Interview transcription - 2021-03-31- 10:00 over Zoom (remote)
Speaker 0 = interviewer
Speaker 1 = interviewee

- Speaker 0 00:00:00 All right. Okay. So, uh, basically, um, first of all, thank you for participating in the study. Uh, so the recording will be on this conversation will be recorded and transcribed. Uh, we will then forward the transcription, uh, at which point you can add or remove information where you feel comfortable. Um, so the goal of this, uh, insight, uh, the goal of this survey or this research is to gain an insight into how the practice of splitting a user story is done within the industry, and then comparing that to what the literature and other academic studies, uh, suggest should be so to speak. Uh, so, uh, it will take about an hour, I would say, uh, we're going to ask some general questions regarding you as a person first, just to get some demographic information and then we'll go into more technical, specific questions. So, uh, would you mind telling us a little bit about yourself in terms of, uh, experience within the industry and, uh, the company that you now work at?
- Speaker 1 00:01:34 Sure. Um, yeah, well, I, uh, I've been in the business of insurance since 2014. Um, and I have sprung from, from claims now in sort of business development, which I've been working with for three years, a total of, I guess, total of four years with a product ownership and, uh, requirements and, um, developing software with, with IT um, being in between the business and IT, um, yeah, so that's the, the business part of me. Should I jump into my personal profile or did you
- Speaker 0 00:02:19 No, just, uh, uh, I think it would be, I think it's fine.
- Speaker 1 00:02:25 Uh, I don't know how detailed you want it to be.
- Speaker 0 00:02:32 No, it's not really, it's small to know a little bit about your perspective and, uh, the, the experience, uh, in order to provide more weight to whatever you say, I think, uh, so I think that's fine. So what are you doing now at the moment at your company? Or what are your, uh, work, uh, tasks so to speak?
- Speaker 1 00:02:53 Yeah, well, right now I would say I'm not doing much at all. We are in between reorganizations, uh, the [COMPANY NAME] standard. So, um, I have been for the, like the last quarter, I've been product owning, uh, our squad in the current, uh, agile model that we are, uh, working with, um, um, a squad that is, uh, has, the main purpose of kind of driving automation strategy and implementing automation solutions that would improve our efficiency and reduce manual work and save FTEs. And ultimately also of course, uh, do something good for the customer journey. Um, but as of right now, as we speak, um, I am not longer a product owner. We have done further reorganizations and I am jumping into our new, uh, freshly shaped squad that is called, uh, claims services, which is a bundle of people now who are coming together to become a squad that is more kind of self autonomous and can be more in line with the actual organizational thought that we have, that you should be able to go all the way with your, uh, with your development. Previously, we were just kind of, it was an old team that just got named a squad. Um, wasn't capable of coworking the whole nine yards with automation. So if so our total realize that and decided to do a reorganizations and now put together people who can cover the whole kind of system landscape in one squad that would be able now to work with automation, uh, in the beginning of the claim journey. And that would be a squad member in that squad being like, I dunno, uh, I guess some kind of a hybrid between

a product owner and a requirement analyst. Uh, I think it would be more than that liquor and product owner will be more on, uh, on the political horse, riding around, uh, dictating meetings and trying to have the higher level agenda for us with more like on a PowerPoint level, more than an actual work level. And I will be in the squad think PO with all the ceremonies and such. Okay. That's what I believe anyway, but we haven't really started yet. So it's kind of in that sense, it's kind of poor timing for this, but yeah, we'll see where it takes us. I can still, uh, answer from what I, what I would expect this to become, I guess. So

