Design of Interactive Systems
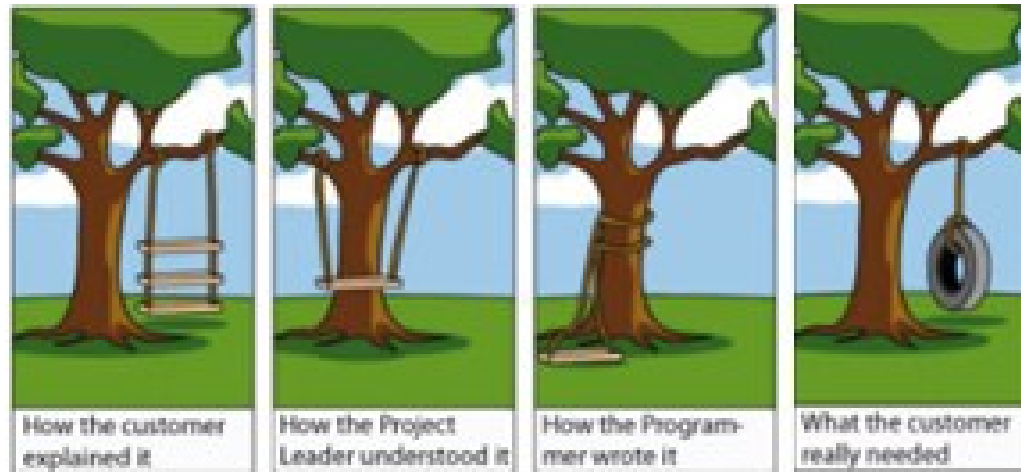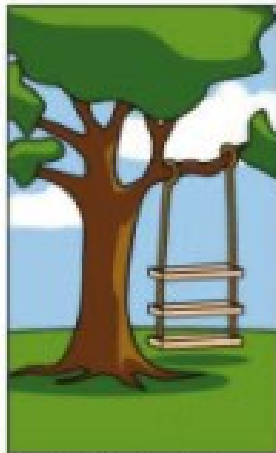## Human-Centered design of interactive systems and Usability Engineering Lifecycle

How the customer explained it

How the Project Leader understood it

How the Programmer wrote it
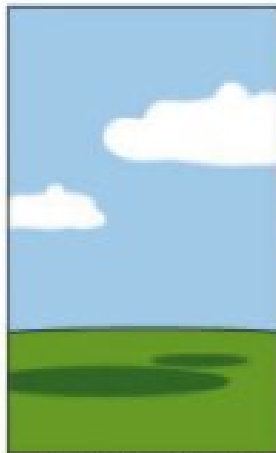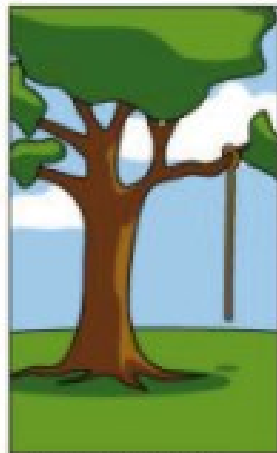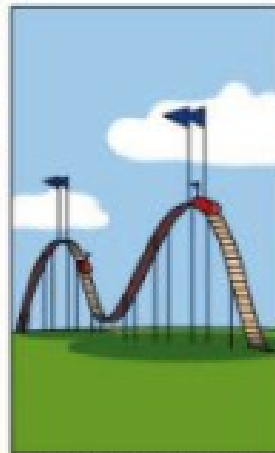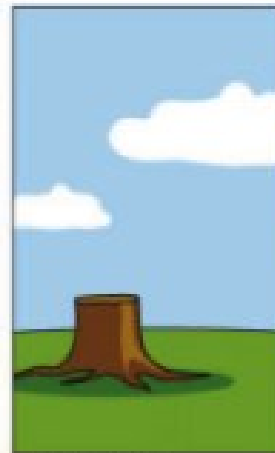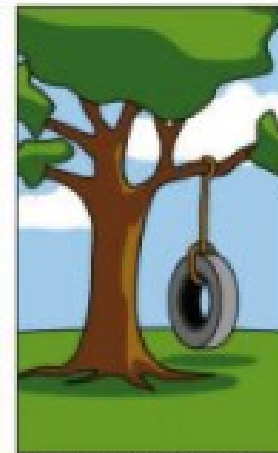
What the customer really needed
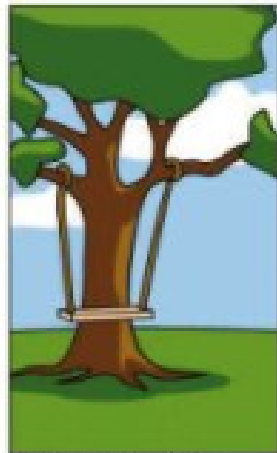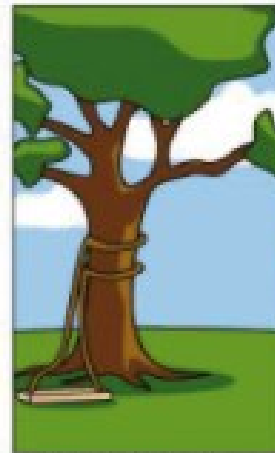
The problem of interactive systems design…

Human Centered Design of interactive systems

- Also known as User Centered Design (UCD)

- Complementary approaches to the design that should be used:

  Usability principles (technology independent)

  Usability paradigms (more technology dependent)

  Guidelines and standards (more specific)

- **S/W development technologies**

- **Evaluation**

- ISO standard 13407 (1999) - *Human centered design processes for interactive systems*:

"Human-centered design is an approach to interactive system development that focuses specifically on making systems usable. It is a multi-disciplinary activity."

