Evaluating interfaces with the users – Qualitative Methods

Human Computer Interaction

Based on slide deck
Part 3: Designing with the user. Evaluating interfaces with the users - Qualitative Methods
Human Computer Interaction I: Principles and Design
by
Saul Greenberg
Professor
University of Calgary, Canada
The new slides are marked with a *

Evaluating interfaces with the users

Why evaluation is crucial
Quickly debug prototypes by observing people use them
Methods that reveal what a person is thinking about
Ethics

Why bother?

Iterative design

- – does system behavior match the user's task requirements?
- – are there specific problems with the design?
- – what solutions work?

Acceptance testing

- – verify that system meets expected user performance criteria
  - • 80% of 1st time customers will take 1-3 minutes to withdraw $50 from the automatic teller

design

evaluation   implementation
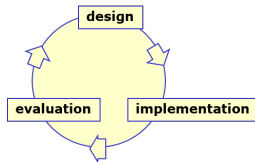
*Evaluation

Naturalistic approach

Experimental approach
- ▶ usability engineering
  - ▶ usability inspection methods
    - ▶ qualitative methods
    - ▶ quantitative methods
  - ▶ usability testing methods
    - ▶ qualitative methods
    - ▶ quantitative methods

Naturalistic approach

Observation occurs in realistic setting

- – real life

Problems

- – hard to arrange and do
- – time consuming
- – may not generalize

Experimental approach

Experimenter controls all environmental factors

- – study relations by manipulating *independent* variables
- – observe effect on one or more *dependent* variables
- – Nothing else changes

*There is no difference in user performance (**time** and **error rate**) when selecting an item from a **pull down** or a **pull right** menu of 4 items*
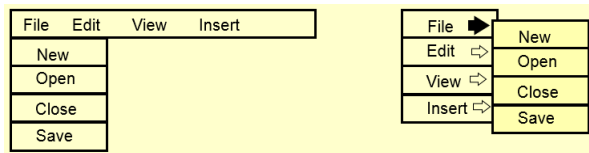
Validity

External validity

- – confidence that results applies to real situations
- – usually good in natural settings

Internal validity

- – confidence in our explanation of experimental results
- – usually good in experimental settings

Trade-off: Natural *vs* Experimental

- – precision and direct control over experimental design
  versus
- – desire for maximum generalizability in real life situations

*Usability engineering approach

- **usability engineering** - iterative process to improve usability of a system
- **usability** - the extent to which a product can be used by specified users to achieve specified goals with *effectiveness*, *efficiency* and *satisfaction* in a specified context of use [ISO 1998]
  - **effectiveness** - accuracy and completeness in achieving specified goals
  - **efficiency** - resources expended in relation to the accuracy and completeness in achieving goals
  - **satisfaction** - freedom from discomfort, and positive attitudes towards the use of the product

*Usability engineering approach

Types of evaluation (according to its purpose)

- ▶ **exploratory** - how is it (or will it be) used?
  - ▶ explores current usage and the potential design space for new designs

- ▶ **predictive** - estimating how good it will be
  - ▶ estimates the overall quality of an interface (once a design has been made)

- ▶ **formative** - how can it be made better?
  - ▶ informs the design process and helps improve an interface during design

- ▶ **summative** - how good is it?
  - ▶ assesses the overall quality of an interface

Usability inspection methods

Designer tries the system (or prototype)

- does the system "feel right"?
- benefits
  - can catch some major problems in early versions
- problems
  - not reliable as completely subjective
  - not valid as introspector is a non-typical user
  - intuitions and introspection are often wrong

Usability inspection methods:

- task centered walkthroughs
- heuristic evaluation

Usability testing methods

Observe people using systems in **simulated** settings

- people brought in to artificial setting that simulates aspects of real world setting
- people given specific tasks to do
- observations / measures made as people do their tasks
- look for problem areas / successes
- good for uncovering 'big effects'

Usability testing methods

Is the test result relevant to the usability of real products in real use outside of lab?

Problems

– non-typical users tested

– non-typical tasks

– different physical environment

– different social context

  • motivation towards experimenter vs motivation towards boss

Partial solution

– use real users

– task-centered system design tasks

– environment similar to real situation

Usability testing methods

How many users should you observe?

- – observing many users is expensive
- – *but* individual differences matter
  - • best user 10x faster than slowest
  - • best 25% of users approx. 2x faster than slowest 25%

Partial Solution

- – reasonable number of users tested
- – reasonable range of users
- – big problems usually detected with handful of users
- – small problems / fine measures need many users
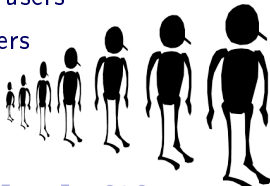
Usability testing methods

Low cost methods to gather usability problems

– approximate: capture most large and many minor problems

How?

