



Vietnam National University of HCMC
International University
School of Computer Science and Engineering



UI/UX Design & Evaluation

★ Synthesizing: Define Needs & Tasks ★

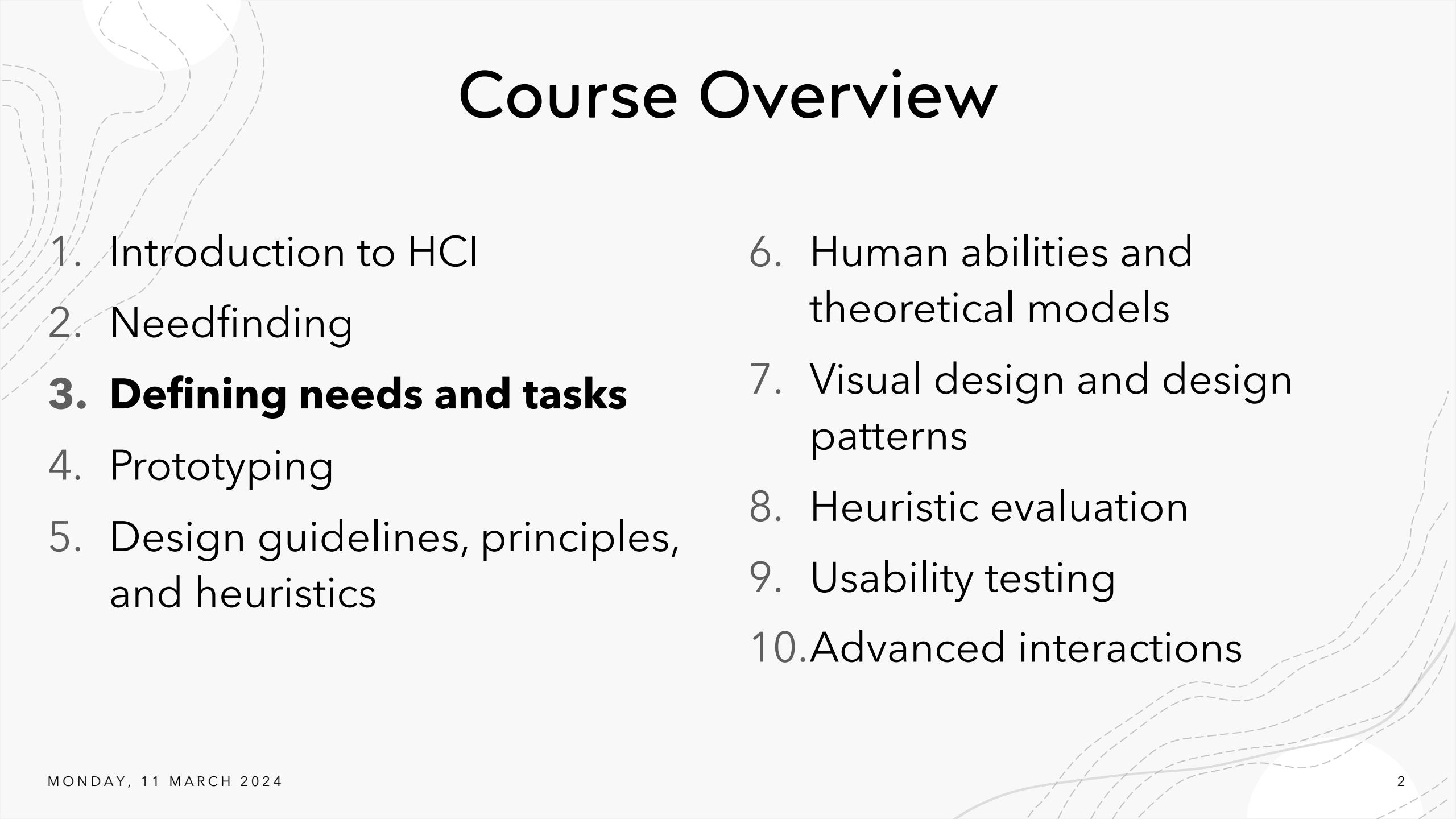
Dr Vi Chi Thanh - vcthanh@hcmiu.edu.vn

<https://vichithanh.github.io>



SCAN ME

Course Overview

- 
1. Introduction to HCI
 2. Needfinding
 - 3. Defining needs and tasks**
 4. Prototyping
 5. Design guidelines, principles, and heuristics
 6. Human abilities and theoretical models
 7. Visual design and design patterns
 8. Heuristic evaluation
 9. Usability testing
 10. Advanced interactions

Goals

Introduction to User-Centred Design Methods:

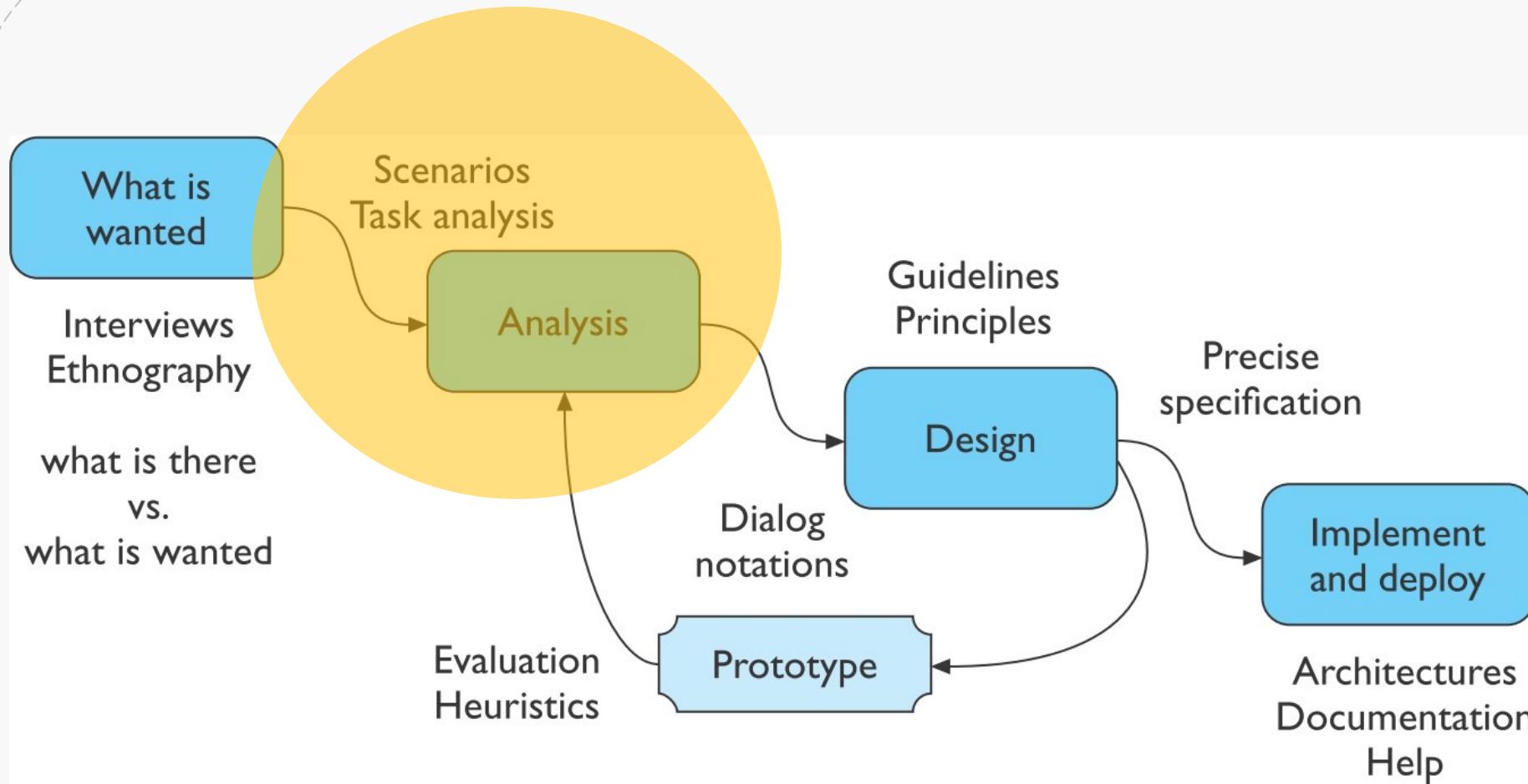
- User Personas
- User Scenarios
- User Requirements
- Hierarchical Task Analysis (HTAs)

