Evaluating with Users / Design Focus: Text Entry

CS 5115
Fall 2014
October 24
Ethics and the protection of ‘human subjects’

• Unethical… questionable ethics
  • [http://www.prisonexp.org/](http://www.prisonexp.org/)

• Remedies
  – Informed consent
  – Confidentiality
  – Permission to stop any time
  – Institutional Review Boards - [www.irb.umn.edu](http://www.irb.umn.edu)
    • [http://www.irb.umn.edu/basics.html](http://www.irb.umn.edu/basics.html)
Evaluating with users
The big picture

• Why are you conducting the test?
• What are you going to learn?
• What will the results be used for?
• What kind of claims do you want to make?
## Two broad approaches

<table>
<thead>
<tr>
<th>Kind of learning</th>
<th>Formative</th>
<th>Summative</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goals</td>
<td>Exploration</td>
<td>Evaluation</td>
</tr>
<tr>
<td>Type of data</td>
<td>Qualitative</td>
<td>Quantitative</td>
</tr>
<tr>
<td>Level of control</td>
<td>Less</td>
<td>More</td>
</tr>
<tr>
<td>Formality</td>
<td>Generally Less</td>
<td>Generally more</td>
</tr>
<tr>
<td>Phase</td>
<td>Design/Prototype</td>
<td>Testing</td>
</tr>
<tr>
<td>Cost</td>
<td>Often cheaper</td>
<td>Often $$$</td>
</tr>
<tr>
<td>User tasks</td>
<td>Relatively open</td>
<td>Assigned</td>
</tr>
</tbody>
</table>

Caveat: these columns are **not** carved in stone.
More formally, different methods

- Controlled (laboratory) experiments
- Field experiments
- Field studies
- Qualitative usability studies
  - Think aloud method
What are you trying to learn?

• Concrete, **quantitative** measures of usability
  – Time to learn a feature
  – Use time for specific tasks
  – Features used (or not)
  – Error rates
  – Measures of user satisfaction
  – Comparison to prior/alternative versions, competitors

• Results
What are you trying to learn?

• **Qualitative** experiences of usability
  – What will they use this thing for anyway?
  – Trouble spots in completing tasks
  – Features found / not found
  – Reactions to design elements/decisions
  – Learning users’ mental models
  – *Why* can’t those silly users do it?

• **Guidance**
Selecting users

• **Real** users
  – Variety
  – Authenticity: definitely not your group

• How many?
  – Key point: users’ time is valuable
  – Diminishing returns (especially formative)

• Recruiting?
Selecting tasks

• Tasks vs. play (exploration)
• What tasks should you choose?
  – Frequent, difficult, uncertain
• How long can tasks be?
  – Recruitment, compensation
• How scripted? (tasks vs. scenarios)
• When is the task over?
What data to collect?

• Observations, notes
  – Videotape?
• Pre-tests
  – To gauge/control for experience level
  – To establish a baseline
• Questionnaires
  – Multiple choice vs. open-ended questions
• Process data
Think aloud method

• User asked to think aloud
  – Goals
  – Confusions
  – Decisions
  – Questions (but you won’t answer!)

• Crucial
The test environment
The test environment

• “Natural” setup
• Eliminate distractions
  – (unless distractions are part of the test!)
• Space between user and experimenters
• Facilities for observers and data collection
Roles

• Facilitator
• Observer / Data Collector
• “Computer” – if testing with LoFi prototype
Facilitator

• Oversees the testing process
  – Makes sure setup is ready
  – Welcomes users
  – Conducts test activities
  – Debriefs subjects

• Better to *not* collect data
  – Let observers do this
Experimenters: very important

• Practice!
• Be consistent
• Don’t “help”
  – Very tempting, but will limit your learning
• Be aware that testing can affect you
Pilot study: very important

- Find obvious mistakes
  - Hopefully, Cognitive Walkthrough and Heuristic Evaluation helped here
- Get practice running tests
- Check that tasks are realistic
- Gain confidence in data collection
- Pilot study likely to change test plan
Running a test with one user

- Preparation
- Introduction
- Scripts
- Debriefing
Running: Preparation

• Room ready?
• Equipment ready?
• Interface in the start state?
• All experimenters present?
• Other needed materials available?
Running: Introduction

- Evaluating the interface, not the user
- No personal stake
- Confidentiality reminder – system, results
- Voluntary participation
- Welcome to ask questions
- Specific instructions
- Any questions?
Running: Scripts

• Introduce each task
  – Go over task with subject
  – Ask for questions

• Refrain from interacting with the subject
  – “Help”ing doesn’t
  – Prompt for thinking aloud, if needed
  – Watch for stuckness: wait, hint, skip, fail
Running: Debriefing

• Administer questionnaires
• Ask follow-up questions
• General discussion
• Any other comments?
• Thank them!

