[00:29] RESEARCHER:

Hello, can you hear me?

[00:31] PARTICIPANT 4:

Hello, Researcher. I was muted. How are you?

[00:36] RESEARCHER:

I'm very well, thank you. Yourself?

[00:40] PARTICIPANT 4:

I'm doing OK. I mean, considering the situation across the globe. All locked up.

[00:48] RESEARCHER:

What we have to do, we have to learn to stay home.

[00:53] PARTICIPANT 4:

At least for the foreseeable future.

[00:55] RESEARCHER:

Yes.

[00:56] PARTICIPANT 4:

I think it will change.

[00:58] RESEARCHER:

Yeah. Hopefully, things will change. I agree with you. Do you have any questions for me before we start?

[01:04] PARTICIPANT 4:

I think your guide explained things, explained what was expected. I went through that. So, I believe we're good to get started. Maybe if there's anything between, maybe we can discuss.

[01:16] RESEARCHER:

Ok. Fantastic. How about if you start by introducing yourself and talk to us about your experience?

[01:25] PARTICIPANT 4:

I have close to thirteen years of experience in the I.T. industry. I started my career with the services consulting company called [Deleted to preserve the participant anonymity] in 2007 where my role was more of a developer and I worked for clients like [Deleted to preserve the participant anonymity] etc. So, the programming model or the process at that time was more of iterative in nature. From there on I moved to [Deleted to preserve the participant anonymity]. It's the largest healthcare company in the world and they have the largest product market share. So that was my first introduction to Scrum. We started following Agile methods, but it was more of a “but Scrum”. We used to do standups, we did retrospective, but the essence of Scrum was not there. So, there was more product development plus I started doing Scrum master roles. The real opportunity where I got in-depth into Scrum was, when I moved to [Deleted to preserve the participant anonymity] That's a Fortune 500 company, which is in the manufacturing space. Again, I started with a technical role, but then being a manufacturing company, most of the manufacturing process had all those processes in place, but from a software perspective because let's say it was all over the place.

[02:59] PARTICIPANT 4:

So, I got the opportunity to do a lot of transformation from a typical Waterfall method to a more refined Scrum or Kanban. But I mean, like for the support products and I basically went with Kanban, and for the product development, it was Scrum that I followed. So, I started playing the role of Scrum master, I got certified in 2016 in Scrum. It was a two day course at that point in time. But the important work was the learnings that I took back, and I came back and started implementing the tools as well. Earlier it used to be some random tools which were not interconnected. Jira was introduced. So, we started following the frameworks of the tool itself. And so, my role over there was also a bit more diverse. I mean, in terms of when I used to travel to customer locations in the U.S. and discuss requirements. So, there was an opportunity to train some of the business people as product owners as well, which is very important when you look at those kinds of systems where you're working with the business.

[04:08] PARTICIPANT 4:

They don't care whether you actually implement Agile or Scrum. It's not a huge buzzword like in a typical software company for them. So, what it was like giving them training, coaching them from both the product owner side and the development team. So, there was a lot of training involved. So pretty much my role was more of a much wider breadth from a technical, business standpoint, implementing Scrum, playing the role of Scrum master. A little bit structured but that's what was needed at the moment. So, when I did those transformations, I got a lot of opposition from global stakeholders for that. And last year I moved to [Deleted to preserve the participant anonymity] which is a consulting company. And there I worked for clients like [Deleted to preserve the participant anonymity] and [Deleted to preserve the participant anonymity]

[05:03] PARTICIPANT 4:

So, what I see here is a bit more structure, the roles are defined. So, we had a specific people in the role who were experienced for that. And the product owner role, although it's a little bit debatable, it's more a proxy product owner that we had here. Because you are expecting more of a consulting role from the business side, but in the end, aspects of Scrum was followed. So, my role is more of a Scrum master with a team up to maximum of eight over here. And also, to drive some more of the program management aspects when you are driving some facilitation. So here is more like a bit closer to 50 percent goes into your Scrum. Somewhere around 50 percent goes into product management, your stakeholder management. You still make sure the scope, schedules are met, so some of the project management aspects as well. So that's what I do right here now. if you want me to get into depth on any particular thing, maybe you have some other questions that we will get into depth on each of the products.

