Vorlesung Mensch-Maschine-Interaktion

Methods & Tools

Ludwig-Maximilians-Universität München
LFE Medieninformatik
Heinrich Hußmann & Albrecht Schmidt
WS2003/2004
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

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

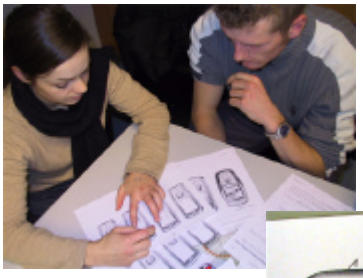
- Users may be wrong
- Users may be resistant to change
- Users may expect disadvantages (e.g. being replaced by software)

- 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)

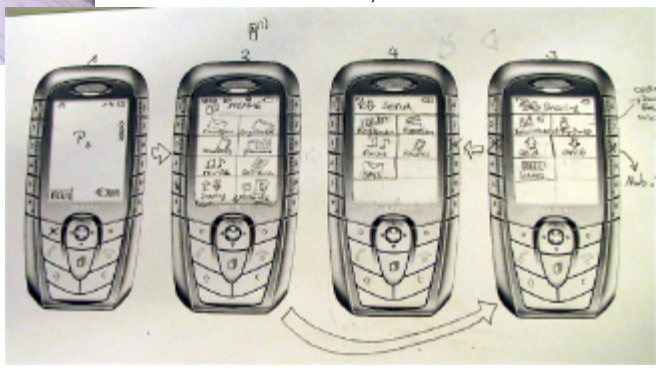
Design Cycles & Prototyping

- 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

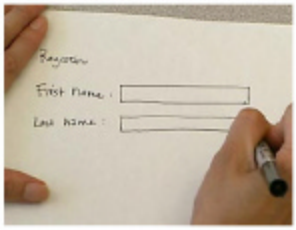
- Without paper prototyping:
  - Idea – sketch – implementation – evaluation

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

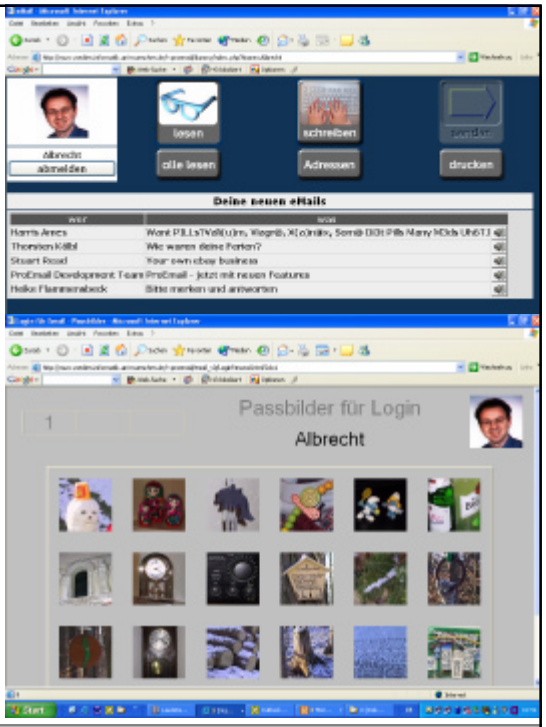
- Demonstrate a selected feature of a product
- Allows the user only to use this specific function
- The details of the function/feature are shown/implemented
- Helps to evaluate/test
  - The optimal design for a particular function
  - Optimize the usability of this function
  - User performance for 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

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

Workshop:

Benutzerschnittstellen und Bedienkonzepte für Leseschwache



- Gemeinsam mit Sonderpädagogik
- Prototyping
- Beispiel E-Mailanwendung

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
- A. Cooper. About Face 2.0