Identifying needs and establishing requirements

8 lecture
Adapted from chapter 10
Kristina Lapin
Overview

• The importance of requirements
• Different types of requirements
• Data gathering for requirements
• Task descriptions: Scenarios
  Use Cases
Articulate: • who users are • their key tasks

Goals:

Methods:

Products:

User and task descriptions

Task centered system design
Participatory design
User-centered design

Psychology of everyday things
User involvement
Representation & metaphors

Participatory interaction
Task scenario walk-through

low fidelity prototyping methods

Graphical screen design
Interface guidelines
Style guides

Usability testing
Heuristic evaluation

high fidelity prototyping methods

Testable prototypes

Completed designs

Alpha/beta systems or complete specification

Interfeiso projektavimas ir panaudojamumo inžinerija (Soul Greenberg)
What, how and why?

• What
  Two aims:
  1. Understand as much as possible about users, task, context
  2. Produce a stable set of requirements

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

• Why:
  Requirements definition: the stage where failure occurs most commonly

Getting requirements right is crucial
Why?

• Stable requirements - a sound basis to project success
  – Software Development Top 30 Mistakes (blog: Carrasco 2006)
    • Error #1: Not understanding the user’s needs. Lack of user input, or not even asking.
  – Software defect reduction top 10 list (Boehm, Basili 2001):
    • #1: Finding and fixing a software problem after delivery is often 100 times more expensive than finding and fixing it during the requirements and design phase.
Why?

• Stable requirements - a sound basis to project success
  – Taylor (2000) IT project: sink and swim

Figure 1: Management activities contributing to failure.
Why?

• Stable requirements - a sound basis to start designing
  – Taylor (2000) IT project: sink and swim

Figure 2: Failure stages.
Why?

- Stable requirements - a sound basis to start designing
  - Taylor (2000) IT project: sink and swim
Why?

• Stable requirements - a sound basis to start designing

![Figure 4: Critical success factors.](chart.png)
Establishing requirements

• Cross-cultural design concerns (Chavan et al, 2009)
  – Market is global, but consumers are always local
  – Product may need to be redesigned
    • Kellogg’s Corn Flakes turn into wet paper after pouring warm milk on them. Kellogg’s had to reengineer them to stand up to warm milk.
    • Whirlpool “World Washer” had slight feature design and styling modifications to reflect local tastes. Sales were well everywhere but India.

– Multinational companies
  • One global website or many local websites?
    – Coca-cola vs Pepsi
Requirements

• A statement about intended product that specifies what it should do or how it should perform.
  – Specific, unambiguous, clear
  – Specific
  • The time to download any complete page is less than 5 sec.

– Abstract
  • Teenager girls should find the site appealing
    – In this vase further investigation is required
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
Dependencies: None  Conflicts: None
Supporting Materials: Specification of Rosa Weather Station
History: Raised by GBS, 28 July 99

Copyright © Atlantic Systems Guild
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
Volere requirements template

<table>
<thead>
<tr>
<th>PROJECT DRIVERS</th>
<th>13. Operational and Environmental Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The Purpose of the Product</td>
<td>14. Maintainability and Support Requirements</td>
</tr>
<tr>
<td>2. The Stakeholders</td>
<td>15. Security Requirements</td>
</tr>
<tr>
<td>PROJECT CONSTRAINTS</td>
<td>16. Cultural and Political Requirements</td>
</tr>
<tr>
<td>3. Mandated Constraints</td>
<td>17. Legal Requirements</td>
</tr>
<tr>
<td>4. Naming Conventions and Definitions</td>
<td></td>
</tr>
<tr>
<td>5. Relevant Facts and Assumptions</td>
<td></td>
</tr>
<tr>
<td>FUNCTIONAL REQUIREMENTS</td>
<td></td>
</tr>
<tr>
<td>6. The Scope of the Work</td>
<td>18. Open Issues</td>
</tr>
<tr>
<td>8. The Scope of the Product</td>
<td>20. New Problems</td>
</tr>
<tr>
<td>9. Functional and Data Requirements</td>
<td>21. Tasks</td>
</tr>
<tr>
<td>NON-FUNCTIONAL REQUIREMENTS</td>
<td>22. Migration to the New Product</td>
</tr>
<tr>
<td>10. Look and Feel Requirements</td>
<td>23. Risks</td>
</tr>
<tr>
<td>11. Usability and Humanity Requirements</td>
<td>24. Costs</td>
</tr>
<tr>
<td></td>
<td>26. Waiting Room</td>
</tr>
<tr>
<td></td>
<td>27. Ideas for Solutions</td>
</tr>
</tbody>
</table>
Types of requirements

• Functional:
  —What the system should do
  —Historically the main focus of requirements activities

