CHAPTER 9: THE PROCESS OF INTERACTION DESIGN

CHAPTER 10: ESTABLISHING REQUIREMENTS
Nog even n.a.v. vorige week:
N.a.v. lab bezoek Rabobank: poor man’s options:
What is involved in Interaction Design?

ID is all about human actors
What is involved in Interaction Design?

- It is a process:
  - a goal-directed problem solving activity informed by intended use, target domain, materials, cost, and feasibility
  - a creative activity
  - a decision-making activity to balance trade-offs

- Generating alternatives and choosing between them is key

- Four approaches: user-centered design, activity-centered design, systems design, and genius design
What is involved in Interaction Design?

Four basic activities

1. Establishing requirements
2. Designing alternatives
3. Prototyping
4. Evaluating
What is involved in Interaction Design?

• A user-centered approach
• Four basic activities
• Importance of involving users
• Degrees of user involvement

Practical issues
• Who are the users?
• What are ‘needs’?
• Where do alternatives come from?
• How to choose among alternatives?
• How to integrate interaction design activities in other lifecycle models?
(What is) a user-centered approach?

User-centered approach is based on:

• Early focus on users and tasks: directly studying cognitive, behavioral, attitudinal & other characteristics

• Empirical measurement: users’ reactions and performance to scenarios, manuals, simulations & prototypes are observed, recorded and analysed

• Iterative design: when problems are found in user testing, fix them and carry out more tests
Approach 1: user-centered design

- Focus on User Needs and GOALS. Designer is translator of user needs and goals.
- Users guide the product decisions.
Approach 2: activity-centered design

- Do not focus on the goals and preferences of the user, but on the activity a user would perform with a given piece of technology.
- Inspired by activity theory
- Research to get insights of the users (observations, interviews)
- Mapping users' activities and tasks
- Users are the performers of activities. Role of the designer is to provide tools to accomplish actions.
Approach 3: systems design

- Structured, rigorous, focusing on context
- Particularly fit for complex problems
- Most appropriate for projects involving large systems or systems of systems
- Solo designers working on small projects may find the same tools a bit cumbersome for their needs
- Entirely compatible with a user-centered approach. The core of both approaches is understanding user goals.
Approach 3: systems design

- For this situation, what is the system?
- What is the environment?
- What goal does the system have in relation to its environment?
- What is the feedback loop by which the system corrects its actions?
- How does the system measure whether it has achieved its goal?
- Who defines the system, environment, goal, etc. and monitors it?
- What resources does the system have for maintaining the relationship it desires?
- Are its resources sufficient to meet its purpose?
Approach 4: Genius Design?

Needed:
• vast previous experience of the team members
• luck?

Jim Leftwich:
• Rapid expert design
Importance of involving users

Expectation management

• Realistic expectations
• No surprises, no disappointments
• Timely training
• Communication, but no hype

Ownership

• Make the users active stakeholders
• More likely to forgive or accept problems
• Can make a big difference to acceptance and success of product
Degrees of user involvement

Member of the design team

- Full time: constant input, but lose touch with users
- Part time: patchy input, and very stressful
- Short term: inconsistent across project life
- Long term: consistent, but lose touch with users

Newsletters and other dissemination devices

- Reach wider selection of users
- Need communication both ways

- User involvement after product is released
- Combination of these approaches
Degrees of user involvement?

CITIZEN MEDIA project

16 partners, 3 public testbeds
30% non-technical user research
50% academic

www.ist-citizenmedia.org
Degrees of user involvement?
A simple interaction design lifecycle model

Figure 9.3 A simple interaction design lifecycle model
Some practical issues

• Who are the users?
• What do we mean by ‘needs’?
• Where do alternatives come from
• How to choose among alternatives
• How to integrate interaction design activities with other lifecycle models?
Who are the users/stakeholders?

• Not as obvious as you think:
  • those who interact directly with the product
  • those who manage direct users
  • those who receive output from the product
  • those who make the purchasing decision
  • those who use competitor’s products

• Three categories of user (Eason, 1987):
  • primary: frequent hands-on
  • secondary: occasional or via someone else
  • tertiary: affected by its introduction, or will influence its purchase
Who are the stakeholders?

- Suppliers
- Local shop owners

Managers and owners

Check-out operators

Customers
What do we mean by ‘needs’?

- Users rarely know what is possible
What do we mean by ‘needs’?

• Users can’t tell you what they ‘need’ to help them achieve their goals

• Instead, look at existing tasks:
  – their context
  – what information do they require?
  – who collaborates to achieve the task?
  – why is the task achieved the way it is?

