

Participant 20

SUMMARY KEYWORDS

qa engineer, team, project manager, developer, test, sprint, development, people, initiatives, processes, identified, team member, scenario, scrum, software, talking, facing, future, interview, career

SPEAKERS

Participant 20, Researcher

Researcher 01:33

Hi, Participant 20. How are you?

Participant 20 01:35

Hello, Researcher, how are you? I'm good. Thanks for asking.

Researcher 01:41

Okay, you don't it's up to you. If you want to put the camera on or not, it's up to you. You don't have to. Okay. It's up to you. It would be nice to see you. But it's entirely up to you.

Participant 20 02:04

Let me just turn some more lights on. Just give me a second, please.

Researcher 02:09

Take your time

Researcher 02:26

All right, Participant 20. Thanks for the opportunity to talk to you and when to the interview, I really appreciate. Thanks. Do you have any questions for me before we start?

Participant 20 02:41

Yes, I have, I got a chance to actually talk to you, I guess a month ago. So we talked about what you doing and your research. And I can know that you are researching on the resources, working as a QA engineer in agile working environment. So you asked some questions around seven, and I gave you some detailed, detailed answers on them. I was actually thinking, as you mentioned in your email that this is going to be a very casual interview, maybe about my experiences working in the team. Might be not very technical, or, you know, well, I'm not actually you can say technically prepared for this interview. But yes, if this interview is related to the actual experiences, which I have faced in the past, then yes, I'm pretty much ready for it.

Researcher 04:09

No, it's not technical. Nothing is technical about it. I'm not going to ask you technical questions. As you said, it's about your past experiences.

Participant 20 04:22

Okay, perfect.

Researcher 04:25

Perhaps we should start with a brief introduction, if you can introduce yourself, mainly your education and your professional experience.

Participant 20 04:34

Okay, okay. Okay, perfect. So, I graduated in back in 2014. I completed my bachelor's in computer sciences with main courses as C++, Java, and mobile development algorithms, object oriented programming, etc, etc. At the start of my career, most of my friends were starting their career as a developer, I was not very, you know, comfortable with the coding part. And I was I always hated the IDEs and the compilers. So I wanted to try something new. I thought of taking my career as a graphic designer, but it was in say, a totally different domain from for the CS graduates. So we had a chance to take a course which was called Software Quality Assurance, then I decided to pursue my career as a QA engineer. I started as, as an intern in a company then spent few months there, I always had that insect in my mind that everyone is opting to development career. So I always wanted to give it to a tribe, at least I cannot blame in the future. So what work as a QA engineer for like two years, then switched my field to development, I did iOS development for good five to six months, it was it was actually not pure development, but something like rescaling of the applications. Soon I realized that this field is actually not for me. So I always as you might serve as a critical thinker, a good problem solver. Always I always tried to think outside the box to you know, create complex scenarios to test application, I was always eager to improve the quality of the product, which we are delivering to the end customer. So after the realization, I moved my career back to the cure field, and now I am working as a QA engineer. Mostly, I have freelance projects, and my overall career is now almost eight years.

Researcher 07:24

Excellent. Thanks for the introduction. So I'd like you during this interview to talk about specific, agile team you worked for. And in your email, you told me you answer the questions based on a past experience of an Agile team. If we can continue on that if you could answer the question based on that particular team. Is that okay?

Participant 20 07:56

Yes, that's okay. Perfect.

Researcher 07:58

Okay, so that's particular teams was using Scrum or XP or.

Participant 20 08:06

Okay, so, when I entered the IT industry, I had seriously had no idea about the development models being used. Soon I a project manager introduced me to this scrum model. We were implementing Scrum, in which we used to conduct daily standup meetings sprint sprint sprint planning meeting in the the sprint, we had retrospective meetings by the end of the sprint, in which we used to take take notes about what but what went good through the spread, but went wrong through the sprint and few action items which we you know, need to work for the future. So that in that particular team, we implemented ages from models.

Researcher 09:03

Okay, so can you take me through how did you use Scrum in that team? What's what's what's how the team was using Scrum?