• Non-functional:
  —memory size, 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. OMS insects, ATM)
— social: sharing of files, of displays, in paper, across great distances, work individually, privacy for clients
— organisational: hierarchy, IT department’s attitude and remit, user support, communications structure and infrastructure, availability of training
An extreme example: WetPC – underwater wearable PC for Kord Pad
An extreme example: WetPC with SeaSlate and KordGrip
Types of requirements

• Users: Who are they?
  — Characteristics: ability, background, attitude to computers
  — System use: novice, expert, casual, frequent
  — Novice: step-by-step (prompted), constrained, clear information
  — Expert: flexibility, access/power
  — Frequent: short cuts
  — Casual/infrequent: clear instructions, e.g. 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

- Capture user characteristics
- Not real people, but synthesised from real user characteristics
- Should not be idealised
- Bring them to life with a name, characteristics, goals, personal background
- Develop multiple personas
Personas

Ginnie

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

Motivations
- Keeping in touch with her network
- Fashion/street cred
- Keeping up with peers.

Frustrations
- Sad people trying to be ‘friends’ on Facebook
- Having to be in bed @ 11pm
- Being swamped in friends updates
- Missing important status 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 shows: ER and San Valley High on Sky+ and spends some of her time on her Laptop that Daddy bought her watching videos on YouTube, downloading music, keeping up to date with her friends on Facebook and chatting via MS IM to her cousin who is at University in Leeds.

She loves Ugg boots and Abercrombie & Fitch and uses the Internet to shop and find the cheapest prices.

“I want to easily hook up with my friends whilst watching TV”
Kinds of requirements

What factors (environmental, user, usability) would affect the following systems?

1. An interactive product to use in a university’s self-service cafeteria that allows users to pay for their food using a credit system

2. An interactive product to control the functioning of a nuclear power plant
Data gathering for requirements

Interviews:

— Props, e.g. sample scenarios of use, prototypes, can be used in interviews
— Good for exploring issues
— But are time consuming and may be infeasible to visit everyone

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

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

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
Combining data gathering in requirement activities

Diary and interview

Dearman et al. 2008)
Combining data gathering in requirement activities

<table>
<thead>
<tr>
<th>1. Persons</th>
<th>2. Establishments &amp; Organizations</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1 Well-being</td>
<td>“How sick is my student feeling?”</td>
</tr>
<tr>
<td>1.2 Background</td>
<td>“Is my ex-girlfriend seeing anyone?”</td>
</tr>
<tr>
<td>1.3 State of an asset</td>
<td>“Did I forget to turn the TV off?”</td>
</tr>
<tr>
<td>1.4 Contact information</td>
<td>“What is my uncle’s email address?”</td>
</tr>
<tr>
<td>2.1 Properties</td>
<td>“A list of all organization members.”</td>
</tr>
<tr>
<td>2.2 Operating procedures</td>
<td>“Does Toys-R-US allow exchanges without a receipt?”</td>
</tr>
<tr>
<td>2.3 Contact information</td>
<td>“What is the phone number for EL Trompo.”</td>
</tr>
</tbody>
</table>
Mobile probe:
Mobile worker’s everyday problems

Figure 2. Technology Platform

User J
Wed, 21 Jan 2004 11:39

What kind of information do you need at the moment?

Where is my car?

Huilkii et al, 2004
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
Data gathering for requirements

• How to involve new ideas?
• Brainstorming for innovation
• How to make a brainstorming session successful?
  – Include participants for a wide range of disciplines
  – Don’t ban silly stuff
  – Use catalysts for further inspiration
  – Keep records
  – Sharpen the focus
  – Use warm-up exercises
Contextual Inquiry

• An approach to ethnographic study where user is expert, designer is apprentice
• A form of interview, but
  — at users’ workplace (workstation)
  — 2 to 3 hours long
• Four main principles:
  — Context: see workplace & what happens
  — Partnership: user and developer collaborate
  — Interpretation: observations interpreted by user and developer together
  — Focus: project focus to understand what to look for
Examples

Future Technology Workshops:
New ways of interacting with images

The Living Box: ethnographic interviews, focus groups and questionnaires
The result of contextual inquiry
The room of contextual inquiry
Problems with 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: traditionally a problem in software engineering, but better now
Problems with 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
Problems with data gathering (3)

- Political problems within the organisation
- Dominance of certain stakeholders
- Economic and business environment changes
- Balancing functional and usability demands
Some basic 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
Some basic guidelines

• Support the process with props such as prototypes and task descriptions
• Run a pilot session
• You will need to compromise on the data you collect and the analysis to be done, but before you can make sensible compromises, you need to know what you’d really like
• Consider carefully how to record the data
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
Summary

• Getting requirements right is crucial

• There are different kinds of requirement, each is significant for interaction design

• The most commonly-used techniques for data gathering are: questionnaires, interviews, focus groups, direct observation, studying documentation and researching similar products

• Scenarios, use cases and essential use cases can be used to articulate existing and envisioned work practices.

• Task analysis techniques such as HTA help to investigate existing systems and practices
References

References


• Mobilearn projekto aprašas