Vorlesung
Mensch-Maschine-Interaktion

Methods & Tools

Ludwig-Maximilians-Universität München
LFE Medieninformatik
Andreas Butz & Albrecht Schmidt
WS2004/2005
http://www.medien.informatik.uni-muenchen.de/

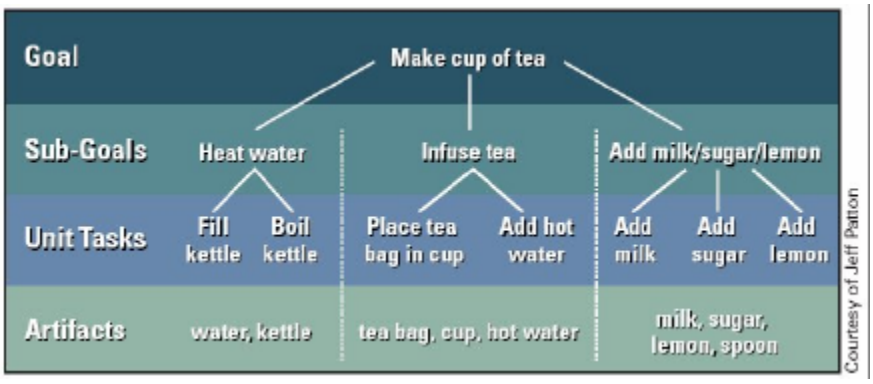
Table of Content

- Task Analysis
- Scenario Development
- Sketches
- Prototyping

Task Analysis - Motivation

- Basically it is about all the actions performed by the user to accomplish a task
  - Its is about what we can observe
  - It is not really about the mental model
- Example – setting up a video projector:
  - unpacking the projector and placing it on the table
  - connecting the power cable to the projector and the socket
  - connecting a data cable between projector and computer
  - switching on the projector
  - waiting for the projector to be ready
  - switching the computer to dual screen mode
- Some issues
  - There is no single way to do that…
  - Granularity and details
  - Order of action

Task Analysis - Example



see: William Hudson. HCI and the web: A tale of two tutorials: a cognitive approach to interactive system design and interaction design meets agility. interactions Volume 12, Number 1 (2005), Pages 49-51

Task Analysis – High level Questions

- How do users know their goal is attainable?
- How do users know what to do?
- How will users know they have done the right thing?
- How will users know they have attained their goal?

William Hudson. HCI and the web: A tale of two tutorials: a cognitive approach to interactive system design and interaction design meets agility. interactions Volume 12, Number 1 (2005), Pages 49-51

Wharton, C., Rieman, J., Lewis, C., & Polson, P. (1994). The cognitive walkthrough method: A practitioner's guide. In J. Nielsen & R. L. Mack (eds.). Usability inspection methods. New York, NY: John Wiley.

What can we examine in Task Analysis?

- Input to the computer (keyboard, mouse, etc.)
- Physical actions, e.g. head movement, turning on the chair to reach for a document, lifting the mouse
- Perceptual actions, e.g. recognizing things that appear on the screen, finding a tool again
- Cognitive actions
- Mental actions and decision making
- Memory recall

Task analysis
Set of basic questions

- Who is going to use the system? .
- What tasks do they now perform?
- What tasks are desired?
- How often are the tasks carried out?
- What time constraints on the tasks?
- What knowledge is required to do the task?
- How are the tasks learned?
- Where are the tasks performed (environment)?
- What other information and tools are required to do the task?
- What's the relationship between user & data?
- What is the procedure in case of errors and failures?
- Multi-user system: How do users communicated (CSCW Matrix)?

Hierarchical Task Analysis

- Identify the goals the user wants to achieve
- Relate the goals to tasks (and potentially planning) done by the user
- Task decomposition
  - Ordering
  - Alternative plans
- How to limit the tasks to consider?
  - Defining a threshold based on probability of the task and cost in case of failure
  - If (failure_cost(task) * probability(task)) < threshold do not further consider this task
- For a detailed discussion on Task Analysis (hierarchical task analysis, knowledge based analysis, entity-relationship based technique, see Dix et. al – chapter 7)

Walk-Through

- Task performed on a existing system or a simulation
- Go step by step through a selected task (if possible with multiple people)
- Collect data about the procedure (video/audio)
- Collect data on performance and potentially on differences between users
- Encourage the user to comment his actions

Task Action Mappings

- Creating a directional link between a task and the action performed
- Mappings
  - One-to-one
    each task forces the user to perform a specific action
  - Many-to-one
    a set of tasks can be done by performing one action
  - One-to-many
    for one task a set of actions may be performed
  - Many-to-many
    a set of tasks is done by performing a set of actions

http://www.psy.gla.ac.uk/~steve/HCI/cscln/trail1/Lecture8.html

Scenario Development

- Especially useful for novel systems where there is little experience or knowledge
- Important methods
  - **general scenario** (fictional story featuring the product to be developed and explaining implications on users experience) – similar to describing conceptual models
  - **"day in the life" scenario** (creating a fictional user, describing a day in her life augmented with the product to be developed)
  - **situation scenarios** (fictional story concentrating on a specific situation, e.g. an emergency case)
- Forms of presentation
  - writing
  - video
  - acting/playing it – connected to paper prototypes

Scenario Development
What user? Who to design for?