GOALS IN UI/UX DESIGN

- **Final goal: Making money!!!**
- How to do this?
- → Need to make good product.
- What is a good product?
- → A product that can **give user an “optimal” experience**



Human-Centered Design Process

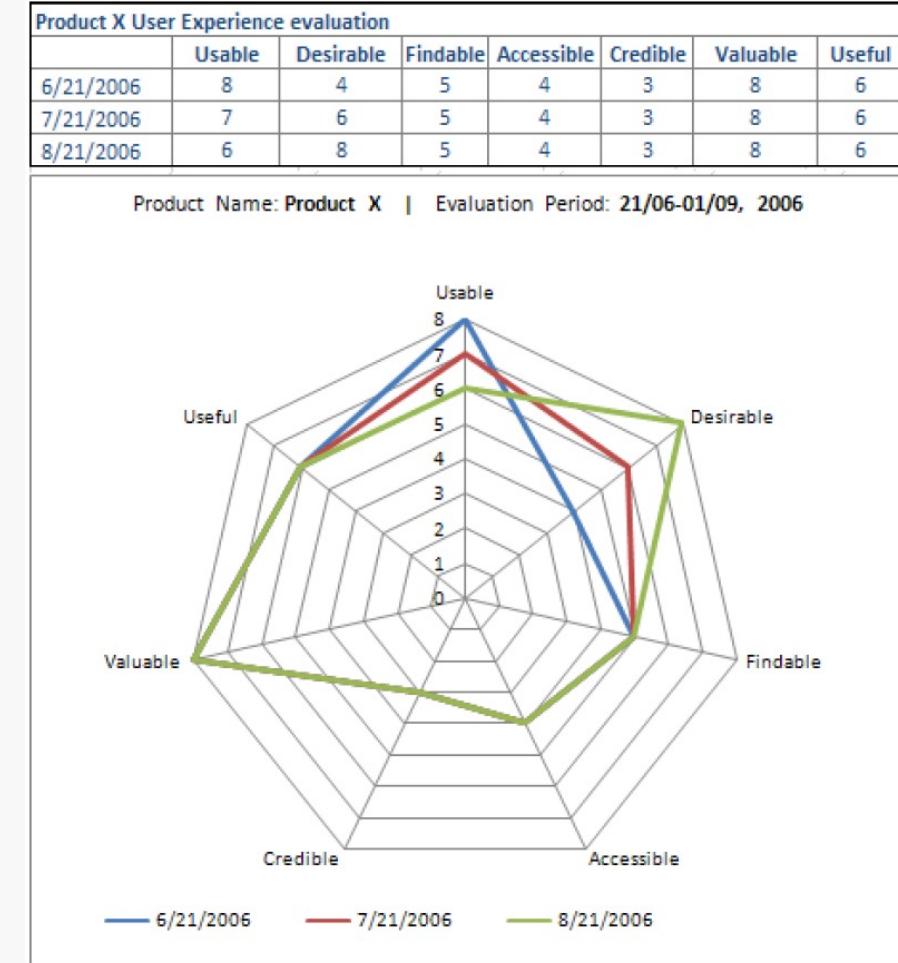


A portrait of an elderly man with a white beard and glasses, wearing a blue jacket, pointing his right index finger towards the camera.

NN/g

Don Norman: The Term UX

Optimal? Measurable?



Factors influencing optimal user experience

- Factor that we, as a product designer can control is
 - ➔ User Interface design factors

