

# Participant 8

## SUMMARY KEYWORDS

team, code, quality, application, hackathon, people, add, developing, manager, developer, happened, complete, team members, issue, automation, product, validate, learn, helped, create

## SPEAKERS

Researcher

Participant 8

**Researcher** 00:12

Hello. Oh hi Participant 8 How are you today? Hello. Can you hear me? Yeah, I'm doing well.

**Participant 8** 01:05

Yeah, I'm doing great. Thank you.

**Researcher** 01:06

Okay, fantastic. I'd like to start because I do have a lot of questions and things to discuss, and I'm looking forward to the discussion. Before I do that, I'd like to thank you for participating in the interview I really appreciate. Thank you. Yeah. Okay, let's start the if we start with an introduction, would you like to introduce yourself briefly? Mainly just your education and experience?

**Participant 8** 01:34

Yeah, so my name is Participant 8. And people to call me Participant 8. And I did my graduation, like in 2013. In electronics field, but later, I started my career as a software engineer in 2014. And from then, it has been over like, close to seven plus seven and a half years of the software engineering. So during my career, I worked as a developer, I worked as an automation engineer and DevOps, like it's, it's kind of its own rules, so have multiple rules on that. I work for various product based and service companies as well. And my primary tech skill is Java. And that's my primary skill. And the database that you that I use is Oracle and MySQL. Yeah, so that's pretty much and apart from my personal experience, I do have a freelance experience like developing mobile games, preferably Android. Like, that's my hobby, and I do as a freelance on that.

**Researcher** 02:54

Okay, fantastic. So now you're working on an Agile team, right? It's yeah, what do you use the scrum? Yes. Okay. So how long have you been working together in the same team?

**Participant 8 03:13**

Yes, Scrum. Like it's been any year, like, since I joined this company? And yeah, I won't say that that team is so big. It's like, only a 10 people on my team, because since right now, I'm working in a bird based company. So we are working on like, single module, I can say that it's related to some marketing divine. Yeah, so the team is having only 10 people.

**Researcher 03:45**

Okay. Are they cross functional? So or only developers?

**Participant 8 03:54**

Like the kind of mixer like, here, we don't have any, like, dedicated team for QA. But we are the developers, and we are the testers like my role, like it's mixer. But since I'm good at automation, I also get more involved in QA, QA team, as well.

**Researcher 04:20**

What type of software do you develop? You said it's a model for marketing. Is it a new development new product, or?

**Participant 8 04:31**

What I can say is it's not a new development, it's an enhancement to the existing product. So what these people do is I mean, what we do is, so basically, we are a CRM application, our complete application is a CRM one. So we have an additional product called Marketing Management System, where this will be integrated to the actual application where let's say, like, you, you came to my company's website and use and you signed up for a newsletter or for any for any event. So that's it, like, as soon as you sign in, until unless you answered for the emails, so you keep on getting emails, like, based on your user interface, like, let's say, we send you an email related to some product, let's say you are, you are in an organisational level, like some little kind of thing. So obviously, you might, you might be interested in our product. So we will be sending me an email stating that we have these products. And this might help you. So let's say if we don't get any response from you like any email, click, or any email read or any of the particular responses, we use a to capture all these things. So they model what it does is. So if you don't do any response, like you didn't read, or you didn't click them, any of our email links, we will send you the next email, like maybe after three months. And even then if you are not interested in that, then we might remove your contact from our system. So that's how it works. Like if you're interested, you will be getting some emails, like on a weekly basis or on a monthly basis. But if you're not trusted, we won't bother you. [deleted to preserve the researcher/participant anonymity] will take care of that.

**Researcher 06:42**

Okay, great, thank you. We'll be talking about quality, most of the time. In this interview, I'd like to make the definition clear. So I propose a definition we wish we use, and we can discuss it. And we'll start with the rest of the question. So we use a definition proposed by ISO standards. And the definition says quality or software quality is the degree to which the system satisfy the stated and implied needs of its various stakeholder and thus provide values. It also proposes a set of non functionals or qualities that the system should adhere to wishes mainly performance, compatibility, usability, reliability, security,

maintainability, and portability. So first, do you agree with this statement? Or definition? Or would you like to add or comment on it?