- Speaker 0 00:05:38 The one I mentioned earlier, we will discuss user stories, user stories say term, which we have been, uh, uh, taught, uh, involves a kind of natural language, uh, written, uh, uh, requirements amongst other things. Um, so it's usually in the format of, uh, assay and then role, uh, I want to achieve some goal in order to have some benefit, for example, but in terms of this interview and our research paper, a user story is a little bit more abstract than that. So it could both involve a Epic, for example, it could involve a feature. It could also involve a user story if you have user stories at your company. So the interesting thing for us, and when we have this conversation is to remember that when we say use the story splitting that is going from something which is big into something smaller, and that doesn't necessarily have to be a user story written in a certain format. So it could be having a big feature that needs to be implemented and breaking that down into separate parts.
- Speaker 1 00:07:01 So it's more requirements splitting, then user story splitting?
- Speaker 0 00:07:04 Uh, I would say that's a valid point. Uh, so the literature and academic side is a little bit, uh, it's not that, um, researched, so there are some problems with the language, um, but you can think of it as any kind of reasoning and it kind of a process which involves dividing something bigger into smaller parts. Yep. Yep. Um, so in terms of what we have initially talked about, uh, how would you say that the hierarchy within your, either within your previous team or your experience, uh, how does the hierarchy of your use of stories or feature Epics? How does that look within your company or team?
- Speaker 1 00:07:56 Okay. Um, well, it's, it is kind of set from top down that like, we have business owners from the very beginning here with, uh, who are the top most, or this company kind of, uh, and they decide on like, what are the, what are the biggie things that they would like to see delivered in a period of time, bringing them a sprint for four months with us. And then that is then, uh, uh, derived down to, um, the, our tribe. And our tribe lead. And then together then with all the product owners, with each squad, uh, we then tried to set up objectives and key results to fulfill those things. And these, then all the things that we create Epic, Epics for, and that's still then the product owners and, and, and kind of the tribe that does the epics. There will be variations within each and every squad, how they work with their teams, given how kind of interested they are with the totality of the squad.
- Speaker 1 00:09:18 Some squads are all interested, or some squad members are really interested in doing their everyday work being. I'm a coder, therefore I code and nothing more. And some squads will have more of a, um, uh, I guess cross-functional, isn't the correct word, but then the members are more interested of every part of the totality of what the squad brings to the table. So there will be variations based on that. And I had to say that, uh, the idea is that everyone is involved on trying to shape an Epic in the squad to fulfill their objectives and curious

holes that we have on the tribal level. Um, and when those epics are created, it's still up to the PO. Um, and, um, and it's some sync then with the tribe lead that this is the episode we will bring in, and then these epics are, are then, uh, to be filled with, with features in our own, um, projects.

- Speaker 1 00:10:26 And those are then linked to the Epics on the board that is for the whole tribe. Um, so then it comes down to like the epics are then sort of delivered by a date. And then we fill them with features and sync with the other squatters to make sure that we don't have dependencies and things that can impede us in the work. And, um, yeah, then the features are filled with, with stories and kind of the, the splitting goals, um, in a way that is like natural, that we would try to set the epics as something that is deliverable by the four months that we are committing to. Features, I don't think that there's like a real, at least I would try to build them or have them as sort of quickly deliverable as possible to get something out. Uh, some squad has to work in Kanban or some work with a, with scrum and sprints within the sprints, which makes things that bit also, uh, complicated with how you do this.
- Speaker 1 00:11:36 So I'd say, uh, I'm not a big, big fan of the whole, uh, sprint thing in general. Uh, I didn't think that causes more problems than anything good actually, but we, um, we tried to set up features with something, um, valuable either for business or for ourselves technical wise, that is deliverable as soon as possible to, to work with that and iterate on it and test it and make sure that it's working and then move further with next features. And, uh, yeah. Stories are within this current team stories or kind of, um, um, they come in from, from various people within the organization, like business specialists, and more often than not, they are quite well-equipped with writing and requirements. So there doesn't need to be that much interaction or, um, or, or meddling from, from my side. Uh, so it can just be kind of prioritized and make sure that it fits to the features that we had. We have.
- Speaker 0 00:12:52 Yeah. So, um, I, to understand it, that you have an Epic with features with stories and those stories are what's actually being coded at, or are there even smaller parts?
- Speaker 1 00:13:10 Yeah. There can be tasks also that are kind of derived from the stories. Okay. Um, but there, those are more like things that are done by the developer, him or herself within the store to produce tasks, just to keep track on, what's supposed to be inside this story. So I'm not kind of as a PO, I'm not involved in, um, in the task creation, so to speak.
- Speaker 0 00:13:38 So, uh, would you be able to describe how this user story splitting is actually done, uh, within the team that you recently worked with or are now working?
- Speaker 1 00:13:53 Yeah. With the team I recently worked with, we had, um, I would say not an optimal setup. It's, um, it's kind of an, a, it was, it's a bit of an inward way that, um, it's, it's a very, very mature team that has been together for many, many years. And they are used to working in a, in an old fashioned way. They, they are doing agile in the sense that they commit to the servers and such, but they aren't agile in the way that the whole team can contribute to everything and requirements and stories. Um, at least when it comes to technical things or mostly dictated by one person, always the architect or that team. So as long as it's not pure business things like how our core system should be filled or, or those kinds of things, then the architect in this team would kind of design the story for the developers. So, uh, she would then in that design phase, both kind of write more in instruction based on how