- There are several proposals of UCD methodologies

- All are iterative

- And include usability evaluation in iterations

We must consider the situation at hand and ponder which are the best fitting and how to use them

https://uxpa.org/
https://www.w3.org/WAI/redesign/ucd
http://www.usability.gov/how-to-and-tools/methods/user-research/index.html
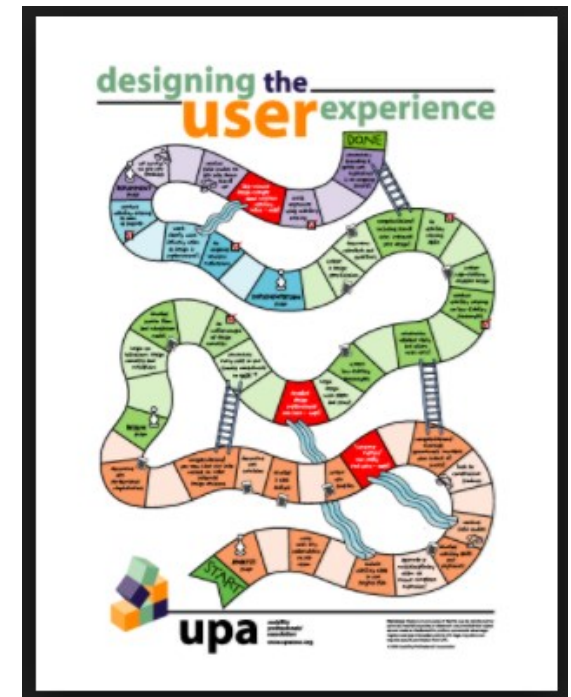
Benefits of Human-centered design

- Following the best practices, helps to identify challenges upfront so that a solution can be found early

- By putting a larger emphasis on principles and practices, iterative improvements can be made and avoid costly large scale rework

- The"10%" rules:
  - 10% of IT staff should be user experience (UX) professionals
  - 10% of budget dedicated to UX.

http://www.usability.gov/what-and-why/benefits-of-ucd.html

Top 12 reasons IEEE identified for why IT projects fail in 2005 (three can be fixed by using adequate UX methods)

- Unrealistic or unarticulated project goals
- Inaccurate estimates of needed resources
- Badly defined system requirements ← May be alleviated by good UX design
- Poor reporting of the project's status
- Unmanaged risks
- Poor communication among customers, developers, and users
- Use of immature technology
- Inability to handle the project's complexity
- Sloppy development practices
- Poor project management
- Stakeholder politics
- Commercial pressures

http://spectrum.ieee.org/computing/software/why-software-fails

A study about why IT projects fail
(and several causes are avoidable using adequate UCD methods)

Comparing with the situation in aviation an **IT accident** may be defined as:
 "project with significant damage, e.g. large cost or schedule overrun, failing business goals, low usability, etc."

Five accident investigations of large government IT projects in Denmark

Identified 37 different causes, each causing damage to one or more projects, e.g.

- surprises with system integration

- wrong estimate of human performance

Only one cause related to programming

Proposes 22 cures that in combination could have prevented most of the damages

Half are familiar to developers, but were ignored in the project (e.g. usability test)

S. Lauesen, "IT Project Failures , Causes and Cures," *IEEE Access*, vol. 6, pp. 72059-67, 2020.

Some causes for IT projects failures
(and several causes are avoidable using adequate UCD methods)

Analysis

Doesn't identify user needs and win-win
Requirements don't cover customer needs
Describes solution in detail. No freedom to supplier
Makes heavy demands and believes it is for free
Oversells technology, e.g. SOA, web-based
Multi-vendor strategy - supplier independent ! ?
Wants everything at once, e.g. cover all types of debt
Doesn't plan the new work processes
No feasible solution, e.g. data missing, performance dubious.
Surprising rule complexity.

Study as-is and plan to-be
Problem oriented requirements (SL-07)

Plan to-be, SL-07 reqs (sections B1 and C)

Design

Doesn't ensure usability, even when they know how
Designs user screens too late
Accepts the solution description without understanding it
Cannot see how far the supplier is

Early prototypes & usability tests
Early prototypes & usability tests
Early prototypes & usability tests
Early prototypes , Monitor remaining work hours

Test

Deploys the system with insufficient testing

Pilot test, Ask expert developers

Deployment

Deploys system with insufficient support/training
The system is not used as intended
Wrong estimate of human performance

Usability tests , Pilot test, Deploy part-by-part
Observe at pilot test, Follow-up study
Check at POC, Check at pilot test

HCD/UCD involves

- Understanding users' needs, motivations, contexts and activities

- Understanding business, technical, and domain opportunities, requirements, and constraints

- Using this knowledge as a grounds to create products whose form, content, and behavior is useful, usable, economically viable and technically feasible

- We must know the successful cases (usability paradigms)

- Understand why they work (usability principles)

- Adopt adequate methodologies

- And test, re-design, test, re-design

...
until usability goals are met

This is applicable not only to conventional interactive systems
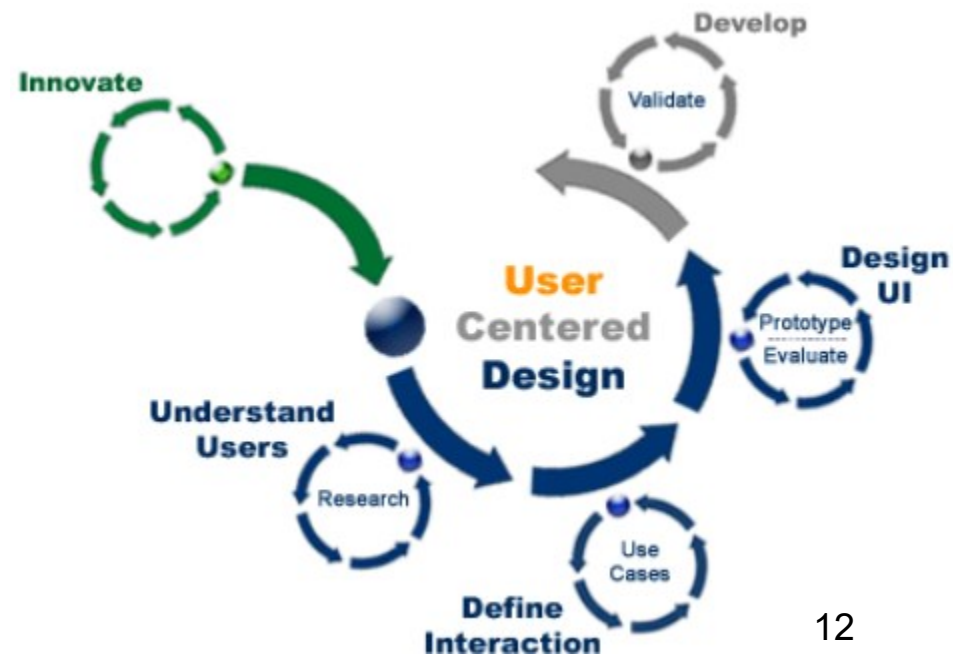
Iteration

- Its role is to enable continual refinement through rapid prototyping and testing through repetition

- Observe and study,

- Decide what the problem might be

- Determine which parts of the design work and which don't

iterate again…

- Fosters getting the requirements right

Examples of several approaches

- The Usability Engineering Lifecycle (Mayhew, 1999)

- Goal-directed design (Cooper, 2007)

- Activity-oriented design (Norman, 2013)

- Etc.

