



Analysis. Development. Test. Management

Ausgabe H2/2020

The Next Big Thing in Software Engineering and Quality

Künstliche
Intelligenz &
Machine Learning

Seite 18

razzfazz.io
Unsere neue Marke für
individuelle Softwareent-
wicklung

Seite 26

Holacracy -
Scrum bis in die
Führungsetage

Seite 30

Titel: „Stadtgemeinde“, Robert Schwarz, Fineliner / Acryl Collage

Analysis. Development. Test. Management. Better IT Results.

Lesen Sie in dieser Ausgabe:

Editorial.....	3	SEQIS Kalender.....	20
Alle Termine auf einen Blick			
Neulich im Netz.....	4	Wissensmanagement und die Demokratisierung des Wissens.....	
Softwaretest - Ergebnisse der aktuellen Umfrage		22	
Autonomes Fahren.....	6	Ten more things.....	25
The next big thing in Software Engineering and Quality?		Online. Proven. Professional.	
OKRs.....	8	razzfazz.io.....	26
Objectives and Key Results			
Kurzfristige Projektunterstützung.....	11	Moco und SEQITracker.....	28
SEQIS Experten!			
...wie RPA bots das Leben des Fachbereichs erleichtern.....	12	Holacracy.....	30
		Scrum bis in die Führungsetage	
Das Cynefin - Framework.....	16	Agile Circle.....	35
Komplex oder kompliziert?		What's next?	
Künstliche Intelligenz & Machine Learning.....	18	Der ROI von Usability.....	36
		KI & Software Security.....	38

Ihre Meinung ist gefragt!

Nach den QualityNews ist bekanntlich vor den QualityNews! Schon bald arbeiten wir wieder auf Hochtouren an der nächsten, spannenden Ausgabe. Lesen Sie nur das, was Sie wirklich interessiert! Sagen Sie uns, welche Themen Sie spannend finden.

Kontaktieren Sie uns: marketing@SEQIS.com
Join us: [@SwTestIsCool](https://twitter.com/SwTestIsCool)

Wir freuen uns auf Ihre Vorschläge und Wünsche!

Über SEQIS QualityNews:

Dieses Magazin richtet sich an Gleichgesinnte aus den Bereichen IT Analyse, Development, Softwaretest und Projektmanagement im IT Umfeld. Die SEQIS Experten berichten über ihre Erfahrungen zu aktuellen Themen in der Branche. Die Leser des Magazins gestalten die Ausgaben mit: Schreiben Sie uns Ihre Meinung im SEQIS Blog (www.SEQIS.com/de/blog-index) oder als Leserbrief.

Wenn Sie dieses Magazin abbestellen möchten, senden Sie bitte ein Mail an marketing@SEQIS.com.

Impressum:
Information und Offenlegung gem. §5 E-Commerce-Gesetz und §25 Mediengesetz

Herausgeber: SEQIS GmbH,
Neusiedler Straße 36, A-2340 Mödling
Tel: +43 2236 320 320 0
Fax: +43 2236 320 320 350
info@SEQIS.com, www.SEQIS.com
Gericht: Bezirksgericht Mödling
Firmenbuchnummer: 204918a
Umsatzsteuer-ID: ATU51140607
Geschäftsführung: Mag. (FH) Alexander Vukovic, Mag. (FH) Alexander Weichselberger, DI Reinhard Salomon

Druck: druck.at Druck- und Handelsgesellschaft mbH, 2544 Leobersdorf
Erscheinungsweise: 2x pro Jahr
Für die verwendeten Bilder und Grafiken liegen die Rechte für die Nutzung und Veröffentlichung in dieser Ausgabe vor.
Die veröffentlichten Beiträge, Bilder und Grafiken sind urheberrechtlich geschützt.
(Kunstwerke: Lebenshilfe Baden und Mödling, Fotos: Shutterstock).

Sämtliche in diesem Magazin zur Verfügung gestellten Informationen und Erklärungen geben die Meinung des jeweiligen Autors wieder und sind unverbindlich. Irrtümer oder Druckfehler sind vorbehalten. Hinweis im Sinne des Gleichbehandlungsgesetzes: Aus Gründen der leichteren Lesbarkeit wird die geschlechtsspezifische Differenzierung nicht durchgehend berücksichtigt. Entsprechende Begriffe gelten im Sinne der Gleichbehandlung für beide Geschlechter.



DI Reinhard Salomon

Mag. (FH) Alexander Vukovic

Mag. (FH) Alexander Weichselberger

QualityNews

Sehr geehrte Leserin,
sehr geehrter Leser,

wir freuen uns, Ihnen die zweite Ausgabe der SEQIS QualityNews im Jahr 2020 zu präsentieren.

Vielen Dank für das positive Feedback zu unseren letzten Ausgaben zu den Themenbereichen „Resilienz und Legacy Code“. Wir hoffen, wir konnten Ihnen damit interessanten Content zur Verfügung stellen und Sie stellenweise auch gut unterhalten. Über weitere Anregungen, Themenwünsche und Feedback Ihrerseits freuen wir uns.

Auch in dieser Ausgabe finden Sie neben den branchenbezogenen Artikeln auch nicht-technische Bereiche :

Im Heft finden Sie einige Kunstwerke der Lebenshilfe Niederösterreich der Werkstätten Baden und Mödling. Denn nicht nur unsere Spezialisten, sondern auch die Klienten der Lebenshilfe leben für ihre(n) Beruf(ung).

In dieser Ausgabe dreht sich alles um das Thema „The Next Big Thing in Software Engineering and Quality“. Was sind die großen Themen der Digitalisierung? Was ist relevant bzw. wird es in Kürze sein?

Auf den folgenden Seiten geben Ihnen unsere Experten einen Einblick in die vielseitigen Aspekte zum Thema.

Wir wünschen viel Lesevergnügen mit unseren SEQIS QualityNews!

Ihre SEQIS Geschäftsleitung

Neulich im Netz: Softwaretest - Ergebnisse der aktuellen Umfrage

von Hansjörg Münster

Zum dritten Mal nach 2011 und 2015/16 führte heuer im Frühjahr das German Testing Board, das Austrian Testing Board sowie die Technische Hochschule Köln und die Hochschule Bremerhaven die Umfrage „Softwaretest in Praxis und Forschung“ durch. Nun liegen die ersten Ergebnisse vor. Diese wurden unter „www.softwaretest-umfrage.de“ veröffentlicht. Dort findet man auch eine Management-Summary. Weitere Publikationen und Interpretationen der Ergebnisse sollen noch veröffentlicht werden. Eine Zusammenfassung findet man auch bei Heise-Online: „www.heise.de/news>Status-Softwaretest-Agile-Vorgehensweisen-establieren-sich-zunehmend-4935676.html“

Ein klares Ergebnis ist, dass die agile Transformation weiter fortschreitet. Nur mehr knapp mehr als 20% geben an, klassische, phasenorientierte Modelle zu verwenden. Dennoch glauben die Herausgeber der Studie, dass die Agilität oft nur halbherzig umgesetzt ist. Merkmale wie Teamverantwortung, flache Hierarchien und Einbeziehung der Enduser sind oft nicht vollständig umgesetzt. Auch dominieren noch die klassischen Rollen wie „Entwickler“, „Testmanager“ und Tester, wohingegen agile Rollen wie „Scrum Master“, „Product Owner“ deutlich in der Minderheit sind.

Mit der Verbreitung von agilen Methoden nimmt auch die Verwendung klassischer Testmethoden und Testverfahren wie „Äquivalenzklassenanalyse“ und „Entscheidungstabellen“ ab. Fast dreiviertel geben an, dass die Bereitstellung von Testdaten eine Herausforderung ist.

Bei der Frage nach der Einhaltung von Teststandards gab fast die Hälfte

der ca. 1.200 Befragten an, einem eigenen Standard zu folgen. Nur mehr 25% halten sich an den ISTQB-Prozess und der ISO-29119 Software Testing Prozess wird von nicht einmal 10% der Befragten als Grundlage für ihre Arbeit verwendet. Damit kann keiner der klassischen Standards als „flächendeckend“ und erfolgreich im Alltag angesehen werden.

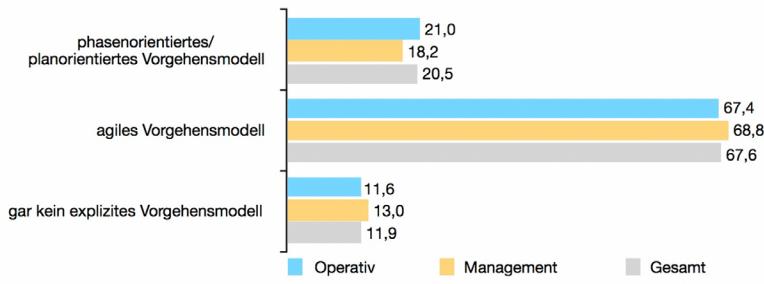
Eine weitere, bedenkliche Erkenntnis aus der Umfrage ist, das fehlende Wissen und fehlende Ausbildung im Testing: Circa ein Fünftel konnten mit fundamentalen Begriffen wie z.B. „Anweisungstest“ oder „Entscheidungstest“ nichts anfangen. Zum Glück schätzen ca. 80% der Befragten für die Weiterbildung „Testen in agilen Kontexten“ und fast 60% für den Bereich „spezialisiertes Testen“ einen hohen Bedarf in den nächsten 5 Jahren.

Eine Reihe weiterer interessanter Erkenntnisse finden Sie in der Management-Summary unter „https://www.softwaretest-umfrage.de/pdf/Ergebnisse_Softwaretestumfrage_2020_ManagementAbstract.pdf“, die die Herausgeber der Umfrage kostenlos zur Verfügung stellen. ■

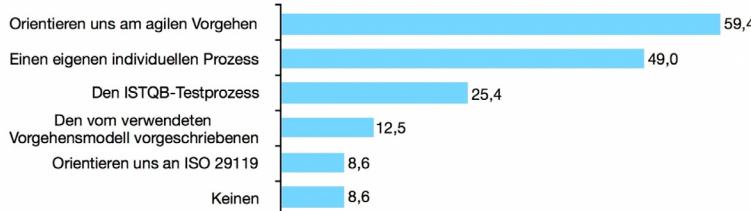


Christian Reithofer, Katharina Stockreiter, Technik: Acryl auf Leinwand

Wie ordnen Sie Ihr Vorgehensmodell in der Softwareentwicklung ein? Eher als ... (in %)



Welchen Prozess verwenden Sie für die Durchführung der Testaktivitäten?
(in %, Mehrfachauswahl)



Unsere Leistungen

Software-Qualitätssicherung
aus erster Hand:
Von A wie Analyse bis Z wie
Zertifikat



Quellen und weiterführende Informationen:

Die Softwaretest-Umfrage 2020 wurde erstellt in Kooperation mit dem German Testing Boards (GTB), der Technischen Hochschule Köln, der Hochschule Bremerhaven und dem Austrian Testing Board (ATB). Alle Rechte liegen bei den Kooperationspartnern.



Hansjörg Münster ist Principal Consultant und Teamlead bei SEQIS.

Als Allrounder deckt er ein breites Spektrum an Aufgaben ab. Die Schwerpunkte seiner Tätigkeit liegen in den Bereichen Test Management, Testautomation und Lasttest.

Ganz oben auf der Prioritätenliste des IT Profis steht, einen Nutzen und Mehrwert in der Qualitätssicherung seiner IT Projekte zu generieren.

Autonomes Fahren - the next big thing in Software Engineering and Quality?

von Alexander Vukovic



<https://www.suraaa.at/presse/>

Die Zukunft ist rosig, wir müssen nicht mehr selbst Fahrzeuge lenken und durch die Welt pilotieren, sondern der Wagen macht das für uns. Einsteigen, schlafen, arbeiten, lesen, entspannen oder einfach nur die Landschaft genießen. So verheißungsvoll wird die Zukunft skizziert. Ermöglichen soll dies das autonome Fahren, also die Steuerung des Fahrzeugs durch ein Konglomerat von unzähligen Sensoren, Steuerungssoftware und vor allem von künstlicher Intelligenz.

Aktuell stocken Entwicklung und Verbreitung des autonomen Fahrens allerdings. Das hat mehrere Gründe: Autonomes Fahren mit Level 3 und höher (also ohne menschliche Überwachung) wird sich noch sehr lange nicht massentauglich durchsetzen. Probleme liegen dabei z.B. in ethischen Fragen, wie z.B. der Autopilot entscheiden soll, wenn er zwischen dem Leben der Insassen und dem Leben einer Person auf der Fahrbahn wählen muss.

1. Die rechtlichen Rahmenbedingungen für Level 3 und höher fehlen noch immer
2. Das Problem ist aber natürlich auch die Abbildung der Realität in der Steuerungssoftware bzw. im Falle der Künstlichen Intelligenz

dahinter im Erfahrungsdatenschatz, den die KI durch Training erlangt hat. Bisher fahren in einigen amerikanischen Städten die großen Player wie Waymo, Google oder auch in Europa mehrere Pilotprojekte wie z. B. SURAAA in Pötschach (<https://www.suraaa.at>). Waymo dürfte unterschiedlichen Quellen zufolge mit rund 30 Millionen Testkilometer über die größte Datenbasis verfügen.

3. Auch die Zulassungsbehörden haben das Problem, dass sie nicht wissen, wie sie autonome Funktionen als betriebssicher prüfen und zulassen sollen. So ist Audi z. B. seit 2017 bereit eine Level 3 Steuerung für die Autobahn anzubieten, konnte diese jedoch nicht an die Kunden ausliefern, da in die Behörden nicht wussten, wie sie diese Funktion ausreichen definieren und testen konnten.

Die Hersteller begannen daraufhin, die Behörden zu unterstützen und diese Tests gemeinsam mit den Herstellern zu definieren bzw. „von diesen diese zu übernehmen“. So titelt beispielsweise die Zeitschrift Auto Motor und Sport in einem Artikel über autonomes Fahren in der aktuellen Ausgabe („Jetzt erklären die Her-

steller den Genehmigungsbehörden, worauf sie beim Genehmigen achten müssen“).

Insgesamt also ein schwieriges Umfeld, dass sich im Moment genau auf die Frage des Testens fokussiert und daran scheitert. So gesehen könnte man postulieren, dass das Engineering und Testen einer KI-basierten Steuerung the next possible big thing ist, aktuell ist es aber eher das next impossible thing.

Wo liegen die größten Probleme beim Entwickeln und Testen von Steuerungssoftware, die auf Erfahrungsdaten basiert?

Es ist unmöglich die Realität vollständig in Daten abzubilden. Natürlich kann man die Navigationsdaten um Spurinformationen anreichern und in Echtzeit aktualisieren, so dass beispielsweise die Spurverengung in einer Nebenstraße, entstanden durch einen temporären Baukran, der aus einer Baustelle heraussteht, auch in der Karte für andere Fahrzeuge verfügbar ist. Das würde voraussetzen, dass alle Hersteller diese Informationen sharen und eben in Echtzeit an alle Fahrzeuge verteilen, oder über die Cloud zur Verfügung stellen. Alternativ könnte die Software auch Ähnlichkeiten mit anderen Situationen suchen und dann entscheiden, sich so ähnlich zu verhalten wie in einer ähnlichen Situation und z.B. langsam am Hindernis vorbeizufahren.

Dafür muss die Software „lernen“ mit Situationen umzugehen, d.h. sie wird in diese Situation gebracht und darauf basierend die Entscheidung hinterlegt, wie die Steuerung verfahren soll. Und jetzt kommt die entscheidende Frage: wie kann diese hinterlegte

Situation nun getestet werden? Mit exakt der gleichen Sensorinformation wie beim Lernen der Situation? Mit geänderten Sensorinformationen? Wäre das aber dann nicht schon eine neue Situation, die die Steuerung erst lernen muss? Woher stammt die Entscheidung für den Test, ob die Software lernen oder selbst entscheiden sollte? Wie können diese Testszenarios ohne überbordende Datenmengen an Beispielsensordaten vorzuhalten, beschrieben werden?

Mit herkömmlichen Verfahren ist diese Mammutaufgabe nicht zu bewältigen. Die einzige Chance ist also grundsätzlich lernende und entscheidungsfähige Steuerungen zu entwickeln, die mittels Training on the Job lernen mit möglichst vielen Situationen umzugehen. Ist es der Felssturz in den Alpen, Aquaplaning bei Starkregen, das Kind, das plötzlich hervorläuft, fehlende oder falsche Verkehrszeichen, drängelnde menschliche Autofahrer, eine Reifpanne oder eine Winterauffahrt mit Eisfahrbahn.

Lernen und Training impliziert Fehler. Die Steuerungen müssen also in Fehlersituationen kommen, um daraus zu lernen. Das kann nur in „Produktion“ erfolgen. Die Frage ist, wie schwer dürfen diese Fehler sein und was kostet dies die Benutzer die als Trainer bzw. Versuchskaninchen in einer komplexen Welt dienen. Mein aktueller Wagen bietet Level 2 autonomes Fahren. An gewissen Stellen/Situations erkennt die Steuerung das voraus an der Ampel stehende Fahrzeug nicht und behält die Geschwindigkeit bei. Ein Vertrauensverlust ist die Folge, da passiert sichtlich ein Fehler. Aber aktuell gibt es keinerlei Möglichkeit als interessierter Nutzer Feedback an die Hersteller zu geben. Die Fahrzeuge lernen selbst nicht und die Feedbackschleife zum Hersteller existiert in diesem konkreten Fall nicht.

Waymo hat seine Robotaxis teilweise

unbemannt in Phönix fahren lassen, die Einwohner haben sich durch die Fahrzeuge gefährdet und bedroht gefühlt. Viele der Fahrzeuge wurden attackiert, besprüht und verunstaltet. Die Nachfrage mit diesen Taxis zu fahren ging massiv zurück, obwohl es nur sehr wenig Unfälle gab, weil Phönix ein sehr einheitliches Straßennetz hat. Deshalb ist Waymo wieder dazu übergegangen die Versuchsfahrzeuge mit Überwachungsfahrern auszustatten, die primär die Aufgabe haben, Fragen zu beantworten und das System im Ernstfall zu überschreiben.

Zusammengefasst ist das autonome Fahren eine mit herkömmlichen testmethodischen Ansätzen nahezu unlösbare Problemstellung. Aus dieser werden sich zukünftige neue Spezialisierungen im Softwaretest herausbilden. Spezialisierungen für KI-Testing, automatisierter Szenarioentwicklung und der automatischen Erkennung und Ableitung von Szenarios aus anderen Datenquellen, wie z.B. den Messwerten aus Unfallfahrzeugen. Die Testmethoden für das autonome Fahren entwickeln sich erst, diese Methodenentwicklung ist eines der next big things in Software Engineering and Quality. ■

Weitere spannende Blogartikel finden Sie hier:

[https://www.seqis.com/
de/blog-index](https://www.seqis.com/de/blog-index)



Alexander Vukovic ist SEQIS Gründer und Chief Evangelist.

