugt, dass die meisten Menschen, mit denen ich im Laufe der Zeit zusammen gearbeitet habe, ihre Sache gut machen wollen und an einem Erfolg des gemeinsamen Produkts interessiert sind. Dafür sind sie gerne bereit, auch einmal unschöne Erlebnisse (nervöser Streit mit Kollegen, Überstunden, Stressmomente, etc.) in Kauf zu nehmen. Wenn ich mit wichtigen Ansätzen für einen erfolgreichen Test an ein Projektteam heran trete, habe ich seltener das Problem des Widerspruchs. Eher habe ich das Problem, dass mich beispielsweise Entwickler thematisch auf der rechten Spur überholen und noch viel Ehrgeizigeres anstreben. Aber genau diese Leute vernachlässigen dann in ihrer täglichen Arbeit die ersten Schritte zur Verbesserung, um sich anschließend mit Verweis auf die Umstände dafür zu entschuldigen. „Es geht halt in der Praxis nicht so, wie wir uns das vorstellen.“ Das ist für alle Beteiligten beschämend und letztendlich auch frustrierend. Was tun? Soll denn jeder Mitarbeiter im Projekt morgens auf sich allein gestellt entscheiden müssen, was jetzt von seinen Aufgaben Priorität hat? Wie er sie jeweils am besten bewältigt? Wenn ja, dann landet die Qualitätssicherung sicher immer ganz hinten – mit Ausnahme bei explizit als solchen auserkorenen Testern.

Test Center versus Alleingang im Projekt

Vor langer Zeit kam daher die Idee auf, dem Test eine eigene Instanz zu geben: Das Test Center oder ähnlich genannte zentrale Organisationseinheiten, die hierarchisch direkt der Abteilungsleitung berichteten. Eine super Idee: Je größer man diese Stabstellen machte, desto höher wurde die Qualität, je kleiner, desto geringer wurde sie.

Perfekt. Man hat so die innerbetriebliche Genehmigungsbehörde geschaffen, die viele der wirklichen Qualitätsprobleme allenfalls am Symptom bekämpft hat. Klar, dass der Streit

zwischen Projektleitung und Test Center schnell bis ins obere Management eskalierte. Dort angekommen, war die eigentliche Detailfrage des richtigen Maßes für Qualität sicher komplett falsch aufgehoben. In dieser Zeit habe ich angefangen, mich als Tester unwohl zu fühlen, wenn wir im Test zuweilen sehr gründlich vorgehen und streng urteilen. Etwas, das situationsbedingt durchaus sinnvoll und nützlich ist.

Die Gegenbewegung war für mich bald spürbar. Projektleiter waren gut beraten, ihre Projekte im Alleingang zu organisieren, ohne eine Bürokratie an ihrer Seite, die Innovation eher verhindern würde, anstatt zielstrebig das Projekt nach vorne zu bringen. Damit hat man die partielle Amnesie einer Organisation geradezu perfektioniert. Jedes Projekt startet quasi bei Null und muss versuchen aus den individuellen Fähigkeiten seines Teams ein sinngebendes Ganzes zu formen. Dabei spielt es übrigens keine Rolle, ob diese Projekte agil oder klassisch organisiert sind. Eine agile Vorgehensweise bringt zwar die Optimierung mit, garantiert sie leider auch nicht – genauso wenig wie klassische Prozessmodelle.

Irgendwie kann es das aber auch nicht sein! Als Berater fühlt man sich da als Sisyphos, der nach jedem Projekt zusehen muss, wie der Stein wieder talwärts rollt. Wo bleibt das kollektive Gedächtnis in der Organisation? Wie kann ich die Erinnerung wach halten? Erfahrung heißt noch gar nichts, ich muss auch als Gruppe aus ihr gelernt haben. Wir brauchen also einen Kompromiss aus einer kompetenten Gruppe an Testern und einem autarken Projektteam. Im agilen Umfeld muss ich dieses Wissen sogar breit im Team verankern – und brauche immer noch hier und da Spezialisten (z.B. für Sicherheitstests).



Abb. 2 Als Berater im Test fühlt man sich wie Sisyphos, wenn jedes Projekt wieder von vorne anfängt, Erfahrung zu machen.

>

Erfahrung im Test

Auf den richtigen Mix kommt es an

Derzeit formt sich in vielen Unternehmen eine Art Dachorganisation für Test heraus, die das Testvorgehen projektübergreifend koordiniert und fördert, andererseits aber die Projekte nicht unnötig bevormundet. Einige nennen das Test Service Center, nicht einfach nur Test Center, oder auch Test Competence Center. Das Problem ist hier der richtige Mix aus Weisungsbefugnis und reinem Experten-Treff. Hier ist noch viel zu tun. Der einflusslose Experten-Treff ist weder neu, noch wirklich hilfreich. Die Weisungsbefugnis mit Blick auf die Tradition der Testcenter als unternehmenseigene Behörde auch nicht. Dennoch gibt es eine Reihe von Aufgaben, die sich gemeinsam koordiniert bzw. projektübergreifend besser lösen lassen: Wer kennt sich gut mit Sicherheitstests aus? Welche Testumgebungen kann ich verwenden? Gibt es gemeinsam nutzbare Testdaten? Wer übernimmt die Systemintegrationstests? Welche Best Practice im Fehlermanagement nutzen wir? Gibt es schon Werkzeug-Lizenzen, die wir im Projekt nutzen können? Auch in agil arbeitenden Projekten entwickeln sich derzeit übergeordnet koordinierende Teams (wie z. B. beim [Nexus Framework](#)). Der Kniff liegt also im richtigen Mix der Maßnahmen und Zuständigkeiten.

Aber nicht nur. Die eigentliche Kunst ist, die Menschen im Projekt mit dem jeweils nötigen Wissen zu versorgen und dann begleitend im Alltag das Wissen immer wieder zu festigen, sodass die kollektive Erfahrung sich irgendwann wie selbstverständlich in die Handlungen jedes Einzelnen einprägt. Das richtige Wissen kann man gut aus den Teststandards ableiten, muss es aber unbedingt über die Erfahrung im Unternehmen an die Bedürfnisse anpassen. Wie man beispielsweise am besten den Testaufwand schätzt, sollte nicht mit Allgemeinplätzen, sondern mit konkret belegten Erfahrungen aus dem Unternehmen unterlegt sein. Auch die Techniken eines Change-Managements sind nicht neu, sie müssen nur überzeugend angewendet werden. Und – ja leider – man braucht einen ganz, ganz langen Atem. Wie zu Beginn zitiert von Tucholsky: „Erfahrung heißt noch gar nichts.“ Denn nur was ein Mensch aus freien Stücken angenommen hat, das behält er auch.

◀ (Martin Klonk)

[Der Autor]

DI Martin Klonk, PMP
Senior Berater Software-Test,
ANECON

Martin Klonk ist seit 2011 Senior Berater für Software-Test und Qualitätssicherung bei ANECON. Seit seiner Ausbildung zum Diplom Ingenieur für Elektrotechnik an der Technischen Universität Berlin war er bei dem deutschen Beratungsunternehmen SQS zunächst in Köln und anschließend in Wien tätig. Der Schwerpunkt seiner Tätigkeit lag hierbei in der Telekommunikation, bei Banken und Versicherungen und es gibt inzwischen kaum einen Bereich im Software-Test, in dem der Senior Berater nicht schon erfolgreich tätig war. Martin Klonk ist Mitglied des Austrian Testing Boards, hat auch an den Lehrplänen des ISTQB® mitgearbeitet und ist Co-Autor des Buches „Agile Testing. Der agile Weg zur Qualität“.



Hier könnte Ihr Inserat stehen!

Haben Sie ein interessantes Testtool?
Suchen Sie einen qualifizierten Tester für Ihr Team?

Nutzen Sie die Möglichkeit, sich direkt an die Testcommunity zu wenden. Kontaktieren Sie das ATB, wenn Sie nähere Informationen zu den Konditionen für Inserate oder Partnerschaften haben wollen.

[Testen eines Services via API - Ein Erfahrungsbericht aus der Praxis]

von Klaus Schlacher

Schon oft hatte ich in meiner langjährigen Praxis als Softwaretester Berührungspunkte mit Schnittstellen. Sei es als leidtragender Tester einer Komponente, die ein Service nutzt, aber nicht immer die erwartete Antwort bekommt, oder aber in der Rolle als Last- und Performancetester, der ein Webservice aus diesem anderen Blickwinkel betrachtet.

Die Gelegenheit, ein Service via API (application programming interface) selbst in der Praxis funktional zu testen, bekam ich vor circa eineinhalb Jahren. Zu diesem Zeitpunkt stand die Entwicklung eines komplexen Services, das sich um langwierige Berechnungen kümmern soll, in den Startlöchern. Die Berechnungen basieren einerseits auf einem Regelwerk, das vom Fachbereich in einer Oberfläche definiert wird, andererseits aber auch auf Kundeneingaben und externen Services, die weitere Daten anliefern. Auch die unterschiedlichen Clients, die dieses Service nutzen, beeinflussen die Berechnung.

Diese Komplexität wird durch eine Besonderheit noch verschärft: Das Service existiert bereits, ist allerdings mit veralteter Technologie umgesetzt. Es soll nun mit Hilfe von neuen Technologien als "drop-in replacement" umgesetzt und abgelöst werden. Die Hauptgründe dafür sind die Möglichkeit auf eine weit verbreitete Programmiersprache zu setzen und damit auch weitere Clients zu unterstützen sowie eine zukünftig bessere Wartbarkeit des Services zu erreichen. Viele Mitarbeiter kennen das Bestandssystem und erwarten auch für die zahlreichen Spezialfälle von Berechnungen ein korrektes Ergebnis. Die fachlichen Anforderungen sind bereits lange bekannt, in manchen Details aber auch nicht dokumentiert. Aufgrund des "drop-in replacements" ist natürlich auch der Aufbau des Requests und des Responses bereits fixiert. Dafür gibt es allerdings keine Dokumentation. Dies ist bei einem JSON Response, der im Schnitt ca. 200.000 Zeichen lang ist (entspricht ca. 130 Buchseiten), durchaus eine Herausforderung.

Entwicklungsnahes Testen

Das Team, das dieses Service nun umsetzt, ist größtenteils ein cross functional Team. Analysten, Entwickler und Tester arbeiten gemeinsam an der Umsetzung. Da das Service Teil eines großen Projekts ist, gibt es natürlich Vorgaben bzw. Empfehlungen für eine entsprechende Vorgehensweise. Continuous Integration und damit einhergehend automatisierte und im Build integrierte Tests werden als sehr wichtig erachtet. In unserem Team stieß diese Vorgabe natürlich auf offene Ohren. Tester und Entwickler nutzen gemeinsam ein GIT Repository in dem zusätzlich zu den Sourcen auch die Testfälle abgelegt sind. Der große Vorteil dabei ist, dass die Entwickler bei jeder Gelegenheit ihre eigenen Unit Tests und die von den Testern erstellten Servicetestfälle lokal – innerhalb ihrer Entwicklungsumgebung – durchführen können. Der Entwickler erhält so unmittelbar Feedback über seine Code-Änderungen und ihre Auswirkungen werden zeitnah aufgezeigt. Beide Ausprägungen von Testfällen wurden mit dem JUnit Framework umgesetzt.

