Exploratory Testing Essay

This essay explores the exploratory testing school in its relation to scripted testing, the exploratory testing process, and its role in agile development.

The Difference Between Exploratory Testing and Scripted Testing

In an "Exploratory Testing" blog post, Martin Fowler (2019) shares his thoughts on exploratory testing and its relation to scripted testing. Scripted testing is a practice where test cases are written down along with the expected behavior of the software. The tests can be executed separately, and the unexpected behavior of the system is considered a failure. This testing style has also been automated, which has led to minimizing human errors and faster test execution. (Fowler 2019)

However, Fowler (2019) also sheds light on the limitations of scripted testing. By using scripted testing, we can only catch known conditions. Scripted testing can be seen as a net that effectively catches bugs, but how do we ensure the net is wide enough to cover the product? Exploratory testing complements scripted testing by testing the boundaries of this net, by exploring behaviors that do not exist in the scripts (Fowler 2019). If software testing is seen as a spectrum of practices, exploratory and scripted testing can be seen as the opposite ends of that spectrum.

It might be tempting to think that exploratory testing is an unstructured practice, but it requires skill and structure. In the article "Exploratory Testing Explained", James Bach (2003) emphasizes that the structure of exploratory testing is highly situational. Bach argues that often exploratory testing can be much more effective in catching bugs than scripted testing, but it requires skills of the tester, such as listening, reading, thinking, and reporting without prescripted instructions. Tests can of course be reduced to a repeatable scripted form, and when exploratory testing is properly managed, even inexperienced testers can reveal vital information. (Bach 2003).

The exploratory testing process from a tester's viewpoint

Bach (2003) states that "Exploratory testing is simultaneous learning, test design, and test execution", where the tester actively designs and performs tests, and uses information gained during a test session. Thus, according to Bach (2003), all testing done by humans is exploratory to some degree. But what does an exploratory testing session entail?

For Bach (2003), a typical session starts with a charter that states the mission and some of the tactics that will be used. In contrast to scripted tests, the charters are often ambiguous, relying on the tester's skills and previous training. An exploratory tester is first and foremost a test designer, on the lookout for issues outside of the test scripts. This requires an ability to think critically, reporting skills, and usage of heuristics. (Bach 2003).

This process is also emphasized by Cem Kaner (2008), who argues that test-related learning, test design, execution, and result interpretation should be seen as mutually supportive activities that are run in parallel during the project. Bach (2003) also references Kaner's "tour bus" principle: have a general attack plan but allow briefly deviating from it. The tester can briefly explore areas not specified in the chart, note some things about them for later exploration, and return to the main task of the session. The learning aspect can also be seen during the test session, as tests are executed the result of the previous test impacts the next test (Bach 2021).

Jonathan Bach (2000) describes one way of structuring exploratory testing in the article "Session-based Test Management". The technique was created as an effort to track testing and separate it from testing-related activities, using sessions as the basic unit for testing. A session is charted – associated with a mission – and uninterrupted by other tasks such as e-mails. Each session is reviewable by using a session sheet report. (Bach 2000).

James Bach also anchors session-based testing as a way of measuring and managing exploratory testing so that it can be connected to ideas of project management, as metrics often are needed (Mindtree 2013).

Another viewpoint in managing exploratory testing is the dichotomy of delegation versus participation. A test lead can delegate charts to testers, or participate in sessions and direct the test session in real-time (Bach 2003).

The Role of Exploratory Testing in Agile Development

Already in 2003, James Bach stated that the main benefits of exploratory testing are rapid feedback and learning. The context-driven school of testing, which exploratory testing belongs to, developed simultaneously as the agile method as a reaction to the factory concept of software development and testing. Both methodologies focus on skill. Work is seen as a craft rather than a factory where the people in projects are interchangeable. Bach argues that the learning theory introduced by agile is that we must learn what we want as we build and modify it as we do so, leading to cycles in development. As in agile, exploratory testing works in cycles. (Mindtree 2013).