**Participant 8 07:49**

I can, I can say like, I can agree with you, because that's what we actually see in our product. Like when we deliver the product, the final approach, the final. The final things that we do see is all of this during the performance.

**Researcher 08:10**

Okay, fantastic. So what do you do in the team to assure quality? So what type of processes you use? You mentioned automation. And you mentioned you test yourself as a test as a developers. So what type of processes do you use to assure a quality?

**Participant 8 08:30**

Okay, so basically, we divide this approach, like, let's say, we are developing a product. And there is, let's say, like, the 10 people in our team, and since everyone has their own stories, so once a person like let's say, I have a story I send to my to me, and I'm working on that. So the first approach we do follow is that as soon as you develop the code, you will be reviewing it with the piece. So it's called a peer reviewer, where the other developer will just review my code, whether I followed my standards, the particular standards, coding standards and like method naming conventions and some major and minor both of the things we do see and once it was done. So, since I said like I was involved, even in the automation process. So my job is to make sure that if that particular that particular story can be automated, then yes, please add to the automation tests you like, we will be giving a deadline. We'll be taking a deadline of a week, just to make sure that don't cost much. We don't take much time on that. And we read this this automation scripts. And then we deliver, we push our code to the Git repository where we have our Jenkins build system, which will take care of like planning it daily. So once we develop our application, and once the automation code also is ready. So the Jenkins job since here, we trigger it using shedule to pay it will run daily, and where we will monitor it like for one or two days, just to make sure that all of our existing scripts were working fine, just to make sure that the new piece of code that we implemented is not breaking any existing system. So if it is working fine, and we don't have much failure, so we don't have any failures related to the new application. And yeah, relative to the new code that we build. So we will be moving it to like, we have this environment staging QA, and after that, Dev and production, so what we do is we will be first moving our code to staging, and then we'll be moving it to QA, and then finally go on to production. So before moving into production, like we need to take the sign up from the release manager, where we have to show these metrics like okay, the manual testing for this one was done, and the automation code was written. And we have seen, we have shown 90% of the pass, pass percentage, and then if at all it needs any, like, it has any recommend of performance thing, like let's say like, it also needs some performance testing, we will be creating it. Like we'll be figuring it out, like based on the stories that we work on. Like if it doesn't need any performance or performance testing, we don't do that. But if it needs we will be adding even that report as well. So that the release manager can sign off and then it moves to production. So this is the process that we follow.

**Researcher 12:26**

Okay, great. Thank you. That's very thorough, quality assurance environment. That's impressive. Thank you. We will get to the we will start with the question of sending in my email. Thanks for answering those questions. When I look at the answers, it seems that the team where you're working now, it seems to be relatively a safe working environment. I will explain what do we mean by a safe working environment? What do we mean is the work environment provide a sense of security from repercussions? So everybody in the team can admit mistake they feel it's okay to admit mistake, they feel it's okay to propose initiative to discuss problem there is a sense of confidence amongst the team that coming forward speaking up they will not be embarrassed or reject or punished for speaking up. So this confidence comes from a mutual respect and trust of each other. So do you agree with my assessment that it is relatively safe work environment as I describe it?

**Participant 8 13:47**

Yes, it is. Yes. But just to add up to your question, like if something went wrong, and since we work with like, like most of the team work in our Indian times, time zone and two other people work in UK time zone. So if let's say like, they were added they identified this kind of like there is an issue with product that code that was delivered and then at least one or two of our people have to look at that even at the midnight so that's the only thing that I think it's not that great, but at least like one or two people from our time zone should work at that midnight just to make sure that okay, we are still working on that and it just take time. So we just need to mention that so that the other li later people just to not think that okay, nobody's looking at this kind of issue. We need to we need to take a call. And so just to make sure that okay, someone is looking at their face, don't panic. So that's the message we have to give that.