[06:18] RESEARCHER:

I have a follow up question regarding your experience. It is interesting, your clients, you work for banks and they use Agile. Banks are known for being rigid structures and for being risk adverse some type of organization. How does Agile in this type of cultures work?

[06:45] PARTICIPANT 4:

So, I talk about two scenarios. There are two banks I work for. There is [Deleted to preserve the participant anonymity] which is quite huge in size. And what I would say is, I would say it's a little bit rigid, but they're very eager to kind of take the Agile journey. So, they have a uniform structure that's defined across the organization, [Deleted to preserve the participant anonymity] itself. They have their own Scrum masters. They actually kind of explored Agile a lot. But to your point, I would see some restriction in terms of what they want us to do. So, it is not flexible for you. You have to go through a lot of approvals when you want to alter their process. Something I would even say even it's the same case with the [Deleted to preserve the participant anonymity] it is a startup bank. It's a [Deleted to preserve the participant anonymity] which is built from scratch last year. So, we had more liberty to kind of implement the processes. I mean, we had to suggest them and then they'll come up with the regulatory aspects. But, for example we implemented CI/CD or BDD practice which is more of XP practices into our framework.

[08:06] PARTICIPANT 4:

If you look at Scrum it keeps that open as part of continuous improvement. Agile suggest you go with automation for testing/deployment. But I mean, like, if I want to do continuous deployment, you need a lot of sign-offs. So even if we had the capability to do CI/CD as part of the automation within the Scrum process, you don't have the approvals to do that. So, there are limitations and bits and pieces. But overall, what I see is most of the established banks and even startups banks, those are the ones who are very eager to get on board with Agile. So, they have the conversations a lot, but one of the challenges with [Deleted to preserve the participant anonymity] it was a remote Agile model. We were doing, like, the team was distributed. Part of the team was in the UK, so there are cases where we are looking at hybrid approaches. So, to your point, Researcher, it's more of a hybrid, a lot of customizations take back, but we tried to make sure that the principles of Agile were met. Does that answer your question, Researcher?

[08:29] RESEARCHER:

How do you define quality in the context of agile software development?

[08:37] PARTICIPANT 4:
Agile came in with a purpose to improve software quality and the team capabilities to deliver software. To be honest with you, after years in the industry, we do not know yet how to measure quality. But most people in the industry know what it is. I think the minimum is a software that meets the business needs and ideally free of defects.

[09:19] RESEARCHER:

Yes, fantastic. I'm just reflecting for the next question. The next question is a general question to start the topic and get into a little bit of depth. So, what do you think of Agile in general?

[09:37] PARTICIPANT 4:
So, Researcher, something that I think is like I've seen projects when... Agile is suitable, according to me it is suitable for 90 percent of the projects. There's still is going to be 10 percent of the projects. Like what is the framework that you want to select. That's your first decision. So, you really look at what is the technology. Is it complex, is it something like very simple. Something that can be done in like two weeks, one month. Agile it is suitable for complex projects, it is suitable for the average level of complexity as well. If it is chaotic, again, Agile is suitable, but like use Kanban, when it is a complex product or it is more of a, what do you say, more like a technology where you are exploring things, you are building a product, again you use Scrum.

[10:40] PARTICIPANT 4:
And if you are, like straight away like if you are very, like as a client, if you think that the requirements are very straightforward. My requirements are not going to change, it is only going to be a one month project. Still, there is no point in Agile. Waterfall is okay that's why I said that 90 percent of the projects are suitable for Agile. So, I support the approach. Coming back to your original question, I support the approach when the right framework is selected. Second you have to consider the fact that there’s always going to be that one project which doesn't require you always implement aspects of Scrum. I mean, you can always use the daily standup, you can always do the retrospective over the shorter duration is. But you have to again, understand that maybe some projects are really not suitable for it. So that making the right hybrid approach or selecting a framework is important and if in that case, it's going to be successful. That's why I support Agile because it did concentrate on maximizing the business value at that point in time.