Participant 20 09:12

Okay. We were almost three QA engineers and 10 developers, three product managers and one project manager. So the project manager was the owner of the you can say the business and the whole scrum team. But how we kicked off the process, it was actually that we decided to conduct two weeks sprint. By the start off each sprint we had a grooming session or you can say a planning meeting in which the three product managers were demonstrating the tickets which we are going to actually the goals which we are going to accomplish for the upcoming sprint, then we have daily stand up meetings. The idea behind stand up meeting is that if we sit down in a meeting room or a conference room and talk about what we did yesterday and what we are going to do today, and if we are facing any blocker issues, if we are sitting down in a conference room and discussing these three items within the team of 10 to 15 people speak might take more than an hour. So, the idea behind the standup is that everyone is standing in the room and has almost 30 seconds to one minute to talk about his yesterday's achievements or you can say yesterday's task and today what he is going to accomplish and if he is facing any profit, so, to reduce this resources time, we always conducted the meeting by standing then we have a retrospective meeting by the end of the each spirit in which we discussed what the goals we actually what what wedding goals throughout the spread, what went back through the spread and if there are any action items, which we can note down and work on for the future scripts.

Researcher 11:33

Okay, fantastic. Thank you. So, well how did you like Scrum and how did the team like Scrum? Were they satisfied with it?

Participant 20 11:45

Actually, I was very much satisfied with it because I never faced or you can say I never experienced any other development model like we have if we are talking about a child we have waterfall in which the basic difference between Waterfall and Agile is that waterfall will not allow you to adapt to changes but while working in an HR and Scrum model and working in a product company. I felt that while if you're working in a product company, you are like forcing a lot of changes. Like if we have set a target of an items for a sprint, then by the end of the sprint we are delivering six of that planned items and like force extra items which are not planned. So I think this way ajl is much better than other development methods because we have a continuous we need to do. What is it those those changes?

Researcher 13:07

Okay, great. Thank you. I looked at your answer, you provided them the email. Thanks for sending me the email with the answer I appreciate. What I observed is the team that you're talking about was relatively a safe working environment. I'll explain what do I mean by a safe working environment and we will can discuss what we mean by a safe working environment is a work environment that provides a sense of security from repercussions. So people don't feel or are afraid of repercussions. So people feel that it's okay to admit mistakes. They feel that it's okay to propose initiatives and discuss problems. The team has this sense of confidence that they will not embarrass or reject or punish someone for speaking up. This confidence comes from the material respects and trust amongst the team members. So based on this definition of safety, do you agree with me that the team was highly safe work environment?

Participant 20 14:25

Sorry, Researcher, but I think this was not my answer for

Researcher 14:30

Nana. I was reading my definitions your Oh, sorry. Sorry. Sorry. Sorry. Yes, but your answer overall show that this team was safe. This is what we mean by safe.

Participant 20 14:42

Yes, because if I remember correctly working in a QA team. You know, we got a chance to implement the software processes which are being followed in the industry, which ultimately helped as improving the quality of the product as well as the processes, which are using within the team. So, we went off my colleagues has had to take an initiative, like, he told the project manager that we are going to do this XYZ activity and it might hurt our deadlines. But if, if, if you managed to do this one time activity, it is going to help us to improve the product or for example, if we are taking two hours extra in in sprint, and this will eventually help us to improve the quality and as well as the processes in the future. So my manager was totally okay with it to you know, take, take any any kind of risks, and to put delivery or stake and the resources where you don't have some kind of freehand to take the deliverables on risk and implement something that they feel like which which are, which are being followed in the industry.

Researcher 16:14

Okay, let's go through the questions you provide on the email. And the first question is, if you make mistakes on the team, it's often held against you. And the answer was, No, it may have happened once or twice in the past, If a mistake is made, the whole team takes responsibility for it. We are more focused to resolve that particular issue and make sure that it doesn't happen again in the future. That's Yeah, can you take me through what happens when people Yes, yeah.