- Don't design for the average user!!!
- Differentiate and create a set of typical users (often also called "Persona")
- You will need background information about the user group to create a set of persona
  - Literature
  - Interviews
  - Statistics
  - Analysis and observations
- Create a set of specific persons (you invent them based on the collected data)
  - Age, place of birth, current location where she lives
  - Education, profession, job profile, background, hobbies
  - Social environment, family, work relationships
  - Goals and abilities
- They are representative for the target audience, but the are NOT average!

Scenario Development
Why Persona?

- Avoiding the “elastic user”
  - If you do not specify the user you can change their abilities to support a design decision made = “elastic user”
- Avoiding self-referential design
  - The designer or developer on assumes (implicitly) that user’s have his goals and his skills and abilities.
- Avoiding design edge cases
  - Focusing on the design issues which are on the edge of the anticipated audience can consume a lot of effort. By use of typical users the focus on edge case can be reduced.

“day in the life” scenario

- Describe the usage of a product in the context of a day
  - In particular for products that are used more than once a day, e.g. mobile services, helps to identify practicalities
- Based on the information gathered invent a day
  - Working day or holiday
  - Make a plan what the persons is going to do on this day
  - Make it a normal day but include real life tension and trade-off (e.g. getting kids to school and having a meeting shortly after that)
  - Don’t let the day to be perfect (e.g. you may forget a document at home)
  - Don’t make the day a nightmare (e.g. do not anticipate the user’s airplane is going to crash)
- Describe a day of the fictional user in detail
  - Concentrate on the relation between the users actions and tasks and the product introduced.
  - Basically asking: “How does the product change the life?”

“day in the life” scenario

Example from the European Project TEA: general approach

- Project Vision: Creating a mobile phone/PDA that is aware of the user’s action and the environment (e.g. user is driving, user is holding the device, user is in a meeting, it is raining, user is at a particular location etc.)
- Technology driven – but what are the applications?
- “day in the life” scenario for 6 users to explore possible uses (user are already mobile phone “power” users)
  - Franz, 34, journalist, Munich
  - Meredith, 38, Vice President, Marketing, Chicago
  - Mike, age 14, lives in Bath in the UK, ordinary school
  - Patricia, 35, Architect & building designer, Bologna
  - Jochen, 24, geo-physics student, Salzburg
  - Janni, 43, field engineer for a power company, Finland

“day in the life” scenario

Example from the European Project TEA: a day in Meredith’s life

- Complete scenario is about 6 pages, excerpts form the main sections
- User and Situation Summary
  - Professional, Female Doctor, Vice President, Marketing
  - Meredith, 38 in Chicago/USA
  - Married to Tom 37 (IT-professional), having a daughter Sheila (7 years).
  - The day: traveling, Medical Conference, A lot of meetings before the Conference duties, in conference Hotels and conference boot
- User

  “*Meredith Miller is a 38 year old Marketing specialist in the pharmaceutical industry. She was born in the U.K. but now she is based in Chicago, USA. She works for a medium company dealing with pharmaceutical products marketing and distribution, which acts as a strategy consultant for large pharmaceutical and medicinal preparations companies worldwide. She has a degree in medicine, and a master's degree in business administration for pharmaceutical and medical industry*….”
- Situation

  “*This week, Meredith is traveling across Europe for her monthly visit to European key customers. It is also a special week because two important events, a scientific convention in Copenhagen and an industry fair in Hannover are being held*…”

Situation Scenarios

- Concentrating on a very specific situation
- Investigate the requirements and the impact in a specific situation
- May be rather short
- Situation were the product and potentially a particular function is situated into a context
  - e.g. scanning a document in a work context (interrupting work, going to the scanner, operating the device, getting the data, ..)
- Unlikely situations that are of major importance
  - E.g. emergency procedures such as a fire or building evacuation (not applicable to a word processor but relevant for a power plant control room)
- Methods
  - Writing a fictional story
  - Playing/acting the scene with anticipated functionality