[11:50] RESEARCHER:
Ok. Fantastic. Let's move to the next question. You can choose either your current job or your

previous job, it's up to you to answer this question. Can you describe your Scrum environment for us? The process, the people, how people work together, etc.?

[12:14] PARTICIPANT 4:
Ok, so I'll start my last project, [Deleted to preserve the participant anonymity] So, like I said, the project was to build a cloud based trade finance bank from scratch. So clearly, the bank did not exist last year. So, it is a huge project. And I mean, like in building an actual product, , we used Scrum. There were multiple tracks and you can imagine, the scope of building that app was pretty huge. I was the Scrum master for a couple of tracks. So, there was like elements of Safe that was used as well. Because multiple teams need to coordinate a little bit. But I'll come back to the two tracks that I had as a Scrum master. So, there were obviously like any other Scrum teams, there are three primary roles. I was playing the role of Scrum master, and like I explained earlier, it was like 50 persons project and the project management roles also varied, you had to have your stakeholders, you had to manage the release management, which maybe we keep this point when it comes to your other deployment questions.

[13:33] PARTICIPANT 4:
So I was doing the Scrum master tasks. Facilitating ceremonies. We started with grooming. Obviously, the standards events are planning, stand up, the retrospective and the client demos. So, let me just touch upon a little bit on how we did it. So, Scrum, as per the Agile guide, it should be time-boxed of fifteen minutes and we did the fifteen minutes which is used for a very specific update in terms of what you did yesterday, what you're doing today and whether you've kind of blocked something. This was a fixed point, so it had few challenges. We wanted to get going with still giving the team the liberty to self-organize. It was 30 minutes for us. So that's a bit of a deviation, which was done at Lloyds as well. The main aim was to kind of just time-box for 30 minutes, but then 15 minutes as the team became more mature. The rest of them were pretty good. So, the demo, we actually... OK let me just cover the roles before I get into the next point. So, the product owner is where, like I said, again, that is a slight difference. You should only have one product owner clearly.

[14:59] PARTICIPANT 4:

And I would prefer it to be from the business. But in this case, the product owner was actually from our side, from the consulting side. When I was doing projects for [Deleted to preserve the participant anonymity] I was part of that organization. It was more of an in-house thing. So, I mean, like, it is much longer term engagement, like for years. So, product owners can be a trained on that culture, build up on that culture. In consulting companies, they have a business analyst. And then I was kind of saying, can somebody from the business side play that role. Because he will take more ownership and be the product owner. He should mainly be the product owner according to me. What I see is that the customer pays a huge bill for the fact that you get a consulting expertise from a company like Sapient. So, the product owner role here is more like a proxy product owner from consulting side (BA)

[16:17] PARTICIPANT 4:
Because ultimately what does the product owner do. He talks to multiple stakeholders, he collects the requirements then prioritize it, then says what should go into the next sprint. So, this

person becomes the face for the team so that's the only person he sees but we actually get into discussions with the business stakeholder, this is the point of contact. But he doesn't really understand a lot of agile terminology, so we have a slightly different structure over there. Just from a project standpoint

[17:00] PARTICIPANT 4:
Then we have the development team. So, the development team had two tracks. Eight member developers on each track. There was one person who was an expert on QA and there was one person who was an expert in terms of designing applications. The designing application person was shared across teams. So, this is what the structure of the team was. I'll stop for a moment to ask if you have any questions on the structure of the team.

[17:35] RESEARCHER:

I do have follow up questions, but we can do the follow up questions once you're done. Just keep going.

[17:43] PARTICIPANT 4:
So basically now that we have defined, what are the roles- product owner, the BA, she runs all the grooming sessions, he runs all of the planning sessions, the way its facilitated, her primary responsibility for driving the grooming and the first part of the planning. The clarity part. If you have shaping, that's fine, also let me just take it in a statement. The first part is, once the requirement comes in, do we need clarification. The person makes sure it’s structured into an epic or user story. Now the story needs to have clear acceptance criteria. And once he has that the product owner might feel like are there more questions in terms the feasibility on will the feature work. In that case, I'll get in touch with him. In that case, I'll setup the shaping session with the architect, maybe the QA will get involved and the product owner (customer side) will get involved just to make sure that any initial questions are answered. So, shaping, I try get the whole team into the shaping as well. It is not an official Scrum ceremony. This is just what we do as part of the planning. If there are any questions, those get answered. And once the acceptance criteria is clear for the story, grooming is setup, which is again scheduled at the sprint start.

