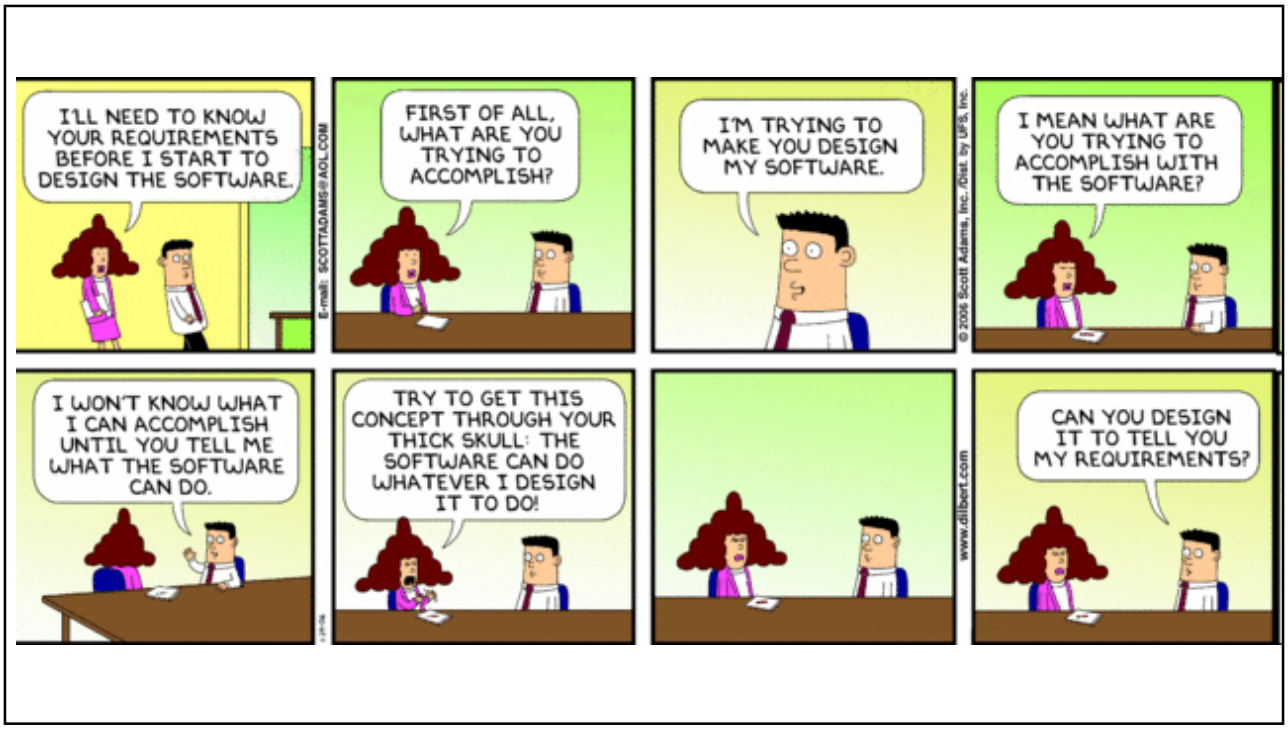


1

Tampereen yliopisto
Tampere University

Software Product Value and Stakeholders

Zheying Zhang

Day 3: Considerations when elaborating on the needs

2

Outline

Basic concepts

- Needs -> Requirements

Techniques

- Interviews
- Laddering
- Brainstorming
- Observation and enthnographic studies
- JAD
- Prototyping
- Scenarios
- Reuse
- Card sorting

Discussion

- Technique classification and selection

3

- Apple picking
- Take what you see
- More reactive less proactive

- Mushroom hunting
- Planned , deliberate search
- More proactive less reactive

4

Elaborating on Needs

The *process* through which to

Discover, reveal, articulate, and understand the problem to be solved, the services, the required performance of the product, hardware constraints, and so on

- It means to bring out, to evoke, to call forth
- It is not a spontaneous phenomenon
  - Determine sources of information and appropriate techniques
  - Get information on domain, problem, constraints

5

Tampereen yliopisto
Tampere University

Sources of inputs

Various stakeholders

- Interviews
- Observing users at work
- Meetings, workshops

Pre-existing documentation

- Documents that describe current or competing products or neighboring products
- Problem reports and enhancement requests for a system-as-is
- Marketing surveys and user questionnaires
- Standards, policies, collective agreements, etc.

External sources

- Other companies, vendors, domain experts, on-line data services, etc.