Er ist erster Ansprechpartner für alle agilen, testmethodischen und testtechnischen Anfragen. In der Praxis arbeitet er als Agile Quality Coach, Berater, Interims-Testmanager, CI-Experte und Lasttester. Mehr als 25 Jahre Beratertätigkeit führten ihn während seiner zahlreichen Projekte in die unterschiedlichsten Branchen und Länder.

Sein persönliches Motto „Es gibt keine Probleme, sondern nur nicht gefundenen Lösungen“ spiegelt sich in jedem Projekt wider.

OKRs - Objectives and Key Results

von Stefan Ladstätter

Immer mehr Organisationen nutzen OKRs, um ihre Ressourcen auf die wichtigsten Prioritäten zu bündeln. OKRs ist eine Zielsetzungsmethode, vergleichbar mit Management by Objectives (MBO) und Balanced Score Card (BSC).

Bekannt geworden sind OKRs vor allem durch Google und Jim Doerr's Buch „Measure What Matters“, welches Googles OKR-Handbuch einer breiten Öffentlichkeit bekannt gemacht. Google selbst führt seinen kometenhaften Aufstieg auf die Verwendung von OKRs zurück, was viele Unternehmen aller Sparten dazu gebracht hat, ebenfalls OKRs einzuführen, mit wechselhaftem Erfolg. Im folgenden Artikel werden wir den Nutzen von OKRs und die Erfolgsfaktoren für die Einführung betrachten.

Wozu OKRs?

In unserer komplexen und dynamischen Marktumgebung sind Unternehmen sehr häufig mit den folgenden Herausforderungen konfrontiert:

- Mitarbeiter kennen die strategischen Ziele nicht und wissen nicht, wie sie zum Erfolg beitragen können
 - Fehlende Klarheit bezüglich der Prioritäten, Tagesgeschäft hat Vorrang gegenüber strategischen Veränderungen
 - Zu viele parallele Initiativen und Aktivitäten, fehlender Fokus auf die richtigen Themen
 - Starrer und langfristig ausgelegter Entscheidungsprozess (1+ Jahre), kein kontinuierlicher Lern- und Entwicklungsprozess
 - Silos verhindern fachübergreifendes Alignment
 - Fokus auf Leistungen statt auf Erfolge
- OKRs helfen hier auf unterschiedli-

chen Wirkungsebenen:

1. Zielsetzungsframework (Setzen und Strukturieren von Zielen)
2. Prozess für Team-Alignment (Zusammenarbeit an gemeinsamen Zielen)
3. Unternehmenstransformation (Silos aufbrechen)

Objectives and Key Results als Zielsetzungsframework

Das **Objective** beschreibt qualitativ, wohin die Reise gehen soll. Es ist ein **Ziel**, das innerhalb einer festgelegten Zeitperiode (meistens ein Quartal) erreicht werden soll. Es soll prägnant formuliert sein (ein Satz) und soll im Team inspirierend sein, Begeisterung wecken. Es ist für sich alleine umsetzbar und unabhängig von der Zulieferung anderer (außer es ist abgestimmt mit dem Objective eines anderen Teams). Es soll herausfordernd sein, aber nicht unschaffbar. Ein gutes Beispiel für ein Objective ist: „Launch von SEQITracker als geniales Tool“ oder „Benutzer unserer Website sind von unseren Inhalten begeistert“.

Key Results (3-5 pro Objective) beschreiben messbar, woran wir erkennen, dass ein Objective erreicht wurde. Empfehlenswert ist eine bunte Mischung aus mehreren Mess-Kategorien wie Wachstum, Umsatz, Performance, User-Engagement, etc. Beispiel: „Durchschnittliche Lese-dauer unserer Blog-Artikel von 1:38 auf 2:25 Min gestiegen“. KRs sollten schwierig zu erreichen sein, aber nicht unmöglich. Beim Einschätzen von KRs auf einer Schwierigkeitsskala von 1 („Sollte mit ein bisschen Anstrengung leicht möglich sein“) bis 10 („Das können wir gleich vergessen“) wollen wir den Sweet Spot dazwischen: „Wenn wir unser Bestes geben, können wir es schaffen!“.

Alle ziehen an einem Strang

Das ideale Umfeld für die Einführung von OKRs sind Unternehmen, in denen bereits agil gearbeitet wird, wo das Management von der Effizienz agiler Methoden überzeugt ist, jedoch selbst noch nicht unbedingt agile Denk- und Arbeitsweisen übernommen hat. Das ist auch gleichzeitig eine der größten Herausforderungen bei der Einführung von OKRs, denn das Management muss eine ähnliche Transformation durchmachen, wie sie der operative Bereich bereits hinter sich gebracht hat. Ohne eine Veränderung in der Einstellung und Arbeitsweise in der Führungsebene, werden OKRs keinen Fuß fassen, denn die Unternehmensführung gibt die strategische Richtung vor aus der sich alle anderen OKRs ableiten. Der gesamte Prozess kann ungefähr so beschrieben werden:

1. Die Geschäftsleitung bzw. das C-Level Board legt für das kommende Quartal (oder Geschäftsjahr) ein bis drei Objectives fest (je nach Unternehmensgröße bzw. Anzahl der Geschäftsfelder) und veröffentlicht diese. Dies sind die übergeordneten Ziele des Unternehmens für den kommenden Zeitraum
2. Passend zu den Unternehmens-Objectives werden von der GL in Zusammenarbeit mit den Team- bzw. Fachbereichsleitern jeweils 3-5 herausfordernde Key Results ausgearbeitet, die ebenfalls veröffentlicht werden
3. Diese OKRs dienen zusammen mit dem Unternehmensleitstern als Inspirationsquelle für die Objectives des Fachbereichs oder der Teams, jeweils mit eigenen Key Results. Auch diese werden veröffentlicht. Objectives können von Ebenen darüber übernommen werden

4. usw.

Hier ist eine grafische Darstellung eines OKR-Zyklus:

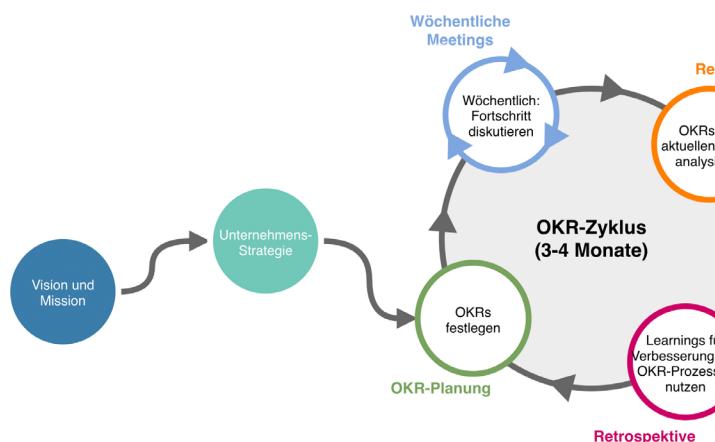


Abbildung: Der idealtypische OKR-Zyklus

So kaskadieren OKRs direkt und indirekt durchs Unternehmen. Der Schlüssel hierbei ist Transparenz. Für das Management ist das erst einmal nicht sehr angenehm, denn messbare Erfolgskriterien zu veröffentlichen bedeutet, sich angreifbar zu machen. Jeder Mitarbeiter kann den Fortschritt nachverfolgen, und da die Ziele auch „Stretch Goals“ sein sollen, besteht immer die Möglichkeit des Scheiterns. Hier geht das Management ein Risiko ein und hat damit eine große Vorbildwirkung für alle Teams und Mitarbeiter (siehe Kasten „Performance und Unternehmenskultur“).

eine Harmonisierung zwischen Management (High Level Strategy) und dem operativen Bereich (Umsetzung) geschaffen, die es in dem Ausmaß bisher nicht gab. Die Entwicklungsabteilung, die durch agile Methoden bereits sehr effizient („richtig“) gearbeitet hat, arbeitet nun auch effektiv („am Richtigen“). Die Energien, die durch Fokus und Alignment freigesetzt werden, sorgen dafür, dass die ehrgeizigen Ziele tatsächlich erreicht werden. In kürzerer Zeit mehr erreichen: der Traum jeder Organisation.

Performance und Unternehmenskultur

Shoot for the Moon and you will get further – klingt alles schön und gut, aber das Streben nach ehrgeizigen Zielen erfordert eine gewisse Unternehmenskultur. Wenn jedes Scheitern sofort sanktioniert wird, wird sich niemand mehr aus dem Fenster lehnen wollen. Es braucht eine Kultur, in der sich die Mitarbeiter sicher fühlen können und Lust haben, sich herauszufordern und auch einmal Fehler zu machen. Ein wichtiger Aspekt hierbei ist das Entkoppeln der Mitarbeiter-Performancereviews und Bonussen vom Erreichen der OKRs. Andernfalls werden Mitarbeiter tendenziell unterschätzen, wieviel sie erreichen können. Ehrgeizige Ziele sind zu gefährlich, denn was, wenn man scheitert?

OKRs können nur funktionieren, wenn Mitarbeiter ein Wagnis eingehen. Die Mitarbeiterbewertung darf daher auf keinen Fall an das Erreichen der Key Results gekoppelt sein. Es empfiehlt sich generell, die Bewertung auf andere Faktoren zu stützen, wie Engagement, Eigeninitiative, etc.

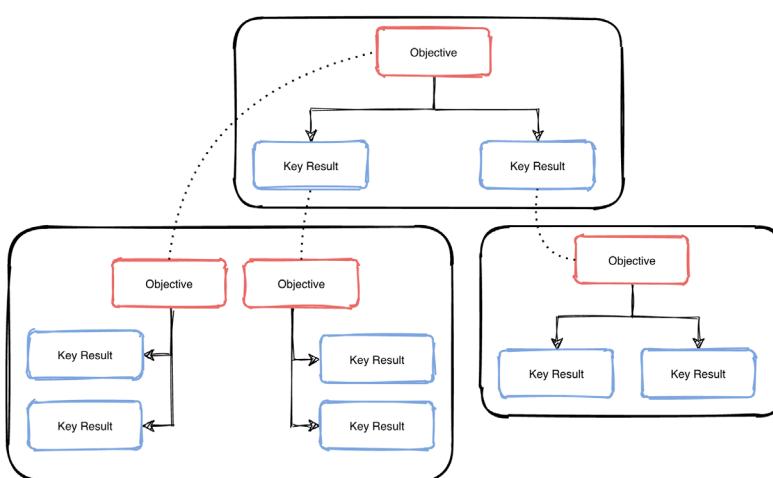
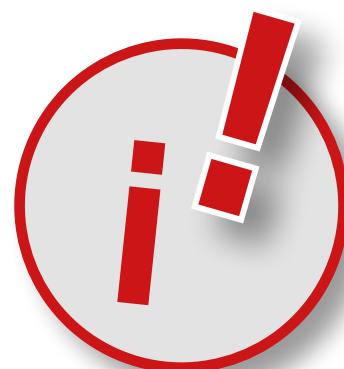


Abbildung: Beispiel einer OKR-Kaskade



Tipps zur Einführung von OKRs

Je nach Unternehmensgröße haben sich folgende Startszenarien herauskristallisiert:

- Erst mal lieber mit nur einem OKR für das gesamte Unternehmen beginnen (top down)
- Erst mal OKRs lieber nur in einem Team ausprobieren (Pilotprojekt), bevor die Methode auf das gesamte Unternehmen ausgerollt wird
- Zuerst OKRs für Projekte, um das Konzept von OKRs zu erproben

Auf jeden Fall sollte der Einführung genügend Zeit und Ressourcen gewidmet werden. Es kann ein paar Quartale (= Zyklen) dauern, bis sich der Prozess „eingeruckelt“ hat.

OKR-Tools können hilfreich sein, aber es empfiehlt sich, zuerst OKRs mit einfachen Mitteln (z.B. Google Sheets) iterativ einzuführen und an die Unternehmenskultur anzupassen, bevor ein Tool gekauft wird.

Wichtig ist in allen Fällen, dass OKRs von Bonusmodellen und Incentives entkoppelt sind.

Weitere Tipps:

- Geben Sie dem Erreichen der einzelnen OKRs die vollen drei Monate Zeit (wie ehrgeizig ist ein Ziel, dass man in nur 1 Woche umsetzen kann?)
- Das Erreichen der Objectives hat oberste Priorität, aber natürlich wird auch an anderen Dingen gearbeitet. Also: Nicht jeden Task in OKRs pressen, sondern den Teams vertrauen, dass sie ihre Arbeit tun werden
- Ermutigen Sie Mitarbeiter, Unternehmens-OKRs vorzuschlagen. OKRs sind bottom up genauso gut wie top down
- Machen Sie alle OKRs unternehmensweit zugänglich. Dadurch fördern Sie den Austausch zwischen den Teams ■



Interessante Videos aus den Bereichen IT Analyse, Development, Softwaretest und Projektmanagement finden Sie auf unserem Videoblog.

www.seqis.com/youtube



Stefan Ladstätter ist Consultant.

In seiner mehr als 20-jährigen Tätigkeit als Entwickler, Requirements Engineer, Usability-Spezialist und Projektmanager hat er Einblick in die unterschiedlichsten Aspekte der Umsetzung von IT-Projekten gewonnen und kann dabei auf einen tiefen Erfahrungsschatz im klassischen und agilen Umfeld vorweisen.

Er sieht erfolgreiches Projektmanagement als Dienst an allen Projektbeteiligten – Auftraggeber, Entwicklungsteam und Anwender.

Kurzfristige Projektunterstützung? SEQIS Experten!

von Dominik Tuchschnitt



Bild: iQconcept/shutterstock.com

Personal finden – Personal einsetzen – flexibler Einsatz sprichwörtlich von heute auf morgen. In den Bereichen Softwaretest – Business Analyse – Projektleitung – Development.

Sie brauchen kurzfristig, vielleicht schon „morgen“, Unterstützung in Ihrem IT Projekt? Aber geeignetes Personal für Ihren Standort außerhalb von Wien zu finden ist problematisch? Und durch die aktuelle Homeoffice Thematik haben Sie momentan wenig bis keine Möglichkeit Ihre gewohnten Recruiting-Prozesse durchzuführen, um geeignetes Personal anzustellen oder zu schulen?

Wir haben die Lösung: SEQIS Consultants unterstützen Sie effizient und zielgerichtet. Sie ersparen sich damit einen Großteil der administrativen Prozesse bei der Personalbeschaffung.

Zusätzlich können sich SEQIS Consultants durch langjährige Erfahrung und zielgerichtete Schulungsmaßnahmen besonders schnell in neue Projekte einfinden und vom Start weg produktiv mitarbeiten.

SEQIS?

Wir sind Softwaretester, IT Analysten, Developer und Projektmanager

aus Leidenschaft. Wir lieben es, Probleme auf den Grund zu gehen, individuelle Maßnahmen zu entwickeln und Anforderungen in Form von effizienten und userfreundlichen Lösungen zu erfüllen. Dafür haben wir genau die richtigen Leute an Bord, die mit ihrer Mischung aus Kompetenz und Persönlichkeit auch Ihr Projekt zum Erfolg führen.

Das Ganze funktioniert denkbar einfach. Sie teilen uns mit, welche Skills für welchen Zeitraum benötigt werden und wir liefern Ihnen ein maßgeschneidertes Unterstützungsangebot.



Bild: Sira Anamwong/shutterstock.com

Wenn Sie uns erreichen möchten, dann scannen Sie einfach den QR Code. Wir freuen uns auf Ihre Kontaktaufnahme!



Ganz einfach per Telefon 0664/8378722 oder per Email sales@SEQIS.com.

Personal finden – Personal einsetzen – flexibler Einsatz sprichwörtlich von heute auf morgen. In den Bereichen Softwaretest – Business Analyse – Projektleitung – Development. ■



Dominik Tuchschnitt ist Sales Mitarbeiter.

Als Allrounder deckt er ein breites Spektrum an Aufgaben ab. Der Schwerpunkt seiner Tätigkeit liegt im Bereich der Neukunden Akquise für die Geschäftsfelder: Analyse, Development, Test und Management.

Ganz oben auf der Prioritätenliste des Sales Profis steht, einen Nutzen und Mehrwert in den IT Projekten seiner Partner zu generieren.

... wie RPA bots das Leben des Fachbereichs erleichtern

von Alexander Weichselberger

Wer sich ernsthaft mit Softwaretest auseinandersetzt, muss sich auch mit den Thema Testautomation auseinandersetzen. Insbesondere durch die fundamentalen kürzeren Releasezyklen und den heute umfangreicher Softwarelösungen ist klar: Manuell geht nur im geringen Umfang, die Hauptlast muss automatisiert getestet werden. Und wer die Testautomationspyramide vor Augen hat,

Interfaces, aka GUI. Ein Schritt zurück vor 2012 - oder doch eine Evolution, die kommt?

Wozu RPA?

Nicht nur in der Software Entwicklung - auch die (End) Kunden von Unternehmen erwarten sich heute rasche Services und einer optimierten Ablauf (Customer Experience) durch das Unternehmen selbst. Dh. alles muss so

müssen die Durchlaufzeiten optimiert werden, damit der Eigenaufwand für das Unternehmen optimiert und die Kosten für den Kunden so gering wie möglich sind.

Tatsächlich stellen sich für die Non-IT die gleichen Herausforderungen, die wir auch schon seit 2001 für die IT im Zusammenhang mit Testautomation diskutieren: Rasche Time-to-Market, hohe Personalkosten, Fehleranfälligkeit insbesondere bei wiederkehrenden Tätigkeiten und zu geringe (Business) Process Automation.

Aus der Praxis: In einem Projekt war die Anlage von Kreditkarten im System durch das verfügbare Personal nicht machbar. Wir haben einfach unsere Automationsscripts für das Produktionsenvironment adaptiert und damit Nacht für Nacht tausende Karten automatisiert angelegt. Ohne die Applikation anpassen zu müssen - in Zeiten, wo durch diese Massenanlage lange Antwortzeiten für die Endanwender unter Tag vermieden werden konnten.

