How to Start a Bug Hunt?

Pradeep Soundararajan
Senior Software Engineer, Flextronics Software and Systems, Bangalore
Soundarararajan.Pradeep@flextronicssoftware.com, Pradeep.Srajan@gmail.com
http://testertested.blogspot.com

Purpose - This document makes an attempt to help out Testers who are on the verge of understanding the system, designing test cases, execute and report the test results. If you take the pain of reading this document completely then you shall be able to judge whether your plan for the upcoming challenges is a good one or you may also be able make the necessary corrections in the plan you have in mind or you may also be able to judge, Where is Pradeep Soundararajan going wrong?

Abstract – As Murphy's Law states, "**Anything that can go wrong, it will**", but it is good on everyone's part to ensure that they are not a major contributing factor for anything going wrong. (Let's not fight against nature). Testing is considered to be a craft and it's about how good your methodological approach is, to achieve a good result, matters. In order to test we need to be methodological and methodology comes through effective planning (moreover practice).

Here the focus is on **'How to prepare before a war (bug hunt)?'** which I expect would help in building a methodological thinking towards preparing for a testing cycle in a simple way.

Introduction – If you don't know where you have to go it is tough to choose a road, similarly, let me introduce where we need to go. It is simple; we need to go to a place where your manager says, "GO ahead" once you have done your homework towards test case design and analysis. Also to know what is good, we need to know, what is bad? A bad analysis or bad set of test Cases are the ones that lead to uncovering less number of bugs any system has. Now it is easy to define a Good Analysis and Good Test case. A good analysis as you would have thought by now is a good understanding of the system that needs to be tested and good set of test cases that operate the system to its full operational limits (even beyond).

Now there are lots of questions that we need to come up, starting from

“How do we do a Good Analysis?”

“How do we write Good Test case?”

Answering such relevant questions would make this document **good**. Isn’t it?

How do we do a Good Analysis?

Let’s think in a simple fashion and co-relate with the work we have to do.

Step 1 – During our school or college days, we have taken up a lot of examinations and can you remember how we used to prepare for the exam?.

To my understanding, exam is basically a set of questions that a student needs to answer. For sure you and I were not aware of the questions unless we got the question paper on the day of exam. This factor of not knowing the questions asked in the exam beforehand made us to reading the whole textbook and sometimes made us refer to articles, journals…what not?

Similarly, here we need to read a lot of documentation to face the exam (Testing phase).

Now another question arises –

“**How do I know whether I am reading the right documentation?”**

Step 2 – When we wanted to score high marks, we would have selected to read a book written by an expert in that field or some standard book that we were suggested to read by our Lecturer/Professor.

Similarly, here we need to ask the experts

“What document should I read to test the system well?”

If you are a bit experienced or smart, you would know that you should read the spec, understand it well and then ask the experts “What more should I read?”

Now the next question arises

“How do I verify that I have a good understanding?”

Step 3 – Again dating back to our childhood, we would have taken our textbooks to someone, who wished to listen to our answers for the questions the book had at the end of every lesson. We used to be happy and confident whenever we heard “Good, dear child. You are good at these lessons”. We would then add more confidence by taking up mock exams, isn’t it?

Similarly, here we need to present our understanding and share ideas, with the team and also request some experts to witness the presentation. At the end of presentation, a new confidence level is visible if the team is confident about you.

Does it stop here? No!

We also need to document our understanding and circulate among other Team members and anticipate their comments on our understanding document.

Aren’t you asking, “What is the next question?”

Here it is, for you…

“How to write the Test Cases?”

Step 4 – When we come out of the exam hall, most of us say, “the paper was tough/easy”. If the paper is set tough we see many students failing the exam along with us. Isn’t it? The reasons being the questions were tough and we should have prepared/designed our strategies for the exam in a better way. Probably we should have referred to more topics/books/articles to pass the exam with the same tough set of questions.

Similarly, tough test cases are the ones that make a system fail (not always). As a person who is setting the question paper for an exam, you should prepare cases that will lead to uncovering much failure (bugs, as we may call).

To write good test cases, we need to look at spec, design, system, feature, scope of testing and so many aspects. We should ensure, those designers and developers feel "Yes, I did miss this point, thanks for finding it for me"

Now again another question, perhaps, last of this article (happy?)

"How do we execute it well?"

Step 5 – Even though you write your exam well, say if the person correcting your answer sheet did not evaluate your answers well, you usually end up getting less marks than what you deserve (desire), you do re-apply for correction and then you get more marks.

Similarly, you need to execute the cases well. If you overlook at a test case and execute something else, you will end up loosing marks (I wont repeat that I am referring to bugs when I say 'marks' here)

Conclusion – Assuming there are very few points that I may would have missed in "How to start the bug hunt?" I also hope this document to be beneficial to all of you in someway or the other. Let us hope to give a good, methodological effort and improve the quality, more than what we could have. (Not by the effect of this article though, just in case this did not add value to you)