Interview 10

Speaker1: [00:00:06] Okay. Interviewpartner, herzlichen Dank, dass du dir Zeit nimmst. Ist es für dich in Ordnung, wenn ich das Interview aufzeichne?

Speaker2: [00:00:16] Ja, klar.

Speaker1: [00:00:17] Passt. Ich untersuche die möglichen Herausforderungen bei der Low-Code- Softwareentwicklung durch den End-User und möchte dabei unter anderem auch schauen, inwiefern das in einem klassischen SCRUM Prozess abbildbar wäre und ob allenfalls Anpassungen erforderlich sind. Jetzt zu Beginn kannst du mir deine aktuelle Position und auch ein bisschen deine Arbeit beschreiben?

Speaker2: [00:00:42] Ja, klar, sehr gern. Ich bin bei der Firma X. Und um die Sachen, die ich mache, mal ein bisschen konkreter zu beschreiben. Ich arbeite nur mit Low Code von den Bereichen Projektmanagement über Systeme, Ingenieurwesen bis zum Thema Anforderungen Management im Prinzip das ganze Themenfeld und übernehme die Kommunikation zwischen Kunden und Entwicklern und auch Testern und dem Unternehmen.

Speaker1: [00:01:11] Und wann bist du das erste Mal mit Low Code Software in Kontakt getreten.

Speaker2: [00:01:18] Letztes Jahr im Oktober als ich angefangen habe. Bei den gibt es das. Bzw kurz davor. Weil ich mich im zuge der einstellungs entscheidung auch damit beschäftigt habe, ob es das ist was ich machen möchte oder nicht. Aber richtig intensiv dann ab oktober.

Speaker1: [00:01:36] Inwiefern unterscheidet sich aus seiner Sicht die Low-Code-Softwareentwicklung von der klassischen Softwareentwicklung?

Speaker2: [00:01:44] Das hängt wirklich stark davon ab, welchen Ansatz ich wähl. Der ich denke, das eine Ende haben wir. Da sind die Methoden und die Ansätze sehr identisch. Ich nutze nur eine andere Technologie. Das ist ein bisschen so, als ob ich Anstatt Java nehme ich die Chap mit verschiedenen Vorteilen. Auch da habe ich einen

leicht anderen Ansatz. Aber wenn ich jetzt die Scrum Methodik und alles was da rumhängt, den Entwicklungsprozess angucke, ist es aus meiner Sicht sehr, sehr ähnlich. Es geht. Das Ganze geht aber auch weiter bis in diesen Citizen Developer Ansatz. Und da gibt es aus meiner Sicht eine hohe Abweichung zu der klassischen Softwareentwicklung, was Prozesse angeht und auch das Rollenverständnis, vor allem, was sich daraus ergibt. Da sollten wir vielleicht nochmal im Detail eingehen. Deswegen ist die Antwort nicht so leicht zu beantworten, weil es sowohl als auch beides sein kann.

Speaker1: [00:02:36] Und welche Herausforderung siehst du, wenn jetzt der Citizen Developer Low Code programmiert?

Speaker2: [00:02:46] Ähm. Es gibt. Ich denke, wir müssen. Den Teil, der vorher auch gesplittet wurde, müssen wir auch weiterhin splitten. Ich habe auch in Projekten im klassischen Softwareentwicklung Umfeld Software Architekten, Developer, Leute, die sich um Cybersecurity kümmern. Spezialisten, die brauche ich auch weiterhin. Und da ist es wichtig, in der Organisationsentwicklung zu gucken Wie spielt sich meine Organisation auf und welche Rollen gebe ich wem und brauche ich eventuell Rollen, die ich noch zusätzlich hinzufügen muss? Und das ist ein absolutes Muss, dort den richtigen Split zu haben, denn kleine Applikationen kriege ich auch als Citizen Development hin. Ja, da habe ich eine Woche. Ich mache irgendwas mit Power Apps oder ähnlichem, verknüpfe ein paar SharePoint Tabellen. Das ist alles okay, aber sobald ich in den Enterprise Application Bereich gehe, brauche ich die gleichen Fähigkeiten. Fehler, die ich am Anfang in der Architektur mache. Die schlagen nach einem halben Jahr so hart auf, dass ich eventuell alles wegtun muss und noch mal von vorne anfangen muss. Und da liegt meiner Meinung nach die Herausforderung, so wie sie vorher auch schon da war und das auch nicht jeder handeln konnte, dieses Rollenverständnis allen klar zu machen und auch der Governance. Ich finde, es gibt kein gutes deutsches Wort dafür, um es abzuschwächen oder widerzuspiegeln, dass man das richtig treibt. Die Projekte, dass man dort die richtige Führung reinbringt und die Leute richtig leitet, ja, dass sich jeder auf seine Kernkompetenzen konzentrieren kann.

