

Rapid Software Testing – a Review

By Kevin McKay, Senior Software Tester/Delivery Manager
January 28, 2015

Recently, I attended a three-day course on Rapid Software Testing (RST) presented by course co-author and instructor, Michael Bolton. The course was a mix of editorial, storytelling, game play and discussion on how to test software under conditions of uncertainty and time constraints. The course participants included a range of test managers, experienced testers and those interested in improving the testing processes in their organizations. I enjoyed the lessons presented and the sharing of real experiences in testing and I highly recommend attending his courses.

**“a mix of editorial,
storytelling, game play and
discussion”**

The course begins by laying down some definitions and the premises of Rapid Testing. Rapid Testing is “a mind-set and a skill-set of testing focused on how to do testing more quickly, less expensively, with excellent results.” It is a methodology for testing that can be adapted for any type of product or project. Performing testing is no different than learning anything else; you will explore and experiment; you will use your experience and involve your team; you will look for patterns and make comparisons and you will learn to tell the story of what you have learned. The testing problems

you need to solve will not begin with reading a specification or use case and then hammering out test cases. Instead, Michael Bolton will ask you to form a mind map of what you think you know and what you need to know and then organize your ideas on how you might close that gap. Over the course of that exercise, you will discover that testing, like learning, is never really “done” and that when to stop testing is a decision made by the “people that matter” based on the telling of that story.

Michael Bolton and James Bach are advocates of a context-driven approach to software testing. They will not advocate best practices for testing software quickly, because they don’t believe in them. They may make reference to a procedure that might be helpful in revealing some information in certain circumstances; however it’s up to the test team to define what works for them. What Michael Bolton and James Bach do teach is how you can build your own rapid testing framework to tell your story on the status of the product, how you tested it and “how good the testing was”.

**“testing, like learning, is
never really “done””**

If you participate in this course, don’t expect to sit back and have knowledge poured into you as this is truly an interactive learning experience.

You will be challenged by puzzles, questioned on what you think you know and on what you know you know and be surprised at the answer(s). You may even see a little magic happen!

You will work together as a group, sometimes in teams, but always as a shared experience with the whole group and the instructor. You will be encouraged to push back, to challenge, to ask questions and, at times, be challenged to think!

“encouraged to push back, to challenge, to ask questions”

Rapid Software Testing will show you how to do credible testing without relying on weak or non-existent documentation. You will see how to employ heuristics and oracles and learn that all testing, whether you consider it formal, informal, scripted or exploratory, is all exploration, learning about the application and reporting what you’ve learned to the people that matter so that they can make an informed decision.

The sessions touched on many topics, from how to define the mission of your testing, to how to

- James Bach: <http://www.satisfice.com>
- Michael Bolton: <http://www.developsense.com>
- You may also be interested in reading <http://context-driven-testing.com> (Cem Kaner)

simplify complex test activities. We talked about feedback, how to involve the decision makers in test activities, how it’s within your power to ask for testability and to own your testing methodology. Along with discussing how to negotiate when asked for estimation on test activities, we learned about session based testing methodology and using a mind mapping program to organize our test ideas, and to trace test coverage and risk.

I learned that while testing is not about numbers of test cases or a reliance on a metric that has no value or can mislead decision making, it IS an activity to be performed. It IS experimentation and discovery and informing the decision makers of what was learned.

The course structure was flexible; it allowed the freedom to move in the direction of any question asked by a participant. The resources, websites and blogs made available provided a wealth of information from many test professionals. I would recommend this course to any test professional or manager interested in learning about testing activities.

For more information on Rapid Software Testing or to reach Michael Bolton or James Bach, you can do so from the following websites:



About The Author

Kevin McKay is a Senior Software Tester and Delivery Manager with Professional Quality Assurance. He has over 15 years' experience in software testing across many domains and has worked with dozens of clients on many different projects. Kevin provides leadership, planning, and testing skills to his teams.