Problems of User Centered Design

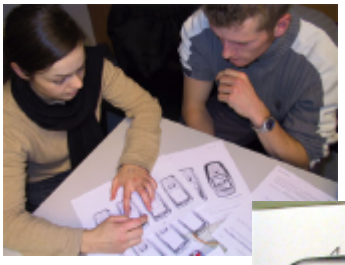
- Users may be wrong
- Users may be resistant to change
- Users may expect disadvantages (e.g. being replaced by software)

- Be aware – you are expected to create an optimal system with regards to the goals specified and this is unfortunately NOT necessarily the system users would like to have (e.g. trade-off between employers and employees)

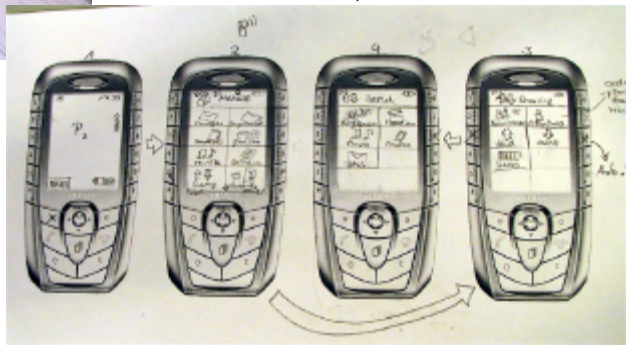
Design Cycles & Prototyping

- Creating prototypes is important to get **early** feedback
  - from the project team (prototypes help to communicate)
  - from potential users
- Different types of prototypes
  - Low-fidelity prototypes (e.g. paper prototypes, sketches)
  - Hi-fidelity prototypes (e.g. implemented and semi-functional UI, could look like the real product)
  - Fidelity is referring to detail
- Tools & Methods
  - Sketches & Storyboards
  - Paper prototyping
  - Using GUI-builders to prototype
  - Limited functionality simulations
  - Wizard of Oz

Sketches & Storyboards



- Storyboards as in movies
  - A picture for each key scene
- Sketch out the application
  - Key screens
  - Main interaction
  - Important transitions
- Helps to communicate and validate ideas
  - Easy to try out different option, e.g. document base vs. application based
- Ignore details, e.g.
  - what font to use, how icons will look like



Paper Prototypes

- Specify the set of tasks that should be supported
- Create a paper prototype using office stationery
  - Screens, dialogs, menus, forms, …
  - Specify the interactive behavior
- Use the prototype
  - Give users a specific task and observe how they use the prototype
  - Ask users to “think aloud” – comment what they are doing
  - At least two people
    - One is simulating the computer (e.g. changing screens)
    - One is observing and recording
- Evaluate and document the findings
  - What did work – what did not work
  - Where did the user get stuck or chose alternative ways
  - Analyze comments from the user
- Iterate over the process (make a new version)

Low-Fidelity Prototyping

- Advantages of paper prototypes
  - Cheap and quick – results within hours!
  - Helps to find general problems and difficult issues
  - Make the mistakes on paper and make them before you do your architecture and the coding
  - Can save money by helping to get a better design (UI and system architecture) and a more structured code
  - Enables non-technical people to interact easily with the design team (no technology barrier for suggestions)
- Get users involved!
  - To get the full potential of paper-prototypes these designs have to be tested with users
  - Specify usage scenarios
  - Prepare tasks that can be done with the prototype

Minimize the time for design Iterations
Make errors quickly!

- Idea of rapid prototyping
- Enables the design team to evaluate more design options in detail
- If you go all the way before evaluating your design you risk a lot!
- Sketches and paper prototypes can be seen as a simulation of the real prototype

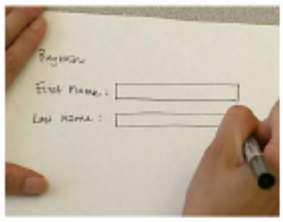
- Without paper prototyping:
  - Idea – sketch – implementation – evaluation

Slow Iteration

- With paper prototyping:
  - Idea – sketch/paper prototype – evaluation – implementation - evaluation

Quick Iteration **Slow Iteration**

Video – N&N Paper Prototyping (second part)



Nielsen Norman Group Video:
Paper Prototyping: A How-To Training Video

High-fidelity Prototype

- Looks & feels like the final product to the user
  - Colors, screen layout, fonts, …
  - Text used
  - Response time and interactive behavior
- The functionality however is restricted
  - Only certain functions work (vertical prototype)
  - Functionality is targeted towards the tasks (e.g. a search query is predetermined)
  - Non-visible issues (e.g. security) are not regarded
- Can be used to predict task efficiency of the product
- Feedback often centered around the look & feel
- Standard technologies for implementation
  - HTML, JavaScript
  - Flash, Director, Presentation programs
  - GUI Builder (e.g. Visual Basic, Delphi, NetBeans)

Functional Prototypes

- Often used as synonym for High-fidelity Prototype

- To encourage feedback that is not related to the look & feel it may be helpful to make the GUI look rough, see reading:
  R. Van Buskirk and B. W. Moroney: Extending Prototyping, IBM Systems Journal - Vol. 42, No. 4, 2003 - Ease of Use.