Bearbeitung des Requests

Servicetestfälle folgen grundsätzlich immer demselben Ablauf: Erstellen und konfigurieren des Requests, absenden des Requests (und Verarbeitung durch das Service) und auswerten des Responses:

- **Erstellen und Konfigurieren des Requests:** Für den Request werden Kundeneingaben sowie der Input eines externen Services benötigt. Um die Kundeneingaben möglichst komfortabel gestalten und nachbilden zu können, wurde von den Entwicklern gemeinsam mit den Testern eine Builder Klasse erstellt. Diese stellt Methoden bereit um die Kundeneingaben in allen Varianten in den Request des Testfalls zu packen. Auch der Input des externen Services wird komfortabel über eine weitere Builder Klasse erstellt. Weitere externe Services, die während der Berechnung aufgerufen werden, wurden mit

entsprechenden vorgefertigten Responses, die im Filesystem abgelegt sind, gemockt. Das Mocken der Responses ist unbedingt notwendig, da diese für das Ergebnis der Berechnung relevant sind. Andernfalls würde es zu unerwarteten Testergebnissen kommen. Das Entwickeln gegen diese Testfälle sowie ein Buildjob, der die Testfälle bei jedem Push ins GIT durchführt, wäre ohne gemockten Inhalt sehr instabil und nicht aussagekräftig.

- **Verarbeiten des Requests:** Dies erfolgt immer durch einen Aufruf, welcher den Request als Input bekommt und den entsprechenden Response zurückliefert. Da dies in der Entwicklungsumgebung direkt mit den Java Objekten erfolgt, kann bei Bedarf die Berechnung von den Entwicklern zur Fehleranalyse debugged werden.
- **Auswerten des Response:** Darin liegt nun natürlich der Schwerpunkt. Je nach Fokus des Testfalls werden spezifische Teile der Response geprüft. Hauptsächlich wird hier auf den Inhalt von Elementen, das Vorhandensein oder Nicht-Vorhandensein von Elementen sowie die Anzahl der Elemente geprüft. Die Checks wurden mit Hilfe von Hamcrest Matchers¹ realisiert. Diese liefern für den Testfall entsprechend ein „passed“ oder „fail“. Es gibt eine Vielzahl von Matchern für diverse Checks sowie die Möglichkeit, bei komplexeren Strukturen oder Überprüfungen, eigene Matcher zu entwickeln. Hat man nun eine Handvoll von Checks, die für einen speziellen Testfall herangezogen werden, so ist dies noch recht überschaubar. Ist dann der Source für diese Funktionalität des Services bereits entwickelt und korrekt, so wird der Testfall, wenn er lokal in der Entwicklungsumgebung durchgeführt wird, auf Anhieb grün, also fehlerfrei, durchlaufen. Leider existiert dieser Idealzustand recht selten und der Testfall wird mitunter einen Fehler liefern.

>

Testen eines Services via API

Mögliche Fehlerquellen an dieser Stelle sind die Funktionalität der Software, die externen Testdaten und natürlich der Testfall selbst. Dabei können in diesem Projekt die gemockten Daten, die Kundeneingaben oder aber die Checks selbst der Auslöser sein. In der Praxis hat sich herausgestellt, dass es durchaus eine Mixtur der zuvor erwähnten Quellen sein kann.

Vorgehensweise zur Testfallerstellung

Um nun korrekte Testfälle schreiben zu können entwickelte ich folgende Vorgehensweise, die das bestehende Service sowie SoapUI nutzt:

- 1.) **Die entsprechenden Regelwerke für den Testfall vorbereiten:** Dies geschieht in der anfangs erwähnten Oberfläche.
- 2.) **Den Testfall mit Kundendaten und Daten vom externen System vorbereiten:** Der Testfall ist nun in der Entwicklungsumgebung vorhanden und grundsätzlich lauffähig. Ein erster Versuch zeigt, ob dies gelingt. War dieser Check positiv kann der vom Builder erzeugte Request String für den nächsten Schritt herangezogen werden.
- 3.) **Den erzeugten Request an das bestehende Service senden:** Das bestehende Service muss dabei im Netzwerk verfügbar sein und die gleichen vorbereiteten Regelwerkdaten benutzen. Das Senden des Requests gelingt sehr einfach mit SoapUI².
- 4.) **Den Response überprüfen:** Der hier erzeugte Response kann nun auf Fehler in den Kundendaten, Daten des externen Services oder Regelwerkdaten manuell überprüft werden. Um den JSON-Response einigermaßen lesbar zu machen gibt es sehr viele JSON Viewer (teilweise auch online verfügbar). Mir persönlich gefällt Notepad++³ mit dem JSTool Plugin⁴ sehr gut. Dies erlaubt es, den JSON in einer übersichtlich formatierten Baumstruktur darzustellen.
- 5.) **Checks im Testfall erstellen:** Nun können in diesem formatierten JSON die essentiellen Elemente, die es zu überprüfen gilt, rausgesucht und entsprechende Checks im Testfall dafür erstellt werden. Kleine Tippfehler im Pfad oder dem Namen des zu überprüfenden Elements oder aber auch die Verwendung der falschen Hamcrest Methode führen sehr schnell zu „false negativ“ Ergebnissen des Testfalls.

6.) **Checks überprüfen:** Um nun einen Testfall mit korrekten Checks erstellen zu können benutze ich die Mock Services von SoapUI. Damit wird der gesamte Service Response gemockt, die Checks im Testfall laufen somit gegen die zuvor ermittelte Response des Bestandsservices. Fehlerhafte Checks werden somit ausgeschlossen, egal ob das neue Service diese Elemente fehlerhaft oder noch gar nicht umgesetzt hat.

Fehleranalyse

SoapUI hat sich auch in der Fehleranalyse bewährt. Das GUI (graphical user interface) für das zu entwickelnde Service kann mit Datumswerten in der Vergangenheit nicht umgehen. Sind nun bekannte Fehler aus vergangenen Tagen zu analysieren, so ist es recht leicht die Situation in der Vergangenheit mit entsprechenden Requests nachzustellen.

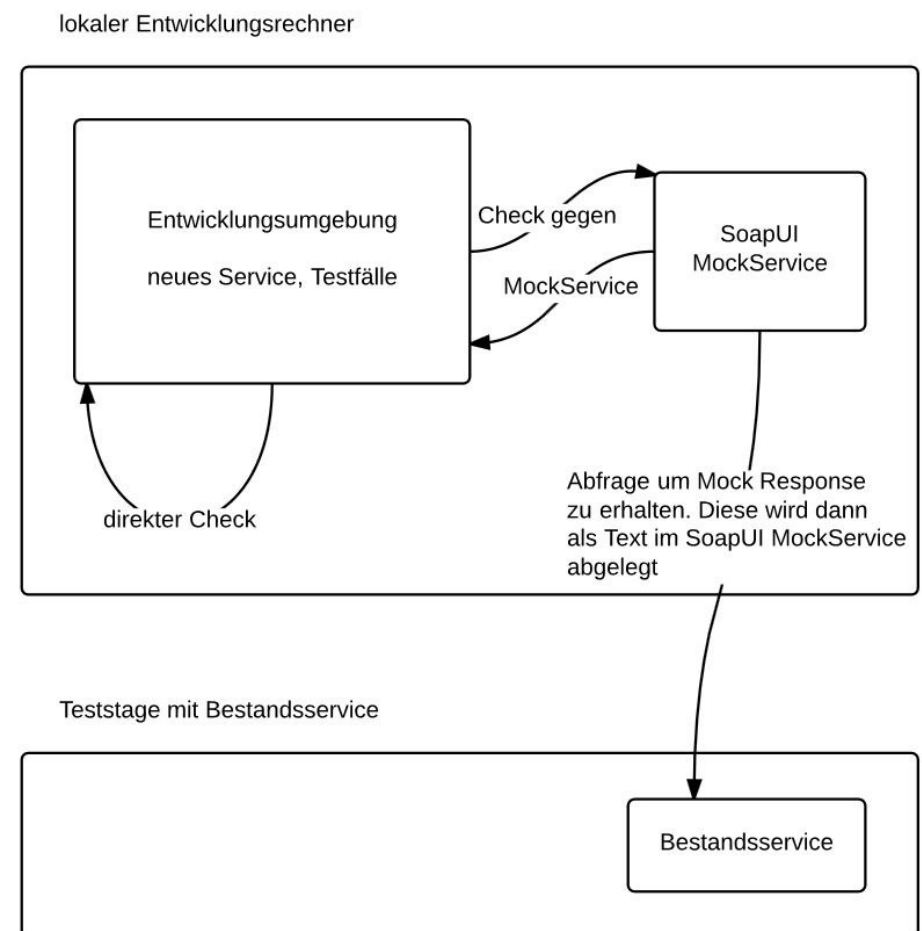


Abb.: Übersicht Testfallverifikation

Testen eines Services via API

Performance vorhersagen

Zusätzlich zur Funktionalität gibt es noch weitere Punkte, die bei einem Service nicht außer Acht gelassen werden sollten. Einer davon ist die Performance. Um einen Trend zu erkennen, wie sich das Service über die Zeit durch Implementierung weiterer Features oder Refactoring entwickelt, wurde ein Performance Indikator etabliert. Dazu wurde JMeter verwendet. Ausgewählte Requests werden durch einen Jenkins Task jede Nacht durchgeführt. Die entsprechenden Durchschnittswerte jeder Probe werden weiterverarbeitet und in einer Grafik eingefügt. Natürlich ist diese Performance Indikator Lösung kein Ersatz für einen echten Last- und Performance-Test.

Durch dieses Vorgehen ist das Entwickeln des Services schon gut abgesichert. Wird nun dieses Service durch Jenkins erzeugt, so werden ebenfalls, wie bereits erwähnt, diese Servicetestfälle gemeinsam mit den Code Unit Testfällen durchgeführt. Dennoch kann es sein, dass dieses Service dann in Kombination mit anderen Services oder im Einsatz mit der entsprechenden GUI fehlerhaft arbeitet. Um dies zeitnah zu erkennen wurden automatisierte Testfälle für Smoke- und Regressionstests erstellt. Diese können bei Bedarf, zumeist nach einem neuen Deployment auf einer Stage und vor einer Freigabe des Services durchgeführt werden.