• Envisioned tasks:
  – can be rooted in existing behaviour
  – can be described as future scenarios
How to generate alternatives

• Humans stick to what they know works

• But considering alternatives is important to ‘break out of the box’

• Designers are trained to consider alternatives, software people generally are not

• How do you generate alternatives?
  • ‘Flair and creativity’: research and synthesis
  • Seek inspiration: look at similar products or look at very different products
How to generate alternatives: copying, stealing?

The most successful portable media player in the world

iPod

The click-wheel browsing interface was ripped from Creative's NOMAD II Jukebox, which debuted over a year before iPod.

74% peak market share in '05

Kane Kramer's IXI DAP sketch from 1979

Apple admitted stealing the iPod concept from Britain's Kane Kramer, who invented a similar device in 1979, but Apple refused to pay him anything for it.
How to choose among alternatives

• Evaluation with users or with peers, e.g. prototypes

• Technical feasibility: some not possible

• Quality thresholds: Usability goals lead to usability criteria set early on and check regularly
  • safety: how safe?
  • utility: which functions are superfluous?
  • effectiveness: appropriate support? task coverage, information available
  • efficiency: performance measurements
  • learnability: is the time taken to learn a function acceptable to the users?
  • memorability: can infrequent users remember how to achieve their goal?
Testing prototypes to choose among alternatives
Yay was ultimately rejected because “it was not universally understood
All these models....?
How to integrate interaction design in other models

• Integrating interaction design activities in lifecycle models from other disciplines needs careful planning

• Several software engineering lifecycle models have been considered

• Integrating with agile software development is promising
  • it stresses the importance of iteration
  • it champions early and regular feedback
  • it handles emergent requirements
  • it aims to strike a balance between flexibility and structure
Agile
Agile: the manifesto

Manifesto for Agile Software Development

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan
• Design is too important to leave it to designers.

• Development is too important to leave it to developers.
Chapter 10
ESTABLISHING REQUIREMENTS
Overview

• The importance of requirements
• Different types of requirements
• Data gathering for requirements
• Data analysis and presentation
• Task description: Scenarios
  
  Use Cases
  
  Essential use cases
• Task analysis: HTA
What, how and why?

What needs to be achieved?

1. Understand as much as possible about users, task, context
2. Produce a stable set of requirements

How can this be done?

- Data gathering activities
- Data analysis activities
- Expression as ‘requirements’
- All of this is iterative
What, how and why?

Getting requirements right is crucial
What, how and why?

Requirements definition is the stage where failure occurs most commonly.
Establishing requirements

• What do users want? What do users ‘need’?

  Requirements need clarification, refinement, completion, re-scoping

  Input: Requirements document (maybe)
  Output: stable requirements

• Why ‘establish’?

  Requirements arise from understanding users’ needs
  Requirements can be justified & related to data
Tools: Volere shell

Requirement #: 75  Requirement Type: 9  Event/use case #: 6

Description: The product shall issue an alert if a weather station fails to transmit readings.

Rationale: Failure to transmit readings might indicate that the weather station is faulty and needs maintenance, and that the data used to predict freezing roads may be incomplete.

Source: Road Engineers
Fit Criterion: For each weather station the product shall communicate to the user when the recorded number of each type of reading per hour is not within the manufacturer’s specified range of the expected number of readings per hour.

Customer Satisfaction: 3  Customer Dissatisfaction: 5  Conflicts: None

Dependencies: None
Supporting Materials: Specification of Rosa Weather Station
History: Raised by GBS, 28 July 99
Tools (fancier)
De behoefte aan validatie: een verkennend onderzoek naar requirements traceability in Agile Development

Bachelorthesis Informatiekunde
Different kinds of requirements

• Functional:
  What the system should do

• Non-functional: security, response time...)

• Data:
  • What kinds of data need to be stored?
  • How will they be stored (e.g. database)?
Different kinds of requirements

Environment or context of use:

• physical: dusty? noisy? vibration? light? heat? humidity? …. (e.g. ATM)

• social: sharing of files, of displays, in paper, across great distances, synchronous, privacy for clients

• organisational: hierarchy, IT department’s attitude and remit, user support, communications structure and infrastructure, availability of training
Prioritizing requirements: MoSCoW

Must have
non negotiable” without which project is a complete failure

Should have
important but not critical to the success of the project

Could have
often described as features that are nice to have e.g colourful user interface

Won't have
are not included in the in the schedule, sometimes dropped or reconsidered whether to be included or not
Underwater computing

**Figure 10.2 (a)** The components of WetPC’s underwater computer.

Underwater computing

Figure 10.3 (a) The KordGrip Interface and (b) the KordGrip in use underwater
Source: (a) Reproduced by permission of WetPC Pty Ltd (b) Reproduced by permission of the Australian Institute of Marine Science.
Different kinds of requirements

Users: Who are they?