Speaker1: [00:04:19] Welche Risiken siehst du bei Citizen Development?

Speaker2: [00:04:24] Ja, wie ich es eben schon beschrieben habe. Wie bei jungen Softwareentwicklern. Die Leute gehen vorwärts und fangen an, die ersten Sachen zu entwickeln. Aber wichtig ist ja alles. Es gibt so viele Themen im Markt, gerne dieses Modell. Es gibt unbewusste Unwissenheit, bewusste Unwissenheit, bewusstes Wissen und unbewusstes Wissen. Es sind Stufenmodell, was verschiedene Reife gerade angeht. Und Leute, die gerade anfangen, bewegen sich auf der untersten Stufe. Das ist die unbewusste Unwissenheit und da liegt das Risiko meiner Meinung nach drin, wenn die Leute nicht die richtige Führung kriegen. Das kann ich nur über Erfahrung haben. Dann wissen die Leute nicht, was sie nicht wissen. Und es sind so viele Themenfelder drum herum und dann legen sie einfach los und haben dieses Learning erst, wenn es eigentlich schon zu spät ist. Das ist aus meiner Sicht in dem Citizen Developer Ansatz das größte Risiko, das zu haben.

Speaker1: [00:05:24] Kannst und hast du selbst schon mit SCRUM gearbeitet?

Speaker2: [00:05:30] Ja, schon seit vielen Jahren, also seit zehn Jahren.

Speaker1: [00:05:34] Wenn du jetzt daran denkst, dass du ein Citizen Development in einem Scrum Framework integrieren würdest, würde das funktionieren, müsste man dort Anpassungen vornehmen?

Speaker2: [00:05:45] Das kommt darauf an, wie stark die Person involviert ist. Ich glaube, dass damit steht und fällt das Ganze. Ich kann natürlich eine Citizenship Developer nehmen und als vollwertiges, vollwertiges Mitglied in meinem SCRUM Team aufnehmen. Das hat aber eine Grenze nach unten. Wenn ich die Leute nur zu zehn 20 % mit involvieren, dann das Ganze. Dann ist irgendwann mein Overhead so hoch in dem ganzen Scrum Prozess, dass ich den Benefit nicht mehr einsammeln kann. Ich denke da gibt es bessere Ansätze wie Kanban oder ähnliches, wie man dort vorgehen kann, wo man die Kapazität und Planung nicht so genau machen muss. Es kommt darauf an, dass wenn die Leute zu 100 % blockiert werden können für die Projekte, dann halte ich das für einen sehr sinnvollen Ansatz, gerade weil ich dadurch auch diese Guidance und das Coaching von erfahrenen Entwicklern sicherstellen kann. Wenn ich die Leute aber zu einem unsicheren Anteil im Projekt habe und sie vom operativen Geschäft nach Bedarf dann wieder zurückgezogen werden, dann hat man ein Riesenproblem. Weil für mich ist der Ansatz von SCRUM, dass ich im Prinzip ja doch

eigentlich eine sichere Planung mache für zwei Wochen irgendwas, wogegen ich mich committe. Und wenn mir dann aber die Ressourcen rausgezogen werden, dann geht dieser. Der Kerngedanke von diesem Scrum geht für mich verloren, weil. Der darauf sich bezieht, zu erreichen, wozu man sich committed hat. Und dann wird es alles in sich zusammenbrechen.

