00:10 PARTICIPANT 30:

Morning Researcher.

00:20 RESEARCHER:

Hi, Participant 30. How are you?

00:22 PARTICIPANT 30:

Hey, busy, busy day even if it's Sunday. I'm good and self?

00:24 RESEARCHER:

I'm doing very well. I'm just going to give few minutes for people to join us and we should be okay to kick off.

00:34 PARTICIPANT 30:

Wonderful, wonderful.

00:59 PARTICIPANT 38:

Hi, Researcher.

01:02 RESEARCHER:

Hi, PARTICIPANT 38. How are you?

01:04 PARTICIPANT 38:

Yeah, doing good. How are you?

01:06 RESEARCHER:

I'm very well. I'm just expecting more people, so it will take a few minutes before we kickoff.

01:19 PARTICIPANT 38:

Hi PARTICIPANT 30:

01:19 PARTICIPANT 30:

Hi, pleased to meet you PARTICIPANT 38.

01:27 RESEARCHER:

PARTICIPANT 31, how are you?

01:29 PARTICIPANT 31:

Hi everyone. I'm good and you?

01:32 RESEARCHER:

Good. I'm good. Hi Participant 36, how are you?

01:34 PARTICIPANT 36:

Very good, Researcher. And you?

01:50 RESEARCHER:

Good. PARTICIPANT 38, hi. How are you?

01:53 PARTICIPANT 38:

I'm doing good. How are you?

03:47 RESEARCHER:

Doing well, thanks. I'm waiting for Participant 37.

03:51 RESEARCHER:

Here he is. Hi Participant 37. How are you?

03:54 RESEARCHER:

Okay. Let's kickoff. I'll do some introductions and we'll start with the slides and statements. The purpose of this focus group is to get your input into the findings of the research. So, we analyzed the interviews and we have made some findings. And we like to take you through these findings and to get your input. Basically, they are six statements which we have taken from the findings section of the paper. And we like you to discuss and to tell us whether you would support these statements, or you wouldn't support them. And we would like also a justification of your decision whether to support or not to support the statement. We will do some brief introduction just so people can know who is around. I'll start with myself. Just the name, what you do, and where are you calling from.

05:04 RESEARCHER:

My name is [redacted]. Okay, Participant 30, would you like to introduce yourself briefly?

05:18 PARTICIPANT 30:

Yeah, definitely. My name is [redacted] and I'm a software engineer. I'm based here in the UK.

05:25 RESEARCHER:

Thanks Participant 30. Participant 31?

05:28 PARTICIPANT 31:

Hey everyone. This is [redacted] and I'm the senior software developer and I had about 17 years of experience and I'm based in Denmark.

05:40 RESEARCHER:

Okay, Participant 31. Thank you, Participant 36.

05:44 PARTICIPANT 36:

Hello, my name is [redacted]. I'm also a senior software engineer from the UK.

05:54 RESEARCHER:

Thanks, Participant 36. Participant 37?

05:57 PARTICIPANT 37:

Hello guys, I'm [redacted], I'm senior software engineer and I'm from Sydney, Australia.

06:05 RESEARCHER:

Okay. Thank you. Participant 38?

06:09 PARTICIPANT 38:

Hello everybody. My name is [redacted]. And I'm also a software developer form the UK.

06:28 RESEARCHER:

Thanks Participant 38. I'm going to share my screen and hopefully you can see my slides, the first slide says Quality in Scrum. That's what you seeing?

06:49 ALL PARTICIPANTS:

Yes.

06:50 RESEARCHER:

Fantastic. Any clarification or questions before we start? So, will go through the slides, I'll read the statements and we can discuss and get your feedback. Okay, it seems to be clear. So, let's start.

07:17 RESEARCHER:

This an overall summary in this slide. Scrum teams achieve quality by adopting several software engineering practices and quality assurance processes. We found various practices used to achieve quality. For example, code review, unit testing, static analyzer, continuous integration, etc.

The Scrum method helps teams to achieve quality by promoting social qualities in the development environment. These are collaboration, psychological safety, accountability and transparency. In addition, some process features of Scrum like iterative development and inspection and adaptation also promote quality.