Das Set basiert größtenteils aus den Regressionstestfällen zum bestehenden Service. Ein weiterer wichtiger Faktor ist ein kurzer Check mittels GUI. Hier entdeckt man oft kleine Unzulänglichkeiten, die mitunter tief im Response durch ein einzelnes Element repräsentiert werden. Da es viele Sonderfälle wie auch Elemente gibt, die sich in der GUI nicht niederschlagen und diese Elemente im Response teilweise auch nicht immer vorhanden sind, wurde noch eine weitere Testvariante etabliert. Ziel ist es, die Responses des Bestandservices und des neuen Services zu vergleichen. Dazu muss der idente Request durch beide Services mit identischer Datenbasis verarbeitet werden. Der Vergleich der beiden Responses erwies sich allerdings als nicht sehr praktikabel, denn einerseits sind viele bewusste Ausnahmen zu definieren, andererseits machen die Unsortiertheit und die komplexe Struktur Probleme.

Generell denke ich aber, dass mit dieser Vorgehensweise eine effiziente Testfallerstellung und damit auch ein hoher Grad an Qualitätssicherung möglich ist. Dies hat sich auch durch die hohe Akzeptanz im Praxiseinsatz dieses Projektes gezeigt.

Link-Nachweis:

¹ Hamcrest Matcher:

<http://hamcrest.org/JavaHamcrest/javadoc/1.3/>

² SoapUI:

<http://soapui.org/>

³ Notepad ++:

<https://notepad-plus-plus.org/>

⁴ JSTool Plugin:

<http://sourceforge.net/projects/jsminnpp/>

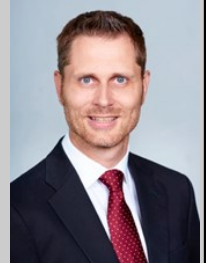
◀ (Klaus Schlacher)

[Der Autor]

Klaus Schlacher, BSc

ist Senior Consultant bei SEQIS. Bereits während seiner Schulzeit erkannte er, dass Computer und die darauf laufende Software eine faszinierende Anziehungskraft für ihn haben.

Er richtete Ausbildung und Berufswahl entsprechend aus, um als Softwareentwickler durchzustarten. Auch nach über 10 Jahren ist der Blick durch die Softwarereporter-Brille mit technischem Background seine treibende Motivation, welche von Kunden und Kollegen gleichermaßen geschätzt wird.



Softwaretest in Praxis und Forschung

Umfrage 2015

www.softwaretest-umfrage.de

Zu den rollenbezogenen Fragebögen der Umfrage 2015:

Vom 15.9. – 15.12.2015

Mitmachen, Mitgestalten

Die aktuelle Umfrage zu Entwicklungen und Trends im Bereich Testen und Qualitätssicherung gibt einen 360°-Blick vom Management über das operative Testen zu Forschungsfragen. Aufbauend auf der „[Umfrage 2011: Softwaretest in der Praxis](#)“ wird ermittelt, was sich in den letzten Jahren konkret verändert hat und wo Nutzen, Stärken aber auch Defizite in Praxis und Forschung liegen. Damit stehen Grundlagen für ein Benchmarking zur Verfügung und Impulse für eine praxisorientierte Ausrichtung der Forschung sowie der Aus- und Weiterbildung.

Die Umfrage dient wissenschaftlichen Zwecken

Testsystem	Grad der Automatisierung (%)
Testumgebung (CI/CD)	100
Definierung und Testanforderungen	100
Testmanagement	100
Performance-Lasten/Tools	100
Daten-Generierung/Generierung	100
Statistische und dynamische Analyse	100
Testfallerstellung	100
Code-Überprüfung	100
Simulation-Emulation/Generierung	100
Generierung von Testdaten/Generierung	100
Strukturierte Testumgebung	100
Hybride Testumgebung	100
Performance-Generierung	100

Abbildung 4-128: Einsatz von Testwerkzeugen

[Was macht Qualität bei mobilen Applikationen aus? Mobile Testing als anspruchsvolle Disziplin im Fokus der 2. Mobile Quality Night]

von Christoph Börner

Die Themen der 2. Mobile Quality Night in Wien am 8. Oktober drehten sich dieses Jahr um die schwierigen Fragen: wie kann Mobile App-Testing möglichst effizient durchgeführt werden und wie wird mit der Herausforderung von immer größerer Diversifikation in der Gerätewelt umgegangen. Gerade in der Entwicklung und im Test von mobilen Applikationen sind kreative Ansätze und neue Methoden gefordert, um den Wünschen und Ansprüchen der User gerecht zu werden und hohe Qualität zu gewährleisten.

Markus Murtinger erörterte im ersten Vortrag die Anforderungen an die Usability bei mobilen Geräten. Hier gilt es nicht nur die Funktionen der Anwendung benutzerfreundlich zu gestalten, sondern auch Kontext, User und Situation in die Überlegungen und das Design mit einzubeziehen.

Im zweiten Vortrag sprach Dan Cuellar, Erfinder von Appium, über Testautomation und zeigte anschaulich, wie ein Roboter laufend auf Appium einzelne Tasks eigenständig durchführen kann.

Bei den zwölf unterschiedlichen Lightning Talks wurde in fünf Minuten erläutert, was die Qualitätsfaktoren bei mobilen Anwendungen sind, wie Theorie und Praxis von Testautomation im Mobile Testing aussieht und wie die Qualitätssicherung den hohen Standards gerecht werden kann. Christoph Börner von TestPlus, Platin-Sponsor des Events, beleuchtete in seinem Vortrag die zukünftigen Herausforderungen im Mobile Testing in Bezug auf neue Technologien und Benutzungskontexte.

Bei der abschließenden Podiumsdiskussion erläuterten Experten die Problematik von immer größerer Gerätevielfalt und die Auswirkungen auf Mobile Testing.

Mit über 60 Gästen war die Veranstaltung ein voller Erfolg. Der Termin für die 3. Mobile Quality Night in Wien steht schon fest: Das Event wird am 6. Oktober 2016 im Stockwerk stattfinden. Sie können wieder zahlreiche Experten und Expertinnen zu den Herausforderungen und Entwicklungen im Mobile Testing erwarten. Aktuelle Informationen erhalten Sie hier: <http://www.mobile-quality-night.com/>.

◀(Christoph Börner)

[Der Autor]

Christoph Börner, Testplus GmbH
Geschäftsführer, Head of Test & Development

Christoph Börner kann auf langjährige Erfahrung als Software-Tester und Entwickler zurückgreifen. Während seines Informatikstudiums an der TU Wien, gründete er erst die Testplus KG und danach die Testplus GmbH. Als Head of Test & Development entwickelt er heute gemeinsam mit seinem Team innovative Testlösungen von morgen. Seine Spezialgebiete sind Mobile Testing, Crowdfunding, Load- und Performance Testing, Testautomation, Code Reviews, Projekt- und Business Process Management. Darüber hinaus arbeitet Christoph am Institut für Gestaltungs- und Wirkungsforschung der TU Wien und leitet gemeinsam mit Rudolf Grötz die Vienna Mobile Quality Crew.



[ASQF zur EU-Neuregelung der Netzneutralität]

Die Mehrheit der Abgeordneten im Europäischen Parlament stimmte heute für eine neue Verordnung zur Netzneutralität. Der ASQF e.V., das größte Expertennetzwerk für Softwarequalität im deutschsprachigen Raum, sieht diese Entscheidung kritisch. ASQF-Hauptgeschäftsführer Stephan Goericke sagte dazu: „Mit dem neuen EU-Gesetz wird das Zweiklassen-Internet vorangetrieben und die Vielfalt des Netzes immer weiter eingeschränkt. Kommunikative Freiheiten wie wir sie bisher kannten, wird es nicht mehr geben.“

Die neue EU-Verordnung wird für die Unternehmen in der Softwareentwicklungsbranche nicht ohne Folgen bleiben und zusätzliche Kosten verursachen. Auch wenn die Prinzipien „gleich“ und „diskriminierungsfrei“ im Sinne der Netzneutralität bemüht werden, dem Gesetz fehlt eine konkrete Eingrenzung von Begrifflichkeiten. Dieser handwerkliche Makel (ob bewusst oder unbewusst) lässt viele Fragen offen und bedeutet ein unkalkulierbares Risiko. Kostenpflichtige Spezialdienste, der Grundstein für das Zweiklassen-Internet, könnten künftig nicht die Ausnahme, sondern die Regel darstellen. Das wird sowohl für Nutzer als auch Anbieter von innovativen Diensten und Produkten spürbare Folgen haben. Für junge Start-Ups bedeutet es vor allem höhere Hürden beim Markteintritt, weil sie sich teure Überholspuren für ihre Innovation kaum leisten können.“

In einer bereits zu Jahresbeginn durchgeführten [Mitgliederbefragung des ASQF](#) bekräftigte eine klare Mehrheit von Führungskräften der IT-Branche (87 Prozent) die Forderung nach einem frei zugänglichen Internet. Allerdings bezweifelte ein Großteil unter ihnen, dass sich die aktuelle Bundesregierung ausreichend für dieses Ziel einsetze.

Ähnliche Ergebnisse lieferte eine repräsentative Umfrage bei der 1.034 wahlberechtigte Deutsche gefragt wurden, ob sie die Netzneutralität beibehalten möchten oder sie für verzichtbar halten. 75 Prozent sprachen sich für die Beibehaltung aus, 13 Prozent hingegen hielten die Netzneutralität für verzichtbar. Lediglich 11 Prozent gaben an, das Thema sei ihnen egal, interessiere sie nicht oder sie wollten sich kein Urteil dazu bilden.

◀(Christin Senftleben)

Neues aus dem ASQF e.V.

[Ina Schieferdecker ist neue Präsidentin des ASQF e.V.]

Ina Schieferdecker steht an der Spitze des größten Kompetenznetzwerks für Softwarequalität im deutschsprachigen Raum. Auf der jüngsten Mitgliederversammlung Ende Oktober in Erlangen (Deutschland) übernahm die Informatik-Professorin den Staffelstab für das Ehrenamt von ihrem Vorgänger Rudolf van Megen. Schieferdecker ist Leiterin des Fraunhofer FOKUS, Professorin an der Freien Universität Berlin und gehört seit 2009 zur Führungsspitze des ASQF (Arbeitskreis Software-Qualität und -Fortbildung e.V.). Als Vizechefin forcierte sie gemeinsam mit van Megen die weitere Öffnung und Internationalisierung des Verbandes. In ihrer neuen Funktion will sie sich mit dem ASQF besonders den Herausforderungen einer zuverlässigen und sicheren digitalen Transformation stellen.