– **qualitative**:
  - observe user interactions
  - gather user explanations and opinions
  - produces a description, usually in non-numeric terms
  - anecdotes, transcripts, problem areas, critical incidents. . .
– **quantitative**
  - count, log, measure something of interest in user actions
  - speed, error rate, counts of activities

Qualitative usability testing methods

Methods

- extracting the conceptual model
- direct observation
  - think-aloud
  - constructive interaction/co-discovery
- query techniques (interviews and questionnaires)
- continuous evaluation (user feedback and field studies)

Conceptual model extraction

How?

- – show the user static images of
  - • the prototype *or* screens during use
- – ask the user explain
  - • the function of each screen element
  - • how they would perform a particular task

What?

- – **Initial conceptual model**
  - • how person perceives a screen the very first time it is viewed
- – **Formative conceptual model**
  - • How person perceives a screen after its been used for a while

Value?

- – good for eliciting people's understanding before & after use
- – poor for examining system exploration and learning

Direct observations

Evaluator observes users interacting with system

- – in lab:
  - • user asked to complete a set of pre-determined tasks
- – in field:
  - • user goes through normal duties

Value

- – excellent at identifying gross design/interface problems
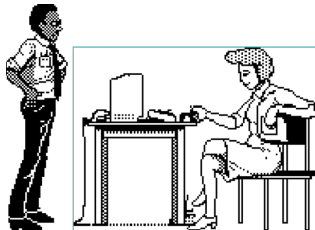- – validity depends on how controlled/contrived the situation is

Simple observation method

User is given the task
Evaluator just watches the user

Problem

– does not give insight into the user's decision process and
attitude

Think aloud method

Users speak their thoughts while doing the task
- – what they are trying to do
- – why they took an action
- – how they interpret what the system did
- – gives insight into what the user is thinking
- – most widely used evaluation method in industry
  - • may alter the way users do the task
  - • unnatural (awkward and uncomfortable)
  - • hard to talk if they are concentrating

Hmm, what does this do? I'll try it… Ooops, now what happened?
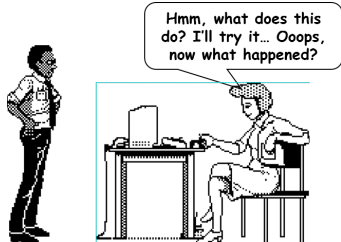
Constructive interaction/Co-discovery method

Two people work together on a task

- monitor their normal conversations
- removes awkwardness of think-aloud
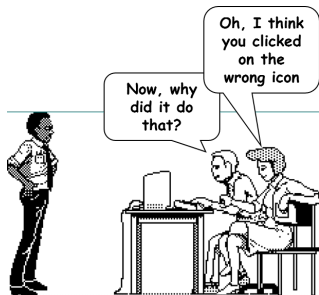
Constructive interaction/Co-discovery method

Co-discovery learning
- use semi-knowledgeable "coach" and novice
- only novice uses the interface
  - novice asks questions
  - coach responds
- gives insights into two user groups

Recording observations

How do we record user actions for later analysis?

- otherwise
  risk forgetting, missing, or misinterpreting events
- paper and pencil
  - primitive but cheap
  - observer
    records events, comments, and interpretations
  - hard to get detail (writing is slow)
  - 2nd observer helps...
- audio recording
  - good for recording think aloud talk
  - hard to tie into on-screen user actions
- video recording
  - can see and hear what a user is doing
  - one camera for screen, rear view mirror useful
  - initially intrusive

Coding sheet example...

tracking a person's use of an editor

Time	General actions			Graph editing			Errors	
	text editing	scrolling	image editing	new node	delete node	modify node	correct error	miss error
09:00	X							
09:02				X				
09:05							X	
09:10					X			
09:13								

Interviews

Good for pursuing specific issues
- – vary questions to suit the context
- – probe mode deeply on interesting issues as they arise
- – good for exploratory studies via open-ended questioning
- – often leads to specific constructive suggestions

Problems:
- – accounts are subjective
- – time consuming
- – evaluator can easily bias the interview
- – prone to rationalization of events/thoughts by user
  - • user's reconstruction may be wrong

How to interview

Plan a set of central questions
- a few good questions gets things started
    - avoid leading questions
- focuses the interview
- could be based on results of user observations

