

Interview Introduction and small talk #00:00:58-8#

I: Q1 About the process #00:01:01-9#

B: Explain about the process... #00:03:55-3#

I: Q2 Expectation #00:04:04-2#

B: aspects... #00:06:00-8#

I: Q3 #00:06:17-4#

B: hmm what do you mean? #00:06:27-9#

I: explain the question again #00:06:55-8#

B: the good thing: is that it helps you catch the difference which might be overlooked while testing, testing focus only on output. (...) another thing is that most developers are not aware of some of the common comments standard that are followed in the industry so maybe it becomes to my responsibility when developers are not ready to such code to give some remarks/ comments within the code review to make the developers aware about some standards. Along with that you know sometime they are not able to figure out any un-tradition or security issues. so these things must be highlighted by the arch. or the analyst and they should ensure that or be aware of that, that not only syntactical issues can cause problems but even the functionality should be very well understood. (explain...). #00:10:03-9#

I: Q4 #00:10:15-0#

B: they ask the developer to fix it ... #00:11:18-4#

I: Q5 #00:11:51-4#

B: i guess there is 10 to 12 people in my team and they are combination of freshers + the experiences people. They're not very specific in one layer they are working across all the layers. (explain what they are working with...) they all have a very wide set of knowledge about the application. #00:12:34-2#

I: how many members are there in your team? #00:12:37-3#

B: i guess 12. #00:12:38-3#

I: and how many of the are developers and how many are reviewers? #00:12:41-6#

B: i guess there 2 to 3 people who do the review part, and the rest are developers #00:12:57-5#

I: can you tell me more about every one of them #00:13:06-6#

B: okay the reviewer is a kind a person who has spent around 4 to five years within the industry and has worked up in production of the app and he has a good understanding of the language or the framework we are using and deve. have 0 to 3 years of experience and in the job they are exploring the technology which we are using. the reviewer should be aware of each technology we are choosing within the application and about the various system design. So they can guide the developers and make sure no matter what occurs on the code which given by the developers is \*not clear\*. #00:14:13-9#

I: tell me about the reviewers in your team, do you work with all of them? #00:14:20-7#

B: i work with all of them. #00:14:29-8#

I: Q6 #00:15:01-3#

B: i will go with the person who have more experience with in the application. he will be able based on his previous experiences, he will highlight quality "as part of the code' so i will him based on these parameter. #00:16:09-4#

I: Q9 #00:16:41-3#

B: sometimes it happen that ...( requirement changes) #00:17:50-1#

I: since when do you work with this team #00:17:57-4#

B: almost one year #00:18:05-4#

I: Q7 #00:18:19-7#

B: ... (standard were not taken care of) that was a rebug should be rewarded by the developers, they should make sure while sending a code to be review that at least these basic things are taken care of. #00:19:54-7#

I: one more one #00:19:58-8#

B: performance issues... Comprehension about code changes... #00:21:15-6#

I: and are you facing with the same difficulties #00:21:38-5#

B: no. #00:21:45-3#

I: accepting the request #00:22:16-5#

B: ... #00:23:06-2#

I: communication? #00:23:11-0#

B: per email. Dialog #00:23:21-1#

I: skype? #00:23:23-5#

B: no, not same place. #00:24:11-9#

I: Q12 #00:25:31-9#

B: testing --> functional aspects fulfilled. When it comes to communication the tester will talk in term of business uses or desired functionality. Whereby reviewer should play the role of a mentor not just highlight the issues with respect to the code which have been written but the have to guide the team members to the best practices that fellow in industry and the aspects which are applied by the organisation. So it is the responsibility of the reviewer to make sure the are mentor the team member and enable them to take up the similar activity for the next set of layer and be able later on to be in a position to do the reviews. so the reviewer should no more or less work as a mentor. #00:26:52-5#

I: team members? two or three? #00:27:12-7#

B: a single person who do the review activity. #00:27:17-8#

I: three things about the reviewer #00:27:27-3#

B: i am the one who do the reviews. #00:29:28-9#

-----Part Two #00:00:12-0#

I: Q14 #00:00:55-2#

I: generally i would like to review the code from the least experience, because the chance that the people, which came new to the project and do not know the standards, might end up making more mistakes. and that might create instability within the App. I would like to review their

code and make them aware of the practices we are following and also ensure to understand the comments that are coming from them.

#00:01:45-4#

B: would you do that for the advantage of the project but which one is more pleasant to you to review code from experienced people or people with less experience? #00:01:48-1#

I: i'll say it is the same. inexperienced people will learn sth and would use it again. (... unclear) we will help each other rather than working with different dimension. #00:02:34-9#

B: question: how do discuss the change request? #00:02:53-6#

I: (...) everything is recorded in JIRA and we expect that the developer would read them and would refer to the reviewer if he do not understand anything. and then we start sharing our knowledge, why we record with this particular observation. so we don't discuss every point that is noted in the excel sheet because most of them are expected but when Confusion there will be dialog about it and it is necessary. #00:04:02-2#

B: can you give me an example #00:04:12-2#

I: yes. (...) #00:04:46-2#

B: Q11 #00:05:09-1#

I: I make sure that if the changes is going to be critical so i write the justification, why is this request proposed. So something it is very comprehensive. (...) and that is how we make sure that the developer won't feel offended. instead of having personal opinion we have to prove it. (...) #00:06:42-5#

B: qu: you told me last time that the job of reviewer is more like a mentor. #00:06:57-5#

I: Yes #00:06:57-5#

B: can you tell me about last experience as a mentor? #00:06:57-5#

I: the premier reason behind that, based on the last experience that i observed, that people who are very new to the project are those who are not so comfortable with respect to coding the complex changes, they most of them end up making mistakes and it could be , you know, very small mistakes, like Dokumentation... and sometime when it comes to functionality, it is working fine but when it comes to maintainability of the code it become a nightmare, that people have to start finding all the references of for example what is the real purpose of the variable and so on (....) #00:09:23-2#

B: tell me about one time, when a developer didn't name the function or the variable correctly #00:09:53-0#

I: (...) so it was necessary to do a refactoring with respect to the code because there was many things that were not considered while writing the code. So i had to guid the developer with respect to layer concerns and tell him how the code should be refactored and tell him about the side effects of coding into another layer and how it impact the code. #00:10:39-4#

B: how do you react when they repeat the mistakes #00:10:53-6#

I: yes it becomes very difficult if they are repeating over and over the some mistakes. But after all ("not sure about it") developers are human and we should take away the emotion from them and stop shouting at them, so we should send this message in a professional manner. some time when sending the change request to a developer and he/she didn't understand the real impact of it. (..) so we reviewer should make sure that they understand the various reason that cause this mistakes. we should maintain our goal and tell them which situation which might

create problem if he pushed such mistakes in the testing environment.  
So we need to explain the various aspects. #00:12:09-6#

B: how do you make sure that the developer really understand the  
reason behind the change request. #00:12:15-6#

I: What I do i crash the code change with them to make sure they  
understand it and if they have questions, in that case his question will  
answered. (...) if he repeats the mistakes i contact to understand why  
this is happening. #00:13:31-5#

B: anything we want add #00:13:38-0#

I: \*recommendation of a book: think like a programmer\* #00:13:48-0#