Participant 20 16:54

Yes. So, what happened? Like we somehow released high priority, but on the production server, which was caught and identified by the end user. So, I was not pointed out by any of my colleagues or even my product project manager. But yes, whole team took responsibility starting from the developer, he admitted that this should have been should have shouldn't have been, you know, it should have been

fixed or it shouldn't have been identified in the unit test. If not, then it I was the second person to take the responsibility Yes, I have made a mistake and I should have identified it, but there was no creed or any kind of pressure from the project manager to any no blame game. Rather we write we wrote a solution for that and market had escaped effect and tried to prevent it from the future. Upcoming, like we included that specific scenario which was important in our regression test.

Researcher 18:09

Okay, great. Do you think this quality or this culture of not of no blame like you call it no blame? Does it have quality? Does it help achieve in in this particular team? Does it help achieve in better quality?

Participant 20 18:28

Yes, I think if we faced that situation, like two of the members from the team are facing this situation, there is a hidden lesson for the rest of the team members that wish we should take full ownership of the task. And we should try our best to you know, avoid such problems in the future. So there is a lesson for every team member.

Researcher 18:58

So how the lesson become part of the team once you learn a lesson, how does it become part of the team and you avoided in the future?

Participant 20 19:08

Like I said, in my particular case, it was a high priority test scenario, which to avoid this, which we added that scenario in our regression test and made sure that that specific thing is tested again and again.

Researcher 19:28

So you you cater for it, and you avoided in the future, right. Okay, then. Great, thank you. So let's move to the second question in the email, which is it states a member of your team can bring problem and tough issues and your answer is says yes, there are scenarios where more destinies require, so we need to decide how much depth and how much breadth we should adapt to that, to test out our particular features. Also, there are times where there are a lot of hidden bugs in our software, which mostly lies in the corner cases, okay? There are complex and not easy catchable for the development team. Can you elaborate a little bit in this? Yes, thank you

Participant 20 20:28

Welcome. One of the most major challenge we QA engineers face is that I have identified an issue, and I am reporting it to a developer, first of all, he is not in a mood to listen and accept it as a bug. So, you should have some convincing power or convincing skills to demonstrate that issue and convince him on your point. So, at times, you have to be very flexible and talking away, or even try to reproduce that issue on this machine so that he can accept it as a bug. Second thing is, I believe, for example, if I am testing a particular feature, I have to first evaluate its business importance, like if a particular thing is generating business for my client, so I think I need to spend more time on that feature, rather than spending time on feature which is very less, you know, not business generating, for example, if I am not, for example, if I have an e commerce website, I don't need to spend a lot of time on about us

Contact Us page homepage. Rather, I need to spend more time on checkout, Add Card, searching, etc, etc. So this is identified as depth and breadth of testing, how deep should I dive into the that particular feature? And how, how much I should work around that particular feature to create more complex scenarios. And most importantly, being a QA engineer, I do not try to, you know, test, one plus one is equal to two, I always try to test how can I make one plus one is equal to 11. So these are the some kind of corner cases, which I think QA engineer is more focused on.

Researcher 22:39

So let's work with this scenario. If you are pressure of time, your project manager pressure Do you have time and ask you to hurry up. And you don't have time to achieve this level of depth and breadth? What do you do? In the case of the team you've been talking about,

Participant 20 23:03

I had only one example from my past and from that particular team. So we were, you know, in a hurry in a rush to deploy the build to the production server and being a senior resource of QA, my manager asked me to, you know, send an email of release notes that we are good to go live. And our most important features were, were not tested yet. And it required like three to four more hours and we were all already running out of time. So, I clearly said no to my manager that I cannot take responsibility of unstable build. In fact, I can take responsibility of stable build, which is already deployed on production, which is V one, I will not take responsibility of V two which is unstable. I need to spend more time so that I can take full ownership of the stable version two, if you asked me to deploy it to the production server, please drop me an email. So that I have some thing to show to the higher management in the future that this is not in my in not in my ownership.

Researcher 24:25

So you confronted him because there is an opportunity of openness and the team and an opportunity of trust and honesty, or because you wanted to protect yourself.

Participant 20 24:42

At that particular time, I thought if I bring up this scenario in front of him, he's going to give me two more hours so I was not in a position to protect me but I I knew he is I'm going to draw up an email and challenge the higher management. So I knew he is going to give me some more time. Like, if we are going four hours late for a production, that is not a big deal. In fact, we are giving a stable bid to the end customer. So I knew it, he's going to listen to me. So that was you can say I was actually blackmailing him, but in a good sense.