6

Tampereen yliopisto
Tampere University

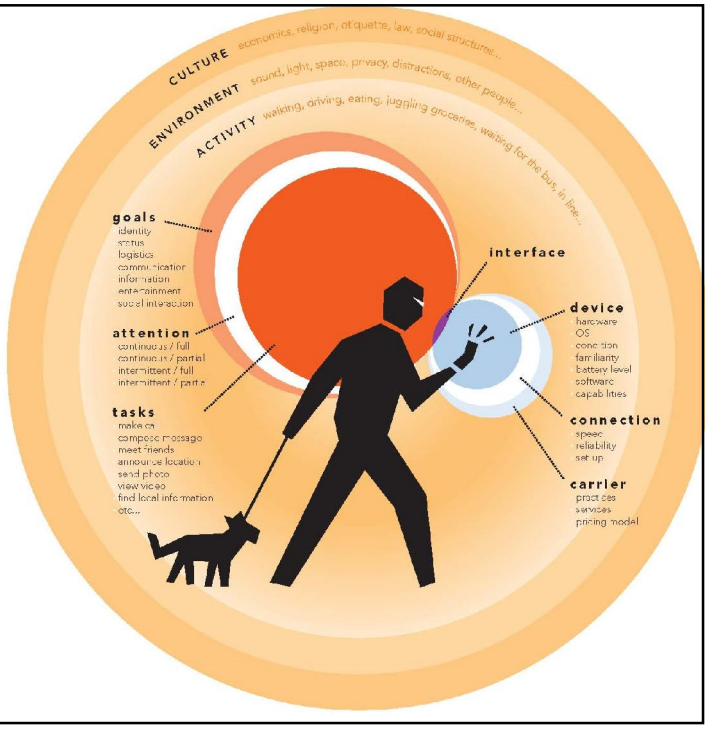
Where requirements come from

Type of projects	Source of requirements
In-house development By an organization's IT department	People within the organization: software users, managers and the IT department itself
Product or Service development For the mass market	Marketing, product management (on behalf of the public and the company), technical experts in different disciplines
Customer development For a single client	Business needs from the clients, terms and conditions from its commercial side
Commercial-off-the-shelf (COTS) Packaging purchase/tailoring	Package selection, often done in-house by matching needs to package capabilities; tailoring, as for either in-house or customer development
Subcontract e.g. withing a product development	Prime contractor, by derivation from the system requirements and design. Most, possibly all of this work is analysis rather than discovery
Maintainance Support for earlier customer development	System/software users within the client organization (via change requests or problem report received once the product is in service)

7

Tampereen yliopisto
Tampere University

"The Context of Use consists of the users, tasks and equipment (hardware, software and materials), and the physical and social environments in which a product is used."
(Maguire, 2001)



Mobile Context Model (Source: Giant Ant https://www.openroad.ca/2010/01/12/ux-wishlist-for-2010/)

8

Tampereen yliopisto
Tampere University

Components of context of use – beyond ”ease of use”

User	Task
User name • User type • User role **Experience, knowledge and skills** • Product experience • Related experience • Task knowledge • Organizational knowledge • Training • Input device skills • Qualifications • Language skills **Personal attributes** • Age, gender • Physical capabilities and limitations • Cognitive capabilities and limitations • Attitude and motivation	**Task list** • Task 1 • Task 2, etc. • Task characteristics (per task) • Task name • Task goal/output • Task steps • Task frequency • Task duration • Task flexibility • Task dependencies • Physical and mental demands • Task output • Risks resulting from error • Safety critical demands

Environment		
Technical environment • Hardware • Software • Network • Reference materials • Other equipment	**Physical environment** **Workplace conditions** • Atmospheric conditions • Auditory environment • Thermal environment • Visual environment • Environmental instability **Workplace design** • Space and furniture • User posture • Location **Workplace safety** • Health hazards • Protective clothing and equipment	**Organizational environment** **Structure** • Group working • Work practices • Assistance • Interruptions • Management structure • Communications structure • Salary or payment **Attitudes and culture** • Policy on computer use • Organizational aims • Industrial relations **Job design** • Job functions • Hours of work • Job flexibility • Performance monitoring • Performance feedback • Pacing, autonomy, discretion

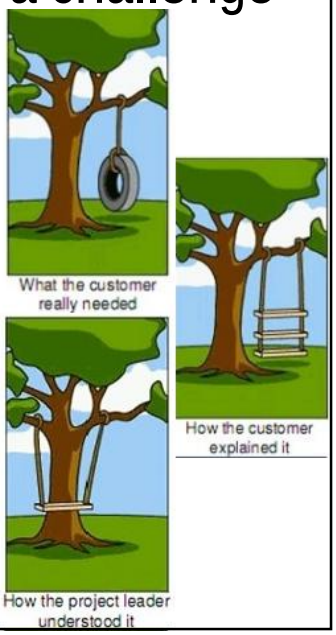
(Maguire, 2001)

9

Tampereen yliopisto
Tampere University

Extracting Requirements is a challenge (1/2)

- “Lack of user input” is the most common factor of failed or challenged projects
- People find it hard to articulate their needs
  - “I am hungry” is insufficient for a waiter to response
- Developers may not listen to the users
  - developers believe they already understand the user’s needs
  - developers may not understand users’ context, issues, or concerns
  - developers tend to overrule or dominate the users



10

Extracting requirements is a challenge (2/2)

The Expert (Short Comedy Sketch)

- Communication barriers
  - Users and developers come from different worlds
  - The users have different concerns from those of developers
  - Problems exist with the medium of communication: Natural languages are inherently ambiguous
  - A social interaction process: incompatible styles of interaction, different personality types, and different value systems among people
- In addition
  - adequate domain knowledge, perfect memory, sources of information, etc.

