

Gain of Sufficient Test Coverage by Leveraging Crowd Testing

Dr. Vijay Bhardwaj

Guru Kashi University, Talwandi Sabo

ABSTRACT

Crowdsourced Testing or Crowd Testing has been gaining more importance because of the value it adds to the in-house testing results. Testing performed by the crowd of expert testers has been termed as Crowd Testing. Product or Project owners leverage the power of crowd testers to assess the quality of their application in short span of time with minimal costs involved. In this paper, we are going to discuss the current approach followed in any crowd testing project. One of the key challenge observed in the crowd testing project is 'Test Coverage'. Testers used to test and find handful of bugs in the AUT (application under test) within shorter timelines but there is no mechanism available to measure the Test Coverage provided at the end of the test cycles. We are going to introduce a new approach by which we can ensure that the Test Coverage has been sufficient before getting into markets. Also, we are going to share the results of a questionnaire submitted to the expert crowd testers in assessing the effectiveness of this approach when implemented. This approach will help us in addressing the challenge of Test Coverage to a greater extent in any crowd testing project.

Keywords: Software Testing, Crowdsourced Testing, Crowd Testing, Test Coverage

I. INTRODUCTION

Crowdsourced testing is an emerging trend in software testing which exploits the benefits, effectiveness, and efficiency of crowdsourcing and the cloud platform. It differs from traditional testing methods in that the testing is carried out by a number of different testers from different places, and not by hired consultants and professionals. The software is put to test under diverse realistic platforms which makes it more reliable, cost-effective, fast, and bug-free. In addition, crowdsource testing allows for remote usability testing because specific target groups can be recruited through the crowd.

This method of testing is considered when the software is more user-centric: i.e., software whose success is determined by its user feedback and which has a diverse user space. It is frequently implemented with gaming, mobile applications, when experts who may be difficult to find in one place are required for specific testing, or when the company lacks the resources or time to carry out the testing internally. (Source: Wikipedia)

II. CHALLENGE TO BE ADDRESSED

Crowdsourced Testing has begun to dominate in the Software Testing field due to the growing number of new devices and huge number of mobile applications had been evolving day by day. Crowd Testing uses three terminologies predominantly:



Crowd Testers – People or Individuals who are experts in the field of Software Testing and willing to contribute towards quality of application by participating in Crowd Test Cycles

Crowd Seekers – Project of Product owners who need their application to be tested by different testers using different mobile or computer configurations

Crowd-testing platform – A medium where in Crowd Testers and Crowd Seekers meet, collaborate and get their things completed

As the crowd testers are from different location and with different set of experience levels, they test based on their expertise. If the testers, find any defect prone area in the application then they used to concentrate more on that particular area to find and report bugs. Because of this tendency, at the end of any test cycle crowd seekers were left with huge number of bugs in specific functionality. This leaves the project owners in a confused state whether they got sufficient coverage in other areas of the application. Based on these results, they have two things to decide:

- a) Crowd Testing has found more number of bugs in particular are and that needs fix before moving into the market
- b) Crowd Testing didn't report considerable bugs in other areas of the application. This might be due to the below factors:

Assumption 1 - They didn't explored other functionalities in the application

Assumption 2 -Really there are no bugs to report in other functionalities

With these results and assumptions, it becomes difficult for the crowd seekers to decide on the coverage they got for their application based on this crowd test cycles. Once they get an understanding on the coverage they got for their application, then they can decide to go into market with more confidence.

In this paper, we will review the existing approach toward crowd testing. The proposed model wherein we take different approach in accessing the crowd testing cycles. The benefits reaped out of the newer approach in attaining the sufficient Test Coverage the crowd seekers anticipate. Finally, the results of the questionnaire distributed among different crowd testers and their views towards the implementation of this newer approach.