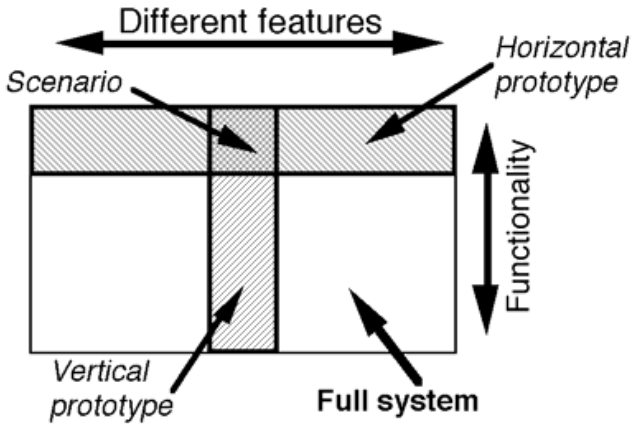
Horizontal Prototyping

- Demonstrate the feature spectrum of a product
- Allows the user to navigate the system
- The actual functions are not implemented
- Helps to evaluate/test
  - Navigation (e.g. finding a specific function or feature)
  - Overall user interface concept
  - Feature placement
  - Accessibility
  - User preferences
- Applicable in low-fidelity prototyping and high-fidelity prototyping
- Used in early design stages
  - To determine the set of features to include
  - To decide on the user interface concept
- Example: overall usage of a mobile phone

Vertical Prototyping

- Demonstrate a selected feature of a product
- Allows the user only to use this specific function
- The details of the function/feature are shown/implemented
- Helps to evaluate/test
  - The optimal design for a particular function
  - Optimize the usability of this function
  - User performance for this particular function
- Mainly use in high-fidelity prototyping but can be applicable to low-fidelity prototyping
- Used in early design stages
  - To compare different designs for a specific function
- Used in later design stages
  - To optimize usage of a function
- Example: a new input methods for writing SMS on a mobile phone

Addition – about Prototypes



- http://www.useit.com/papers/guerrilla_hci.html

1984 Olympic Message System A human centered approach

- A public system to allow athletes at the Olympic Games to send and receive recorded voice messages (between athletes, to coaches, and to people around the world)
- Challenges
  - New technology
  - Had to work – delays were not acceptable (Olympic Games are only 4 weeks long)
  - Short development time
- Design Principles
  - Early focus on users and tasks
  - Empirical measurements
  - Iterative design

  → Looks obvious – but it is not!
- … it worked! But why?

1984 Olympic Message System Methods

- Scenarios instead of a list of functions
- Early prototypes & simulation (manual transcription and reading)
- Early demonstration to potential users (all groups)
- Iterative design (about 200 iterations on the user guide)
- An insider in the design team (ex-Olympian from Ghana)
- On side inspections (where is the system going to be deployed)
- Interviews and tests with potential users
- Full size kiosk prototype (initially non-functional) at a public space in the company to get comments
- Prototype tests within the company (with 100 and with 2800 people)
- “free coffee and doughnuts” for lucky test users
- Try-to-destroy-it test with computer science students
- Pre-Olympic field trail

The 1984 Olympic Message System: a test of behavioral principles of system design John D. Gould , Stephen J. Boies , Stephen Levy , John T. Richards , Jim Schoonard Communications of the ACM September 1987 Volume 30 Issue 9
http://www.research.ibm.com/compsci/spotlight/hci/p758-gould.pdf

Wizard-of-Oz

- “The man behind the curtain”
- Basically don’t not implement the hard parts in the prototype – just let a human do
- Typical areas
  - Speech recognition
  - Speech synthesis
  - Annotation
  - Reasoning
  - Visual Perception
- Provides the user with the experience without extensive implementation effort for the prototype

Table of Content

- An example of user centred design
- What to evaluate?
- Why Evaluate?
- Approaches to evaluation
- Inspection and expert review
- Model extraction
- Observations
- Experiments
- Ethical Issues

What to evaluate?

- The usability of a system!
- … it depends on the stage of a project
  - Ideas and concepts
  - Designs
  - Prototypes
  - Implementations
  - Products in use
- … it also depends on the goals
- Approaches
  - Formative evaluation – throughout the design, helps to shape a product
  - Summative evaluation – quality assurance of the finished product.