Researcher 25:28

So because you feel safe to do so.

Participant 20 25:31

Yeah, I thought I have had I had enough authority to buy time for a stable bit like I knew my this approach is going to show my ownership and reliability for the management. Like I'm, I could have said, Okay, I'm going to maybe there was a test from the project manager that is asking me to deploy an

unhealthy build, I'm, and I'm saying, Okay, go deploy. Maybe that was a test for him? I don't know. But I think I played sensibly. Well.

Researcher 26:06

Okay, excellent, thank you. Let's move to another items on the email, which is about taking risks and initiatives on the teams. So it says if it is safe to take initiatives on your team? And you said, Yes, we are always looking to improve our processes, every resources from our teams is looking to take initiatives on its own, that are all counter and the yearly performance reports, the process involved, etc. So can you elaborate, are people comfortable to come to the team and say, I have an idea? This is my idea. Can we try my idea? Because I think it's a good idea.

Participant 20 26:56

So we had one statement, which I think I also mentioned in that particular answer is that we used to say that processes are not made for humans. In fact, you humans are not made for processes, but processes are made for humans. And we used to alter and bypass a lot of processes, just for the betterment of our quality. So when, when we were new, when our team was formed, we were you know, taking inputs from the management, we were not giving our inputs in the meeting in the sprint meetings, and we were just kind of observing after spending like, good eight to nine months together, by the way, we worked around three years together. So after spending eight to nine months together, understanding each other and had a good bond level. So understanding the business as well. So we ourselves started identifying the areas like which is which is going to benefit our company. And in that way, we suggested that we should pick that particular item, or we should introduce this in our business to generate revenue for our client. So we had a lot of examples, which were presented for to the product manager and project manager and they were accepted as well. They were also counted. This initiative part was also counted in our yearly performances. Like how many initiatives did you give in your project for the betterment of the revenue? Revenue generation?

Researcher 28:42

Do you have an example for me where someone, either yourself or a team member brought an initiative to the team and take me through what happened?

Participant 20 29:05

Sorry, I can't think of one right now.

Researcher 29:06

Okay, no problem. Let's move to the next item wishes it is difficult to ask a team member of your team for help. And you said no. As our team, especially my team members are always willing to help they are always available to resolve the the roadblock, conflicts within the teams. Any other challenges. Can you take me through how do you help each other in the industry?

Participant 20 29:41

Okay, so when I entered the software industry, I didn't know that everyone is nice, everyone is helping. I always thought that people are reluctant to talk and they are just sitting in their own comfort zone. I'll just just Like we had very bad experiences in the university, like people are not willing to share their

assignments or teach each other for, you know, like, people aren't. The guys with good grades are not willing to sit with the guys with the having bad grades, or they are not willing to share knowledge. So I had that perception when when I entered the industry, it totally changed my mind when I had an experience with different people, like everyone was willing to help you. And everyone was like, there was a race, like, who is going to, you know, make more and more impact. And maybe that was things being noticed by the management or the project manager. But everyone was very nice. And they were actually nice, they were not being just fake, nice, but they were actually nice. They were very helping, like, if I do not know anything big a QA engineer, I do not do anything, I can directly even go to my project manager like I have, I am facing this issue, please help me resolve this. So there was a culture of being openness. And everyone was, like people were seeking if there was any person, which we can help. So I had one example in my mind, being a QA engineer, we do not have to interact to the database very much. But the dev read helped me to install the MongoDB ATM as my machine and helped me write few basic theory. So I do not have to go to him every time like if I want XYZ kind of data, I do not want to disturb any of his developer or go to his mind, I can, you know, extract some basic kind of data on my own.

Researcher 31:55

That's great. Why do you think in that particular team, people would help in each other?

Participant 20 32:05

Honestly, I couldn't figure it out. Maybe they were just acting nice. Or maybe they were kind of pressure from the management or like, these psychology things are being noticed by the management. So they are reflected in annual reviews, or I, I personally, I do not know what happened. But I think people are generally nice. They are. Everyone is nice.

