Basics of Session-Based Test Management

This essay explores the Session-Based Test Management (SBTM) method for adding metrics and accountability to the Exploratory Testing method.

SBTM versus Exploratory Testing

Exploratory testing focuses on finding bugs fast in an ad hoc testing process, where time for documentation is minimized, and the tester has the freedom to direct the testing as they wish. While proven to be an effective way to find bugs and latent issues, the tester's freedom and the minimal documentation lead to difficulty in keeping track of each tester's progress. This requires good communication skills from each tester and a way to summarize information for management. (Bach & Bach 2000)

James and Jonathan Bach developed SBTM to target these reporting and communication issues in freeform exploratory testing while maintaining its explorative nature. The session working unit was specified to distinguish testing from testing-related activities. Each session is then guided by a charter, providing a clearer focus for the session compared to regular exploratory testing, and a session report for follow-up. The tester is free to deviate from the charter briefly, to explore possible issues and bugs, but then return to the mission stated in the charter. Other discoveries can be covered in new sessions. A session can be concluded with a debriefing session. (Bach & Bach 2000)

In addition, they ask testers to report time spent on the mission stated in the charter versus opportunity: new possible bugs, or issues. Apart from these metrics, the session sheet can be broken down into a list of bugs, a list of issues, and the tester's notes on the session. The tasks themselves, however, are reported on a more general level. (Bach & Bach 2000)

Some testers have noted that SBTM is exhaustive, but Kalman proposes that SBTM can be more of a mix and match of the key concepts and that the components can vary based on a project's need (Kalman 2007).

Key Concepts of SBTM

The key concepts can be divided into high-level management project components and reporting components that are part of the session report component. The structure of this section is based on Kalman's (2007) article, and the definitions are based on Bach's & Bach's article (2000).

Project Components

Session Report and its Components

The main elements of the Session Report are listed here. In addition, Kalman (2007) also lists: the tester, configuration, start time, and duration, which are simple concepts.

Charter: Guides the tester's focus during the session. The charter sets the main mission, but the tester is allowed to briefly investigate opportunities.

Area: Outline of investigated features and functionalities. By using the area distinction, it is possible to have some indication of test coverage.

Bugs: A simple list of found bugs, that is concerns over the product quality.

Issues: Problems and questions that relate to the test process or project itself.

Notes: The tester writes free-form notes on things that are not listed under bugs or issues, which are later used in the debriefing session

TBS Metrics: In Bach's & Bach's (2000) approach, time is managed by dividing the testing effort into three tasks for TBS metrics: 1) Test design and execution, 2) Bug investigation and reporting, and 3) Session setup.

Charter vs Opportunity: In addition, Bach & Bach (2000) ask testers to report time spent on the mission stated in the charter versus opportunity: new possible bugs, or issues.

Debrief & the PROOF-agenda

Debriefing sessions are held to keep the test manager and tester on the same page regarding testing progress and the session. These sessions are learning opportunities for new testers and are used more frequently when the tester is inexperienced. The test lead gains an understanding of the session report and accepts it. The PROOF-agenda is followed during debriefings:

Past: What happened during the session?
Results: What was achieved?
Obstacles: What got in the way of testing?

Outlook: What still needs to be done? *Feelings*: How does the tester feel about the session?

Parsing Statistics

In Bach & Bach's approach, the session sheet is provided in a tagged-text format that can be scanned, normalized, and summarized. This enables efficient reporting and predicting the duration of the testing effort. When SBTM was created at the turn of the millennium, being able to quickly report to stakeholders online was a major improvement.

Modern SBTM Practices

Djuka Selendic (2021) writes about adopting SBTM practices and elaborates on how the analyzed data can be utilized in the test lead role. TBS metrics were used to not only assess how much time was spent on testing and how much remains, but also to analyze the product areas. Selendic (2021) found that the TBS metrics helped her manage testing, as high test design and execution times and low bug investigation times for an area indicated that the area needed less testing time, allowing the team to shift their work to a more critical area, such as areas with high bug investigation times.

Selendic also found that high bug investigation times also correlated with testers not meeting their testing mission. This allowed the team to shift more time to this area as well. High Session setup times were also found to indicate a need for higher test coverage, as they indicate testability problems. As Kalman proposed, Selendic also found SBTM as a flexible system that could be adapted to suit the team. (Selendic 2021)

David Tzemach (2022) suggests a few ways to incorporate SBTM in agile projects. Each user story has its acceptance criteria and business flows to test, and each story's risk and impact are determined. Technical test flows are used as a starting point for SBTM and after the session documentation is complete, the documentation is attached to the specific user story. (Tzemach 2022)

Advancements in AI can be seen in SBTM as well. In a blog post, Petruta Paraschiv (2023) explores how ChatGPT can be used in exploratory testing. She notes that is important to remember the limitations of an LLM model, such as lack of context awareness, but that it can assist in many test-related tasks. One of the LLM use cases proposed by her includes AI brainstorming in creating the test charters, summarizing the test session, enumerating risks related to stories, and generating sample data. (Paraschiv 2023)

SBTM provides various ways to combine the management tools to better track exploratory testing. As seen in this essay, SBTM can be applied in different contexts in ways that suit the project and team.

References

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

Kalman, S. (2007). *SBT Lite: Components of session-based test management* (Version 1). Quardev, Inc.

Paraschiv, P. (2023). *How to use generative AI for exploratory testing? (With examples)*. Xray Blog. https://www.getxray.app/blog/generative-ai-exploratory-testing

Selendic, D. (2021). *A journey in test engineering leadership: Applying session-based test management*. InfoQ. https://www.infoq.com/articles/session-based-test-management/

Tzemach, D. (2022). *Session-based test management (SBTM) in agile development*. LambdaTest. https://www.lambdatest.com/blog/session-based-test-management-in-agile/