

Methods to measure usability of secure/private systems

Mainack Mondal

CS 60081
Autumn 2021



Doubt clearing session

Please read the papers
in the website

Roadmap

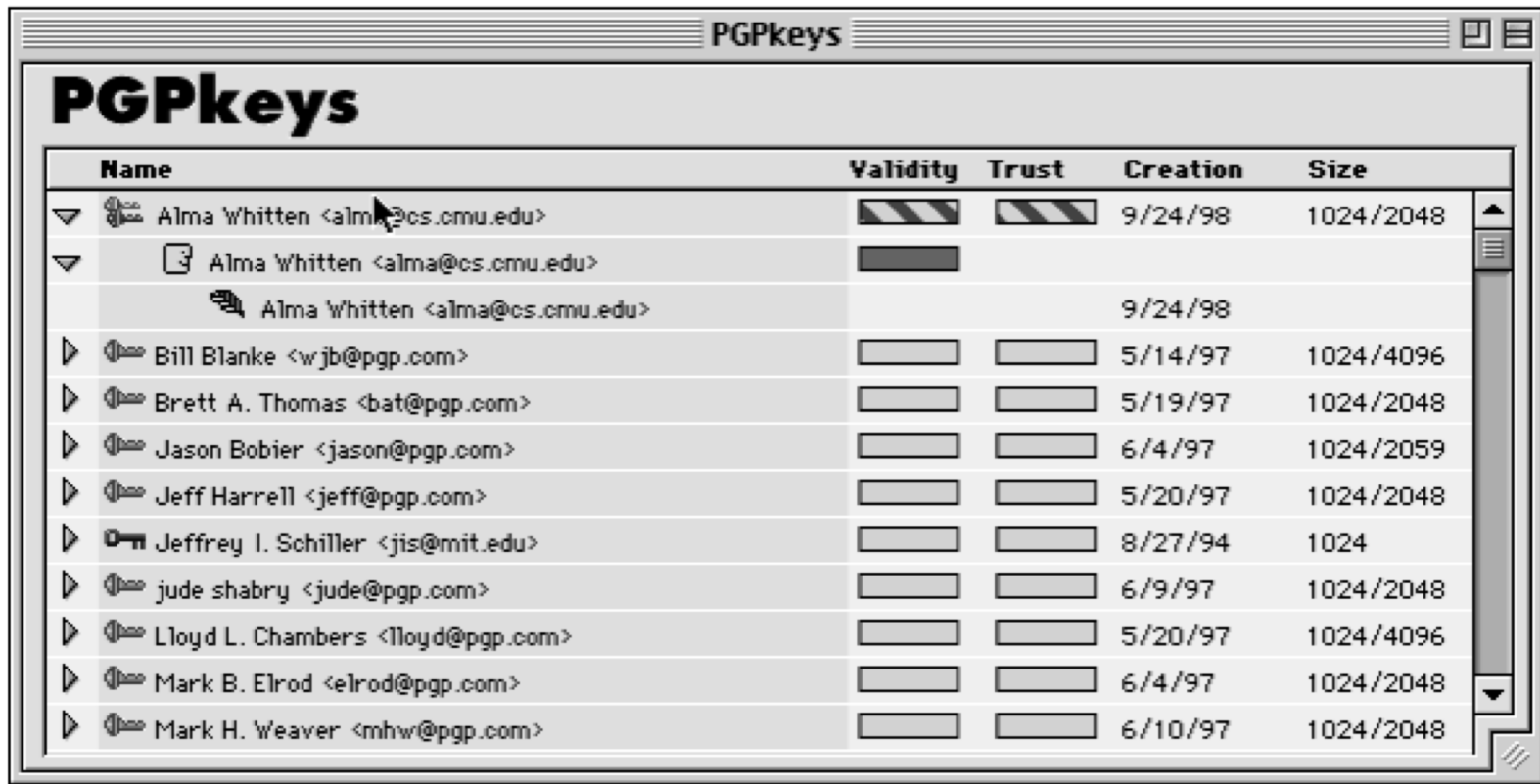
- Example of a classic usable security study
- General methods/workflow for assessing usability in secure/private systems

Material is often based on lectures from: Lorrie Cranor, Blase Ur, Kami Vaniea, Michelle Mazurek, Elissa Redmilles, <https://www.usabilitybok.org/>

Why Johnny can't encrypt

- Classic paper in usable security (1999)
- A bit of history ...