[19:36] PARTICIPANT 4:
There the product owner will clearly explain the requirements. Now, it's just a story, right. It's in a story format now. The acceptance criteria is clearly explained. The team asks any questions if they have. At the end of the grooming, the team should have a really good idea and they should be able to size the stories. So product owner can leave the team to do the planning unless the team has any further questions.. For the planning meeting, I'll just make sure the team, for the sized items, the team will talk about what are the tasks to be done for that to achieve that goal. They forecast what there can be done for the next sprint. So that is what happens on the planning meeting. Then obviously, the daily standup, like I said, there's a slight deviation on the time-box. We changed the time-box to accommodate some aspects which may be again when we get to touch on some other ABS or fixed price project and it has its challenges you need to have a very close tab if you are kind of like falling short somewhere on an impediment.

[20:50] PARTICIPANT 4:

So, the team is self-organizing, we make sure we don't support that part where it goes into typical directing structure. I mean like the old structure where we have to do this by end of the day kind of thing. We still kind of play around a little bit with the capacity and a little bit in terms of the priority. So that's what we do. So, at the end of the Sprint so that's again I'm talking about a slight deviation it comes to Demo. Retrospective obviously with the team it is done. In terms of the demo, there is a slight difference. The end customer, it is only the product owner, me, and the QA typically which attends the client demo, with the business stakeholders, but obviously there is an internal demo, for the team, I mean like when I say the Scrum team, that structure is followed. The reason being again, I mean what we understood is over a period of time, again, we needed a lot of training for the team to run demo to actual client’s hence internal demo.

[21:56] PARTICIPANT 4:

I mean when you talk to business stakeholders, you have to use the business language. So again, since it is a consulting kind of environment, I mean in terms of getting the entire development team to run through their stories itself, so we still create that structure but there is a little bit of which I agree, if you go to Agile purists they will disagree to this. This is something we are working towards and I believe we are pretty close to it. But I mean the sometimes we make a little bit of deviation looking at the customer, if they don't understand Agile, As in Dover, I trained my product owner. I traveled to the offices in the US, I had the opportunity to set up meetings with them, run them through the exercise, and train them in Agile so that they are ready. But in this environments, there's a slightly different approach just to make sure, we don't lose the grip of the project if you get what I say, right?

[23:18] RESEARCHER:

So, PARTICIPANT 4, do you think this setup is a good implementation of Scrum, and why?

[23:26] PARTICIPANT 4:

So, Researcher, like I said there are always ways to improve right. The Agile processes itself is based on kind of continuously improving at the end of the sprints. Some of the things, like I said, what will work for one team, might not work for another. And the different people...and the places where I see agile fail, is where people think it's a methodology.

We are not perfect. I believe we evolved into a process. I mean even within the framework we evolved into a process where it produced results. So, it was there to see and there were a lot of metrics that we captured for project which proved it to be successful.

[24:25] PARTICIPANT 4:

There is a tool called Hygiea in the market which did Agile metrics. So, what [Deleted to preserve the participant anonymity] has done internally is they took that as a base and built their own product called [Deleted to preserve the participant anonymity] It derives all of the agile metrics from Jira itself. And so, you have ways of comparison within the organization, so what is your maturity, so our project ranked well there, so the process I'm talking about. Like I said whether to do the shaping and having entire team did not work so we need a clear workaround by the next sprint. So, there was a core team like QA, BA So, there are multiple scenarios due to which, and that's one of the reasons since you are researching on it. I would like to hear if you have any thoughts on that . The idea was like say we have to keep evolving and customizing our framework. But at the same time ensure you are not breaking the Agile principles, right?

It's not okay if you don't meet the definition of ready, it's not okay. So, we don't do that. We just kind of make alterations. So, it's a successful one for me, so far.

[26:07] RESEARCHER:

You mentioned something, I'd like to follow up, make sure you don't break agile principles. How do you do that sometimes when the team or the organization itself clashes with agile principles?