**Researcher 15:08**

Okay, great. Thank you. So in a scale of strongly agree strongly, strongly agree, agree neutral disagree, strongly disagree. So how do you disagree that the team is a safe work environment?

**Participant 8 15:27**

I would say strongly agree, because during this whole journey I never seen like, okay, an hour felt like someone did a hash ray or anything like that. Like, is it just like, yeah, things might go wrong sometimes. But it's okay. It's it all there in our hands. So we don't worry.

**Researcher 15:54**

Okay, fantastic. So what made this team or this work environment safe as the way you describe it? What make it how this this safety and confidence in the work environment came about?

**Participant 8 16:12**

Yeah, you see this team. I won't say that. It's like an old team. It's relatively new. I can say that it's hardly one and a year, this team was great. So there is no one who is a complete expert in this, like, everyone is learning and implementing it. So we won't say that, okay, this guy is an SME. So we should not treat him like, we can't ask him directly everything. So we need to keep in mind, like, whenever we talk, so there's nothing like that. Everyone like, everyone like, yeah, he's as the same experience like me on this new product. So it's okay. He's just a friend. So I can straight away ask him like, if I start

somewhere. So that is, that is the kind of leverage that they gave us. Like, during, initially, when we did when we were joined, even my product manager. That's what he said, like, everyone is new except me. So just don't think that you are you joined late. So when you start somewhere, you're, you're welcome to ask anyone. So everyone will help. So that's the leverage that they gave, and it really helped us.

**Researcher 17:40**

Okay, great. We will start with the question of sent in the email, and we will discuss some examples. The first one says, if you make mistakes on your team, it is often held against you. So you answered no, but I'm responsible to resolve the issues. But my leader manager will be the one who would be questioning and backing me. That's great. That's fantastic. So can you explain to me how it happens?

**Participant 8 18:13**

Yeah. So it actually didn't happen with me, but it happened with my one of my colleagues. So what happened was, like, it's actually not a mistake, but it's just misunderstood. Like, he merged the code, where the other branch was changed states where, unfortunately, got missed. So what happened, like his changes were there, but the other changes were not there in the light, I mean, in the, in, in, in that application, so it didn't went to production. So it's still in staging phase. So what happened was like, because of that, during knew, like, where it is failing and why it is failing. So that particular piece of code, so later once the debug the that particular code, like where it is failing, so they identified that, okay, this is the commit that was made. And these are the rule set that were re written, like, overwritten. So then, yeah, that manager the lead, I would say, like, he said, like, Okay, we need to be careful on this going forward. Whenever you merge the code, make sure that you're not overwriting this, so it's just a backup like since it's the first time so don't make it repeat because not, we but other people might ask so that's the thing that It gave us so for that particular developer.

**Researcher 20:04**

So he wasn't blame. And it was talked about and resolved in a very constructive manner. Right? Yeah. So do you have an example where this there was a quality issue? And somebody admitted you or a team member, you admitted? And I'd like to understand what happened.

**Participant 8 20:26**

I mean, for this particular scenario.

**Researcher 20:28**

You could you could use another example, if you wish.

**Participant 8 20:33**

Okay, so can you repeat the question?

**Researcher 20:37**

So, is there any example related to quality, and it's caused the problem and mistake done by another developer or yourself? And it's related to it caused a quality issue? It's either came from the code or from the design or? Yes, yeah. Yeah. And it caused an issue and somebody, whoever caused it came forward and admitted the mistake.

**Participant 8 21:08**

Yes. So I won't say that it happened recently. But it happened in during in the past, like, I think it happened even with me. So maybe I can give you my own example. So when I joined the team, and it has been like, less than a year, we have to whatever the code that we write, we have to submit it to the peer review. So at that time, I didn't followed much standards. So what happened was the functions that I have written, it's in capital letters with the starting letter, and even my complete class is like, more than I would say, around 500 lines of code, which is not that which is not, which should not be there. And even my class name, it's like, I can say more than more than 100 characters. That's really not acceptable.