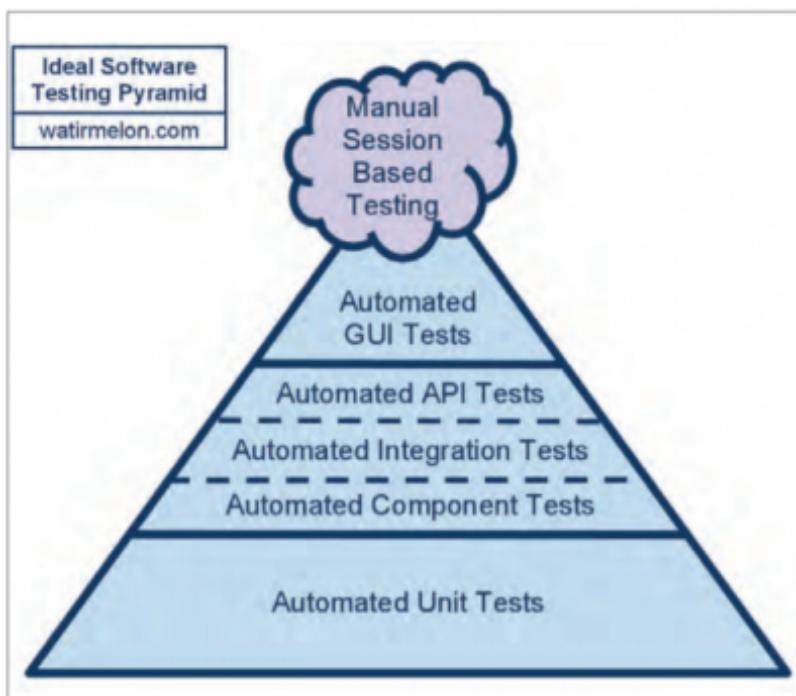


Abb.: Testautomationspyramide

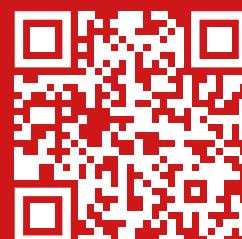
(Quelle: <http://watirmelon.com/2012/01/31/introducing-the-software-testing-ice-cream-cone/>)

kennt auch die übliche Verteilung der Interaktionsmöglichkeiten: Testklassen im Code Unit, viel via Component und API-Schnittstellen - und nur noch ein bisschen GUI. Diese Empfehlung ist ein klares Learning aus den Jahren vor 2012, wo man wider besseren Wissens zu viel über die GUI automatisiert hat. Warum stelle ich das in das Vorfeld der Diskussion? Weil Robotic Process Automation (RPA) jetzt wieder genau darauf ansetzt: Automation über die Standard User

rasch wie möglich ablaufen. Anbieter wie Amazon oder Zalando setzen auf Self-Service durch den Kunden und skalieren so die Workforce für den Inbound. Abgesehen vom tatsächlichen Picking- und Packing im Lager und Weiterleitung der Pakete an die Frachtführer müssen noch viele andere Prozesse durch Mitarbeiterinnen und Mitarbeiter dazu im Hintergrund erledigt werden: Produktentwicklung, Einkauf und Beschaffung, Marketing, Kundenservice, Finance... auch hier

SEQIS Blog

Profitieren Sie von den Erfahrungen unserer Experten!



Was ist RPA?

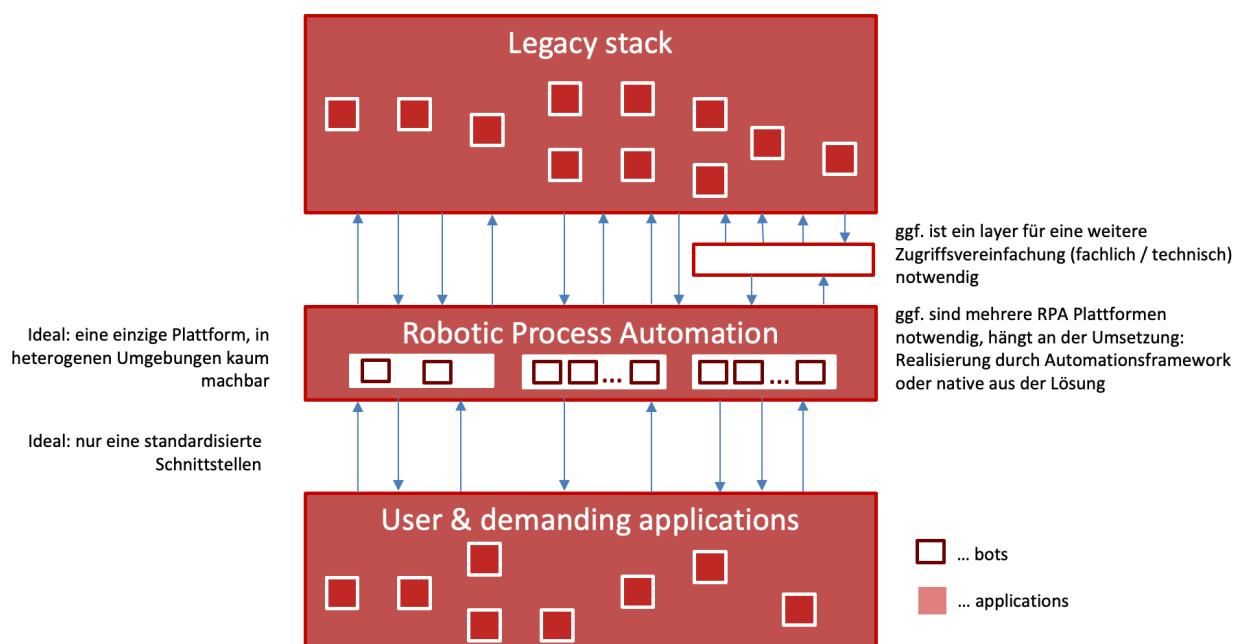
RPA ist ein Interaktions-Layer zur Steuerung vorhandener Anwendungen. RPA-Funktionen bestehen im Regelfall aus Screen Capture und record-and-replay Mechanismen, aufgepeppt durch Zeichenerkennung, Textanalysen (Intelligent RPA, Cognitive RPA) sowie der Möglichkeit Daten für die Verarbeitung aus anderen Quellen, z.B. Excel oder anfordernden Anwendungen, einzubinden.

durch Hinweise auf den ROI (Return Of Investment) wieder auf den Boden der Realität zurückholen: Man darf nicht alles, was man automatisieren kann, automatisieren - es geht um die Haupt-Hauptstraßen in der Wertschöpfung. Herstellungsaufwand und Wartbarkeit müssen passen. Sonst lieber Finger weg!

Die Ansätze bei RPA sind - wiederum - vergleichbar:

- Fokus darauf, wo Manpower

- relevanten Teilausschnitte Vorarbeiten optimieren: Manche Prozesse sind arbeitsintensiv, weil die Eingangsdaten unstrukturiert und deren Bearbeitung unterschiedlich reguliert ist; daher: Realisierung von Bots, die unstrukturierte Daten überarbeiten und die Daten anhand vordefinierter Regeln analysieren
- Fokus könnten beispielhaft CRM Applikationen sein: Viele Aufgaben sind wiederkehrend, regelbasiert und



© SEQIS GmbH, 2020

RPA ist keine Alternative zur non GUI Automation - sondern lediglich eine potentielle Ergänzung zu API. RPA ist - aufgrund des Zugangs durch die Oberfläche - empfindlicher auf Veränderungen wie eine API-Integration.

Wie vermeidet man Shelfware?

Eine klare Erkenntnis beim Einsatz von Testautomationstools war: Viele haben es gekauft, begeistert ins Rennen geworfen... - und mussten es auf Basis der schlechten Ergebnisse wieder zurück ins Regal stellen. Wir hatten in der Vergangenheit viele Projekte deren Ziel ein „back to automation“ war: Sinnvoller Einsatz - fokussiert auf den Nutzen. Nicht selten mussten wir in Projekten den Kunden

- hineinläuft: Automatisieren was oft- und oft verwendet wird
- Erweiterung der Integration: Schaffen von Interfaces to Services, die bislang - weil sie im Unternehmen nicht nativ integriert sind - manuell abgefragt wurden; ggf. wird man auch einfache Interfaces über spezialisierte Applikationen legen und damit direkte Maschine-zu-Maschine Kommunikation etablieren (-> kein manueller Eingriff mehr notwendig)
- Teilabdeckung: Oft sind die wahren Bottlenecks nicht gesamte Wertschöpfungsketten (komplexe Umsetzung), sondern nur Teile davon; deshalb: Fokus auf die

umfangreich i.S. einer hohen Anzahl. Ein echtes Potential!

Warum nicht alles in die Software integrieren - process improvement und Automation dorthin, wo es hingehört?

Meiner Meinung nach gibt es heute kein vernünftiges IT Projekt mehr, wo Prozessoptimierung und -automation keine Rolle spielt. Aber die IT Abteilungen sind überlastet oder die Lösungen sind cloudbasiert bzw. (Standard-)Produkte. Letztere sind daher mit nur einem reduzierten Umfang an Anpassungsmöglichkeiten durch/für das eigene Haus veränderbar (bzw. sind Anpassungen sehr teuer).

Wo wir mit RPA hinwollen und hinmüssen sind:

Anforderung	Lösungsansätze	Umsetzung
Rasche Umsetzung	<p>a.) Lösungsunabhängiges Automationsframework</p> <p>Einsatz eines 3rd Party Automationsframeworks mit umfangreichen Support für die eingesetzten Technologien (z.B. Ranorex, Tosca) Geringe Wartung durch object-based automation model für Änderungen in der Software Hohe Anzahl von nicht betreuten bots für einfache Aufgaben Dabei kann eine Wiederverwendung einer ggf. bestehende IT Testautomation eine gute Ausgangsbasis sein.</p> <p>b.) Integrierter „Taskrecorder“ in Lösung</p> <p>Nutzung von Aufzeichnung und Wiedergabe-Tools in der Lösung selbst (z.B. Microsoft Dynamics Finance & Operations (D365)) Nutzung der Aufzeichnung- und Wiedergabefunktionen im System (nicht nur für Schulungs- und Trainingszwecke) Viele damit automatisierte Teilstrecken (bei D365 ggf. Integration in RSAT (Regression Suite Automation Tool))</p>	<ol style="list-style-type: none"> 1. Auswahl des relevanten Prozesses 2. Modellieren des Ablaufs 3. Umsetzung der Automation 4. Stabilisierung des bots 5. Durchführen und Monitoring (KPIs) *) 6. Anpassung bei Prozess- und Interface Änderungen <p>*) Im Zusammenhang wichtig: Kontinuierliche Gegenüberstellung von Nutzen (wieviel Zeit spart die Automation) vs. Aufwand (zumindest: Aufwand die Erstellung der Automation, Durchführungszeiten sowie Zeitaufwand für Anpassungen); sonst kippt ggf. der ROI und wir sollen uns diese Automatisierungen sparen</p>
Einfache Verwendung durch die Endanwender	<p>Das Framework steht den Endanwender zur Verfügung - damit wird die IT skaliert und Bottlenecks werden vermieden</p>	<ol style="list-style-type: none"> 1. Bereitstellung von „Keywords“ (einzelne automatisierte Prozessschritte) *) 2. Zusammenbau der Keywords durch einen Keyuser im Fachbereich 3. Eventgesteuert integriert oder mit Scheduler getriggerte Durchführung 4. Monitoring Interface für den Fachbereich 5. Rascher Support durch IT im Problemfall <p>*) Die für den Keyword-and-data-driven-Approach notwendige Datenbereitstellung wird oft auch durch Excel realisiert; weitere Details zur Methode siehe https://de.wikipedia.org/wiki/Keyword-Driven_Testing</p>

Fachabteilungen haben den Lead bei der RPA

Für die Umsetzung und Realisierung von IT Lösungen ist es üblich, dass die IT Abteilung federführend ist und bleibt.

Dieses Prinzip sollte für RPA aufgeweicht werden: RPA ist eine Geschäftsinitiative. Änderungen sollten auch unter Berücksichtigung bestehender produktiver (!) Automations-scripts umgesetzt werden. Planen Sie mit ein, ggf. Wartung bei den bots zu machen ist. Hier ist die IT Abteilung wichtig, weil - selbst bei cloudbasierten Lösungen - Änderungen immer durch die IT Abteilung laufen und sie insbesondere das Timing der Umsetzung führt. Andererseits sollte die notwendige Anpassung von RPA bots jedoch durch den Fachbereich umgesetzt werden - sie sind aus meiner Sicht im Lead.

Gleichermaßen sollten Umsetzungen in diesem Bereich von der IT unterstützt werden (z.B. durch Konzeption, Architektur, Hilfe bei technisch komplizierten Umsetzungen), aber folgende Punkte sollten durch den Fachbereich gemacht werden:

- Bedarfsanalyse, wo welche Strecken automatisiert werden sollten (Fragestellung: WAS ist zu tun?)
- Detailanalyse, was konkret - Schritt für Schritt - automatisiert werden soll (Fragestellung: WIE ist es zu tun? - eine übliche Domäne der IT Abteilung)
- Umsetzung, Test und Dokumentation der RPA, dabei auch Festlegung KPIs und Einrichten des fachlichen Monitoring
- Verwendung der RPA

In der IT Abteilung bleiben zumeist - wenn notwendig - das Deployment, sowie die Realisierung des technischen Monitoring der RPA. Ggf. - falls dies notwendig ist - natürlich auch technische Hilfe für die Realisierung komplizierter Automation Keywords.

Aber Achtung: Ein Schlüsselfaktor muss jedenfalls bei der IT Abteilung bleiben, die letztlich für den Betrieb der Lösung verantwortlich ist: Die Spielregeln für RPA müssen von der IT kommen - insbesondere auch wg. der vermeintlich erhöhten Privilegien der RPA bots im Vergleich zu „Standard-User“. Darüber hinaus sollten weitere Bereiche, wie z.B. Human Ressource und ggf. auch der Betriebsrat, bei der Abstimmung der Spielregeln berücksichtigt werden.

Warum ist RPA ein next „big thing“ im Software Engineering?

Die Vorteile von RPA liegen in der schnellen Implementierung von Prozessautomatisierung. Mehr und mehr „Entwicklungsleistungen“ werden dabei von der IT Abteilung in Richtung Fachbereich gehen. Die IT Abteilungen sind aufgerufen, diesen Weg zu begleiten, um damit mündigere Keyuser in den Abteilungen zu etablieren. Mit dem wesentlichen Vorteil, dass RPA ein Serviceangebot aus der IT Abteilung ist - im Gegensatz zu produktiven Lösungen, die durch die Fachabteilungen realisiert wurden und ohne Kenntnis durch die IT betrieben werden.

RPA - richtig realisiert - ist ein kontinuierlicher Improvement-Prozess. Dabei...

- ... sollte es keinesfalls für komplexe Aufgabenstellungen herangezogen werden
- ... muss eine Agile Umsetzung (schneller Nutzen durch umgesetzte Prozesse) gewählt werden
- ... darf das RPA Engagement kein Argument sein, dass eine Anwendungsmodernisierung verzögert wird und damit die Lebensdauer von Legacy-Anwendungen unnötig verlängert wird
- ... sollten weitere Self-Service Optionen z.B. im DWH Bereich evaluiert werden - damit steigert man Selbstbewusstsein der Keyuser erneut und führt zu mündigeren IT Partnern in den eigenen Fachbereichen (Product Owner)

Die Reise geht in Richtung gemeinsam - statt einsam (nur durch die IT Abteilung). Das ist das wahre Next Big Thing in Software Engineering. ■



Alexander Weichselberger ist Managing Partner.

Er hat seine Einsatzschwerpunkte in den Bereichen Systemanalyse, Softwaretest, Koordination und Management von exponierten Großprojekten und kann auf jahrelange Erfahrung zurückblicken.

Dieses Wissen gibt er gerne in Form von Coachings, Methodentrainings und Fachvorträgen weiter.

Das Cynefin-Framework: Komplex oder kompliziert?

von Stefan Ladstätter

Warum kommen so viele (IT-)Projekte ins Strudeln? Budgetüberschreitungen, verpasste Deadlines, mangelnde Akzeptanz beim Endbenutzer, ... Die Gründe sind sehr unterschiedlich, doch eines ist offensichtlich: Je höher die Komplexität eines Projekts, desto höher die Wahrscheinlichkeit, dass es scheitert. Das Cynefin-Framework liefert wertvolle Strategien zum Umgang mit Komplexität.

Die Zahlen lügen nicht: Laut der Standish Group und ihrem alle 2 Jahre erscheinenden CHAOS-Report, der den Verlauf von 50.000 IT-Projekten über einen Zeitraum von 5 Jahren analysiert, werden nur 16% aller IT-Projekte im vereinbarten Zeitrahmen, im Budgetrahmen und mit der versprochenen Funktionalität abgeschlossen. Mehr als die Hälfte (53%) der Projekte gerät ins Strudeln, nahezu ein Drittel (31%) sind komplett Fehlschläge. Quelle: <https://www.standishgroup.com/>

Das Cynefin-Framework

Cynefin (ausgesprochen: ke-néh-win) ist ein Analyse- und Entscheidungs-Framework, oft wird es auch als Sensemaking-Framework bezeichnet. Es kommt in den unterschiedlichsten Bereichen zur Anwendung, wie Projektmanagement, Wissensmanagement und IT-Systemdesign. Erfunden wurde Cynefin 1999 von Dave Snowden bei IBM Global Services, seither wird es ständig adaptiert und weiterentwickelt. Dabei berücksichtigt es neueste Erkenntnisse aus System- und Komplexitätstheorie sowie unterschiedlicher Lerntheorien.

Aus der Systemtheorie kennen wir 3 grundsätzliche Systeme: Geordnet, komplex, chaotisch. Cynefin beleuchtet die kausalen Unterschiede

zwischen verschiedenen Arten von Systemen und hilft beim Wechsel von einem System ins andere. Im Cynefin-Modell kommt eine weitere Kategorie hinzu („konfus“). Außerdem wird das „geordnete“ System unterteilt in „klar“ und „kompliziert“.



Abbildung: Cynefin-Framework, Quelle: Cognitive Edge, Copyright 2017

Die fünf Domänen: Klar, Kompliziert, Komplex, Chaotisch, Konfus

In der **klaren Domäne** gibt es einen vorhersehbaren Zusammenhang zwischen Ursache und Wirkung, und dieser Zusammenhang ist auch für Laien offensichtlich. Wir erkennen, was auf uns zukommt, ordnen es zu und reagieren dann mit der entsprechenden besten Vorgehensweise. Dies ist die Domäne von „Best Practice“.

Auch in der **komplizierten Domäne** gibt es einen Zusammenhang zwischen Ursache und Wirkung, dieser ist jedoch nicht immer offensichtlich. Das Problem muss erst analysiert werden, oder wir holen den Rat eines Experten ein. Dies ist die Domäne von „Good Practice“. Wichtig ist hier der Unterschied zwischen Best Practice und Good Practice. In der komplizierten Domäne gibt es unterschiedliche Wege zum Erfolg, mehrere mögliche anerkannte Verfahren. Die Entscheidung, welcher davon gewählt wird, ist abhängig von der verfügbaren Expertise, alles andere ist sehr riskant.

In **komplexen Systemen** sind Ursache und Wirkung erst im Nachhinein erkennbar. Alles ist vernetzt, das System ändert sich ständig. Die Lösung ist nicht vorhersehbar, und egal, was wir tun, wir müssen mit unerwarteten Nebeneffekten rechnen. Hier sind unterschiedliche Experimente mit jeweils wenig Einsatz gefragt. Geht das Ergebnis einer Intervention in die richtige Richtung, setzen wir verstärkende Maßnahmen. Geht es in die falsche Richtung, bremsen wir. Wir sprechen von „Emergent Practice“, einer Herangehensweise, die sich erst auf dem Weg herauskristallisiert und dabei oft Praktiken aus anderen Feldern zweckentfremdet.