[26:25] PARTICIPANT 4:

So that's a really good question because I mean there are cases where you need to push the max, but I'll let me just be honest with you again comparing my both my experiences. In organizations like [Deleted to preserve the participant anonymity] where I was part of the organization itself, right? Because it's much easier there, I think. So, to your question, what I found really tough was in organizations like [Deleted to preserve the participant anonymity]. Where ultimately most of your projects will be a fixed price project. I mean like say, you still prioritize and can go wrong. The organization also takes care that you are running Agile, and have a clear DOD defined. And then you know, there are legal implications if you do not deliver. DOD says all of the acceptance tests should pass for me to release .

[27:53] PARTICIPANT 4:

Imagine here are some unplanned events which happen. If somebody falls sick and there was a slight delay in delivery, but you are playing around here with Agile. You might choose to reduce your DOD for a sprint and add it as a tech debt though it is not suggested.

[29:23] RESEARCHER:

Yeah, it does. It's a question that we can spend days talking about. It's a complex question.

[29:32] PARTICIPANT 4:

Because when I talk to my directors, sometimes we also have this debate in terms of... Are we Agile if we are doing this? So maybe we'll just move ahead for time being . So that comes across sometimes.

[29:59] RESEARCHER:

So, what do you do to assure software quality in this Scrum setup? You can choose the same example you talked about.

[30:05] PARTICIPANT 4:

Now the software quality assurance starts from the day one. From when the project starts, it's not like you do all the development and then you go to testing. So, the first part is like I said, the clarity in the requirements that's the first part. So, you are making sure that acceptance criteria is clearly defined. Before it even comes into the sprint backlog, before the team takes it. It's when it's really defined. And coming to the next part now, you have the acceptance criteria the team starts with writing acceptance tests. So that's one part that they are expected to do. We follow BDD which is again extreme programming practice. But it doesn't stop you from implementing in a Scrum environment as well. So, the team typically does acceptance criteria first. Even at the point where the story's entered into Jira.

[31:24] PARTICIPANT 4:

Once the acceptance criteria is written since like I said that when you've done the internal, the PO is involved in terms of writing the sprint's goal & acceptance criteria as well. I mean just like it's clearly structured so that when the team writes the acceptance tests, they are very clear how to write it. So, the team starts writing the acceptance tests again, I see little bit of deviations for some teams. Initially they will start by writing the code and then write the acceptance test. But the ideal scenario we start by writing the acceptance test and the unit test and then more of a test-driven development so that ensures the quality. The unit tests ensure minor aspects are not missed out. We had maybe like five environments because the team develops on the dev environment, they deploy on the QA environment. So, during the course of the Sprint once you deploy into QA, you are doing a continuous integration.

[32:39] PARTICIPANT 4:

You are ensuring to make sure the quality is being met. You are ensuring the acceptance tests are passed. That's a prerequisite to kind of move to QA. And so, and then once all the tests are passed, it is automatically pushed to the QA environment. And again, they're like I said, we use assure DevOps so on the CI aspects making sure the code compiles. QA will have already written the integration test. I mean as the team was developing. Again, I mean we can see the role of the testing is different now. Now, you use more of a test engineer, he'll be kind of doing the automated testing. Basically, the integration test, and feature level test, he would have been planning for that. That would be written. The integration test would be run on the QA environment.

[33:58] PARTICIPANT 4:

So once the QA environment tests are passed, what we are doing is then we are pushing it to the UAT environment. So, my Sprint DOD actually means, in this scenario, like I said because we need a lot of validations on UAT from client to push it to production. we are not automating that complete deployment to production. So, we do a T minus 1 deployment, see for the current Sprint, whatever is done, the end user gets one week to kind of verify or validate on UAT to see everything is fine. So, in our case, that part was not automated or anything. The testing also does a sanity once everything is deployed. It automatically deploys, there's a sanity and hands over to the business to kind of test. So, they do a UAT testing, typically one week. So, once they sign off that's where I mentioned, it is more of a continuous delivery deployment and ready to release the packages are all created. It's just a click of a button. You could have just automation to push to the production as well. But once we give the UAT sign off, again then we move the code to production. So, this is the process, so I believe there are extra point the multiple points. To summarize this testing internally, acceptance tests are written. We do ensure the code does not break. It doesn't go to the higher environment without acceptance being passed. Then there is the UAT testing which is manual. There are sanity tests, integration tests happening on the QA before it goes to the higher environment. So that's in a summary, Researcher.

