Interview 2

Speaker1: [00:00:11] Hallo, Interviewpartner 2. Herzlichen Dank für die Bereitschaft des Interviews. Ist es für dich in Ordnung, wenn wir das Gespräch aufzeichnen?

Speaker2: [00:00:39] Ja, gerne. Ja.

Speaker1: [00:00:41] Perfekt. Ich würde kurz eine eine Einleitung machen, um auch den Kontext des Interviews herzustellen. Für mich geht es heute darum, mehr über das Thema Low Code Softwareentwicklung durch den End-Nutzer in Erfahrung zu bringen. Und dabei möchte ich unter anderem schauen, inwiefern dies in einem klassischen Scrum-Model abbildbar wäre und wo man allenfalls Anpassungen berücksichtigen müsste. Zu Beginn möchte ich gerne mehr über dich erfahren. Kannst du mir etwas über deine aktuelle Position und auch woran du arbeitest eine einleitende Beschreibung geben?

Speaker2: [00:01:24] Ich arbeite bei Unternehmen X und ich verantworte in der Schweiz in der Schweiz unser No Code Low Code Geschäft. Unternehmen X ist eigentlich bekannt geworden als IT Service Management Plattform, aber die Grundidee der Unternehmen X Plattform ist eigentlich eine Low Code Plattform und wir unterstützen somit hauptsächlich bestehende Kunde der Schweiz in ihrer digitalen Transformation, um eben mit dieser Low Code Plattform Prozesse schnell und effizient zu digitalisieren, so digitale Brüche, sage ich mal zu entfernen und manuelle Schritte ja loszuwerden.

Speaker1: [00:02:12] Deine konkrete Berufsbezeichnung ist.

Speaker2: [00:02:16] Solution Sales, Manage Creator und ERP Workflows.

Speaker1: [00:02:22] Danke. Wann bist du das erste Mal mit Low Code Software in Kontakt gekommen?

Speaker2: [00:02:28] Das war 2014, also so ziemlich direkt nach dem Studium. Der erste richtige Job. Da habe ich bei einer Consultingfirma angefangen zu arbeiten und die hatten ein kleines Team. Also wir waren drei Leute am Anfang, die auf der

Unternehmen X Plattform Applikationen entwickelt haben für Kunden, also als Implementation, Partner und Beziehungspartner. Genau. Und das war mein erster Kontakt. Mein Hintergrund ist ja Java Entwickler. Vorher habe ich Informatik studiert. Java entwickelt. Jahre davor. Das war für mich wirklich so, okay. Was habe ich überhaupt gemacht? Die letzten fünf Jahre? Als war für mich komplett neu. Es war für mich sehr erstaunlich. An der Universität war das nie ein Thema.

Speaker1: [00:03:32] Inwiefern unterscheidet sich für dich generell Low Code Softwareentwicklung von einer klassischen Softwareentwicklung?

Speaker2: [00:03:41] Ich denke mal im Vordergrund steht, steht die Geschwindigkeit. Da ist erstaunlich, wie schnell man zu einer Lösung kommt. Ich vergleiche das immer sehr gerne - für die das Thema neu ist - mit einem Lego Bausatz sag ich mal, anstatt dass ich jetzt mein Auto, mein Haus, was weiß ich auch immer, dass ich da bauen will, komplett von null aus aus Holz oder von mir aus aus irgendwelche gegossenen mit einem 3D Drucker oder jeden Baustein selber Drucke kann ich ja mit einem Lego Bausatz eigentlich bestehende Komponenten wiederverwenden. Und ich glaube, was ganz wichtig ist, ist auch das, dass ich mir selber aus diesen kleinen Lego Bausteinen wieder größere Komponenten bauen kann, die ich dann in einem anderen Projekt wiederverwende. Also das ist mal diese Wiederverwendbarkeit von Baustein-Komponenten ein ganz wichtiger Bestandteil jetzt nebst nebst dem dieser einfach der der Geschwindigkeit. Die Zusammenarbeit mit dem end user mit dem business: da gibt es verschiedene autoritäts stufen um das mal so zu nennen. Es gibt natürlich das Wort citizen development. Ich lasse die wirklich selber entwickeln, aber was ich auch gesehen habe, da unterscheidet sich auch schon die Entwicklung extrem stark ist, ich kann viel näher am Business entwickeln. Also ich sage mal klassischerweise im Scrum Ansatz werden User Stories geschrieben, die kommen in einen Backlog und dann werden die User Stories abgearbeitet und dann werden die wahrscheinlich getestet oder so irgendwas und dann entweder okay oder wieder zurück. Natürlich, das macht man auch. Definitiv. In der No Code Low code Entwicklung, aber sich auch ganz oft gemacht habe ist. Einfach mit dem Anwender. Mit der Person, die die Prozess oder das Prozess Know how hat. Er weiß, was er von der Applikation will. Gemeinsam mal einen Nachmittag an einen Tisch setzen und einfach gemeinsam mal eine erste Version des Prozesses abbilden. Eine erste Version des Formulars abbildet. Eine erste Version was auch immer der Applikation ist Logik abzubilden, um einfach dieses minimal valuable

Product bauen und dann schauen wie kann ich das iterativ dann vielleicht wieder mehr im klassischen Softwareentwicklung Ansatz verbessern? Ich habe jetzt noch den dritten Punkt, aber ich weiß nicht, ob du.

Speaker1: [00:06:13] Es ist ein unstrukturiertes Interview. Also wir müssen nicht eine fixen Struktur folgen.

Speaker2: [00:06:19] Okay. Weil der dritte Punkt - Ich nenne das mal durch diesen Experience Ansatz. Oder wenn ich, wenn ich eine Plattform habe oder von mir aus auf zwei, drei Plattformen, auf der ich Applikationen entwickle. Durch diese Wiederverwendung von Komponenten haben diese Applikationen wahrscheinlich alle ein bisschen, das ähnliche Look and Feel. Also und das ist auch nicht unwichtig. Oder wenn ich jetzt eine neue Applikation entwickle und ich brauche dann zwei Monate, um meine ganze Belegschaft zu trainieren, wie diese Applikation zu nutzen ist zum Beispiel, oder eine neue mobile Applikation, oder Sie müssen die zuerst installieren. Und so weiter und so fort, dann ist das auch in der Zeit. Ich brauche, wenn ich eine No Code Plattform schon strategisch einsetze, im Unternehmen und sie machen ihre Spesenabrechnungen darüber, alle nutzen die Plattform, wenn Sie eine Anfrage ans Marketing oder ans Department haben, wenn ich jetzt eine neue Applikation darbringe Corona oder ich muss mich jetzt anmelden am Morgen, wenn ich ins Office kommen will und einen QR Code zeigen, zum Beispiel muss ich meine Leute da nicht mehr enabelbn oder trainieren, dann ist das vielleicht halt ein weiterer Menüpunkt in einer bestehenden Applikation oder auf Ihrem Mobiltelefon oder in einem Portal, auf das Sie Zugriff haben und dann funktioniert das einfach. Also dieser User Experience Ansatz, wo ich mich schnell als User auch wiedererkenne in gewissen Applikationen, das ist für mich auch ganz essenziell, weil eben, es geht mir zurück in Time to Market, dass ich halt schnell was auf die Beine stellen kann, aber das auch schnell nutzen kann.