08:49 RESEARCHER:

This is a broad summary we will go into the details shortly. For this summary, I just need your feedback and comments, but for the upcoming slides, I need more details whether you would support the statements or not. Does this summary need any further clarification?

09:30 PARTICIPANT 31:

It's okay.

09:33 RESEARCHER:

Okay, I'll start with Participant 31. Participant 31, would you support or disagree with the statement, and why?

09:56 PARTICIPANT 31:

Yes, I would agree, but let's keep in mind that it's not working perfectly like in theory. Well, yeah, Scrum can be a tool to help us to engage team collaboration for example. But in the meantime, all the team members must commit to what we want to deliver, even if it's a team or large scale or small scale. And in theory, it's like people over processes and documentation. I can see it as a tool, and it can be useful.

10:51 RESEARCHER:

Thanks, Participant 31. I'll move to Participant 38.

11:05 PARTICIPANT 38:

Yes, absolutely agree with this notion that Scrum does bring some of the social qualities as you call them to the team like collaboration. The only perspective that I would want to add here is that you mentioned that to achieve quality, the process would be to maintain software engineering practices. I would add better practices and sustain commitment from the team. What's important to distinguish quality into two segments. Quality of the deliverable by the outcome. The other is the quality of the backbone of the source code. And these are two different elements. Your product can be free of bugs and working absolutely as per the stakeholder requirements. While the code behind could be something that's not really scalable, or your code can be following the best practices in the world while the functionality features exhibit a lot of bugs. Although good software engineering practices and commitment from team members cover both of these elements. But I think it is important, to me, it's important to segregate these two dimensions of software quality. Because QA and engineering practices may improve internal quality but not always the fit for purpose. Fit for purpose need strong collaboration with the business and continuous feedback which is fundamental and more effective than software engineering practices.

13:51 RESEARCHER:

Thanks, Participant 38. I agree and we will discuss those comments in the upcoming slides. I'll move to Participant 36.

14:18 PARTICIPANT 36:

Hello. Yes, definitely, I agree. I think Scrum enhances the social aspects of the team including collaboration, but for good quality, we need to apply good software practices and have team members who are committed to the work they do. That's the key.

14:38 RESEARCHER:

Thank you, Participant 36. I'll move to Participant 30.

14:52 PARTICIPANT 30:

Yes, I think that every time you have so I support this statement, any time you have the commitment from team members to achieve something especially, well, good quality in a product, then it's something that I think achieves good results and good collaboration is a must. And as this statement says, Scrum helps in because it makes collaboration better, people are more accountable and everything is transparent. So, I'm on board with this statement.

15:17 RESEARCHER:

Ok, great. Participant 37?

15:20 PARTICIPANT 37:

Yes, I also agree to the statement stating that Scrum enhances collaboration, accountability, etc. But definitely we need to respect this is on top of everything from working to executing. And on top of that, sometimes I think we have to kind of track the resources, whether they had followed the processes or not. And that is covered under the commitment to both performance and excellence. I think so. I would support this statement.

16:01 RESEARCHER:

Thanks to you all. So, this is the first statement:

Scrum enhances team collaboration. It also improves the participation, access and collaboration with the business users or sometimes the Product Owner.

How does collaboration in Scrum help quality?

Collaboration in Scrum helps quality in 3 ways:

Collaboration helps better knowledge sharing within the team and the business users or the Product Owner.

Collaboration with the business users allows receiving feedback which helps the team to adjust misalignments to business needs.

It also helps to understand the requirements better because of the interaction with end users or the Product Owner. This reduces assumptions about the requirements and subsequently less defects

Collaboration within the team helps peer to peer learning and support which contribute to better decisions, for example, on coding and design

18:17 RESEARCHER:

So, before I start gathering feedback, support or rejection of this finding, I'd like to ask whether this statement is clear or I need to clarify?

18:40 PARTICIPANT 30:

Sound good.

18:45 PARTICIPANT 31:

Reads fine to me.

18:49 RESEARCHER:

Alright. Thank you. Participant 31, would you like to comment?

