Interview transcription - 2021-04-12- 15:00 over Zoom (remote)
Speaker 0 = interviewer
Speaker 1 = Interviewee

- Speaker 0 00:00:02 All right. So first of all, thank you for participating in our study. This conversation will be recorded, uh, on, you will later be transcribed. Uh, we will remove any names that are mentioned due to confidentiality, and then our, uh, we will forward the transcription to you. So you are able to comment or revise if you would like to. Um, so, um, our research is to see how user story splitting is conducted and why it's, why it is being conducted in the industry. And for this interview specifically how it works within you, uh, for you and your team. Uh, so the goal of the interviews to get insight into how you and your team work with user stories splitting and why you do it as you do. So, um, yeah, I should also say that the interview will take up to an hour and, um, we will first ask you some general questions about you and your position at the company. And then we will introduce the topic and some definitions, and then we will move on to the actual questions regarding our research topic. So, uh, if you could tell us a little bit about yourself, what your current role is, and what's your previous experience within the company and outside the company?
- Speaker 1 00:01:34 Oh, of course. Uh, so my current role in the company is product owner, um, for the, all the, um, things regarding the online, um, part of the company. So everything that is either something that an end consumer can reserve online or something that we have chosen to have as an online, um, feature for users instead of having it in the main system. That's what I'm working with now, uh, combined with, uh, um, support, uh, role, basically answering questions from customers, calling in with everything, from, uh, pricing, uh, how to find, uh, functions in the system, uh, reservations that has gone wrong. That's what I'm doing today. Uh, what I, when I started at the company, I was working only with the support part and if we look on my background in it, it's very, uh, basically never worked with it before. So I had a system administrator role in a hotel before, uh, and then I'm actually from the restaurant industry working as a chef. Um, so, um, this is kind of new for me, so it's really interesting to, um, grow with, uh, um, with, uh, Rolanda the company. It's a very, they are working very much with, um, developing things for the customer and always trying to meet the requirements. So that's fun. So that's basically what I do today and a very short background.
- Speaker 0 00:03:35 Great. So a lot of experience in, within the business area.
- Speaker 1 00:03:41 Yeah, yeah. In the business area, but not like in the, uh, how to work with, um, the things that you're already interested in. So that's kind of new for me.
- Speaker 0 00:03:52 Well, it's all integrated, so that should be fine. Yeah. So, um, yeah, so introduction to the topic, um, just for clarity, uh, when, when we speak about, or when we ask you questions here, uh, regarding use of stories, it's not necessarily, uh, uh, use a story with the specific, familiar formats this as a user, I would like to, for some reason, but rather any kind of work item or user story as we will call it, which can be Epic feature task. And then, so what we also mean with user story splitting is that the process of which you and your team are dividing or breaking down a larger, uh, Epic or task into smaller ones. So, um, usually in teams, there are some kind of hierarchy, uh, which could be Epic feature user stories, and, uh, those

are combined together. So, uh, if you were to describe how that currently looks within your team, how, how would you describe that hierarchy?