Gerät man in ein **chaotisches Umfeld**, ist schnelles Handeln gefragt, um die Situation zu stabilisieren (Beispiel: Corona-Krise). Hier sind komplett neuartige Herangehensweisen gefragt („Novel Practice“), daher ist dies die Domäne der Innovationen, und ein absichtlicher Ausflug ins Chaos kann sehr lohnenswert sein.

Diese Einteilung ist eine sehr brauchbare Grundlage für Entscheidungen und Lösungswege. Je nach Kontext unterschiedlich denken und analysieren – so simpel, nicht wahr?

Der Mensch ist ein Gewohnheitstier

Damit kommen wir zur letzten Domäne, in der wir die meiste Zeit verbringen. Konfus ist ein Zustand, in dem wir uns in Unklarheit befinden, welcher Domäne eine Problemstellung oder Situation zuzuordnen ist. Dieser Unklarheit sind wir uns oft nicht bewusst, was katastrophale Folgen haben kann, wenn wir daraus die falschen Lösungsansätze wählen. Der Grund? Unsere Einschätzung einer Problemstellung ist abhängig von unserer persönlichen Erfahrung

und unserer persönlichen Vorlieben.

Wer viel Zeit in einem bürokratischen, von Prozessen definierten Umfeld verbracht hat, wird irgendwann alle Probleme als Prozessmängel erklären. Ein Fachexperte rechtfertigt auftauchende Probleme mit Mangel an Zeit und Ressourcen für die eigentlich erforderliche, tiefe Analyse der Situation. Wer es gewohnt ist, sich in komplexen System zu bewegen, wird möglichst viele Menschen aus möglichst vielen unterschiedlichen Bereichen um sich versammeln, in der Hoffnung, dass sich aus der Zusammenarbeit eine Lösung herausträumt (übrigens eine sehr empfehlenswerte Herangehensweise). Jemand, der gerne andere herumkommandiert, wird einen autokratischen Führungsstil wählen und alle Entscheidungen top-down vorgeben.

Ein Framework wie Cynefin erlaubt es uns, diesem ersten Instinkt und der gewohnten Reaktion zu widerstehen und zu sagen: „Dieses Problem ist kompliziert, wo finden wir einen Experten, der uns weiterhelfen kann“ oder „Wartet mal, das hier ist komplex, daher sollten wir uns ein paar Experimente überlegen.“

Was bedeutet dies konkret für Software-Projekte?

Aus der Domäne ergibt sich die jeweilige Herangehensweise für die Umsetzung. Bei komplizierten Aufgabenstellungen haben sich Waterfall- und V-Modell bewährt. Hier ist für Experten ein klarer Weg von A nach B erkennbar, daher kann man alle Anforderungen im Vorfeld abklären und eine genaue Aussage darüber treffen, wie lange die Umsetzung braucht und welche Ressourcen dafür nötig sein werden.

Bei komplexen Problemstellungen bewährt sich ein agiler Ansatz. Hier kommen multifunktionale Teams zum Einsatz. Anforderungen sind anfangs sehr grob skizziert, da viele Aspekte

noch unbekannt sind, einschließlich der Marktentwicklung. Priorisiert wird nach Geschäftsnutzen. In die „nice to haves“ wird erst investiert, wenn deutlich wird, wohin die Reise geht. So wird das Risiko minimiert, Ressourcen zu vergeuden, und auf Veränderungen kann sofort reagiert werden. Die Entwicklung erfolgt iterativ und timeboxed, d.h. als eine Reihe von kleinen, „safe-to-fail“ Experimenten. In Retrospektiven wird ermittelt, was funktioniert und was nicht, und die jeweiligen Aspekte werden gefördert oder abgeschwächt.

Man könnte auch sagen, dass sich Agilität im Übergangsbereich zwischen komplex und kompliziert abspielt. Wir holen uns aus der komplexen und oft nebulösen Problemstellung ein paar wenige, klar definierbare Aspekte heraus, um sie mit bewährten Mitteln als „komplizierte“ Probleme zu lösen.

Vorsicht, Absturzgefahr!

Während die meisten Grenzen zwischen den Domänen fließend sind, gibt es eine wichtige Ausnahme. Der Übergang von klar zu chaotisch ist sehr abrupt. Wer aufgrund vergangener Erfolge an die eigene Unverwundbarkeit glaubt, wird bequem und selbstgefällig. Im Fall einer unerwarteten Problemsitu-

ation besteht die große Gefahr, die Klippe zum Chaos hinunterzustürzen. Eine Erholung ist nur sehr schwer möglich und sehr teuer.

Es empfiehlt sich daher, sich mehrheitlich in den Domänen komplex und kompliziert aufzuhalten und nur wenige Aspekte in die klare Domäne zu verlagern, um die Resilienz gegen abrupte Veränderungen zu erhöhen.

Abschlussgedanken

Wir Menschen fühlen uns am wohlstesten, wenn wir Voraussagen über unsere Zukunft treffen können. In der heutigen, schnelllebigen Welt ist das oft nicht der Fall, daher wird es immer wichtiger, Strategien zu entwickeln, mit Komplexität umzugehen. Das Cynefin-Framework bietet uns dafür eine wertvolle Hilfestellung, da es uns einerseits hilft, unseren Standort zu bestimmen, und andererseits passende Strategien vorschlägt, auch wenn es auf den ersten Blick selbst etwas kompliziert wirkt. ■

Quellen und weiterführende Informationen:

Weiterführende Links:
www.cognitive-edge.com



Stefan Ladstätter ist Consultant.

In seiner mehr als 20-jährigen Tätigkeit als Entwickler, Requirements Engineer, Usability-Spezialist und Projektmanager hat er Einblick in die unterschiedlichsten Aspekte der Umsetzung von IT-Projekten gewonnen und kann dabei auf einen tiefen Erfahrungsschatz im klassischen und agilen Umfeld vorweisen.

Er sieht erfolgreiches Projektmanagement als Dienst an allen Projektbeteiligten – Auftraggeber, Entwicklungsteam und Anwender.

Künstliche Intelligenz & Machine Learning

von Markus Schwabeneder

Das sind 2 Begriffe, die spätestens durch die selbstfahrenden Autos und die immer besser werdenden Sprachassistenten in aller Munde sind. Ich habe mir Gedanken darüber gemacht, was man darunter versteht, welche verschiedenen Arten es davon gibt und ob das alles wirklich so neu ist.

Arten der Intelligenz

Denken wir mal darüber nach, welchen technischen Dingen das Adjektiv intelligent angeheftet wird: Da gibt es intelligente Staubsauger, intelligente Abgasreinigung bei Kraftwagen, intelligente Programmwahl bei Waschmaschinen, intelligente Beschattung in Smart Homes und vieles mehr. Das sind Dinge, die wir jetzt nicht unbedingt mit Künstlicher Intelligenz in Verbindung bringen und sicherlich wird hier das Wort "intelligent" hauptsächlich aus Werbezwecken verwendet, aber ist dies auch gerechtfertigt? Wir denken doch eher an einen menschenähnlichen Roboter oder zumindest an einen Computergegner in einem Spiel, der sich annähernd wie ein Mensch verhält. Beides ist intelligent, man spricht hier aber von schwacher und starker Intelligenz. Schwache Intelligenz bedeutet, dass ein konkretes Problem mit teilweise unbekannten Parametern selbstständig gelöst werden kann. Starke Intelligenz verlangt darüber hinaus, dass sehr komplexe Aufgaben erledigt werden können, das System dazu lernt und mit diversen Störungen umgehen kann, ähnlich wie es ein Mensch machen würde.

Alles schon dagewesen?

Im Kern würde ich sagen: "Ja". Aber natürlich nicht in der Qualität und Quantität. Im Bereich der schwachen KI sind die augenscheinlichsten Verbesserungen Hardware-technischen

Ursprungs. Schon Mitte des letzten Jahrhunderts gab es funktionierende Steuerungselektronik, die ich als schwach intelligent bezeichnen würde. Es war halt teuer und groß. Jetzt kriegt man Laserabtastung und Routenplanung auf einen kleinen Chip um wenige Euro und packt das in einen Saugroboter. Ebenso ist eventuell schon ein Steuerungsschip mit Abwassersonde billiger als ein Schaltpanel mit zig Waschprogrammen. Jede zweite Zahnbürste hat ja jetzt schon WLAN und zeigt an, wie man putzen soll. Ich will hier nicht hinterfragen, ob das sinnvoll ist - die Kosten so etwas einzubauen, sind offensichtlich aber nicht sehr hoch.

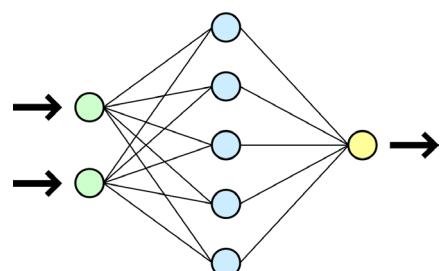
Auch auf der Seite der starken Intelligenz hat sich einiges getan, hier aber schon eher auf der Software-Seite. Die Grundlagen gibt es zwar auch alle schon ähnlich lange wie die der schwachen Intelligenz, es hat sich hier aber schon einiges getan. Verschiedene Methoden wurden kombiniert und verfeinert. Natürlich hilft hier auch, dass Rechenleistung immer günstiger wird. Es ist aber auch ein Faktor, dass einfach immer mehr Menschen diese nutzen und damit auch "testen". Die Suchmaschinen von Google wären nie so gut geworden, würden sie nicht milliardenfach am Tag benutzt und damit getestet werden.

Machine Learning

KI kann sich auch selber verbessern bzw. lernen. Dies kann prinzipiell auf 2 Arten passieren: Symbolisch und nicht-symbolisch.

Symbolisch bedeutet, dass man den Erkenntnisgewinn ablesen kann, z.B. weil ein neuer Eintrag in eine Wissensdatenbank geschrieben wurde oder ein neuer Algorithmus erstellt

wurde. Bei nicht-symbolischem Lernen ist dies nicht möglich, es verändert sich das ganze System, ein Beispiel dafür sind neuronale Netze (das sind Modelle, die das menschliche Gehirn nachahmen: Beim Lernen werden Knoten und Kanten angelegt bzw. verändert, ähnlich wie es im menschlichen Gehirn passiert).



Dake, Mysid (https://commons.wikimedia.org/wiki/File:Neural_network.svg), „Neural network“, <https://creativecommons.org/licenses/by/1.0/legalcode>

Eine weitere Unterscheidung ist, wie das Lernen erfolgt. Dies kann "überwacht" passieren, das heißt, die KI versucht eine Aufgabe zu lösen und bekommt vom "Lehrer" Feedback, ob sie das gut gemacht hat oder nicht. Unüberwachtes Lernen läuft anders ab. Hier bekommt die KI Daten und die KI erstellt ein statistisches Modell, das möglichst gut zu den Daten passt.

Weiterentwicklung der KI am Beispiel von Schachprogrammen

Da ich selber Schachspielen zu meinen Hobbies zähle, interessiere ich mich dafür und hab auch schon als Kind Schachcomputer besessen und deren Weiterentwicklung als User miterlebt. Anfangs waren nur einfache Algorithmen implementiert. Der Computer kannte die Regeln, berechnete alle möglichen Stellungen die nach 2 (eventuell 3) Zügen auftreten könnten und bewertete diese nach einem vordefinierten Schema.

Damit war es jedem halbwegs guten Schachspieler weit unterlegen. Die ersten Verbesserungen waren, dass man Datenbanken für Eröffnungen und Endspiele hinzufügte (also das Gedächtnis von Menschen nachbaute). Danach schaffte man es, Algorithmen zu finden, die nicht jeden möglichen Zug ausprobierten, sondern interessante Züge sehr weit und uninteressante Züge kaum zu analysieren (wie es eben auch ein Mensch machen würde). Damit war man dann schon auf Augenhöhe mit den besten menschlichen Spielern.

Der nächste Durchbruch war, dass man Monte-Carlo Methoden mit einbezog. Das heißt, es wurden zufällig eine große Menge von Stellungen ausgewählt und bewertet, die nach einem Zug entstehen könnten (damit wurde das "Bauchgefühl" eines Menschen simuliert). Ab diesem Zeitpunkt waren Schachprogramme den Menschen weit überlegen. Die aktuell stärksten Schachprogramme benutzen auch neuronale Netze und lernen ständig dazu.



waterborough (<https://commons.wikimedia.org/wiki/File:Radioshack16504.jpg>), „Radioshack16504“, als gemeinfrei gekennzeichnet, Details auf Wikimedia Commons: <https://commons.wikimedia.org/wiki/Template:PD-self>

Ausblick: Wie wird sich die Software-Entwicklung verändern

Ein Ausblick in die Zukunft ist natürlich wie Glaskugel-Lesen, ich werde hier aber trotzdem meine Gedanken dazu darlegen.

2 Dinge sind in den letzten Jahren erkennbar:

1. Die schwache KI nimmt uns

schon derzeit immer mehr Arbeit ab. Sie wird in noch mehr Bereiche Einzug finden und auch im Softwarebereich noch mehr Aufgaben erledigen

2. Die starke KI wird derzeit noch weiterentwickelt, indem die Natur nachgeahmt wird. Noch (und ich denke in der nächsten Jahrzehnten) ist die Kopie dem Original deutlich unterlegen. Diese Aussage mag seltsam wirken, wenn ich gerade im Abschnitt vorher gemeint habe, dass die KI schon deutlich besser Schach spielt als der Mensch. Schachspielen ist aber eine zwar relativ komplizierte, aber doch ein wenig komplexe Aufgabe mit wenig und klaren Regeln und Eingabeparametern

Daraus leite ich ab, dass in Zukunft unsere Aufgaben weniger in der konkreten Umsetzung, sondern noch mehr in der Analyse, Entwurf und Kontrolle liegen werden.

Ich kann mir gut vorstellen, dass bereits in naher Zukunft die KI selbstständig Programme schreibt (diese müssen nicht "menschlesbar" sein, ein Beispiel dafür sind eben neuronale Netze) um Problemstellungen zu lösen. Ich denke aber nicht, dass die KI diese Problemstellungen

selbst herleiten kann, das wird auch in Zukunft Aufgabe von uns sein. Der Programmier wird sich dann mehr in Richtung Requirements-Engineer entwickeln. Seine Aufgabe wird sein, die Bedürfnisse und Anforderung zu sammeln und ein Modell zu bauen, das die KI versteht. Ähnlich werden sich auch die Aufgaben der Tester verändern. Es wird weniger wichtig werden, Flüchtigkeits- oder Logikfehler im Programmcode zu finden als viel mehr zu testen, ob das Programm überhaupt das macht, was es soll.

Ich bin auf jeden Fall gespannt, wie sich unsere Arbeitswelt verändern wird. Ich denke hier bieten sich viele Chancen, man muss aber offen für Veränderung sein und Freude an der Arbeit gemeinsam mit anderen Menschen haben. Softskills wie Konfliktlösungskompetenz, Empathie und Kommunikationsfähigkeit werden noch wichtiger. Der einsame Programmier-Nerd im Hinterzimmer hat endgültig ausgedient. ■



Markus Schwabeneder ist Senior Consultant.

Er begann seine Karriere in der Softwarebranche als Consultant für komplexe mathematische Themen und Optimierungsaufgaben. Seitdem ist er in den Bereichen Softwareentwicklung, Anforderungsanalyse, Testing und Softwarearchitektur tätig. Seine Fähigkeiten setzt er in unterschiedlichen Branchen ein.

Besonders begeistert er sich für das Erarbeiten von komplexen Anforderungen und spezifischen Vorgehensweisen.

Die nächsten
Termine im
Überblick:

Agile Circle ONLINE

28. Jänner 2021

Weitere Informationen finden Sie auf Seite 35 und auf unserer Website www.agilecircle.org

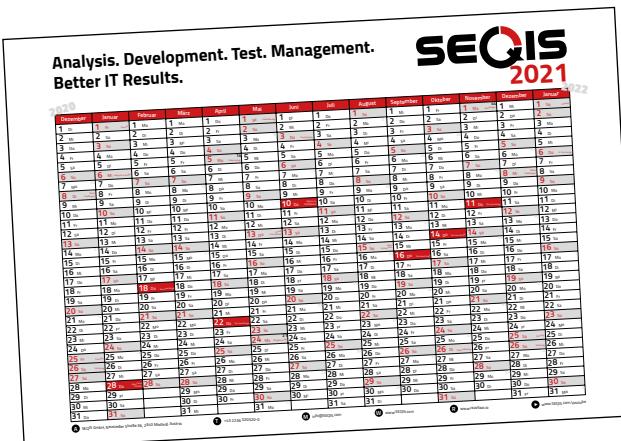
Haben Sie schon Ihre
nächste Weiterbildung
geplant?

www.SEQIS.com

November		Dezember
1 So	Allerheiligen	1 Di
2 Mo		2 Mi
3 Di		3 Do
4 Mi		4 Fr
5 Do		5 Sa
6 Fr		6 So 2. Advent
7 Sa		7 Mo
8 So		8 Di Mariä Empfängnis
9 Mo		9 Mi
10 Di		10 Do
11 Mi		11 Fr
12 Do		12 Sa
13 Fr		13 So 3. Advent
14 Sa		14 Mo
15 So		15 Di
16 Mo		16 Mi
17 Di		17 Do
18 Mi		18 Fr
19 Do		19 Sa
20 Fr		20 So 4. Advent
21 Sa		21 Mo
22 So		22 Di
23 Mo		23 Mi
24 Di		24 Do
25 Mi		25 Fr Christtag
26 Do		26 Sa Stefanitag
27 Fr		27 So
28 Sa	1. Advent	28 Mo
29 So		29 Di
30 Mo		30 Mi
		31 Do

Januar	
1 Fr	Neujahr
2 Sa	
3 So	
4 Mo	
5 Di	
6 Mi	Hl. Drei Könige
7 Do	
8 Fr	
9 Sa	
10 So	
11 Mo	
12 Di	
13 Mi	
14 Do	
15 Fr	
16 Sa	
17 So	
18 Mo	
19 Di	
20 Mi	
21 Do	
22 Fr	
23 Sa	
24 So	
25 Mo	
26 Di	
27 Mi	
28 Do	Agile Circle ONLINE
29 Fr	
30 Sa	
31 So	

Planung ist die halbe Miete



Der neue SEQIS Jahresplaner ist da!

Diesmal wieder in **zwei Größen** und jetzt neu **wiederbeschreibbar**. Bestellen Sie jetzt Ihren kostenlosen Jahresplaner und Sie bekommen einen SEQIS **Whiteboardmarker**, zum Beschriften des Kalenders dazu. Als Geschenk erhalten Sie ebenfalls einen **Schokokrampus**.

Bestellen Sie Ihren Jahresplaner noch heute, denn die Anzahl ist limitiert.