Speaker1: [00:07:16] Siehst du? Siehst du den Bedarf nach neuen Rollen im Rahmen von Citizen Development? Also, wenn wir uns jetzt mit SCRUM vergleicht, man hat dort den Product Owner und man hat den Business Analysten, man hat den Entwickler. Welchen Einfluss könnte das Citizen Development auf diese Rollen haben?

Speaker2: [00:07:36] Ja, ich sehe da ganz klar eine neue Rolle die Rolle des Coaches. Die gibt es aktuell nicht, so dass es passiert inhärent. Man geht davon aus, die Leute sind schön trainiert. Aber ich werde immer wieder den Fall haben, dass sich Leute onboarde. Wenn ich eine komplexe Applikation habe. Es steht für mich der Fokus beim Citizen Development eher auf dem Applications Wissen, also den Use Cases vom Business her, was ja dazu führen kann, dass ich mir immer wieder andere Leute rein hole, je nachdem, welches Modul ich entwickle. In dem traditionellen Ansatz habe ich Entwickler, deren Kernkompetenz das Entwickeln ist und nicht die Business Application. Das ist nur eine der Teil Kompetenz. Und da dreht sich genau. Das Ganze einmal um. Ich brauche nicht den Input von dem Product Owner in erster Linie über die Requirements, sondern ich brauche in erster Linie einen Coach, der dabei hilft, Software qualitativ hochwertig zu entwickeln. Und die Product Owner braucht du bestimmt auch. Der ist nicht ersetzt dadurch. Das will ich nicht sagen. Aber diese Rolle, diese Coaching Rolle spielt eine ganz entscheidende. Alle in der in der Thematik für den Erfolg.

Speaker1: [00:08:50] Da wäre für dich der Product Owner.

Speaker2: [00:08:55] Ja, das ist sowieso von Projekt zu Projekt ein bisschen unterschiedlich. Der Product Owner wäre für mich jemand, der den Gesamtüberblick behält über die Applikation, der wie vorher auch mit den einzelnen Fachbereichen die Diskussionen aufnimmt. Mit dem Unterschied, dass die Fachbereiche dann ihre Applikationen oder ihre Module selber entwickeln. Der geht mehr in Richtung Produktmanager bzw Projektleiter in dem ganzen, als dass er einen detaillierten

Software Wissensstand hat. Aber auch da gibt es ja sehr unterschiedliche Produkte. Ich habe welche, die sind sehr, sehr dicht an der Software dran. Ich habe welche, die fliegen deutlich höher. Ich glaube, da gibt es eher eine Verschiebung zu Leuten, die ein bisschen weiter oben sich bewegen, was den Detailgrad angeht. Mehr den Überblick behalten und vielleicht eher eine Rolle eines System Architekten einnehmen. Ein bisschen mehr als die von einem detaillierten User Interface Designer, was ja auch manchmal Product Owner übernehmen, das sie dann Mock ups entwickeln und sagen so soll es aussehen, denn die Sachen sollten ja eigentlich eher direkt aus dem Business kommen, aus den Fachabteilungen, durch die Citizen Development.

Speaker1: [00:10:09] Und wenn du jetzt an die Priorisierung denkst, wie sollte man dort priorisieren. Vor allem, wenn es jetzt kleinere Vorhaben sind.

Speaker2: [00:10:23] Sorry. Kannst du das noch mal wiederholen?

Speaker1: [00:10:26] Wenn du an die Priorisierung denkst. Vor allem bei kleinen Vorhaben sollte das stattfinden?

Speaker2: [00:10:35] Priorisierung wovon?

Speaker1: [00:10:36] Von kleineren zum Beispiel Prozessoptimierung.