One of the first Human-Centered Design approaches:

The Usability Engineering Lifecycle
(Mayhew, 1999)

THE USABILITY ENGINEERING LIFECYCLE
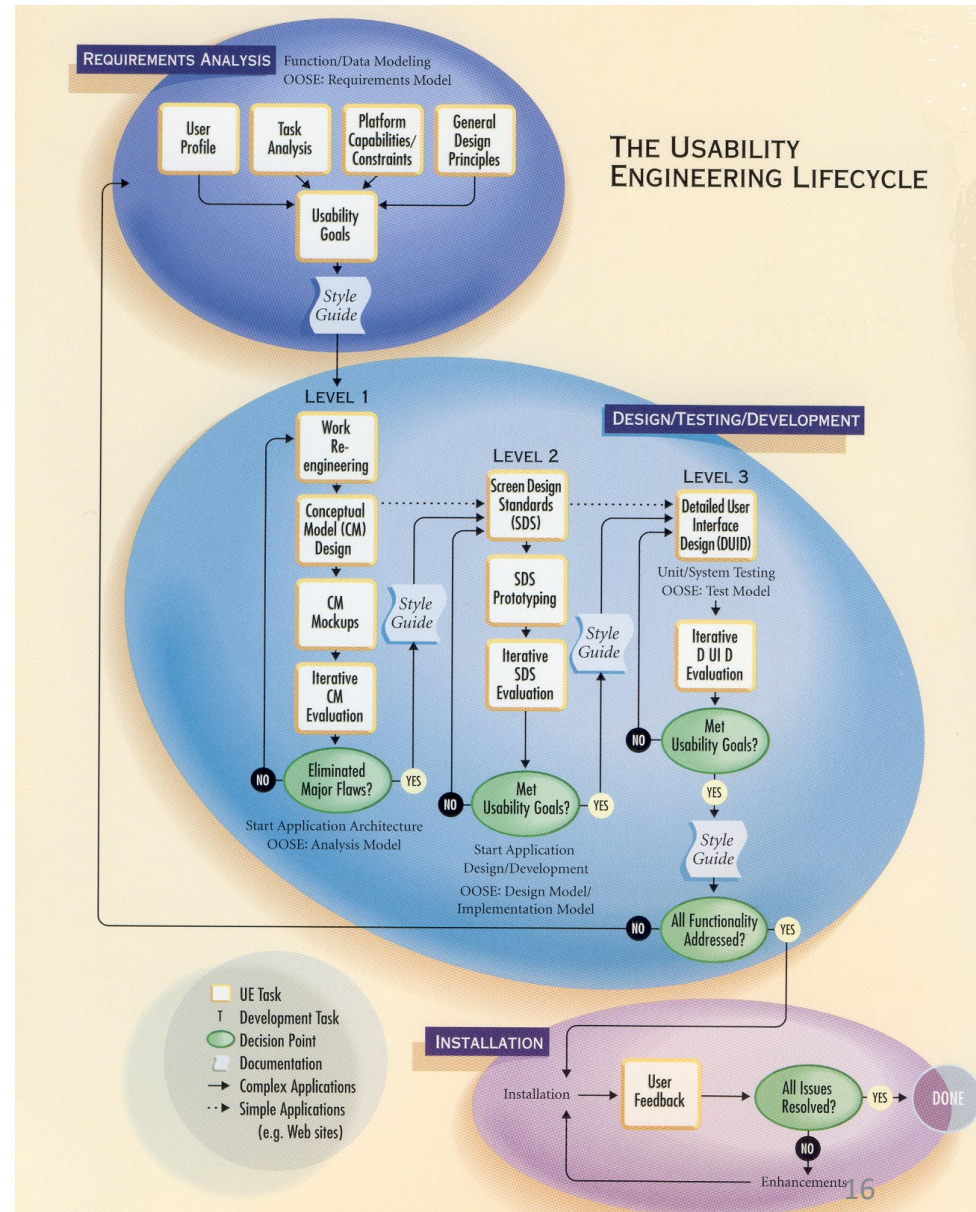
Requirement Analysis

User Profiles	Establish user characteristics important for UI design
Contextual Task Analysis	Obtain a user-centered model of work as it is currently done; extract the product usability requirements
Usability Goal Setting	Establish specific quantitative and qualitative usability goals to drive UI design
Platform Capabilities and Constraints	Establish capabilities and constraints of the technology platform which limit UI design alternatives
General Design Principles	Identify principles and guidelines that may be relevant for the product under development

Design, Testing, Development – Level 2

Screen Design Standards	Establish a set of design standards to set the stage for detailed UI Design
Screen Design Standards Prototyping	Support the evaluation, refinement and validation of the Screen Design Standards
Iterative Screen Design Standards Evaluation	Evaluate, refine, and validate the Screen Design Standards
Style Guide Development	Document the Conceptual Model Design, the Screen Design Standards and the output of Requirement Analysis