1. a product that functions

- ➔ usefulness

2. a product that functions more efficiently and easily

- ➔ usability

3. a product that looks good

- ➔ affectability

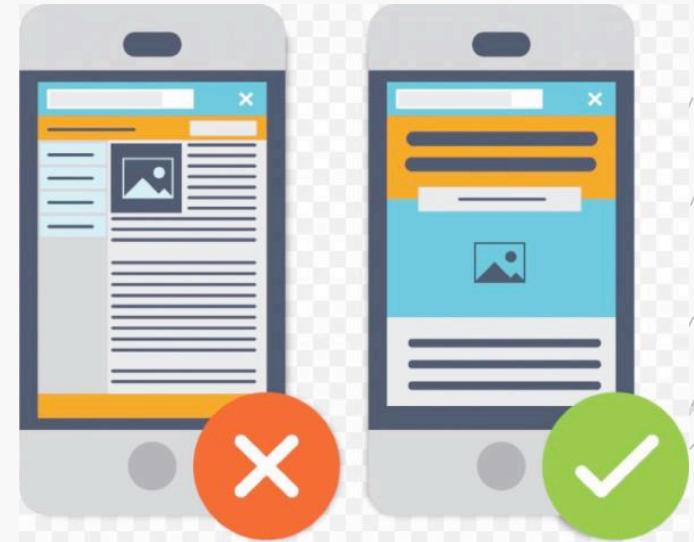
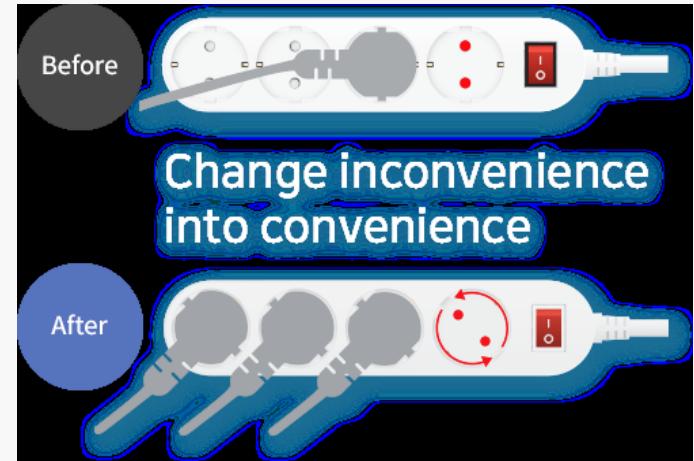
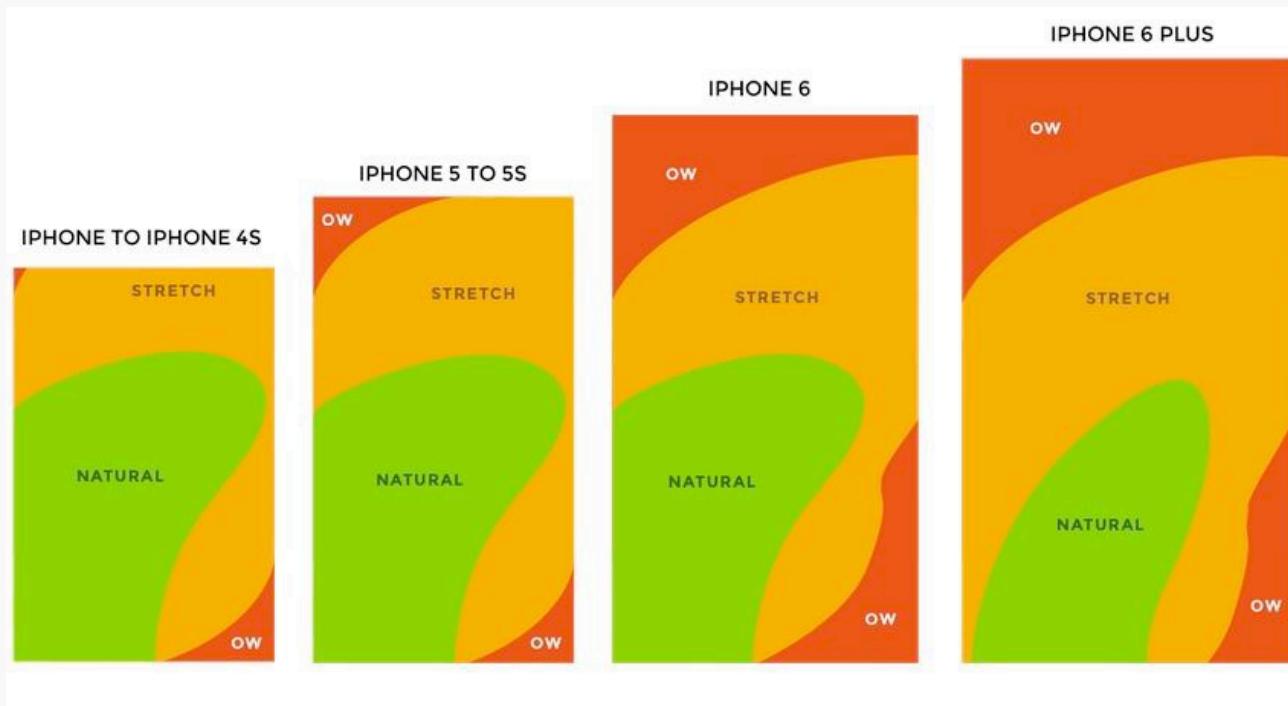
Usefulness

- **What user demands == what product can provide**
- Should be the same not only in the artifacts' actual function, but also in user's prediction before using and user's satisfaction after using.
- **Phone**
- Text message example
- **Car**
- Auto parking example



Usability

- According to the ISO: The effectiveness, efficiency, and satisfaction with which specified users achieve specified goals in particular environments



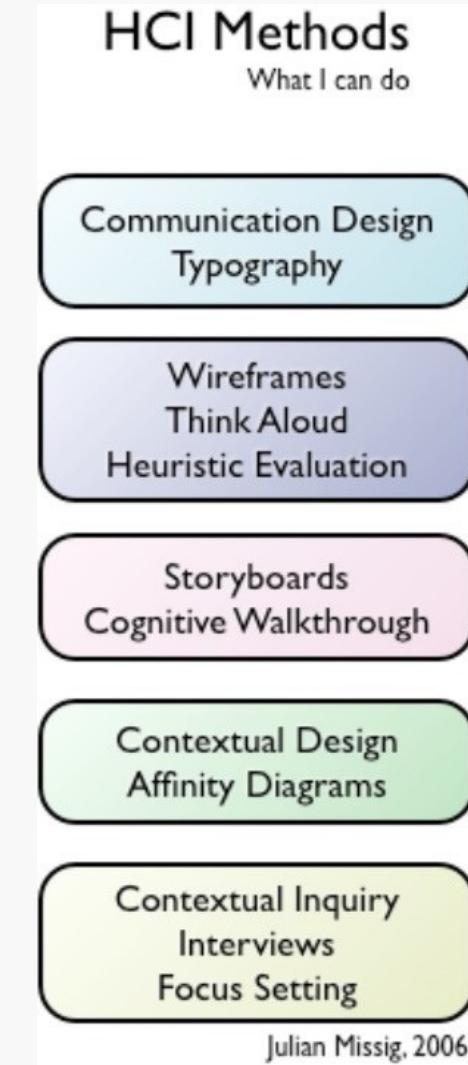
Affectability

- Aesthetic aspect of the product
- User's visceral level preference
- Connected to human instinct



