Time: August 22, 2017, 2:00 p.m. – 3:30 p.m.
Location: BigCompany Quality Assurance Group
Participant:

A1: first author of the paper
A2: second author of the paper
B1: Project manager of the quality assurance group
B2: Project manager of the crowdtesting group
B3: Assistant project manager of the crowdtesting group

Topics plan to discuss:

Current practice and Pain points of crowdtesting management in BigCompany crowdtesting platform

Interview details:

A1: Could you describe the crowdtesting process in BigCompany crowdtesting platform?
B2: The customers prepare the crowdtesting task (including the software under test and test requirements) and give it to us. We then distribute it on our crowdtesting platform. Crowdworkers can sign in their interested tasks and submit test reports, typically summarizing test input, test steps, test results, etc.
The customers can be the third-party company, who want have their developed apps tested in our platform. The customers can also be the test managers from other groups of our company. We can help them test their apps with high cost-efficiency.

A1: How to decide whether a crowdworkers can or cannot get paid?
B3: In our platform, also in most platforms as we know, the crowdworkers are paid by the bugs. This is to say, if their submitted reports contain a bug, they can get paid; if their submitted reports do not contain a bug, they could not get paid. Currently, we do not distinguish whether their submitted bugs are duplicate, that is to say, as long as they submit bugs, they can get paid. And we do not distinguish the severity or important levels of their submitted bugs, that is to say, as long as they submit bugs, they can get equally paid in the task. We assume this is to encourage the participation of the crowdworkers.

A1: How to decide when to close a crowdtesting task?
B2: Currently, we do not have some precise criteria about when to close a crowdtesting task. Usually, we will set a fixed number of days (e.g., five days) or a fixed number of reports (e.g., five hundreds reports). When either of the criteria is satisfied, we will close the task.

A2: How do you think of the current practice of task management?
B1: It is quite like the black-box testing. We can receive constantly arriving reports, but we are often out of clue about what is going on within the crowdtesting, for example, the remaining number of bugs as yet undetected, or the required cost to find those additional bugs.
I think the management of crowdtesting is conducted as a guesswork. And this frequently results in many blind decisions in task planning and management. For example, in order to ensure all the bugs are submitted, we usually set up a large threshold for the duration for closing a task. However, we find that almost all the reports submitted in the later stage of the task are duplicates, which

contributes very little to the testing results, but still needs to be paid, to the cost-effectiveness of current practice of our crowdtesting is quite low.

B2: Let me add something about the current practice of crowdtesting. We find that for different tasks, the duration for achieving certain test objectives is quite different. For example, for some tasks, it only takes maybe 24 hours to find all bugs; while for some tasks, it might take 6 days to find all bugs. Meanwhile, the number of received reports also differs a lot. For example, for some tasks, it might have received 600 reports to find all bugs; while for some tasks, it takes only 100 reports to find all bugs. Therefore, we have realized that by setting a unified threshold for all the tasks is not a good practice. But we could do nothing because we did not know how to tackle it.

B3: I also want to say something about the pain point of the crowdtesting. Currently, we not only do not know when to close a crowdtesting task, bug also do not know what is going on. I think what is going on within the task is equally important as the final decision about task closing. I think we need to have greater visibility into the crowdtesting processes.

A2: We also want to know whether the test objective for different tasks are the same?

B2: Actually, I donot think so. For some customers, they have sufficient money and they require their apps to be in high quality, therefore, they would like to spend more money even extra money for the test, and ask us to detect as many bugs as possible. For some customers, they hope to get the test done as soon as possible so that the app can come to the market soon to seize market share. In this case, they just require maybe 80% of bugs are detected out because they know it is always costly, both in time and money, to find the remaining little number of bugs.