Breit aufgestelltes Präsidium

Unterstützt wird Ina Schieferdecker vom neu gewählten Präsidium, das aus den Vizepräsidenten Ludger Meyer (Research und Technology Center der Siemens AG) und Norbert Kastner sowie den Präsidiumsmitgliedern Manfred Baumgartner (ANECON Software Design und Beratung GmbH), Prof. Dr. Joachim Hornegger (Präsident der Friedrich-Alexander-Universität Erlangen-Nürnberg), Prof. Dr. Karin Vosseberg (Hochschule Bremerhaven mit den Schwerpunkten IT-Systemintegration und Software Engineering) und Günter Limböck besteht. Sie bringen Erfahrung aus Wissenschaft und Wirtschaft ein und verfügen über langjährige Erfahrung auf ihrem Gebiet.

ASQF soll Unternehmen bei der Digitalisierung begleiten und unterstützen

„Der ASQF wird als wichtiges Netzwerk für Softwarequalität wahrgenommen und ist über die iSQI GmbH, deren Gesellschafter der Arbeitskreis ist, auch weltweit aktiv. Für mich ist die digitale Transformation das zentrale Thema für die IKT-Branche in den nächsten Jahren. Das Internet der Dinge, Industrie 4.0 und auch Smart Cities gehen mit ihr einher und stellen hohe Anforderungen an die Softwarequalität. Den technischen Kern der digitalen Transformation bilden software-basierte Systeme, die flexibel, offen, verteilt und vernetzt funktionieren müssen. Ich möchte als ASQF-Präsidentin vor allem kleine und mittlere ITK-Unternehmen dabei unterstützen, sich an diesem Prozess zu beteiligen. Das ASQF-Netzwerk mit Partnern aus Wissenschaft und Wirtschaft ist dafür ideal“, fasst Ina Schieferdecker ihre Ziele für den Verband zusammen.



Staffelstabübergabe zwischen der neuen ASQF-Präsidentin Ina Schieferdecker und ihrem Vorgänger Rudolph van Megen.



Das neue ASQF-Präsidium (v.l.n.r.): Prof. Dr. Joachim Hornegger, Norbert Kastner, Prof. Dr. Ina Schieferdecker, Günter Limböck, Prof. Dr. Karin Vosseberg, Manfred Baumgartner. Nicht im Bild: Ludger Meyer

Netzwerk mit vielen Perspektiven

Ina Schieferdecker übernimmt ein gut ausgebautes Netzwerk, das in den vergangenen Jahren auf kontinuierlich steigende Mitgliederzahlen verweisen konnte. Über rund 1.400 private und Firmen-Mitglieder werden mehr als 10 000 Vertreter der Software-Branche im deutschsprachigen Raum erreicht. Der Verband macht sich insbesondere für die Vermittlung und Weiterentwicklung von Qualitätssicherungs-Know-how in der Softwarebranche und für eine einheitliche Aus- und Weiterbildung von IT-Fachkräften stark. Zu seinen Mitgliedern gehören sowohl führende Unternehmen der IKT-Branche als auch klein- und mittelständische Firmen und Experten aus Wissenschaft und Forschung.

◀(Christin Senftleben)

[Effiziente Nutzung von Big Data]

Datenberge an sich schaffen keinen Nutzen, es geht um ihren Informationsgehalt. Die betriebswirtschaftlich sinnvolle Nutzung von Big Data zählt zu einer der künftigen Kernaufgaben für Unternehmen und ist das Schwerpunktthema des nächsten SQ-Magazins. Die neue Ausgabe erscheint am 1. Dezember 2015. Mehr Infos im Internet auf www.sq-magazin.de

◀(Christin Senftleben)

[Die Autorin]



Christin Senftleben ist Referentin für Communications & Events des International Software Quality Institute (iSQI) mit Standorten in Deutschland, Großbritannien, den Niederlanden und in den USA. Im Jahr 2014 zertifizierte iSQI weltweit mehr als 18.000 Personen und förderte damit die Aus- und Weiterbildung von qualifizierten Fachkräften gemäß international einheitlichen Standards.

[„Agile Skalierung mit Kanban“ & „Mobile Testing“]

von Alexander Weichselberger

Richtig! Noch 1x Expertentreff in diesem Jahr, dann geht's in die Weihnachtsferien, und wenn alles klappt, stehen wir nach dem Jahreswechsel vor dem nächsten Paket Expertentreffs aka „Networking mit Schwerpunktthemen“ (so beim letzten Treff durch eine Teilnehmerin zusammengefasst und auf den Punkt gebracht). Aber bleiben wir in dieser Ausgabe des Newsletters noch im heurigen Jahr ☺.

Rückblick auf den Treff im Oktober – Agile Skalierung mit Kanban

Klaus Leopold, leanability.com, at his best! Pragmatische Einführung ins Thema, rege Diskussion mit den Teilnehmern, gute Stimmung und kreativer Austausch aka Networking. Für mich als Moderator ein „Heimspiel“. Zusammengefasst hab ich mir aus dem Vortrag folgende 3 Top-Punkte mitgenommen:

1. Sich agil zu organisieren bedeutet nicht, dass an allen Ecken und Enden der Organisation auf Basis agiler Methoden gearbeitet werden muss. Für mich persönlich eine wesentliche Feststellung – sonst müssten ITIL oder andere gut eingeführte Normen auf deren Standards verzichten - das hätte keinen Vorteil (denken wir z.B. an die Medizinbranche). Sich agil zu organisieren bedeutet vielmehr, agile Interaktionen im System zu ermöglichen.

2. Agile Skalierung basiert darauf, dass sich alle wesentliche Akteure gemeinsam auf eine Arbeitshypothese einigen und versuchen, diese strukturiert und im kontinuierlichen Verbesserungsmodus zu realisieren. Die Einigung auf den „aktuellen Irrtum“ sollte mit Augen-zwinkern aufzeigen, dass sich alles auch ändern darf und soll - dies sollte jedoch abgestimmt passieren.

3. Bei der agilen Skalierung geht es im Kern um Abstimmung und Synchronisierung zwischen den Gruppen. In diesem Umfeld brauchen die Teams, aber auch die Manager, entsprechende Soft Skills.

Es war wieder fein, mit anderen Interessierten über Details und Metaebene zu diskutieren!

Vorschau auf den Treff im November – Wolkig mit Aussicht auf hohe (Mobile) Device-Abdeckung

Ein, zwei Monate vor Weihnachten stellen sich viele Menschen die Frage „Was schenken wir zu Weihnachten?“ Wussten Sie, dass mobile Endgeräte seit Jahren Garant für Umsatzsteigerungen jenseits der 20% sind? Dass die Benutzer der Geräte, sobald ein neuer HW-Typ auf den Markt kommt, trotz der teilweise schon frechen Anschaffungskosten bereitwillig auf die jeweils neuen iPhones und Android-Devices umsteigen? Und neben dem B2C haben auch im B2B Smartphones im Innovationsbereich längst die Nase vorn.

Smartphones & Tablets sind also ein Hit – wären sie allerdings „nur“ Telefone, würde niemand diesen Trend unterstützen. Und was macht ein Smartphone so reizvoll? Richtig, die Software und Zusatzfunktionen auf den Geräten. Aber nur, wenn sie funktionieren...

Die Fragen, die wir SoftwaretesterInnen in diesem Kontext lösen müssen, sind:

- Wie kann die Testabdeckung mobiler Endgeräte erhöht werden?
- Wie kann man die Anwendungsbereitstellung beschleunigen, aber dabei Kosten und Risiken reduzieren?
- Wie setzt man Testautomation in einem mobilen Projekt auf?

Beim Expertentreff am 18.11.2015 wird **Rudolf Grötz** unter dem Titel „Wolkig mit Aussicht auf hohe (Mobile) Device-Abdeckung“ die speziellen Anforderungen der Gerätesegmentierung aufgreifen und praxisbasierte Vorschläge machen, wie dieses Thema anhand von Mobile Device Clouds (MDC) gelöst werden kann. Schwerpunkte der Information werden manuelle und automatisierte Test im Zusammenhang mit MDC sein, obenstehende Kernfragen stehen im Mittelpunkt des Vortrags.

Im Anschluss werden Vortragsinhalt und weiterführende Themen moderiert diskutiert. Die Diskussionspartner sind **Alexander Vukovic** (SEQIS Software Testing GmbH) sowie ein weiterer Experte.

Alles Liebe,
Ihr
Alexander Weichselberger
ATB Expertentreff

P.S. Wie immer im Berufsleben sind eigenverantwortliche Ausbildung und Networking wichtig für Rückhalt und Profession: Kenn ich mich aus, kann ich auch in schwierigen Situationen die eine oder andere bessere Wahl treffen. Und bin ich vernetzt, habe ich auch noch Jemanden (außerhalb), den ich kontaktieren und um Hilfe bitten kann. Beides bieten wir im Rahmen der ATB Expertentreffs – dh. kommen Sie zu unseren Veranstaltungen und vernetzen Sie sich.

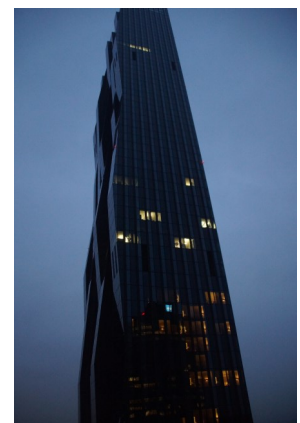
◀(Alexander Weichselberger)



Von links:
Klaus Leopold, Alexander Vukovic,
Helmut Pichler, Alexander Weichselberger



Wie immer gab es eine rege Diskussion mit vielen Wortmeldungen aus dem Publikum



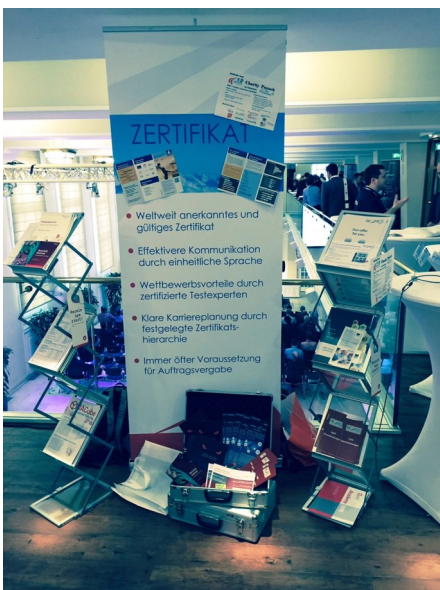
Gleich nebenan der DC Tower, derzeit das höchste Gebäude in Österreich

Accelerate 2015

Am 22. Oktober 2015 fand in Wien die alljährliche User Konferenz von Tricentis zum Thema „Testing in the Digital Enterprise Age“ statt. Über 200 Teilnehmer aus aller Welt und aus unterschiedlichsten Branchen trafen sich im Novomatic Forum, um Erfahrungen und Ideen auszutauschen und die neuesten Trends und Innovationen im Bereich Softwaretesten zu diskutieren. Ein spannender Tag voller Vorträge und Workshops erwartete die Teilnehmer. Viele namhafte Aussteller wie Infosys, Mobile Labs, ANECON, Neotys, QACube, Tasktop, Wipro und Service Now präsentierten ihre Lösungen und Produkte. Auch das Austrian Testing Board war wieder mit einem Stand vertreten. Den Abschluss dieses ereignisreichen Tages bildete ein extravagantes Dinner mit viel Zeit für interessante Gespräche.