Speaker1: [00:08:01] Wie wirkt sich das dann aus auf die klassischen Rollen? Also, wenn man bedenkt exemplarisch man hat ja Rollen in der klassischen Softwareentwicklung, man hat einen Product Owner, einen Business Analysten, einen Entwickler. Wenn du jetzt an die Entwicklung durch den end user denkst, wie kann sich dies auf das zusammenspiel der rollen auswirken?

Speaker2: [00:08:29] Hm, also es gibt sicher eine neue Rolle. Behaupte ich jetzt mal, dass ist die Rolle des Platform Owners oder wir benennen das auch gerne mal das Centre of Excellence and Innovation, diese das Team oder diese Person, je nachdem halt, wie groß das ist. Das ist dann einfach mal grundsätzlich verantwortlich für den Betrieb der Lösung. Weil eine große Frage, die natürlich mit diesem Thema Citizen Development oder Entwicklung durch den End User kommt, ist eigentlich der Betrieb nicht unbedingt die Entwicklung, oder weil ja, wer nimmt das Telefon nachts um 2:00 dann ab oder wer bearbeitet das, wenn die Applikation nicht mehr funktioniert? Das wird vielleicht nicht dieser Business User sein, oder? Also das ist mal schon mal, das ist dann natürlich etwas, das kommt nach der Entwicklung jetzt für die Entwicklung. Also ich sag jetzt mal natürlich ganz einfache Applikation jetzt wirklich. Ich bin ein Team von fünf Personen. Irgendjemand hat eine Idee. Wir machen unsere Tasks, machen wir eine Excel oder sowas? Jetzt lass uns mal in eine kleine App bauen, damit wir hier einfach uns Task zuschieben können. Natürlich da. Ich denke da zum Beispiel an die Power apps Plattform oder so irgendwas. Da braucht es aus meiner Sicht dann kein Business Analyst, kein Product Owner. Dann wird diese Person wahrscheinlich einfach in diesem Team in zwei, drei Tagen diese app bauen und dann kann das Team die nutzen. Jetzt mal vom Betrieb abgesehen. Was ist, wenn die Person das Unternehmen verlässt? Das sind dann fragen, die geklärt werden müssen. Aber wenn wir jetzt an komplexere Applikationen denken oder komplexere Prozesse, dann braucht es diese Personen definitiv auch noch. Also und natürlich auch die professionellen Entwickler. Weil sobald es zum Beispiel auch in und um Integrationen um Systeme geht oder so irgendwas, Anbindung an Master Daten oder schon nur Anbindung an irgendwie ein SAP System oder so, weil ich vielleicht irgendwas brauche, dann kommt man nicht ohne die aus. So ich meine das wird ein Citizen Developer nie selber machen. Natürlich, der kann wieder die vorgefertigte Komponente nutzen, die vielleicht ein Etwas aus SAP ist oder es etwas in SAP reinschreibt. Aber diese Komponente wurde entwickelt von einem professionellen Informatiker sage ich mal mit gewissen Anforderungen, die von einem Business Analysten wahrscheinlich geschrieben wurden. Und der Product Owner ist dann wahrscheinlich auch der Product Owner/Plattform Owner. Und dann vielleicht ist dann zuständig dafür. Okay, wie wird diese Schnittstelle weiterentwickelt und ist auch dafür zuständig, dass die Schnittstelle funktioniert, wenn sich irgendwas am SAP ändert oder so irgendwas. Also die braucht es da trotzdem noch. Ich. Sobald die Applikation ein bisschen komplexer wird, ändert sich aus meiner Sicht da nicht so viel. Es ändert sich vielleicht ein bisschen was, wie die Leute da zusammenarbeiten. Vielleicht nicht

mehr so? Getrennt ist ja eigentlich auch das falsche Wort bei diesem agilen SCRUM Ansatz. Aber ich sage es eben Man kann halt der Entwickler, der No Code, Low Code Entwickler, der kann noch viel mehr, dann vielleicht mit dem Business zusammenarbeiten. Also das es immer über diesen Business Analysten geht, dann denke ich ist nicht unbedingt nötig.

Speaker1: [00:11:29] Wir haben jetzt die Business Analyse besprochen, wo du sagst, braucht es vielleicht für ein paar kleinere Projekte gar nicht mehr. Wie siehst du das mit der Priorisierung. In Scrum gibt es einen Product Backlog. Du hast in einem Priorisierung Mechanismus ein Backlog wie Wie denkst du wirkt sich das auf die Priorisierung von Vorhaben aus, wenn du den End User entwickeln lässt?

Speaker2: [00:12:12] Hm, also ich kann vielleicht mal ein Beispiel machen von einem Unternehmen, mit dem wir zusammenarbeiten, die so wirklich ein Citizen Development Programm haben und die da fängt halt die Priorisierung oder die Kategorisierung der Application, die beginnt eigentlich schon mit. Was will der User? Ich sage es mal so, Die, die stellen eigentlich. Ich glaube, es sind drei Optionen dem User zur Verfügung. Das ist die wirklich klassische Citizen Developer Option. Das ist, du bekommst ein Training. Es gibt vielleicht sogar so Office Hours, wo ich mal mit einem Entwickler sprechen kann, wenn ich nicht weiterkomme. Du brauchst null Budget und du kannst selber entwickeln, wie du willst. Natürlich. Da gibt es wahrscheinlich ein paar technische Limitationen. Und da, da gibt es diese klassischen Rollen dann wirklich alle nicht. Da kommt einfach der Citizen Developer, der, der schreibt da, ich will diese und diese Applikation. Vielleicht gibt es eine kleine Überprüfung oder durch einen durch, dass diese Center of Exzellenz einfach so, um sicherzustellen, okay, jetzt will der Kundendaten in ein Portal stellen oder so was geht nicht zum Beispiel oder sage ich mal, aber ansonsten diese Person arbeitet dann komplett alleine und baut ihre Applikation eben. Vielleicht kann mal eine Hotline anrufen, um da Hilfe zu bekommen. Zweite Option, die das Unternehmen hat, die nennen das Monday to Friday und da ist dann der Ansatz eher so okay, du hast Budget, du willst diese App eigentlich nur teilweise selber bauen oder diesen Prozess nur teilweise selber digitalisieren.Und wir als Centre of Excellence oder als App Factory, wie sie das dann nennen wir helfen. Wir bauen dir diese Applikation eigentlich und da natürlich gibt es auch wieder eine klassische Priorisierung von diesem Projekt und ich weiß jetzt ehrlich gesagt nicht, wie genau das funktioniert, aber ich meine, da geht es dann wahrscheinlich um was ist der