11

Tampereen yliopisto
Tampere University

Creativity is needed

- Software product development is a creative process in which stakeholders and designers work together to create ideas for new systems that are eventually expressed as requirements (Kauppinen et al. 2007)
- Creativity involves the production of novel (i.e. original, unexpected), useful (i.e. appropriate) products (Mumford, 2003, p. 110)
  - Discovering hidden customer/user needs
  - Inventing new product features that satisfy the needs
  - Innovative solution to the feature development

12

Tampereen yliopisto
Tampere University

John Cleese on Creativity

- https://www.youtube.com/watch?v=y70nbDJI5Uk
- Creativity is not a talent nor an ability that you either have or do not have.
- Creativity is a way of operating, and requires time, effort and curiosity – boundary of space and boundary of time
  - Understanding the problem
  - Idea generation -> brainstorming
  - Planning for action

13

Tampereen yliopisto
Tampere University

Brainstorming

- A simple group technique for generating ideas
- To invent new way of doing things or when much is unknown
  - When there are few or too many ideas
  - Early on in a project particularly when:
    - Domain is uncertain
    - There is little expertise for the type of applications
    - Innovation is important (e.g., novel system)
- Two main steps (in a 60-min session)
  - Generation (20 min)
    - Participants encouraged to offer as many ideas as possible without discussion of the merits of ideas
  - Consolidation (40 min)
    - Ideas are discussed, revised and organized
      - Pruning ideas that are not worthy of further discussion
      - Grouping of similar ideas into one super topic
    - Various voting techniques may be used to prioritize the ideas created

14

Outline

(Wiegers and Beatty, 2013) Chapter 7

Basic concepts

- Needs -> Requirements

Techniques

- Interviews
- Laddering
- Brainstorming
- Observation and enthnographic studies
- JAD
- Prototyping
- Scenarios
- Reuse
- Card sorting

Discussion

- Technique classification and selection

15

Tampereen yliopisto
Tampere University

Interviews

- The requirements engineer or analyst discusses the system with different stakeholders and builds up an understanding of their requirements
- Process consists of steps
  - Planning and preparation, interviewing session, consolidation of information, following-up
- Types of interview
  - Closed interviews. The requirements engineer looks for answers to a pre-defined set of questions
  - Open interviews. There is no predefined agenda and the requirements engineer discusses, in an open-ended way, what stakeholders want from the system
- Advantages and disadvantages
  - + Rich collection of information
  - + can probe in depth and adapt follow-up questions to what the personal tells you
  - - Large amount of qualitative date -> hard to analyze
  - - Hard to compare different respondents
  - - Interviewing is a difficult skill to master

16

Tampereen yliopisto
Tampere University

Interviewing essentials

- Interviewers must be open-minded and should not approach the interview with pre-conceived notions about what is required
- Stakeholders must be given a starting point for discussion. This can be a question, a requirements proposal or an existing system
- Interviewers must be aware of organizational politics - many real requirements may not be discussed because of their political implications
- Application domain knowledge is difficult to elicit during interviews
  - Difficult to discuss the domain without using the terminology
  - Difficult to explain some types of familiar domain knowledge (e.g. describe how to tie the shoelace)
- Interviews are rarely completely adequate for requirement elicitation

17

Tampereen yliopisto
Tampere University

Common Interviewing Mistakes (1)

- Not interviewing all of the right people
  - Different points of view of stakeholders
- Asking *direct* questions too early
  - e.g., design of a transportation system:
    - How much horsepower do you need?
    - How many passengers? How far? How fast?
  - e.g., camera design for a mobile device:
    - How important is control over shutter speed and aperture?
    - Will you be taking action shots, still shots, or both?
- Trying to convince stakeholders that YOU are smart – wrong place to do that!
  - *Client A: My elevators are too slow.*
  - *Req. Analyst A: I do not think so. I think you have an elevator throughput problem, not a speed problem.*
  - *Req. Analyst B: I see. Tell me why you feel they are too slow.*

18

Tampereen yliopisto
Tampere University

Common Interviewing Mistakes (2)

- Interviewing one-at-a-time instead of in small groups
  - More people might help get juices flowing as in brainstorming
  - Users cannot think of everything they need when asked individually, but will recall more requirements when they hear others' needs - group responses outperform the sum of the individuals' responses
  - Reduces spotlight on individuals (may produce more interesting answers), and do not let one participant dominate the discussion
- Assuming that stated needs are exactly correct
  - Often users do not know exactly what they want and order "what he is eating"
  - Need to narrow what is asked for down to what is needed

19

Laddering (Rugg & McGrorge 95) – An interview technique to analyze customer's preferences/elements of value

- Laddering is performed on the basis of a one-to-one interviewing technique, and starts out by focusing on a certain product or service and its *attributes*
- Definition
  - Probe seed item using question types – "why" question
    - Why you buy/like a product?
    - *Why is that (attribute) important to you? / What does it mean to you?*
    - Why is it important that …?
  - Construct conceptual graph of domain – Attributes, Consequences, Core values
- Advantages and disadvantages
  - + quick, simple to use
  - + structured but flexible
  - + amendable to automation
  - - tedious
  - - sometimes difficult in reflecting on abstract reasons