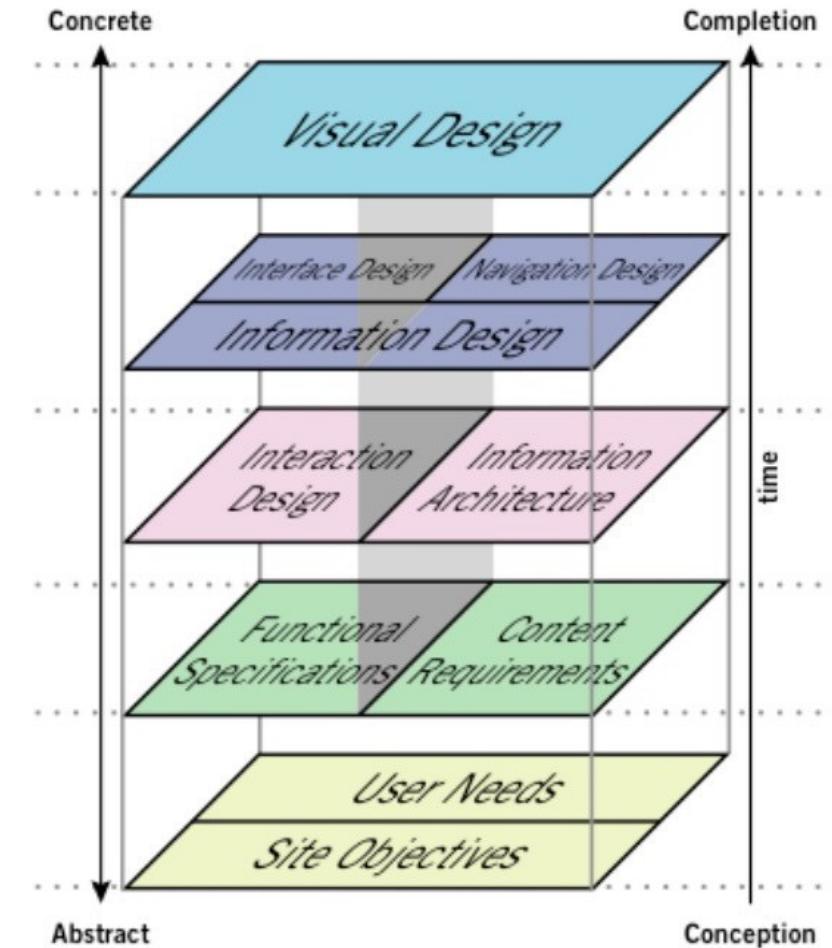
The Elements of User Experience

MONDAY, 11 MARCH 2024



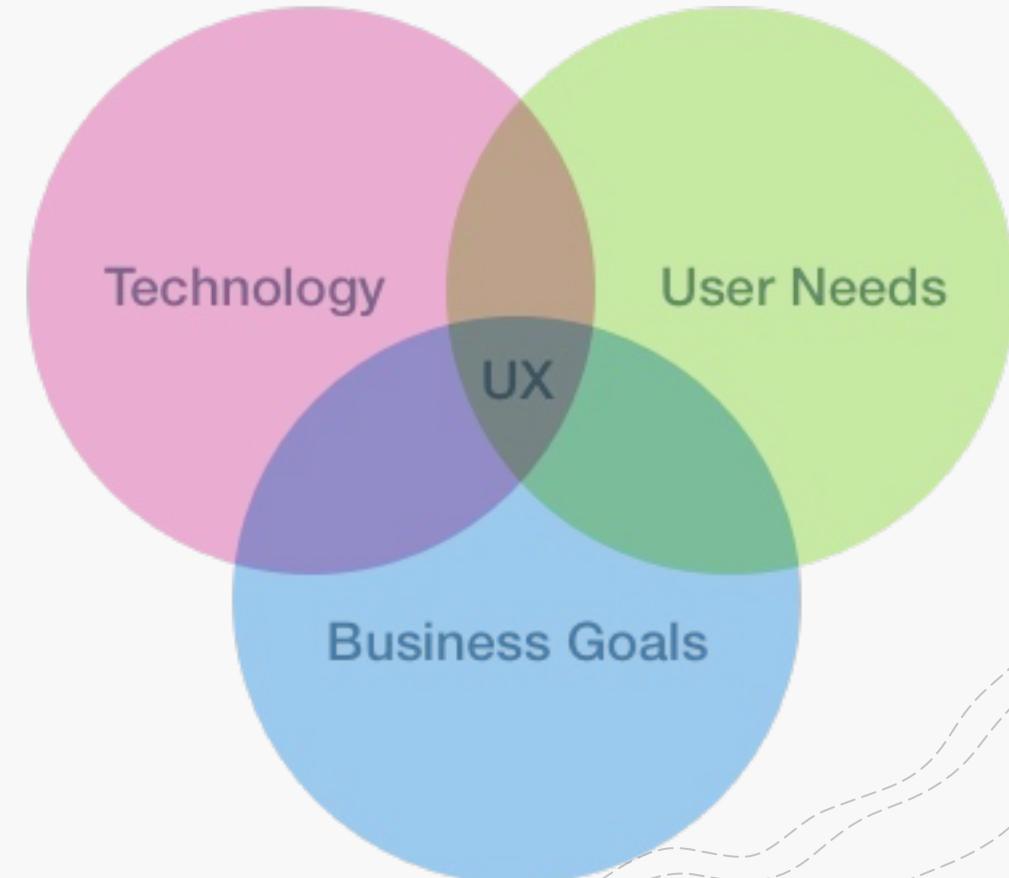
Elements of User Experience

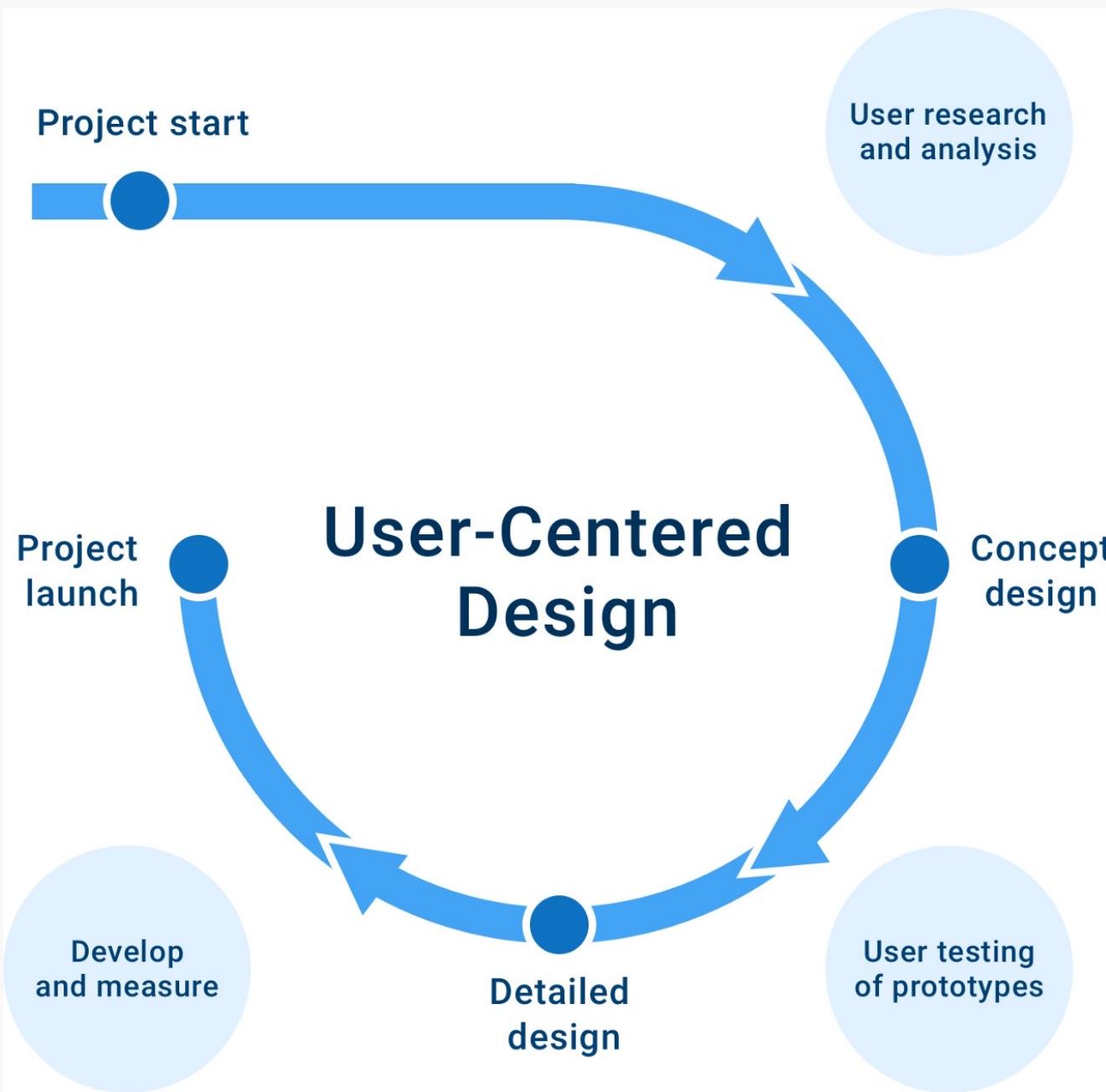
Jesse James Garrett, 2000



User-Centred Design Process

- **User-Centred Design (UCD)** is an iterative design process in which