[35:55] RESEARCHER:

Yeah. I have follow-up questions. You mentioned that the testers comes into the picture once the code is pushed to the QA environment.

[36:11] PARTICIPANT 4:

They come into picture from day one. They don't come into picture only at testing time QA are also functional experts and they are not just purely testing guys, like maybe earlier times. Let me just touch on the part of the QA first, QAs come from the point there is a shaping, when the product owner is evaluating a requirement QA is involved because he will help refine the acceptance criteria. He is expected to bring out scenarios which might not be thought through by the product owner. So that's where my product owner leverages the QA role. So that's the first place where QA gets in, so he starts even prior to the development team according to me because of the value he brings in at that stage itself.

[37:22] PARTICIPANT 4:

Next part is the acceptance test. When I said the QA, or the quality engineer role has changed a little bit. The developers will also write the acceptance tests. So, the role of the QA in agile has changed a little bit because he becomes more important to kind of validate or review in most of those cases. And at the beginning of the Sprint itself, his role is more on to ensuring the overall process goes fine. It’s no longer an entry role for me. The next part is like during the course of the sprint, now there is a feature level thing. I mean like this is a scenario, we usually come across. Like say now we are building a payments side of the bank. And there are multiple COTA applications involved. I mean like the developer is worried about his story, but who is going to worry about the feature? And so, the QA from the beginning of the sprint though he'll pitch in were required on the acceptance test, story level testing. For me, that is more of an additional support he is providing but for me the feature level, making sure the integration works fine, automating those test cases becomes his primary one where he'll pitch in were required to support the functionality and acceptance questions for the team as well as a consultant. They will also be involved through the Sprint in terms of making sure he's ready once it is put into the QA environment. So, where he will be running... because the acceptance tests have already passed. So, the story level will be fine, and he has also validated and reviewed it.

[39:21] PARTICIPANT 4:

Does it work as a system, that's what he should be worried about more and the next one is even what I see is like about UAT testers take a lot of the support from the QA. We need to say for the UAT testing as well. So, he like I said again when we do the specific stakeholder demo, it's just the QA that usually runs these things and so that I mean he was the point of contact to kind of answer any question.

[40:09] RESEARCHER:

No, we're doing very well, thank you. I would like to follow up in another statement. You mentioned that the testers role has changed in Agile and they become more of reviewers of the process and the artifacts. Are they also empowered in the process to do their job?

[40:36] PARTICIPANT 4:

So again, I mean again there is a slight difference that I see there as well. Because I mean they are empowered like every other team member. So, we don't like to make a distinction between as a developer as a QA engineer. Let me just tell you an interesting aspect. When I joined, when I'm talking about 12 years back, the appraisal, the performance review of a tester was kind of based on how much defects he raised. And the developer's appraisal or the performance review was based on how much less number of defects he wrote. So, you can notice there is a conflict of interest, right?

[41:21] RESEARCHER:

Yes.

[41:22] PARTICIPANT 4:

So even if the QA person sees a defect, he'll wait. Because I mean he is incentivized if there are maximum number of defects in this QA environment, and there is very low on production. So why I stated this example is because it drives a cultural enmity, there will always be enmity in that kind of structure. So that has changed. So, what we encourage is at the team together if they don't have any issues on UAT, so that's a metric. So the QA roles has changed. They are empowered like any other person. If they feel the quality of the product is in question, it's like any other person, they can stop the release. And there isn't distinction to answer your questions. I hope I explained that and if you need more clarifications.

[42:30] RESEARCHER:

So, you emphasized the importance of engaging the QAs early on the process. Then, how does this help achieving better quality?

[42:35] PARTICIPANT 4:

Good question! They become intimately knowledgeable of the business requirements and the expectations. This helps testing the real users’ expectations rather than making assumptions. This knowledge helps identifying and reducing bugs.