**Researcher 22:16**

Really, I haven't been developing code for a year for 10 years. I wouldn't do that. It's okay. Sorry. Sorry. Good. Continue.

**Participant 8 22:28**

Yeah. So when the reviewer read my class name, he asked us call the horizontal bar, like, completely to the end. So it's that big class name. So obviously, I didn't compile it. I didn't compile it in. Like, we have our inbuilt compiler, where it will check based on our framework, like whether we are missing any standards or not. So I didn't do that, since I am new to that particular team. So I just since my lead said that, okay, you have to submit the critical review, to get approved, and then only you can push your code to the branch. So that's what he said. And I thought, like, okay, it's simple. And I submitted my code as a review. So then I got, like, nearly 20 review comments for just a single class. So, obviously, you know, right, like, since in the reviewers, we also have watchers, where it's one of my leads, who is also a watcher. So he said, like, what, why did that review came, and you didn't follow the coding standards? Like, you need to ask me for you doing that. Right. So then he just, he just gave me like, Okay, this is what our plugin is there, where it can check for any coding standard issues. So I know it's pretty basic one, but for code quality we have to follow our standards. Yeah. When I did, yeah. So when I did my, when I use that plugin, like, the plugin name is something like it starts something with the application name and code, check something it is there. So when I just do that, I've seen some red lines throughout my entire code stating that okay. The class name should not be more than, like 20 characters like that. They're like, some bunch of comments that were appended to that. So then I like okay.

**Researcher 24:50**

So that's a good example. That's a good example was me thanks a lot for sharing the, the example so So the your, your team leader dealt with it in a constructive way talk to you. And there were no repercussions as far as I understand. So how did you feel about that healthy approach to, to the quality issue of your codes? The way you approach it was very healthy with very constructive. How did you feel about that?

**Participant 8 25:26**

Yeah, I felt great. But I also felt that why didn't I ask this before? Like, I know, right. I mean, what I observed was like, until unless you do any mistake, no one will teach you. But that's what I first learned

it, because everyone will think that okay, this is easy. And so he can do that. So but sometimes the easiest task is the one where we make mistakes.

**Researcher 25:56**

Yes. So do you think you learned from this experience, you become better at following standards?

**Participant 8 26:06**

Not only I learned, I became, I can say that, maybe obsessed, I can say because if my peer who writes a method which starts with capital letter, I can I would simply say, no, you should not do that. I will show him and explain to him that the method is failing our coding standards. So since I joined this company and the new team, like I learned many things fast from my mistakes.

**Researcher 26:34**

Okay, fantastic. Thank you. We will move to the next items, which was in the email, it says member of your team can bring up problems and tough issues. You said yes, sometimes, and it will make us to do a rework if we don't see beforehand. So that's really good. Do you have a good example, which related to quality? Somebody brought up a series or an issue or a problem that creates? Yeah, yeah. Go ahead, please.

**Participant 8 27:08**

Yes, as I said, right, like, in my first example, there was this guy who merged his code by overwriting existing code. So what happened was, he merged it. And after that, there are so many comments, merge commits that happen. So obviously, we can't revert the merge commit, because it has been like, if we revert the merge commit, that he does, we also need to ask the other developers who did the merge commits. So it's like, pretty messy happen. And then what happened was, like, we have seen, okay, since my code that I have, I mean, that the developer has over it's not that that much. Like, since I already have my code in my local base, so I can maybe fix that myself, instead of asking him to revert as much commit. So what happened was like, my code has only like, two or three pages where I can pull his changes, pull all the changes, and then submit that, submit it as a new commit. So that nobody, what I can say is, so that it won't be a river. But it's a revert for me. So that happened, like I spent nearly like, an hour to fix that issue. So that's what happened. So it's the it's just a way but that that was done.

**Researcher 28:48**

So what did you learn from this experience? It created the rework for you.