Value? Wie kritisch andere sage ich jetzt mal Parameter, die dann entscheiden, wie kritisch diese Anfrage ist und ob man die umsetzen soll, ist jetzt nicht unbedingt an einzelne Features, also Product Backlog mäßig Priorisierung, sondern einfach schon mal Priorisierung von von der Anfrage. Und da entscheidet dann halt ein Wir haben dann so auch ein Steering Committee bestehend aus Enterprise Architekten bestehend aus. Ich weiß nicht, wer da alles drin sitzt. Vielleicht noch andere Plattform Owner, die dann entscheiden, welche Vorhaben jetzt entsprechend abgebildet werden. Ist vielleicht dann auch wirklich halt auch wie so oft eine Frage des Geldes intern und extern. Und dann die dritte Version, das ist dann so ein Mix, also das ist dann ich habe ein bisschen Geld, will viel selber machen, aber brauche Hilfe, halt für mal eine Integration und so und da weiß ich ehrlich gesagt nicht, wie die Priorisierung funktioniert. Aber ich glaube, dass das ändert sich jetzt für die größeren Vorhaben. Ändert sich da nicht groß etwas.

Speaker1: [00:15:08] Also von einem klassischen Prozess dann.

Speaker2: [00:15:11] Ja, weil eben, da ist dann halt einfach die Frage, wie passt das nun zur. Ich sage jetzt mal passt das in unser Portfolio. Wie passt das in unsere Architektur oder diese Applikation, diesen Prozess, den man da bauen will? Genau. Aber der klassische Citizen Development Ansatz, da denke ich, sieht das ganz anders aus. Wobei, man muss natürlich auch sagen, wenn ich mit meinen Kunden Citizen Development bespreche, dann sage ich ihnen immer, das ist zu 80, 90 % kein Technologiezentrum, das ist ein reines: Wie organisiert ihr euch und was für Governance Strukturen habt ihr? Weil was du ja auch nicht willst ist okay. Jetzt kann jeder Apps entwickeln und in fünf Jahren haben wir 1500 Applikationen und 500 davon tut das gleiche. Also das kann nicht das Ziel sein. Das war schon mal Lotus Notes, das gab es schon mal und jetzt sind es genau die Kunden. Mit denen spreche ich jetzt. Ja, wir haben Lotus Notes. Kommt nicht weg davon. Was machen wir? Wir wissen gar nicht, was diese Apps alle machen. Also, ich hatte mal mit einem großen Pharmakonzern eine Diskussion, die meinten, Sie schätzen, Sie haben zwischen 30.000 und 35.000 Applikationen. Sie wissen natürlich die Großen, kein Problem, aber was sie nicht wissen und das ist ein großes, großes Security Problem: Forscher, die irgendwo unter ihrem Schreibtisch ein Server stehen haben, da haben die mal irgendwie weil Forscher, die können man selber noch etwas programmieren oder so irgendwas programmiert. Das nutzen jetzt ein paar Teams oder so, vielleicht uralte Java Versionen, uralte Dot Net Version. Keine Ahnung was die sind, keine Informatiker, die

wissen nicht, dass das gepatcht werden muss oder so, also das dieses wir nennen das Apps prall. Das Problem, das dann halt wirklich einfach entwickelt wird, ohne ohne Kontrolle. Das ist etwas, das unbedingt beachtet werden muss.

Speaker1: [00:17:00] Und für mich ein Thema, das noch nicht finalisiert ist, ist generell das Thema testen. Wenn du in einem in einer klassischen Softwareentwicklungsmethode unterwegs bist, hast du ja das Testen integriert. Wenn der End User entwickelt wie gestaltet ihr das Testing?

Speaker2: [00:17:26] Ja, da ist für mich wieder komplett abhängig von der Komplexität. Weil es gibt halt wirklich, es gibt so simple Applikationen, die einfach das ersetzt, ein Excel Sheet. Und jetzt kommt halt jemand, der baut sich das in Power Apps oder in ServiceNow oder was auch immer welche Plattform und da. Ich glaube, da wird gar nicht getestet. Einfach. Und wenn es halt nicht business kritisch ist, spielt das auch nicht wirklich eine Rolle, wenn es nur dazu dient. Okay, wir erleichtern jetzt unsere Arbeit ein wenig im Team, da ist ja egal. Und dann natürlich noch Klammer auf, keine kritischen Daten, die irgendwo auf ein Portal landen könnten oder was weiß ich auch immer. Und eben für alles andere auch da wieder. Da braucht es eine klare Testing Strategie. Also wir haben zum Beispiel auch auf unserer auf unsere Plattform legen wir einen großen Wert auf eben diesen Governance Themen und haben zum Beispiel auch eingebaut in der Plattform Continuous Integration, Continuous Development Pipelines. Und da wäre natürlich dann auch ganz wichtig, eben, dass die IT bleibt bzw. hat den die Kontrolle. Wie geht jetzt so eine Applikation durch den Entwicklungsprozess? Wie oder was sind die Kriterien, dass die auf der Test Instanz die Deployed werden? Welche Tests müssen dann da durchgeführt werden, dass ich dann auf Produktion deployen kann und dieser Prozess muss abgebildet werden und oder hoffentlich mehrere Prozesse werden da abgebildet. Vielleicht eben Prozesse für. Mach was du willst, Egal-Typ-Applikationen. Dann vielleicht so ein Mittelweg. Okay, die sind Business kritisch, da müssen wir auch Support sicherstellen. Und so weiter und so fort. Und dann wirklich vielleicht ein Prozess für gerade im Umfeld Pharma oder sage ich mal Banking, eine validierte Applikation, sprich wir müssen, bevor die in Produktion geht, müssen wir Dokumentationen schreiben, keine Ahnung. 20, 30 Seiten Dokumentation. Damit klar ist, wenn es um Drug Development geht oder so was oder irgendwie damit klar ist, was diese Applikation macht, wie sich der Prozess aufbaut. Und ist dieser Prozess Compliance mit FDA Regulatorien zum Beispiel irgendwas in diese Richtung? Aber