- Characteristics: nationality, educational background, attitude to computers
- System use: novice, expert, casual, frequent
  - Novice: prompted, constrained, clear
  - Expert: flexibility, access/power
  - Frequent: short cuts
  - Casual/infrequent: clear menu paths
What are the users’ capabilities?

Humans vary in many dimensions:

- size of hands may affect the size and positioning of input buttons
- motor abilities may affect the suitability of certain input and output devices
- height if designing a physical kiosk
- strength - a child’s toy requires little strength to operate, but greater strength to change batteries
- disabilities (e.g. sight, hearing, dexterity)
Personas

bekend van opdracht 2 (en de meesten OIS)

- Capture a set of user characteristics (user profile)
- Not real people, but synthesised from real users
- Should not be idealised
- Bring them to life with a name, characteristics, goals, personal background
- Develop a small set of personas with one primary
At the heart?
Example Persona

**Ginnie**

**Background**
- 15, Female
- Ongoing Private Education
- Ambitious
- Comfortable using technology to communicate

**Motivations**
- Keeping in touch with her network
- Fashion/trendy girl
- Keeping up with pop culture

**Frustrations**
- Sad people trying to be 'friends' on Facebook
- having to be in bed at 11pm
- Being swamped in friends updates
- Missing important event updates

Receives private tutoring in Maths and English as these are not her strong subjects. Enjoys playing for the school’s 2nd teams for netball and Lacrosse and is good at art.

She loves recording her favourite High on Sky+ and spends some that Daddy bought her watching download music, keeping up Facebook and chatting via MSN while studying.

She loves Eiffel Tower and Arc de Triomphe.

In this image, she is in a coffee shop.

---

**Noah**

**Interesting as:**
- Herkenbare beelden
- Alle zintuigen geprikkeld worden
- Inspirerende gedachten

Noah houdt er van om verhalen te maken.

Door woorden en beelden probeert zij momenten vast te leggen.

Zij vindt het leuk om naar de wereld om zich heen te kijken en zich voor te stellen hoe het er vroeger uit zag. Als ze door de stad wandelt probeert zij zich voor te stellen hoe het er uit zag toen er nog geen auto's geparkeerd stonden. In gedachten ziet zij een paardentram voorbij komen en ruikt ze de vieze lucht van het afval en de uitwerpselen op straat.

Ze houdt er van om historische foto’s te bekijken en fantaseert er zelf verhalen bij.

---

![Image of Noah](image.png)
Data gathering for requirements

Partly also already elaborated on in chapter 7

Interviews:
- sample scenarios of use, prototypes, can be used in interviews
- Good for exploring issues
- Development team members can connect with stakeholders

Focus groups:
- Group interviews
- Good at gaining a consensus view and/or highlighting areas of conflict
- But can be dominated by individuals
Data gathering for requirements

Partly also already elaborated on in chapter 7

Questionnaires:

• Often used in conjunction with other techniques
• Can give quantitative or qualitative data
• Good for answering specific questions from a large, dispersed group of people

Researching similar products:

• Good for prompting requirements
Data gathering for requirements

Partly also already elaborated on in chapter 7

Direct observation:
- Gain insights into stakeholders’ tasks
- Good for understanding the nature and context of the tasks
- But, it requires time and commitment from a member of the design team, and it can result in a huge amount of data

Indirect observation:
- Not often used in requirements activity
- Good for logging current tasks
Data gathering for requirements

Studying documentation:

- Procedures and rules are often written down in manuals
- Good source of data about the steps involved in an activity, and any regulations governing a task
- Not to be used in isolation
- Good for understanding legislation, and getting background information
- No stakeholder time, which is a limiting factor on the other techniques
Considerations for data gathering (1)

• Identifying and involving stakeholders: users, managers, developers, customer reps?, union reps?, shareholders?

• Involving stakeholders: workshops, interviews, workplace studies, co-opt stakeholders onto the development team

• ‘Real’ users, not managers

• Political problems within the organisation

• Dominance of certain stakeholders

• Economic and business environment changes

• Balancing functional and usability demands
Considerations for data gathering (2)

• Requirements management: version control, ownership

• Communication between parties:
  • within development team
  • with customer/user
  • between users… different parts of an organisation use different terminology

• Domain knowledge distributed and implicit:
  • difficult to dig up and understand
  • knowledge articulation: how do you walk?