Speaker2: [00:10:44] Es wird ja ein bisschen dezentraler das Ganze. Ich glaube, da liegt auch ein Unterschied. Wir werden auch in den Fachabteilungen ein bisschen agiler. Wenn die Fähigkeit, dort in den Fachabteilungen selber vorherrscht, Sachen zu entwickeln und umzusetzen, dann wird die Priorisierung auch nach Business Value stattfinden. Aber die Diskussion wird eine leichtere, weil die Fachabteilungen sich dann nicht untereinander austauschen müssen, wie sie die Priorisierung machen, sondern nur in sich. Klar, du brauchst diese Coaching Rollen, die müssen auch weiterhin bereitstehen. Ansonsten läuft es vielleicht in die falsche Richtung, aber das heißt die. Priorisierung nach Mehrwert für die Fachabteilung. Für die einzelne Fachabteilung wird wahrscheinlich deutlich besser, als wenn man sich mit 5,6,7,8 anderen Fachabteilungen austauschen muss. Wie ist es in der Realität? Der Faktor Wer schreit am lautesten? Spielt dann doch immer irgendwo eine Rolle. Und die nehme ich ein Stück weit damit

raus, indem ich das wieder in die Units zurück schiebe. Diese Verantwortung. Diese Entscheidungsfähigkeit.

Speaker1: [00:11:57] Die Grenze eines Citizen Development. Wo ist Citizen Development? Unterwegs und wo nicht mehr.

Speaker2: [00:12:08] Das hängt echt stark von dem Citizen Development ab. Wenn ich den Ansatz mache, dass sich die Leute nicht zu weit vom Operationellen wegnehme, dann liegt der Ansatz. Also dann ist die Grenze dort, wo die Architektur der Applikation anfängt, wo das System anfängt, wo ich Themen habe wie Cybersecurity, Datenschutz. Da ist dann irgendwann der fließende Übergang zwischen Citizen Developer und IT Abteilung, denn das ist so ein bisschen wie mit Entwickler und Tester. Also der Entwickler entwickelt Qualität, der Tester kann sie nie rein testen. Aber der Tester kann vorher Vorgaben geben, was er am Ende erwartet. Und auch da ist es wieder ein Austausch von Erwartungshaltungen und Unterstützung darin, diese Erwartungshaltung auch zu erfüllen.

Speaker1: [00:13:04] Und gibt es aus deiner Sicht noch noch weitere Themen, die für meine Forschung relevant sein könnten, die ich persönlich noch nicht angesprochen habe.

Speaker2: [00:13:18] Ich hatte es angesprochen, wie man die Leute einbindet in dem SCRUM Prozess. Das ist ja der Fokus von deiner Forschung, wie ich diesen Prozess anpasse, wie sie eigentlich die Zeremonien aus. Wenn ich dort Citizen Developer mit drin habe. Wie sieht mein Projekt aus? Weil das ist ja relativ leicht, von Sprint zu Sprint zu denken und sich den Fokus zu nehmen. Aber wie sieht denn jetzt eigentlich, dass das Portfolio oder diese Roadmap aus? Wenn ich jetzt 40 Sprints habe für eine Business Application oder ein modulares System, nennen wir es so? Wann bin ich denn welche Citizen Development ein und wie gehe ich dort vor? Ich glaube, das haben wir noch nicht. Das habe ich von dir jetzt noch nicht so gehört, weil ich werde wahrscheinlich immer wieder Leute haben, die kommen rein und gehen raus. Und wie gehe ich mit dieser Fluktuation um, die ich auch haben möchte? Und exakt die Leute in dem Projekt zu haben, die wissen, wie es am Ende aussehen soll, die diese Business Analysten Kompetenzen schon direkt mitbringen.

Speaker1: [00:14:24] Ja, die Themen, die ich ansprechen wollte, konnte ich ansprechen. Von diesem Punkte würde ich hier die Aufzeichnung beenden.

Speaker2: [00:14:38] Super. Vielen Dank.

Speaker1: [00:14:40] Ich danke dir.