Product Hall of Fame or Shame?

Apple One Button Mouse

How to hold this?
- No tactile clue that you were holding the mouse in the correct orientation
- Later designs added a dimple in the button yet remained ergonomically difficult to use

User Testing & Automated Evaluation

Prof. James A. Landay
University of Washington
Autumn 2012

Outline
- Visual design review
- Why do user testing?
- Choosing participants
- Designing the test
- Collecting data
- Analyzing the data
- Automated evaluation

Visual Design Review
- Grid systems help us put information on the page in a logical manner
  - similar things close together
- Small changes help us see key differences (e.g., small multiples)
- RGB color space leads to bad colors
- Use color properly – not for ordering!
- Avoid clutter – remove until you can remove no more
Why do User Testing?
- Can’t tell how good UI is until?
  - people use it!
- Expert review methods are based on evaluators who?
  - may know too much
  - may not know enough (about tasks, etc.)
- Hard to predict what real users will do

Choosing Participants
- Representative of target users
  - job-specific vocab / knowledge
  - tasks
- Approximate if needed
  - system intended for doctors?
    - get medical students or nurses
  - system intended for engineers?
    - get engineering students
- Use incentives to get participants

Ethical Considerations
- Usability tests can be distressing
  - users have left in tears
- You have a responsibility to alleviate
  - make voluntary with informed consent (form)
  - 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

User Test Proposal
- A report that contains
  - objective
  - description of system being tested
  - task environment & materials
  - participants
  - methodology
  - tasks
  - test measures
- Get approved & then reuse for final report
- Seems tedious, but writing this will help “debug” your test

Selecting Tasks
- Should reflect what real tasks will be like
- Tasks from analysis & design can be used
  - may need to shorten if
    - they take too long
    - require background that test user won’t have
- Try not to train unless that will happen in real deployment
- Avoid bending tasks in direction of what your design best supports
- Don’t choose tasks that are too fragmented
  - e.g., phone-in bank test

Two Types of Data to Collect
- Process data
  - observations of what users are doing & thinking
- Bottom-line data
  - summary of what happened (time, errors, success)
  - i.e., the dependent variables
Which Type of Data to Collect?

- Focus on process data first
  - gives good overview of where problems are
- Bottom-line doesn’t tell you:
  - where to fix
  - just says: “too slow”, “too many errors”, etc.
- Hard to get reliable bottom-line results
  - need many users for statistical significance

The “Thinking Aloud” Method

- Need to know what users are thinking, not just what they are doing
- Ask users to talk while performing tasks
  - tell us what they are thinking
  - tell us what they are trying to do
  - tell us questions that arise as they work
  - tell us things they read
- Make a recording or take good notes
  - make sure you can tell what they were doing

Thinking Aloud (cont.)

- Prompt the user to keep talking
  - “tell me what you are thinking”
- Only help on things you have pre-decided
  - keep track of anything you do give help on
- Recording
  - use a digital watch/clock
  - take notes, plus if possible
    - record audio & video (or even event logs)

Using the Test Results

- Summarize the data
  - make a list of all critical incidents (CI)
    - positive & negative
  - include references back to original data
  - try to judge why each difficulty occurred
- What does data tell you?
  - UI work the way you thought it would?
    - users take approaches you expected?
    - something missing?

Using the Results (cont.)

- Update task analysis & rethink design
  - rate severity & ease of fixing CIs
  - fix both severe problems & make the easy fixes
Will thinking out loud give the right Answers?

- Not always

- If you ask a question, people will always give an answer, even if it is has nothing to do with facts
  - panty hose example

→ Try to avoid specific questions

Analyzing the Numbers

- Example: trying to get task time ≤ 30 min.
  - test gives: 20, 15, 40, 90, 10, 5
  - mean (average) = 30
  - median (middle) = 17.5
  - looks good!

- Did we achieve our goal?
- Wrong answer, not certain of anything!

- Factors contributing to our uncertainty
  - small number of test users (n = 6)
  - results are very variable (standard deviation = 32)
    - std. dev. measures dispersal from the mean

Measuring Bottom-Line Usability

- Situations in which numbers are useful
  - time requirements for task completion
  - successful task completion %
  - compare two designs on speed or # of errors

- Ease of measurement
  - time is easy to record
  - error or successful completion is harder
    - define in advance what these mean

- Do not combine with thinking-aloud. Why?
  - talking can affect speed & accuracy

Analyzing the Numbers (cont.)

- This is what statistics is for

- Crank through the procedures and you find
  - 95% certain that typical value is between 5 & 55

- Usability test data is quite variable
  - need lots to get good estimates of typical values
  - 4 times as many tests will only narrow range by 2x
    - breadth of range depends on sqrt of # of test users
    - this is when online methods become useful
      - easy to test w/ large numbers of users

<table>
<thead>
<tr>
<th>Participant #</th>
<th>Time (minutes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>20</td>
</tr>
<tr>
<td>2</td>
<td>15</td>
</tr>
<tr>
<td>3</td>
<td>40</td>
</tr>
<tr>
<td>4</td>
<td>90</td>
</tr>
<tr>
<td>5</td>
<td>10</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>number of participants</td>
<td>6</td>
</tr>
<tr>
<td>mean</td>
<td>30.0</td>
</tr>
<tr>
<td>median</td>
<td>17.5</td>
</tr>
<tr>
<td>std dev</td>
<td>31.8</td>
</tr>
<tr>
<td>standard error of the mean</td>
<td>stddev / sqrt (#samples) = 13.0</td>
</tr>
<tr>
<td>typical values will be mean ± 2*standard error</td>
<td>4 to 55</td>
</tr>
</tbody>
</table>

| what is plausible? = confidence (alpha=5%, stddev, sample size) | 25.4 = 95% confident between 5 & 55 |

Analyzing the Numbers (cont.)