• Team debrief?
Interpreting the data

• Some way of prioritizing
  – Severity
  – Number of observers who record problem
  – Ease of fixing

• Summarizing the results
  – A report

• Note: Keep a design focus!
  – Or you may be analyzing for months....
General Guidelines

• Always have a pilot study
• Get professional help for big studies
• Best if you aren’t there
  – too much bias
  – subtle clues (the counting horse)
  – stay behind one-way glass
  – (but, realistically....)
Test Plan Checklist, 1

• Goal of the test?
• Specific questions you want to answer?
• Who is taking on each role?
• Who are the users going to be?
• How many users are needed?
• What kind of instruction will the users be given?
• What tasks will you ask the users to perform?
• What criteria will be used to determine the end of each task?
A Test Plan Checklist, 2

• What aids will be made available to users?
• To what extent will the facilitator be allowed to help the users?
• What data is going to be collected and how will it be analyzed?
• What is the criterion for judging the interface a success?
A Test Plan Checklist, 3

• Where and when will the evaluation be done?
• How long will the evaluation take?
• What computer support?
• What software?
• Initial state of the system?
• Are there any system/network load requirements?
Exercise

• Things you want to find out when you test with users
• General plan for your test
• Tasks?
• Data you will collect
• Any concerns/issues?
Design Focus
Design Focus

• Text entry on small devices
• Major challenge: limited space for a “keyboard”
Old school: 9 Keys

From Computer Desktop Encyclopedia
© 2001 The Computer Language Co. Inc.
Simplest text input method: Multi-tap

- Entering ‘the’ requires typing 8-4-4-3-3
Smarter/easier method: T9

[T9] combines the groups of letters on each phone key with a fast-access dictionary of words. It looks up in the dictionary all words corresponding to the sequence of keypresses and orders them by frequency of use. As T9 gains familiarity with the words and phrases the user commonly uses, it speeds up the process by offering the most frequently used words first and then lets the user access other choices with one or more presses of a predefined Next key.

- Entering ‘the’ requires typing 8-4-3
Now is the time for all good men to come to the aid of their country.
T9 in action
Design factors?
Design factors

• Keyboard layout / organization
• Entering letters / words
• Visual feedback
• Prediction
• Personalization
• Learning
Touchscreen keyboards
Touchscreen keyboards
Basic text input methods

• Letter at a time
• Gesture typing / shape drawing
Swype – Origin: Shape drawing

- http://www.swype.com/
- https://www.youtube.com/watch?v=SJR-AefCG_c
Swiftkey – Origin: Prediction

Saved 50 billion keystrokes

50 billion keystrokes is:

- 350 million tweets (if you use all 140 characters)
- 634 year of typing (if you type at 30 wpm – pretty fast for a smartphone)
- More than halfway to the moon (if you print out every character in 12 point font)

Hey guys, shall we go to the pub later?
Let's meet at my place.

There are 62 characters in the message, but to type it with SwiftKey only required 30 taps. That means SwiftKey has saved you 32 keystrokes, and you are 53% more efficient. You can see your personal stats by tapping Settings > SwiftKey stats, and you can find out more about what your SwiftKey stats mean here.
SwiftKey helps you type when walking (or drunk)

“SwiftKey knows me better than I know myself.”

I use SwiftKey all the time because I am a multi-tasker and can text with one hand and half an eye and still walk down the streets of NYC and not fall down a manhole. Where it really helps out is when I have had too much to drink. I can’t type when I am drunk, no one can! SwiftKey fixes that for me. I set the keyboard to rapid style and the intuitive predictions do their job. All I need to do is be in the vicinity of the letter I am trying to hit and it reads my mind. When my wife asks me if I am still at the bar… 2 hours after I said I would be home, What would look like this “uk om mh wsy hpmw nq” is corrected to “I’m on my way home now”.

SwiftKey has replaced Nanuk, my dog, as my new best friend. SwiftKey knows me better than I know myself, it is like a soulmate who completes my sentences and can find words for me when I lose them. Sometimes it even suggests better ones!

Jeff
New York, USA
• http://wordability.net/2012/11/01/the-swiftkey-way-to-learning-new-words/
Recent stock Android keyboard has both “gesture typing” and prediction