

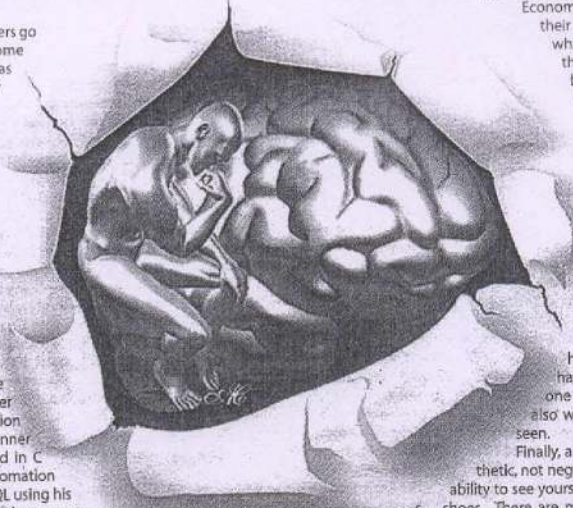
THE MIND OF A SOFTWARE TESTER

By KRISHNA IYER

"Failed developers go on to become testers. This was a maxim I heard a few years back when I joined the IT industry as a tester. No one says such things today. Today Software Testers are looked up to rather than looked upon. Today a developer would tip toe to a tester and discuss with him in advance about what he could do to make his code error free. Today a tester develops an automation framework in Winrunner using his background in C and wires in his automation framework into MSSQL using his SQL skills. Today the industry recognizes Software Testing as a discipline and appreciates Software Testers as much as designers or developers and pays testers as much. Open any edition of Ascent (a recruitment supplement) and invariably you shall see multiple openings for testing.

For those aspiring to make or grow a career in Software Testing, understanding the mental makeup of a software tester is necessary. Classic books such as "The Art of Software Testing" shall tell you that the mind of a software tester has to necessarily be a critical mind. These books and therefore hundreds of trainers who tom tom these preachings without a second thought expound that testing requires a destructive mindset and a negative attitude. They emphasize that the focus should be on how to be destructive in thinking; how to fail test conditions, etc. Though it seems to be a good piece of advice, there is a lot of unseen damage that this stream of thought is doing. Testers read these books and make it an obsession to seek the negative and sadly become too focused only on finding defects (and maybe even categorizing every minor defect as critical or fatal). The ability to find defects is just a small part of what a tester needs to bring to the table. There are more important faculties that aspirants and even existing testing professionals should focus on. Let us examine some important qualities that a testers mind necessarily needs to possess to have a successful and rewarding career.

A tester has to be creative, not necessarily critical. Creativity is foremost for a tester. Creativity comprises of flexibility, variety and originality in ideas. Flexibility of ideas signifies the ability to generate a number of test cases. Variety of ideas signifies the ability to cover a broad spectrum of test cases - including usability, security, and performance rather than just focusing on functionality. Originality of ideas signifies the ability to think of test cases no one else has thought of before and more importantly the ability to generate unique approaches to test software. Every tester should train himself in these three aspects of creativity. In ZenSOFT, we send our testers to a ten evening soft skills training program where the trainer asks all participants to think up of as many circular objects in one minute as they can. He then asks them to count the number



of objects. The ability to generate a number of ideas in a given period of time is flexibility, he explains. Participants average ten to fifteen on the flexibility score. He then asks them to count the categories of objects. Tennis Ball, Basket Ball, Cricket Ball, etc. are all to be counted as one category. Participants average seven to ten on the variety score. He then asks them to share at least one from the list that they think no one else has thought of. Participants average only one or many times nil on this score of originality. Originality is the key. A tester has to be trained to think out of the box.

A tester has to discover, not destruct. The word "discovery" signifies that defects are already there. You have to just find them. The question is how do you train testers to discover more? Discovery again could be broken into three aspects - an eye for detail, a nose for sniffing out bugs and a brain for understanding logic. For a tester to train himself to have an eye for detail, he could begin with proof reading articles. Take up any newspaper or magazine article like this one. Find errors in English - grammar, punctuation, sentence formation etc. For e.g.: Did you notice that there was no "period" (full stop) before the word "etc" in the preceding line. A trained eye is one that can consistently and continuously look at all aspects and keep finding what is not working the way it ought to. A nose for sniffing out bugs - the quality of exploring is more the nature of a person and may be difficult to nurture. In ZenSOFT, our interview panel has created a set of questions to find out whether the candidate has this attribute inbuilt into him or not. The practice we have in place to improve this ability is the simple practice of peer testing - have one tester test the same thing someone else has. This helps both compare what they missed out and opens their mind to sniff better in the future. As in originality for creativity, the third attribute of a brain for logic is the key to discovery. You can't discover fully unless you understand the business logic of the functionality being tested. A number of best practices such as involving testers right from the requirements stage, training testers in domain knowledge, etc. already exist in the industry. Aspiring testers should minimum begin with reading magazines such as Busi-

ness Today or newspapers such as Economic times to increase their business acumen which in turn affects their ability to garner business logic quickly. Bringing all these three together, an eye for detail helps you to see in depth what everyone else has seen. A nose for sniffing helps you to see what no one else has seen. A brain for logic helps you to decide what else you have to see once you have seen what everyone else has seen and also what no one else has seen.

Finally, a tester has to empathetic, not negative. Empathy is the ability to see yourself in someone else's shoes. There are many testers whom I have met who are too focused on finding defects and cannot understand that testing is just one part of the whole software development cycle. A tester needs to have the ability to see himself in the business analyst's shoes to understand requirements and the purpose of a particular functionality. A tester has to empathize with the developer when he fills a defect report so that he is aware of what the developer goes through when he reads an unclear description of a defect. A tester has to most importantly see himself in the user's shoes to envisage all the possible uses that may cause a defect. To develop empathy, a tester has to speak more to developers, designers, business analysts, users. More importantly, he needs an exposure to various stages of the software development life cycle. A practice we have in ZenSOFT has helped us train our testers to be empathetic to developers specially when writing defect reports. The team lead selects the defects written by one tester and asks another tester to read the same and explain what he understood to the whole team. This usually brings in a lot of humour, but most importantly gets everyone to see what pains developers go through. The best way to improve empathy is just to be constantly aware of other people involved and with time this quality too can be mastered.

To summarize, for testers, being creative is more important than being critical. Discovery is more important than destruction. Being empathetic is more important than being negative. Each of these skills has further attributes that can be mastered with time so that you have the mind that takes to be a successful software testing professional. Change your mind and keep the change!

The author is the CEO of ZenSOFT Services, an independent software testing company. He could be contacted on krishnai@zensoftservices.com