designers focus on the **users** and **their needs** in each phase of the design process.

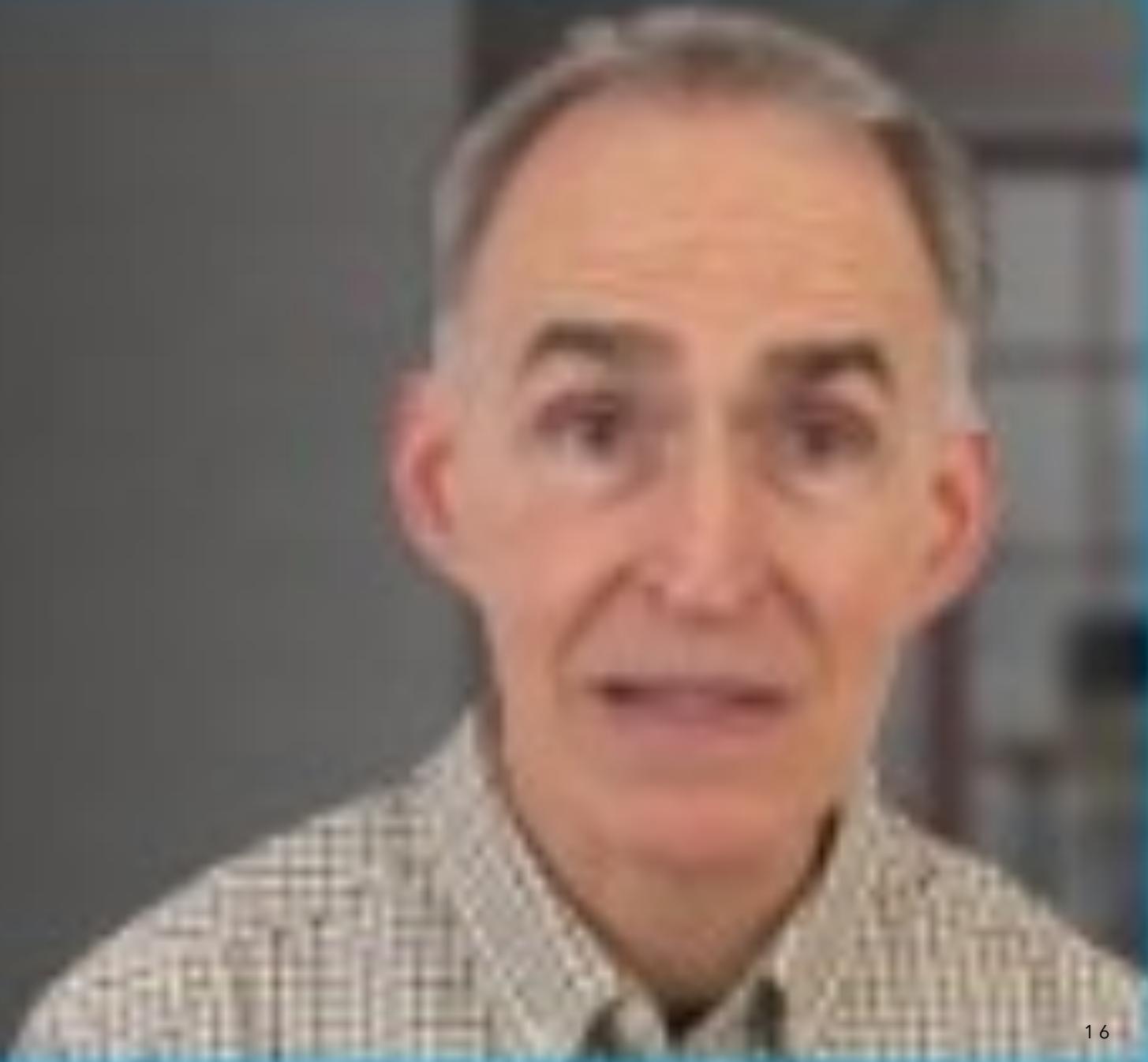




Human- Centered Design



UNIVERSITY OF TECHNOLOGY
SYDNEY



Identifying needs

1. User driven

- Analysis of users (who are users?)
- Learn from people, experts, similar settings, and immerse yourself