Why evaluate?
Goals of user interface evaluation

- Ensure functionality (effectiveness)
  - Assess (proof) that a certain task can be performed
- Ensure performance (efficiency)
  - Assess (proof) that a certain task can be performed given specific limitations (e.g. time, resources)
- Customer / User acceptance
  - What is the effect on the user?
  - Are the expectations met?
- Identify problems
  - For specific tasks
  - For specific users
- Improve development life-cycle
- Secure the investment (don't develop a product that can only be used by fraction of the target group – or not at all!)

There is not a single way …

- Different approaches
  - Inspections
  - Model extraction
  - Controlled studies
  - Experiments
  - Observations
  - Field trails
  - Usage context
- Different results
  - Qualitative assessment
  - Quantitative assessment

Usability Methods are often not used!

- Why
  - Developers are not aware of it
  - The expertise to do evaluation is not available
  - People don't know about the range of methods available
  - Certain methods are to expensive for a project (or people think they are to expensive)
  - Developers see no need because the product "works"
  - Teams think their informal methods are good enough
- starting points
  - Discount Usability Engineering
    http://www.useit.com/papers/guerrilla_hci.html
  - Heuristic Evaluation
    http://www.useit.com/papers/heuristic/

Inspections & Expert Review

- Throughout the development process
- Performed by developers and experts
- External or internal experts
- Tool for finding problems
- May take between an hour and a week
- Structured approach is advisable
  - reviewers should be able to communicate all their issues (without hurting the team)
  - reviews must not be offensive for developers / designers
  - the main purpose is finding problems
  - solutions may be suggested but decisions are up to the team

Inspection and Expert Review Methods

- Guideline review
  - Check that the UI is according to a given set of guidelines
- Consistency inspection
  - Check that the UI is consistent (in itself, within a set of related applications, with the OS)
  - Birds's eye view can help (e.g. printout of a web site and put it up on the wall)
  - Consistency can be enforced by design (e.g. css on the web)
- Walkthrough
  - Performing specific tasks (as the user would do them)
- Heuristic evaluation
  - Check that the UI violates a set (usually less than 10 point) rules

Informal Evaluation

- Expert reviews and inspections are often done informally
  - UIs and interaction is discussed with colleagues
  - People are asked to comment, report problems, and suggest additions
  - Experts (often within the team) assess the UI for conformance with guidelines and consistency
- Results of informal reviews and inspections are often directly used to change the product
- … still state of the art in many companies!
- Informal evaluation is important but in most cases not enough

- Making evaluation more explicit and documenting the findings can increase the quality significantly
- Expert reviews and inspections are a starting point for change

Discount Usability Engineering

- Low cost approach
- Small number of subjects
- Approximate
  - Get indications and hints
  - Find major problems
  - Discover many issues (minor problems)
- Qualitative approach
  - observe user interactions
  - user explanations and opinions
  - anecdotes, transcripts, problem areas, …
- Quantitative approach
  - count, log, measure something of interest in user actions
  - speed, error rate, counts of activities

Heuristic Evaluation

http://www.useit.com/papers/heuristic/

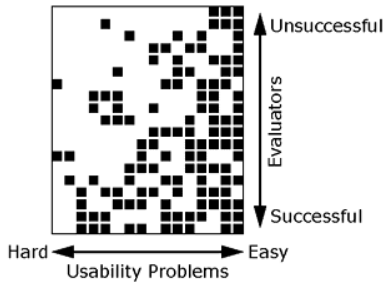
- Heuristic evaluation is a usability inspection method
- systematic inspection of a user interface design for usability
- goal of heuristic evaluation
  - to find the usability problems in the design
- As part of an iterative design process.

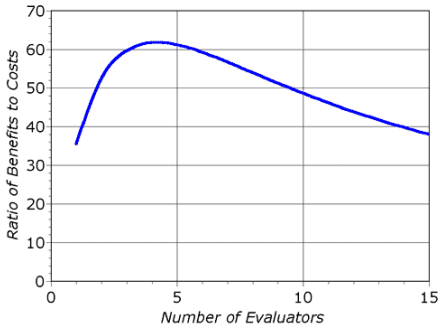
- Basic Idea:
  Small set of evaluators examine the interface and judge its compliance with recognized usability principles (the "heuristics").

