

User testing

Introduction, Observational Techniques

Introduction

User testing: What is it?

- Let a group of users perform certain tasks with the system.
- Record what happens
- Analyze the results

User testing: Why do it?

- Not all users are like you
- It is difficult to pretend you are a user
- It is difficult to forget experience
- In particular if you are the designer of the system...
- It makes it easier to convince the designers

User testing: How do you do it?

- Determine what you want to learn
- Design the test
- Implement the test & Select users
- Test the test (= Pilot test)
- Perform the test
- Analyze the data
- Write a report

What do you want to learn? (1)

Discuss in pairs what questions you may want to have answered by a user test.

For instance:

- How fast can users perform a certain task with this system?

What do you want to learn? (2)

- How fast can users perform certain tasks?
 - When they see the system for the first time
 - When they are experienced users
- How many mistakes do they make?
- How do they like it?
- How do two systems / designs compare?

This is called Summative evaluation (judgmental)

Examples

- Compare two systems:
 - Is Virgin's on-line shopping side more usable than Amazon's?
 - Is the new system more usable than the old one?
- Compare two designs:
 - Is my system more usable with drop-down menus or with lists?

What do you want to learn? (3)

- What slows users down and why?
- What confuses users and why?
- Why do users make mistakes?
- What do users like and dislike and why?
- Why is one design better than another?

This is called Formative evaluation (improvement)

Design the test

- Identify users
- Determine experimental design
- Determine tasks
- Determine hardware and software

=> Experimental design lecture

Implement the test

- Prepare consent forms, information sheet;
Get ethics approval
- Prepare hardware and room
- Prepare software (including logging)
- Prepare instruction material for users
- Prepare instruction material for test leaders
- Prepare tasks
- Prepare questionnaires (if needed)

Select users

- Optionally: use questionnaire to determine suitability
- Make a schedule
- Recruit users, using information sheet

Test the test (Running a pilot)

- Are instructional materials clear?
- Does the test leader know what to do?
- Does the experiment not take too much time? (≤ 1 hour?)
- Does the software and hardware work correctly?

What is wrong with this?

A videotaped test starts as follows:

Test leader: “Welcome to this test. In this test, I will give you three tasks to perform with our prototype system. We will measure how fast you perform the tasks and how many errors you make. Here are the tasks (Hands the user a piece of paper). Start as soon as you have read the first one.”

Perform the test

- Prepare user for test and obtain consent
- Run user through tasks and collect data
One task at a time! And do not help!
- Debrief the user

=> Observational techniques

=> Questionnaire / Interview

Analyze data

- Summarize performance data (such as task completion times and error rates):
 - mean
 - standard deviation
- Summarize ‘opinion’ data
- Perform statistical tests

=> Basic statistics lecture

Write report

- Experimental design
- Participants
- Description of system evaluated and tasks
- Results
- Discussion
- Appendix with material used

Others should be able to repeat your test

Where does the test take place?

- Real work setting
- Usability lab

Usability lab

- Dedicated room for doing usability testing
- Quiet, no distraction
- Includes computer and observational equipment such as video cameras, microphone
- Sometimes a one-way mirror
- Sometimes decorated to convey a certain atmosphere

Examples of a usability lab



Prepare room

- Make sure (mobile) phones and tannoy are switched off
- Make sure that nobody will interrupt the experiment: put a notice on the door
- Use a sound-proof room
- Blind windows

Observational techniques

Gathering quantitative data

- Time to complete each task
- Number of errors
- Buttons pressed
- Sensor data
- Eye-tracking data

Sensor data



Heart-rate
(stress, emotion - arousal)



Galvanic skin response



EEG brain activity
(engagement, attention, effort?)



Posture, leaning back, forwards
(engagement)



Eye-tracking



Card, et al, 2001

Gathering qualitative data

- While the user is working with the system make notes about user confusion, hesitation, mistakes
- Log the circumstances (which task, what moment)
- Log the time so that you can compare it with video

- In pairs, discuss what information you are missing

Thinking aloud: what is it?

- Have test subjects use the system while continuously thinking out loud.
- Verbalizations allow researcher to understand how users are interpreting the interface

Thinking aloud: how do you do it?

Ask users to tell you

- what they are trying to do
- things they read
- questions that arise in their mind
- things they find confusing
- decisions they make

Preparing the user

- Demonstrate thinking aloud for everyday task, e.g. looking up word in dictionary.
- Show user video clip of a previous thinking aloud test.
- Have user practice technique

Example

- Task:
Look up the word “usability” in the dictionary.

During the test

- Encourage user
 - non-committal ``uh huh"
 - ``Tell me what you are thinking"
 - ``Keep talking"
- Don't ask specific questions:
 - ``What do you think those prompts mean?"
 - ``Why did you do that?"

After the test

- Carefully analyze the video data
- Use multiple analyzers
- Use categories decided on beforehand

Strengths of thinking aloud

- Large amount of qualitative data with small number of subjects
- Finds many usability problems
- Finds why they occur
- Requires little expertise of test leader
- Generates colourful quotes
- Generates user ‘vocabulary’

Weaknesses of thinking aloud

- Unnatural, users may find it difficult to do
- Requires user training and prompting
- Can interfere with cognitive processes and influence user's behaviour
- User tend to justify their errors
- Cannot provide quantitative (performance) data
- Time consuming to analyze data

Co-discovery

- Put two users who know each other well in front of the system
- Let them solve the tasks together
- Record their conversation

Advantages of co-discovery

- Natural
- Reflects workplace: most people have someone else to help them
- Interaction between subjects can bring out more insights
- No training needed, no prompting needed
- Subjects more at ease

Retrospective testing

- User works with the system and actions are recorded on video
- After finishing the tasks, the subject views the video and provides comments on what is happening.

Coaching method

- The user is provided with a coach (an expert user) who can answer questions
- Experimenter records what problems arise and what information is necessary to give