Design, Testing, Development – Level 3

Detailed User Interface Design

Design the complete, detailed product User Interface

Iterative User Interface Evaluation

Evaluate, refine, and validate key subsets of the detailed User Interface Design

Installation

User Feedback

Obtain usability data after a product has been installed and used

Inform the UI Design for later releases or related products

User Interface design
from a S/W engineering perspective
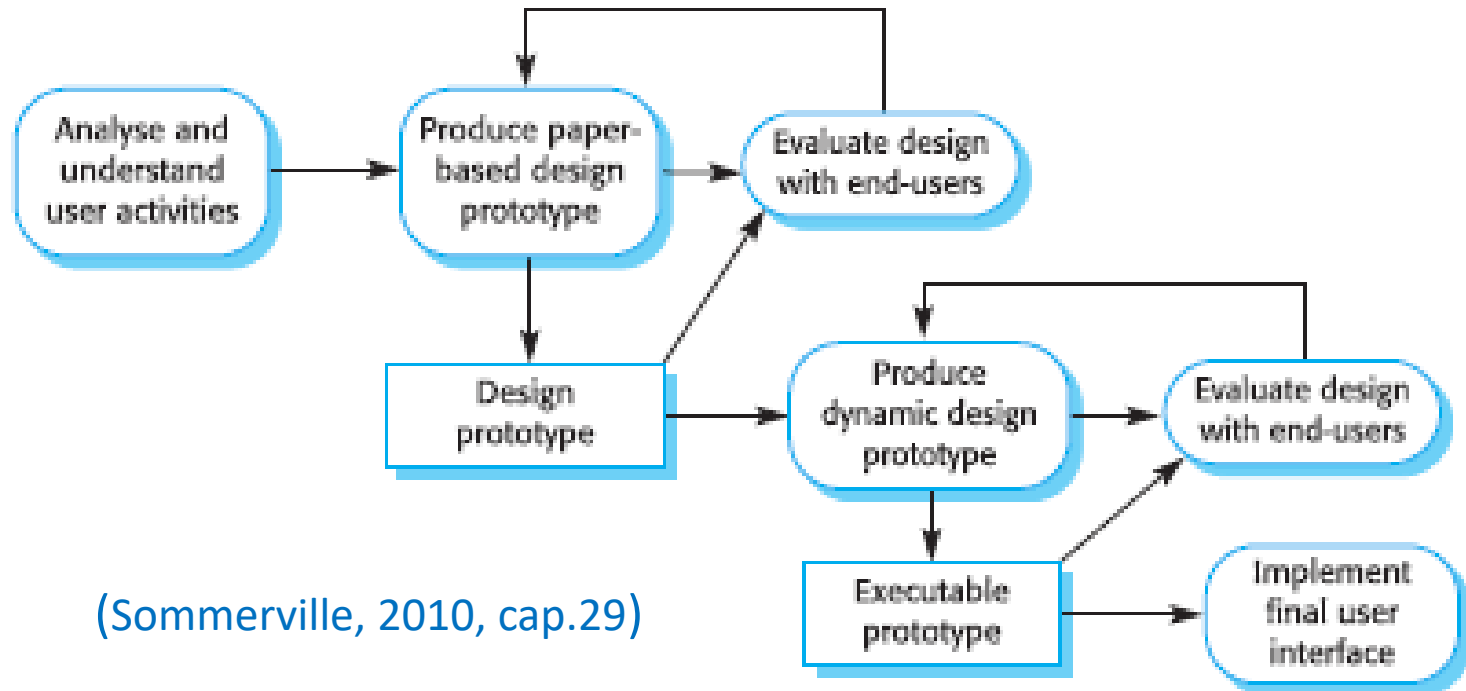
(Sommerville, 2010, cap.29)

Goal-directed design (Cooper, 2007)

- Works on any platform, in any subject area, and with any users

- Is based on understanding the users and their desired end-state

- Not on any particular technology

- It involves personas and workflows

Activity-based design (Norman, 2013)

- Let the activity define the product

- Let the conceptual model be built around the activities

- Activity is a set of tasks performed together toward a goal

    It is different from task

    Is a high-level structure (e.g. "go shopping")

- Tasks are lower level components (e.g. walk to the shop, ask for a product…)

User Research Methods

There are a lot of methods; we must consider the situation at hand to select and adapt the ones we use:

- Context interviews
- Focus groups
- Individual interviews
- On-line surveys
- **Personas**
- **Scenarios**
- **Task analysis**
- Activity analysis
- First click tests
- …
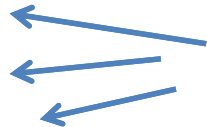
Methods to use in the mini-project

http://www.usability.gov/how-to-and-tools/methods/user-research/index.html

Personas and scenarios

1. Persona

Defines who the story is about. This main character has attitudes, motivations, goals, and pain points, etc.

3. Goal

Defines what the persona wants or needs to fulfill. The goal is the motivation of why the persona is taking action. When that goal is reached, the scenario ends.

2. Scenario

Defines when, where, and how the story of the persona takes place. The scenario is the narrative that describes how the persona behaves as a sequence of events.

https://www.interaction-design.org/literature/article/personas-why-and-how-you-should-use-them

Personas

- **Fictional characters based upon user research** to represent the different user types that might use a service/product in a similar way

- make the design task less complex, and guide the ideation process

Your User

- help:
  - understand users' needs, experiences, behaviors and goals
  - step out of oneself and recognize that different people have different needs and expectations
  - uncovering universal features and functionality
  - create a good user experience for your target users

History of Personas

Stem from IT system development during the late 1990s

How to best communicate an understanding of the users?

Various concepts emerged:

        user archetypes, user models, lifestyle snapshots, …

Alan Cooper (1999) proposed personas to describe fictitious users

There is **no single definition** of what a persona should contain

Nor a unified understanding of **how to apply** the method

Benefits of Personas

- Offer a quick and inexpensive way to test and prioritize features throughout the development process

- Help to
    - focus decisions by adding a layer of real-world consideration

    - Stakeholders evaluate new feature ideas

    - Information architects develop informed wireframes, and interface behaviors

    - Designers create the overall look and feel

    - System engineers/developers decide which approaches to take based on user behaviors

https://www.usability.gov/how-to-and-tools/methods/personas.html

Types of Personas

- Several types (most based on previous user research):

    - **Goal-directed** Personas (Cooper, 2007)

    - **Role-based** Personas (goals + behavior)

    - **Engaging** Personas (goals + behavior + backgrounds)

    - **Fictional** Personas (based on assumptions, not user research)

- **Fictional personas can only be used as an initial sketch of user needs**

    https://www.interaction-design.org/literature/article/personas-why-and-how-you-should-use-them

Best Practices for Developing Personas

- Create 2-4 personas of the product/service main audiences

- Conduct user research:
    - Who are the users
    - Why are they using the system?
    - What behaviors, assumptions, and expectations?

- Develop the appropriate descriptions of each persona's: background, motivations, and expectations

- Do not include too personal information
- Be relevant and serious

Elements of a persona

- Persona Group (i.e. web manager)

- Fictional name

- Job titles and major responsibilities

- Demographics such as age, and education

- The goals and tasks they are trying to complete using the product

- Their physical, social, and technological environment

Personas have no value in themselves, until they become part of a scenario they do not have real value!

Example of a Persona

Persona:	USDA Senior Manager Gatekeeper
Photo:	
Fictional name:	Matthew Johnson
Job title/ major responsibilities:	Program Staff Director, USDA
Demographics:	• 51 years old • Has a Ph.D. in Agricultural Economics.

Goals and tasks:

Spends his work time:

- Requesting and reviewing research reports,
- preparing memos and briefs for agency heads, and
- supervising staff efforts in food safety and inspection.