Weitere Details finden Sie auf unserer Website
www.SEQIS.com



Viel Spaß beim Planen wünscht Ihnen Ihre
 SEQIS GmbH



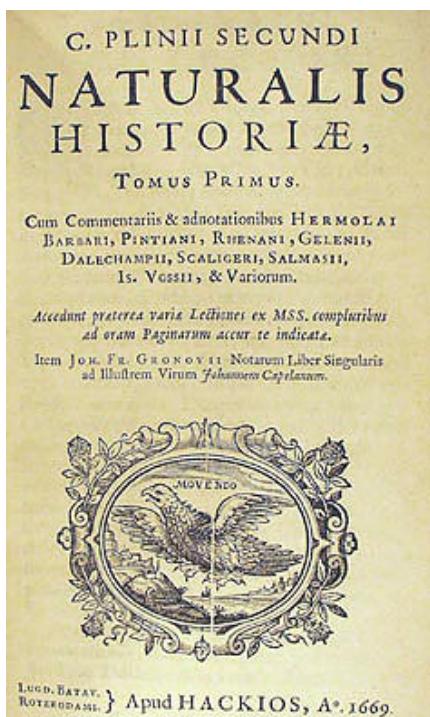
Bild: Anura-dsgn/Shutterstock.com

Wissensmanagement und die Demokratisierung des Wissens

von Andreas Steiner

Ursprünge des Wissensmanagements

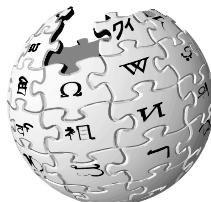
Genaugenommen ist Wissensmanagement bereits mehrere tausend Jahre alt. Die bekannteste Form ist die Enzyklopädie. Die älteste vollständig erhaltene Enzyklopädie reicht bis zum ersten Jahrhundert nach Christus zurück. Die *Naturalis historia*. Erst im 18. Jahrhundert hat sich aber die heutige Form des Nachschlagewerks entwickelt.



anonym (<https://commons.wikimedia.org/wiki/File:Naturalisthistoria.jpg>), „Naturalisthistoria“, als gemeinfrei gekennzeichnet, Details auf Wikimedia Commons: <https://commons.wikimedia.org/wiki/Template:PD-old>

Der bekannteste deutschsprachige Vertreter ist wohl der Brockhaus (seit 1808). Seit den 1980er Jahren haben sich das Medium dafür immer weiterentwickelt. So entstand 1993, die weitbekannte CD-ROM von Microsoft, die Microsoft Encarta. Mit dem Internet hat sich die Demokratisierung des Wissens weiter gewan-

delt und so wurde 2001 Wikipedia ins Leben gerufen. Eigentlich als Nebenprojekt der Nupedia, welche im Gegensatz mit Fachartikeln von Experten gefüllt wurde, von denen jeder Artikel einen eigenen Reviewprozess durchlaufen musste. Durch die hohen Kosten wurde die Nupedia aber nach wenigen Jahren eingestellt und die Artikel in die Wikipedia übernommen.



penubag ([https://commons.wikimedia.org/wiki/File:Wikipedia_logo_\(svg\).svg](https://commons.wikimedia.org/wiki/File:Wikipedia_logo_(svg).svg)), „Wikipedia logo (svg)“ <https://creativecommons.org/licenses/by-sa/3.0/legalcode>

Wie schon oft berichtet, ist auch sehr viel Unwissen in Online-Enzyklopädien dokumentiert. Daher ist es ratsam, Quellenverweise zu überprüfen und das eigene Wissen weiter zu vertiefen. Da es aber auch viele Fachgebiete gibt, welche nicht abgedeckt werden können, haben sich unzählige Wikis und auch Communities für die unterschiedlichsten Themen-Bereiche entwickelt. Es ist nämlich nicht nur wichtig Wissen zu dokumentieren, auch der Austausch über Wissen muss ermöglicht werden. So gibt es mittlerweile unzählige Plattformen im Internet, um Wissen zu dokumentieren und auch auszutauschen. Beispielsweise die Plattform StackOverflow (<https://stackoverflow.com/>), welche einem Entwickler schnell Lösungen für Problemstellungen liefern kann aber genauso neue Problemstellungen diskutiert werden können, um eine passende Lösung für den spezifischen Anwendungsfall zu liefern.

Um mehr über Programmierer und Programmierung zu erfahren, informieren wir Sie in dem Video zum **Tag des Programmierers** auf unserem YouTube-Kanal „SEQIS GmbH“



Auch heute gilt noch, das Rad muss nicht immer neu erfunden werden. Natürlich ist aber eine gewisse Kompetenz notwendig, die beste Lösung zu finden und diese auch zu bestätigen.

Wissensmanagement in Unternehmen hat seinen Ursprung erst in den 60er Jahren und wird bis heute weiterentwickelt. Bestimmt haben Sie in Ihrem Unternehmen mindestens eine Software-Lösung, vermutlich mehrere, zur Dokumentierung von Wissen. Wozu braucht man Wissensmanagement eigentlich?

Wissensmanagement hilft uns dabei zu wissen, was wir wissen!

Jede/r Mitarbeiter/in in Ihrem Unternehmen, verfügt über Wissen, welches geteilt werden kann – und eigentlich auch sollte. Dieses Experten- und Fachwissen befindet sich oft nur in den Köpfen der Mitarbeiter und wird an neue Kollegen in Form von Einschulungen weitergegeben. Steht man nun vor einem Problem und fragt nach, passiert es allzu oft, dass man an einen Mitarbeiter weitergeleitet wird. Dieser schickt einem zu dem

Nächsten, bis man im schlimmsten Fall eine Runde gedreht hat und so weit ist wie zuvor. Da wäre es doch gut, eine Plattform zu haben, bei der vorhandenes Wissen im Unternehmen dokumentiert wird und eingesen werden kann. Das hilft dabei, effizient Lösungen für bekannte Problemstellungen zu finden aber genau so Lücken zu finden, welche man über externes Expertenwissen füllen kann, um auch weiterhin wettbewerbsfähig und erfolgreich zu bleiben.

Wissen ist das einzige Gut, das sich vermehrt, wenn man es teilt!

In der Praxis gibt es aber einige Stolpersteine, welche einer erfolgreichen Umsetzung im Weg stehen. Immer wieder trifft man noch auf die Einstellung „Wissen ist Macht!“. Doch muss man Wissen auch einzusetzen wissen, ansonsten ist es eher nutzlos. Teilt man sein Wissen und erhältt dafür das Wissen von anderen und kann dieses auch einsetzen, führt das unausweichlich zu Erfolg.

Wissen ist das einzige Gut, das sich vermehrt, wenn man es teilt!

Im Gegensatz zu „Wissen ist Macht!“ hat es sich aber viel mehr herausgestellt, dass Wissen eher aus dem Grund nicht geteilt wird, weil man selbst gar nicht erkennt, wie wertvoll das eigene Wissen ist. Schafft man eine Umgebung, in der man sich aktiv über Wissen austauschen kann und bekommt Rückmeldung, dass das Wissen auch hilfreich war, sind wir Menschen viel mehr daran interessiert es zu teilen.

Oft fehlt es auch an Transparenz über die Expertisen anderer. Man weiß gar nicht, wer einem bei einer Problemstellung helfen kann und scheut dadurch die Fragestellung. Ein weiteres Hindernis ist die Zeitknappheit, denn zur Dokumentation von Wissen braucht man nun mal Zeit und die haben wir heutzutage so sehr komprimiert, dass man auf die Dokumentation lieber verzichtet.

Datenschutz und Sicherheit sind bei firmeninternen Wissensmanagement-Lösungen auch nicht zu vernachlässigen. In der digitalen Welt kann Wissen sehr leicht gestohlen werden. Das ist vor allem bei geheimen Informationen relevant. Schützen Sie geheime und persönliche Daten vor Dritten. Und gehen Sie nach dem Minimumprinzip: Bieten Sie nur so viel Information wie nötig und stellen Sie nur bereit, was für die spezifische Lösungsfindung notwendig ist.

Lösungsansätze

Wenn Sie eine neue Wissensmanagement-Lösung einführen möchten, gilt es folgende Punkte zu beachten:

- Beziehen Sie die User in die Auswahl der Wissensmanagement Lösung mit ein. Sie müssen das was gesucht wird, leicht finden und auch ihr Wissen einfach teilen können
- Achten Sie darauf, dass Probleme, Wissen und Lösungen diskutiert werden können. Schaffen Sie eine Plattform in dem Wissen nicht nur dokumentiert, sondern auch aktiv ausgetauscht werden kann. Beispielsweise mithilfe eines Forums, Kommentarfunktionen oder auch Blog-Einträgen
- Oft werden Informationen auf dem eigenen PC, in eigenen Ordnerstrukturen, über das Mailing-Programm und auch Einzeldokumenten abgesichert. Dieses soll am besten aber in der Wissensmanagement-Lösung landen. Daher stellen Sie eine Möglichkeit zur Verfügung, direkt in der Wissensmanagement-Lösung diese Dateien zu verwalten. Je nach Kontext kann das sogar automatisiert über ein Pull-Prinzip geschehen: Im User-Support könnten so Mailverläufe anonymisiert und automatisch hochgeladen werden, um in Zukunft ähnliche Problemstellungen schneller lösen zu können. Oder abgelegte Daten in einem bestimmten Ordnersystem werden laufend mit dem Wissensmanagement abgeglichen, um jeden Mitarbeiter im Projekt Zugriff zu gewähren.
- Bieten Sie die Möglichkeit einer individuellen Seite für jede/n Mitarbeiter/in zur Selbstvorstellung. Hier können Skills, Expertisen, Tätigkeiten und ähnliches dokumentiert werden. Ähnlich wie bei einem sozialen Netzwerk hat jede/r ein persönliches Profil, das selbst gestaltet werden kann und

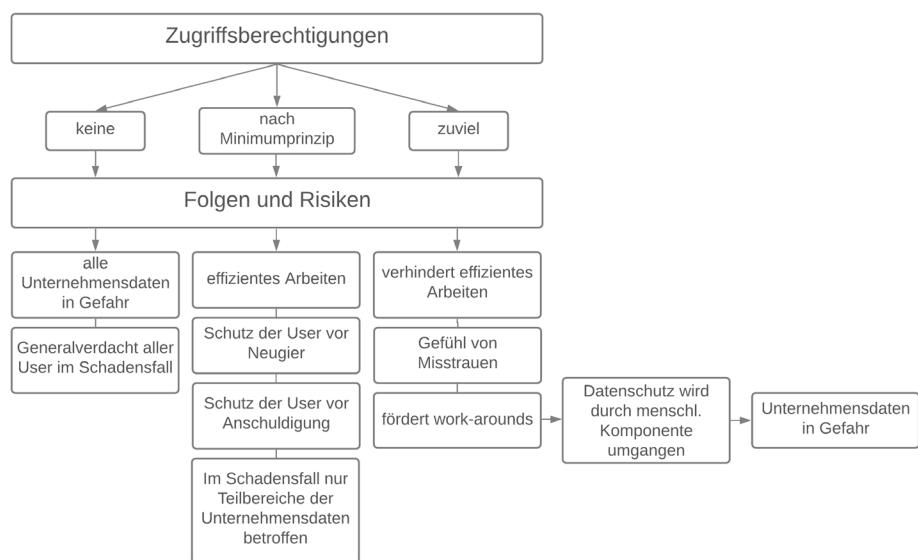


Abbildung: Folgen und Risiken von Zugriffsberechtigungen

informiert, welche Informationen essenziell sind

- Ermöglichen Sie die Bewertung von Wissen nach unterschiedlichen Kriterien, um eine schnelle erfolgreiche Suche zu ermöglichen
- Verwenden Sie das Minimumprinzip, um einen sicheren Umgang mit den Daten zu gewährleisten
- Unabhängig von der Wissensmanagement-Lösung, gilt es die Einstellung der Mitarbeiter zu individuellem Wissen zu verbessern und die Zusammenarbeit zu fördern. Ermöglichen Sie Lessons Learned-Meetings, das Gründen von Communities-of-Practice, welche sich regelmäßig austauschen und sich zur Steigerung der Wissensqualität austauschen

Haben Sie eine Wissensmanagement-Lösung eingeführt, holen Sie sich regelmäßig Feedback von den Usern, um festzustellen, ob wichtige Funktionen fehlen oder es überflüssige Funktionen gibt („Continuous Improvement“). Schnell läuft man Gefahr, dass das Wissen zwar initial gut aufbereitet ist, aber nicht mehr erweitert oder verbessert wird. ■



Bild: Trueffelpix/shutterstock.com

Quellen und weiterführende Informationen:

<https://www.youtube.com/watch?v=LfcfA4xwVrQ> (Wegweiser für ein erfolgreiches Wissensmanagement/ Experience Map – von Interculture TV)

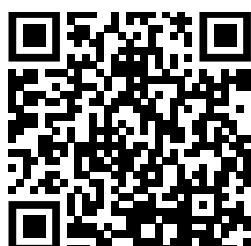
https://www.youtube.com/watch?v=aSBPvLO_yqI (Webinar – Wissensmanagement - oft versucht, oft gescheitert! | Samhammer AG)

<https://www.innolytics.de/was-ist-wissensmanagement/>

https://www.deutschlandfunk.de/die-demokratisierung-des-wissens.724.de.html?dram:article_id=100074

<https://www.re-systems.com/allgemein/essenziell-das-need-to-know-prinzip/>

Weitere Beiträge von
Andreas Steiner finden
Sie auf unserer Website



Andreas Steiner ist Consultant.

Die Schwerpunkte seiner Projekterfahrung liegen in den Bereichen Testautomation, Testdurchführung, Softwarequalitätsicherung und Requirements Engineering.

Kommunikation, Teamwork sowie vorausschauendes Arbeiten sind für ihn essenzielle Bestandteile für einen reibungslosen Projektlauf.

Ten more things

von Helena Thurner



Ten more things

Online. Proven. Professional.

- ✓ Praxisbezogene Fachvorträge zu aktuellen IT-Trendthemen
- ✓ Kompetentes, fachliches Know-how aus der Praxis
- ✓ 10 Tipps & Tricks für Ihren Arbeitsalltag
- ✓ Kostenlose Teilnahme
- ✓ Online

Wissenswertes rund um topaktuelle Themen aus den Bereichen IT Analyse, Development, Softwaretest und Projektmanagement auf jeweils 10 Punkte gebracht.

Unsere Experten lassen Sie an ihren langjährigen Erfahrungen teilhaben und geben ihre Tipps und Tricks in praxisnahen Vorträgen weiter. Holen Sie sich Ihren kostenlosen Wissensvorsprung und innovative Lösungen für Ihren Erfolg!

Warum Ten more things?

Steve Jobs zauberte am Ende seiner Keynote mit „One more thing“ noch ein letztes Ass aus dem Ärmel und begeisterte sein Publikum – wir bieten Ihnen **Ten more things**.

Das Programm 2021

Für Sie haben wir folgende spannende Fachvorträge vorbereitet:

Klemens Loschy	Last- und Performancetests	18.02.2021
Alexander Vukovic	Software Escrow	22.04.2021
Melanie Gau	Gamification	16.09.2021
Stefan Ladstätter	UX und Usability	14.10.2021
Markus Schwabeneder	KI und Machine Learning	11.11.2021

Details und Anmeldung

Weitere Informationen und die Möglichkeit zur Anmeldung bekommen Sie auf unserer Website www.SEQIS.com. Die Teilnahme findet online statt und ist kostenlos, aber die Anzahl der Plätze ist limitiert!

Wir freuen uns Sie online bei den *Ten more things* begrüßen zu dürfen! Sie auf unserer Website www.SEQIS.com. ■

**Wir organisieren
Veranstaltungen die
Mehrwert bieten.**



Helena Thurner ist Marketing Mitarbeiterin.

Von Drucksortengestaltung über Videodreh und -schnitt, klassischer Pressearbeit, Betreuung der SEQIS Onlinekanäle bis hin zur Organisation von Veranstaltungen übernimmt sie Marketing- und Kommunikationsagenden.

Besonders am Herzen liegt ihr das Schaffen neuer Ideen und Konzepte.

razzfazz.io

von Alexander Vukovic

razzfazz.io

boost results

razzfazz.io ist unsere neue Marke für individuelle Softwareentwicklung!

SEQIS feiert 2021 das 20-jährige Firmenbestehen und hat immer schon Software entwickelt. In den Jahren 2001 bis 2011 entwickelten wir mit dem KeepLiquid TestManager und der KeepLiquid TestAutomation auf Basis von Java und Javascript technisch innovative Produkte für unsere Kunden im Testbereich. Danach folgten unzählige Kundenprojekte, in denen wir individuelle Software für unsere Kunden erstellt haben. Sei es ein Simulator im Rahmen eines Lasttestprojektes oder ein eigenes Automationsframework. Dennoch haben wir uns strategisch auf den Dienstleistungsbereich insbesondere die Business Analyse, Softwaretest, Projektmanagement und Agile Coaching konzentriert.

In den letzten 20 Jahren haben sich die Möglichkeiten im Software Engineering rasant weiterentwickelt. Wir beraten viele Kunden, die diese rasante Entwicklung noch nicht nachvollzogen haben und mit Legacy Code, aufwändiger Wartung und schlechter Qualität von Legacy Software kämpfen.

Wir sind der Meinung, es ist an der Zeit, nun hier auch als Anbieter am Markt und auf Basis einer modernen Architektur auch individuelle Softwareentwicklung anzubieten, die den Ballast alter Plattformen, Architekturen und Oberflächen überwindet. razzfazz.io war damit geboren.

„razzfazz.io is faster than you can say performance“

Wir konzentrieren uns bei unserer Architektur auf Fullstack Javascript, d.h. wir entwickeln sowohl die Backend-Serverseite mit Javascript (auf Basis von Node.js), als auch das Frontend auf Basis eines Javascript Frameworks wie z. B. vue.js, React oder Angular. Damit realisieren wir Desktop Apps für Windows, Linux, Mac OSX, Progressive Web Apps (Web Apps mit Offline Funktionalität) und natürlich auch einfache Webapplikationen. Aber auch Mobile Apps zählen zu unserem Angebot, auch hier setzen wir primär auf Javascriptframeworks wie React Native oder Nativescript, entwickeln aber auch gerne native Android oder iOS Apps.

