

Topic: Usability Testing
(Previous topic: Agile)

Getting user feedback early and often is the heart of any Agile development process!

- What is the difference between usability testing and functional testing
- Types of usability testing:
 - Individual testers
 - Focus groups
- Means of getting feedback
 - Surveys
 - Interviews
 - Recordings of testers- audio and/or video
 - Instrumenting the code (for alpha and beta testing)
- What to test
 - Sketches (on a napkin even!)
 - Wireframes
 - Mock-ups
 - Prototypes
 - Real code

My experience at Intel with focus groups

- Built mock-ups and prototypes
- Wrote a test script
(Mock-ups had “canned” responses, so following a test script was essential).
- Three or four people were in each focus group. They were selected and paid by a professional agency.
- Participants were in a room with a computer and a moderator.
 - Observers were in another room watching through a one-way mirror.
 - We had video cameras recording the participants.
- A professional moderator read the script.
 - This helps avoid the problem of participants not wanting to hurt the developers’ feelings and of the developer grimacing and looking hurt in front of the participants
 - The participants didn’t touch the computer; they just watched and answered questions.
- We all watched the videos afterward and discussed the participants’ responses.

Rocket Surgery made easy approach

Based on the Book: *Rocket Surgery Made Easy, The Do-It-Yourself Guide to Finding and Fixing Usability Problems*, by Steve Krug, 2010.

Web site: <http://www.sensible.com/rsme.html>

- A morning a month, that's all we ask
It's a minimum, for Agile, at the end of every sprint.
- Start earlier than you think makes sense
 - Show participants anything from sketches to working code
 - The earlier you start, the less you'll have to change
- Recruit loosely and grade on the curve
 - Getting people who are like your "target audience" is a nice goal, but don't be too strict about it.
 - Don't use the same group of testers twice (unless all they saw were sketches)
- Make it a spectator sport (about the observers)
 - The eureka moment: the users aren't all like me!
 - Watching live is more impactful than watching a recording
 - Get the whole development team to watch – even the project sponsors (but having them in another room is less intimidating to the participants)
 - Give them instructions (what to watch for)
 - Be ready for criticism, take it as constructive and helpful
 - Provide food (snacks) ☺
- Focus ruthlessly on a small number of the most important problems.
Hold a debriefing and produce
 - A list of the most serious problems
 - A list (smaller) of what you will fix before the next test
 - More food (lunch) ☺
- When fixing problems, always do the least you can do
 - "What's the smallest, simplest change we can make that's likely to keep people from having the problem we observed?"
 - "The perfect is the enemy of the good"
(Opposite of the "good is the enemy of the best")
 - Taking something away is sometimes better than adding something.
 - Avoid redesign unless it's really necessary
 - Only re-test major changes, not tweeks