Environment:

He is comfortable using a computer and refers to himself as an intermediate Internet user. He is connected via a T1 connection at work and dial-up at home. He uses email extensively and uses the web about 1.5 hours during his work day.

https://www.usability.gov/how-to-and-tools/methods/personas.html

Example of using personas in VR

Model the people who will be using the VR application

Help to prevent the design from being driven by design/ engineering convenience

Personas should
- **Not be too detailed**

- **be validated** in later stages

(Jerald, 2016)

Name	• Job • Experience • Activities • Attitude • Competencies • Age
• Problems • Pain points • Needs • Concerns • Fears • Desires	• Knowledge of VR • Dream VR system • Vision of VR • VR hardware access • Budget for VR • Activities that fit VR

Describe 2–4 characters representing the range of targeted users

Sketch/photo and name

Basic description of the person

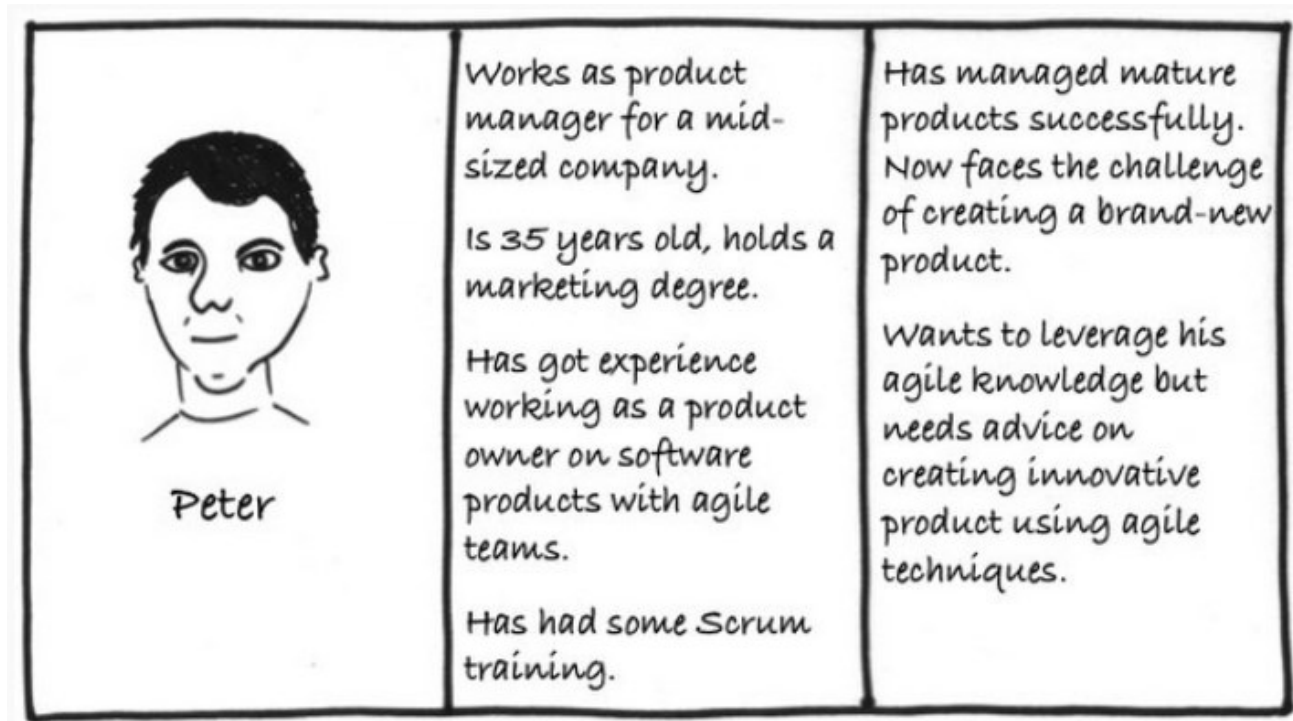|	
Name	• Job • Experience • Activities • Attitude • Competencies • Age
• Problems • Pain points • Needs • Concerns • Fears • Desires	• Knowledge of VR • Dream VR system • Vision of VR • VR hardware access • Budget for VR • Activities that fit VR

Challenges the person has

Relation to type of system (VR in this case)

If personas are especially important (e.g., for therapy applications), then data should be carefully collected with interviews and/or questionnaires

Another example of a Persona

| Peter | Works as product manager for a mid-sized company.

Is 35 years old, holds a marketing degree.

Has got experience working as a product owner on software products with agile teams.

Has had some Scrum training. | Has managed mature products successfully. Now faces the challenge of creating a brand-new product.

Wants to leverage his agile knowledge but needs advice on creating innovative product using agile techniques. |

- A main difficulty of the persona method is getting the team members to use it

- The 10-step process of creating a persona can help

https://www.interaction-design.org/literature/article/personas-why-and-how-you-should-use-them
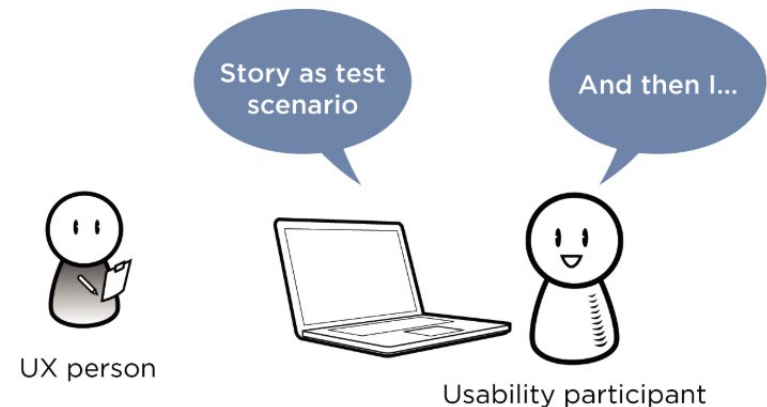
Personas: The Take Away

- Personas are fictional characters based on user research to help understand:

  - users' needs,

  - experiences,

  - behaviors

  - goals.

- Make the design task at hand less complex

- Guide the ideation processes, and help to achieve the goal of creating a good user experience for the target user group

- The 10-step process covers the entire process from the preliminary data collection, through active use, to continued development of personas.

  https://www.interaction-design.org/literature/article/personas-why-and-how-you-should-use-them

Scenarios

- Stories and contexts about how the user groups use a future product/service

- Note the goals and questions to be achieved and sometimes define the possibilities of how the user(s) can achieve them on the product/service

- Scenarios are critical for
  - designing
  - usability testing

Story as test scenario

And then I...