Researcher 32:31

Thank you what type of software you were developing back then.

Participant 20 32:36

It was actually ecommerce website, which is being updated in the UK, it is a website called [deleted to preserve the participant anonymity], which is being used to [deleted to preserve the participant anonymity]. [deleted to preserve the participant anonymity].

Researcher 33:05

Understand, and what type of quality assurance services you had on the team.

Participant 20 33:15

We were totally doing manual testing for the website for their forums for their blogs, and Android application and iOS application as well as they were operating an internal application for their mechanics, which has a list of like, almost 200 questions on the basis of which they were checking the car. So we were actually purely on the manual testing side. But I got a chance to automate few of their business scenarios using selenium with Java. And I had an experience of iOS development. So I wrote a few test cases for iOS application as well using Exodus framework of Xcode.

Researcher 34:10

Okay, thank you, and what other software engineering practices you had in place for software quality? For example, did you have code analyzer that you have?

Participant 20 34:22

No, no, no, no. Let me let me let me clarify you. Code Analysis and code reviews are something related to the developers only. What we do what we did, we used to write test cases, like when there is development ongoing from the developers, the QSR, writing the test cases for the whole sprint. Once we are done with the test cases, they were evaluated by the dev lead, who had all the knowledge of the tickets, so he had a list of like 20 items in which he used to check there. The QA resources are added this way this will this will. So after that scoring, he gave us a rating that you have, from 20, you have adopted 15 points or 16 points out of the 20. So he used to give us a rating. And after the sprint, we used to conduct our testing practice review, there was another checklist of testing practice review, which we which some other team from the company came, the team of QA come to comes to us and evaluate the overall process over testing throughout that sprint, which was another set of like 25 questions. And we had already.

Researcher 35:44

Okay, so in this environment, how was the relationship with the between the developers and the key aides?

Participant 20 35:54

We were three sources, three key resources. One of my, there was one lead, and second person was me. And the third person was my junior. So my lead was never been good friends with a developer, I was always been good friends with developer, and the third person was neutral. So I think it's the relationship between the keywords and developers, there are only two types of relationships which can be made between a dev and a QA, when can be very friendly and when can be very, you know, toxic for enemies kind of relationship. So I knew if my lead is enemy of the developer, I can be spread, you know, to get my work done. You can see I was selfish, for my my own tasks. So I had to be friends with the dev team.

Researcher 36:57

So but other colleagues who have not Do you do you do? Do you have an idea why that particular colleague didn't have a good relationship with developers.

Participant 20 37:10

Like I said, in the start of the call, the developers are not ready to listen, or they are not ready to accept the bug. As you are talking to them, so maybe my lead was short tempered, and maybe he has some more responsibilities than me. So he chose that part to be you know, to be strict with the dev team and and give the give them no to about the work. Maybe that was the case. I knew it. If I if you have to, you know, get your tasks done. You could be polite, and you should be friends with the tech team.

Researcher 37:51

Okay, great. That was great. Participant 20. Thank you a lot. I don't have more questions. Do you have any questions for me? Or would you like to add anything that we haven't discussed?

Participant 20 38:05

No, I actually like very much talking to you and sharing my past experiences with you. Just give me a brief introduction, like what you are doing and what is this all activity about?

Researcher 38:26

Well, it is a research project. What I'm researching is I'm looking at whether agile teams help each others, whether they admit mistakes, whether they take initiatives. And I'm looking at whether this would help the software quality of the product. So I'm talking to a lot of people and interviewing people and trying to understand in a particular agile team, how do they manage? Or how do they go about admitting mistakes? How do they go about helping each other? How do they go about bringing up problems and etc. So this environment of safety, which we call it safe, feeling safe, feeling comfortable, like you did with your manager, you feel comfortable to question his decision, whether these things help as our teams to meet the expectations on software quality. So what we do we interview people and we analyze the data, the interviews, and what we do we propose findings and we write papers, academic papers and the topics. Okay, there wouldn't be an academic paper in this topic.

Participant 20 39:55

Perfect. It was very lovely talking to you, Researcher.

Researcher 39:59

Thank you very much I wish you a good night bye

Participant 20 40:02

Thanks bye bye