it should go, uh, leaving no room for own interpretation or design. Uh, and she would also estimate that, like, how many story points would this ultimately be also then there from kind of saying that you have this amount of time to do it on otherwise it's a good or bad. And, um, yeah. She then would split, Uh, things into what she would deem deliverable as a story.

- Speaker 0 00:15:52 And when is this, when are these processes, uh, uh, done? So when it's actually uses story splitting, uh, done as a process,
- Speaker 1 00:16:05 Um, there is no, no ceremony for that. So that kind of happens, uh, every day, depending on what kind of, what requirements come in and what we have on the table to do.
- Speaker 0 00:16:16 Um, and w what do you have an example of something that could force a user story split to occur if it's on a, whenever it's necessary basis?
- Speaker 1 00:16:39 It, uh, if we have our requirements coming in from say a local business that aren't used to working with us or with IT or anything, um, it's common that they only kind of describe because they, they describe a feature that they want to have most of the times, which is what they kind of should be doing too. They shouldn't think about too much detail or trim down. They should only think about what they want to have working or, or more in the business lingo. And that kind of then makes us having to split things and kind of rearrange things a bit or redesign things to, to achieve that goal, uh, in a way that's also scalable for someone else, not only for them. Um, but many times there's there there's quite many people around our organization that, that, uh, kind of right away can write a requirement in the user story shape that is actually a user story that is understandable for the sake of business value and still, uh, in interpretable for our developer to start working with, without any like, need for, uh, further defining that or even splitting it. So it's a, it's a bit of, it can go both ways.
- Speaker 0 00:18:07 And you mentioned that, uh, one team had a one person who basically did all of the splitting. Yeah. Um, but is this a, so if you were to speak generally, who is involved in user story splitting, is that the case within a several teams, it would say, or is that a specific case, but not the standard?
- Speaker 1 00:18:34 This, this that I mentioned now is a specific case. I need to kind of keep a track on where, where I am now, mentally in this interview, because I might be drifting a bit bit in between teams because that's where I am right now. So I'm really looking forward to this transcript to see how confusing this will be. But, um, speaking of like my, my, my previous situation with that team, I think that is at least from what I've seen is fairly unique. And I, I hope, and I think that's good because, uh, I haven't been too keen on working that way. And I don't think it's, it's good for anyone really.
- Speaker 0 00:19:17 So why would you, is, do you think that there is any reason behind having certain roles or certain, uh, I guess roles is best to be involved in the splitting of a user story?
- Speaker 1 00:19:40 Um, I think I lost track of the question if there is a need for specific roles in my, uh,
- Speaker 0 00:19:46 Um, we're not really supposed to ask leading questions or I have to try and ask very generally, but so. Uh, why are certain roles involved in a use of storage split thing? If you understand what I mean? Is there a reason why