Let user responses lead follow-up questions
- follow interesting leads *vs* bulldozing through question list

Retrospective testing interviews

Post-observation interview to
- perform an observational test
- create a video record of it
- have users view the video and comment on what they did
  - clarify events that occured during system use
  - excellent for grounding a post-test interview
  - avoids erroneous reconstruction
  - users often offer concrete suggestions
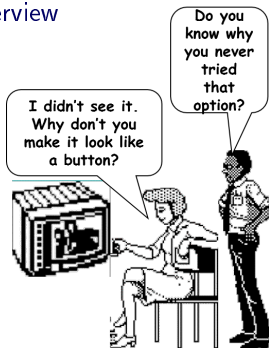
Critical incidence interviews

People talk about incidents that stood out
- – usually discuss extremely annoying problems with fervor
- – not representative, but important to them
- – often raises issues not seen in lab tests
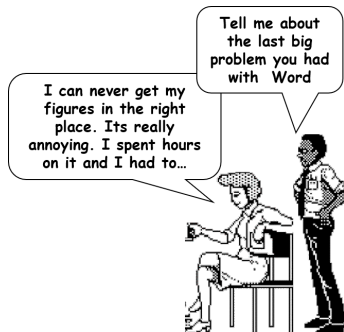
Questionnaires and surveys

Questionnaires / Surveys
- – preparation "expensive," but administration cheap
  - • can reach a wide subject group (e.g. mail)
- – does not require presence of evaluator
- – results can be quantified

But
- – only as good as the questions asked

Questionnaires and surveys

How

- – establish the purpose of the questionnaire
    - what information is sought?
    - how would you analyze the results?
    - what would you do with your analysis?

- – do not ask questions whose answers you will not use!

- – determine the audience you want to reach

- – determine how would you will deliver / collect the questionnaire
    - on-line for computer users
    - web site with forms
    - surface mail
        - – pre-addressed reply envelope gives far better response

Styles of questions

Open-ended questions
- – asks for unprompted opinions
- – good for general subjective information
  - • but difficult to analyze rigorously

Can you suggest any improvements to the interfaces?

Styles of questions

Closed questions
- – restrict respondent's responses by supplying alternative answers
- – makes questionnaires a chore for respondent to fill in
- – can be easily analyzed
- – watch out for hard to interpret responses!
  - • alternative answers should be very specific

Do you use computers at work:

✓ often   ◯ sometimes   ◯ rarely

vs

In your typical work day, do you use computers:

◯ over 4 hrs a day
◯ between 2 and 4 hrs daily
✓ between 1 and 2 hrs daily
◯ less than 1 hr a day

Styles of questions

Scalar

- – ask user to judge a specific statement on a numeric scale
- – scale usually corresponds with agreement or disagreement with a statement

Characters on the computer screen are:

- – hard to read   easy to read
- –   1   ②   3   4   5

Styles of questions

Multi-choice

- respondent offered a choice of explicit responses

How do you most often get help with the system? (tick one)

☑ on-line manual

◯ paper manual

◯ ask a colleague

Which types of software have you used? (tick all that apply)

☑ word processor

◯ data base

☑ spreadsheet

◯ compiler

Styles of questions

Ranked

– respondent places an ordering on items in a list

– useful to indicate a user's preferences

– forced choice

Rank the usefulness of these methods of issuing a command (1 most useful, 2 next most useful..., 0 if not used

_1_ command line

_2_ menu selection

_3_ control key accelerator

Styles of questions

Combining open-ended and closed questions

   – gets specific response, but allows room for user's opinion

It is easy to recover from mistakes:

disagree   agree   comment: <u>the undo facility is</u>
<u>really helpful</u>
      1 ② 3 4 5

Continuous evaluation

Monitor systems in actual use
- usually late stages of development
  - ie beta releases, delivered system
- fix problems in next release

User feedback via gripe lines
- users can provide feedback to designers while using the system
  - help desks
  - bulletin boards
  - email
  - built-in gripe facility

- best combined with trouble-shooting facility
  - users always get a response (solution?) to their gripes

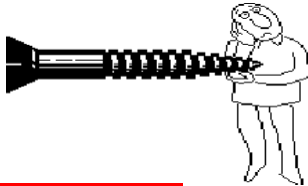# Continuous evaluation

Case/field studies

- careful study of "system usage" at the site
- good for seeing "real life" use
- external observer monitors behavior
- site visits

Ethics

Ethics

Testing can be a distressing experience
- pressure to perform, errors inevitable
- feelings of inadequacy
- competition with other subjects