III. CURRENT APPROACH TO CROWD TESTING

As the advancement in the technological arena has given an enormous increase in the mobile devices, applications and other online services. In order to test any newer application among the available device or computer configuration becomes a tedious job. Most of the in-house testing team will have pre-defined browser, device combinations and environments. This will add more cost to the project team to buy different new devices to get things tested completed before making it to the markets.

Crowdsourced Testing can be utilised in these areas to avoid the cost incurred in obtaining those different gadgets to get their testing completed. Crowdsourced testing is the process of outsourcing the testing activity to the crowd of expert testers with different devices spread across different geographical locations. This way, the crowd seekers get the different combinations of devices tested along with the effective feedback from different testers across the globe.



Crowdsourced Testing can be leverage for functional, non-functional and any kind of usability testing as well. Crowd Testers will be submitting a feedback on their experience with the test application which will be helpful for crowd seekers to analyse upfront before release their products.

With those advantages of Crowd Testing, we can't completely ignore the conventional testing of the crowd seekers. Crowd Testing can be utilised as a supplemental testing activity on top the testing carried down by the conventional testing team of the product owners. Crowd Testers will be performing exploratory testing on the applications to uncover any hidden defects that leaks through the conventional testing.

We have several reputed organisations that provides Crowd testing platforms to help crowd seekers to get their application testing. UTest(crowd testing platform) has one of the largest community of testers who can deliver Functional, Security, Usability, Load and Localisation testing. We have different organisations providing a platform for crowd testing services.

At present, the following approach has been adopted by most of the crowd testing platform:

- 1) Crowd Testers / Crowd Seekers register in any of the available Crowd Testing Platform
- 2) Crowd Seekers submit their project and request for the test results
- 3) Crowd Testing Platform allocates a Test Manager to run this project
- 4) Test Manager sends out invitation for available Crowd Testers based on the project scope and pre-requisites (if any)
- 5) Crowd Testers accepts the invitation and start with their testing to submit bugs in the AUT
- 6) Usually the test duration will be shorter (Eg: 2-3 days of time)
- 7) Once the duration is completed, Test Manager closes the test cycle
- 8) Test Manager reviews the bugs reported by each of the crowd testers and either Accept or Reject those bugs
- 9) Test Manager consolidates the bug reports and send it across to the Crowd Seekers
- 10) Few Crowd Testing Platform provides access to Crowd Seekers to directly access the Bugs from the platform as soon the Crowd Testers reports a bug
- 11) Crowd Seekers reviews and Approves or Rejects the bugs based on their expectations or needs
- 12) Finally, Crowd Testers get paid based on the number of bugs they logged / the test cases they have executed.

In this model, the Crowd Seekers will be getting only the bugs and they couldn't confirm anything on the Coverage provided by the Crowd Testers. They will be under the assumptions that they might have received the ROI. But - The coverage, still remains a darker area.

IV. PROPOSED MODEL FOR CROWD TESTING

In the existing approach, Crowd Seekers launches the test cycle to get the test results based on their entire application. Instead of releasing the entire application, crowd seekers can logically divide their application based on their functionalities, impacted areas or business priorities. So that they can request for Crowd Testing cycles for each of these groups. Once the test cycles has been concluded for all these groups then we might assess the Test Coverage for our entire application.



Let's consider an example with an E-Commerce application. We can group the e-commerce application based on its functional groups as below:

- a) Product and Product details
 - i) Filter
 - ii) Sort
 - iii) Quantities
 - iv) Landing Pages
- b) Basket and Cart
 - i) Adding a product to cart
 - ii) Viewing cart / basket
 - iii) Updating or removing products from cart
 - iv) Discounts / Promo codes
- c) Checkouts
 - i) Credit / Debit Card payments
 - ii) Paypal
 - iii) Cancelling payments
 - iv) Guest / Registered user checkouts
- d) Account Information
 - i) User Details including their billing address
 - ii) Email Preferences
 - iii) Notifications
 - iv) Validations on user details
- e) Miscellaneous
 - i) Delivery / Shipping Charges
 - ii) Tracking Orders
 - iii) Returns Processing
 - iv) Store Locators

In this way, we have grouped an e-commerce application into five functional groups and sub-groups. Now, Crowd Seekers can request for a test cycle for each of the functional groups individually. So that the testers will test each of these functional groups one at a time and report bugs. This ensures that each of the crowd testers test the entire functional groups without providing an option for testers to skip any of the features.