Die Konzentration auf Javascript als zentrale Programmiersprache bringt viele Vorteile für unsere Kunden mit sich:

- Weniger Entwicklungsaufwand durch Wiederverwendung zwischen Server und Client
- Nutzung der größten Platform für wiederverwendbaren open source code NPM (Node Package Manager), dadurch viel weniger Entwicklungsaufwand
- Maximale Skalierbarkeit und Performance, insbesondere bei lastintensiven Webapplikationen
- Leichte Integration von Datenbanken, REST-Services, GraphQL-APIs, oder beliebigen anderen Systemen
- Zentraler Fokus auf Qualität mit Unitests, einheitlichem automatisiertem Buildprozess und Erfahrung des Teams

Unser Team hat allerdings auch langjährige Erfahrung mit Java und .Net. Mit diesen Technologien haben wir in den letzten 20 Jahren gearbeitet. Um so klarer und fundierter konnten

Eine Auswahl unserer empfohlenen Möglichkeiten

Area	Realisierung durch / für
Backend-Serverseite	Javascript (Node.js)
Frontend	Javascript Frameworks (vue.js, React, Angular, ...)
Desktop Apps	Windows, Linux, Mac OSX, Progressive Web Apps (Web Apps mit Offline Funktionalität), Webapplikationen
Mobile Apps	Javascript Frameworks (React Native, Nativescript,...), native Android bzw. iOS Apps

wir unsere neue Architektur auf Basis von Javascript designen und in diversen Projekten weiterentwickeln.

Die perfekte Architektur ist allerdings nichts ohne perfekte Anforderungen und perfekter Qualitätssicherung. Das macht razzfazz.io so einzigartig, da wir genau diese Bereiche, die viele Entwicklungsanbieter stark vernachlässigen, durch unsere Beratungsleistung in den letzten 20 Jahren fix in unserer DNA verankert haben.

razzfazz.io wickelt Projekte agil unter starker regelmässiger Einbindung des Kundenfeedbacks ab und setzt dabei auf eine fundierte Business Analyse und Requirement Engineering, sodass für den Kunden von Anfang an klar ist, welche Lösungen er in den kommenden Iterationen übergeben bekommt. Klarerweise entwickeln sich auch die Anforderungen über die Projektdauer weiter, was im agilen Vorgehensmodell von razzfazz.io ein unterstützter und gewünschter Weg zu einer besseren Lösung für den Kunden ist.

„Das Leben ist zu kurz und Ihre Software ist zu wertvoll um keine Unit Tests zu schreiben!“

Unsere Wurzeln sind auch in der Qualitätssicherung und dem Wissen, das wir im Zuge der Beratung und Testunterstützung vieler Entwicklerteams gewonnen haben. Wir wissen, wie man richtig Qualität von Anfang an in der Software verankert. Unit Tests, Test Automation, Last- und Performancetests, Usability und Securitytests sind integraler Bestandteil unserer Entwicklungsprojekte. Eine Übersicht über aktuelle Projekte und mehr finden Sie auf unserer Homepage <https://razzfazz.io>



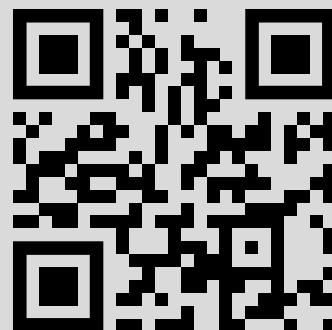
Abbildung: razzfazz.io Homepage

Wenn wir Ihr Interesse wecken konnten, besuchen Sie unsere Homepage unter <https://razzfazz.io> und abonnieren Sie unseren Newsletter. In den nächsten Wochen und Monaten werden wir sukzessive weitere Details und Infos zu razzfazz.io veröffentlichen.

Wenn Sie eine Aufgabenstellung im Hinterkopf haben, die wir für Sie mit Software lösen dürfen, kontaktieren Sie uns direkt unter +43 664 83 787 01 oder anfrage@razzfazz.io.

Wir freuen uns auf Ihre Herausforderungen! ■

www.razzfazz.io



Alexander Vukovic ist SEQIS Gründer und Chief Evangelist.

Er ist erster Ansprechpartner für alle agilen, testmethodischen und testtechnischen Anfragen. In der Praxis arbeitet er als Agile Quality Coach, Berater, Interims-Testmanager, CI-Experte und Lasttester. Mehr als 25 Jahre Beratertätigkeit führten ihn während seiner zahlreichen Projekte in die unterschiedlichsten Branchen und Länder.

Sein persönliches Motto „Es gibt keine Probleme, sondern nur nicht gefundenen Lösungen“ spiegelt sich in jedem Projekt wider.

Moco und SEQITracker

von Alexander Vukovic

2021 wird SEQIS 20 Jahre alt. Wie bei jedem Unternehmen sind unsere Strukturen über die Jahre gewachsen und haben sich laufend verbessert. So ist auch der Software Stack, den wir für unsere Backoffice Prozesse wie Sales, Marketing, Zeiterfassung, Verrechnung, Controlling, usw. einsetzen, in die Jahre gekommen.

Als qualitätsorientiertes Unternehmen die ideale Lösung für das Unternehmen selbst zu finden, ist – auch für uns keine leichte Aufgabe. So gab es für die Ablöse unterschiedliche Projektansätze:

- Ablöse durch eine komplette Eigenentwicklung
- Ablöse durch spezialisierte Einzellösungen für jeden Prozess
- Ablöse durch eine Standardlösung und Anpassung der Prozesse

Letztlich setzen wir nun auf eine Standardlösung: Moco (<https://www.mocoapp.com>) in Kombination mit einer Eigenentwicklung.

Seit 1. Juli 2020 ist das neue System nun produktiv im Einsatz und unterstützt uns bei unserer täglichen Arbeit.

Moco ist für den deutschsprachigen Markt konzipiert und deckt dabei die auch die wichtigsten Funktionen im Bereich der Zeiterfassung mit Leistungserfassung und Arbeitszeiterfassung über Web und Mobile App ab. Dies ist allerdings für die österreichische kollektivvertragliche Struktur und Rechtslage im Bereich der Arbeitszeiterfassung nicht ausreichend. Branchenspezifische Regeln wie das Führen eines Arbeitszeitsgleitzeitkonto, Überstundenpauschalen oder Regeln zur Maximaltagesarbeitszeit sind in Moco direkt nicht möglich. Aber Moco bietet eine sehr umfangreiche REST-API und Webhooks, die es externen Systemen ermöglichen, an Moco anzudocken.

Auf dieser technischen Basis hat unser razzfazz.io-Team (siehe



Unternehmensbericht 2019

Zeiterfassung Abrechnung Kundenakquise Einsatzzplanung Kontaktverwaltung Ausgaben

- ① Zeiterfassung
- ② Abrechnung
- ③ Kundenakquise
- ④ Einsatzzplanung
- ⑤ Kontaktverwaltung
- ⑥ Ausgaben

Weniger ist mehr.

Konzentration auf wesentliche Funktionen, klaren Aufbau und gute Performance. Das heißt weniger Stress und mehr Zeit für Business.

Quelle: mocoapp.com

Moco ist eine Schweizer Software für projektbasierte Dienstleister und verfolgt den Ansatz, Dinge so einfach wie möglich zu lösen. Wir haben unserer Prozesse so weit wie möglich ebenfalls vereinfacht und an die Prozesse in Moco angepasst.

razzfazz.io - Eine neue Marke für Software Entwicklung der SEQIS GmbH auf Seite 26) eine Lösung für österreichische Kunden geschaffen, die Moco einsetzen wollen, aber diverse österreichische Rechtsvorgaben erfüllen müssen.



razzfazz.io SEQITracker

Klemens Loschy
QDO

Online

Abmelden

Einstellungen

Aktueller Monat

Monatsabschlüsse

Klemens Loschy / Oktober 2020
Waiting for HR approval

Projektübersicht

QDO
Zieltermin: 30.06.2025

Oktober 2020

	ÜBERTRAG GLEITZEITKONTO	RESTURLAUB MONATSANFANG	GEPLANTER URLAUB (MONAT/GESAMT)	
	28.05 h	15 Tag(e)	0/0 Tag(e)	
01.-04.10. KW40	SOLL 15.4	HABEN 17.5	DIFF +2.1	GLEITZEIT +30.15
05.-11.10. KW41	SOLL 38.5	HABEN 42.75	DIFF +4.25	GLEITZEIT +34.4
12.-18.10. KW42	SOLL 38.5	HABEN 43.5	DIFF +5	GLEITZEIT +39.4
19.-25.10. KW43	SOLL 38.5	HABEN 42.2	DIFF +3.7	GLEITZEIT +43.1
26.-31.10. KW44	SOLL 30.8	HABEN 38.25	DIFF +7.45	GLEITZEIT +50.55

Zusammenfassung Projekte & Aufgaben

Stundensaldo	Gleitzeitkonto	Urlaubskonto
Haben Soll Ü-Pauschale Stundensaldo Oktober	Stand Vormonat Manuelle Korrektur Stundensaldo Endstand Oktober	Stand Vormonat Manuelle Korrektur Konsumiert Resturlaub (Heute/Vor. Ende Oktober)
184.2 h -161.7 h -9.5 h +13 h	+28.05 h +/- 0 h +13 h +41.05 h	15 Tag(e) +/- 0 Tag(e) -1 Tag(e) 14/14 Tag(e)

STATUS
Waiting for HR approval JIRA TICKET

Quelle: SEQIS GmbH

SEQITracker erweitert Moco um folgende Features:

- Führen eines Gleitzeitkontos pro Mitarbeiter auf Basis der Arbeitszeitaufzeichnung und des Zeitmodells in Moco
- Hinterlegung und korrekte Einberechnung einer Überstundenpauschale pro Mitarbeiter
- Unterstützung und farbige Markierung div. Arbeitszeitregeln, so dass Regelverletzungen auf einen Blick sichtbar sind
- Führen einer kompletten Monatsübersicht für jeden Mitarbeiter, der die Kombination aus Arbeitszeit, Leistungszeit und Abwesenheiten immer aktuell im Web und am Handy zeigt
- Anzeige des Gleitzeitkontos
- Anzeige des Urlaubskontos
- Abbildung eines Monatsabschlussworkflows unter Einbindung von Jira
- Abbildung eines Projektmonatsabschlussworkflows unter Einbindung von Jira
- Hinterlegung unterschiedlicher Rollen in der Projekt-/Team-

struktur, die sich über Moco nicht abilden lassen

- Gleichzeitige Erfassung von Arbeitszeit und Leistungszeit über das Beschreibungsfeld in Moco
- Geplante Features sind u.a.: Direkte Zeiterfassung auch über SEQITracker, erweiterte Reports für HR, div. prädiktive Anzeigen, die dem Mitarbeiter ermöglichen seine Arbeitszeit zu planen und viele mehr...

Wenn Sie Interesse am SEQITracker oder einer eigenen individualisierten Erweiterung für Ihre Moco-Instanz haben, kontaktieren Sie uns bitte unter anfrage@razzfazz.io. Wir entwickeln diese gerne auf Basis unserer eigenen Lösung für Sie, oder stellen Ihnen eine Installation unserer Lösung out of the box zur Verfügung. ■



Alexander Vukovic ist SEQIS Gründer und Chief Evangelist.

Er ist erster Ansprechpartner für alle agilen, testmethodischen und testtechnischen Anfragen. In der Praxis arbeitet er als Agile Quality Coach, Berater, Interims-Testmanager, CI-Experte und Lasttester. Mehr als 25 Jahre Beratertätigkeit führten ihn während seiner zahlreichen Projekte in die unterschiedlichsten Branchen und Länder.

Sein persönliches Motto „Es gibt keine Probleme, sondern nur nicht gefundenen Lösungen“ spiegelt sich in jedem Projekt wider.

Holacracy¹ - Scrum bis in die Führungsetage

von Melanie Gau

Inspiriert von agilen Methoden entwickelte Brian Robertson ein Organisationsprinzip, das auf kollektive Intelligenz, Flexibilität und Selbstorganisation setzt. Statt starren Hierarchien schafft Holacracy einen sich selbst verwaltenden Organismus, der die Herausforderungen von sowohl Industrie- als auch Gesellschaft 4.0 meistern möchte.

Der Mensch als Sensor

Wie ein sehr feinfühliger Sensor ist jeder Mitarbeiter in Organisationsstrukturen mehr als seine/ihre Job-Description. Jede/r hat nämlich durch sein/ihr individuelles Expertenwissen und Erfahrung die Gabe, Veränderungen und Spannungen zu spüren. Jede/r ein wenig anders, jede/r in unterschiedlichen Bereichen. Und genau diese Gabe war für Brian Robertson, den Erfinder von Holacracy, 2007 der Anstoß, nach einer Möglichkeit zu suchen, wie diese Fähigkeit in einer Firmenstruktur genutzt, statt unterdrückt werden kann, um den Herausforderungen von beschleunigten Märkten und einer zunehmend komplexer werdenden Welt gerecht zu werden.

In seiner Softwareentwicklungs firma Ternary Software waren agile Arbeitsweisen bereits eingeführt und so versuchte Robertson zuerst auch die Managementstruktur so zu adaptieren, dass jede/r Einzelne mehr direkten Einfluss und Gehör bekam. Er stellte jedoch fest, dass innerhalb der vorhandenen Machtstrukturen diese „Befreiung“ nicht greifen konnte, da die bestehenden Hierarchien – egal wie flach sie waren – systemisch auf einem Kontrollfluss von oben nach unten („Predictive Control“) beruhten.

Der Begriff *Holacracy* ist eine Wortschöpfung inspiriert von Griechisch *holos* (όλος - „ganz“) – auch mit einem Kreis symbolisiert – und *kratia* (κρατία - „Herrschaft“), analog zu z.B. Democracy. Und vielleicht erinnert sich der eine oder andere auch noch an Arthur Koestlers „The Ghost in the Machine“ (1967), wo er mit dem Begriff Holarchie eine Organisation beschreibt, die sehr ähnlich aus unabhängig handelnden Einheiten aufgebaut ist.

Als Metapher verwendet Robertson den menschlichen Körper, in dem jede Zelle eigenständig und durch dynamische Interaktion sowohl untereinander als auch mit größeren Strukturen, wie z.B. Organen, funktionieren. *Holacracy* imitiert diesen organischen Prozess durch ein strukturiertes, sich ständig im Austausch befindendes Netz an autonomen Funktionseinheiten („Adaptive Control“), die nicht durch starre, vorgegebene Barrieren in ihrem Handeln („Reactive Control“) eingeschränkt sind.

Wie funktioniert also Holacracy?

Holacracy baut auf einem absolut transparenten Set an Richtlinien auf, die in der *Holacracy Constitution*² niedergeschrieben sind. Diese „Verfassung“ beschreiben die Grundprinzipien und deren Umsetzung.

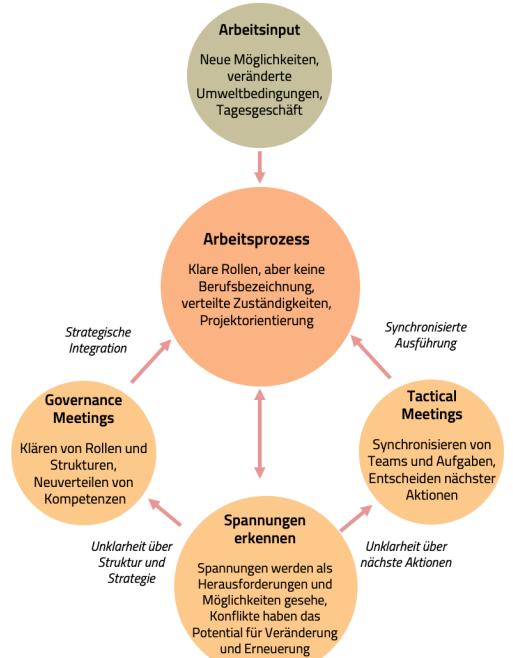


Abbildung 1: Workflow eines *Holacracy Circles*.

Ein zentraler Aspekt von *Holacracy* ist, dass Mitarbeiter flexibel verschiedene Aufgaben (*Rollen*) übernehmen und sich frei zwischen Teams (*Kreisen*) bewegen können.

Wer ist in unserer Firma eigentlich genau für eine Neuerung verantwortlich?

Die Rolle von Rollen

Eine Person ist also nicht an die vertraglich festgelegten Inhalte und Aufgaben gebunden, die mit den tatsächlichen Tätigkeiten der täglichen Praxis selten übereinstimmen. Sie kann nach den gerade benötigten Fähigkeiten oder Kenntnissen eine oder mehrere, i.d.R. kleinere, Rollen ausfüllen, die für die Organisation und die aktuelle Aufgabe nutzbringend sind.

Erspüren und Umsetzen – statt Frustration und Bürokratie.

Der Inhaber einer Rolle hat die Befug-

nis, Entscheidungen eigenständig zu treffen ohne eine „Absegnung von oben“ dafür einholen zu müssen. Dies zeigt zum einen glasklar, wer die Kontrolle und damit die Verantwortung für welche Handlung trägt und eröffnet zum anderen die Möglichkeit, eigenverantwortlich auch unkonventionelle, aus dem eigenen Expertenwissen heraus als sinnvoll erkannte Lösungen zu implementieren. Gerade komplexe Problemstellungen erfordern mitunter innovative oder mutige Maßnahmen sowie eine zeitnahe Umsetzung, wie es in starren, hierarchischen Strukturen kaum möglich ist. Diese unmittelbare Identifikation einer Rolle mit ihrem Arbeitsfortschritt und Ergebnis und die klare Definition von Verantwortung, Autorität, Funktion und Kontrollbereich führen insgesamt zu einer besseren Kontrolle der Prozesse. Zusätzlich steht der/die Verantwortliche dadurch auch psychologisch völlig hinter der Aufgabe und erntet selbst die Lorbeeren für seine/ihre Erfolge.

Holokratische Kreise

Die thematischen Domänen einer Organisation sind in Kreise gruppiert. Jeder Kreis etabliert seinen eigenen Strukturprozess in sogenannten **Governance Meetings**, in dem Rollen und Ausgestaltung etabliert werden. So ergibt sich eine organische Struktur aus kleineren und größeren Kreisen, die wiederum von übergreifenden Kreisen umschlossen sind. Zur nahtlosen Kommunikation werden inhaltlich verbundene Kreise jeweils durch einen Vertreter aus dem anderen Kreis repräsentiert. Jeder Kreis hat vier festgeschriebene Rollen, die eine funktionierende Selbstorganisation gewährleisten, aber es können nach Bedarf weitere Rollen hinzugefügt werden.

Schluss mit Meetings in denen nichts passiert!

Der operative Prozess wird in sogenannten **Tactical Meetings**⁴ geregelt. Sie folgen einem festgelegten Prozess, der sehr straff durchstrukturiert ist und keine Redundanzen zulässt. Hier werden auch (mögliche) Barrieren sofort angegangen und Lösungsmaßnahmen erarbeitet. Weiterhin wird konkret geplant, welche nächsten Schritte wie und wann umgesetzt werden sollen und welches Ziel damit erreicht werden muss.

Dieser strukturierte Prozess, bei dem alle relevanten Rollen gehört werden, ähnelt bekannten agilen Development- oder Projektmanagementprozessen und macht sich dieselben Vorteile zunutze, u.a. die Möglichkeit rascher, dynamische Veränderungen, feste Rahmenbedingungen (Zeit, Anforderungen) und ständigen, direkten Wissens- und Informationsaustausch. *Holacracy* ist also eine elegante Erweiterung der „Agile Family“ für alle Domänen eines Unternehmens, wie z.B. auch typische Bereiche „klassischen“, hierarchischen Managements wie Marketing, Budgeting oder Strategieentwicklung. So bleibt die Organisationsstruktur flach und durchlässig und es greifen alle Teile als ein einziger Organismus nahtlos ineinander. Probleme konkurrierender Strukturen, wie z.B. eine agile F&E-Abteilung als Insel in einer hierarchischen, silo-artigen Mutterorganisation, werden dadurch gelöst.