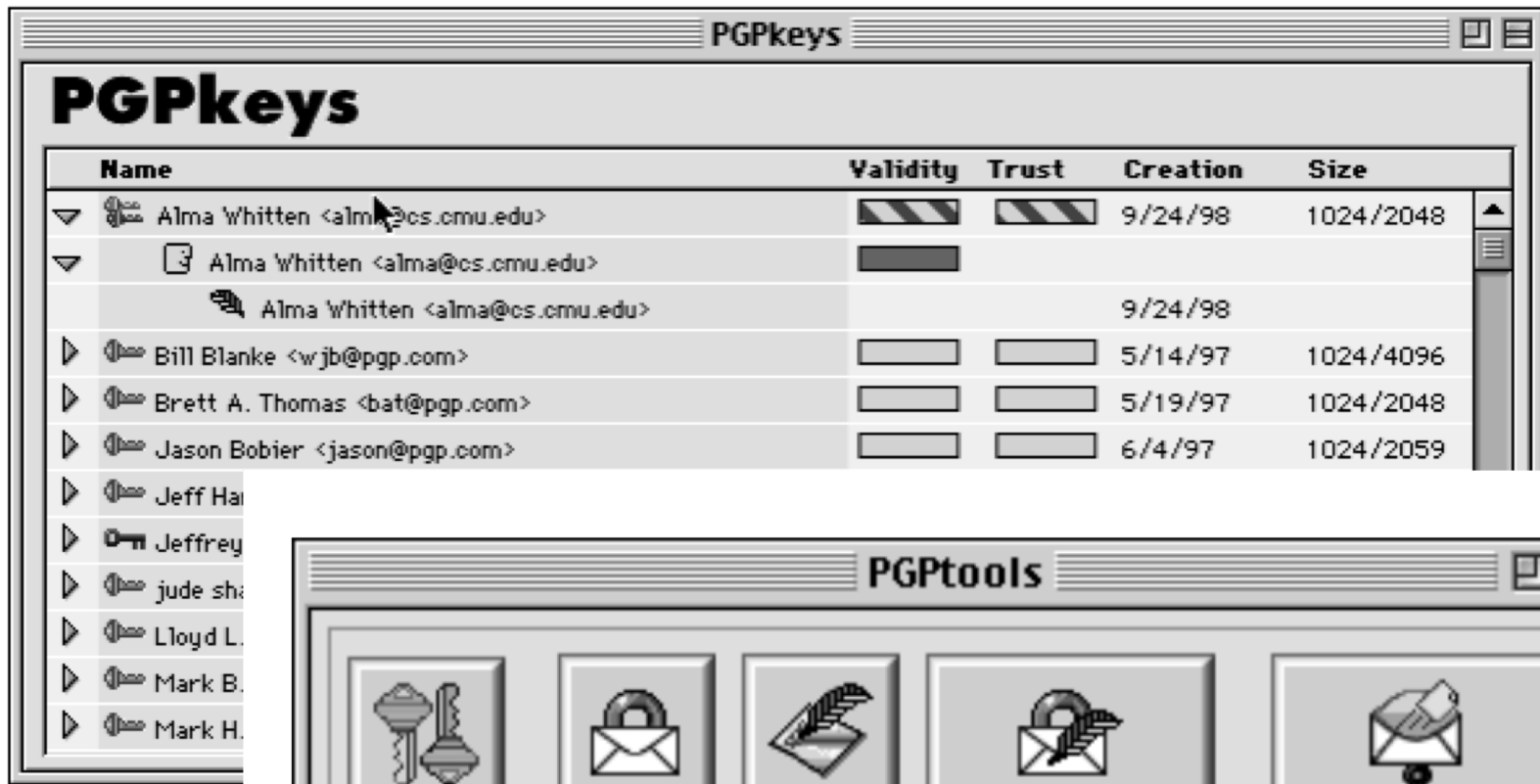
Why Johnny can't encrypt



The screenshot shows a window titled "PGPkeys" with a list of public keys. The table has five columns: Name, Validity, Trust, Creation, and Size. The first two entries for Alma Whitten show a "Validity" of "Unknown" (diagonal lines) and "Trust" of "Unknown" (diagonal lines). The other entries show "Validity" as "Valid" (solid black) and "Trust" as "Never" (empty box).