Ein kurzes Video mit den Höhepunkten der Accelerate 2015, sowie Videos von den Vorträgen und einige Bilder sind bereits auf der [Webseite](#) verfügbar.

◀ (Nicole Neumayer)



Der Stand des ATB auf der Accelerate



Das ATB (H.Pichler) begrüßt SIXSENTIX/QACube als neuen GoldPartner des ATB – Herzlich Willkommen

DACH Meeting

[STB lädt ATB & GTB zum DACH Meeting nach Zürich]

von Helmut Pichler

Nachdem die DACH Boards letztes Jahr beim GTB in Hamburg zu Gast waren, lud dieses Jahr das Schweizer Testing Board nach Zürich. Bei dem mittlerweile schon traditionellen Treffen wurden aktuelle Themen und Strömungen im ISTQB, aber auch innerhalb unserer nationalen Boards, besprochen und abgestimmt.

Nach der kompletten Neuaufstellung an der Spitze des ISTQB gab es dieses Jahr ja auch einen Wechsel im Vorstand des GTB. Solche Wechsel bringen immer neuen Wind in alte Strukturen, so auch in diesen beiden Fällen. Neben der Vorstellung der neuen GTB Strategie lag der Schwerpunkt vor allem bei ISTQB Themen und zwar konkret bei der anstehenden geplanten Erweiterung der Zertifikatslandschaft. Welche davon sehen wir als zukunftsweisend, welche Abhängigkeiten sieht das ISTQB, welche können wir unterstützen, welche sehen wir als kritisch... bei solch wichtigen Themen vergeht die Zeit wie im Flug. Abschließend gab es auch noch eine Abstimmung der DACH Arbeitsgruppen darüber, wie wir über-regional „Fahrt aufnehmen“ können und werden.

Nach einem arbeitsreichen Samstag und einem zünftigen „Znacht“ (Abendessen) hieß es am Sonntag: „Auf zum Sport“. Wenngleich einige von uns die Sportart „Curling“ erst googlen mussten ;o).

Ich hätte nicht gedacht, dass ein ovaler, 20 kg schwerer Stein mit Henkel sowie Eisschrubben mit einem Besen so viel Spaß machen können. Und was wenige für möglich hielten: nach kurzer, kompetenter Einführung hatten wir das Grundprinzip & die Technik verstanden. Gut, verstanden heißt nicht, dass wir es auch anwenden konnten ;) – aber wir waren mit Eifer bei der Sache – fast wie die Profis... Ein herzliches Dankeschön ans STB und Andrea (Backoffice) für die tolle Organisation Und ein besonderer Dank auch an die ATBler: Hannes Bergsmann, Günther Wind, Andreas Günther, Martin Klonk und Richie Seidl, die mich tatkräftig unterstützt haben.

◀ (Helmut Pichler)





Kurs	Termin	Ort	Anbieter
E-Learning ISTQB® Certified Tester Foundation Level	Jederzeit	Anmeldung zum eCTFL	Software Quality Lab
ISTQB Certified Tester Foundation Level	10.11. – 13.11.2015	Lustenau	Software Quality Lab
	30.11. – 03.12.2015	Wien	ANECON
	11.01. – 14.01.2016	Wien	SQS
	11.01. – 14.01.2016	Linz, Wien, Graz	Software Quality Lab
	18.01. – 21.01.2016	Lustenau	
	01.02. – 04.02.2016	Linz, Graz	
	08.02. – 11.02.2016	Wien	SQS
	08.02. – 11.02.2016	Lustenau	Software Quality Lab
	15.02. – 18.02.2016	Wien	
	15.02. – 18.02.2016	Wien	
	07.03. – 10.03.2016	Wien	SQS
	14.03. – 17.03.2016	Mödling	SEQIS Software Testing GmbH*
	04.04. – 07.04.2016	Linz, Wien, Graz	Software Quality Lab
	11.04. – 14.04.2016	Lustenau	
ISTQB Certified Tester Foundation Level Englisch	22.02. – 25.02.2016	Linz, Wien, Lustenau	Software Quality Lab
ISTQB® Certified Tester Foundation Level Extension, Agile Tester	24.11. – 25.11.2015	Linz, Wien	Software Quality Lab
	19.01. – 20.01.2016	Wien	
	13.04. – 14.04.2016	Linz, Wien	
	10.03. – 11.03.2016	Mödling	SEQIS Software Testing GmbH
ISTQB Certified Tester Advanced Level Test Manager	30.11. – 04.12.2015	Linz, Wien, Graz, Lustenau	Software Quality Lab
	07.03. – 11.03.2016	Linz, Wien	
	09.03. – 15.03.2016	Wien	ANECON
	11.04. – 15.04.2016	Mödling	SEQIS Software Testing GmbH*
	18.04. – 22.04.2016	Wien	SQS
ISTQB Certified Tester Advanced Level Test Analyst	16.11. – 19.11.2015	Wien	SQS
	17.11. – 20.11.2015	Lustenau	Software Quality Lab
	22.02. – 25.02.2016	Linz, Wien	
	04.04. – 07.04.2016	Mödling	SEQIS Software Testing GmbH*
	19.04. – 22.04.2016	Wien	ANECON
ISTQB Certified Tester Advanced Level Technical Test Analyst	14.12. – 16.12.2015	Linz, Wien, Graz, Lustenau	Software Quality Lab
	15.03. – 17.03.2016	Linz, Wien	

*) in Kooperation mit Diaz-Hilterscheid



Kurs	Termin	Ort	Anbieter
Certified Agile Tester® Training (Prüfung in deutsch oder englisch)	22.02. – 26.02.2016	Wien	ANECON
	29.02. – 04.03.2016	Linz, Wien	Software Quality Lab
	25.04. – 29.04.2016	Mödling	SEQIS Software Testing GmbH
CATDD® Certified Agile Test Driven Development Training und Zertifizierung	09.02. – 11.02.2016	Wien	ANECON
	07.03. – 09.03.2016	Mödling	SEQIS Software Testing GmbH
IREB Certified Professional for Requirements Engineering Foundation Level	16.11. – 18.11.2015	Linz, Wien, Graz	Software Quality Lab
	24.11. – 26.11.2015	Lustenau	
	25.11. – 27.11.2015	Wien	ANECON
	14.12. – 16.12.2015	Wien	SQS
	12.01. – 14.01.2016	Lustenau	Software Quality Lab
	26.01. – 28.01.2016	Linz, Wien, Graz	
	01.03. – 03.03.2016	Linz, Wien, Graz	
	02.03. – 04.03.2016	Wien	ANECON
	08.03. – 10.03.2016	Lustenau	Software Quality Lab
	05.04. – 07.04.2016	Lustenau	
	11.04. – 13.04.2016	Wien	SQS
	19.04. – 21.04.2016	Linz, Wien, Graz	Software Quality Lab
IREB® Certified Professional for Requirements Engineering Elicitation and Consolidation Advanced Level	15.12. – 17.12.2015	Linz, Wien	Software Quality Lab
	09.02. – 11.02.2016	Linz, Wien	
CMAP© Mobile App Testing – Foundation Level	23.11. – 24.11.2015	Mödling	SEQIS Software Testing GmbH
	18.04. – 19.04.2016	Mödling	SEQIS Software Testing GmbH

[Impressum]

Herausgeber: Austrian Testing Board Alser Straße 4/Hof 1/Eingang 1.5
A-1090 Wien, Austria
Telefon: +43 676 64 35 688 , Fax: +43 2256 65969
Email: office@austriantestingboard.at.

Dieses Magazin richtet sich an Software-Tester im deutschsprachigen Raum. Anregungen, Feedback, Kritik und ähnliches richten Sie bitte an backoffice@austriantestingboard.at

Wenn Sie dieses Magazin abbestellen wollen, nutzen Sie bitte den Abmeldelink im Mail oder senden Sie eine mit Betreff „Storno Magazin“ an backoffice@austriantestingboard.at.

Sämtliche in diesem Magazin zur Verfügung gestellten Informationen und Erklärungen geben die Meinung des jeweiligen Autors wieder und sind unverbindlich. Das ATB übernimmt keinerlei Haftung und Gewähr, insbesondere auch für die Richtigkeit oder Vollständigkeit der darin enthaltenen oder referenzierten Informationen oder deren Anwendung, sowie Druckfehler oder Irrtümer und es werden keinerlei Garantien, Zusicherungen oder sonstige Rechtsansprüche daraus begründet.

Die Redaktion behält sich Kürzungen vor. In keinem Fall spiegeln Leserbriefe die Meinung der Redaktion wieder.



Veranstaltung	Termin	Ort	Anbieter
Tricentis Certified Professional (TCP)	17.11. – 19.11.2015	Wien	Tricentis
Tricentis Certified Professional (TCP)	15.12. – 17.12.2015	Wien	Tricentis
Quality Brunch V2.0	24.11.2015	Wien	SQS
360° Testautomatisierung	27.01. – 28.01.2016	Wien	ANECON
	12.04. – 13.04.2016	Wien	
CMAP© Mobile App Testing – Foundation Level	06.04. – 07.04.2016	Wien	ANECON
Model Based Testing Hands On	01.12. – 02.12.2015	Linz, Wien, Graz	Software Quality Lab
	10.12. – 11.12.2015	Lustenau	
	14.03. – 15.03.2016	Linz, Wien	
Testen für Softwareentwickler mit Unit-Tests	26.01. – 28.01.2016	Linz, Wien	Software Quality Lab
	12.04. – 14.04.2016	Linz, Wien	
GUI-Testautomatisierung in Theorie und Praxis	02.12. – 03.12.2015	Linz, Wien, Graz	Software Quality Lab
	15.12. – 16.12.2015	Lustenau	
	22.02. – 23.02.2016	Linz, Wien	
Risikomanagement in Softwareprojekten	14.01.2016	Linz, Wien	Software Quality Lab
	07.04.2016	Linz, Wien	
ISAOB® Certified Professional for Software Architecture , Foundation Level	14.12. – 17.12.2015	Linz, Wien, Graz, Lustenau	Software Quality Lab
	08.02. – 11.02.2016	Linz, Wien	
UML-Basics für Fachbereichs-mitarbeiter	09.11. – 10.11.2015	Linz, Wien, Graz	Software Quality Lab
	24.11. – 25.11.2015	Lustenau	
Aufwandsschätzung in Softwareprojekten	26.04. – 27.04.2016	Linz, Wien	Software Quality Lab
Professionelles Requirements Engineering und Management	09.12. – 11.12.2015	Linz, Wien, Graz	Software Quality Lab
	15.12. – 17.12.2015	Lustenau	
	12.04. – 14.04.2016	Linz, Wien	
Moderationstechniken im Requirements Engineering	12.01. – 13.01.2016	Linz, Wien	Software Quality Lab
Software Usability	15.03. – 17.03.2016	Linz, Wien	Software Quality Lab
Scrum Master Professional	09.12. – 10.12.2015	Linz, Wien, Graz	Software Quality Lab
	16.12. – 17.12.2015	Lustenau	
	08.03. – 09.03.2016	Linz, Wien	
Certified Agile Business Analysis	21.03. – 22.03.2016	Mödling	SEQIS Software Testing GmbH
SEQIS „10 things“-Expertentreff zum Thema „Continuous Integration: Mehr als nur ein Schlagwort“	19.11.2015	Wien	SEQIS Software Testing GmbH

