



BOF: Exploratory testing or how to improve you manual testing skills...

Thomas Ritter?

?

Manual Testing?

What to improve?



Skill-set

Domain
Language

Measure

Exploratory Testing?

"Any testing to the extent that the tester actively controls the design of the tests as those tests are performed and uses information gained while testing to design new and better tests. "

[3] James Bach

Workflow?

- 1) Charter
- 2) Timebox
- 3) Protocol (Action, Bugs, Opportunities)

**Necessary
Skills?**

- Chartering
- Reporting
- Branching/Backtracking

[5]

Example

<http://www.erp.com/salesorder/edit/1>

<http://www.erp.com/salesorder/edit/99999>

<http://www.erp.com/salesorder/edit/-1>

<http://www.erp.com/salesorder/edit/0>

<http://www.erp.com/salesorder/edit/a>

...?

How to start?

**Find missing
skills**

Retrospective

Track
progress

Heuristics

(rules of thumb)

- Test with and without anti-virus software installed
- Verify Escape cancels the dialog
- Text length (e.g., of a list entry, in a field)
- 4096 x 4096 pixel image

[6]

Alone or as a
team

Spread the word

Debugging tools. Testing tools. **Testing on the Toilet** Episode 23

Understanding Your Coverage Data

Code coverage (also called test coverage) measures which lines of source code have been executed by tests. In Episodes #20 and #21, we discussed the significance of coverage analysis, and ways to obtain coverage data successfully for your projects.

However, a common misunderstanding of code coverage data is to think:

My source code has a high percentage of code coverage, therefore, my code is well-tested.

The above statement is **FALSE!** A high percentage of code coverage is a necessary but not sufficient condition for well-tested code.

Well-tested code	—————>	High coverage
Well-tested code	<==== X	High coverage

[7]

<http://googletesting.blogspot.com/>

Learning materials

13.10.2007 (delete) (release)

SAP TechEd 2007 Munich: BOF session

Articles

- ◆ Cem Kaner has an excellent collection of articles covering a wide range of testing topics: <http://kaner.com/articles.html>
- ◆ James Bach has another interesting collection of articles: <http://www.satisfice.com/articles.shtml>
- ◆ Brian Marick's collection of essays and papers: <http://www.testing.com/writings.html>

Blogs

There are a few blogs which I followed or still follow.

- ◆ My favourite at the moment is The Braidy Tester which is published by Michael Hunter at Microsoft.
- ◆ Brian Marick

<http://tknight.org/projects/117>

References

- [1] Wind Bound Lerwick, <http://flickr.com/photos/ccgd/112068476/>
- [2] Fore!!!, http://flickr.com/photos/bengchye_loo/370012689/
- [3] James Bach, “Exploratory Testing Explained”,
<http://www.satisfice.com/articles/et-article.pdf>
- [4] Jon Bach, How to Manage and Measure Exploratory
Testing, http://www.quardev.com/articles/exploratory_testing
- [5] Jon Bach, Dynamics of Exploratory Testing,
http://www.quardev.com/articles/dynamics_of_exploratory_testing
- [6] Michael Hunter, Did I Remember To,
<http://blogs.msdn.com/micahel/articles/175571.aspx>
- [7] Google, “Testing on the Toilet”,
<http://www.flickr.com/photos/niallkennedy/330227455/>