**Participant 8 29:00**

It was my call that have taken. It was his job to fix that issue. But since it's easy for me to like, recommit the code, instruct him to revert these changes and do that. So I said, like, it was not set by me, but it was said by my team leader, were going forward. Please be careful. So that's what the said. And he obviously learned. It's in the previous company. The team lead spent some time coaching him on the proper process we use to merge. It was a good experience, because since I committed my code and I didn't pull the latest changes I have my local code bits available. So it's easy for me so that although he made a mistake, I can fix it quickly. If I don't have my changes in my local system, it definitely a hard



way for him. It's always better to take a backup. So that's what I can say. So we discussed and decided to make a check list to follow before we merge our code and everybody agree and we started using it.

**Researcher 30:21**

Okay, great. Thank you. So the next example, the next item on the list is people on your team sometimes reject other for being different. You said yes, but they don't show it in your face, they will be taken on your back, talking your back. Sorry. What did you What did you understand from reject here? What was your interpretation from reject?

**Participant 8 30:52**

Yeah. So what I mean is, there's one certain example where it's not related to product that we are developing, like, product that the whole team is developing. Like, we have hackathons where we can present our own ideas, like any new in innovations. So decide to use the hackathons to experiment and propose innovative ideas. It is up to us how to decide, it can be new features, trying new technology, new tools or improve our coding quality. We show each other's code and we learn from each other's coding.

**Researcher 32:55**

So this is in your current team?

**Participant 8 32:58**

Yes my current team. My team is like, highly appreciate that take initiatives like this. Even my, because they just want to learn like how each one of us does stuff. So like, even in my current team, I just presented them like, I'm developing an MMO game. Multiplayer Game. So where once I showed them the sample demo video. So everyone in my team approached me, like texting me, like, how did you do that? And what is the technology that you use? And I learn like that from multiple questions. It's like, it's when I see that I felt really good. Because they have some interest in that, like, yeah, that's what I thought.

**Researcher 34:00**

Oh, that's great. So in this team, you feel good that people appreciate your ideas and taking initiative. Do you have an example in this team where you took an initiative? And yeah, yeah, yeah, please go ahead.

**Participant 8 34:16**

Again, it's with the hackathon, where even we got an email like, the whole UK team will be working on a hackathon project like for two days. So they had given they didn't give any problem statement. So they said like, the problem statement will be given on the start date of the hackathon. So we had to form a team. So what happened was like, I approached to my team leader, and one and immediately said, like, yes, actually, I am planning to invite you on that. So he said like, we need one more person. So we added one more team member to our team. And we just thought like, Okay, let them give the problem statement. And then we can work on like splitting the task and to that. So in my current team, my team leader is also a manager for me. So he's the one who actually took, like, initiative and assign like, Okay, you will, he asked us, like, who are good at who are good at what. So there's one guy who is good at



developing UI. So he said, like, Okay, I will help you I, and I'm good at developing, I mean, good at configuring the infrastructure. So we use AWS. So I created some dummy instances, just to make sure that the application is running. And I can host my Spring application into that AWS mentions so that that part I worked on, and my league worked on like integrating both the front end and the back end. So a whole of team, we developed the product, I won't say that the product was 100%, complete the hackathon product. But what we felt like it's a new experience for us, we learn something new, we use some new frameworks, new, new tools, so data doesn't know something and it doesn't know something. So we'll learn it over. So yeah. So this helped us.

**Researcher 36:40**

Yes. So how it helped you? That's great. I mean, collaboratively, you came together, you brought initiatives, and you developed a product, even though as you said, was not complete. So. So it's the experiment, it's just an experiment to push your technology or technical knowledge and try new things. So do you think this experimentation and taking initiative in this safe environment? Does it help also quality?

**Participant 8 37:13**

In this case no. It was just a prototype. Because since we have less time within spend time on quality. Since this is the example for action not something for production.

**Researcher 37:27**

Yeah, I didn't mean that example, I mean, in general, so for example, if you use coding standard, like you said in the beginning, yeah.