20

Example of using Laddering

- A salty-snack study - elicit consumers' preferences towards salty-snacks
- The consumer is asked about the product's attributes.
- On the basis of the attributes, probe: "*Why is that (attribute) important to you?*", "Why is it important that…"
- Answers from one interview:
  - Self-esteem(V)
  - Better figure(C)
  - Don't get fat(C)
  - Eat less(C)
  - Strong taste(A)
  - Flavored chip (A)
- And then you draw a Hierarchical Value Map (HVM) that show the different concepts and relations (ladders) (http://academic.csuohio.edu/cbresearch/papers/Good%20PDFs/Laddering_A%20How%20to%20do%20it%20manual.pdf)

21

Tampereen yliopisto
Tampere University

Joint Application Design (JAD)

- Developed at IBM in the 1970s
  - Lots of success stories
- "Structured brainstorming", IBM-style
  - Full of structure, defined roles, forms to be filled out…
  - Promoting co-operation, understanding and teamwork
- JAD session may last few days, it facilitates the creation of a shared vision of what the system should be
  - Used for making decisions on different aspects of a project
  - Any process where consensus-based decision making across functional areas is required, e.g.,
    - Planning a project
    - Defining requirements
    - Designing a solution

22

JAD steps

- Project definition
  - JAD facilitator interviews managers and clients to determine objectives and scope
  - Forms a team of users, clients and developers: all stakeholders are represented; participants are able to make binding decisions
- Evaluate project - identify issues and scope of JAD session
  - JAD facilitator interviews present and future users, gathers domain info and describes work flows
  - Starts a list of issues to be addressed during the session
- Preparation
  - Create the working Doc., a first draft of Final Doc.
  - Make an agenda for the session
  - Make overheads, flip charts etc. to represent information gathered during previous session

23

JAD steps (Cont.)

- Working session
  - Over 3-5 days guide teams in creating the system specification
  - Teams define and agree on work flow, data elements, screens and reports
- Final document
  - JAD facilitator prepares the Final Doc. from the draft and decisions made during previous session
  - Final Doc. distributed to session participants for review
  - Participants meet for 1-2 hour meeting to discuss the reviews and finalize the document

24

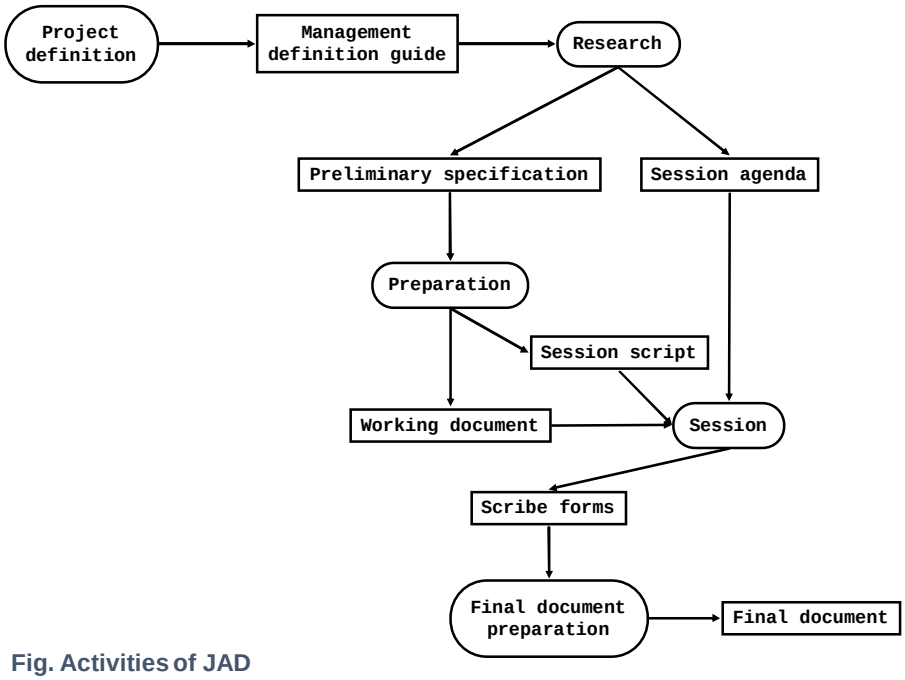


Fig. Activities of JAD

25

Observation and Ethnography (1/2)

- People often find it hard to describe what they do because it is so natural to them. Sometimes, the best way to understand it is to observe them at work
- Ethnography is a technique from the social sciences which has proved to be valuable in understanding actual work processes
  - Actual work processes often differ from formal, prescribed processes
  - Ethnography involves an observer spending an extended period in a society or culture, making detailed observation of all their practice, and building up a picture of how work is done

26

Observation and Ethnography (2/2)

- Assume that people are good at doing their job
- Spend time getting to know the people and establish a trust relationship
- Keep detailed notes of all work practices. Analyze them and draw conclusions from them
- Combine observation with open-ended interviewing
- Combine ethnography with other elicitation techniques