“If it doesn’t feel weird, you’re doing it wrong” – Brian Robertson

Holacracy ist also das radikale Adaptionieren agiler Prinzipien in allen Organisationsbereichen für jeden Mitarbeiter einschließlich aller klassischen Führungsebenen! Dies geht so weit, dass beim Übertritt in die Holacracystruktur von den Eigentümer/innen die absolute Führungsgewalt zugunsten der Konstitution formal abgegeben wird!

Zum Risikomanagement baut *Holacracy* als Selbstorganisationskultur auf ein System von Checks & Balances. Prinzipiell gilt: Es ist alles erlaubt,

was nicht in die Domänen von anderen (in ihren definierten Rollen) eingreift oder gegen die Verfassung verstößt. Falls sich Entscheidungen trotz aller systemischen Absicherungen als Fehler herausstellen, gilt hier wie im Agilen: Fail fast! – kombiniert mit sofortiger Kurskorrektur. So können sich auch wirklich innovative Lösungen bewusst umgesetzt und weil Fehler früh im Prozess erkannt werden, ist eine nötige Korrektur mit den geringsten Kosten und Risiken verbunden.

Softwareunterstützung für *Holacracy*?

Holacracy profitiert ungemein von Softwareunterstützung: Durch diese werden die Prozesse und deren Einhaltung vereinfacht, visualisiert und automatisiert. Hier können alle Strukturen, z.B. Rollen, Kreise, Meetingformate, etc. benutzerfreundlich abgebildet und für die Beteiligten in Echtzeit bereitgestellt werden. Es gibt eine breite Auswahl an Tools und die prominentesten Vertreter sind GlassFrog (das auch z.B. dwarfs and Giants⁵ verwendet) und holaSpirit⁶.

Keine Manager, keine Probleme?

Holacracy in der Praxis

Holacracy braucht zunächst einmal eine passende Unternehmenskultur mit dem dazugehörigen Mindset. Die radikale Übertragung von Verantwortung und Entscheidungsbefugnissen an alle Mitarbeiter/innen verlangt eine kompromisslose Bereitschaft sowohl von „oben“, als auch von „unten“ – um Vertrauen aufzubauen und loszulassen respektive sich in dem nötigen Maße einzubringen und proaktiv an den Geschicken der Organisation mitzubestimmen. Wer also lieber von oben gesteuert wird oder sich mit der nötigen flexiblen und proaktiven Arbeitsweise schwertut, wird sich in einer *Holacracy* nur bedingt wohl fühlen.

“Any critique talking about lack of structure is not understanding what

Holacracy actually is and does.” – Brian Robertson

Ein verbreiteter Kritikpunkt ist die mangelnde Struktur in der *Holacracy*. Tatsächlich gibt es sogar mehr Struktur als bei traditionellen Systemen. Der entscheidende Unterschied ist aber, dass sie dynamisch anpassbar ist und sich in und mit der Organisation weiterentwickelt.

“Holacracy doesn’t empower people, it empowers processes.” – Jürgen Appelo

Ein weiterer Kritikpunkt ist gerade die strikte, kompromisslose Einhaltung von Prozessen, die diszipliniert umgesetzt werden müssen, dass das System funktionieren kann. Doch gerade dieser vorgegebene Rahmen eröffnet einen geschützten Raum, innerhalb dessen freie Entfaltung ermöglicht wird, absoluter Fokus auf eine straffe Abwicklung der Inhalte gerichtet werden kann und „Politisieren“ unterbunden wird.

„Holacracy exposes you to reality.“ – Brian Robertson

Darüber hinaus ist *Holacracy* ein faszinierendes, disruptives (Um-) Denkmodell, dass jeder Organisation zumindest als „Sandkasten“ ans Herz gelegt werden kann. Durch das Hineindenken, Umsetzen und Probieren der Methode, z.B. initial als Pilotprojekt in einem Unternehmensteil, können völlig neue Wege der Entscheidungsfindung und Arbeitsweise erlebt und erlernt werden. *Holacracy* wird so zu einer andernfalls unanzapfbaren Quelle für tiefe Erkenntnisse über die eigene Organisation, Selbstorganisation, Innovation und neues Miteinander.

“Any company designed for success in the 20th century is doomed to failure in the 21st.” – David S. Rose

Holacracy in der Zukunft

Der Komplexitätsgrad unserer vernetzten, globalisierten, hoch technologisierten Welt wird zukünftig weiter ansteigen und althergebrachte Formen der Organisationsführung, besonders in Branchen an der Frontlinie von Innovation und Technologie stoßen zunehmend an ihre Grenzen, um im Wettbewerb zu bestehen. Parallel beobachten wir einen Zuwachs von psychosozialen Belastungen, die durch Entfremdung und Druck in der Arbeitswelt gefördert werden. *Holacracy* hat hier ein unglaubliches Potential in der Zukunft eine systemische Lösung schlechthin für beide Herausforderung zu werden.

Folgen Sie uns gerne auf **XING** und **LinkedIn**

XING: www.xing.com/companies/seqisgmbh

LinkedIn: www.linkedin.com/company/seqis-gmbh

Quellen und weiterführende Informationen:

¹ <https://www.holacracy.org>

² In der aktuellen Version 4.1 von 06/2015 (hier Vers. 4.1.4.DE https://www.dwarfsandgiants.org/wp-content/uploads/2015/11/Holacracy-Verfasung-v4_1_4_DE.pdf)

Die noch für dieses Jahr angekündigte Version 5.0 ist als 5.0 beta bereits zum Testen auf GitHub verfügbar:

<https://github.com/holacracyone/Holacracy-Constitution/blob/v5.0-beta2/Holacracy-Constitution.md> und in GlassFrog implementiert. Sie beinhaltet viele nützliche Verbesserungen und Verschlankungen und ist übersichtlicher strukturiert, was sie als Referenzwerk in jeder holokratischen Organisation noch einfacher in der täglichen Praxis verwendbar macht. Außerdem bietet diese neue Version eine Struktur in 6 Modulen an, die eine leichtere Anpassung an ein individuelles Unternehmen, sowie einen graduellen Umstiegsprozess erleichtern.

³ Für weiterführende Erkenntnisse zur optimalen Entscheidungsfindung in verschiedenen Kontexten (geordnet – kompliziert – komplex – chaotisch – konfus) empfehlen wir „Das Cynefin Framework: komplex oder kompliziert?“ Seite 16.

⁴ Zu beiden Meeting-Typen stehen Moderationskarten zur Verfügung (in deutscher Sprache verfügbar unter <https://www.dwarfsandgiants.org>).

⁵ Deutschsprachiger Premiumprovider von Holacracy, siehe auch das Interview mit Björn Rabethe auf Seite 33.

⁶ <https://www.glassfrog.com>, <https://www.holaspirt.com> – Beide als Gratisversion oder günstiger Premiumvariante (<10€/Monat) browser-based oder als App verfügbar und mit Integration für eine Vielzahl an Drittanbieter-SaaS.

Robertson, Brian J.: *Holacracy: The New Management System for a Rapidly Changing World*. 2015

Geniale Ressourcensammlung von Medium: <https://medium.com/@chrcowan/introducing-the-holacracy-practitioners-guide-3ea6df523623>

Umfangreiche deutschsprachige Quelle: <https://www.dwarfsandgiants.org/we-share>

Interview mit Björn Rabethge Evolutionary Catalyst bei *dwarfs and Giants*

Datum: 29. September 2020

Ort: Mödling, Niederösterreich

Melanie Gau, SEQIS: Ist *Holacracy* also eine Art „Basisdemokratie“ für Unternehmen?

Björn Rabethge, dwarfs and Giants: Nein, sie funktioniert gerade *nicht* wie Basisdemokratie: *Holacracy* beruht *nicht* auf Konsens, sondern auf „Konsent“. Der entscheidende Unterschied ist, dass *nicht* alle einverstanden sein müssen und es dadurch *keine* „verwässerten“ Kompromisse gibt. Vielmehr darf alles umgesetzt werden, solange es nicht in eine fremde Domäne eingreift oder die Verfassung verletzt. Begründete Einwände sind zwar möglich und erwünscht, aber ein Entschluss kann trotzdem umgesetzt werden! Wer einen Einwand hat, ist in der Folge *verpflichtet* gezielt mitzuwirken, dass dieser Einwand bei der Umsetzung integriert wird. *Holacracy* verhindert damit auch endlose Diskussionen und unnötige Machtspiele.

Melanie Gau, SEQIS: *Holacracy* als evolutionärer Prozess: Welche Veränderungen bringt Version 5.0?

Björn Rabethge, dG: Die neue Verfassungsversion ist das Produkt der Erfahrungen aus dem praktischen Einsatz von *Holacracy* vieler Jahre. Sie ist reifer und noch stärker zu einem wirklich praxisbezogenen „roten Faden“ kondensiert, der sehr benutzerfreundlich und übersichtlich als permanenter Begleiter von allen Prozessen, Meetings und Fragen zur Verfügung steht. Eine der wichtigsten Neuerungen ist sicherlich die Zerlegung der Strukturen in Module, die für eine individuellere Anpassung an die jeweilige Organisation, sowie einen

stufenweisen Übergangsprozess zu *Holacracy* kombiniert werden können.

Melanie Gau, SEQIS: Was kann einen Umstiegsprozess am meisten gefährden?

Björn Rabethge, dG: Eine der größten Herausforderungen stellt in der Praxis sicher das Umdenken der beteiligten Akteure dar. Viele Arbeitnehmer und vor allem Führungskräfte sind *in* und *für* die Arbeit in hierarchischen Systemen ausgebildet worden und so kann es schwer fallen sich in neue Rollen einzufinden. Nicht zuletzt, da dies gerade für Führungspositionen als Macht- und Prestigeeinbuße empfunden werden kann. Wenn *Holacracy* dann nur wie ein „Feigenblatt“ fungiert und die Führung nicht wirklich bereit ist ein neues Führungsverständnis zu entwickeln, ist eine konsistente Umsetzung gefährdet.

Melanie Gau, SEQIS: Wie funktioniert dann Leadership in *Holacracy*?

Björn Rabethge, dG: Hier werden alle zu *Führenden und Geführten gleichermaßen*. Dies geht natürlich mit mehr Verantwortung einher, hat aber umgekehrt den klaren Vorteil, dass jede/r selbst etwas verändern kann und dafür die unmittelbare Bestätigung bekommt. Und hier gilt es für die Eigentümer und Führungskräfte eine neue, vertrauensvolle Haltung zu entwickeln und loslassen können. So kann Führen in einem beratenden und unterstützenden Verständnis gelebt werden, das eine Kultur gegenseitiger Förderung der Verbesserung eigener Fähigkeiten etabliert und die holokra-



tische Organisation *als Ganzes* nach vorne bringt.

Melanie Gau, SEQIS: Psychologische Vorteile: Bringt *Holacracy* mehr Gleichberechtigung? - Gerade auch Frauen melden zurück, dass sie *Holacracy* schätzen, weil sie sich mehr gehört fühlen.

Björn Rabethge, dG: *Holacracy* hat einen Gender-Aspekt, bzw. - genderneutral formuliert – profitieren allgemein Menschen, die sich – auch im Entscheidungsprozess – weniger gerne in den Vordergrund stellen. Durch den klar definierten Entscheidungsprozess, in dem alle Perspektiven gehört werden, ist garantiert, dass sie zu Wort kommen! Dadurch ist es auch einfacher für introvertierte Menschen, die sich schwerer tun, sich ihren Raum zu nehmen.

Melanie Gau, SEQIS: Gibt es Organisationen, die besser für *Holacracy* geeignet sind?

Björn Rabethge, dG: *Holacracy* scheint im digitalen Umfeld leichter umsetzbar zu sein, weil hier *Agilität* vielfach bereits bekannt ist. Hier muss also das Mindset von Selbstverwaltung, dynamischer Anpassung und Kommunikation in interdisziplinären Teams nicht von Grund auf neu gelernt werden. Für Neugründungen ist *Holacracy* geradezu prädestiniert, weil nicht einmal ein Umstellungsprozess notwendig ist.

Man kann sagen, je agiler eine Organisation bereits ist, desto einfacher ist der Umstieg. Aber prinzipiell ist *Holacracy* für jede Organisation geeignet, wenn sie mit der nötigen Konsequenz und in einem verantwortungsvollen, die Mitarbeiter/innen ins Zentrum stellenden Prozess eingeführt wird.

dwarfs and Giants

Die systemische Organisationsberatungsfirma *dwarfs and Giants* mit Hauptsitz in Wien basiert seit ihrer Gründung auf Selbstorganisation und *Holacracy*. Sie sieht sich selbst auch als Laboratorium für den Weg ins



dG_Nextland
<https://www.dwarfsandgiants.org/web/nextland-the-map>

#Nextland – *The Future of Work*⁷ ebnet, d.h. Management wirklich neu denkt und Strategien entwickelt und an sich selbst validiert, um die Transition in die Arbeitswelt der nahen Zukunft mit all ihren Herausforderungen zu meistern. *dwarfs and Giants* ist Premium-provider im deutschsprachigen Raum und zeichnet verantwortlich für die deutsche Übersetzung des offizielle *Holacracy*-Materials. ■

Infotipp: #Nextland – die interaktive, graphische Visualisierung der Zukunft und wie man sie erfolgreich erreichen kann.

Sie haben die letzten Ausgaben der Quality-News verpasst?

Lesen Sie hier nach!



Melanie Gau ist Consultant.

Ihr Steckenpferd ist die reibungslose Realisierung von IKT Projekten, und zwar von der Geburt einer innovativen Idee bis zur erfolgreichen Markteinführung. Sie verbindet technisches Know-How mit enger Kommunikation mit allen an einer gelungenen Umsetzung Beteiligten, stets mit Blick auf angemessenes, wirtschaftsorientiertes Management. Sie hat eine ganz persönliche Liebe zu agilen Methoden und der Aktivierung von Potentialen für die Herausforderungen im Zeitalter von Industrie 4.0.

Agile Circle - what's next?

von Helena Thurner



Agile Circle

Der Agile Circle ist der Treffpunkt für UmsetzerInnen digitaler Innovatio-nen.

Hier können sich agile Führungs-kräfte, SoftwareentwicklerInnen und Softwareentwickler, TesterInnen und Tester, Product Owner und Scrum Master treffen, austauschen, inspirie-ren und neue Ideen entwickeln.

Unser Fokus

Unsere gesamte Wirtschafts- und Geschäftswelt befindet sich im ra-santesten Umbruch seit Menschen-gedenken. Agil auf die neuen Her-ausforderungen zu reagieren, darauf kommt es an.

Unser Programm

Der Agile Circle bringt die unter-schiedlichsten Blickwinkel auf die agi-le Software Entwicklung zusammen: das Management, die Entwicklung, sowie das Testing & die Delivery.



Bild: Rido/Shutterstock.com

Rückblick 2020

Der erste Agile Circle ONLINE war ein voller Erfolg: Vier Impulsvorträge, drei Workshops, eine Podiumsdiskus-sion und eine Unconference gingen erfolgreich am 28. Mai 2020 über die Bühne. Wir bedanken uns an alle an-wesenden Personen, die diesen Event so großartig gemacht haben. Danke auch für die viele positiven Rückmel-dungen!

Ausblick 2021

Bestärkt gehen wir den Weg weiter. Und so darf ich an dieser Stelle auch den nächsten Event ankündigen: Den Agile Circle ONLINE 2021

Wann: 28.01.2021

Wo: Online

Thema: Game Changer 2021

Weitere Informationen
finden Sie auf unserer
Website



www.agilecircle.org

Am **28. Jänner 2021** dreht sich alles um das Thema Game Changer 2021. Genaugenommen werden wir hier die Frage behandeln: Was sind die gro-ßen Themen, die in der Digitalisierung relevant sind bzw. in Kürze werden. Seien Sie daher gespannt und mar-kieren Sie sich schon jetzt den 28. Jänner 2021 in Ihrem Kalender. Neh-men Sie teil, ganz bequem und sicher von zu Hause aus.

Wir freuen uns, Sie im Jänner online zu treffen! ■



Helena Thurner ist Marketing Mitarbeiterin.

Von Drucksortengestaltung über Video-dreh und -schnitt, klassischer Pressearbeit, Betreuung der SEQIS Onlinekanäle bis hin zur Organisation von Veranstaltungen übernimmt sie Marketing- und Kommunikationsagenden.

Besonders am Herzen liegt ihr das Schaffen neuer Ideen und Konzepte.

Der ROI von Usability

von Stefan Ladstätter

Bis vor kurzer Zeit wurde gute Usability noch als „nice to have“ betrachtet. Im besten Fall war es ein Unterscheidungsmerkmal zwischen konkurrierenden Produkten. Doch die Zeiten haben sich geändert. Kunden erwarten sich gute UX (= User eXperience). Gute Usability erleichtert die Akquise von neuen Benutzern und erhöht das Vertrauen der Kunden in das Produkt. Kurzum, es erhöht die Bereitschaft, mehr zu investieren. Software mit guter Usability erhöht die Kundenzufriedenheit und hilft, Kosten für Support und Schulungen einzusparen.

Dennoch wird UX im Software-Entwicklungsprozess immer noch stiefmütterlich behandelt. Die Argumente dafür kreisen immer wieder um das gleiche Thema: Zu wenig Zeit, zu wenig Geld. Wie sich zeigt, lassen sich beide Argumente leicht als Ausreden enttarnen, denn gute Usability verbessert nicht nur unser Produkt. Gutes UX-Design spart Geld und Zeit, weil Fehler bereits zu einem frühen Zeitpunkt vermieden werden, und erhöht den Gewinn, weil das resultierende Produkt attraktiver wird.

Der menschzentrierte Gestaltungsprozess

User Experience ist die Wissenschaft (und Kunst), interaktive Systeme auf eine Weise zu Gestalten, dass sie leicht zu benutzen sind und die Erwartungen, Bedürfnisse und Ziele der Benutzer erfüllen.

Durch gezielte Aktivitäten kann die Usability und User Experience von Software, Webapps und Devices enorm gesteigert werden. Dies klappt jedoch nur, wenn der Benutzer im Zentrum des Gestaltungsprozesses steht. Der ISO-Standard 9241-210 („Human-centered design for interaction“) beschreibt die Grundprinzipien für ein benutzerorientiertes Design.

tive systems“) zählt dafür die folgenden Aktivitäten auf:

1. Menschzentrierten Gestaltungsprozess planen
 2. Nutzungskontext verstehen
 3. Nutzungsanforderungen spezifizieren
 4. Gestaltungslösungen entwickeln
 5. Gestaltungslösungen evaluieren

diese Behauptung anhand einiger Beispiele.