Name	Validity	Trust	Creation	Size
Alma Whitten <alma@cs.cmu.edu>	Unknown	Unknown	9/24/98	1024/2048
Alma Whitten <alma@cs.cmu.edu>	Unknown	Unknown		
Alma Whitten <alma@cs.cmu.edu>			9/24/98	
Bill Blanke <wjb@pgp.com>	Valid	Never	5/14/97	1024/4096
Brett A. Thomas <bat@pgp.com>	Valid	Never	5/19/97	1024/2048
Jason Bobier <jason@pgp.com>	Valid	Never	6/4/97	1024/2059
Jeff Harrell <jeff@pgp.com>	Valid	Never	5/20/97	1024/2048
Jeffrey I. Schiller <jis@mit.edu>	Valid	Never	8/27/94	1024
jude shabry <jude@pgp.com>	Valid	Never	6/9/97	1024/2048
Lloyd L. Chambers <lloyd@pgp.com>	Valid	Never	5/20/97	1024/4096
Mark B. Elrod <elrod@pgp.com>	Valid	Never	6/4/97	1024/2048
Mark H. Weaver <mhw@pgp.com>	Valid	Never	6/10/97	1024/2048

Why Johnny can't encrypt



Why Johnny can't encrypt

- Classic paper in usable security (1999)
- A bit of history ...
- So, why can't Johnny encrypt?
 - Why was it so hard for the users?
 - How did the experiments motivate the tasks?

Why Johnny can't encrypt

- Classic paper in usable security (1999)
- A bit of history ...
- So, why can't Johnny encrypt?
 - Why was it so hard for the users?
 - How did the experiments motivate the tasks?
- Findings
 - Interfaces are “bad”, metaphors are confusing, non-transparent, key management is difficult

Define usable secure software

- Security software is usable if the people who are expected to use it:
 - Are reliably made aware of the security tasks they need to perform
 - Are able to figure out how to successfully perform those tasks
 - Don't make dangerous errors
 - Are sufficiently comfortable with the interface to continue using it

Question

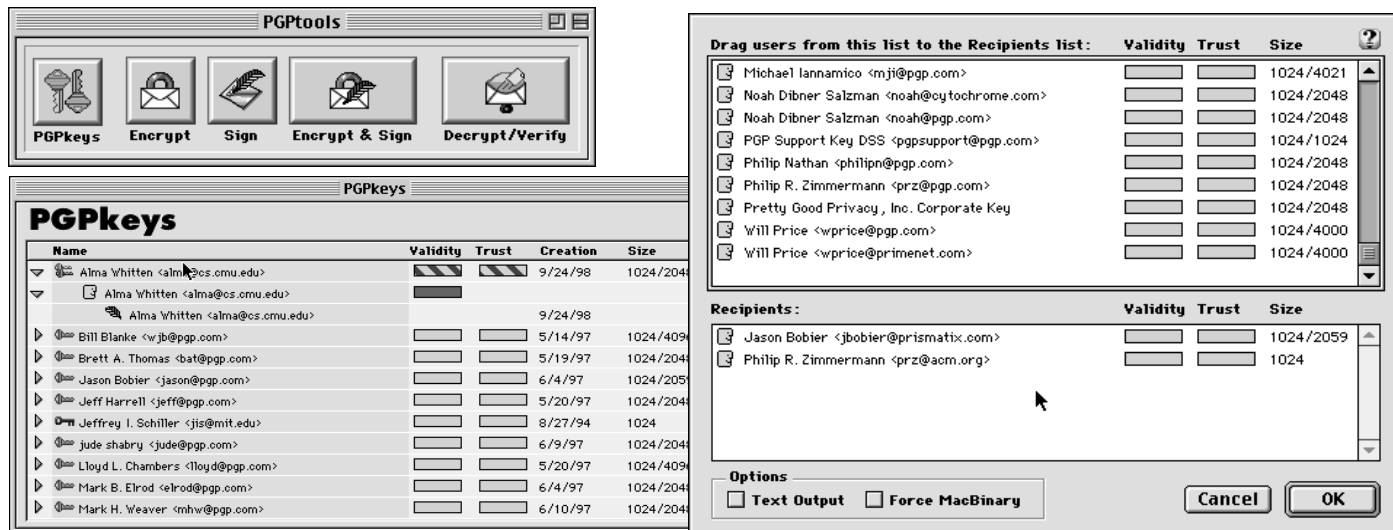
*If an average user of email feels the need for privacy and authentication, and acquires PGP with that purpose in mind, will PGP's current design allow that person to **realize what** needs to be done, figure out **how** to do it, and **avoid** dangerous errors, **without becoming so frustrated** that he or she decides to give up on using PGP after all?*