UX person

Usability participant

https://www.usability.gov/how-to-and-tools/methods/scenarios.html

https://www.interaction-design.org/literature/topics/user-scenarios

40

- Scenarios should be used in the ideation phase of a project

- Scenarios need to be based on research with users

- Do *not* represent *all* possible users

- Typically account for the *most common* users or user motivations

- Are commonly based on personas

- Can be used to determine the most important areas to test during usability testing, and to provide guidance to the test

https://www.interaction-design.org/literature/topics/user-scenarios

What to Consider When Writing Scenarios

• Good scenarios are concise but answer the following questions:

- **Who is the user?** Use the personas

- **Why does the user uses the product?** Note what motivates the user and their expectations, if any

- **What goals does s/he have**? Use task analysis

- **How can the user achieve their goals with the product?**

https://www.usability.gov/how-to-and-tools/methods/scenarios.html

Types of Scenarios

- **Goal/Task-based Scenarios** state only what the user wants to do

Example: You are traveling to Paris for your job next week and you want to check on the amount you can be reimbursed for meals and other expenses

- **Elaborated Scenarios** give more user story details

- **Full Scale Task Scenarios** include the steps to accomplish the task

https://www.usability.gov/how-to-and-tools/methods/scenarios.html

Scenarios: The Take Away

- User scenarios are a great way of communicating the key tasks a user will perform with a system

- They can also help define the usability testing regime

- To create user scenarios is a simple process and should be used for developing and iterating interactive products

https://www.interaction-design.org/literature/topics/user-scenarios

Task Analysis

- The process of learning about ordinary users by observing them in action to understand in detail how they perform their tasks and achieve their intended goals.

- Helps identify the tasks that product/service must support

- Helps support other aspects of the user-centered design process

- It is important to perform a task analysis early in your process, in particular prior to design work

https://www.usability.gov/how-to-and-tools/methods/task-analysis.html

Types and benefits of Task Analysis

Hierarchical task analysis

- Types

Cognitive task analysis

- Help support several other aspects of the user-centered design process, including:

    - Requirements gathering

    - Developing structure

    - Prototyping

    - Usability testing

https://www.usability.gov/how-to-and-tools/methods/task-analysis.html

- **It is also useful for** understanding:

  - Users' goals and what they are trying to achieve

  - The steps that users currently take to achieve their goals

  - The personal, social and cultural experiences that users
    bring to the tasks

  - The influence of the physical environment on the users
    while attempting to meet a goal

https://www.interaction-design.org/literature/article/task-analysis-a-ux-designer-s-best-friend

- **Task analysis may be performed**:

   - in a more formal way (e.g. Hierarchical Task Analysis)

   or

   - in a **more informal way**:

   - using several different methods

- First **use the 11 questions (at least 1, 2, 3 and 5)**

- Then **decompose the main tasks**

Standard/Informal Questions to be answered

1. **Who is going to use the system?**
2. **What tasks do they now perform?**
3. **What new tasks are desired?**
4. How are the tasks learned?
5. **Where are the tasks performed?**
6. What is the relationship between customer and data?
7. What other tools does the user have?
8. How do users communicate with each other?
9. How often are the tasks performed?
10. What are the time constraints on the task?
11. What happens when things go wrong?
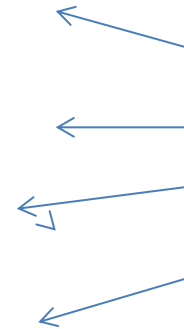
Minimum set of questions to be answered

1. Who is going to use the system?

- Use **all the information obtained previously about the users** (e.g. to develop the personas), concerning:

   age, needs, motivations, background, experience,   technology literacy, physical characteristics…

2. What tasks do they now perform?

- Identify the **tasks that users perform currently**, without using the system under development, including:

   relative importance, frequency of performing the tasks, if they are performed by one or more users, …

3. What new tasks are desired?

- Identify new tasks that might empower the users taking advantage of the new way of performing the tasks

- Be careful and prioritize the new tasks to support …

5. Where are the tasks performed?

- Observe the environment where users currently perform the tasks

- Identify other activities, the type of space (office, shop floor, hospital, class room, shopping mall, …), noise, light and dust conditions, stress level, …

# How to conduct a Task Analysis

- Decompose a high-level task into the following steps:

  - Identify the task to be analyzed

  - Break this high-level task down into 4 to 8 subtasks

  - Draw a layered task diagram of each subtasks

  - Produce a written account as well

  - Present the analysis to someone else who knows the tasks

- The decomposition level of detail should be coherent across subtasks

http://www.usabilitybok.org/task-analysis

Example of a Hierarchical Task Analysis
(it will be addressed later in the semester)

Diagram for the goal: warm up a furnace
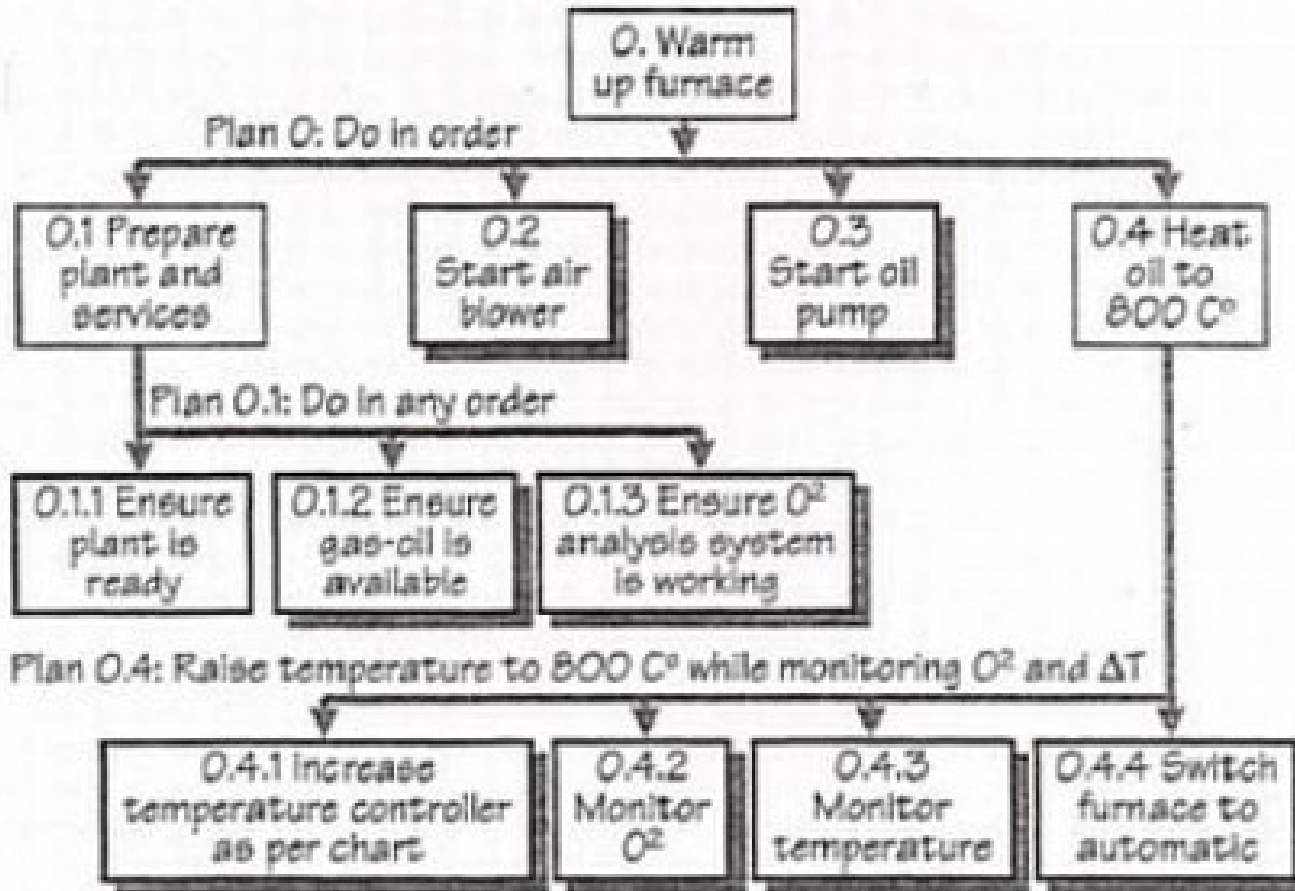
Task Analysis: The Take Away