19:02 PARTICIPANT 31:

Happy to comment. The statement is fine and I support it, at least it reflects my experience. Collaboration in Scrum happens when the team works together and support each other's inputs to produce better quality. I agree with the point that Scrum makes collaboration better, but the commitment of the individuals to it is also a must. You talked about peer to peer support, that's very true. But it required the person on the team to commit and be open to share, help and receive feedback. The person has to invest in this collaboration so this cohesive team that Scrum calls for can happen. You also talked about business collaboration and I think that is the best advantages of Scrum. The requirements are never complete or details to understand the immediate and future business needs. Those conversations we can have with the PO or the business in daily bases make huge difference. And I agree they help reduce bugs because we know what they want and they can check our understanding all the time. And like you put it they can validate and check our assumptions. Unfortunately, the business collaboration is not always ideal, not every product owner is an expert of the business. That's not true. So, I like the statement very much, a good summary of my experience.

22:07 RESEARCHER:

Thanks PARTICIPANT 31. PARTICIPANT 37, would you like to go next?

22:18 PARTICIPANT 37:

I like the statement. We talked a lot about this in the interview. Scrum boost the team mentality and working together mindset. Developers come out of their silos and help each other's, talk to the QAs and avoid silly bugs. I agree with PARTICIPANT 31 especially when he said business collaboration is not ideal! Very true. You don't always get the right expertise. But that is the management and the business commitment to agile. Obviously, if they do not make effort to support the agile implementation then it won't give the results you mentioned in the statement. Developers will collaborate in Scrum but without business expertise to support them then they will make assumptions and bugs will pop up in UAT.

24:07 RESEARCHER:

Thanks PARTICIPANT 37. Great input. PARTICIPANT 30, would you like to go next?

24:22 PARTICIPANT 30:

Sure. I agree 100 percent with PARTICIPANT 31 and PARTICIPANT 37. For quality, collaboration helps in the way you describe it in your statement and I'll add more. When developers work together the quality assurance processes become more efficient. I'll explain. In my team, we do more code review because we talk to each other's all the time. My mate can just walk up to me and ask me to review his code, we not have no formalities for that. Junior developers feel comfortable to ask for advice and so on. When it come to collaboration with the business, it can be super efficient or it can go bad and create more problems. I had good and competent product owners. They know their stuff and invested on the product. They hand on, test, check all the time and give you feedback and I had crappy ones, they have little expertise. Each time we ask questions we have to wait, he doesn't know. We developers get frustrated when a product owner is just a manager assigned to the team to tell us what to do and no business expertise. Yes, I agree when we have business expertise on the team, there are less and less bugs.

26:17 RESEARCHER:

Thanks PARTICIPANT 30. I like these passionate inputs. PARTICIPANT 38?

26:29 PARTICIPANT 38:

Yeah, I agree. I think the guys said it better, no much to add. Happy to see this in your analysis and the guys explained how it works better. With the right expertise and team mindset, it gives better results.

26:57 RESEARCHER:

Thanks PARTICIPANT 38. PARTICIPANT 36?

27:04 PARTICIPANT 36:

No much to add really. Good analysis! Perhaps, I can add to PARTICIPANT 31 comment on the personal commitment to collaborate and work on a team. I have been around long enough and I saw some cases where people can't work in an intensive collaborative process like software development. In my experience, I worked with people who wouldn't share knowledge. Because they don't know what they know, they not good at explaining what they know or they are afraid of being wrong. Some people don't see the benefit of collaborating, they try but if their opinions are ignored then they stop working with their teammates. Here, comes the importance of openness and to work together you need to treat each other's equally to start with.

29:15 RESEARCHER:

Interesting. Thanks for sharing your thoughts. What do you suggest when a team member doesn't want to share knowledge?

29:34 PARTICIPANT 36:

No magic solution really. I think universities are not good at teaching computer science students how to communicate in the first place. Like PARTICIPANT 31 said commitment to collaboration comes from the team and the individuals on the team, that's when things improve.

30:03 RESEARCHER:

Great discussion. Let's move to the next slide. Before I do that, I just like to mention that we already half time through and it is Sunday. I'd like to let you go on time, so will try to be brief and efficient and brief. It is getting late for PARTICIPANT 37 in Sydney!

30:24 PARTICIPANT 37:

I'm fine Researcher.

