

Goal-Directed Design

Jörg Cassens

Contextual Design of Interactive Systems



1 Introduction

Assignment 3.1: Pruitt & Grudin, Chapman & Milham

- Required reading for week 2
 - Pruitt, John, and Jonathan Grudin. “Personas: practice and theory.” In Proceedings of the 2003 conference on Designing for user experiences, ACM, 2003.
 - Chapman, Christopher N., and Russell P. Milham. “The personas’ new clothes: methodological and practical arguments against a popular method.” In Proceedings of the human factors and ergonomics society annual meeting, vol. 50, no. 5, pp. 634-636. Sage Publications: Los Angeles, CA, 2006.
- The texts will be discussed in the tutorial 30.04.2019
- Course readings can be downloaded in the learnweb
- Every text has a wiki-page in the learnweb
 - Use it to describe the text
 - Use it to link the text to the course
- Results of the discussion may also be written up

2 User Goals

User Goals

- What are user goals?
- How can we identify them?
- How do we know that they are real goals, rather than tasks users are forced to perform by poorly designed tools or business processes?
- Are they the same for all users? Do they change over time?
- Often quite different from what we might guess them to be
 - Accountant clerk’s goal is probably not to process invoices efficiently
 - More probably appearing competent at his job and keeping himself engaged with his work while performing routine and repetitive tasks

Business Goals

- Most of us share these simple, personal goals
- Even if we have higher aspirations, they are still more personal than work-related: winning a promotion, learning more about our field
- Products designed and built to achieve business goals alone will eventually fail
- When the design meets the user's personal goals, business goals are achieved far more effectively
- User interfaces fail to meet user goals with alarming frequency
 - Make users feel stupid
 - Cause users to make big mistakes
 - Require too much effort to operate effectively
 - Don't provide an engaging or enjoyable experience

Assignment 2.1: D. Norman

- Required reading for week 1
 - Norman, Donald A. "Human-centered design considered harmful." *interactions* 12, no. 4 (2005): 14-19.
- The text will be discussed in the tutorial 16.04.2019
- Course readings can be downloaded in the learnweb
- Every text has a wiki-page in the learnweb
 - Use it to describe the text
 - Use it to link the text to the course
- Results of the discussion may also be written up

Goals & Activities

- Goals are not the same as tasks or activities
- Goal expectation of an end condition
- Activities and tasks are intermediate steps
- Donald Norman describes his extended hierarchy based on Activity Theory
 - **Activity:** Coordinated, integrated set of tasks. For example, staying at a hotel.
 - **Tasks:** An individual task is for example to check into the hotel.
 - **Actions:** Tasks consist of collections of actions. An action is performed consciously, the hotel check-in, for example, consists of actions like presenting the reservation, confirmation of room types, and handover of keys.
 - **Operations:** Actions consist themselves of collections of non-conscious operations. Writing your name on a sheet of paper or taking the keys are operations.
- Activity-Centred Design focuses on the activity

Reminder: Hierarchical Structure of Activity

- **Activity:** An individual activity is for example to check into a hotel. Individual activities can be part of collective activities, e.g. when someone organises a workshop.
- **Actions:** Activities consist of collections of actions. An action is performed consciously, the hotel check-in, for example, consists of actions like presenting the reservation, confirmation of room types, and handover of keys.
- **Operations:** Actions consist themselves of collections of non-conscious operations. Writing your name on a sheet of paper or taking the keys are operations.

This hierarchical composition is not fixed over time. If an action fails, the operations can get conceptualised (*breakdown*).

Goals

- Activities useful in breaking down the “what” of user behaviors, but it really does not address the first question any designer should ask: *Why* is a user performing an activity, task, action, or operation in the first place?
- Goals motivate people to perform activities; understanding goals allows you to understand your users’ expectations and aspirations, which in turn can help you decide which activities are truly relevant to your design
- Asking, “What are the user’s goals?” lets you understand the meaning of activities to your users
- Activities and tasks are much more transient, because they are based almost entirely on whatever technology is at hand
- In our example, the goal of with the hotel stay is for it to be comfortable, safe and affordable

Meet Goals in Context

- Let us assume that making user interfaces and product interactions easier to learn should always be a design target
- Ease of learning is an important guideline, but in reality, the design target really depends on the context - who the users are, what they are doing, and their goals
- You simply cannot create good design by following rules disconnected from the goals and needs of the users of your product
- Example:
 - automated call-distribution system
 - people who use this product are paid based on how many calls they handle
 - most important concern is not ease of learning, but the efficiency and rapidity of call handling
 - If the product is a kiosk in a corporate lobby helping visitors find their way around, ease of use for first-time users is clearly a major goal

Good Design

What is good design?