- This is what statistics is for

- Crank through the procedures and you find
  - 95% certain that typical value is between 5 & 55

- Usability test data is quite variable
  - need lots to get good estimates of typical values
  - 4 times as many tests will only narrow range by 2x
    - breadth of range depends on sqrt of # of test users
    - this is when online methods become useful
      - easy to test w/ large numbers of users
Measuring User Preference

- How much users like or dislike the system
  - can ask them to rate on a scale of 1 to 10
  - or have them choose among statements
    - “best UI I’ve ever…”, “better than average”...
    - hard to be sure what data will mean
      - novelty of UI, feelings, not realistic setting ...
- If many give you low ratings → trouble
- Can get some useful data by asking
  - what they liked, disliked, where they had trouble, best part, worst part, etc.
  - redundant questions are OK

Comparing Two Alternatives

- Between groups experiment
  - two groups of test users
  - each group uses only 1 of the systems
- Within groups experiment
  - one group of test users
    - each person uses both systems
    - can’t use the same tasks or order (learning)
    - best for low-level interaction techniques

Comparing Two Alternatives

- Between groups requires many more participants than within groups
- See if differences are statistically significant
  - assumes normal distribution & same std. dev.
- Online companies can do large AB tests
  - look at resulting behavior (e.g., buy?)

Experimental Details

- Order of tasks
  - choose one simple order (simple → complex)
    - unless doing within groups experiment
- Training
  - depends on how real system will be used
- What if someone doesn’t finish
  - assign very large time & large # of errors or remove & note
- Pilot study
  - helps you fix problems with the study
    - do two, first with colleagues, then with real users

Instructions to Participants

- Describe the purpose of the evaluation
  - “I’m testing the product; I’m not testing you”
- Tell them they can quit at any time
- Demonstrate the equipment
- Explain how to think aloud
- Explain that you will not provide help
- Describe the task
  - give written instructions, one task at a time

Details (cont.)

- Keeping variability down
  - recruit test users with similar background
  - brief users to bring them to common level
  - perform the test the same way every time
    - don’t help some more than others (plan in advance)
    - make instructions clear
- Debriefing test users
  - often don’t remember, so demonstrate or show video segments
  - ask for comments on specific features
    - show them screen (online or on paper)
Reporting the Results

- Report what you did & what happened
- Images & graphs help people get it!
- Video clips can be quite convincing

Automated Analysis & Remote Testing

- Log analysis
  - infer user behavior by looking at web server logs
- A-B Testing
  - show different user segments different designs
  - requires live site (built) & customer base
  - measure outcomes (profit), but not why?
- Remote user testing
  - similar to in lab, but online (e.g., over Skype)

Web Logs Analysis Difficult

Google Analytics – Server Logs++

http://www.redflymarketing.com/blog/using-google-analytics-to-improve-conversions/
Web Allows Controlled A/B Experiments

- Example: Amazon Shopping Cart
  - Add item to cart
  - Site shows cart contents
- Idea: show recommendations based on cart items
- Arguments
  - Pro: cross-sell more items
  - Con: distract people at checkout
- Highest Paid Person’s Opinion
  “Stop the project!”
- Simple experiment was run, wildly successful

Windows Marketplace: Solitaire vs. Poker

Which image has the higher clickthrough? By how much?

A: Solitaire game
B: Poker game

A is 61% better. Why?

CSE 440: User Interface Design, Prototyping, & Evaluation
11/27/2012

The Trouble With Most Web Site Analysis Tools

Unknowns
- Who?
- What?
- Why?
- Did they find it?
- Satisfied?

CSE 440: User Interface Design, Prototyping, & Evaluation
11/27/2012

NetRaker Usability Research

See how customers accomplish real tasks on site

CSE 440: User Interface Design, Prototyping, & Evaluation
11/27/2012
Advantages of Remote Usability Testing

- Fast
  - can set up research in 3-4 hours
  - get results in 36 hours
- More accurate
  - can run with large samples (50-200 users → stat. sig.)
  - uses real people (customers) performing tasks
  - natural environment (home/work/machine)
- Easy-to-use
  - templates make setting up easy
- Can compare with competitors
  - indexed to national norms

Disadvantages of Remote Usability Testing

- Miss observational feedback
  - facial expressions
  - verbal feedback (critical incidents)
- Need to involve human participants
  - costs some amount of money (typically $20-$50/person)
- People often do not like pop-ups
  - need to be careful when using them

Summary

- User testing is important, but takes time/effort
- Early testing can be done on mock-ups (low-fi)
- Use real tasks & representative participants
- Be ethical & treat your participants well
  - want to know what people are doing & why? collect
  - process data
- Bottom line data requires ??? to get statistically reliable results
  - more participants
- Difference between between & within groups?
  - between groups: everyone participates in one condition
  - within groups: everyone participates in multiple conditions
- Automated usability
  - faster than traditional techniques
  - can involve more participants → convincing data
  - easier to do comparisons across sites
  - tradeoff with losing observational data

Next Time

Interactive Prototype Presentations