[42:35] RESEARCHER:

Yes, that's was fantastic. I'll jump a little bit because I'm running out of time. Can you share with me a negative story about Scrum or agile? You've been talking about it in a very positive sense.

So, can you share with me a negative story? It doesn't go all rosy, it doesn't go all perfect all the time, unfortunately. Yeah, so.

[43:05] PARTICIPANT 4:

There is an ideal world scenario that we talked about. I mean like it's not an ideal world scenario. The team is much experienced, the team I worked with at [Deleted to preserve the participant anonymity] This was a prestigious project for our company because I mean like it was building a bank in six months. So, the team members they were mature enough. The assumption that Agile makes is the people are mature enough, which is not really the case in some scenarios. So, I will be very honest with you.

[43:43] RESEARCHER:

I agree with you.

[43:45] PARTICIPANT 4:

So that happens in a project like [Deleted to preserve the participant anonymity] where our company itself had to market it to industry. It works. I will tell you the least experience of team member working on this project was 8 to 10 years. So, you can imagine, and they have been within the system for some time, all very matured. You can drive those conversations with them. But let me tell you there are projects where - you lose a lot of time in fact in trying to set up the process itself because you will receive all kinds of questions. Whether daily tasks, and one person will come and say this process is actually slowing me down. I mean like say, the retrospective is not good. So, you will lose a lot of time, cause like that training part is huge and to your point Researcher, you can't just go in and go with Agile for a couple months of the project. I can assure you that because the team itself you have to accommodate for the... time team will take it up, seeing the value in maybe the process itself. I mean the team is not committed. You can see the difference the way they give the update starting from the standup. I mean like say, today I did this, tomorrow I'll do that. That's fine. You don't get to know the actual, it has to be transparent.

[45:28] PARTICIPANT 4:

I have run through those scenarios where short-term projects, I mean like the two-month project, a team, the expectation, even if you look at the Agile guide, they have changed right? Earlier it was very clear -no manager, you don't need an Architect, no need of roles. When it comes to SAFe, they brought back the architect. So, all of this, I mean in a nutshell it is with the assumption people are really self-organizing, which is not the case. In half the projects I would say maybe in my experience, those projects if you talk about issues and somebody expects you to deliver two months Agile. I will tell you and straight away be very open with my leadership. Say, this is something we will have to either go light on either of the cases whether the team will need some time to evolve, mature, once they are ready.

[46:42] RESEARCHER:

Thank you very much PARTICIPANT 4. I'll have to conclude it here because I have another interview after this one. I need to prepare myself for it and take a little bit of a break. Thank you very much. Do you have any questions for me before we conclude?

[46:57] PARTICIPANT 4:

So, Researcher you talked about the paper you were intending to publish. I mean if you don't mind once you're done if you don't mind sending a copy.

[47:07] RESEARCHER:

Yeah, I will. I will put you in a distribution list and I will send you a copy. Unfortunately, these paper takes a little bit of time to produce. The whole interview need to be transcribed and the analysis of the content of the interviews and the writing and it needs to be accepted in a software engineering news. So, it may take time, but you will hear from me. Definitely sure.

[47:36] PARTICIPANT 4:

And do you have any feedback for me Researcher? I mean like anything that you observed can be improved.

[47:40] RESEARCHER:

You're very knowledgeable of your process. You're a very mature professional. It's shows in the interview. It does show the professional maturity does show in the interview. Congratulations. I mean, it looks like you have a reached a great milestone in your career, which is really good. The way you articulate yourself, the way you talk about Agile, you talk like a professionals. Well done.

[48:15] PARTICIPANT 4:

Thanks a lot Researcher. And if you have any queries or anything further, please feel free to reach out.

[48:21] RESEARCHER:

We do some have some quality assurance stuff we do as well as researchers. I will transcribe the interview and send it to you for validation. Just skim through the interview and make sure that I transcribed exactly what you've said, and everything is okay.

[48:43] PARTICIPANT 4:

Sure, I'll do that.

[48:45] RESEARCHER:

Thank you very much PARTICIPANT 4, I wish you a good day. Bye.