Good design makes helps users to effectively accomplish tasks related to their goals, fulfil them efficiently and be satisfied in doing so

- Software that enables users to perform their tasks without addressing their goals rarely helps them be truly effective
- If the task is to enter 5,000 names and addresses into a database, a smoothly functioning data-entry application will not satisfy the user nearly as much as an automated system that extracts the names from the invoicing system
- The user’s job is to focus on her tasks, the designer’s job is to look beyond that

3 Models

Models in Human-Computer Interaction

- Model of a system describes how it works
 - its constituent parts and how they work together to do what the system does
- We are here concerned with three models:
 - The **system model** (sometimes called implementation model) is how the system actually works.
 - The **interface model** (or represented model) is the model that the system presents to the user.
 - The **user model** (or conceptual model) is how the user thinks the system works.
- There are more models
 - The model the developers have about how they think the user model is like
 - The model the system has about the user (inscribed, in terms of Actor Network Theory)

System Model

- Pixel editing *vs.* structured graphics
 - pixel based as in Gimp, Photoshop
 - vector based as in Inkscape, Visio
- Text file as single string *vs.* list of lines
 - End of line as a normal character, like in Emacs
 - End of line as a special character, like in vi
- Asset Management system
 - List of assets in a flat file
 - Records of assets in a database
- It is much easier to design software that reflects its implementation model
- From the developer's perspective, it is logical to provide a button for every function, a field for every data input, a page for every transaction step, and a dialog box for every code< module

User Model

- Electricity as water
 - Electricity moves through wires like water through pipes
 - Plug it in, use the gadget (water the flowers)
- Thermostat as a valve
 - A thermostat needs to be opened fully to get as much heat (water) out as quickly as possible
- Mobile phone like a landline
 - A mobile phone is a radio transceiver that might swap connections between a half-dozen different cellular base antennas in the course of a two-minute call
 - This knowledge does not help us how to use a phone

User Models may be Wrong

- Sometimes harmless
 - Electricity as water
 - * The power cord is no water hose, electrons don't move fast because of "electrical friction"
- Sometimes misleading
 - Thermostat as a valve
 - * What is the fastest way to heat up a room? Fully opening the thermostat, like a valve?
 - * A thermostat is basically an on/off switch: full power till warm enough
- Ignorant user?
 - The heater in a car actually much more like a valve
 - A dimmer is working as a thermostat: set it to the desired lighting conditions
 - Problem: the thermostat does not effectively communicate its model to the user, in particular, there is not enough feedback

Interface Model

- Help system
 - as a binder
 - as a paper clip
- Discussion spaces
 - as a forum
 - * threaded
 - * linear
 - as a wiki
- On a more general level: learning management systems
 - among peers (kitchen party)
 - with leaders (lectures)

Interface models can be built on different levels (activity design, information design, interaction design) and should connect to existing knowledge (metaphors)

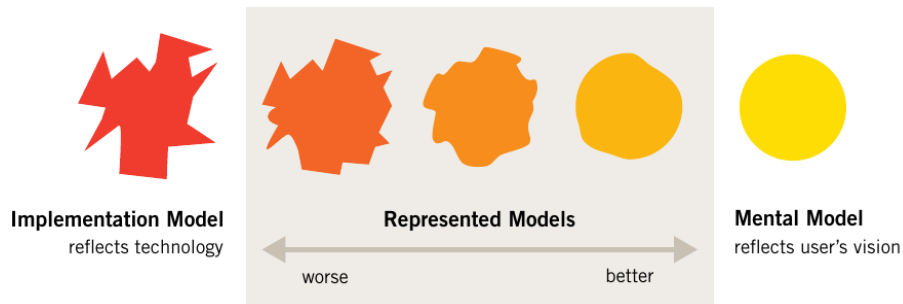
Interface Model Hides System Model

- The interface model should be:
 - Simple
 - Appropriate: reflect user's model of the task (learned from task analysis)
 - Well-communicated
 - Usable
 - * Efficient
 - * Effective
 - * Satisfying
- Implementation model does not have to be exposed
 - Text editor can store list of lines, expose a continuous text
 - A cell phone is not a wired phone, still it acts like one – no need to show things like handover between base stations
 - The interface model should closely reflect the user's model of the actual task