• Availability of key people
Data gathering guidelines

- Focus on identifying the stakeholders’ needs
- Involve all the stakeholder groups
- Involve more than one representative from each stakeholder group
- Use a combination of data gathering techniques
- Support the process with prototypes and task descriptions
Data interpretation and analysis

• Start soon after data gathering session

• Initial interpretation before deeper analysis

• Different approaches emphasize different elements e.g. class diagrams for object-oriented systems, entity-relationship diagrams for data intensive systems
Task descriptions

• Stories: a natural way for people to explain things

• Scenarios
  • an informal narrative story, simple, ‘natural’, personal, not generalisable

• Use cases
  • assume interaction with a system, step by step
  • assume detailed understanding of the interaction
  • show actions that are required to reach a goal

• Essential use cases
  • abstract away from the details
  • does not have the same assumptions as use cases
Task descriptions (not in book)

- User stories from the perspective of the end user, what the user **wants to do** simple in structure:

  As a runner I **want to** track the miles I run each day **so that** I can have a good understanding of how much I have exercised.

![Diagram of user story structure](image)
Scenarios and Personas

Figure 10.10 The relationship between a scenario and its associated persona
Source: http://www.smashingmagazine.com/2014/08/06/a-closer-look-at-personas-part-1/
Example use case diagram for travel organizer
“The Thomson family enjoy outdoor activities and want to try their hand at sailing this year. There are four family members: Sky (10 years old), Eamonn (15 years old), Claire (35), and Will (40). One evening after dinner they decide to start exploring the possibilities. They all gather around the travel organizer and enter their initial set of requirements – a sailing trip for four novices in the Mediterranean. The console is designed so that all members of the family can interact easily and comfortably with it. The system’s initial suggestion is a flotilla, where several crews (with various levels of experience) sail together on separate boats. Sky and Eamonn aren’t very happy at the idea of going on vacation with a group of other people, even though the Thomsons would have their own boat. The travel organizer shows them descriptions of flotillas from other children their ages and they are all very positive, so eventually, everyone agrees to explore flotilla opportunities. Will confirms this recommendation and asks for detailed options. As it’s getting late, he asks for the details to be saved so everyone can consider them tomorrow. The travel organizer emails them a summary of the different options available.”
Use case for travel organizer

1. The system displays options for investigating visa and vaccination requirements.
2. The user chooses the option to find out about visa requirements.
3. The system prompts user for the name of the destination country.
4. The user enters the country’s name.
5. The system checks that the country is valid.
6. The system prompts the user for her nationality.
7. The user enters her nationality.
8. The system checks the visa requirements of the entered country for a passport holder of her nationality.
9. The system displays the visa requirements.
10. The system displays the option to print out the visa requirements.
11. The user chooses to print the requirements.
Alternative courses for travel organizer

Some alternative courses:

6. If the country name is invalid:
   6.1 The system displays an error message.
   6.2 The system returns to step 3.

8. If the nationality is invalid:
   8.1 The system displays an error message.
   8.2 The system returns to step 6.

9. If no information about visa requirements is found:
   9.1 The system displays a suitable message.
   9.2 The system returns to step 1.
<table>
<thead>
<tr>
<th>USER INTENTION</th>
<th>SYSTEM RESPONSIBILITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>find visa requirements</td>
<td>request destination and nationality</td>
</tr>
<tr>
<td>supply required information</td>
<td>obtain appropriate visa info</td>
</tr>
<tr>
<td>obtain copy of visa info</td>
<td>offer info in different formats</td>
</tr>
<tr>
<td>choose suitable format</td>
<td>provide info in chosen format</td>
</tr>
</tbody>
</table>
Task analysis

• Task descriptions are often used to envision new systems or devices

• Task analysis is used mainly to investigate an existing situation

• It is important not to focus on superficial activities
  
  • What are people trying to achieve?
  
  • Why are they trying to achieve it?
  
  • How are they going about it?

• Many techniques, the most popular is Hierarchical Task Analysis (HTA)
Hierarchical Task Analysis

- Involves breaking a task down into subtasks, then sub-sub-tasks and so on. These are grouped as plans which specify how the tasks might be performed in practice.

- HTA focuses on physical and observable actions, and includes looking at actions not related to software or an interaction device.

- Start with a user goal which is examined and the main tasks for achieving it are identified.

- Tasks are sub-divided into sub-tasks.
Example Hierarchical Task Analysis

0. In order to buy a DVD
1. locate DVD
2. add DVD to shopping basket
3. enter payment details
4. complete address
5. confirm order

plan 0: If regular user do 1-2-3-4-5.
If new user do 1-2-3-4-5.
Example Hierarchical Task Analysis (graphical)

Figure 10.15 A graphical representation of the task analysis for buying a DVD
So at last….the living persona!!
Zo meteen.....
1. Bedenk onderzoeksvraag: een vergelijking
2. Maak materiaal - 2 webpaginas, A en B: HTML
3. Host dit op eigen serverruimte
4. Richt Google Analytics en Hotjar in
5. Modificeer HTML met GA & Hotjar code snippets
6. Zoek 30 proefpersonen
7. Start experiment
8. Verzamel, preparer en analyseer data: SPSS
9. Rapportage