Golden rule
- subjects should always be treated with respect

Ethics - before the test

Don't waste the user's time
- use pilot tests to debug experiments, questionnaires etc
- have everything ready before the user shows up

Make users feel comfortable
- emphasize that it is the system that is being tested, not the user
- acknowledge that the software may have problems
- let users know they can stop at any time

Maintain privacy
- tell user that individual test results will be completely confidential

Inform the user
- explain any monitoring that is being used
- answer all user's questions (but avoid bias)

Only use volunteers
- user must sign an informed consent form

Ethics - during the test

Don't waste the user's time

– never have the user perform unnecessary tasks

Make users comfortable

– try to give user an early success experience
– keep a relaxed atmosphere in the room
– coffee, breaks, etc
– hand out test tasks one at a time
– never indicate displeasure with the user's performance
– avoid disruptions
– stop the test if it becomes too unpleasant

Maintain privacy

– do not allow the user's management to observe the test

Ethics - after the test

Make the users feel comfortable

– state that the user has helped you find areas of improvement

Inform the user

– answer particular questions about the experiment that could have biased the results before

Maintain privacy

– never report results in a way that individual users can be identified

– only show videotapes outside the research group with the user's permission

What you now know

Debug designs by observing how people use them

- quickly exposes successes and problems
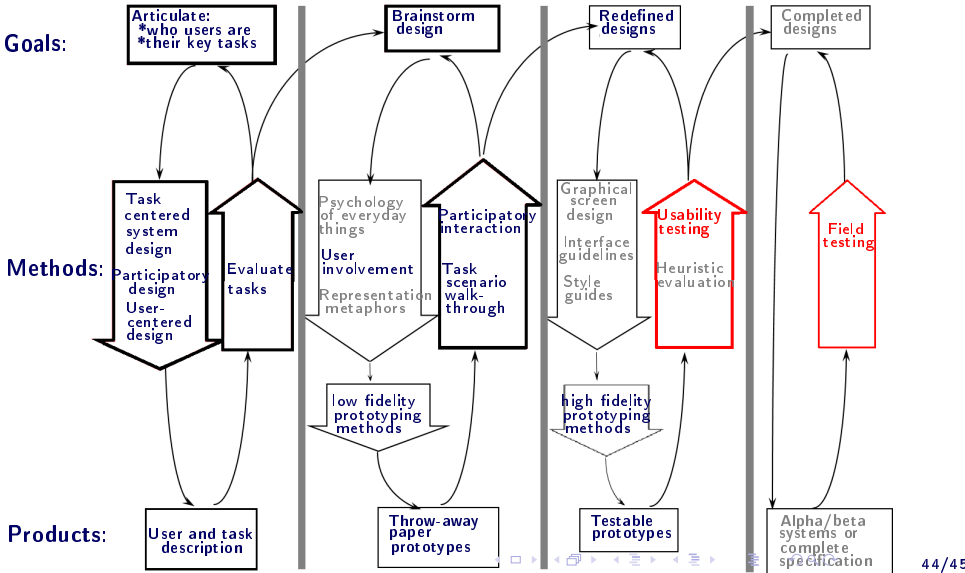- specific methods reveal what a person is thinking
- but naturalistic vs laboratory evaluations is a tradeoff

Methods include

- conceptual model extraction
- direct observation
  - think-aloud
  - constructive interaction/co-discovery
- query via interviews, retrospective testing and questionnaires
- continuous evaluation via user feedback and field studies

Ethics are important

Interface Design and Usability Engineering

Goals:

| Articulate: *who users are *their key tasks | Brainstorm design | Redefined designs | Completed designs |

Methods:

Task centered system design

Participatory design

User-centered design

Evaluate tasks

Psychology of everyday things

User involvement

Representation metaphors

Participatory interaction

Task scenario walk-through

Graphical screen design

Interface guidelines

Style guides

Usability testing

Heuristic evaluation

Field testing

low fidelity prototyping methods

high fidelity prototyping methods

Products:

| User and task description | Throw-away paper prototypes | Testable prototypes | Alpha/beta systems or complete specification |

*Bibliography

- Saul Greenberg, **Designing with the user. User centered design and Prototyping**, University of Calgary, Canada
  `http://pages.cpsc.ucalgary.ca/~saul/481/`

- Keith Andrews, **Human Computer Interaction, Chapter 3. Usability Engineering, Chapter 9. Usability Testing Methods**, TU Graz, Austria
  `https://courses.isds.tugraz.at/hci/hci.pdf`