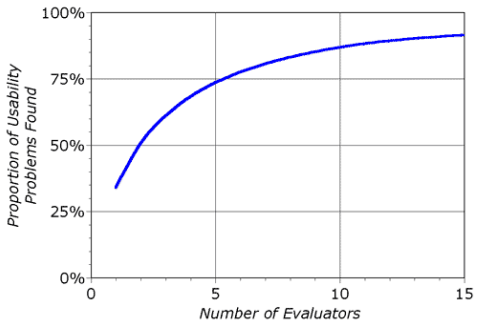
Heuristic Evaluation

http://www.useit.com/papers/heuristic/

- How many evaluators?
- Example: total cost estimate with 11 evaluators at about 105 hours, see
  http://www.useit.com/papers/guerrilla_hci.html







Heuristic Evaluation - Heuristics

- Heuristics suggested by Nielsen
  - Visibility of system status
  - Match between system and the real world
  - User control and freedom
  - Consistency and standards
  - Error prevention
  - Recognition rather than recall
  - Flexibility and efficiency of use
  - Aesthetic and minimalist design
  - Help users recognize, diagnose, and recover from errors
  - Help and documentation
- Depending of the product and goals a different set may be appropriate

Heuristic Evaluation - Steps

- Preparation
  - Assessing appropriate ways to use heuristic evaluation
  - Define Heuristics
  - Having outside evaluation expert learn about the domain and scenario
  - Finding and scheduling evaluators
  - Preparing the briefing
  - Preparing scenario for the evaluators
  - Briefing (system expert, evaluation expert, evaluators)
  - Preparing the prototype (software/hardware platform) for the evaluation
- Evaluation
  - Evaluation of the system by all evaluators
  - Observing the evaluation sessions
- Analysis
  - Debriefing (evaluators, developers, evaluation expert)
  - compiling list of usability problems (using notes from evaluation sessions)
  - Writing problem descriptions for use in severity-rating questionnaire
  - Severity rating

Heuristic Evaluation – Severity Rating

- Severity ratings are used to prioritize problems
- Decision whether to release a system or to do further iterations
- The severity of a usability problem is a combination of three factors:
  - The frequency with which the problem occurs: Is it common or rare?
  - The impact of the problem if it occurs: Will it be easy or difficult for the users to overcome?
  - The persistence of the problem: Is it a one-time problem that users can overcome once they know about it or will users repeatedly be bothered by the problem
- 0 to 4 rating scale to rate the severity of usability problems:
  - 0 = I don't agree that this is a usability problem at all
  - 1 = Cosmetic problem only: need not be fixed unless extra time is available on project
  - 2 = Minor usability problem: fixing this should be given low priority
  - 3 = Major usability problem: important to fix, so should be given high priority
  - 4 = Usability catastrophe: imperative to fix this before product can be released

Observations & Protocols

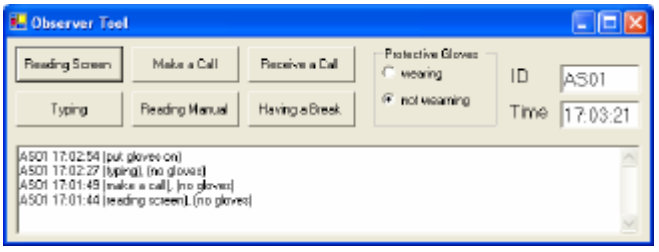
- Paper and pencil
  - Cheap and easy but unreliable
  - Make structured observations sheets / tool
- Audio/video recording
  - Cheap and easy
  - Creates lots of data, potentially expensive to analyze
  - Good for review/discussion with the user
- Computer logging
  - Reliable and accurate
  - Limited to actions on the computer
  - Include functionality in the prototype / product
- User notebook
  - Request to user to keep a diary style protocol

Structured observations

- Observation sheet

time	typing	reading screen	consulting manual	phoning	…
14:00		X		X	
14:01	X		X		
14:02	X				
14:03	X				
14:04				X	
…					