- Speaker 1 00:05:15 Yeah. Um, we work with, um, epics for bigger, um, uh, issues that also are linked since we have two different, uh, we have the online team and we also have a team that works with our backend and the main system. So when they are, um, connected, so both my team and the other team has to work on the same because they are very dependent of each other. Then we use epics, um, because then we can put them into the sprint for this specific time, we will always work in two weeks sprints. Um, and we can just pick the specific issue that we want to solve, because sometimes they are too big to fit into one Epic, uh, one sprint or, uh, one of the teams has to do their work before the other team can start their work. So that, that's how we work with the bigger issues. Uh, if they are smaller, we sometimes work with, um, a main issue and sub tasks because we know that if I start the main issue, I will solve all the sub tasks within the sprint. So I can start that solve the sub tasks and then just close the issue.
- Speaker 1 00:06:39 And then if it's a smaller issue, we just create an issue and staple up what's supposed to be done within that issue. So that basically how it works, you have to like headers either an Epic or a task or a main issue. And then we have all of those that's just loose errands.
- Speaker 0 00:07:05 Okay. So, and is it correctly understood that an Epic doesn't break down into issues?
- Speaker 1 00:07:14 Yeah. Uh, the, the Epic is split down to many, many, many issues. It can be 10, 15, five if I know that they are bigger. So sometimes the Epic can have a issue that has a sub-task beneath it. Yeah. So they, so they are definitely issues beneath the Epic.
- Speaker 0 00:07:44 Okay. But it's also possible to directly create an issue?
- Speaker 1 00:07:49 Yeah. Without any Epic or some tasks or anything. Yep.
- Speaker 0 00:07:53 Great. So, um, in terms of more general user story splitting, um, could you describe how, how you do it within your team?
- Speaker 1 00:08:08 It depends sometimes we create an Epic in order to just collect, uh, issues that are related, uh, where we know that if I create an Epic and put all of them in the same Epic, um, we can have a time, uh, time wins. Um, we, we can be more effective with the hours that the developers have. So sometimes we go that way, we just take off one hour or two hours and just find errands that are connected to each others and put them in an Epic and in that way creates a smaller project. So that could be one way. And the other way, the more common way is that we, um, during a product meeting, figure out okay, which main issue should be developed within this two month period, five, 10. It depends. Uh, and then we decide, uh, on, uh, product group, um, products, um, project group of the different people in the company, and we discuss how this should be done.
- Speaker 1 00:09:23 Uh, and then we create, um, tasks for every task that has to be done in order for this work. So that I, um, I would say that that's the most common way now before the other one, the one where we just found all the issues and put them together was how it worked before. But now we're more like creates a pattern. This is what we want to do. And then we do all the tasks beneath and says, this is how we should, what we have to do in order to be able to, uh, close down the Epic and thereby send out the new development or the new feature or whatever. So that's how we do it today.

- Speaker 0 00:10:11 Okay. So more top down now than before.
- Speaker 1 00:10:17 Yeah, because we had too many, too many issues just lying around, uh, in the backlog and no one ever knew what they were. So now we know what we would like to do. And then we discuss with someone from my team and mostly someone from the other team, which is shows that has to be done in order to make that work.
- Speaker 0 00:10:40 Okay, great. So you, you mentioned sprint and I believe you mentioned sprint planning as well, but you also mentioned something like a product meeting.
- Speaker 1 00:10:52 Yup. That's more, um, specific to that, that we find if we on, um, um, project meeting decide, um, it's more me [TWO OTHER COLLEAGUES], uh, who is, uh, um, product owner, uh, support manager and, um, product owner. And we decide which way we would like to develop the, uh, product. And then we decide which, um, employees that should be in this meeting in order to get the best possible feedback on how to do it before we start to develop it. Because if we just have a, okay, yeah. For example, make bike rental. It's very hard to know what does that really mean? So now we, on the first meeting decide, what would we like to do? Uh, and then we create a meeting for that and basically a plan how it should be done. And then the next step is that we plan it into one of the sprints.
- Speaker 0 00:12:04 Okay. Okay. So a product meeting is more related to this, for example, background, the last you said, and then you fit a big part or some part of that into a sprint planning session.
- Speaker 1 00:12:20 Yes. Yes.
- Speaker 0 00:12:24 Great. So would you say that those are the times when you actually split an Epic or a, an issue or are you splitting epics or issues at any other time as well?
- Speaker 1 00:12:43 It can be done at other times as well. Um, if I do it as a product owner, uh, mostly I do with, with my development, uh, colleague, since he has more technical issues than I have, no knowledge than I have. Um, but basically anyone can divide or split, uh, issue at any time. There is no, we don't have any, um, anyone can do it so whenever, um, but if we want to do it with, um, like more of a project, we tend to do it in a more structured form because otherwise someone is just gonna create issues and never, ever go back to them. So, so the, the product owners should be involved somehow we're in some way, when we do the bigger splits, if it's just a smaller one, I need, okay, I need online to do this in order to make my work or on, in the other way around then they can just, um, do that themselves. Um, otherwise it would have been too much of a micromanagement, uh, environmental niche would be too hard for them or me to do the work.
- Speaker 0 00:14:04 Okay. Okay. So, uh, and in terms of this, um, Uh, maybe in relation to these types of, uh, either the product meeting or the spring planning session, uh, what roles are usually involved in user story splitting?
- Speaker 1 00:14:28 Um, since we're a very small company, it's very different because we, we can say that it, it's mostly one developer at one from the support team and the product owner. Um, sometimes there's two product owners and two, but mostly there is one from each of those like legs in the company. Uh, cause it's very hard. The support, um, staff knows how they would like to have it, but they don't know if it's actually possible to do it in that way. Uh, the, um, product owner, is there more as a, it's good to know what's been decided. Um, and also I have your way to