Konferenz	Termin	Ort	Deadline Call for Paper
Agile Testing Days	09.11. - 12.11.2015	Potsdam / Deutschland	
HUSTEF 2015	18.11. - 19.11.2015	Budapest / Ungarn	
Software Quality Days 2016	18.01. - 21.01.2016	Wien	
Ignite 2016	26.04. - 28.04.2016	Düsseldorf / Deutschland	

◀ (Richard Messnarz)

cartoontester.blogspot.com © 2013

[Auf zu neuen Ufern!]

von Torsten Zimmermann

Microsoft wagt mit Visual Studio Code im Bereich der Softwareentwicklung den Sprung auf andere OS Plattformen jenseits der Windows-Welt.

Auf der Build-Konferenz wurde eine neue Visual Studio Edition unter dem Namen Visual Studio Code (VS Code) vorgestellt. Dieser Artikel erläutert den aktuellen Funktionsumfang dieser brandneuen Visual Studio Edition sowie die dahinterliegende Microsoft-Strategie. Eine aktuelle Bewertung bezüglich der Nutzbarkeit dieses Entwicklungswerkzeugs in der Praxis rundet den Bericht mit einem Fazit ab.

Bereits mit unseren Berichten zu den Visual Studio und Team Foundation Server Softwareversionen zeigten wir auf, dass Microsoft seine Business Strategie ändert. Die Vorstellung von Visual Code wirkt in diesem Sinne wie ein weiterer Paukenschlag. Weniger wegen der gezeigten Features des neuen Editors, sondern begründet in der Tatsache, dass Microsoft seine Softwareentwicklung nicht mehr auf die eigenen Betriebssystemplattformen beschränkt.

Zum einen erkennt damit Microsoft die Technologieplattformen außerhalb der Windows-Welt stärker an als dies in der Vergangenheit der Fall war.

Zum anderen forciert Microsoft den Wettbewerb der verschiedenen Softwarelösungen. In der Dekade 2000-2010 hatten wir in diesem Sinne ein statisches Bild. Die Schwerpunkte der Produktportfolios der einzelnen globalen Softwareanbieter waren klar definiert und wurden nur im geringen Maß angepasst. Seit Mitte der zweiten Dekade des neuen Jahrtausends kommt hier deutlich mehr Bewegung ins Spiel: Die Karten werden neu gemischt!

Wer sich mit Cloud-Entwicklungsumgebungen beschäftigt hat, dem werden Projektnamen wie „Napa“ oder „Monaco“ geläufig sein. „Napa“ ist eine komplette Entwicklungsumgebung für neue Office und Windows Apps, welche in der Cloud und im Browser innerhalb weniger Minuten eingerichtet werden kann. Das Einrichten einer SharePoint-Entwicklungsumgebung auf einem lokalen Rechner kann beispielsweise

durchaus einen ganzen Tag in Anspruch nehmen und mehr Hardwareressourcen erfordern, als viele Systeme zur Verfügung stellen. „Monaco“ ist die gleiche Entwicklungsumgebung, die über Microsoft Azure für Web und Cloud Anwendungen angeboten wird. Der nächste logische Schritt bestand darin, die Entwicklungsumgebung auch lokal anzubieten und das genau ist Visual Studio Code. Technisch basiert es auf modernen Webtechnologien wie HTML5 und JavaScript / TypeScript, verwendet einen Node Server und kann dadurch auf einer Vielzahl von Plattformen betrieben werden.

Erster Eindruck von Visual Studio Code

Die Setup-Dateien können direkt von der Visual Studio Online Website (siehe auch Infokasten) heruntergeladen werden. Die Installation geht rasch von Hand. Die Programmdateien werden unter dem aktuellen Benutzer in AppData/Local/Cod/app-0.1.0 abgelegt. Als Ergebnis wird der Editor am Ende der Installation mit ersten Erläuterungen zur Bedienung aufgerufen.

Der Funktionsumfang von Visual Studio Code konzentriert sich auf das Editieren, Versionieren und Debuggen. Im Gegensatz zum klassischen

Visual Studio gibt es keine Projektdateien. Der Editor kennt lediglich Dateien und Ordner. Visual Studio Solution- und Projektdateien können aber gelesen und beispielsweise für IntelliSense-Funktionen genutzt werden.

- siehe Bild 1.

Die Oberfläche ist sehr übersichtlich gestaltet. Diese lässt sich in folgende Bereiche unterteilen:

- **Editor:** Im Editor können bis zu drei Fenster nebeneinander dargestellt werden.
- **Sidebar:** Die Sidebar beinhaltet verschiedene Sichten wie beispielsweise die Explorer-Darstellung der Dateien.
- **Statusbar:** Die Statusbar liefert Informationen über die geöffneten Dateien.
- **Viewbar:** Die Viewbar dient zum Umschalten der verschiedenen Sichten.

Auf der Sidebar befindet sich neben der Suche auch der Explore-Icon. Hierüber lässt sich ein Root-Verzeichnis festlegen, so dass alle darunterliegenden Verzeichnisse und Dateien auf der linken Seite neben dem eigentlichen Bearbeitungsfenster für die Dateiinhalte angezeigt werden. >

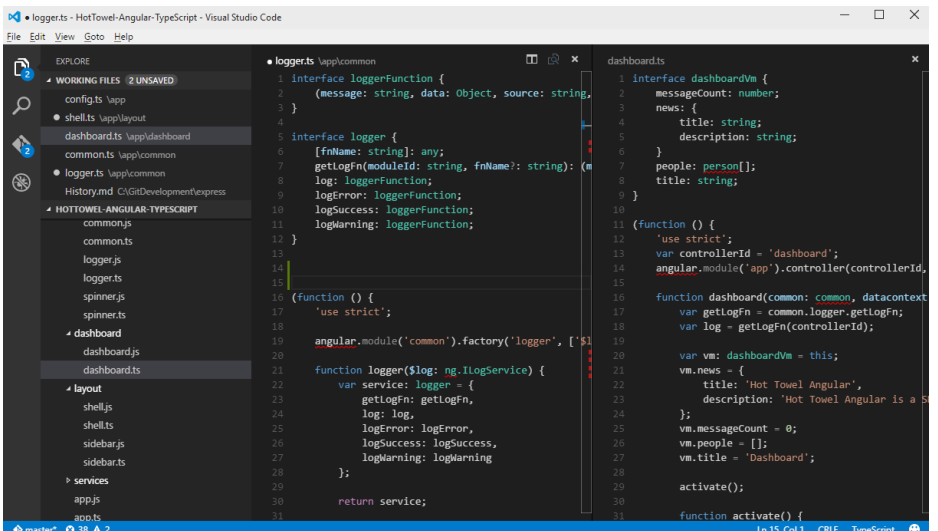


Bild 1: Oberfläche von Visual Code

Visual Studio Code

Durch einfaches Klicken auf eine Datei wird der entsprechende Inhalt im Editor angezeigt. Führt man danach einen Ctrl+Klick auf eine andere Datei aus, so wird der Inhalt rechts neben der zuvor geladenen Datei angezeigt. So lassen sich Dateiinhalte rasch und bequem vergleichen. Geöffnete Daten führt VS Code auch unter einem Working Files Verzeichnis. So behält man auch bei vielen gleichzeitig geöffneten Dateien den Überblick.

Der Zustand von Visual Studio Code wird beim Schließen gespeichert, so dass beim nächsten Start die gleichen geöffneten Dateien wieder vorliegen. Die „Auto Save“ Funktion, welche per Option im File Menü aktiviert werden kann, unterstützt eine lückenlose Sicherung aller Codeänderungen. Bei Suchvorgängen im Quellcode wird eine Vorschau mit der Anzahl der Treffer angeboten. In den Suchoptionen kann der Datenbereich in Form von Ordnern oder Dateien festgelegt werden, in welchem gesucht werden soll.

Oft ist die Liste der geöffneten Dateien lang. Deshalb ist es notwendig, schnell zwischen diesen navigieren zu können. Über „Ctrl+Tab“ kann eine Liste der geöffneten Dateien mit kompletter Pfadangabe in einem separaten Fenster übersichtlich dargestellt und die gewünschte Datei zur weiteren Bearbeitung ausgewählt werden.

Das Aussehen des Editors lässt sich im geringen Umfang anpassen. Microsoft hat hierzu drei Themes dem Editor zur Auswahl beigelegt.

Unterstützte Sprachen

Entwickler müssen heute in der Lage sein, rasch korrekten Programmcode zu realisieren. Ohne geeignete Tools ist das jedoch sehr schwierig. Offen gesagt bietet hier Visual Studio Code noch wenig. Einige interessante Werkzeuge existieren aber bereits und sollten unbedingt erwähnt werden.

Funktion	Sprache
Syntax coloring bzw. Highlighting und bracket matching	Batch, C++, Clojure, Coffee Script, DockerFile, F#, Go, Jade, Java, HandleBars, Ini, Lua, Makefile, Markdown, Objective-C, Perl, PHP, PowerShell, Python, R, Razor, Ruby, SQL, Visual Basic, XML
zusätzlich IntelliSense und linking	CSS, HTML, JavaScript / TypeScript, JSON, Less, Sass
zusätzlich Refactoring und Ref-	C#, TypeScript

Tabelle 1: Unterstützte Sprachen und Features

Tabelle 1 gibt einen Überblick, welche Features bei welcher Sprache von Visual Studio Code dem Entwickler zur Verfügung gestellt werden.