27

Requirements Reuse

- Reuse involves taking the requirements which have been developed for one system and using them in a different system
- Saves time and effort, and reduces risk
  - Studies have shown that similar systems can re-use up to 80% of the requirements
  - Reused requirements have already been analyzed and validated in other systems
  - Reused requirements have a better chance of being understood by all the stakeholders.
- Requirements reuse may lead to additional reuse in other lifecycle activities
- Currently, requirements reuse is an informal process but more systematic reuse could lead to larger cost savings
  - Generic application domain information
  - Style of information presentation, e.g. UI patterns
  - Company policies or regulations, e.g. security policies

28

Scenarios

- A set of interaction scenarios can be used to elicit and clarify system requirements
- A scenarios is one example of interaction sessions which describe how a user interacts with a system – a single instance
- Discovering scenarios exposes possible system interactions and reveals system facilities which may be required – abstraction
- For complex systems, a fairly large number of scenarios will usually be required – all possible situations

29

Scenario Example: Warehouse on Fire

- Bob, driving down main street in his patrol car notices smoke coming out of a warehouse. His partner, Alice, reports the emergency from her car.
- Alice enters the address of the building, a brief description of its location (i.e., north west corner), and an emergency level. In addition to a fire unit, she requests several paramedic units on the scene given that area appear to be relatively busy. She confirms her input and waits for an acknowledgment.
- John, the Dispatcher, is alerted to the emergency by a beep of his workstation. He reviews the information submitted by Alice and acknowledges the report. He allocates a fire unit and two paramedic units to the Incident site and sends their estimated arrival time (ETA) to Alice.
- Alice received the acknowledgment and the ETA.

30

Tampereen yliopisto
Tampere University

Find use cases in the scenario

- Identify use cases in the scenario – abstractions of possible coherent scenarios
  - Bob, driving down main street in his patrol car notices smoke coming out of a warehouse. His partner, Alice, reports the emergency from her car.
  - Alice enters the address of the building, a brief description of its location (i.e., north west corner), and an emergency level. In addition to a fire unit, she requests several paramedic units on the scene given that area appear to be relatively busy. She confirms her input and waits for an acknowledgment.
  - John, the Dispatcher, is alerted to the emergency by a beep of his workstation. He reviews the information submitted by Alice and acknowledges the report. He allocates a fire unit and two paramedic units to the Incident site and sends their estimated arrival time (ETA) to Alice.
  - Alice received the acknowledgment and the ETA.
- Describe the use cases in more detail
  - Using use case templates

31

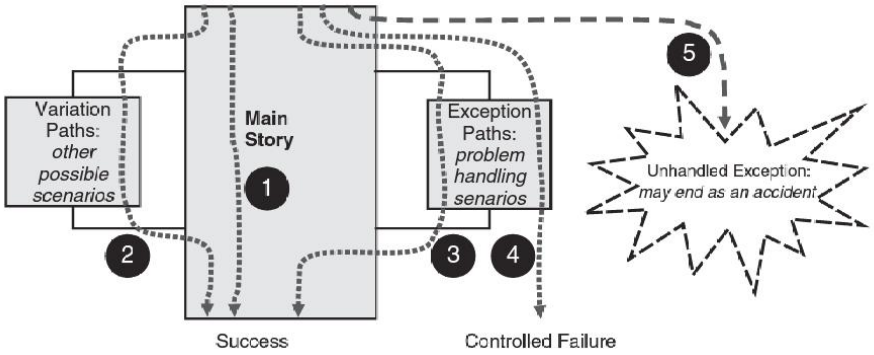
Use Case Example: ReportEmergency

- Use case name: ReportEmergency
- Participating Actors:
  - Field Officer (Bob and Alice in the Scenario)
  - Dispatcher (John in the Scenario)
- Flow of Events
  - The FieldOfficer activates the "Report Emergency" function of her terminal. It responds by presenting a form to the officer.
  - The FieldOfficer fills the form, by selecting the emergency level, type, location, and brief description of the situation. The FieldOfficer also describes possible responses to the emergency situation.
  - Once the form is completed, the FieldOfficer submits the form, at which point, the Dispatcher is notified.
  - The Dispatcher reviews the submitted information and creates an Incident in the database by invoking the OpenIncident use case.
  - The Dispatcher selects a response and acknowledges the emergency report.
  - The FieldOfficer receives the acknowledgment and the selected response.
- Exceptions:
  - The FieldOfficer is notified immediately if the connection between her terminal and the central is lost.
  - The Dispatcher is notified immediately if the connection between any logged in FieldOfficer and the central is lost.
- Special Requirements:
  - The FieldOfficer's report is acknowledged within 30 seconds. The selected response arrives no later than 30 seconds after it is sent by the Dispatcher.

32

Tampereen yliopisto
Tampere University

The structure and behaviour of a use case



- Scenario ❶: the normal way of achieving the goal;
- Scenarios ❷: other possible ways of achieving the same goal;
- Scenarios ❸: recovering from exception events to achieve success;
- Scenarios ❹: handling unrecoverable exceptions safely;
- Scenarios ❺: the exceptions you choose not to handle.