Security evaluation of PGP 5.0

- PGP 5.0
 - Pretty Good Privacy
 - Software for encrypting and signing data
 - Plug-in provides “easy” use with email clients
 - Modern GUI, well designed by most standards at that time

Usability Evaluation Methods

- Cognitive walk through
 - Mentally step through the software as if we were a new user.
 - Focus on interface learnability

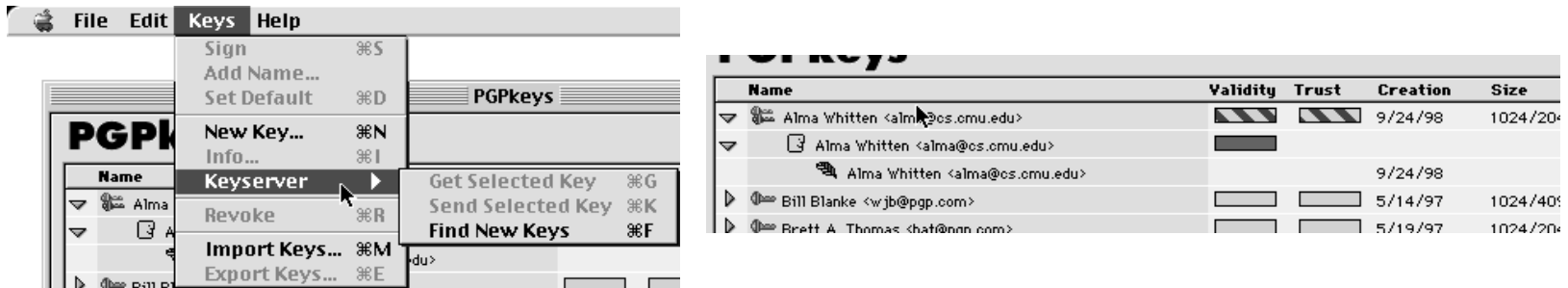


Cognitive Walk Through Results



- **Visual metaphors:** What does the pen mean?
- **Confusion with keys**
 - Different keys
 - What is **key** server? (non-transparent)
 - What is **key** management? (non-transparent)

Cognitive Walk Through Results (contd)



- **Irreversible actions:** Need to prevent costly errors
- **Consistency:** Status message was “Encoding” (and icon message was ...)
- **Too much information**
 - More unneeded confusion
 - Suggestion: show the basic information, make more advanced information available only when needed.

Lab usability study with users

- User Test
 - PGP 5.0 with Eudora
 - 12 participants: at least attended college; none with advanced knowledge of encryption
 - Participants were given a scenario with tasks to complete within 90 min
 - Tasks built on each other
 - Participants could ask some questions through email

Lab study results

- 3 users accidentally sent the message in clear text
- 7 users used their public key to encrypt and only 2 of the 7 figured out how to correct the problem
- Only 2 users were able to decrypt without problems
- Only 1 user figured out how to deal with RSA keys correctly.
- A total of 3 users were able to successfully complete the basic process of sending and receiving encrypted emails.
- One user was not able to encrypt at all

Lab study results: Summary

- Interfaces are bad
- Metaphors are wrong (and confusing)
- Opaque process
- Key management is difficult

Roadmap

- Example of a classic usable security study
- General methods/workflow for assessing usability in secure/private systems

Planning a research study

1. Define your research question
2. Identify your variables
3. Pick one/multiple study methods
4. Run your study
5. Evaluate the outcome

1. Defining research questions

Bad research questions

Is [a password manager] usable?