In der Vergangenheit wurde das IntelliSense Feature beim klassischen Visual Studio am häufigsten von der Entwickler Community nachgefragt. Aus diesem Grunde entschloss sich Microsoft dazu, schon in der ersten Version von Visual Studio Code diese wichtige Funktion anzubieten. Für JavaScript, JSON, HTML, CSS, Less, Sass, C# und TypeScript kann der Entwickler bei VS Code auf das IntelliSense Feature zurückgreifen. Bei JavaScript, TypeScript und C# werden zusätzlich die verschiedenen Parameterleisten für überladene Funktionen angezeigt. Mittels Cursortasten wird ein einfaches Navigieren zwischen den besagten Parameterleisten möglich. Ferner werden Code-Snippets in Form von „Emmet Abbreviations“ in HTML, Razor, CSS, Less, Sass, XML und Jade unterstützt.

Kommandos können über „Ctrl+Shift+P“ eingegeben und über „?“ die Liste der verfügbaren Kommandos angezeigt werden. Mittels „Ctrl+F12“ werden Sie zur Definition eines Symbols geführt. Mittels „Ctrl“ oder einfach mit dem Mauszeiger darüberfahren, können Sie sich im aktuellen Editorfenster die Definition anzeigen lassen. Sie können auch gezielt zu Symbolen über „Ctrl+Shift+O“ suchen.

Die Suche nach Referenzen kann schnell unübersichtlich werden. Deshalb werden die Ergebnisse der Referenzsuche oder der Definitionsanzeige im Editor dargestellt. Im betreffenden Inline-Fenster können gar Änderungen vorgenommen werden. Eine wichtige Zusatzinformation im Editor ist die Anzahl der Referenzen, um auf diese Weise das Risikopotential einer Änderung abzuschätzen oder auch den Aufbau einer Anwendung besser zu verstehen. Somit steht ein Teil der CodeLense-Funktionalität für C# bereits heute im Visual Studio Code zur Verfügung.

>

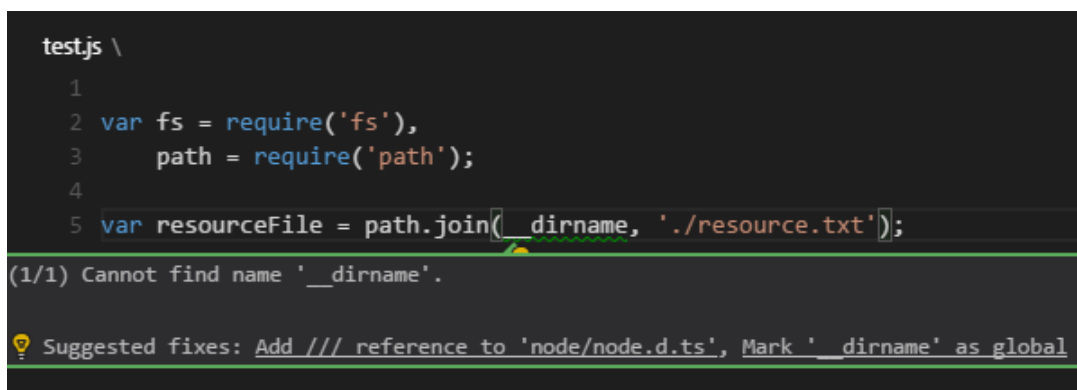


Bild 2: Hinweise zur Codeoptimierung

Visual Studio Code

Das Umbenennen von Bezeichnern gehört auch zur täglichen Entwicklerarbeit und wird von VS Code für TypeScript und C# in allen relevanten Projektdateien unterstützt. Dank der Unterstützung des Compiler-Frameworks „Roslyn“ werden Hinweise während der Eingabe in Form von kleinen Glühbirnen dargestellt. Nach dem Klicken auf das Glühbirnen-Symbol werden weitergehende Erläuterungen zum Fehler gegeben und mögliche Behebungsstrategien vorgeschlagen.

- siehe Bild 2

Um die Codequalität zu erhöhen, wird im Hintergrund der Code permanent analysiert und entsprechende

Warnungen werden in der Statuszeile und in einem separaten Fenster angezeigt.

Tastaturbefehle selbst festlegen

Bei aller Usability über grafische Oberflächen sind vor allen Keyboard Shortcuts die zweckmäßigsten Werkzeuge für eine schnelle Codebearbeitung. Neben vielen anderen interessanten Informationen sind diese in der Visual Code Onlinedokumentation erläutert.

Es können auch eigene Tastaturbefehle definiert werden. Der nachfolgende Code zeigt hierzu einige Beispiele:

Auf jeden Fall müssen in der Definition „key“ und „command“ festgelegt sein. Optional ist das „when“ Kommando, welches definiert, ob ein Tastaturkommando nur in bestimmten Fällen gelten soll. Hinter „command“ ist eine entsprechende Methode anzugeben, welche mit dem Tastaturkommando ausgeführt werden soll.

Git Versionsmanagement

Git erfreut sich als verteiltes Versionsmanagementsystem immer größerer Beliebtheit. Microsoft traf die generelle Entscheidung Git mit seinen Entwicklungswerkzeugen zu unterstützen. So können inzwischen Visual Studio Online, das neue Visual Studio Code und auch die lokale Variante des Team Foundation Servers mit Git Daten austauschen. Natürlich besteht die Möglichkeit auch jedes weitere Git Repository einzubinden.

- siehe Bild 3

Auch Visual Studio Code kann auf Git Repositories zugreifen. Eventuelle Merge-Konflikte werden dabei direkt im Editor angezeigt. In Kombination mit einem Git Repository werden die Zeilen in einer entsprechenden Farbe angezeigt, welche gelöscht (rot), hinzugefügt (grün) oder geändert (blau) wurden.

>

```
// Keybindings that are active when the focus is in the editor
{ "key": "home",      "when": "editorTextFocus", "command": "cursorHome" },
{ "key": "shift+home", "when": "editorTextFocus", "command": "cursorHomeSelect" },

// Keybindings that are complementary
{ "key": "f5",        "when": "inDebugMode",    "command": "workbench.action.debug.play" },
{ "key": "f5",        "when": "!inDebugMode",    "command": "workbench.action.debug.start" },

// Global keybindings
{ "key": "ctrl+f",      "command": "actions.find" },
{ "key": "alt+left",    "command": "workbench.action.navigateBack" },
{ "key": "alt+right",   "command": "workbench.action.navigateForward" },

// Global keybindings using chords
{ "key": "ctrl+u l",    "command": "omnisharp.show.generallog" },
{ "key": "ctrl+u ctrl+l", "command": "omnisharp.show.generallogOnSide" },
```

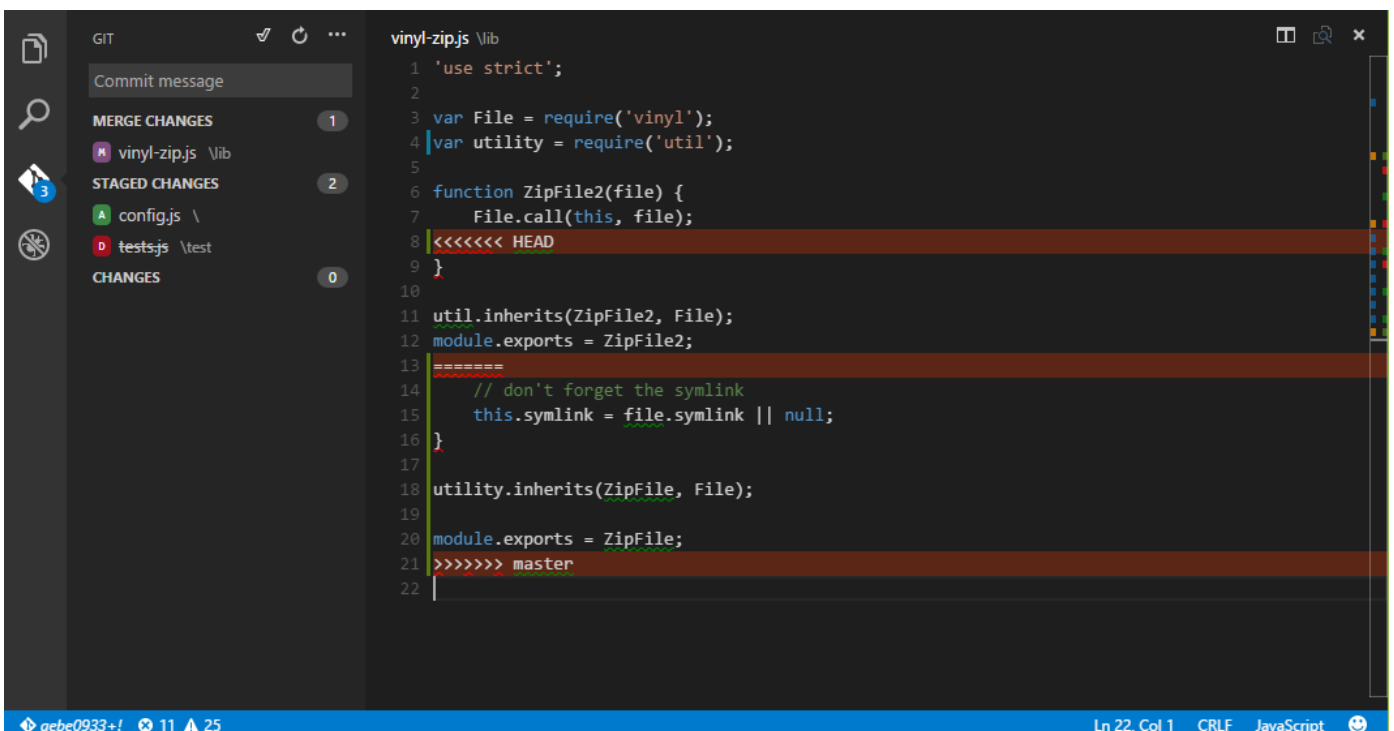


Bild 3: Darstellung von Merge-Konflikten

Visual Studio Code

Debugger

Interessant ist das Debugger Feature. Es hat mich überrascht in einem kompakten Editor wie den Visual Studio Code einen integrierten Debugger vorzufinden.

- siehe Bild 4

Das Debuggen ist jedoch nur ansatzweise mit der aktuellen VS Code Version möglich. Lediglich bei Node.js und ASP.NET 5 kann der Editor Debugging Funktionalitäten anbieten. Die Unterstützung weiterer Sprachen ist jedoch für spätere Softwareversionen des Editors geplant.

Um die Anwendungen mit VS Code zu debuggen, ist vorab die launch configuration Datei launch.json zu erstellen. Durch das Anklicken auf die launch oder configure Aktionen erzeugt Visual Studio Code eine entsprechende launch.json. Danach kann der Debugger zur Untersuchung der betreffenden Anwendung genutzt werden.

Microsoft im Umbruch

Microsoft entwickelt sich rasant zum Anbieter für unterschiedliche Plattformen. In der Vergangenheit war ganz klar Windows die ausschließliche Zielplattform. Andere Plattformen wurden ausschließlich von Partnern abgedeckt.