eben, das geht natürlich schon. Da sehe ich Citizen Development dann vor allem in einzelnen Bereichen. Also haben wir jetzt gerade auch das Thema Pharma Kunde. Das Team Drop Produktion der Medikamente. Also man kann sich das so vorstellen, die sind so ein bisschen fast wie in Silos. Es gibt die Entwicklung oder die Forschung oder neue Medikamente entwickeln. Dann gibt es so ein bisschen die Clinical Trial Silo und dann gibt es jetzt produzieren wir das Medikament, also alles ist gut, es wird produziert und auch innerhalb von dieser Produktion gibt es natürlich, sage ich mal, ein Life Cycle von einem, von einem Medikament, also das sind das ganze Supply Chain, wo wird produziert, wie viel wird produziert und irgendwann wird das Medikament vielleicht nicht mehr produziert. Und da arbeiten wir jetzt auch gerade dran, diesen gesamten Prozess abzubilden. End-to-End. Also, dass man wirklich immer weiß, wo steht ein Medikament in diesem, in diesem Life Cycle. Und da ist auch das Thema Citizen Development ein oder ist es kommt. Dadurch kommt die Anforderung von Business, aber jetzt nicht um die Applikation sage ich es mal als solches zu entwickeln, sondern die wollen dann einfach selbstständig zum Beispiel die Business Rules anpassen können, die wollen selbstständig approval Regeln anpassen können. Und so weiter und so fort. Ich sage jetzt mal, die wollen halt nur einen Teil der Applikation selber verwalten können. Ist aber trotzdem aus meiner Sicht, das ist aus meiner Sicht auch Citizen Development, weil bis jetzt müssen sie jedes Mal zur IT und sagen Hey, wir müssen unsere Business Rules anpassen, irgendwie approval Regel oder so was. Und dann dauert es zwei, drei Monate bis bis diese Anpassung kommt. Und wenn Sie das jetzt selber machen können, umso besser.

Speaker1: [00:21:33] Und jetzt grundsätzlich Welche Herausforderung siehst du bei der Low Code End User Programmierung generell?

Speaker2: [00:21:43] Äh, also es ist mal sicher eine. Es ist so die klassische Herausforderung, dass das Business anders denkt als ein Programmierer. Meistens irgendwie. Sie haben halt eine ganz andere Herangehensweise. Ich sage jetzt mal, ein Informatiker, der würde mal, wenn er irgendein Problem bekommt, würde das sehr strukturiert, vielleicht mal das Datenmodell überlegen. Was hab ich eben für Schnittstellen oder so was? Und der, der End User, der denkt an okay, das will ich machen oder wie sieht das Formular aus oder so was nicht vielleicht an andere Dinge. Und ich glaube. Die Tools gehen natürlich unterdessen in diese Richtung, also dass man wirklich so What you see is what you get eigentlich ja entwickelt. Aber

nichtsdestotrotz jetzt rein aus technischer Sicht denke ich, braucht es da noch ein bisschen Weiterentwicklung, dass es für wirklich Leute, die keine Ahnung haben von Programmierung oder halt keine Ahnung haben von grundsätzlich so diesen Basic Informatiker Know how, dass das noch einfacher wird. Wobei eben aus Technologie sicht sowieso. Wenn. Wenn. Wenn uns leute fragen Citizen Development wer ist dafür geeignet? Dann sagen wir schon ganz klar. Das muss jemand sein, der schon technikaffin ist und auch Lust hat dazu. Ich meine, das ist ja. Das ist schon mal jemand für mich, in dessen Job Description steht nichts von irgendwie du entwickelst Application oder schon nur irgendwie was in diese Richtung geht oder? Alle anderen sind für mich schon einfach mal No Code / Low-Code Entwickler. Also die sind für mich wirklich weg von der IT. Und die muss man halt wirklich so entsprechend abholen, auch auf der technischen Ebene, dass die das einfach verstehen oder trainieren können. Es braucht für mich ein klares Citizen Developer Programm. Es braucht für mich auch klar Support vom Executive Support sage ich mal, es muss dann auch klar sein, dass wenn ich jetzt eine Plattform einsetze, in meinem Unternehmen wirklich eine Plattform, dass das die Strategie ist, weil was macht das Business heute? Das ist ja auch ein Grund für Citizen Development. Zur IT wollen sie nicht, weil teuer, langsam und mühsam. Also, wenn Sie einen anderen Weg finden, nehmen Sie Ihre Company, Kreditkarte und gehen auf Monday Dotcom oder was weiß ich. Irgendein SA Service und holen sich da die Lösung. Das ist der, das will man ja auch verhindern. Die Katastrophe, als CIO. Wenn du nicht mehr die Kontrolle hast, was das Business macht und deshalb muss es auch ganz klar dann Kommuniziert werden. Internes Marketing. Deshalb ist eben auch dieses Centre of Excellence and Innovation für uns so wichtig. Deren Aufgabe ist nicht nur, die Plattform zu betreuen, sondern auch intern Marketing zu machen, für die Plattform. Workshops zu machen, mit dem Business, gemeinsam mit dem Business oder dem Business zu erklären, was ist möglich auf der Plattform. Und das geht eben alles schon wieder in Richtung Governance und Organisation, also die Herausforderung für mich No Code Low Code oder Citizen Development ist, wie schon gesagt 90 % der Organisation Governance.

Speaker1: [00:24:56] Welche weiteren Maßnahmen, denkst du, diese Herausforderung zu überwinden? Neben dem, dass man die Attention auf Executive Level haben sollte.

Speaker2: [00:25:12] Es braucht eine Plattform-Strategie aus meiner Sicht. Also wir sagen meistens okay, ein Unternehmen hat vier, fünf, sechs Plattformen oder so was.