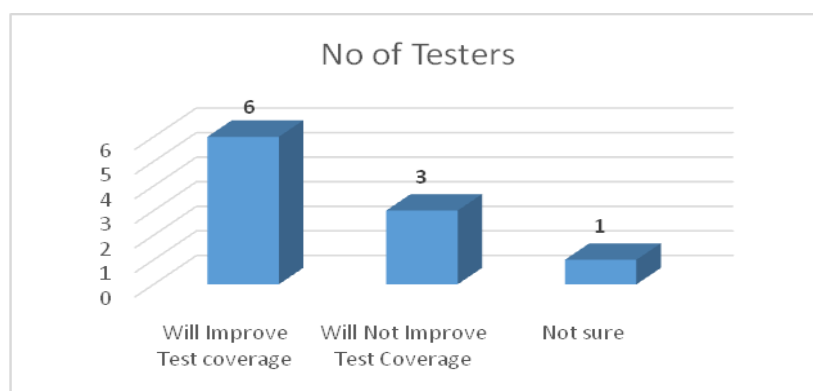
E-Commerce Application				
S.No	Test Cycle	Functional Scope	Start Date	End Date
1	Cycle 1	Product and Product Details	Day 1	Day 2
2	Cycle 2	Basket and Cart	Day 5	Day 6
3	Cycle 3	Checkouts	Day 9	Day 10
4	Cycle 4	Account Information	Day 13	Day 14
5	Cycle 5	Miscellaneous	Day 17	Day 18
6	Cycle 6	Complete Application	Day 21	Day 22

Test Cycles and duration plan

By this approach, the cycle duration might get extended. But the results will across different functional areas of the application. With this result, Crowd Seekers can be assured of the Test Coverage to sufficient level. This approach can be implement to any application irrespective of the domain and complexity of the application. The cycle duration and functional groups can be altered based on the requirement of the individual project or crowd seekers.

V. QUESTIONNAIRE COLLECTED FROM REPUTED CROWD TESTERS

In order to ascertain the effectiveness of this newer approach, we have prepared a questionnaire. Circulated this questionnaire to ten different crowd testers to answer them. Based on their opinion we have consolidated the results in a tabular format:



Questionnaire outcomes in tabular format

VI. CONCLUSION

We have reviewed the current approach towards Crowd Testing by crowd seekers. We have also suggested a newer approach towards Crowd Testing which might improvise the Test Coverage obtained at the end of each test cycles. This approach can be implemented in one of the Crowd testing platform to realise the benefits of test coverage. Once the results are satisfactory then this can be implemented across different crowd testing platform to enhance the test coverage through crowd sourced testing. The questionnaire serves as the proof where in majority of the Crowd Testers feels that this might enhance the coverage provided for an application in crowd test cycles. Once the Test Coverage challenge has been addressed, then the Crowd Sourced Software Testing or Crowd Testing will reach even higher levels in the upcoming days.

REFERENCES

- 1) M. Sharma and R. Padmanaban, Leveraging the Wisdom of the Crowd in Software Testing. CRC Press, 2014.
- 2) K. Mao, L. Capra, M. Harman, and Y. Jia, "A Survey of the Use of Crowdsourcing in Software Engineering," RN, vol. 15, p. 1, 2015.
- 3) S. Zogaj, U. Bretschneider, and J. M. Leimeister, "Managing crowdsourced software testing: a case study based insight on the challenges of a crowdsourcing intermediary," J. Bus. Econ., vol. 84, no. 3, pp. 375–405, 2014.
- 4) "uTest." [Online]. Available: <https://www.utest.com/>. [Accessed: 01-Nov-2018].