2. Technology driven

- Technology analysis
- High tech does not always mean high UX

3. Concept driven

- Need deep understanding of users

Bringing requirements to life

- Augmenting the basic requirements expressed as stories, in Volere template, or in other form
- **Personas**
 - Characterizes someone who might use the product, not specific people
- **Scenarios**
 - An informal narrative story, simple, 'natural', personal, and not generalizable

Scenarios and personas

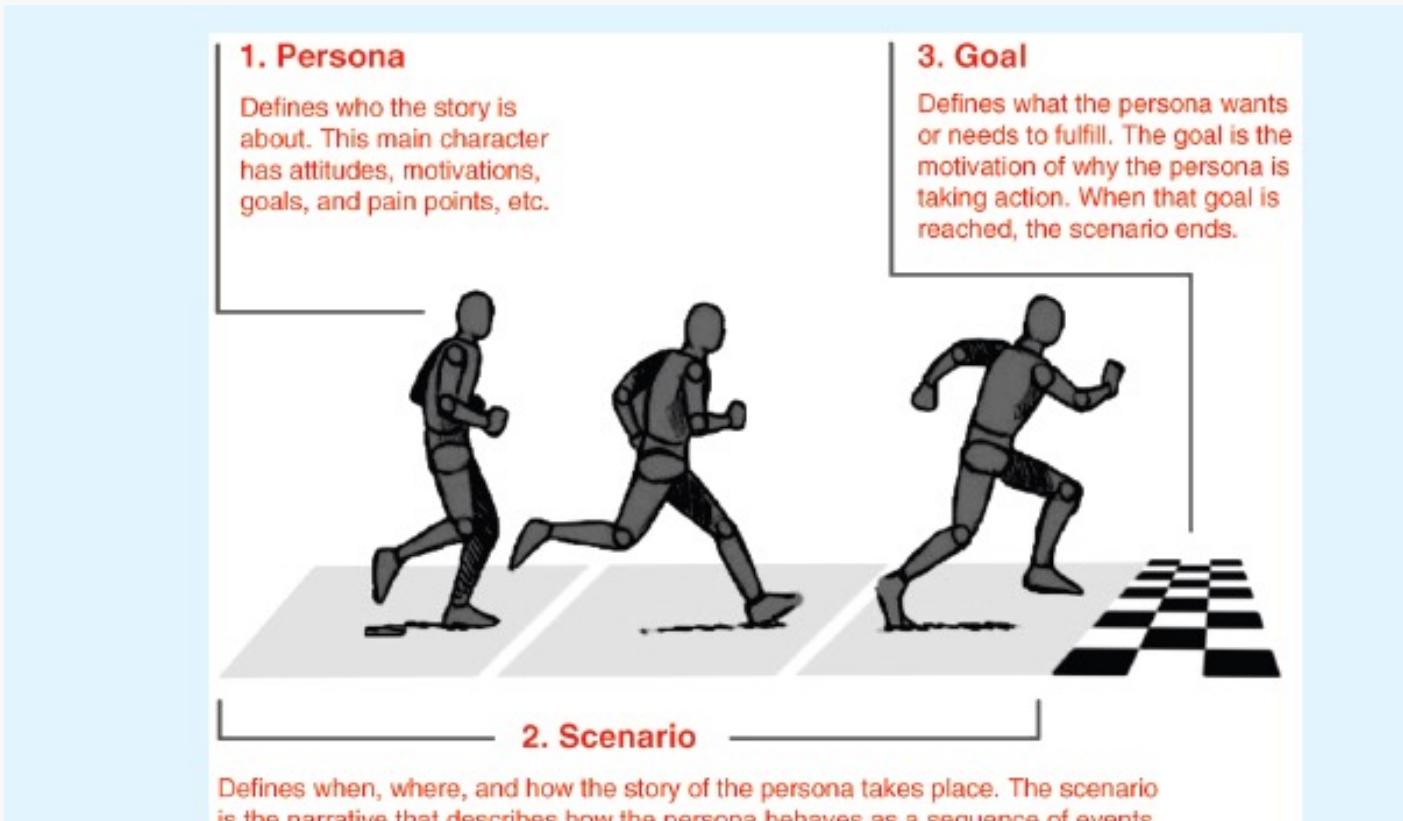


Figure 10.10 The relationship between a scenario and its associated persona

Source: <http://www.smashingmagazine.com/2014/08/06/a-closer-look-at-personas-part-1/>

User Personas

- Personas are realistic fictional characters which **represent different target users**.
- A persona helps UX designers and researchers answer the question: **who are we solving/designing for?**



Personas

- Capture a set of user characteristics (user profile)
- Synthesized from real people based on research
- Typical, not idealized
- Bring to life with name, characteristics, goals, and personal background
 - Relevant to product under development
- Good persona helps designer with design decisions and reminds team about who will use the product
- Develop a small set of personas with one primary

Why are personas useful?

- Personas help you understand **who** you are designing for.
- Personas help you get to know users more closely to create **better experiences**.
- Personas represent a **group of users** with **similar goals and characteristics**.