Would [you] fall for a phishing attack?

Is [Institute] fun?

Is [person name] knowledgeable in computer security?

1. Defining research questions

Bad research questions

Is [a password manager] usable?

**Not measurable/
testable**

Would [you] fall for a phishing attack?

**Need to be
more specific**

Is [Institute] fun?

Is [person name] knowledgeable in computer security?

What is a good research question?

- Specific topic (not how you feel about your privacy)
- Theoretical/practical significance (you should likely to uncover something novel)
- Viable / answerable
- Concrete Ability to know when answered (should have a concrete final anticipated answer)

How to create good research questions : rule of thumb

- Relationship between at least two variables
- Testable, falsifiable
- Variables are clearly defined
- Relationship / how you measure it is clearly defined
- Should be *interesting*

Example research questions

Is [a password manager] usable?	Can users use [new password manager] faster and with less errors than [old password manager]?
Would [you] fall for a phishing attack?	Would [a user] click on a malicious url communicated via email from a known sender? Would [set of factors] will increase/decrease the clicking rate?
Is [Institute name] fun?	Do students of [Institute A] spend equal time in [academic activity] and [non academic activity] than [Institute B]?
Is [person name] knowledgeable in computer security?	Can [person A] write down correct definition of [set of security definition] faster and with less errors without consulting external resources compared to [person B]?

Revisit: usable encryption

- Define what is “usable”
 - Do task within X minutes, #errors, remember the steps etc.
- Identify what your users need to be able to do using your system
- The goals need to be specific and easy to identify if they have or have not been completed
- Example: Digitally sign an email
- Bad example: Show me how to use the sign functionality

1. Define your research question
2. Identify your variables
3. Pick one/multiple study methods
4. Run your study
5. Evaluate the outcome

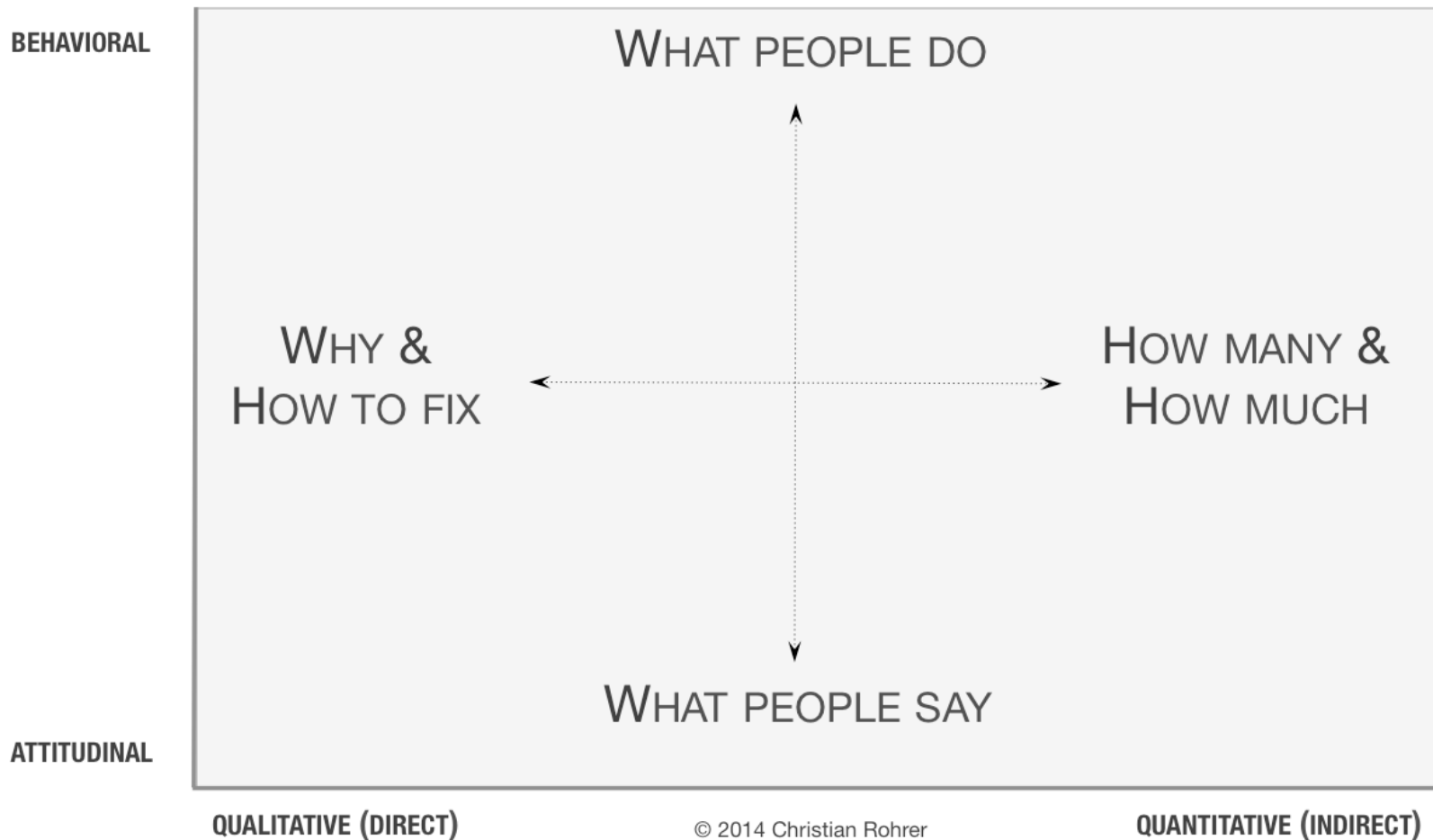
Variable measure ...