Striving for Perfection

- ability to represent the computer's functioning independent of its true actions is far more pronounced in software than in any other medium
- It allows a clever designer to hide some of the more unsavory facts of how the software really gets the job done
- An application's represented model can (and often should) be quite different from an application's actual processing structure
 - For example, an operating system can make a network file server look as though it were a local disk
- The closer the represented model comes to the user's mental model, the easier he will find the application to use and understand
- One of the most significant ways in which computers can assist human beings is presenting complex data and operations in a simple, easily understandable form
- User interfaces that are consistent with users' mental models are vastly superior to those that are merely reflections of the implementation model

Choice of Represented Model



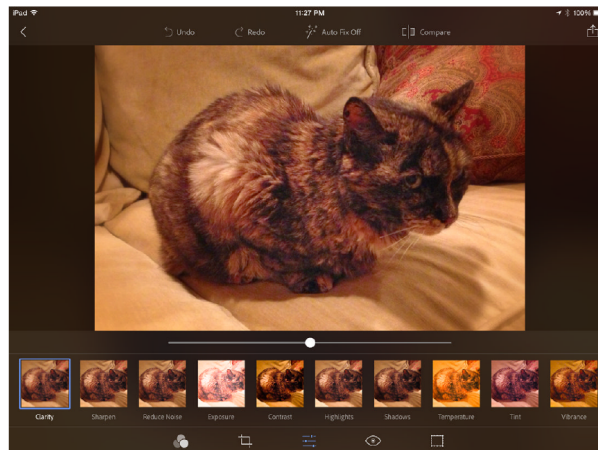
A comparison of the implementation model, mental model, and represented model.

[Cooper et al., 2014]

Simplification

- We tend to form mental models that are simpler than reality
- If we create represented models that are simpler than the implementation model, we help the user achieve better understanding
- In software, we imagine that a spreadsheet scrolls new cells into view when we click the scrollbar
- Nothing of the sort actually happens – There is no sheet of cells out there, but a tightly packed data structure of values, with various pointers between them, from which the application synthesizes a new image to display in real time.
- One of the most significant ways in which computers can assist human beings is presenting complex data and operations in a simple, easily understandable form
- As a result, user interfaces that are consistent with users' mental models are vastly superior to those that are merely reflections of the implementation model.

Example: Goal-Directed UI



Adobe Photoshop Express for iPad - what the user wants to achieve dominates interface, not sliders.

[Cooper et al., 2014]

4 Goal-Directed Design

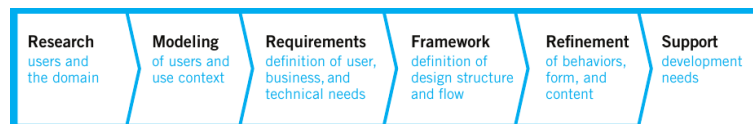
Justification

- Few design methods in common use today incorporate a means of effectively and systematically translating the knowledge gathered during research into a detailed design specification
- Rather than providing information about user goals, most methods provide information at the task level

- This type of information is useful for defining layout, work flow, and translation of functions into interface controls
- It is less useful for defining the basic framework of what a product is, what it does, and how it should meet the user's broad needs
- Need for explicit, systematic process to bridge the gap between research and design for defining user models, establishing design requirements, and translating those into a high-level interaction framework
- Goal-Directed Design seeks to bridge the gap that currently exists in the digital product development process - the gap between user research and design

Overview

- Goal-Directed Design combines techniques of ethnography, stakeholder interviews, market research, detailed user models, scenario-based design, and a core set of interaction principles and patterns
- It provides solutions that meet users' needs and goals while also addressing business/organizational and technical imperatives
- Can be roughly divided into six phases:
 - Research, Modeling, Requirements Definition, Framework Definition, Refinement, and Support



[Cooper et al., 2014]

Research

- Employs ethnographic field study techniques (observation and contextual interviews) to provide qualitative data about potential and/or actual users of the product
- Includes competitive product audits as well as reviews of market research, technology white papers, and brand strategy
- Includes one-on-one interviews with stakeholders, developers, subject matter experts (SMEs), and technology experts as suits the particular domain
- One principal outcome of field observation and user interviews is an emergent set of behavior patterns - identifiable behaviors that help categorize modes of use of a potential or existing product
- These patterns suggest goals and motivations (specific and general desired outcomes of using the product)
- Behavior patterns and goals associated with them drive creation of personas
- Market research helps select personas that fit business models