30:33 RESEARCHER:

Let's move to next slide. This is about psychological safety. We found that psychological safety in Scrum teams motivates people to care more about quality. Subsequently, they speak out to point out errors and defects. When, people feel safe, they also invest the required effort to achieve quality. I'll start gathering feedback, comments and your support of rejection of this finding, I'd like to ask whether this statement is clear?

31:32 RESEARCHER:

Ok. PARTICIPANT 37, would you like to start?

31:44 PARTICIPANT 37:

Of course! I agree. The safer for me to do my work, the better I become at doing it. Because I know how to do my work. All I need is the safety to empower me to do it better. I strongly agree.

32:38 RESEARCHER:

Straight to the point, great. PARTICIPANT 30?

32:44 PARTICIPANT 30:

I'll be brief and also to the point. I agree. Safety makes Scrum function better. The retrospectives are productive, the Standups become more transparent, people say what they think and report mistakes so they can get help. As a developer, just leave me alone to do my job and allow me to speak up when things are wrong then I'll write the best code to the best of my abilities.

34:18 RESEARCHER:

Straight to the point indeed, great. [laughter] PARTICIPANT 31?

35:04 PARTICIPANT 31:

I also agree. Agile is about continuous improvement. It doesn't happen without feeling that sanctions are an eventuality; people need that safety to talk about mistakes. As you put it in the statement, I won't invest effort on the quality of my work unless I feel it is safe to do it. I won't like a manager to tell me why it takes you so long to deliver. So, I won't care about quality anymore. That's sense of safety helps a lot and I'm happy to see it in your analysis.

37:08 RESEARCHER:

Great. PARTICIPANT 38?

37:14 PARTICIPANT 38:

I can't articulate it better that that. So, happy to simply say yes to this statement.

37:48 RESEARCHER:

PARTICIPANT 36?

38:54 PARTICIPANT 36:

Don't have much to add to what has been said. Over the years, I saw teams' performance and quality crumbles because managers sanctioned people who do a good job and takes them longer to do it and that was not within the manager plan and schedule. So, like PARTICIPANT 31 said, why would you care?

40:04 RESEARCHER:

I'll move to the next slide and a new finding. So, this is about accountability. We found hat Scrum promotes accountability in the individual and team levels. When team members feel accountable, they thrive to meet expectations for quality because they do not want to "let the team down" or substandard quality would reflect on them. Before, I start asking for comments, does this statement make sense or I need to clarify?

40:48 RESEARCHER:

We have 20 mins left. So, in 15 mins, I'll ask those who can stay a bit longer to continue. For now, let's see how much we get through? PARTICIPANT 36?

41:04 PARTICIPANT 36:

I understand that your finding suggest Scrum brings accountability to the team. This is partly true. I agree with the finding, but I'd like to add other factors contribute to this accountability. First, team unity. The more cohesive, the more supportive they become, then this accountability becomes more obvious from what I experienced. Then, your statement is true, they won't let each other's down. Developers show commitment to quality, they write better code because their mates may judge them. They care more not because they paid more, just because they are professionals, well-educated and know how to do their jobs.

43:08 RESEARCHER:

Nice input. PARTICIPANT 38?

41:04 PARTICIPANT 38:

PARTICIPANT 38 said it better. I can add my 2 cents. I work in an academic environment and we have so much freedom to decide how to do our work and the whole team has this sense of accountability. Everybody want to do a good job. I agree, in my previous job, I was less worried about the quality of my code because everybody was doing just the minimum. Honestly, if you treat us like factory workers, don't expect from us to be accountable.

43:08 RESEARCHER:

PARTICIPANT 30?

43:10 PARTICIPANT 30:

I agree. We are a small team and the transparency of Scrum made us more accountable. Because, we demand from each other's the same expectations we impose on ourself. If I write good code, then I expect my colleague to do the same.

43:58 RESEARCHER:

PARTICIPANT 31?

44:03 PARTICIPANT 31:

I think this is an interesting analysis and is my favorite so far. We talked about this in the interview. So you have my support for this.

44:48 RESEARCHER:

PARTICIPANT 37?

44:58 PARTICIPANT 37:

We also talked about this in the interview. I think accountability is not only confined to Scrum. We professionals, when there is less interference on our work and we can make our own decisions, then we are more accountable. And of course, this mean I want to do good job and my job is quality code and a product free of bugs for the business.