- Variables are a proxy for measuring the concepts you want to know
 - These proxies are called “constructs”
- You can measure:
 - Facts: characteristics, frequency of behaviors
 - Attitudes, preferences

2. Identify your variables

- **Attitudinal:** User attitudes and opinions
- **Behavioral:** What the user actually does or is capable of doing
- **Qualitative:** Unstructured data. Typically unstructured language data
- **Quantitative:** Structured data. Typically numerical data that can be summed or counted or do mathematical analysis on

QUESTIONS ANSWERED BY RESEARCH METHODS ACROSS THE LANDSCAPE



<https://www.nngroup.com/articles/which-ux-research-methods/>

More on quantitative variables

- Two types of measurements (variables): dependent and independent
- Dependent / outcome variable
 - “Dependent” on the study
 - Measures the usability goal
- Independent variable
 - Anything you are directly manipulating
 - An element of the study which is under your control
 - A pre-existing feature of your participant

Example

- Research question: Can users use [new password manager] faster and with less errors than [old password manager]?
- Dependent / outcome variable
 - Time spent to create a password
 - #errors while inputting the password
- Independent variables
 - Study group (which interface shown, old or new)
 - If the password was meant to be used for bank or Facebook
 - Order of the tasks
 - Time of day
 - Demographics of participants

Common dependent things to measure

- #dangerous errors made
- Time spent in errors
- Time to complete task
- Percent of task completed per unit of time
- Ratio of successes to failures
- Percent or number of errors
- Frequency of help and documentation use

1. Define your research question
2. Identify your variables
3. Pick one/multiple study methods
4. Run your study
5. Evaluate the outcome

Study methods

- | | |
|---|---|
| <ul style="list-style-type: none">• A/B Testing• Affinity Diagramming• Card Sorting• Case Studies• Cognitive Walkthrough• Competitive Testing• Critical Incident Technique• Customer Experience Audit• Desirability Testing• Diary Studies• Ergonomic Analysis• Experience Sampling• Experiments• Eye tracking | <ul style="list-style-type: none">• Fly-on-the-wall Observation• Focus Groups• Graffiti Walls• Heuristic Evaluation• Interviews• KJ Technique• Lab studies• Observation• Participatory Design• Surveys• Think aloud |
|---|---|

1. Lab studies

- Concept: Have a participant come to a lab and interact with the interface there
- You setup the lab so that it mimics the situation you need to have (e.g., set up a really badly designed password manager)
- Pros:
 - Full control over the environment
 - Detailed data, you can ask “why”
- Cons:
 - Small sample size
 - Being in the lab changes user (they can feel safer or stressed)