Microsoft, klar, da wird irgendwie ein SAP noch da sein und dann abhängig von der Branche, sage ich jetzt mal Versicherungsumfeld, gibt es irgendein ich weiß gar nicht, was das ist, aber im Kern Versicherungssystem in Banken Umfeld ein auch im Kern-Backend-System. Dann gibt es wahrscheinlich ein Sales Tool, oder CRM. Marketing gibt es irgendwas. HR gibt es irgendwas? Und dann sollte aus unserer oder aus meiner Sicht ist dann eigentlich noch eine weitere Plattform. Ist dann eben diese No Code-Low-Code Plattform, die dann dazu dient, auch um diese System herum oder diese Systeme zu verbinden, da Prozesse end to end abbilden zu können. Also eine klare Strategie aus auch aus IT-Sicht ist sicher wichtig. Dass da auch sich mal jemand überlegt okay, was wird für was eingesetzt? Was machen wir, wenn jetzt jemand kommt mit der Anforderung weiß nicht. Dieser E-Cigarettes ist jetzt gerade so ein Beispiel. Jetzt müssen wir. E-Cigarettes verkaufen wir anders als unsere klassischen Zigaretten. Da weiß ich ganz Sie auch haben wir an Kiosks diese Leute, die da e Zigaretten andrehen wollen, was weiß ich. Die müssen wir trainieren. Die müssen wir onboarden. Da müssen wir sicherstellen, dass die bezahlt werden, wenn sie irgendwie 50 Zigaretten verkauft haben, was weiß ich. Aber immer, wenn jetzt jemand kommt, oder wir wollen uns eine Lösung bauen, was machen wir jetzt? Das kaufen wir jetzt wieder eine Point-Solution. Bauen jetzt selber was from Scratch. Mit unserem traditionellen Softwareentwicklung Methode nutzen wir einen Low Code Plattform. Haben wir vielleicht von unserem CRM System die Möglichkeit einfach etwas kurz anzupassen und dann können die auch das nutzen. Also diese Fragen, die müssen zumindest, sie müssen gestellt werden. Oder dass da nicht einfach ist jedes Mal. Ich sage mal der, der am lautesten schreit, der bekommt dann oder auf dieser Plattform wird das dann entwickelt oder so, also eben diese. Ja, das ist. Es geht alles zurück zu diesem Executive Alignment, auch Support und daneben internes Marketing, internes Marketing. Das ist sicher ganz wichtig. Ja.

Speaker1: [00:27:38] Jetzt, wenn du schaust. Ich versuche hier noch noch die die Brücke zu Scrum zu machen. Scrum hat ja wie klassische Elemente Rollen, auch Plattformen für die Kollaboration, Rhythmen. Inwiefern denkst du, dass and End-User-Development in einem Scrum Prozess abgebildet werden kann? Und falls bzw müssten dort Anpassungen gemacht werden an einzelnen Elementen, dass das sinnvoll wird?

Speaker2: [00:28:12] Nein, denke ich eigentlich nicht. Also Anpassung denke ich nicht. Was Scrum, hat sich bewährt. Oder ich eben nochmals Klammer auf für ganz einfache

Dinge, die einfach innerhalb von zwei Tagen entwickelt und interessiert niemanden, braucht es aus meiner Sicht kein Scrum. Also da braucht es dann andere Mechanismen, einfach für eben, dass jemand die Kontrolle hat, über wie viele Applikationen wer was entwickelt, aber jetzt für dann wirklich den Softwareentwicklung Teil. Also aus meiner Sicht ist das das ganze Thema Scrum und agile Entwicklung. Das Business hat sich schon ziemlich daran gewöhnt, also in allen Projekten, seit ich 2014 in diesem No Code-Low-Code Umfeld unterwegs bin, war immer dieser Scrum Ansatz. Also haben wir immer mit dem Scrum Ansatz gearbeitet und ich glaube, das Business ist sich heute dann natürlich über die Business Analysten, die sind sich absolut auch gewöhnt, dass man als User Stories schreibt, dass man die entsprechend dann auch testet, Acceptance Criterias schreibt und so weiter und so fort. Vielleicht kann kann man schneller. Also ist dann halt ein Sprint eine Woche und zwei Wochen zum Beispiel, weil man einfach mehr machen kann in einem Sprint. Aber ansonsten sehe ich da. Weil eben auch No Code Low Code kann schnell mal eine gewisse Komplexität erreichen oder von der Softwareentwicklung. Das heißt nicht nur weil es No Code Low Code ist, dass es eine dumme kleine Applikation ist. Das kann auch ein Projekt sein, das fünf Monate dauert, mit mehr Releases oder so was. Oder wenn ich wie in diesem Pharma Beispiel, wenn ich für da diese Leute für den kompletten Prozess, den Life Cycle abbilde. Das wird ein Eineinhalb zwei Jahres Projekt sein. Also das heißt nicht, dass es weniger oder schneller geht, also schneller im Vergleich, wenn ich das von Null aufbauen würde, aber. Trotzdem ein langes Projekt. Da muss ganz klar da gibt es den genau gleichen Ansatz wie in der klassischen Softwareentwicklung.

Speaker1: [00:30:16] Ja, ich denke, ich denke dabei bewusst nochmals auch an die Sicht, dass man beim Citizen Developer teilweise auch dezentralisierte Entwickler in den Bereichen hat, vielleicht auch Teilzeit Hobby Entwickler. Wie siehst du da die Kompatibilität mit Scrum auch in Bezug zum Beispiel auf eine Daily?

Speaker2: [00:30:42] Ja, das ist eine gute Frage. Das habe ich ehrlich gesagt nicht ganz ehrlich noch nie so erlebt, dass der der Citizen Developer also entweder ist es wirklich Citizen Development oder der Mix kommt erst nach der ersten Version, sag ich mal und dann wird es. Dann ist der Betrieb. Es wird oft anders gehandhabt. Wir haben zum Beispiel jetzt, wir haben da eben die Möglichkeit, dass man halt dann einschränkt und das passiert dann oft, was der Citizen Developer machen kann. Oder man gibt ihm dann zum Beispiel nur die Möglichkeit. Sag jetzt mal das Formular anzupassen. Eine

Dropdown Box oder so was? Und jetzt kannst du da halt andere Optionen hinzufügen, oder? Und die werden dann verwendet. Okay, wenn es jetzt Option A ist, dann gibt es zwei Kugel Level zum Beispiel oder so und das sind dann Anpassungen. Ich bin gerade überlegen. Die würden dann gar nicht unbedingt so in diesem Scrum. Ja Gebilde gemacht werden. Das wäre dann mehr so okay, der macht das einfach nebenbei. Hat dann natürlich trotzdem die Möglichkeit das zu deployen. Das ist eine gute Frage, weil ich mir noch nie so war. Hat aber hatte ich wirklich noch nie die Situation so es war oft, es war eher so okay wir entwickeln die Lösung Scrum eben unter Einbezug nah am Business. Und so weiter und so fort. Dann ist mal ein Release draussen. Das Core Team oder das IT Team arbeitet weiter mit Scrum, aber die Citizen Developer, die arbeiten daran in ihrer eigenen Pace sag ich mal und die sind dann auch nicht Teil von Daily. Also sie sind natürlich Teil von dem Teil über den Business Analysten vielleicht, aber jetzt nicht als Entwickler.