Modeling

- Behavior and work flow patterns discovered by analyzing the field research and interviews are synthesized into domain and user models
- Domain models can include information flow and work flow diagrams
- User models, or personas, are detailed, composite user archetypes that represent distinct groupings of behaviors, attitudes, aptitudes, goals, and motivations observed and identified during the Research phase
- Personas are the main characters in a narrative, scenario-based approach
- Iteratively generating design concepts in the Framework Definition phase
- Powerful communication tool that helps developers and managers understand design rationale and prioritize features based on user needs
- Synthesize, differentiate, and prioritize personas, exploring different types of goals and mapping personas across ranges of behavior to ensure that no gaps or duplications exist

Requirements Definition

- Scenario-based design methods, focusing the scenarios on meeting the goals and needs of specific user personas
- Personas help us understand which tasks are truly important and why, leading to an interface that minimizes necessary tasks (effort) while maximizing return
- Personas become the main characters of these scenarios
- Accomplished through an iteratively refined context scenario
 - A “day in the life” of the persona using the product, describing high-level product touch points, and thereafter successively defining detail at ever-deepening levels
 - In addition, designers consider the personas’ skills and physical capabilities as well as issues related to the usage environment
- The output of this process is a requirements definition that balances user, business, and technical requirements of the design to follow

Framework Definition

- Creating the overall product concept, defining the basic frameworks for the product’s behavior, visual design, and, if applicable, physical form
- Interaction design teams synthesize an interaction framework by employing two other critical methodological tools in conjunction with context scenarios
 - first: a set of general interaction design principles that provide guidance in determining appropriate system behavior
 - second: a set of interaction design patterns that encode general solutions
- After data and functional needs are described at this high level, they are translated into design elements according to interaction principles and then organized, using patterns and principles, into design sketches and behavior descriptions
- Output is an interaction framework definition, a stable design concept that provides the logical and high-level formal structure for the detail to come
- Successive iterations of more narrowly focused scenarios provide this detail in the Refinement phase

Refinement

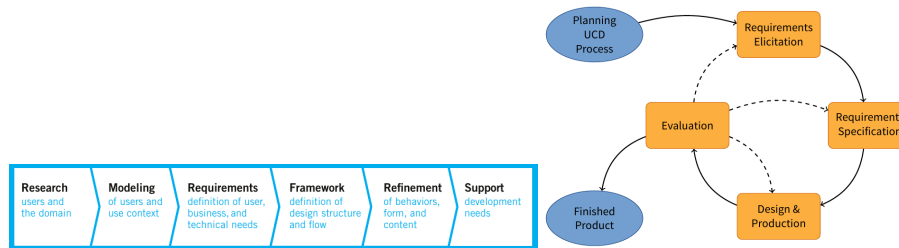
- Proceeds similarly to the Framework Definition phase, but with increasing focus on detail and implementation
- Interaction designers focus on task coherence, using scenario walkthroughs and validation scenarios focused on storyboarding paths through the interface in great detail
- Visual designers define a system of type styles and sizes, icons, and other visual elements that provide a compelling experience with clear affordances and visual hierarchy
- Industrial designers, when appropriate, finalize materials and work closely with engineers on assembly schemes and other technical issues
- Culmination is the detailed documentation of the design - a form and behavior specification or blueprint, delivered in either paper or interactive media form

Support

- Even a very well-conceived and validated design solution cannot possibly anticipate every development challenge and technical question
- Important to be available to answer developers’ questions as they arise during the construction process
- As the development team prioritizes their work and makes trade-offs to meet deadlines, the design must be adjusted, requiring scaled-down design solutions
- If the interaction design team is not available to create these solutions, developers are forced to do this under time pressure, which has the potential to gravely compromise the integrity of the design.

Assignment 4.1: Goal-Directed Design as Human-Centred Process?

- Form groups of 3-6
- Discuss whether this design process can be considered an instantiation of human-centred design
 - What fits, what does not
- Report and discuss your findings in the course



References

Literatur

- Chapman, C. N. and Milham, R. P. (2006). The personas' new clothes: methodological and practical arguments against a popular method. In *Proceedings of the human factors and ergonomics society annual meeting*, volume 50, pages 634–636. SAGE Publications Sage CA: Los Angeles, CA.
- Cooper, A., Reimann, R., Cronin, D., and Noessel, C. (2014). *About Face (fourth edition): the essentials of interaction design*. John Wiley & Sons.
- Norman, D. A. (2005). Human-centered design considered harmful. *interactions*, 12(4):14–19.
- Pruitt, J. and Grudin, J. (2003). Personas: practice and theory. In *Proceedings of the 2003 conference on Designing for user experiences*, pages 1–15. ACM.