give input and say, this is not going to work because we are also doing this so-so they are there more as a moderator and keeping up so that the development doesn't get too big or too small, we need to have a balance in what we do. Well, maybe we should have it in more versions. And then the developer is there, um, mostly to, to give the technical information and say, okay, but if we do this, this is going to take more time and it will go this way. We can do that, but that's gonna end up in this. So, so I would say that these three people or these three different types of, um, um, colleagues should be in the meeting in order to make it work.

- Speaker 0 00:16:05 Okay. Great. And that was also a good description of why roles are involved as well. And that's really good. Um, what, what do you think. Do you think outside of your role for a minute, uh, what do you think your colleagues let's say a developer would say on how your story splitting are done within your team or company?
- Speaker 1 00:16:40 Um, if we look at the bigger part where we have an Epic and the tasks underneath, I think they would say that, that works kind of good, but if we look at the issues that are just created, um, I think they would say that sometimes it's hard for them to know, um, if there is something created during a sprint, should they attack that? Should they not attack that? Um, so sometimes I think they can, it depends also on the developer because some of them doesn't care. The, the issues are just created, then don't even bother to look at them and other, um, others could some, I think it's hard for them to find which issues that actually are connected to their, um, to their field, but that's just what I think they would say. I have no idea how they actually feel.
- Speaker 0 00:17:44 Um, so in terms of, um, like guidelines or techniques, uh, for, for splitting news to stories, are there any specific ones in use when you do this?
- Speaker 1 00:18:00 Uh, no. Um, we started using this, um, um, before everybody could, there are some guidelines, sorry, uh, before everybody could just, uh, create an issue. And then it was up to the product owner before we only had one. I was up to him to, um, in combination with the support staff to prioritize and, um, keep an eye of the backlog. Uh, now, uh, if I want to create an issue, uh, that I, I want someone to handle during, uh, already planned sprints. I need to, um, contact the product owner for that, like, uh, and say, this is a very critical issue. It has to be, um, handled. Otherwise. I have to decide if this issue is something that is just a wish for, from a customer. And then we put it in something that we call a freezer. So it's just a customer requests is nothing that we have the promised, nothing that we have to look at during, uh, a high season or they can, uh, do, uh, issue in, uh, uh, in the active, um, projects. And they can decide if they wanted, uh, they want us to lift it on the next planning or the planning after that. So that's basically before they have to, um, have in mind when they create yours so that they don't just end up in a backlog that no one even ever looks at.
- Speaker 0 00:19:45 Okay. Great. Um, do you know if you have a, sort of a definition of when you've done the correct split of an Epic or an issue?
- Speaker 1 00:20:03 No. There is no stuff. No.
- Speaker 0 00:20:08 Um, do you have any thoughts on why you don't have that?
- Speaker 1 00:20:12 Uh, I think because we are kind of new, um, uh, with working at this, uh, so we still trying to learn what works for us. Uh, and there is, we're still in a, um, what should I say, uh, where still in, um, time where the users of the, the system has a very big impact on what we actually do, which makes it hard to say if

it's actually right, because we don't really know which way is the best way before we tried all of them. So, so now there is no, well, I think there's very easy to say if it's wrong, but there is no way to say if it's actually right, if it's wrong, they haven't connected it to next sprint or the sprint, or they haven't contacted the product owner if it's a critical issue, but there is no way to say that it's right. It's just big for you to say that it's maybe wrong. So per definition, we maybe have a way to say that it’s right. I don't know.

