

PROFESSIONAL TESTER

SUBSCRIBE
It's FREE
for testers

Essential for software testers

April 2012 | £4 / €5 | v2.0 | number 14

TESTING
METHODS
&
TOOLS

Including articles by:

Huw Price
Grid-Tools

Suri Chitti

Jim Holmes
Telerik

Paul Fratellone
MindTree

Ends and means

by Paul Fratellone

Improve testing using product management principles



The trouble with testers, explained by Paul Fratellone

The trouble with testers is they never understand how the product works.

Of course testers understand how the product *functions* better than anyone. But when business people make this criticism of testers, which I have heard often, by “works” they mean “makes money”. Their point is that testing tends to be driven by the testers’ view of what is important, and that view is sometimes narrow and inaccurate. And they are right.

This shortcoming harms testing. Among other bad effects, it has helped to fuel the appalling idea of deliberately releasing defective software and relying upon users to hunt for bugs in it, a travesty made more palatable by calling it twee names such as

“beta testing” and more recently “crowd testing”. Software organizations who indulge in that have abdicated their responsibility for quality and are admitting their incompetence openly.

Testing for the end-user perspective

How can real testing continue to assert its importance? Its best known argument is that it finds defects early, potentially preventing waste (wrong development, fixing, retesting) and risk (failing to fix, missed deadlines, exhausted budgets). This is valid but difficult for non-testers to understand and a hard sell to decision makers in a fast-moving business world. More importantly, testers themselves must not fall into the trap of accepting it too readily: “potentially” does not necessarily mean “actually”.

Testing must take responsibility for ensuring it adds, and is seen to add, business value to product delivery. That means learning how to drive testing from understanding of the commercial environment in which the product will operate. Good testers study, rightly, related disciplines. One of them should be product management, with a view to aligning testing more closely with the end-user experience. This is not about how easily a theoretical person can use the product, but how real people relate and react to it and the resultant real-world actions they take.

Broadening the test basis

Most testers demand comprehensive, explicitly documented requirements, although few get them. If you are one of the lucky ones, beware of tunnel vision: insisting that the only role of your testing is to compare a product with its requirements may be unrealistic. If you are not, beware of relying on your own assumptions about relative importance.

Product management (I took the certification course from Pragmatic Marketing, see <http://pragmaticmarketing.com/seminars/certification>) teaches us to ask questions to increase understanding and get more, and more relevant, information on which to base business-effective test design and execution:

What problem or problems will this upgrade, enhancement, or new feature solve? This is the *value proposition*.

For whom do we solve the problem? This is the *target market*.

How will we measure success? This is the *business result*. What metrics will be needed to validate that success has been attained?

What alternatives are out there? What is the competition doing? Is ours a *blue ocean* (low competition) or red ocean (the opposite) industry? Should we be more focused on existing customers or potential new ones?

Why are we best suited to pursue this? What is our differentiator?

Why, how, and when does it need to be delivered? This is the *market window* of opportunity.

How will we deploy this? What will be our deployment strategy?

What is the preliminary estimated cost and benefit? Will there be a return on investment, customer satisfaction increase, or cost avoidance?

Knowing *what the business wants the user to do* can help testers to keep a proactive mindset. How well a product performs in the marketplace is a good enough measure of usability that it can be used to validate usability testing, but defining good usability before that is difficult and subjective. A better understanding of the big picture is needed to help ensure that tests connect with the end user so that validation is passed: in other words, to

make testing more effective in tangible rather than speculative ways.

Narrowing the test scope

Product management thinking can also be used to make testing more efficient. Here, many of the questions it teaches us to ask are more immediately familiar to testers, because they deal with risk and robustness:

What are the most common and critical areas of the functionality from the user's and the business' point of view?

How accepting should the system be to incorrect input?

Does the product inherently rely heavily on user interface and interaction?

Are there behaviour attributes, quality dimensions and conditions that need to be assessed?

Are we gaining in market share; attracting new users and keeping our customer base?

What is an acceptable response time? Which business risks could arise from poor user experience and interface and what is their impact to goals?

What are the users saying about our products on social media sites?

Are there limitations in the software or hardware?

Testing for the end-user perspective

So a product management view can improve testing by moving its focus more towards business, rather than only technical, success. But how to persuade

product and marketing groups, who spend all their time considering these questions and trying to gather empirical evidence to answer them (often when it's too late to do anything about the results), to want to use that testing more?

The key is to communicate in numbers: to be able to quantify by how much business risk can be reduced and in what time. Many approaches to achieving that don't appeal much to testers nor those they wish to persuade because they tend to be qualitative rather than quantitative: discussing designs, mockups and animations, interviewing users, observing users (with or without complex equipment), studying user demographics and so on.

The most successful approach I have seen – I would go so far as to say it always works – is one of the simplest. Based on what is currently known about the user interface and making educated guesses about the rest, produce a detailed “user quick start guide” with step-by-step flows to guide novice users to the paths that will get the most important, according to the product management principles explained above, jobs or transactions done. Now use this as the basis for risk analysis. Consider the impact if the product fails to behave (including within a specified time) as the guide promises it will, at each of the steps, and state, as a simple time, how long it will take to perform testing that you assert (with your own reputation at stake) will prevent that. Thus we come full circle: the essential proper behaviour of the product, and the need to assure it by testing, is expressed clearly in terms of business outcomes yet is testable ■

Paul Fratellone is program director of quality and test consulting in the testing business unit of MindTree (<http://mindtree.com>)