**Participant 8 37:37**

Yes. We retooled quality, I can say that, we hardly seen any, any quality problems, because since I said, right, like I have been almost a year. So we have never seen any issue with that. Although, like, I'm the one who will be even looking at the quality thing. So I will like, as soon as I think like, okay, something is something is having issue, I will be just bringing it up to the team in my daily standup, so that I will give them like a brief summary of the issue. Like, we need to fix this, like, in a couple of days, so that failure won't come again. So I'll be like, involving in that to the team just telling that. Okay, this is the deadlines that be passed out of focus on that. So yes, we are we are good with quality, it has become our priority.

**Researcher 38:44**

Oh, so you mentioned a very good example. So for example, if you see an issue, you bring it in the stand up and you talk about it. So do you think this behaviour, this habit of bringing issue in the stand that help reducing defects, for example, less bugs and the cause?

**Participant 8 39:05**

Yeah, yes of course. So what happens is we have to stand up, maybe I can say every morning, which will be just 15 minutes of time, where we'll be talking about the issues that we have currently and if anyone can help like how we can do that, so that okay, we can mark it as like priority tasks so that after our morning stand up, that person can approach someone else in the team just to resolve that. And we

also have evening standup where we'll be having our call with the product manager since he's in US time zone, so we have that particular standard, it lasts around four minutes to one hour. So where we will be discussing, like, what we have done, like during the whole day, and whether we start anywhere so that he can resolve our issues. So these are the things that we do in the evening stand up. So it's basically, like these two candidates really helped us.

**Researcher 40:23**

Okay. We talked about initiative and how you use the hackathon to experiment. Do you have any example for example, where a team member proposed something new that's helped quality, for example, a new tool, a new method, which has helped you Yeah, please go ahead.

**Participant 8 40:43**

Yes. Yeah. I can give you my own example. I'm the one who I can give you an example. So the application has a feature not that great. I would say that it's not that way. It's it runs on a Silverlight application. So. So what it does is, as soon as the application get completed, it will generate a PDF report, where we have some calculations and some texts that we have to validate it based on the database values or Excel values. So what happened was just to validate one more, because application, it takes around 15 to 30 minutes of time. So what happened, it's taking, like, let's say, the complete application, the complete Mater is testing the one single loan application takes around for the past 45 minutes, including the PDF report validation. So my, one of my like, my team manager, he asked me whether is there any way to reduce this effort? Like, can you build any tool? Yes, to validate a PDF. PDF document. So I asked him, like, are you able to hear me because I'm hearing some noise?

**Researcher 42:39**

No, I'm hearing you very well. Sorry.

**Participant 8 42:42**

Okay. Yeah. So my team lead just asked me like whether we can develop a tool where we can validate that PDF. And I asked him like, okay, I can validate the PDF, but do we have anything? Like, the expected values, the expected things that you have? So is that like, yes, we have some Excel where we have all the things that we need to validate. So that's what we are doing currently manually. So I took a week of time, just to create a tool. So it's, I used using Java strings, the just to create a UI tool, where we just need to upload the PDF, we just need to give the PDF location that the report was generated. And just we have to click validate button. So that's it within like, some hardly 10 seconds, it will validate everything, and it will show the result in the UI, in the Java application UI, like whether everything was met or not. So it nearly saved us, like 20 minutes per one loan application. So I can say that. If you do that manually, it will take around like in a week, let's say it will take I mean, I can say that it varies a lot of time, just to in one hour, you can validate like, nearly 40 to 50 applications. So that's what it achieved. It even we have even published it as a value add for our customer. So I don't know the exact amount that they have awarded for that. But it's a really good amount. Although it's not for me, but it's for the team that they have awarded.

**Researcher 44:43**

That's a great example. That's a great achievement. Thank you. I would move to the next one. Yeah, we talked about taking initiative and we had an examples. We will move to the next one which is about helping each other right? It says it is difficult to ask other member of the team to help, right? So you said no, there are people out there who would help, but it will take more time as they are busy with their work. It's normal. I mean, you ask somebody to help if he or she is not available. You can back and but people overall are willing to help. Right? So my first question, it's a two part question. Do you think people help each other in this team in your current team? Because they feel it's safe to do so. So it is safe to go and to your teammates? And say, Can you please help me? Do you feel comfort? Confident dude doing that?