- Speaker 1 00:20:10 Aye, That it, um, it feels a bit, maybe also country-based like a more cultural thing that in certain team culture is it has become a natural hierarchy that is like this, and then also then time makes things worse.
- Speaker 1 00:20:39 I guess that may be in some sense, um, it is the belief that some people within a team or more qualified than others do these type of, uh, say big decisions of the design. Um, but, um, yeah, I know, I think it's mainly if you take one direction a couple of years ago, and then you just kind of follow it without looking over your shoulder and realizing where you're been heading. I, think that's the main reason there's no, I can't find a good reason for, for not involving squad members, being developers to have own design impact design decisions and the impact on how things should be done.
- Speaker 0 00:21:33 Uh, so if you were to put yourself in your colleagues' shoes, uh, what, what do you think that they would have to say about how the, um, how the results of this splitting process is for them? So to speak?
- Speaker 1 00:21:55 I would guess they're they think it's, um, and here's kind of the problem that I think they think it's good because they, they have for so long time now working in a way that they are used to just typing the code. They don't see a reality where they are, um, allowed to take one step closer to, to the business they are coding for, or like, they don't understand that I could make a difference with how I choose to type my code. They are only like, I don't know, they see the software developing as, um, if you would work in a car wash or like mowing the lawn, the lawn must be more, just sit on the mower and then it's done. There's no way of kind of thinking that should a mow it from right to left, left to right. Or whatever, to make a difference on how it looks. There's nothing like that. So they're used to that. And, um, therefore I think also they are like in, in the, the end game of their career, they don't, they haven't cared earlier on. I don't think they're going to start now. So it's a bit of a, it's a very specific case here.
- Speaker 0 00:23:15 Okay. Uh, so what would, uh, what would you say that you take into account when you actually perform the use of story splitting apart from the things you mentioned earlier, or if you would like to be more specific in terms of, uh, this question, because you met, you talked around it earlier, but, um, so for the sake of the interview and the flow, uh, what do you think you take into account when you split the user story?
- Speaker 1 00:23:49 Um, what I take into account is deliverability and business value too easy, but like try to do something that is Not only a smallest possible, it needs to carry value too, but it's kind of the combination of those two gives a factor that is kind of easy to compare one story over another. Should we do this or that based on this factor that I would receive of combining these two values? Um, yeah.
- Speaker 0 00:24:28 Uh, do you have any, do you have any specific either guidelines or techniques for performing this user story splitting? So this could be either something which is prescriptive in terms of guidelines within your company or something that you have picked up elsewhere.
- Speaker 1 00:24:52 Um, I don't have like a go to slide or a guide or a belief. Um, we haven't had any specific training. I mean, we product owners had training now in this new squad set up, but I mean, I've been working with for years with the same thing, without even seeing anything remotely related to being a training. I picked up one, um, off-prem training course in, in Stockholm, which kind of verify the way that we are, are working with. Uh, but it's like, I, there's no part during my working day

that I stop and pause and think that, should I use methodology or method B to fulfill this splitting or something? I mean, I, I think I have some, some books laying around in, in the office that I've picked up and of course, but I haven't ever opened them. I think again, at the cornea that maybe she has opened them. Um, so no, I wouldn't say that I use any type of methods that are prescribed by the scrum doctors