Using personas in your design process

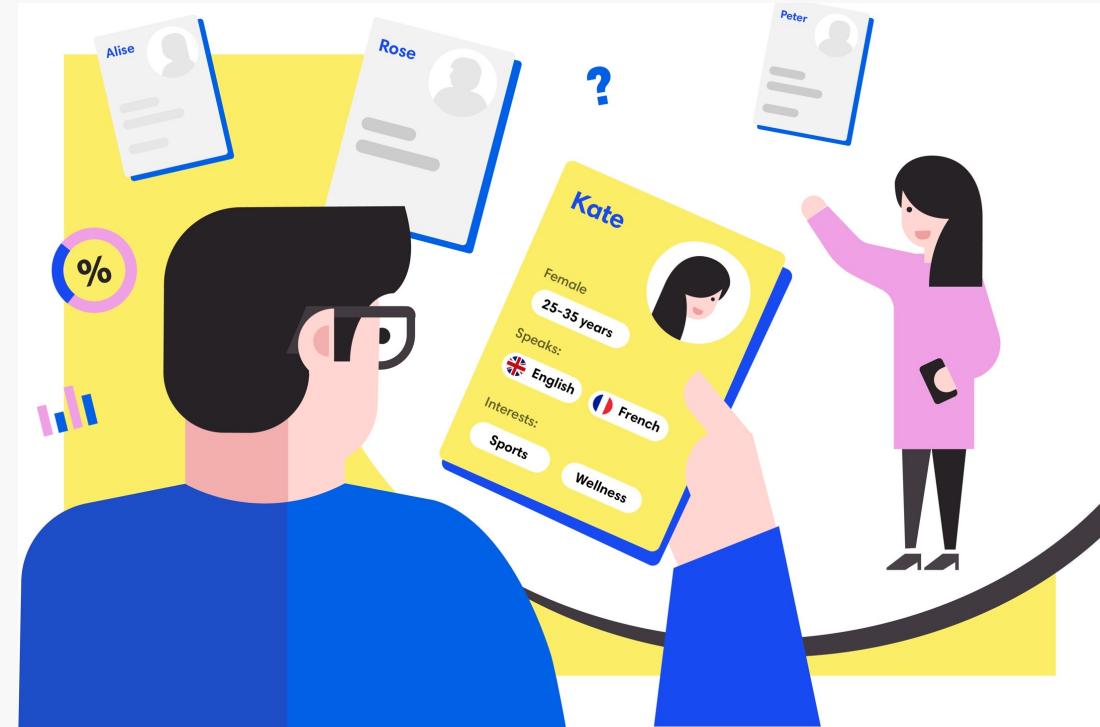
- Don't seek to serve the masses, **focus on target users**.
- Focus on **specific user groups**, their **interests, needs and goals**.
- Try to define your personas **before** you think of design solutions.

Case Study



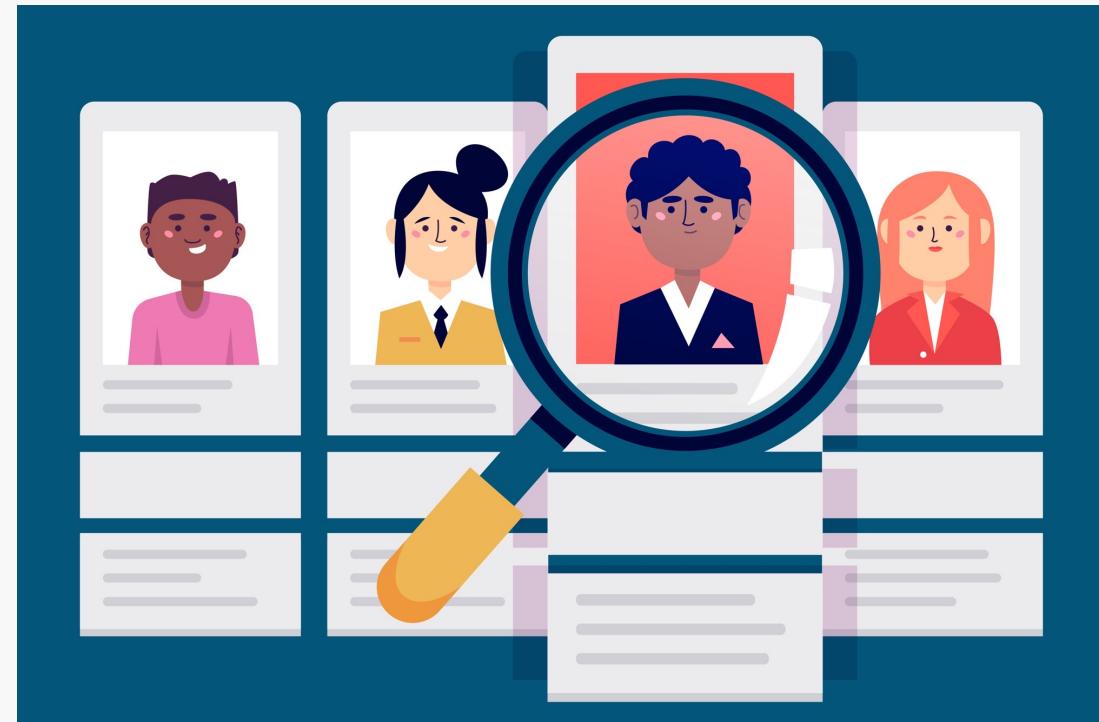
Building your user personas

- Name
- Age
- Occupation
(Income?)
- Education
- Where are they from?
- Personality
- Motivations
- Interests and hobbies
- What other apps do they use?
- What gadgets do they use?
- What are they seeking?



Building your user personas

- Who are they?
- What are their goals?
- What are the barriers preventing them from achieving their goals?



Jill Anderson



"I'm looking for a site that will simplify the planning of my business trips."

AGE: 35

WORK: Regional Director

FAMILY: Married, 1 Child

LOCATION: Austin, Tx

ARCHETYPE: The Frequent Flyer

Organized

Practical

Protective

Hardworking

Bio

Jill is a Regional Director who travels 4-8 times each month for work. She has a specific region in which she travels, and she often visits the same cities and stays at the same hotel. She is frustrated by the fact that no matter how frequently she takes similar trips, she spends hours of her day booking travel. She expects her travel solutions to be as organized as she is.

Personality



Preferred Channels



Goals

- To spend less time booking travel
- To narrow her options quickly

Frustrations

- Too much time spent booking - she's busy!
- Too many websites visited per trip
- Not terribly tech savvy - doesn't like the process