45:32 RESEARCHER:

That's great. I'll move to the next slide and statement number 4. This is about transparency in Scrum team. We found Scrum enhances the transparency of the process. When information, decisions and deliverables are transparent, then team members voluntarily inspect and review each other' work to provide feedback which helps quality. Any clarification? Any clarifications or questions?

46:08 PARTICIPANT 37:

Not from me.

46:12 RESEARCHER:

PARTICIPANT 37, would you like to comment?

46:18 PARTICIPANT 37:

Another interesting finding. It is accurate, but it does not work without accountability, good collaboration, especially a supporting team. So, I would say it support the previous statements and has its own impact as well, which this is statement is all about.

46:42 RESEARCHER:

PARTICIPANT 30?

46:48 PARTICIPANT 30:

I agree with PARTICIPANT 37. For example, I would not review somebody else code if we were not supportive of each other's and I'm afraid to do it. He may not like it. Scrum helps transparency in many ways and has this effect you describe here. For example, in the Standups, we just ask who needs help and people speak up and say I don't know how to do this and that and this is openness of the person and transparency of work progress. When people are not progressing, we become aware of it in the Standups and we jump to help, it can be review, comments on how to go about a problem, etc. This help quality in many ways, transparency brings too many eyes to inspect same work. I think you call it voluntary inspections. We are 7 developers in my team and we always checking each other's code sometimes because it is right there not hidden.

48:22 RESEARCHER:

Great example. PARTICIPANT 38?

48:38 PARTICIPANT 38:

Thanks Researcher. I agree with the guys and the statement.

48:52 RESEARCHER:

PARTICIPANT 31?

48:58 PARTICIPANT 38:

Same. I think we discussed this in the interview.

49:12 RESEARCHER:

Cool. PARTICIPANT 36?

49:18 PARTICIPANT 36:

I also agree. I maybe the older here [laughter]. In waterfall development, the business users had no visibility on the work, they had to wait months to see the first delivery. If you have good representation of competent end users, in Scrum, they can see first iteration much quicker and have access to the features as they become available in testing. So they start testing without us asking them. So, this is what the Researcher call voluntary inspections. Yes, transparency encourages people to test more compared to the old ways.

50:10 RESEARCHER:

Fantastic! I have 2 slides and 2 statement to discuss and we have only 10 minutes left. So, in 10 minutes, if you wish to leave then, I understand that.

50:18 PARTICIPANT 36:

I'm OK.

50:20 PARTICIPANT 30:

I'm fine.

50:22 PARTICIPANT 37:

I'm OK Researcher.

50:30 RESEARCHER:

Ok. So far, we have been discussing these social qualities of Scrum values. This part of the findings is more related to the process aspects. Scrum approach to software development. We found that the iterative approach of Scrum and the continuous inspection and adaptation contribute to quality. We will start with iterative development. The iterative development of Scrum enables better modularity of tasks. This allows developers to focus on the quality, because they develop small units of code. It also allows them better control over the code. Any clarification?

51:10 RESEARCHER:

PARTICIPANT 30, why don't you start?

51:20 PARTICIPANT 30:

I wouldn't disagree with this one. One of the best, the smaller the user story the better my code is, that's for sure.

51:50 RESEARCHER:

PARTICIPANT 31?

51:20 PARTICIPANT 31:

Same. This does not only help me as a developer but also allows me to fix future bugs and the scalability of the software once I learn more about the business. So, my understanding form the statement is that the long term effect may not be relevant but actually it is.

52:15 RESEARCHER:

PARTICIPANT 36?

51:20 PARTICIPANT 36:

I support your statement. I'd like to add that the team's expectations should be also talked about. I may write modular code, but other developers may not. To achieve better quality, as a team we need also to set these expectations and check every now and then.

53:01 RESEARCHER:

PARTICIPANT 37?

53:08 PARTICIPANT 37:

Very true. I'd like also to add to PARTICIPANT 36 comment, in my team we do it in the checks of the Definition of Done. But I agree, it is a behavior and a coding practice. The developer must adopt it and maintain it and as Researcher found Scrum helps to encourage it.

54:21 RESEARCHER:

PARTICIPANT 38?

