Testing

Terry Winograd

CS247 - Human-Computer Interaction Design Studio

Stanford University

Winter 2009

Different kinds of testing

- Software/device testing
  - Does it do what the specs say?
- Performance testing
  - Speed, space used,…
- Usability testing
  - Does it do what the user wants and expects?
- Market testing
  - Do people want it?
- Hypothesis testing
  - Generalizeable scientific statement

Different kinds of testing

- Software/device testing
  - Does it do what the specs say?
- Performance testing
  - Speed, space used,…
- Usability testing
  - Does it do what the user wants and expects?
- Market testing
  - Do people want it?
- Hypothesis testing
  - Generalizeable scientific statements

When do you evaluate?

- Formative Evaluation – During design and development process
- Summative Evaluation – After design is deployed

Expert Evaluation

- Usability Inspection
  - Heuristic Evaluation
  - Cognitive Walkthrough
  - Feature Inspection
  - Consistency Inspection
  - Standards Inspection

Nielsen and Mack, *Usability Inspection Methods*

Usability Testing

- Quick Evaluation (Discount usability)
  - With users
  - With experts
- Laboratory Usability Testing
- Field Studies
  - In user context
  - Web-based
    - Log analysis
    - A/B testing

Google “airport lounge” / user testing lab in Heathrow airport

Planning for a test

Scope

- What are you testing?

Purpose

- What concerns, questions, and goals is the test focusing on?

Schedule and location

- When and where will the test take place?

Participants

- How many users of what types will you recruit?

Scenarios

- What will participants do with the product in this round of testing?

Questions

- What will you ask at the beginning and end of the session?

Data to be collected

- What will you count?

Set up

- What system will you use for testing? Will you be videotaping and/or audiotaping? Will you be using a specific technology to capture data?

Roles

- Who will do what in the usability test?

Goals – What are you trying to learn from the test?

- Goals and questions should guide all evaluation studies
  - Problem spotting
  - Comparison of alternatives
  - General assessments
- What's important for this project at this time?

You won't see it if you don't look for it.

Getting ready

- Make sure you have everything you need
  - the prototype you are going to test
  - the computer set up for the participant with the monitor, resolution, and connection speed that you indicated in the test plan
  - note-taking forms on paper or set up on a computer
  - consent forms for participants to sign and a pen in case the participant does not bring one
  - questionnaires, if you are using any
  - the participant's copy of the scenarios
  - cameras, microphones, or other recording equipment if you are using any
  - folders to keep each person's paperwork in if you are using paper
- Do a dry-run and a pilot test

Artifacts – What will they be working with?

- Representations (e.g. sketches)
- Mockups at various levels of detail
- Working prototypes
  - Physical prototype
  - Interaction prototype

Before you start, run through the full test yourselves to be sure all the relevant pieces are there and working

Logistics – How do you treat the user?

- Mechanics of the test setting, enrollment, welcoming, etc.
  - Laboratory vs. informal
- Permissions
  - Privacy – Use of captured data
  - Identity – Video, Photos, etc.
  - Human Subjects approval if needed
- Quid pro quo
  - Payments, friendship, ….

Responsibility in testing

- Sometimes tests can be distressing
  - users have left in tears
- You have a responsibility to alleviate
  - make voluntary with informed consent
  - avoid pressure to participate
  - let them know they can stop at any time
  - stress that you are testing the system, not them
  - make collected data as anonymous as possible
- Often must get human subjects approval

Framing - What does the experience mean to the user?

- Why you are doing this
- Who/what is being tested
  - NOT the intelligence of the user
  - NOT their response to you
- How will data be used
  - The feeling of being watched and assessed

To minimize distortions, try to think of the situation from the user's point of view

Tasks – What is the user asked to do?

- Scripted tasks
  - Needed for incomplete prototypes
  - Valuable for cross-comparison
    - » *"Add CS160 to the list of courses and see if it conflicts with anything"*
- Open-ended tasks
  - Can be in speculative or operational mode
    - *"What would you expect this screen to let you do?"*
    - *"Try browsing some pages about …"*
- Naturalistic
  - Requires thorough prototype
    - *"Try doing your regular email"*

Script Advice

- Have a clear script for all the tasks before each test.
- Choose script tasks that cover the functionality you are interested and the questions you want the test to aswer
- Run through the script yourself before the test.
- You can revise between tests if you aren’t doing quantitative comparisons.

Capture – What can you observe?

- User actions
  - In the system (loggable)
  - Physical (observation notes and video)
- Overt comments
  - Spontaneous
  - In response to questions
    - Don’t lead: Be aware of the tester-friendly trap
- Think-aloud (individual or pair)

Use capture technology appropriately – decide what you will learn from audio or video recordings, system logs, notes, etc. and whether they are justified for this test.

Analysis – When do you do what?

- In-session notes
- Debrief and interpretive notes
- Review of capture (e.g., video)
- Formal (quantitative) analysis

Always do an interpretive debrief as soon after the session as possible, with more than one person. You won't remember as much as you think you will.

Analysis - What can you learn from the results?

- Quantitative
  - Measurable items
    - Usage, Efficiency, Subjective satisfaction …
  - External vs. internal validity
  - Statistical significance
- Qualitative
  - Identify problem areas and priorities
  - Suggest possibilities
  - Material for presentations

Some quantitative measures

- **Time on Task** -- How long does it take people to complete basic tasks? (For example, find something to buy, create a new account, and order the item.)
- **Accuracy** -- How many mistakes did people make? (And were they fatal or recoverable with the right information?)
- **Recall** -- How much does the person remember afterwards or after periods of non-use?
- **Emotional Response** -- How does the person feel about the tasks completed? (Confident? Stressed? Would the user recommend this system to a friend?)

Making it scientific

- Internal validity
  - Manipulation of independent variable is cause of change in dependent variable
    - Requires removing effects of confounding factors
    - Requires choosing a large enough sample size, so the result couldn't have happened by chance alone.
- External validity
  - Results generalize to real world situations
  - Requires that the experiment be replicable
  - No study "has" external validity by itself!

What is hard to test

- How will people really use it in the long run
  - How do users learn and adapt?
  - What is the actual utility
- What will happen with technologies with network effects
  - Until a lot of people use it it doesn't pay off
- How does your test generalize
  - Real user group may not be amenable to testing
  - High variability in tasks, users, interactions,…

When have you tested enough?

- HCI research vs. product development
  - Generalizability vs. specficity
- Resource limitations
  - Testing costs
  - Time to market
    - (assignment deadlines!)
- Diminishing returns
  - Pareto Principle

Finally

- Have Fun!