- Electronic version

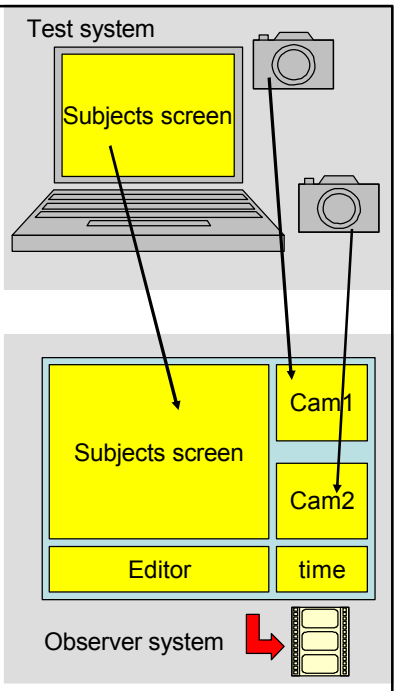


Observations and Protocols

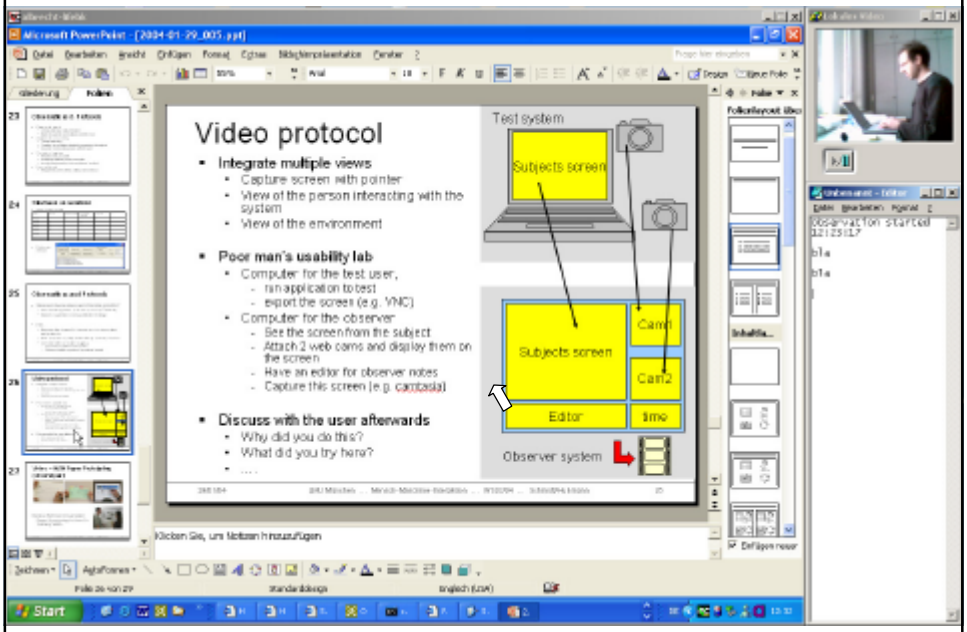
- What are observations and Protocols good for?
  - Demonstrating that a product improves productivity
  - Basis for qualitative and quantitative findings
- Hint
  - Minimize the chance for human error in observation and protocols
  - Most people are pretty bad at doing manual protocols
  - Combine with computer logging
    - Log what you get from the system
    - Observer makes a protocol on external events

Video protocol

- Integrate multiple views
  - Capture screen with pointer
  - View of the person interacting with the system
  - View of the environment
- Poor man's usability lab
  - Computer for the test user,
    - run application to test
    - export the screen (e.g. VNC)
  - Computer for the observer
    - See the screen from the subject
    - Attach 2 web cams and display them on the screen
    - Have an editor for observer notes
    - Capture this screen (e.g. camtasia)
- Discuss with the user afterwards
  - Why did you do this?
  - What did you try here?
  - ….



Screen video



References

- Alan Dix, Janet Finlay, Gregory Abowd and Russell Beale. (1998) Human Computer, Interaction (second edition), Prentice Hall, ISBN 0132398648 (new Edition announced for October 2003)
- Ben Shneiderman. (1998) Designing the User Interface, 3rd Ed., Addison Wesley; ISBN: 0201694972

- Discount Usability Engineering
  http://www.useit.com/papers/guerrilla_hci.html
- Heuristic Evaluation
  http://www.useit.com/papers/heuristic/

Vorlesung
Mensch-Maschine-Interaktion

Evaluation

Ludwig-Maximilians-Universität München
LFE Medieninformatik
Andreas Butz & Albrecht Schmidt
WS2004/2005
http://www.medien.informatik.uni-muenchen.de/

Examples of methods used in different phases

- Analysis
  - Goal & user analysis
  - Task analysis
  - Contextual enquiry and observations
- Early design phase
  - Sketches and paper prototypes
  - Cognitive walkthroughs
  - Heuristic evaluation
- Late design phase
  - Functional prototypes
  - User studies and experiments
- Implementation
  - User studies
  - Functional tests
  - Acceptance tests
  - Performance tests
- Operational product
  - Support analysis
  - Interaction logs
  - Field studies
  - Acceptance tests

Cognitive Walkthrough

Dix et al. Chapter 11

- For interfaces that can be learned by exploration
- Experts step through a task to question the design
  - focusing on the users' knowledge and goals
  - asking whether the users will experience difficulties at each step

- Requirements
  - A description of the system prototype i.e where will it be located, exact wordings of menus or a prototype.
  - A description of the task the user will be expected to do - the most common
  - A list of the correct actions that are required to complete the task
  - A description of who the users will be, their experience and prior knowledge

Cognitive Walkthrough - Questions

Dix et al. Chapter 11

- Evaluator works through the action list and at each step they ask:
  1. Will users be trying to produce whatever effect the action has?
  2. Will users see the control (button, menu, switch, etc.) for the desired action?
  3. Once users find the control, will they recognize that it produces the effect they want?
  4. After the action is taken, are the users given adequate feedback, so they can go on to the next action with confidence?