- Speaker 0 00:26:00 And there are no, uh, company company-wise, uh, guidelines that helps you, or that dictates on what should be.
- Speaker 1 00:26:13 Yeah. We have an agile playbook. I don't know what that is. I haven't opened it. I haven't read it. I don't, I haven't searched for it. I, uh, I think it's mainly something that I would use to say that if, if something would come my way, that I don't want to be doing, I could try looking for that playbook and see if I can use that to bat that thing out of my way, but it's not like I see that it's needed for doing my, my work. I don't get stuck. I haven't sensed that. Um, we are delivering things that aren't measurable or usable, so maybe we should stop and think that how could we split this otherwise? So there is a playbook just that I don't know exactly what it contains. I think it's more, a more higher level things and actual tips and tricks to do that. We have an agile coach that we can call and discuss these things with. I think that's the closest thing.
- Speaker 0 00:27:15 Uh, and in terms of deeming a user story split successful, uh, in comparison to one that might not be successful, do we have any definitions of what makes it successful and what are those or something that makes it unsuccessful and examples of those?
- Speaker 1 00:27:38 The double question? Um, no, I don't think me and you sort of story blurring to me has been just like, um, a natural thing that happens when you try to divide work within the team and have something festival. Um, so as long as everyone is occupied with their things, and we see that business is in a, in a shape where they can, uh, receive the, the chain shifts and releases and tested and everything is, is having a good, like, um, there's a good flow in the pipe then there's no, I haven't given one single thought to, did we split this good or bad? And how could it be otherwise? Um, I think that occurs much, much, much sooner in that case that when the developer starts working with it, they're like, um, this is, this is too heavy. And then it's kind of, uh, maybe, uh, we can, uh, refine that together right away. And as split that as we work or it's calls for, uh, refinements meeting within the team that this actually looked like a story, but it's more of a feature in terms of how much time it actually will take. So can we do this in some other way and then it's split and then it's just kind of forgot about, I would say there is no following up on that. Okay.
- Speaker 0 00:29:18 And in one of those refinement meetings, when you look at something which you mentioned might be, uh, in reality too big or too heavy, uh, what are those meetings? What do you discuss in those meetings? Or how do you take those, uh, how do you, uh, know which road to take so to speak or how perform this refinements?
- Speaker 1 00:29:46 Um, the goal is to chop it down as said area where with small pieces that would bring value. And then we would look at, I think, um, more like, um, are there elements that can be chopped? Like, can we divide this into pure UI elements or pure backend elements, or are there integrations for something so that one person can work with one thing and one person can work with the other. Could this, um, this kind of end-to-end feature that we now have as your story. Could that

be something that we could work with in a smaller part, but still vertical in the sense that every, every part like UI backend integration can still be tested or do we need to look at it more like let's first complete the UI and see this even usable for, for that person? Yes. That's an example of anything to discuss the splitting. Um, yeah, I think that about sums it up.

- Speaker 0 00:30:58 Uh, so you mentioned a little bit about that you do not look back too much, uh, but is there any, in terms of the agile methodology, there is a, um, there is a process or there's an aspect of the retrospective. So looking back at how things have been or worked out, uh, so is this ever applied to, uh, the process of splitting something in terms of making it from big to smaller pieces? Do you ever evaluate that process?
- Speaker 1 00:31:37 I don't think I ever have actually, um, retros, um, well with now the, my, my most recent team, those retros hasn't been so, so, um, alive, so to speak have been, uh, uh, felt a bit more forced, as I said before, more like we are doing agile rather than being agile. Um, and something as concrete as actually evaluating how things would have been a split now, taking concrete examples in the retrospective that certainly hasn't happened.
- Speaker 0 00:32:19 Do you think that this has a, do you think that this is a problem or is it, is this something you feel like there has been, or there's a need for actually looking back at this specific process as well to see if you can improve or if it's evaluated, if it's good enough?
- Speaker 1 00:32:40 I don't, um, I don't know. I kind of, um. I have never seen this as an issue. The issue I've had with this is on a, like the fact that you have to split things every once in a while, but then it's just, it's done and you get to work again. And I don't know, I was kind of wondering curiously, wondering what, what would come out of this discussion since I haven't been that much involved in, in, problems around this topic.
- Speaker 1 00:33:26 So, um, no, I am, I am not too alarmed or whatever word I want to use in, uh, in issues around storage spending. It's not a big part of my, it doesn't take too much of my, my work days to think of, uh, how it should be done or what can come, um, what, what, uh, negative effects can come out of it if we don't do it exactly correct the first time around or something like that. It's just, it's a natural thing that just happens to us. I know anything else that's not, not a thought too much about.
- Speaker 0 00:34:12 So to try and wrap things up, I have two more questions. Uh, so the first is, uh, uh, what do you think that the main, uh, benefits of performing user story splitting is?
- Speaker 1 00:34:31 As long as they do it within the whole team with, with developers that gives the benefit of everyone, knowing what we are working with, everyone is involved. Everyone can contribute to at least think about our things can be split. Um, even though maybe you are, you're only doing, say that you will only be doing front-end stuff, you can still be involved in the splitting to see what my front-end would bring to the totality and so forth. I think it's, um, it's good to, well, to get everyone to have. To have work items to be dealt among with the team, not only to have like, otherwise it would be there's one big story for, for one person to work with. And then I don't know, should, should other developers jump in on that story and create tasks? I don't think that's a, it looks and sounds a bit weird to me. So for the sake of dividing work, I think it's good for the sake of using the backlog as a