33

Tampereen yliopisto
Tampere University

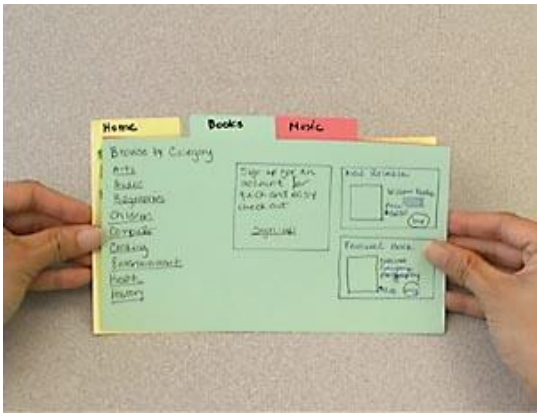
Prototyping (1/5)

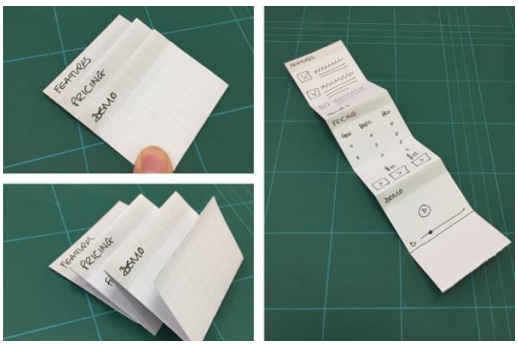
- A software requirements prototype is a mock-up or partial implementation of a software system
  - Helps to better understand system requirements
  - Helps clarify and complete requirements
  - Provides early response to IKIWISI ("I'll know it when I'll see (or won't see) it") attitude
  - Effective in addressing the "Yes, But" and the "Undiscovered Ruins" syndromes
- Prototyping is effective in resolving uncertainties early in the development process – donot prototype requirements you already understand
  - Focus prototype development on these uncertain parts
    - A functional prototype
    - A user interface prototype
  - Encourages user participation and mutual understanding

34

Tampereen yliopisto
Tampere University

Prototyping (2/5) - Paper prototypes on index card



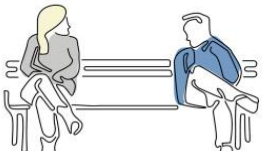


35

Tampereen yliopisto
Tampere University

Prototyping (3/5) - Storyboard

Alice and Bob meet each other for the first time and have a 10-minute conversation.



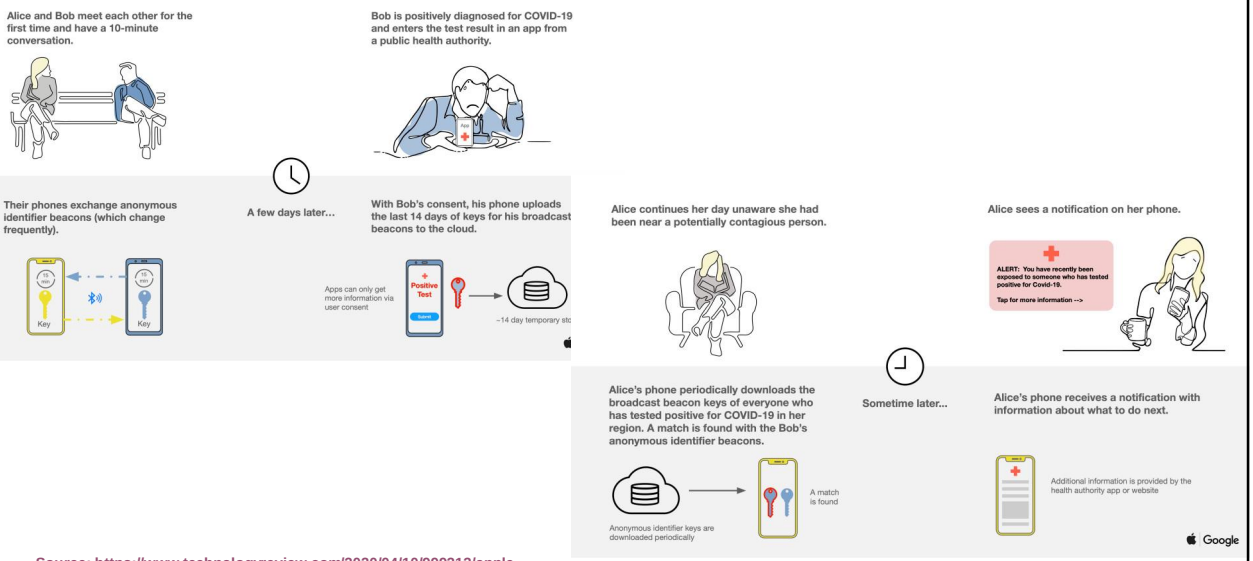
Bob is positively diagnosed for COVID-19 and enters the test result in an app from a public health authority.

Alice continues her day unaware she had been near a potentially contagious person.

Alice sees a notification on her phone.

