

# Testing Mindset

11 July 2011 · bug reports, exploratory testing, learning, psychology

---

Once upon a time there was a young and naive tester, he was new to the world of software testing. He often felt he didn't have what it took to be a tester. Sure, he found the odd bug, and he enjoyed his work, but he also often missed bugs, issues or problems.

After a while, he admitted to himself that this was a problem, and decided to seek help. He stood up from his desk and walked over to his test manager's desk. His manager was wise and experienced. He was the [Mr Miyagi](http://en.wikipedia.org/wiki/Keisuke_Miyagi) ([http://en.wikipedia.org/wiki/Keisuke\\_Miyagi](http://en.wikipedia.org/wiki/Keisuke_Miyagi)) of testing, and as such was always offering zen-like advice for his team. A simple question about where the stapler had escaped to could turn into a somewhat baffling series of [Haiku](http://en.wikipedia.org/wiki/Haiku) (<http://en.wikipedia.org/wiki/Haiku>), leaving our young tester baffled.

Our novice explained his problem, and his concerns about how maybe he wasn't cut out for testing. The wise test manager smiled, thought for a moment and then opened his little Mole-skin notebook. He turned carefully through the pages, settled on a page, looked up and said: "I overheard there was an issue with the login screens. A few users have reported issues, but not so many to suggest it was completely broken. Can you take a look?"

The tester was young and inexperienced, and this simple mis-direction worked well. Five minutes later he had isolated an issue with the login pages on the company's website. It fitted the bill, it only affected users in a minority web-browser, but would stop some users from logging in. He reported this issue to the programmers to be fixed.

The tester settled back into his routine, and got on with his normal testing work. But it wasn't long before his melancholy returned. He had a couple issues pass him by, despite being especially diligent in checking the conditions of satisfaction and even automating tests for several of them. He thought back to his last attempt at getting help from his test manager and realised he hadn't received an answer or ANY help for that matter.

He walked over to the test managers desk, and asked again for the test manager to help. The test manager looked somewhat puzzled, then lifted his notebook from his pocket, and leafed through the pages. Again he settled on a page, paused, and stated: "We got some feedback that the search function wasn't working. We don't have much to go on, but some users 'saw an error' whatever that might mean."

The tester, was somewhat annoyed at this second attempt at mis-direction and insisted on resolving the problem he asked about FIRST. This was after-all his career they were talking about! So the test manager suggested they talk about it after he'd investigated this latest issue, saying it probably wouldn't take long to sort out if there 'was a real bug' or not.

Within 20 minutes our tester had noticed that certain search terms caused an error in a popular web browser. While not an issue that would appear with every query, it was certainly one that would affect a large section of the users at some time. He quickly reported the issue to the programmers and marched back to the test managers desk.

Our tester started the conversation and explained that he didn't want to be chasing other peoples bugs. He wanted to find his own. Again the test manager smiled his zen-like smile, and picked his Moleskine (<http://www.moleskine.co.uk/>)™ notebook and handed it to the young tester. The young tester was about to lose his temper, when the still calm and collected manager suggested he flicked through the pages.

The now confused tester did just that, expecting to see notes from meetings and bug reports like those he had been given earlier. But instead he just saw one or two words on each page. The words didn't make a sentence or any other pattern - until the tester realised. They were just titles for sections of the website.

One page had "Login" written on it, another had "SECURITY" printed out in capital letters, and yet another page had "Search" scrawled across it. In fact many of the titles would apply to any or at least 'most' websites. The book was a checklist, the test manager wasn't giving him 'second hand' bug reports, he was enlightening him. All software has problems, ambiguities and bugs. So by suggesting a feature had a bug was just the test manager's way of getting his apprentice to approach the problem as a tester should. To approach with an investigative eye and with the expectation that he will learn more about how the software is and isn't working.