visualization tool for other stakeholders to see what's going on. It's easy to see just based on the, on the headings that is on the board, that, uh, these are the things that are done and prioritize accordingly. Um, and I just think it brings a better flow to have everything split as small but deliverable as possible. And of course, the idea that I don't think the business, at least in our end is capable of getting a release of everything right away. Uh, they are, they need to be dealt smaller chunks so they can test and verify everything in, in order and build up at the totality and be comfortable with that rather than having a big chunk, because then I think it will be hard to digest. And I think it's more room for error too.

- Speaker 0 00:36:35 Hmm. Uh, so, uh, paired with a benefit of something it's usually a drawback. So do you think that there are any drawbacks with performing user stories, planning, or have you experienced any drawbacks? Um,
- Speaker 1 00:37:01 I can't think of a concrete example. I would maybe, maybe it could be, it could be. So that actually, um, you take one big thing and you split it and it may be a couple of lines in the original story. A feature would be falling in between the chairs, which you won't see now that you have five stories that are split and you only do your part, you know, but, um, the idea that if your five developers in our team who has played a feature down to five stories and you do your own part, then there, there wouldn't should be awareness of what you're trying to deliver for the business. And there should be interaction between the developers as you do contribute to the feature, uh, when you work in parallel. So I haven't really noticed that things get missed out on just because we split it too drastically, so to speak.
- Speaker 1 00:38:01 But I think just thinking about it, I think that's something that could eventually happen that if we would do too much in this existing team, that where my, my most recent team then definitely that could happen because they don't, they are more used to, uh, working like this is my story. I need to do these bullet points. And when that is done, that goes to acceptance and it's, it's done. And then I can pick up the next door, which would be whatever, I don't even know how that goes with the business, but in a fully functioning team, I wouldn't be too afraid of doing, doing screenings, but more in, um, um, yeah, less functioning team. And then it could be a risk that you split it too harsh. And, uh, you miss out on totality because that's like also maybe the culture that you do your part, and hopefully that adds up to one in the end. Hmm. Okay.
- Speaker 0 00:39:00 So, uh, I, we don't really have any further questions, I think, uh, in terms of what we have discussed, do you have, do you think of anything that you are, you think are beneficial to add, uh, from your perspective that you might, that we have missed asking or that you would, that you want to highlight or yeah, as a closing remark.
- Speaker 1 00:39:33 No. Um, I don't know right now, I can't think of anything.
- Speaker 0 00:39:40 Uh, so, uh, I just received a question. Uh, so which is only, well, what do you prefer? Which work way, I guess, which process, which way of working. Okay. In terms of vertical and horizontal splitting, what do you prefer and why?
- Speaker 1 00:40:06 Well, if I had to pick one and not pull the old, that it depends on what we are doing, then I would pick a vertical to try and get a piece of everything to see how it adds up and how that is scalable for the rest to have something delivered to the business users that is actually easily more testable. Because if you just deliver one part to say that we have now built the UI element for you, you're

going to go work there. It doesn't connect to anything. So you still need to copy paste that into this and that and that place. Uh, it needs to do that for X amount of months then sure. You could gain some benefit in seeing how this UI actually works and looks, but that totality will be a, I guess, somewhat negative that people will already be sick and tired of this UI because it only kind of is linked with more work for you. So, um, if I had to pick one, I would, I would go vertigo, but I am, I still want to highlight that there are also issues without approach, um, uh, could be the less beneficial way in, in certainly instances too.

- Speaker 0 00:41:27 Thank you.