Speaker1: [00:32:29] Ich finde, ich finde das Thema noch interessant, wie wie man das gestalten könnte. Inwiefern, dass man sich fragt, dass es gewisse Gefäße gibt in Scrum, die, die nützlich sind, die ja alle ihren Zweck haben. Andererseits schon alleine der Gedanke, dass man ein Teilzeit Entwickler sein könnte, führte dazu, dass das vielleicht gewisse Dinge dann einfach nicht mehr geeignet sind oder man sie zumindest anpassen müsste.

Speaker2: [00:33:04] Das ist nicht eben, es ist nicht der Job, meistens von einem Citizen Developer, das heißt, ich meine die Zeit, die er dafür verwendet, irgendeine Applikation zu entwickeln oder schöner zu machen oder was weiß ich. Ich meine, ich muss auch eigentlich, ich muss so kurz oder so klein wie möglich halten. Ich meine, da muss irgendwas anderes arbeiten. Was, was, was hoffentlich Geld bringt. Und da ist schon, nur sage ich jetzt mal, seine Zeit zu blockieren für diese Scrum Meetings. Gute Frage.

Speaker1: [00:33:39] Ich habe hier eine Auflistung. Du kannst mir ja einfach spontan reinwerfen, wenn ich jetzt genau auf diesen Teilzeit Entwickler adaptieren möchten? Was sind zu deinen ersten Gedanken. Die Rollen haben wir schon grob besprochen in meinem Verständnis. Wir haben Entwickler, Product Owner, Scrum Master. Ich denke, dort hast du gesagt, dass es vielleicht eine neue Rolle braucht mit dem Application Owner oder generell Center of Excellence.

Speaker2: [00:34:12] Ist ein bisschen übergreifender, die müssen dann auch die sind werden nicht in jedem. Was auch immer da in jedem Meeting oder so dabei sein.

Speaker1: [00:34:19] Aber wenn man dann an die klassischen Scrum Events denkt, dann gibt es zum Beispiel den Sprint, vielleicht dazu deine Gedanken. Inwiefern das für einen End User geeignet wäre. Oder ob du dort Anpassungsbedarf siehst?

Speaker2: [00:34:41] Also nein, Anpassungsbedarf nicht, wenn er jetzt wirklich aktiv mit entwickelt, weil es jetzt gehen wir mal davon aus, der der arbeitet jetzt zu 30 % wirklich an diesem Projekt mit und programmiert. Also was heisst programmiert, baut diese ab auf. Also dann gehört er wie ein anderer Programmierer genauso in dieses Sprint-Gefäss. Muss er sich daran halten, an dieses Gefäß oder muss er da mitarbeiten, weil ansonsten, das geht ja ansonsten gar nicht mehr. Oder hast du da noch Abhängigkeiten, je nachdem von was er entwickelt, was die oder unabhängig oder wahrscheinlich andersherum. Er ist abhängig davon, was die professionellen IT Entwickler zum Beispiel eine Schnittstelle bereitstellen um selber seine Prozesse baut zum Beispiel und er ist dafür ist abhängig davon, dass jetzt eine SAP Schnittstelle bereitsteht. Dann muss er ja auch wissen, in welchem Sprint zum Beispiel ist es geplant, dass diese, dass diese Schnittstelle entwickelt wird?

Speaker1: [00:35:40] Also ich finde den Gedanken noch interessant. Er müsste dann in diesem Team berücksichtigt sein, weil du ja wie sagen wir, wenn jetzt jemand für sich selbst entwickelt oder für seinen Bereich und dort eigentlich auch alle Rollen für sich erfüllt, wer wäre dann eigentlich Teil dieses Sprint Teams, oder?

Speaker2: [00:36:02] Ja. Er wäre allein. Aber ja. Ja. Gleich wusste ich, es ist eine sehr gute Frage. Und ich habe das noch gar nie so. Ja, ich hatte die Situation jetzt wirklich noch gar nie so.

Speaker1: [00:36:19] Er ist sehr bewusst auch auch eine Frage, die vielleicht gar nicht abschließend beantwortet werden kann. Ja. Ich habe noch weitere Fragen. Gibt es aus deiner Sicht noch Punkte, welche für meine Forschung interessant sein könnten, aber bisher vielleicht noch gar nicht angesprochen wurden?

Speaker2: [00:36:51] Oh, das ist eine gute Frage. Ich meine. Und ich, ich. Ich wiederhole mich da ich glaube, das größte Problem, das, das die Unternehmen heute haben, die jetzt auch wirklich so nen no code low Code, diesen no Code Low Code Ansatz wirklich strategisch fahren wollen ist sind eigentlich genau die Stellen ähnliche Fragen wie du jetzt in deiner Arbeit vielleicht weniger detailliert auf Scrum oder so, aber einfach okay, wie mache ich das überhaupt? Und wir versuchen natürlich da auch. Einen. Wir versuchen da auch mit Blueprint für dieses Centre of Excellence und Innovation oder einfach gewisse Prozesse da Hilfestellung zu leisten. Vielleicht habe ich jetzt gerade überlegt, vielleicht braucht es wirklich halt etwas, das ich meine Scrum oder auch Safe. Das ist etwas, das kann man irgendwie adaptieren für sein Unternehmen. Ich oder viele haben das schon gemacht und vielleicht braucht es wirklich was wie Scrum for No Code, Low Code oder Scrum for Citizen Development. Also so wirklich einen generellen Blueprint, der aber auch anpassbar ist auf die Gegebenheiten eines Unternehmens, weil das ist das Problem, das wir häufig haben. Was Bei Unternehmen A funktioniert und wir richtig stolz darauf sind. Okay, wir haben dem jetzt geholfen, die können jetzt richtig da Agil entwickeln. Das funktioniert nicht unbedingt für Unternehmen B. Weil die vielleicht schon ganz anders aufgestellt sind. Also die IT ist vielleicht anders integriert im Business oder eben komplett nicht integriert im Business. Und ein anderes Unternehmen arbeitet vielleicht schon viel enger zusammen oder sie sind sich schon, ja haben schon andere Gewohnheiten. Also ich habe bis jetzt das Gefühl, ganz ehrlich, es ist etwas, da habe ich noch nie darüber gelesen. Irgendwie so eben Agile und Citizen Development oder so, also. Vielleicht müsste es wirklich etwas geben in dieser Richtung.