2. Think aloud studies

- Concept: A participant uses the interface and speak while doing it
- Can be very versatile, long or short, detailed or minimal, planned or ad-hoc
- Pros:
 - Learn what the user is trying to do and why they do some things
 - Detailed data,
 - Requires small sample
- Cons:
 - Talking aloud might change how the user will think about the task
 - You CANNOT measure timing

Requirements of think aloud

- You need to know more than your participant
 - What the participant must do?
 - What errors are dangerous?
- Pre-planning
 - Make sure tasks are interesting to researcher
 - Know what you want to take notes on (or you can record)
- Precise
 - Don't bias the user, know exactly what you will say
 - Give tasks they can perform
- Analysis
 - Measure # and types of errors. What caused them?

3. Field studies

- Concept: Similar to think aloud, but observe in real world
- Observation may be either direct, where the investigator is actually present during the task, or indirect, where the task is viewed by some other means like a video recorder set up in an office
- Pros:
 - Greater “ecological validity” than think aloud
- Cons:
 - Limited scope
 - Active questions can again change user behavior

3. Field studies

- Concept: Similar to think aloud, but observe in real world
- Observation may be either direct, where the investigator is actually present during the task, or indirect, where the task is viewed by some other means like a video recorder set up in an office
- Pros:
 - Greater “ecological validity” than think aloud
- Cons:
 - Limited scope
 - Active questions can again change user behavior

Contextual enquiry is a between think aloud and field studies

4. Diary studies

- Concept: Participants keep track of activities or events in some form of diary or log for a particular period of time.
- Track specific items like mobile device usage or general activities (what you did for each day from 1 pm to 2pm)
- Pros:
 - Information about the user's experience over time.
 - Less lag between feedback
- Cons:
 - All are self-reported
 - Users forget

5. Interviews

- Concept: You ask participants questions (with or without aid of data or interface)
- Can be structured (exact questions), or semi structured (let the user speak after you ask initiation question to set context)
- Pros:
 - Detailed data,
 - Good for exploration (identify themes, gains new perspectives)
- Cons:
 - Usually don't generalize
 - Potential for extra bias from the interviewer

Summary: broad types of studies

- What people want
 - Contextual inquiry
 - Interviews
 - Focus groups (discussion with a group of people with moderator)
 - Surveys (will come next)
 - Diary study (prompt people)
- What/how users think
 - Interviews
 - surveys

Summary: broad types of studies

- Expert evaluation of usability
 - Cognitive walkthrough
- Usability test
 - Laboratory (“think aloud”)
 - Survey
 - Log analysis
- Controlled experiment to test causation
 - A/B testing
- Varying the independent variables
 - Full factorial?

Summary: broad types of studies

- What people want
 - Contextual inquiry
 - Interviews
 - Focus groups (discussion with a group of people with moderator)
 - Surveys (will come next)
 - Diary study (prompt people)
- What/how users think
 - Interviews
 - surveys

6. Survey and question creation

Questions to ask

Biases to avoid

Pre-testing / piloting

6. Surveys

- Ask participants to answer a set of pre-defined questions.
- Pros:
 - gather data from a large number of people quickly
 - can determine how prevalent an issue or concern is □ close-ended questions are easy to analyze
- Cons:
 - can only gather data you know about
 - careful planning is required before running a questionnaire
 - open-ended questions can take a lot of time to analyze and require careful setup

Why surveys?

- Understanding people
 - Understand the target population, mental models
- Testing a theory
 - Do people think that $A=B$?
- Testing a prototype design
 - How do people interpret functions of my interface?
- Testing the final design
 - How are people actually using my tool?
 - What do people think after they use it?

Common questions

- **Attitudes**
 - Are you comfortable using X?, Would using X work?
- **Behaviors**
 - How often do you use X?, Do you regularly do X?
- **Knowledge**
 - What is the best definition of X?
- **Expectations**
 - If the webpage did X what would you expect to happen?
- **Capabilities**
 - Can you write a “hello world” code?

Common parts of a survey

- Single and multiple choice checkboxes
- Matching
 - Rank the following from 1 to 5
- Rating scales
 - Likert Scales (3, 5, 7 points scales, agree or disagree)
- Semantic scales (“very comfortable” to “not comfortable at all”)
- Open ended responses

Open ended vs. close ended

Open ended

Where does this URL go? What does it do?