In agile contexts, scripted regression tests are typically automated for quick feedback. However, automated tests cannot ensure that the system has the expected functionality. Exploratory testing adapts better to changes in an agile environment, while still providing rapid feedback,

and as previously stated, offers us information on things we did not know about the product. Human insight is still needed to assess aspects such as usability as well, in addition to the tests that can be automated.

The test automation process should not dismiss the tester's skills either. Test automation cannot be done without exploring to some degree. This is noted by Pyhäjärvi (2017) in her blog post "The Myth of Automating without Exploring". In her experience, the creation of automated tests also exposes flaws in the test design and requires a skilled tester with an exploratory mindset to be successful (Pyhäjärvi 2017). This real-world example intersects with Bach's notion that there is no pure exploratory or scripted testing, but rather that all testing is always a mixture of both to different degrees.

In the article "What is exploratory testing" Pyhäjärvi (2019) shares an anecdote from a project where developers and automation testers did not have a broad enough definition of the expected functionality. The team could try to uncover more of the desired functionality by exploratory testing or real-world user feedback. The project lacked a tester with an exploratory mindset who could bridge the gap between our initial understanding of the product and the expectations of the product delivered. Pyhäjärvi defines modeling and learning of the real world as exploratory testing. We should not just test to learn the program but also what we should be testing. (Pyhäjärvi 2019)

The real-world findings of Pyhäjärvi are great examples of how exploratory testing can improve quality in agile teams as a complement to test automation of scripted testing. They also show that automated testing should not be dumbed down, but rather that test automation also benefits from the exploratory testing mindset, and vice versa. To quote Pyhäjärvi (2022): "You can't automate well without exploring. You can't explore well without automating".

Pyhäjärvi (2022) also defines exploratory testing as testing that bridges the gaps between other testing techniques. Exploratory testing is not just used as a name for testing strategies that do not have a name of their own, but in Pyhäjärvi's approach performance, security, and regression testing (among other types) can be used within exploratory testing as constraints and starting points (Pyhäjärvi 2022).

Pyhäjärvi's viewing point aligns with the ones of Fowler and Bach, as it does not entirely separate exploratory testing from scripted testing. Often test automation is linked to scripted and regression testing, but it can also be used to document exploratory testing, as seen in the "Contemporary Exploratory Testing" demo (Pyhäjärvi 2022).

References

Bach, James. (2003, April 16). *Exploratory testing explained* (Version 1.3) [PDF]. Satisfice. https://satisfice.us/articles/et-article.pdf

Bach, James. (2021, July 7). *Inside the mind of an exploratory tester*. Satisfice. https://www.satisfice.com/download/inside-the-mind-of-an-exploratory-tester

Bach, Jonathan. (2000, November 1). *Session-based test management*. Satisfice. https://www.satisfice.com/download/session-based-test-management

Fowler, M. (2019, November 18). *Exploratory testing*. martinFowler.com. https://martinfowler.com/bliki/ExploratoryTesting.html

Kaner, C. (2008, April). *A tutorial in exploratory testing* [PDF slides]. Florida Institute of Technology. https://13j276.p3cdn1.secureserver.net/pdfs/QAIExploring.pdf

Mindtree Ltd. (2013, May 15). *Understanding exploratory testing* [Video]. YouTube. https://www.youtube.com/watch?v=I-ItEKt_N_s

Pyhäjärvi, M. (2017, March 27). *The myth of automating without exploring*. A Seasoned Tester's Crystal Ball. https://visible-quality.blogspot.com/2017/03/the-myth-of-automatingwithout-exploring.html

Pyhäjärvi, M. (2019, July 3). *What is exploratory testing?* Medium. https://medium.com/@maaretp/what-is-exploratory-testing-620057cf75b4

Pyhäjärvi, M. (2022, July 7) *Contemporary Exploratory testing – Maaret Pyhäjärvi.* TestAutomationTalks. https://www.youtube.com/watch?v=T_67oQrPZhQ