54:28 PARTICIPANT 38:

It was explained well by the guys. I'd like to add, it starts by the user stories and the Sprint goal. If the user stories are messy and the Sprint is either too long or unrealistic than it is difficult to practice modularity.

55:06 RESEARCHER:

Great! Last statement. We suggest that Scrum's inspection and adaptation allows continuous feedback and frequent testing. This helps quality because adjustments are made to align with the business needs as results of the continuous feedback and adaptation. Let's start with PARTICIPANT 37.

56:02 PARTICIPANT 37:

Very accurate. But you need dedicated and competent product owner. We talked about that already.

56:21 RESEARCHER:

PARTICIPANT 38?

56:28 PARTICIPANT 38:

This is one of the best things about Scrum helping quality. I agree with PARTICIPANT 37.

56:41 RESEARCHER:

PARTICIPANT 30?

56:48 PARTICIPANT 30:

Good finding. I agree with PARTICIPANT 37. I would add he is willing to test and give us feedback not only to show up only when there is a Sprint planning so he can tell us what to do!

57:01 RESEARCHER:

PARTICIPANT 31?

57:03 PARTICIPANT 31:

I agree with the guys.

57:21 RESEARCHER:

PARTICIPANT 36?

57:23 PARTICIPANT 36:

Nothing to add. All the comments were good.

57:42 RESEARCHER:

Thank you very much. We come to the end. We do have few minutes, if you don’t mind few more minutes. I’d like to go around and if you have any feedback for me, would be great. So, feedback regarding these findings, the overall impressions and you have one minute each. Just take turn and give me an overall assessment of these findings. I'll start with PARTICIPANT 37.

58:23 PARTICIPANT 37:

If I understand your research, right. You looked for what Scrum does best for teams to achieve quality, right?

58:43 RESEARCHER:

Exactly.

58:45 PARTICIPANT 37:

Then what we discussed is what I experienced and validate my thoughts about agile. Agile didn’t change much the way we do quality assurance neither it did influence much modern software engineering concepts and tools. It changed the way we approach the development process and we work together to deliver the end product. These Scrum values you presented work as you describe them. I’d like to add that you need to continue in your Scrum implementation for these value to continue giving results. We also practice continuous learning. It didn’t come up in your findings. It is important because you learn about yourself as a team and adapt which is an agile core principle. Plus, you need to keep up with good modern software engineering, learn what is out there and experiment and improve. Thanks for sharing and please send me a copy of the paper, curious to read it.

1:01:03 RESEARCHER:

I will. Thanks PARTICIPANT 37. Have a good night. PARTICIPANT 38?

1:01:22 PARTICIPANT 38:

Good stuff, Researcher! Your analysis show that quality doesn’t happen without people. I strongly agree. Scrum mobilizes some of team and individuals’ capabilities and gives better results for productivity, performance and quality like you shared in the statements. I understand you want feedback out of this, so all the findings you shared reflect my experience. We all made comments but none of them dismiss your analysis. Actually, I’m surprised you focused on this area, because it is underestimated by managers. They usually think by dumping a tool in the process it will do its magic and the software free of bugs [laughter].

1:03:13 PARTICIPANTS:

[laughter].

1:03:23 PARTICIPANT 31:

I'll go next. Thanks for sharing. I also like a copy of the paper.

1:03:43 RESEARCHER:

Of course, I'll send it.

1:03:53 PARTICIPANT 31:

Briefly, I liked the analysis. Very thoughtful and has depth and focus on an area we don't hear much about. People when they think about quality, they think about tools, metrics. They ignore that people at the end of the day make the tools and concepts either work of fail. So, thanks again.

1:05:06 PARTICIPANT 30:

Thanks Researcher. I also would like to read the report. And yeah, I haven't worked in waterfall before. I won't be able to make comparison. But, I know when Scrum fails to produce quality not because they do not have tools and processes for quality. It is because some of the things you presented are not working for them. If you have unsatisfied and dysfunctional team, do you think UAT will clean up all the bugs? No, it won't.

1:07:33 PARTICIPANT 36:

Thanks Researcher. I think the guys gave you their validation, so am I. Thanks for this discussion.

1:08:10 RESEARCHER:

Thank you all. I'm grateful for your contributions and I'll get in touch soon. Sorry, we went 8 minutes over. I wish you good weekend.