A - Entwicklungskosten einsparen

Fehler im Erfassen der Nutzungsanforderungen führen zu erheblicher Mehrarbeit. Laut einer Studie von IEEE („Why Software Fails“) verschwenden Entwickler 50% ihrer Zeit

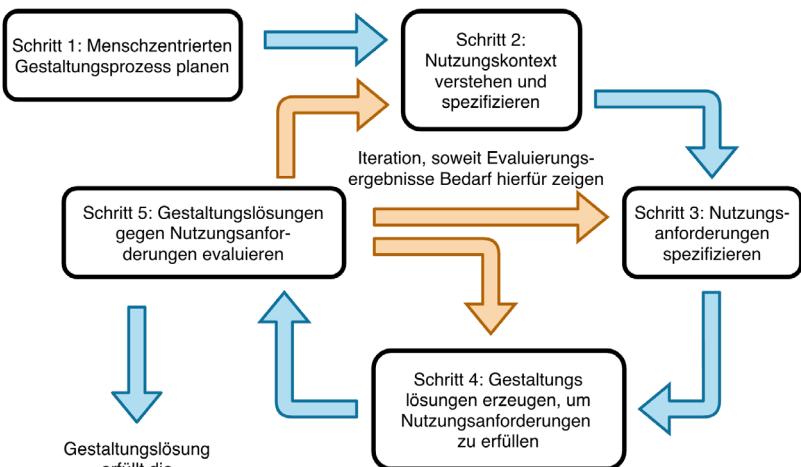


Abbildung: Der menschzentrierte Gestaltungsprozess nach DIN EN ISO 9241-210

Dieser Prozess ist iterativ. Die Evaluierung der Designs beginnt bereits in einem möglichst frühen Entwicklungsstadium. Je nach Erkenntnissen werden die Schritte 2 bis 4 erneut durchgeführt, bis die Evaluierung (Schritt 5) zeigt, dass die Gestaltungslösung die Nutzungsanforderungen erfüllt.

Ist es das wert?

Egal welches Projekt, Zeit und Geld sind immer knappe Ressourcen. Und egal wie lobenswert gute UX wäre, zusätzliche Prozessschritte kosten Zeit und Geld. Diese Betrachtungsweise ignoriert jedoch die Kosten von mangelhaften Usability-Design und -Testing bzw. den Gewinn von gutem UX. Laut einem Artikel von Forbes (2015) ist der ROI von UX \$100 pro investierten \$1. Folgend prüfen wir

damit, Arbeiten zu verrichten, die vermeidbar wären. Darunter fallen auch Änderungen an der Funktionalität der Software, die durch die richtige Erfassung der Nutzeranforderungen vermeidbar gewesen wären. Es ist allseits bekannt, dass die Behebung von Fehlern umso kostspieliger sind, je später sie entdeckt werden. Die Behebung eines Fehlers in einer Anforderung kostet einen Bruchteil an Zeit und Geld im Vergleich zu einem Fehler, der erst spät in der Entwicklung oder gar erst nach Auslieferung eines Produkts entdeckt wird.

Berechnung des ROI von UX als Kostenersparnis:

(Zeit pro Sprint, die für Fehlerbeseitigung aufgewendet wird) x (Anzahl der Entwickler) x (Stunden-

satz) x (Anzahl Sprints pro Jahr) = Ersparnis pro Jahr

Beispiel:

$$(20 \text{ Stunden pro Sprint}) \\ \times (10 \text{ Entwickler}) \times (100 \text{ EUR/h}) \times (26 \text{ Sprints}) = € 520.000 \text{ Ersparnis pro Jahr}$$

Nicht nur die Fehlerbehebung kann kostspielig sein, auch die Folgekosten müssen betrachtet werden. Manche Folgekosten sind Opportunitätskosten. Vergeudete Ressourcen für die Fehlerbehebung hätten auch gewinnbringender investiert werden können. Aber auch der potenzielle Vertrauens- und Gesichtsverlust sowie Folgeschäden beim Endanwender müssen hierbei berücksichtigt werden. Die Einhaltung des menschzentrierten Gestaltungsprozesses stellt sicher, dass die Anzahl der Fehler auf ein Minimum reduziert wird, und hilft somit nicht nur, Kosten zu sparen sondern ist auch eine wirksame Methode, Risiken zu minimieren.

B - Kundennutzen erhöhen

Zufriedene Kunden sind die wertvollsten Kunden. Ein wesentlicher Weg dorthin ist die Reduktion von Benutzerfehlern. Die Auswirkungen von Benutzerfehlern sich monetär berechnen, und damit auch das Einsparungspotenzial.

Berechnung des ROI von UX als Ergebnis von verringerter Fehlerrate bei den Endbenutzern:

$$(\text{Anzahl der Fehler}) \times (\text{Zeit zur Behebung}) \times (\text{Anzahl Mitarbeiter}) \times (\text{Stundensatz}) = \text{Ersparnis pro Woche}$$

Beispiel:

$$(6 \text{ Fehler/Woche}) \times (0,5 \text{ h}) \\ \times (50 \text{ Angestellte}) \times (50 \text{ EUR/h}) = € 3.750 \text{ Ersparnis pro Woche (bzw. € 195.000 Ersparnis pro Jahr)}$$

Zufriedenheit entsteht auch durch

gesteigerte Produktivität. Verbesserte Usability kann zu einem Performance-Gewinn führen.

Berechnung des ROI von UX als Ergebnis von Produktivitätssteigerung:

$$(\text{Zeit gespart}) \times (\text{Stundensatz}) \times (\text{Anzahl Mitarbeiter}) = \text{Ersparnis pro Woche}$$

Beispiel:

$$(4 \text{ h pro Mitarbeiter}) \times (35 \text{ EUR/h}) \times (50 \text{ Angestellte}) \\ = € 3.500 \text{ Ersparnis pro Woche (bzw. € 182.000 Ersparnis pro Jahr)}$$

Ein lohnendes Investment

Die Umstellung auf einen menschzentrierten Gestaltungsprozess ist ein laufendes Investment, das sich jedoch rasch amortisiert. Es gibt zahlreiche KPIs, die nachweislich im Zusammenhang mit Usability stehen: Conversion Rates, Anzahl der Registrierungen, weniger Abbrüche im Checkout, weniger Support-Anfragen, reduzierter Training-Aufwand, höheres Engagement, Zeitersparnisse für Entwickler und Benutzer und nicht zuletzt reduzierte Fehlerrate, sowohl im Endprodukt als auch in der Benutzung. Egal welcher dieser Faktoren zur Berechnung des ROI von UX herangezogen wird, gute UX lohnt sich! ■

Trainings & Schulungen

Haben Sie schon Ihre nächste Schulung geplant?

Alle Informationen finden Sie auf unserer Website:

www.SEQIS.com



Stefan Ladstätter ist Consultant.

In seiner mehr als 20-jährigen Tätigkeit als Entwickler, Requirements Engineer, Usability-Spezialist und Projektmanager hat er Einblick in die unterschiedlichsten Aspekte der Umsetzung von IT-Projekten gewonnen und kann dabei auf einen tiefen Erfahrungsschatz im klassischen und agilen Umfeld vorweisen.

Er sieht erfolgreiches Projektmanagement als Dienst an allen Projektbeteiligten – Auftraggeber, Entwicklungsteam und Anwender.



KI & Software Security

von Markus Schwabeneder

Der Einsatz von künstlicher Intelligenz (KI) hilft schon jetzt, auf effiziente Art und Weise Sicherheitslücken zu schließen und Cyberangriffe zu vereiteln. Auf der anderen Seite sollte man sich nicht zu sehr auf die KI verlassen, manches steckt noch in den Kinderschuhen und benötigt noch Verfeinerungen. Aber auch die Angreifer bedienen sich dieser Methoden und schaffen so neue Angriffsszenarien. Ich will hier einige Chancen und Herausforderungen ohne Anspruch auf Vollständigkeit aufzeigen.

Chancen

Verbesserte statische Codeanalyse

Künstliche Intelligenz sorgt dafür, dass Programmcode tiefer und genauer analysiert werden kann, ohne dass er wirklich ausgeführt werden muss. So werden Fehler schon während der Entwicklung gefunden und aufgezeigt und bedeutet enorme Kostenersparnis.

Analyse von Userverhalten

Viele Cyberattacken könnte man abwehren, wenn man die Usereingaben und -aktionen überwacht und analysiert. Dabei stößt man aber auf diverse Probleme. Zum einen ist das sehr aufwendig, zum anderen auch datenschutzrechtlich problematisch. Beide Probleme können mittels KI dennoch angegangen werden. Es wird dabei ermöglicht, dass die Checks schnell erfolgen, anonymisiert durchgeführt werden und die gesammelten Daten nur im Alarmfall gespeichert und aufbewahrt werden.

Analyse von Datenströmen

Auch hier kann die KI helfen, Unregelmäßigkeiten und Schwachstellen zu entdecken. Aktuelle Firewalls und Software zum Prüfen von E-Mailverkehr machen jetzt schon Gebrauch

von KI. Auch hier kommt unter Umständen das Datenschutzhema zum Tragen, es ist aber mit Sicherheit eher vertretbar, dass solche Kommunikation nach objektiven Kriterien von einer KI beurteilt wird, als dass man menschlichem Securitypersonal Zugriff auf den Inhalt solcher Datenströme gewährt (abgesehen von den wahrscheinlich enormen Datenmengen, die da auf das Securitypersonal zukommt).

Automatisierte Penetrationstests

Der Einsatz von KI in Penetrations- tests ist mittlerweile State-of-the-Art. Hier gibt es zwei Herangehensweisen. Die eine erfolgt nach einem Black-Box-Verfahren. Hierbei gibt man lediglich die Adresse des Servers bzw. der zu testenden Anwendung an und die KI sucht selbstständig nach Sicherheitslücken und Durchbruchsstellen. Es gibt aber auch White-Box-Varianten. Hierbei wird der KI auch Zugriff auf den Server selbst gegeben, sodass die KI Informationen sammeln kann, die der Angreifer nicht hätte. Früher wurden solche Aufgaben ausschließlich von Spezialisten erledigt, die nicht dauernd und mit unbegrenzter Zeit zur Verfügung stehen. Ich will nicht behaupten, dass die KI diese Spezialisten schon ersetzen kann, jedoch können diese Test mittels KI Unterstützung viel häufiger und in einem viel größeren Umfang durchgeführt werden.

Herausforderungen

KI Unterstützung bei Angriffen

Es wird zwar auf diverse Arten von den Herstellern von Security-Software sichergestellt, dass deren Tool nicht für Angriffe genutzt werden können, trotzdem wäre es naiv, anzunehmen, dass die Angreifer nicht auch KI einsetzen. Zum Beispiel gibt

es "Crawler", die selbstständig das Internet nach Sicherheitslücken absuchen und diese an potentielle Hacker melden.

Uns muss also klar sein, dass unsere Applikationen angegriffen werden. Gedanken wie "Wer sollte was von uns wollen?" oder "Wer sollte in so einen Angriff auf uns etwas investieren?" sind mittlerweile absolut falsch am Platz. Der Angriff erfolgt automatisch, ohne a priori ein Ziel zu haben und verursacht beim Angreifer kaum Aufwand.

Eine andere Möglichkeit, KI zu Angriffszielen zu nutzen, ist Big-Data Analyse von geleakten Daten. Damit wird einerseits das Knacken von Passwörtern einfacher (Es müssen nicht einfach alle Kombinationen durchprobiert werden, sondern die KI kann unterschiedliche Wahrscheinlichkeiten ableiten und so viel besser "raten"), andererseits werden dadurch die "Sicherheitsfragen" ("Wie hieß ihr erstes Haustier?", "Wie ist der Mädchenname ihrer Mutter") deutlich unsicherer. Auch "Social Engineering" (Also das Erlangen von vertraulichen Informationen über soziale Interaktionen) wird so einfacher, da man das "Opfer" deutlich besser kennt. Dies bedeutet, dass wir bei der Auswahl von Passwörtern noch deutlich vorsichtiger werden müssen (oder überhaupt auf andere Identifikationsverfahren ausweichen sollten). Es muss uns bewusst werden, dass wir, ob wir das wollen oder nicht, immer gläserner werden und niemanden vertrauen sollten, nur weil er uns gut "kennt".

Anlernen der KI

Damit die KI richtig arbeiten kann, muss sie oft erst mit Informationen gefüttert werden, also sie muss lernen. Dies kann prinzipiell auf 2

Arten erfolgen. Entweder man stellt ihr die Informationen aufbereitet zur Verfügung (z.B. als Musterdaten oder Regeln) oder man lässt die KI arbeiten und gibt ihr Feedback und verbessert sie so ständig. Im Normalfall wird eine Kombination aus beidem sinnvoll sein.

Es erfordert also Spezialisten, die sowohl die KI am Anfang richtig einstellen als auch im Betrieb dann warten und gegebenenfalls korrigieren.

Analysieren und Auswerten der Alarne

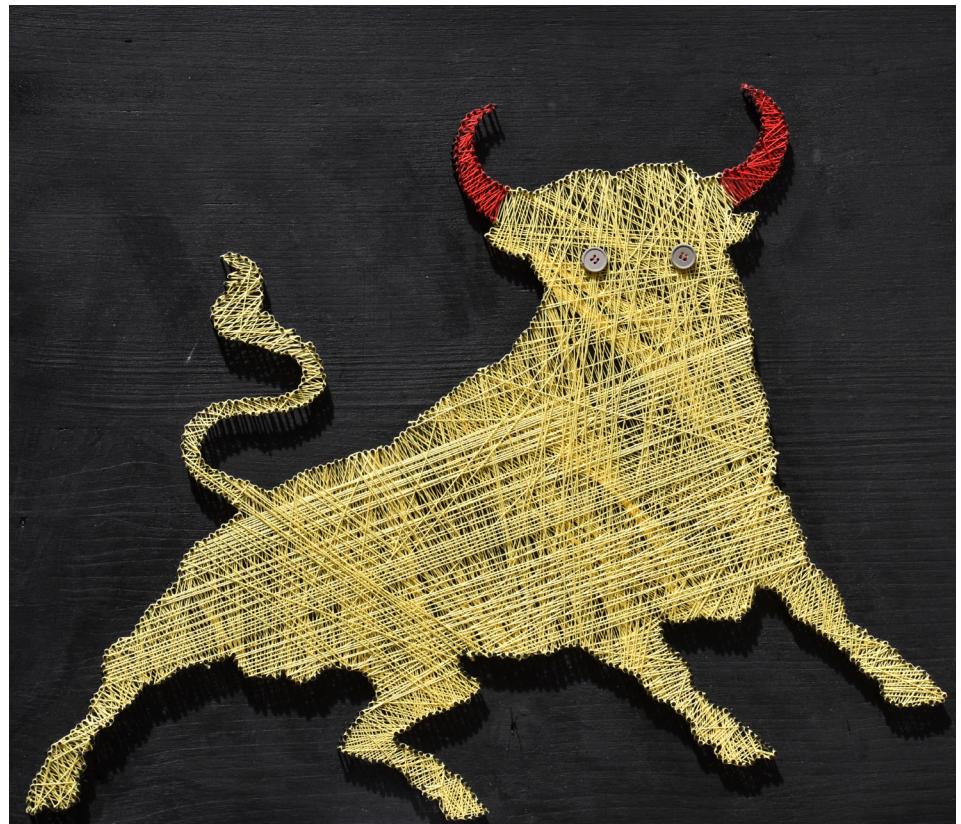
Oft sind die KIs so eingestellt, dass eher einmal zu oft "Alarm" geschrien wird als zu wenig. Prinzipiell ist das auch gut so, man muss diese Alarne dann halt richtig interpretieren und weiterverarbeiten. Es wird auch notwendig sein, das Verhalten der KI anzupassen, um die Trefferquote zu erhöhen und die False-Positive-Rate nach unten zu treiben. Sorgfalt und Gewissenhaftigkeit ist hier geboten. Schnell werden manche Überprüfungen einfach abgeschaltet, weil sie dauernd anschlagen, anstatt sie richtig einzustellen und kontinuierlich zu verbessern.

Falsche Sicherheit

Man darf nicht der Versuchung erliegen, die ganze Verantwortung an die KI zu übergeben. Das System ist immer nur so sicher wie das schwächste Glied. Es ist ein gesamtheitliches Security Konzept notwendig. Defensive Programmierung, saubere Serverarchitektur und Aufmerksamkeit bezüglich Social Engineering kann nicht durch KI-gesteuerte Überprüfungen ersetzt werden.

Fazit

KI ist kein Allheilmittel. Sie wird uns immer mehr Aufgaben abnehmen und uns unterstützen, aber es werden sich auch immer neue Herausforderungen auftun. Verzichten kann man auf sie mittlerweile nicht mehr, man muss sich aber bewusst sein, wo die Schwächen und Stärken liegen und wie man damit umgeht. ■



Titel: „Stier“, Hans Steininger, Technik: Fadenbild



Markus Schwabeneder ist Senior Consultant.

Er begann seine Karriere in der Softwarebranche als Consultant für komplexe mathematische Themen und Optimierungsaufgaben. Seitdem ist er in den Bereichen Softwareentwicklung, Anforderungsanalyse, Testing und Softwarearchitektur tätig. Seine Fähigkeiten setzt er in unterschiedlichen Branchen ein.

Besonders begeistert er sich für das Erarbeiten von komplexen Anforderungen und spezifischen Vorgehensweisen.

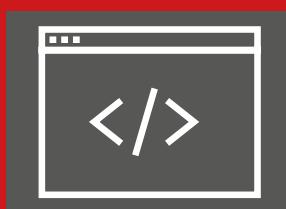
SEQIS ist der führende österreichische Anbieter in den Spezialbereichen
IT Analyse, Development, Softwaretest und Projektmanagement.

Beratung, Verstärkung und Ausbildung:
Ihr Partner für hochwertige IT-Qualitätssicherung.



IT ANALYSE

Notwendige Änderungen analysieren und IT-gerecht aufbereiten



CODING &

Development
Guten Code schreiben und schlechten Code überarbeiten



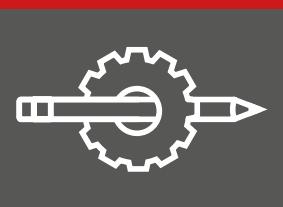
TESTING

Probleme durch methodischen Soll-Ist Vergleich erkennen



RELEASE &

OPERATE
Reibungsloser Go Live und Betrieb der IT-Lösungen



DEVOPS

Neuerungen abgestimmt mit Entwicklung und Betrieb live setzen



METHODOLOGY & TOOLS

Vorgehensweisen optimieren und auf die richtigen Tools setzen



TRAINING & WORKSHOPS

Mitarbeiter Know-how stärken – standardisiert oder maßgeschneidert



PROJEKT-MANAGEMENT

Verantwortung übernehmen, zielorientiert und pragmatisch agieren