- Is one of the most powerful tools in UX design

- It is not hard to get to know how to do it

- The difficult part is remembering to keep the user's perspective

- It is useless when it is not backed by rigorous user research

- is not a one-off process; can be repeated later in the process

- It requires time, resources, people and budget. Be sure to have a sufficient amount of all

- Like any other activity in UX design!

https://www.interaction-design.org/literature/article/task-analysis-a-ux-designer-s-best-friend
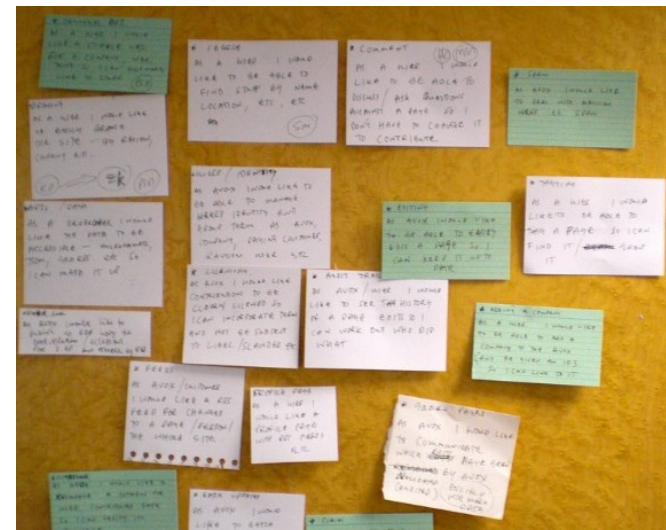
54

User Stories

Emerged from agile development methods, are **short concepts or descriptions of features customers would like to see**

Should:

not go into too much detail

be written:

- from the **user's point of view**

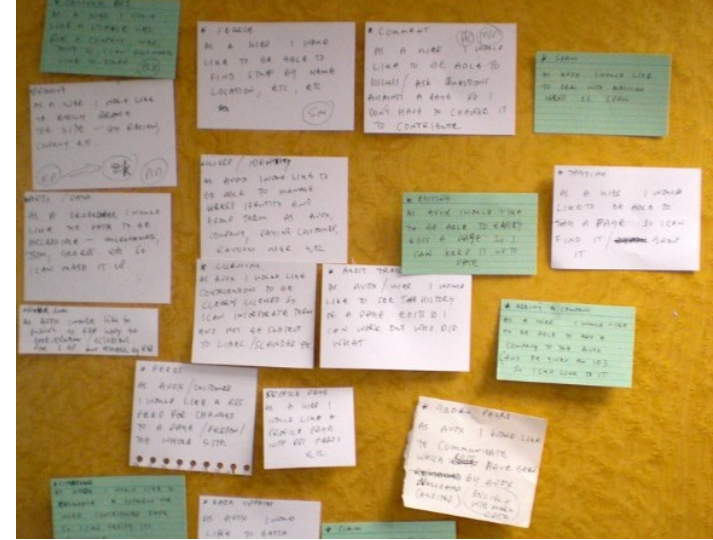- **with the client** and team members

User Stories

"As a <type of user> I want <some goal>
   so that <some reason>."

Should be written with the **minimum amount of detail** necessary to fully **encapsulate the value that the feature** is meant to deliver

https://manifesto.co.uk/how-much-detail-should-a-user-story-have/
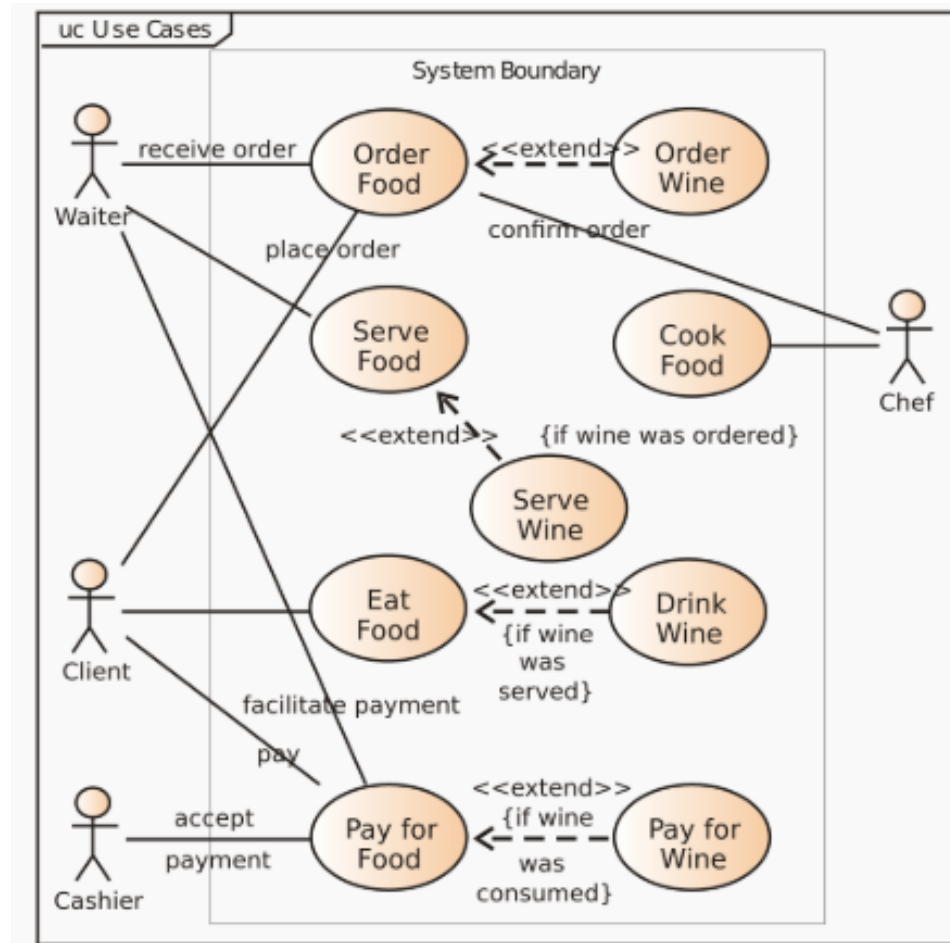https://www.interaction-design.org/literature/topics/user-stories
https://www.interaction-design.org/literature/article/user-stories-capturing-the-user-s-perspective-quickly-and-simply