Im Zuge der Neuausrichtung als „mobile first and cloud first company“ ergänzt um das strategische Unternehmensziel „productivity and platform“ war es notwendig, auch andere Betriebssysteme zu unterstützen. Daher erfuhrt die Visual Studio Produktfamilie eine Ergänzung durch Visual Studio Code, um auch Linux und OS X Nutzern eine integrierte Entwicklungsumgebung zur Verfügung zu stellen.

Fazit

Aktuell ist der Funktionsumfang der „klassischen“ Visual Studio Editionen für Windows wesentlich größer und es ist auch kein Wechsel auf Visual Studio Code empfehlenswert. Es ist eher als eine Ergänzung zum klassischen Visual Studio für weitere Betriebssysteme und Entwicklungen im Browser gedacht. Es ist ein schöner, schlanker Editor. Er ist rasch zu laden und sofort einsatzbereit. Schnelles Ausprobieren und Entwickeln auf einem Mac beispielsweise und ein nahtloser Übergang zu Visual Studio Enterprise oder Professional sind mögliche Nutzungsszenarien. Sicherlich wird Visual Studio Code weiterentwickelt werden. Denn es fehlen noch wichtige Features. Nicht nur im Bereich des angesprochenen Debuggers, auch bei Standardfunktionen wie das Suchen und Ersetzen wären noch weitere Zusatzfeatures vorstellbar.

Gemäß der Roadmap werden zukünftige Versionen von Visual Studio Code unter anderem auch Plug-Ins unterstützen und die Debugging Möglichkeiten werden ausgebaut. Diesbezüglich bleibt zu hoffen, dass der Editor seine Vorzüge als leichtgewichtiges Tool im Rahmen der Weiterentwicklung nicht einbüßt. Denn das Arbeiten macht wirklich Spaß. Da verzeiht man gerne das Fehlen des einen oder anderen Funktionsgimmicks. Doch hinter der Nachricht über ein neues Tool innerhalb der Visual Studio Produktfamilie liegen die entscheidenden Schlussfolgerungen woanders.

Wir hatten bereits in einigen unserer früheren Artikel entsprechende Fragen formuliert. Nun besteht jedoch Gewissheit: Microsoft vollführt einen radikalen Umbau seines Unternehmens. Es erfährt eine komplette Neuausrichtung. Mit ihr wird sich auch Microsofts Produkt Portfolio grundlegend ändern. Und das ist gut so! Kontinuität ist in der Softwarebranche sicherlich zu befürworten. Doch hierin steckt auch ein gewisses Risiko. Schnell wird aus Kontinuität Stagnation: ein Alptraum für ein Technologiesegment. Denn die Softwarebranche konkurriert mit anderen Technologiesparten um die Gunst – um die Aufmerksamkeit des globalen Marktes.

>

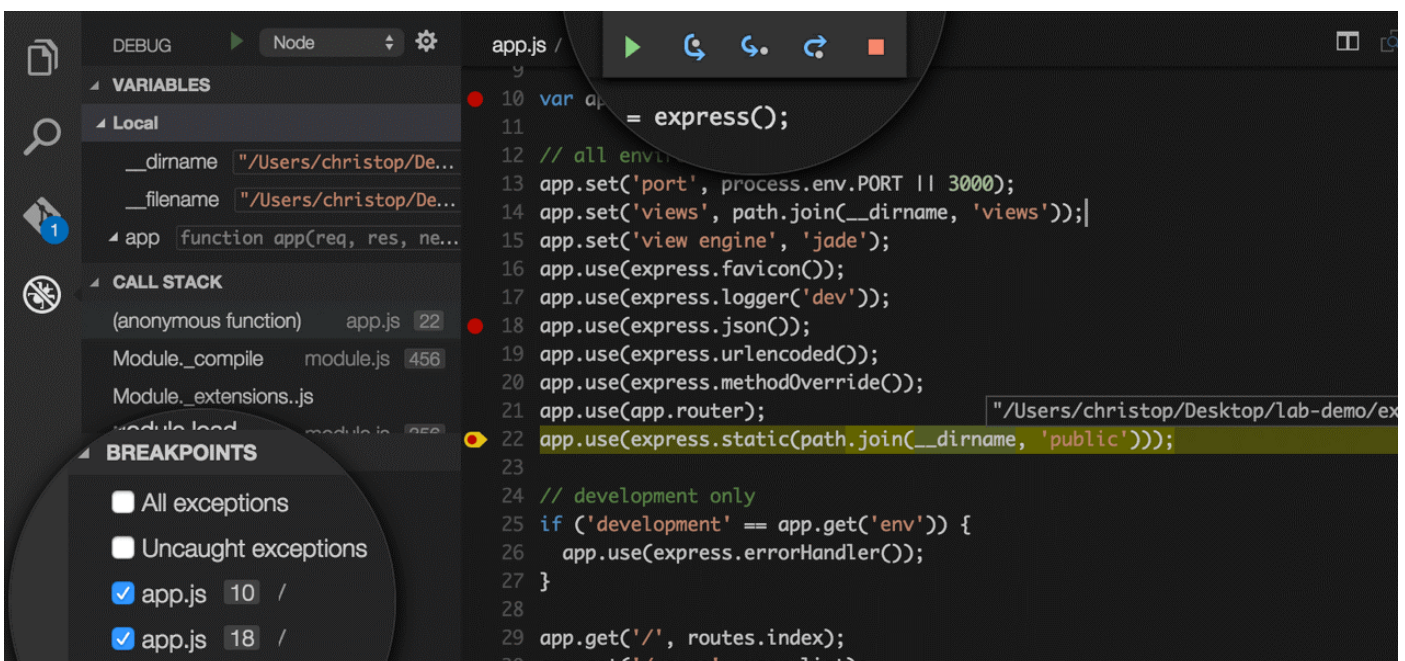


Bild 4: Debugger

Visual Studio Code

Was kann da schlimmer als Gleichgültigkeit sein? So gesehen kommt die eigene Neuerfindung Microsofts zum richtigen Zeitpunkt. Nicht nur für Microsoft selbst, sondern für uns alle! Sie wird unsere Auffassung über Software und deren Bedeutung in allen Bereichen des Lebens verändern. Damit werden neue Gedankengänge, Ansätze, Verfahren und Lösungen möglich, welche wir heute noch nicht erkennen.

In diesem Sinne: Danke, Microsoft!

◀ (Torsten Zimmermann)

Demos zu verschiedenen Microsoft Entwicklungsprojekten

Ein weiterer Unterschied in der Microsoft Business Strategie liegt in der neuen Offenheit gegenüber seinen Kunden im Softwareentwicklungsumfeld. Zu Visual Studio gibt es eine Reihe von Demo-Umgebungen auf denen Softwareentwickler der ganzen Welt quasi die Produktweiterentwicklungen online verfolgen können. Nachfolgend sind hierzu einige Links zu interessanten Entwicklungsversionen zum Ausprobieren aufgeführt:

1. „Napa“ ist die Entwicklungsumgebung für neue Office und Windows Apps und kann unter <http://www.dev.office.com> getestet werden.
2. Auf Visual Studio Online (<http://visualstudio.com>) kann der aktuelle Visual Studio Release Kandidat, der Team Foundation Server CPT sowie die aktuelle Visual Studio Code Version heruntergeladen werden.

TypeScript

TypeScript ist eigentlich keine vollständige Programmiersprache, sondern nur eine Art Zwischensprache. Aus ihr wird immer JavaScript erzeugt, dessen Code dann im Browser ausgeführt werden kann.

Online Dokumentation

Für Visual Studio Code gibt es eine Online Dokumentation unter <http://code.visualstudio.com/docs>, welche die wichtigsten Bedienungselemente des Editors erläutert. Diese Dokumentation kann auch über den Hilfeintrag im Visual Studio Code Menü erreicht werden.

[Über Torsten Zimmermann]

Nach seinen vollendeten Studium als Diplom Wirtschaftsinformatiker begann Torsten Zimmermann 1993 mit seiner beruflichen Karriere. Seit 1995 beschäftigt er sich im Rahmen international angelegter Projekte mit den Themen Software-Qualität und Test-Management bei verschiedenen Unternehmen wie (u.a.) BMW, Daimler, Hewlett-Packard, Hoffmann-La Roche und Logica.

Im Rahmen seiner Arbeiten entwickelte er (u.a.) den risiko-basierten Testansatz, welcher im Fachmagazin "QZ" vorgestellt wurde und heute sich als Basiswissen in der Software-Qualitätssicherung etabliert hat.

Weitere Ergebnisse und Erkenntnisse aus der Qualitäts-Management-Praxis führten zu dem T1 TFT (Test Framework Technologies, 2001), welche der Beginn einer neuen Generation von Testsystemen begründete.

Heute entwickelt Torsten Zimmermann neue Ansätze für leistungsfähigere Testkonzepte und -Frameworks, wie dem T2 TFT (2004) und dem T3 TFT (2006). In Kooperation mit einem Netzwerk aus Hochschulen und Universitäten entstehen hierbei neuartige Lösungen im Bereich regel- und modellbasierter Testsysteme. Als Referent auf Kongressen und Fachautor präsentiert er regelmäßig seine Erfahrungen, Ergebnisse und Konzepte in zahlreichen Vorträgen und Fachartikeln auf nationaler wie auch internationaler Ebene.

Weitere Informationen unter:

http://www.xing.com/profile/Torsten_Zimmermann2



[Zum Abschluss: seit wann produziert das ATB „Sauna boards“?]

Manchmal landen e-Mails einfach falsch, trotzdem ist es mir ein Rätsel, wie jemand auf die Idee kommen kann, dass wir ein Sauna board liefern können...

Diese Mail erreichte uns vor kurzem:

Dear Sir/Madam,

We are an imports and exports company located in Xi'an Shaanxi, China. Due to business needs, we are looking for Sauna board.

The specific requirements as follows:

1. Name: Sauna board

Material: red cedar, pinus sylvestris or Douglas fir.

2. Specification:

a.thickness: 15mm.

width: 120mm.

length: 2750mm.

quantity: 80000pcs

b. thickness: 10mm.

width: 100mm.

length: 2750mm.

quantity: 80000pcs

3. Fine workmanship. Planing process. At most one scab within 1000mm. Drying procedure.

4. Delivery time: 6 months.

We know that your factory is expertised in wood working from the database with actual strength. If you are interested in this order, please send me a FOB or EXW quotation in US dollar or Euro. If there is a equitable price, we will put a formal order.

Best Regards,

Robert Wang

Keine Angst, wir wechseln nicht unser Betätigungsfeld, sondern bleiben weiter der kompetente Partner zu allen Fragen bezüglich Softwaretest und ISTQB Zertifizierungen. Und Anfragen an uns sind jederzeit gern willkommen.

◀ (Karl Kemminger)