Organizing a Cognitive Walkthrough

Dix et al. Chapter 11

- Requires good and precise documentation
  - task description
  - details on action steps
  - user information
- For each action step the evaluator comments of the four questions
- If the answer to any question is no, this indicates a usability problem → create a separate report
- For each problem found the evaluator should give a severity rating (helps to set priorities)

Questionnaires and Interviews

- Lot of information available in psychology, communication studies, market research
- …here is just a quick overview

- Process to get the "right" questions
  - Brainstorm (within the project/design team) on issues that are relevant and should be put to the users
  - Select the set of relevant questions (make the size appropriate, don't ask questions you are not interested in)
  - Create a first version of the questionnaires or interviews
  - Run a few pilot interviews/questionnaires
  - Discuss the answers/results given – did participants understand what you wanted to ask them?
  - Potentially redesign the questionnaires or interviews
  - Run the interviews/questionnaires

Interviews

- Find out about users viewpoint
- Level of detail is not predetermined
- Allows more explanation and going into detail
- Open ended questions
- Good for exploration
- Often very dependent on the interviewer

- How to interview
  - Prepare a set of questions (core set for some consistency)
  - Ask question neutral and do not imply answers
    - "what is your opinion on the audio feedback" vs.
      "did you think the use of the audio feedback was really helpful"
- Group interviews
  - More discussion style
  - Finding a consensus
  - Often only the opinion of a few people in the group

Interviews

- Recognize the users response

- Problem
  - Time consuming
  - Interviewer can "steer" the outcome

- Examples
  - Retrospective interview after a test session
    - Show video recording and ask questions
    - Ask questions to clarify situations
  - Critical incident interviews
    - Ask about critical situation related to the software product
    - Rare events that may still be important

Questionnaires/Surveys

- To reach larger groups
- Initial effort may be large (creating the questionnaire and the analysis function)
- Creating them online (or at least machine readable) saves time
- Little effort per participant after the questionnaire is created
- Good for statistical analysis of results

… however if the questions are not good or the participants responding are the wrong ones the results may be poor

Questionnaires/Surveys

- How to create a questionnaire
  - Find out what the information is that you are you interested in
  - What should be analyzed and how should it be analyzed
  - What will the results be used for (e.g. redesign, new product, new features)
- Who is the audience
  - Specify the audience for questionnaire
  - How will representative participants be found
- What technology / approach will be used
  - Online / Webpage
  - Software
  - Paper

Style of Questions

- General
  - Explorative
  - Establish background
- Open ended questions
  - Set of answers are not pre-determined
  - Ask for opinion or subjective general comments
  - E.g. “what would you like to have different change on this web page”
  - Very hard to analyze automatically
- Closed questions
  - Types
    - Scalar
    - Ranked
    - Alternatives
    - Multiple choice
  - Response is restricted to alternatives
  - can be easily analyzed

- sometimes combined
  - “how did you hear about us? – TV, Radio, Google, other _____”

Closed Questions be specific

- Minimize interpretation for responses!
  - alternative answers should be very specific

how often do you use computers at work:
O frequently
O sometimes
O rarely

vs

how often do you use computers at work on a typical work day
O more than 6 hours a day
O between 1 and 6 hours a day
O less than 1 hr a day

- For closed questions you must cover all sensible answers
- Watch the language (clear, avoid jargon)

Question Formats
Scalar

- Odd number allow neutral value
  I found the audio feedback annoying
  Disagree 1 2 3 4 5 Agree
- even number forces a choice
  I found the audio feedback annoying
  Disagree 1 2 3 4 5 6 Agree

- Likert scale. 1-to-5 (or 1-7, 1-9)
  1. strongly disagree
  2. disagree
  3. undecided
  4. agree
  5. strongly agree

Question Formats
Ranked

- As participants to rank options
- Example

  What method did you use most often to print the document?
  Please rank 1=most often, 2=middle, 3=least often

  Keyboard [3] Toolbar [1] Menu [2]

- Forces a choice on the participants

Question Formats
Alternatives & Multi Choice

- Alternatives
  - Give different options – but only one can be selected
  - Example
    what is your preferred way for electronic communication?
    [X] Email
    [ ] Fax
    [ ] SMS
    [ ] Video conferencing
- Multiple choice
  - Give different options – allow to select multiple of them
  - Example
    what forms of electronic communication have you use in the last 6 weeks?
    [X] Email
    [X] Fax
    [X] SMS
    [X] Video conferencing

References

- Alan Dix, Janet Finlay, Gregory Abowd and Russell Beale. (1998) Human Computer, Interaction (second edition), Prentice Hall, ISBN 0132398648 (new Edition announced for October 2003)
- Ben Shneiderman. (1998) Designing the User Interface, 3rd Ed., Addison Wesley; ISBN: 0201694972