36

Tampereen yliopisto
Tampere University

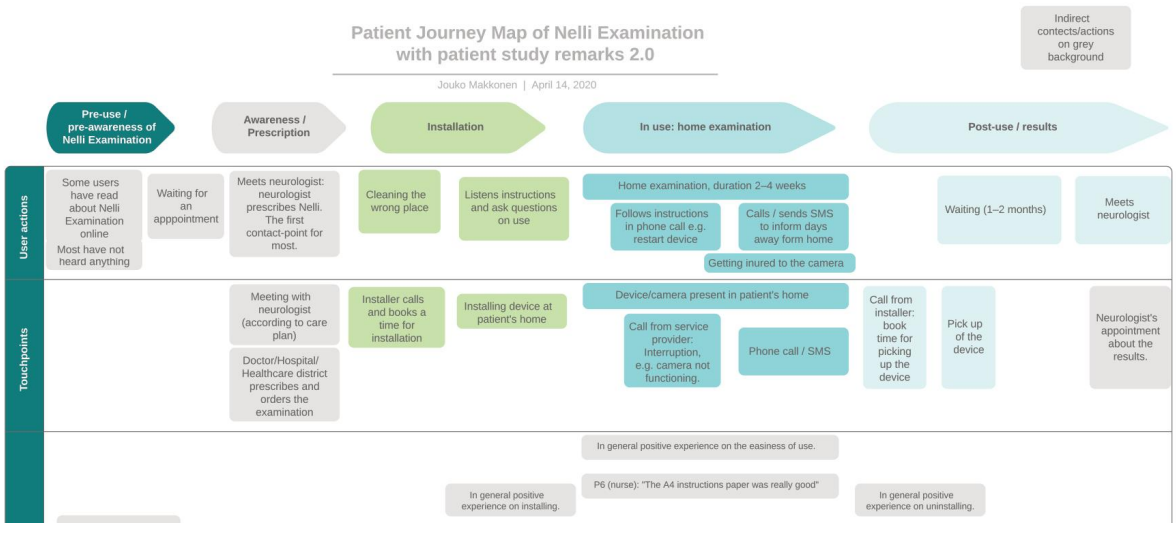


Source: https://www.technologyreview.com/2020/04/10/999213/apple-and-google-are-building-coronavirus-tracking-into-ios-and-android/

37

Tampereen yliopisto
Tampere University

Prototyping (4/5) - User Journey Map

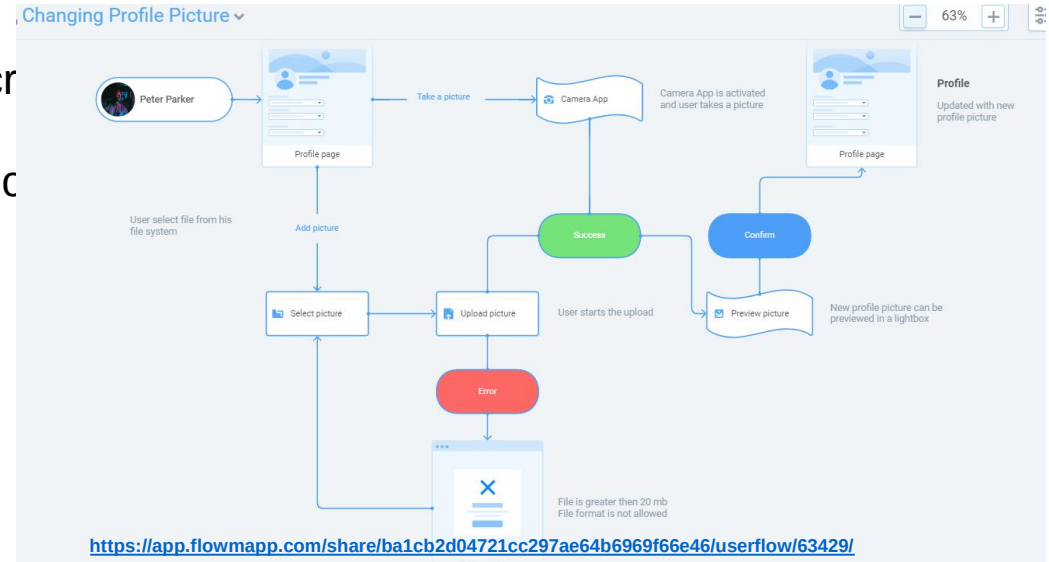


38

Tampereen yliopisto
Tampere University

Prototyping (5/5) – Interactive prototypes

- Scr
- Mc

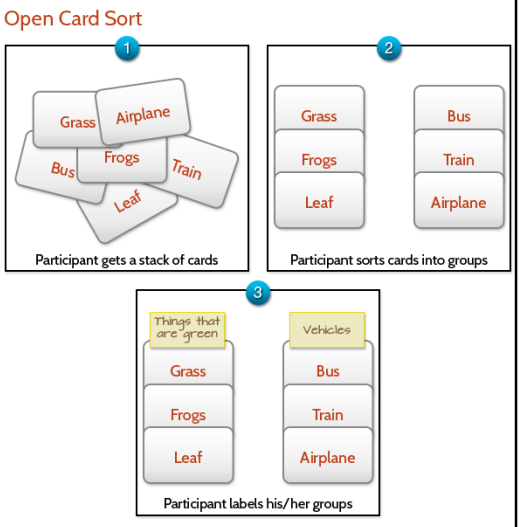


https://app.flowmapp.com/share/ba1cb2d04721cc297ae64b6969f66e46/userflow/63429/

39

Card Sorting (Gammack 1987, Rugg et al. 1992)