- Speaker 0 00:21:33 That's a good answer. Um, so in terms of this, that you mentioned that you're fairly new to this, uh, sort of setup at the moment. Um, um, do you, do you know why you should change or try to implement this setup?
- Speaker 1 00:21:57 Uh, yeah, because, uh, before, as I said, anyone could just create an issue. Anyone could create an Epic, um, and that made the backlog impossible to handle and the, the, the way to prioritize the issues was nearly impossible. So that's why we decided, um, the two big things that we did was one was to split the product into, uh, different teams. Uh, and then also, um, give the product owner a big responsibility of the actual issues that are created. Um, so we had some issues that just laid in the backlog for four, five, six years, and no one ever looked at them. And then there is no point of creating the issue from the beginning, if we know that we will never even ever touch that issue. So, so that was the main point.
- Speaker 0 00:23:06 Okay. So with this new set up, are you trying to evaluate your current process and possibly improve it?
- Speaker 1 00:23:18 Uh, yeah. Uh, we have, um, um, meetings every week, uh, with the team. Um, and let's maybe once a month in that meeting, we'll lift, uh, if there is someone that thinks that the process doesn't process doesn't work. Um, so, so we're always trying to improve it since it's a new way of working. And, um, we've tried before have only some tasks, but that didn't work because when we put this into sprint and sprint looked so, uh, hacked up, so then we ready. I started working with Epic. So we were trying to adjust and have a very smooth and easy way, and it should be easy for everybody in the company to always know what's in the pipeline, because that was also a very big concern before no one never knew what was up and coming. Um, so, so we always trying to improve and new new labels and, um, pick if it's a bigger issue. And so definitely trying to improve the process and the, from the feedback from the team, because it's up to them to decide if they think they can follow what's in the pipeline. And what's have been, um, developed the last two or five weeks.
- Speaker 0 00:24:48 So, um, coming a bit to a closure here, but, um, what, what would you say that, um, the main benefits of splitting epics or user stories are,
- Speaker 1 00:25:04 Uh, for us it's, um, um, the documentation part, I would say because before we could have create, um, white rental and that was all of it, and then we just did that and close the issue, and then no one knew where had the developer been at, what was the demand from the customer from the beginning? Uh, so, so that's, and it's much easier to, uh, give a good time estimate from the developer if there is more tasks, because if you just have one big Epic with no sub tasks were no, no nothing. Then if they should, they will just say, Oh, I think, uh, 500 hours for that. They have no nothing to go on, but if it's split it into smaller tasks, we can get a better estimate. So we know how much we will be able to do for the next sprint. And it's also easier for us to just put the specific things from that header that

they are supposed to do instead of just taking their issue from sprint one to two to three to four, and then just moving it along until it's done. So it gets them, it gives us a more clear picture of what we actually did. Otherwise we would just have an issue that goes on forever and then B closes that, and that will just be presented sprint 10 or something like that. But we don't know what we actually did in sprint one to mine. So that's the, the biggest, um, uh, yeah, that was, that's what we find most useful for us.

- Speaker 0 00:27:09 Right. And of course, um, do you see any drawbacks with splitting larger user stories into smaller ones?
- Speaker 1 00:27:21 Um, sometimes not for us, because since we are one from the developer team, one from the product owner team, and one from the support that does this there's rarely any issues created that has that they don't touch, because I think that could be the big concern that if you try to make them too small, then sometimes it just creates issues and you get an even bigger backlog since you create too many more than you actually need. But since we are up to three people or even more, that actually goes through every, each, each and every, uh, issue that is created. And so for us, I don't think that we create when we have an Epic, I don't think we create more issues than we actually need, but I think that could be a concern if me, myself should just assume with petitions, I think should be done. But when we have a discussion, we actually just create the issues that has to be done for that specific task.
- Speaker 0 00:28:35 Right. And, um,
- Speaker 1 00:28:41 Yes,
- Speaker 0 00:28:42 I would guess that's pretty much it from my part. I'm going to check with a colleague here, if there's anything we see. Yeah. We're seem to be. Okay. So, um, yeah. Lastly, um, do you, is there anything that you have thought about, uh, during the interview that you might want to add?
- Speaker 1 00:29:11 No, I don't think so. Um, so it's an interesting topic and a good way for us to work. So,
- Speaker 0 00:29:24 Yeah, it seems like it's been rather beneficial for you to move into this direction.
- Speaker 1 00:29:30 Yeah, it was a bit, uh, massive before there was very many issues and no one even knew if it's something that has to be done or, uh, so this works fine. Um, there will always be more issues than, um, developers, but I think that's an industry standard more than a issue for our specific company, so, or crate, but, uh, yeah. Thanks again for participating.
-