Use Cases

A common way for developers to explain user scenarios using UML

57

Scenarios, User stories and Use cases

- Scenarios are created by user researchers to help communicate with the design team

- User stories are created by project/product managers to define the requirements prior to a sprint in agile development

- Use cases are created for developers to help with testing

- The difference in target audience means that the structure and information contained in the three approaches also varies.

Example

Scenario

"Jim, a second year internal medicine intern at Mount Pleasant Hospital, walks into the room of his patient, Andrew Ross. Since Andrew stayed the night in the hospital, Jim needs to review Andrew's medical records to see if the nurses on the night shift had checked in and recorded any changes in Andrew's condition."

User Story

As a doctor, I need to get up to medical date records so that I know how to proceed with my patients' treatment

(it does not reflect the context of use)

Use case: *Review Records*

Actor: *Doctor*

Steps:

Doctor walks into room

Doctor sees patient in bed

Doctor identifies patient in bed

Doctor sees medical charts on foot of bed

Doctor gets medical charts from foot of bed

Doctor opens medical charts

Doctor reads medical charts

Doctor changes pages to continue reading

Doctor closes medical chart

https://www.akendi.com/blog/scenarios-user-stories-and-use-casesoh-my/
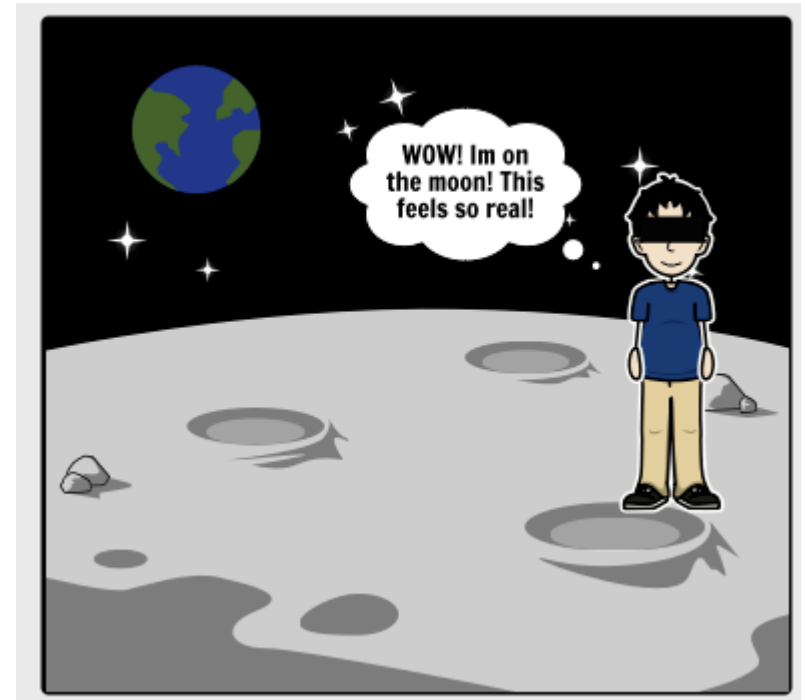
Storyboards

Are early visual forms of an experience

Derived from the film industry

Particularly useful for VR

The user can be shown directly interacting with objects

https://www.interaction-design.org/literature/topics/storyboards

https://www.storyboardthat.com

2nd practical assignment – next steps

Lab 5 - Presentation of the analytical evaluation assignment

Limit for selection of the 2nd assignment topic

🅰 2nd assignment - Development of an interactive application

🔗 2nd Assignment - topic selection

📁 Examples of mini-projects 2020

Lab 6 - User Centered Design (UCD). 2nd Assignment

- Introduction to personas and scenarios and task analysis

- Introduction to the technologies to be used

🅰 How to do a requirement analysis

Interacção Humano Computador 2021

Goals/motivation → Personas → Scenarios → Tasks analysis → Requirements ⇒ LF Prototyping

Requirement analysis

Topics

- Personas
- Scenarios
- Task Analysis
- Requirements
- Prototyping

Main bibliography

- Cooper, A., R. Reimann, and D. Cronin, *About Face 3 The Essentials of Interaction Design*. Wiley Publishing, Inc., 2007
- Mayhew, D., *The Usability Engineering Lifecycle*, Morgan Kaufmann, 1999
- Norman, D., *The Design of Everyday things, Revised and Extended Edition,* Basic Books, 2013
  https://books.google.pt/books/about/The_Design_of_Everyday_Things.html?id=I1o4DgAAQBAJ&printsec=frontcover&source=kp_read_button&redir_esc=y#v=onepage&q&f=false
- Sharp, H., Y. Rogers, *Interaction Design : Beyond Human-Computer Interaction*, 5th ed., Wiley, 2019
  https://books.google.pt/books/about/Interaction_Design.html?id=HreODwAAQBAJ&redir_esc=y
- Sommerville,, I., *Software Engineering*, 9th ed., Addison Wesley, 2010
  http://ifs.host.cs.st-andrews.ac.uk/Books/SE9/WebChapters/PDF/Ch_29%20Interaction_design.pdf

Interesting Links

- https://www.nngroup.com/
- http://www.usability.gov/;
- https://www.usability.gov/how-to-and-tools/index.html
- http://uxpa.org/
- https://www.w3.org/WAI/redesign/ucd