Speaker1: [00:38:43] Aus den vergangenen Interviews kam die Idee auch auf, dass man so wie Quality Gates hätte, sowie ein Check in und ein Check out jetzt für den klassischen End-User hätte. Wie findest du den Gedanken?

Speaker2: [00:39:04] Also Check in check out, was man im Sinne von?

Speaker1: [00:39:10] Dass du ein Starterpaket. Wenn jetzt jemand sich mit dem Thema auseinandersetzen möchte, das du ein ein Starterpaket kriegst mit sagen wir einen kurzen Dialog mit was möchtest du auch eine Einordnung der Größe des Projekts. Wenn dann etwas umgesetzt ist, dann wiederum auch ein Check out im Sinne von den wichtigsten Fragen in Bezug auf Compliance. Wie willst du die Daten weiter verwalten?

Wer ist für den Prozess verantwortlich? Das finde ich noch einen nachvollziehbaren Gedanken.

Speaker2: [00:39:46] Und das definitiv. Und das ist genau das was. Was? Ja auch dieses. Erstens wir machen das selber dazu gleich mehr. Aber dieses andere Unternehmen macht. Also da ist wirklich der erste Schritt ist ein Formular, das nicht mal so kurz wo du ausfüllen musst. Eben. Was ist das Thema? Was ist der Business Case? Wie viele User? Was ist der Impact? Und was weiß ich alles, oder? Und. Abhängig davon, was du willst, wird es priorisiert oder nicht und du baust es einfach selber. Wir haben auch ein Citizien Development Programm, intern natürlich sind. Sollte man ja auch machen, wenn man das schon verkaufen will. Und da gibt es einen ganz klaren Onboarding Prozess. Du musst dir einen Citizen Development Badge zuerst mal verdienen. Sozusagen. Da musst du. Äh Trainings machen muss doch alljährlich wieder das Training wiederholen oder diesen Badge erneuern weiß ich nicht ganz genau, wie das funktioniert und daneben auch. Wir haben dann auch Office Hours zum Beispiel, da kannst du dann einmal zweimal pro Woche stehen. Der Entwickler in einem Call, der kann selbst dazu kommen, stehen die zur Verfügung und Fragen zu beantworten. Und dann findet tatsächlich auch dann ein Hand over, wie du gesagt hast, statt. An den Betrieb eben. Also ich bin fertig, das habe ich gebaut. Hier ist meine Dokumentation, die muss auch gewissen Richtlinien entsprechen. Danke. Jetzt dürft ihr das betreiben. Das braucht es definitiv. Ja. Aber. Wiederum, das ist für mich so ein bisschen, behaupte ich mal, ein bisschen, vor Scrum und uns nach Scrum okay. Natürlich. Scrum geht immer weiter, weil es wieder kontinuierlich entwickelt. Aber sicher vor Scrum der Prozess, oder?

Speaker1: [00:41:38] Ich will vielleicht dennoch kurz nochmals auf die Scrum Events gesprungen. Ich finde, für mich ist das Thema Schätzen von Aufwänden in Low Code noch sehr. Und weil die Entwicklungszeit schneller ist, wie schätzt du dann die Entwicklungsaufwand? Wie definierst du Tasks?Das ist für mich noch. Noch nicht geklärt.

Speaker2: [00:42:03] Na also. Also nochmals, ich glaube, dass das Thema sieht irgendwie, das steht noch extrem in den Kinderschuhen, oder? Also ich muss dir ganz ehrlich sagen. 80 % unserer Kunden ist Citizen Development kein Thema. Die nutzen die No Code / Low Code Plattform aus anderen Gründen nicht für das Thema Citizen

Development. Okay, und jetzt wirklich klassisches Citizen Development. Wie schon erwähnt, dass Sie dann selber mal etwas ändern können. Das schon eher. Und da natürlich macht die Aufwand Schätzung. Das machen erfahrene Partner auf dieser Plattform oder oder halt erfahrene Entwicklung. Da ist die Aufwand schätzung genau genau gleich wie wie wenn man klassisch Software entwickelt. Jetzt ja. Thema Citizen Development. Ja, gute Frage. Die machen gar keinen Aufwandschätzung. Die machen ja dann einfach.

Speaker1: [00:42:58] Aber das machen sie dann irgendwie. Wenn, wenn ich hier dazwischen greifen darf. Das macht ihr dann auch die Planung. Es hat Impact auf die Planung. Oder wenn du einen Sprint planst, aber keine Schätzung hast, wie planst du dann den Sprint?

Speaker2: [00:43:15] Gar nicht. Das ist eine gute Frage. Das ist wirklich extrem Wunder, was du da was, was, was da das Resultat deiner Arbeit ist. Weil eben vorher hab ich gesagt, ich weiß nicht, ob das Teil sind. Ich aus meiner Sicht schon, aber da müssen Sie natürlich auch genau gleich wie alle anderen in Ihrem Sprint, bis Sie sagen, okay, ich erledige bis nächste Woche da oder in diesem Sprint erledige ich das, das und das. Weil, wenn Sie das nicht können. Gibt es ja nicht.

Speaker1: [00:43:44] Ja, ich denke, Scrum hat ja heute mein Verständnis. Ich glaube, auch in der Theorie hat sie bewusst unterschiedliche Themen, die sie abdecken wollen, wie Transparenz etc. Aber das ist auch alles als Collaboration nutzen möchtest. Deshalb haben in meinem Verständnis die Gefäße auch einen gewissen Wert. Oder dass man zum Beispiel in einem Daily kurz auch über mögliche Obstacles redet. Wenn man frage, stellt sich für mich, wenn man natürlich nur keine Ahnung, vier Stunden in der Woche programmiert, macht es tatsächlich, an einem Daily teilzunehmen, wenn man keinen eigenen Fortschritt kommunizieren kann.

Speaker2: [00:44:24] Ja. Er wahrscheinlich nicht. Ah, gut. Es ist natürlich auch wichtig zu wissen, was die anderen, wenn ich Abhängigkeiten habe.

Speaker1: [00:44:33] Und ich finde das Thema Review. Also, dass man eine Review macht. Wie siehst du das? Sinnvoll? Wenig sinnvoll? Welchen Wert siehst du in solchen Reviews?