**Participant 8 45:48**

Yes. So it happened to me. Like, as I said, there's a new team, like when I joined this team, like last year, I was completely new, like, although they have some knowledge transfer videos, and documents, that we have seen and understood the things about this application, but the overall application is completely new for me. So what happened was, I approached my lead. Just if I have any questions, so I asked him and since he's busy, he directed me to another team member, where he spends his time with me, like, stating like, explaining me things like Okay, so you have to create a step is to create complete flow. So, he spent, he took like, like, most of the time, just to explain me like the so it works. So yes, in my team people help whenever we start somewhere, and even me, like, recently, some new people joined to my team. So even myself spend time on, like, navigating them to the complete application. And this is what we developed this. So we saw even I have them. So yes, it's complete, peer to peer helping it is there. We do it without being asked because we do it for each other's.

**Researcher 47:30**

Yes. This is a good word, a word you mentioned peer to peer helping do this does this peer to peer helping also helps quality? Do you think that when you help each other? Yes, yeah. Do you have an example?

**Participant 8 47:46**

Yes. So I think this was started recently, just to improve the quality by my manager. So what happened was like, he said, like, once you write your code to add, you review each other code. Like, let's say, if I am developing, developing my application, and I'm, instead of me pushing code directly so I just add my own team members, like anyone, anyone of my choice, not the manager or anyone whom you think that okay, they are good at reviewing your code. So just add a couple of reviewers, just to just to make sure that you're not missing anything and our code meets our own expectations. So that's what we are doing right now. So we are adding two reviewers. And we are just checking like, he will be just giving comments, like if we missed anything, something here. So once we think that, okay, this comments can be incorporated, and everything goes well, then we do a pull request. So that's what's happening right now.

**Researcher 49:05**

So that's a good example, but I will challenge you a little bit. Code Review is used in many organisations in many teams, right? I'm sure you have you had it before in other teams. So what do you think it's better in this team? Do you think the safety make it better? How does it make it better?

**Participant 8 49:29**

Yes, I will tell you. So in my previous companies, the code review will be done by other teams, not my were not my own team members. Because like when I say other team members, they will be like, even subject matter experts. So obviously, if they're from other team, they, they just look at the coding standards, not the complete functionality. But in my current team, the reviews that we're adding is our own team members where they know, like, they have the product knowledge, like, since everyone in the same team, we know like what stories we are working on. And we know like, Okay, this guy is working on this module. So we know, we know some things. So let's say if I, if I write a new function, just to get my yes to get my model working, so he can simply say like, Okay, if this module, I mean, this function is already existing in a different tool set, why can't you use the same one, instead of creating a new one? So that way, these people can help me writing not writing a duplicate code. So I can say that quality is also having mean quality means even not writing a duplicate code. Maybe I can say that, right? So, we learn about ourselves and each other's and the knowledge is invested within the team.

**Researcher 51:05**

Yes, of course. Yeah. Yes. Reuse is, is a concept in software engineering. Yes. Yes. Yeah. Reuse reduces. Create modular modularity. Scalability. Of course. Yeah. That's, that's quality. I think we talked about it. It's part of the non functional, I think it's, it fits within the maintainability. Quality or criteria? Yes, yes. Yeah, thanks. Yeah, that's a good example. Thanks, verse, me. So when you review each other code, do you learn from each other as well as the way you code? And do you think that learning makes you a better at producing good code?

**Participant 8 52:00**

I would say like, it's useful. Because whatever the function that I have done, maybe in might be in the future, it might help for him, like he can reuse the same. So yes, maybe it might help even for learning.

**Researcher 52:23**

So if you see your peers doing something in a better way, do you do your things do you adopt? Yeah, yeah, that's learning, isn't it? Yes. So you learn from your peer, he or she does things better. So you adopt her way of doing and that's makes you your work? It advances the way you code somehow, right?