Easier to write, harder to analyze

Close-ended

If you clicked on the link above, what web page would open?

- IIT Kgp's main page
- Amazon's main page
- SBI's main page
- I will be taken to one of the sites above, but not their main page
- I will be taken to a website not listed above
- Other _____

Harder to write, easy to analyze

Types of questions

Fill in the blank	What is your age _____																		
Typical MCQ	<p>What is the highest level of education you have achieved?</p> <p><input type="radio"/> High school or less <input type="radio"/> Some college <input type="radio"/> Bachelors degree</p> <p><input type="radio"/> Masters degree <input type="radio"/> Doctoral degree</p>																		
Scale where multiple questions are meant to be summed together	<p>To what extent do you agree or disagree with the following statement (select one answer per row)</p> <table border="1" data-bbox="542 743 1792 982"> <thead> <tr> <th></th> <th>Strongly Disagree</th> <th>Disagree</th> <th>Neither Agree or Disagree</th> <th>Agree</th> <th>Strongly Agree</th> </tr> </thead> <tbody> <tr> <td>I often ask others for help with the computer</td> <td><input type="radio"/></td> <td><input type="radio"/></td> <td><input type="radio"/></td> <td><input type="radio"/></td> <td><input type="radio"/></td> </tr> <tr> <td>Others often ask me for help with the computer</td> <td><input type="radio"/></td> <td><input type="radio"/></td> <td><input type="radio"/></td> <td><input type="radio"/></td> <td><input type="radio"/></td> </tr> </tbody> </table>		Strongly Disagree	Disagree	Neither Agree or Disagree	Agree	Strongly Agree	I often ask others for help with the computer	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Others often ask me for help with the computer	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
	Strongly Disagree	Disagree	Neither Agree or Disagree	Agree	Strongly Agree														
I often ask others for help with the computer	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>														
Others often ask me for help with the computer	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>														
Likert scale question using a pre-defined anchor (skill)	<p>In terms of internet skills, do you consider yourself to be:</p> <p><input type="radio"/> Not at all skilled <input type="radio"/> Not very skilled <input type="radio"/> Fairly skilled</p> <p><input type="radio"/> very skilled <input type="radio"/> Expert</p>																		

Question design principles

- Wording matters
 - “usually” can be interpreted in 24 different ways
 - Variation gives incomparable data (I understand security as protecting my computer, for you it is protecting your password)
 - For Usable Security and privacy its worse (domain specific, technical language)
 - Respondents WILL ignore your instructions

A possible method

- Try to come up with higher level section guided by RQs
 - Can be steps of software use
 - Can be specific actions/events you have in mind
- Then for each section create high level things you want to know (as much as you can)
 - Two or more people working together helps
 - The more you know the better you are
- Create constructs for your questions
 - Might not be very good, so need iteration

6. Survey and question creation

Questions to ask

Biases to avoid

Pre-testing / piloting

Using Likert scales in the right way

- Used to assess nuanced feelings (e.g., agreement)
 - Good scales are between 4 and 10 points
 - Even scales (4, 6, 8 options) elicit stronger responses – no neutral option
 - Scales should be balanced

Level of Acceptability

- 1 – Totally unacceptable
- 2 – Unacceptable
- 3 – Slightly unacceptable
- 4 – Neutral
- 5 – Slightly acceptable
- 6 – Acceptable
- 7 – Perfectly Acceptable

Avoid double barreled questions

- Do you believe that you should update your phone number to Google and change your password every three months?
 - Yes
 - No

Avoid double barreled questions

- Do you believe that you should update your phone number to Google and change your password every three months?
 - Yes
 - No
- Two/more questions: one answer
 - You end up not collecting answers to either

Avoid desirability bias

- Can be priming or leading
 - Question statement might force users likely give a specific answer
 - Question statement hints at a correct answer
- **Asking questions so that they think have a correct or a societally correct answer**
- Solution: soften the wording
 - People take many rules to create their password for ease of use, which of the following most closely matches a rule that you used or you know others use

Avoid order bias

- Ordering of questions change responses
 - Online survey: people pick top choice
 - On phone: they pick last choice
- Randomize questions and answers