- Definition
  - Cards and items
  - Examines perceptions of categories and criteria
  - Sort cards into groups
  - Hierarchical and non-hierarchical sorts
- Advantages and Disadvantages
  - + quick, simple to do
  - + Quantitative information you want
  - - captures structure and categories only
  - - needs preparation



40

Outline

(Wiegers and Beatty, 2013) Chapter 7

Basic concepts

- Needs -> Requirements

Techniques

- Interviews
- Laddering
- Brainstorming
- Observation and enthnographic studies
- JAD
- Prototyping
- Scenarios
- Reuse
- Card sorting

Discussion

- Technique classification and selection

41

Categorize the techniques (1/5)

- Requirements development is an intensive interaction process between stakeholders and analysts
- The techniques can be classified based on the means of interaction
  - Conversational
  - Observational
  - Analytic
  - Synthetic

(Zhang, 2007)

42

Categorize the techniques (2/5)

Conversational methods

- A means of verbal communication between two and more people
- e.g. interview, workshop, brainstorming, laddering, etc.
- Practical and efficient to collect non-tacit knowledge
- Labor intensive, and challenge to facilitate the session

(Zhang, 2007)

43

Categorize the techniques (3/5)

Observational methods

- A means of developing a rich understanding of the application domain by observing human activities
- e.g. observation, think-aloud protocol, etc.
- Tacit requirements elicitation, understanding the domain rather than making judgements
- Longitudinal studies

(Zhang, 2007)

44

Categorize the techniques (4/5)

Analytic methods

- A means of exploring the existing documentation or knowledge and acquire requirements from a series of deductions
- e.g. documentation studies, requirements reuse, user feedback analysis, etc.
- Requirements are captured from other sources, e.g. expert knowledge, documents, etc.

(Zhang, 2007)

45

Categorize the techniques (5/5)

Synthetic methods

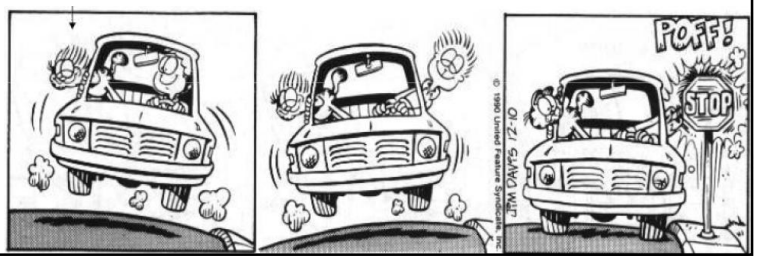
- A means of combining conversation, observation, and analytic techniques into one
- e.g. prototyping, JAD/RAD session, etc.
- Good cues for requirements recognition
- Harmonize the requirements with other development activities

(Zhang, 2007)

46

Perspectives of methods selection

- Type of knowledge
  - problem analysis
  - Specific product description
- Source of knowledge
  - Human being
  - Other entities in the environment
- Acquisition context
  - Culture
  - Organizational context
  - Individual cognitive limitation
  - across time and space
- Requirements certainty
  - Well-understood problem domain
  - Unstructured problem domain



47

Reading materials

- Maguire (2001) Context of use within usability activities, Int. J. Human-Computer Studies(2001) 55, 453-483

48

Tampereen yliopisto
Tampere University

Day3: Group assignments, due at 12pm, Aug. 8 2020

- With your group members, continue working on the proposed software development project (20 points)
  - Analyze the context of use of the software product – 5p
  - How to elaborate on the needs? Prepare for a plan to include the selection of techniques (at least two) for elaborating on the needs and the steps. -10p
  - Follow the plan to conduct the requirements elicitation and select one prototyping technique to report the requirements. – 5p
- In addition, Groups 5, 7, 8 and 9 prepare for a presentation (including 3 or 4 slides) of your answers to the above assignment, and present it tomorrow (Aug. 8) in the course (五七同城，喇叭组，就这就这组，秃头小队)

49

Tampereen yliopisto
Tampere University

Day2: Group assignments, due at 12pm, Aug. 7 2020

- With your group members, please propose a software development project, and prepare for a specification that covers the project's (20 points)
  a) Vision statement or NABC – 10p
  b) key stakeholders? – 2p
  c) stakeholder profile – 2p
  d) power/interest grid – 2p
  e) And explain how you analyze and validate the project's stakeholders? – 4p
- In addition, Groups 2, 12, 3 and 10 prepare for a presentation (including 3 or 4 slides) of your project plan to cover the product vision and stakeholder analysis, and present it tomorrow (Aug. 7) in the course (影流之组，就很牛皮，Cardigan，阿勇废了，下一组)

50

Tampereen yliopisto
Tampere University

Presentation of Day 2's group work

- Group 2 - 影流之组
- Group 12 - 就很牛皮
- Group 3 - Cardigan
- Group 10 - 阿勇废了，下一组

51