**Participant 8 52:51**

Yes, yes. It happens me with me. Because, obviously, if I start somewhere, and if I, instead of me, dragging my brain, I can just simply go to my colleague, like, and share with him the exception that I'm getting. So obviously, he helped me a lot. In many times. So I just modified my code with his suggested code to get a few actionable wizards. Yes, I would say that. Yes, so learning from each other's helps our code quality.

**Researcher 53:32**

Fantastic, thank you. We will move to the next items will almost there. So no one on my team would deliberately act in a way that undermine my effort. You say yes, because I have already proved myself I'm capable of doing things to the team. So you are not undermined for your efforts? Obviously. Your efforts are now acknowledged, does this also applies to the quality of your work?

**Participant 8 54:14**

I don't know about that by because, as I said, we are the one who I mean, in my current company, we are the one who develop, and we are the one who tests the application. But since I have even the automation knowledge, so I'm, I'm like, one step ahead of them. I can say that, because obviously in the end, I'm the one who is going to look at the quality thing like it's whether it's whether the application is working without any issues or not. So, so I would say that yes, since I have little burden on me, because the other people who are less experienced in quality side compared to developing side. So I would say that I definitely have to keep in mind whenever because I am the one who needs to sign up quality in the end. After that, I have to move it to the team lead. So that's what I have to have to keep in mind before I'm working it. So obviously it will improve my quality. I'm sure.

**Researcher 55:36**

Okay, thank you. So the last one, it says working with members of my team, my unique skills and talents are valued and utilise. So you have answered said yes. Not all the members in my team are within the same skill set of skill. So my skill are outmost valued. Yeah. So do you think that's in this team? Because you feel safe? And you call up? All right, you help each other? Do you think that you pass your skills on other team members who are less skilled? Do you share your skills and knowledge with other team members? less knowledgeable? less skilled?

**Participant 8 56:30**

Yes. Like every week, yeah, weekly, once a day, we have a session called Lunch and Learn where during our lunchtime, like, it's not mandatory one, but it's just I can say that. Yeah, it's just like, if you are willing to you can join that session. So where everyone in the team, they will be, it's not only my team, it's like the company, the organisation in the UK in my current location that they do, like, we have to register ourselves before we just to tell that okay, I am knowledgeable in this particular area. So I will be willing to present it, like I will be willing to teach them. So if anyone interested? Yes, they're good to join. So we have that kind of session where I can teach them like whatever things I know. So we are doing that already. It can be someone's code was exceptional so we can learn from it or someone experimented with something with better results and so on.

**Researcher 57:47**

So do you think that people's quality improves when they do that? Is quality part of that learning as well? For example code quality. Does it influence people's code quality?

**Participant 8 57:57**

Of course. If someone show me his code that might be better than mine, then obviously I learn and I try to match his quality. The learning can be also tool or technology.

**Researcher 58:35**

For example, automation, isn't it? It uses tools? Don't you tell people this is how the way to use it. This is how it works.

**Participant 8 58:44**

Yes, I teach them the automation, and obviously they can't learn it in a single day. I know that it might take time. So maybe like once they're interested in that they will come to me with questions. We have no problem asking questions and learning from each other's.

**Researcher 59:16**

Of course. Thanks for those good example, I really appreciate. Do you have any things you'd like to add, which we didn't discuss? I mean, you work in a safe environment, the team help each other. Do you have any examples that we didn't talk about? Or you'd like to share with me?

**Participant 8 59:39**

I think we covered most of the scenarios.

**Researcher 59:46**

Sorry! Last thing. I forgot to ask. How self-managed are you as a team? I mean how much freedom you have to make your own decisions?

**Participant 8 59:49**

Most technical decisions we make them as a team with the teal lead. Product decisions and Sprint goals with management and product owners, but we always involved in the discussions.

**Researcher 59:53**

Great! We talked a lot about a lot of things. So that's me. Thanks a lot. Thanks a lot for your time. I really appreciate. Yeah, thank you. Okay, bye. Have a good night. Bye

**Participant 8 1:00:20**

Bye