Speaker2: [00:44:49] Das lasse ich. Das sehe ich das sehr sinnvoll, weil schon nur ja dann auch für das Testing oder für Feedback. Der Vorteil ist ja dann auch, dass ich noch näher Business am bin, oder? Also ich meine, ich bin jetzt gar nicht ganz sicher, aber ich meine, der Business Analyst und der Product Owner, die haben ja dann auch ein ganz wichtigen, wichtige Rolle in diesen Reviews. Und wenn ich jetzt direkt noch jemand aus dem Business da drin habe, umso besser, oder? Also dann. Ja dieses typische Telefon Spiel. Da geht dann eine Leitung weniger. Aber es ist wieder, wenn dann der Citizen Developer erklären muss, warum er jetzt also ich kann es denn ja nicht so Challenge wie einen Entwickler im Review, was man im Review vielleicht auch macht. Dann ey, warum hast du das so gemacht? Es wäre doch besser gewesen. So, das ist ja für den. Den Developer ja dann.

Speaker1: [00:45:44] Vielleicht. Hat das Review dann auch eine. Eine andere, einen anderen Nutzen, dass man vielleicht Informationen teilt oder. Eskaliert.

Speaker2: [00:46:02] Ja, ich denke, das also. Und ich glaube, das ist. Das ist schon. Weil man schneller entwickelt als du schon. Das ist ja auch gesagt. Weil man schnellere entwickelt, muss auch die Information ein bisschen schneller fließen. Also vielleicht hat auch der Daily dann einen anderen Charakter, sage ich jetzt mal, wenn dann ein Citizen Developer drinnen damit macht. Vielleicht hat es auch mehr den Charakter von teilweise wirklich auch so wie ein Office Hour, kann mir mal kurz jemand helfen. Wie bekomme ich jetzt den Button grün anstatt rot oder sowas? Vielleicht kann das Daily ja für sowas genutzt werden.

Speaker1: [00:46:41] Ich finde es ein sehr spannender Gedanke.

Speaker1: [00:46:49] Für mich noch die zweitletzte Frage. Wo siehst du die Grenzen? Wenn du jetzt sagst die End User Programmierung? Es wird. Ja, es gibt. Ja. Also, ich muss schauen, dass ich nicht allzu sehr werte. Es gibt für Verkaufsfolien, die projizieren gewisse Dinge. Wo siehst du die Grenze des Citizen Development Ansatzes oder der Programmierung durch den End User? Und wo beginnt für dich eigentlich der Einsatz: Hier muss ein klassischer Entwickler ans Werk gehen.

Speaker2: [00:47:27] Ja, du hast recht. Das ist ein bisschen Theorie. Und Verkauf. Und was ist dann die Realität? Ich hab's ja schon gesagt. 80 %. Die interessieren sich nicht mal für Citizen development, würde ich sagen. Also ich sehe im Moment nicht. Nicht nicht aufgrund der Technologie, sondern einfach aufgrund von. Okay, jetzt baust du einen wirklichen Business kritischen Prozess oder wo auch Themen wie Security oder Integrationen ganz wichtig sind. Natürlich kann ein Citizen Developer das vielleicht bauen, aber wenn es dann sowieso komplett was. Die Maintenance und der Betrieb ist dann wieder in einem anderen Team sowieso, dann stelle ich mir ein bisschen die Frage okay, wieso macht es dann einfach nicht das Team gleich selber, oder? Also da sitzen die. Man muss auch Zeit aufwenden. Das Team ist ziemlich sicher schneller als der Citizen Developer. Da kommt dann höchstens ins Spiel Natürlich, es gibt einfach zu wenig Leute, in der IT die das bauen. Das ist dann das Problem. Aber ich sehe eigentlich wirklich nicht. Ich unterscheide da gerne mal, ich nenne das dann eben Personal Productiviy Applications. Und der andere krasse Bucket nenne ich dann Enterprise Great oder Enterprise Wide Enterprise, Great Applications. Und Enterprise Great ist für mich ganz klar kein Thema. Personal Productivity Application in Citizen Development. Jetzt gibt es natürlich einen riesigen Graubereich dazwischen. Und wo da die Grenze ist, die ist nicht hart, denke ich. Das hängt schon allein vom Citizen Developer ab, oder was kann er? Was will die Person erreichen? Wie viel Zeit hat sie? Aber ja, es ist auch wieder für mich sicher teilweise ein Technologie Thema eben wenn es um Themen wie Integration in Security und so weiter und so fort geht. Aber es ist auch da wieder hauptsächlich für mich ein Thema. Vom Betrieb, dann von Wer kümmert sich um die Applikation? Habe ich auch SLA's auf der Applikation? Oder wenn es jetzt Business kritisch ist, die muss wenn sie Wenn mal etwas nicht funktioniert, muss sie in einer halben Stunde wieder funktionieren. Zum Beispiel habe ich das ganze Citizen Developer Thema. Also das wäre das Risiko wäre viel zu hoch für die IT, da irgendeine Citizen Development Application die Verantwortung zu übernehmen. Dann für ein SLA 30 Minuten funktioniert die App wieder. Das würde ich nicht machen, als als.

Speaker1: [00:49:48] Das finde ich eine schöne Dimension ab dem Zeitpunkt, wo du es SLA's benötigst, oder? Dann musst du das auch. Ja gewährleisten können.

Speaker1: [00:50:04] Gibt es in deinem Netzwerk noch noch weitere Personen, welche ich im Rahmen meiner Forschung befragen könnte, befragen sollte?

Speaker2: [00:50:14] Ja gut. Ich weiß nicht, wie interessant das ist. Ich habe noch ganz viele Arbeitskollegen, aber ich könnte. Ich kann dir sehr gerne noch. Mit dem arbeite ich sehr eng zusammen. Thomas Gehring ist sein Name. Warum er spannend sein könnte? Er kam vor einem halben Jahr zu uns.Hat vorher jetzt auch noch sehr lange als Partner gearbeitet, Partner gearbeitet, bei Swisscom gearbeitet und Projekte implementiert. Also sein, sag ich mal sein Know how betreffend. Wie machen haben Sie das mit Scrum gemacht? Usw Es fällt sicher einiges frischer als meines. Ja, weil bei mir ist das. Das ist sechs Jahre her, acht Jahre her. Also, ich kann.

Speaker1: [00:51:03] Kannst du mir den Kontakt herstellen, dass ich dort anklopfen könnte?

Speaker2: [00:51:10] Mach ich.

Speaker1: [00:51:12] Das wäre nett. Herzlichen Dank. Und dann? Ich denke einfach auch im Sinne der Transparenz. Ich werde jetzt insgesamt etwa 8 bis 10 Interviews machen, die die Themen anonymisiert auswerten. Das sind keine einzelne Rückschlüsse auf deine Aussagen möglich. Und mein Ziel ist es dann wirklich, ein eher generisches Modell zu kreieren