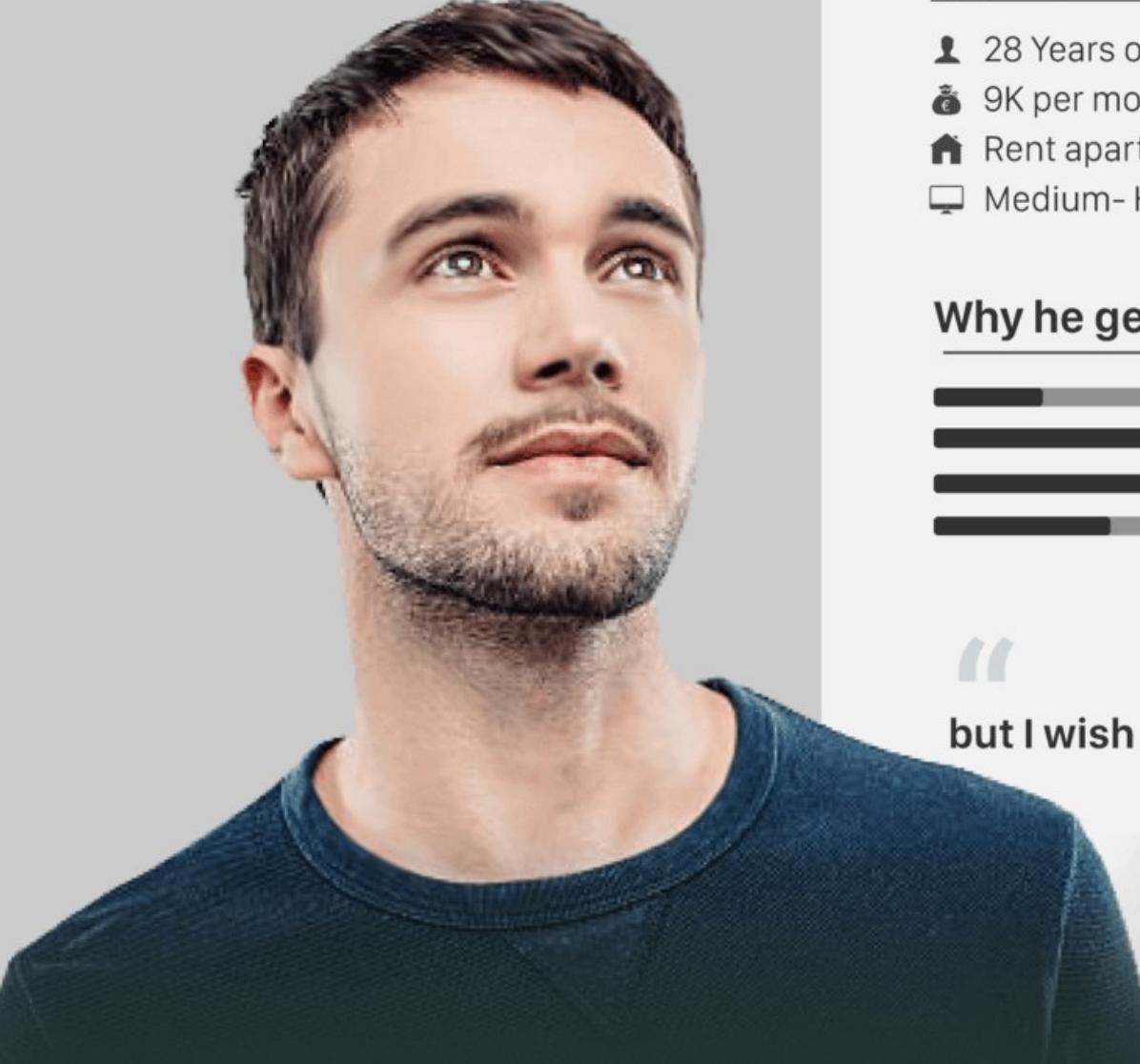
Motivations



Brands

KAYAK Expedia

ACE HOTEL lyft

A portrait of a young man with short brown hair and a light beard, looking upwards and to the right with a thoughtful expression.

Assaf

About

- 👤 28 Years old.
- 💰 9K per month.
- 🏡 Rent apartment in Tel Aviv
- 💻 Medium- High Tech proficiency.

Why he get to the market ?



**I love doing my shopping at the market,
but I wish to finish it quickly so I could get something to eat**

2-3

Per month
Market visits

Core Needs

- Find exactly where are the groceries he wants.
- Remembers where to find each shops he liked.
- Makes shopping faster with direct route.
- Finds a vegetarian place to eat.
- Find a new and recommended place to eat.

Motivation

- Loves the market vibes and variety.
- Usually finds something new to try.
- Easy access to things he needs
- Love the market location and like to hang around after he finishes shopping.

Pain Points

- Have a hard time to find a restaurant that serve vegetarian food .



Lena, 50

Lena works in London as a civil servant. She lives with her partner in a commuter town, and they both own a car.

She commutes to London by train, taking an early morning service which takes over an hour. She leaves the house early and often doesn't get home until after 7pm. She drives 15 minutes to the station and usually arrives 5 minutes before the train departs. It costs a lot to park there everyday, but there are no buses direct from her house to the station.

Lena enjoys her job but finds that she is often so busy travelling from one meeting to another that she is left with little time to complete work. At the weekends, she and her partner drive to the countryside to go walking or to visit friends.

Lena has two Android smartphones – one provided by her employer and one for personal use.



She has an Apple laptop for business use. It has to go everywhere with her as it contains confidential files.



She is always relieved if there is a charger available in the taxi so she can make sure her laptop is charged.



Travel & Transport

Top 3 modes of transport

Train

Shared car

Taxi

Business

Leisure

Holidays

Willingness to share a taxi

Technology and Income

Technology acceptance

Openness to experience

Budget

 ServCity

Figure 5. Example persona "Lena." Photo by [RODNAE Productions](#) from [Pexels](#).

Family traveler



"I want a travel organiser that will offer me a range of potential vacations that suit our needs"

Age: 35

Work: Plumber

Family: Married, two children

Personality



Organised Practical Expects high standard

Goals

- To book comprehensive travel quickly
- To find a trip that meets the needs of the whole family
- To feel supported and guided from the beginning of the booking experience right to the end.

Frustrations

- Wasting time filling in forms
- Too much irrelevant information
- Existing systems tend to be too diverse and complicated

Bio

Will loves to take his family on adventure holidays to explore new challenges. His children, Sky (8) and Eamonn (15) are old enough to take part in several sporting activities and he wants to make the most of this before they no longer want to go on trips with him and his wife, Claire. He likes the fact that choosing travel options is so much easier than it used to be, but is frustrated by the many different sources and disjointed options that this can result in. He wants a travel organiser that can provide clear support for family holidays while offering as wide a choice as possible.

Motivation

Price

Comfort

Choice

Favourite destinations



Task Analysis

How people perform their activities

Task Analysis

- Task Analysis is the study of the way people perform their activities
- Aim is to determine:
 - what they **do** (steps)
 - what things they **use** (artifacts)
 - how well they **succeed** (goals, pain points)

Sample Task: To Clean The House (I)

- Steps:
 - get the vacuum cleaner out
 - fix the appropriate attachments
 - clean the rooms
 - when the dust bag gets full, empty it
 - put the vacuum cleaner and tools away
- Must know and use different artifacts:
 - vacuum cleaners, their attachments, dust bags
 - cupboards, rooms
 - ...

Sample Task: To Clean The House (II)

- Goals:

- Here your point of view comes in
- Removing dust? => **narrow goal**
- Tidying up the house after a party?
- Hosting people for the dinner?
- Having a satisfying evening? => **wide goal**

Sample Task: To Clean The House (III)

- Pain points:
 - Narrow version: Why I need to empty the dust bag?
 - Broader version: Why I need a vacuum cleaner to have the house cleaned up?

Another Example of Task (with Steps)

- A person preparing an overhead projector for use would be seen to carry out the following steps:
 1. Plug in to main and switch on supply.
 2. Locate on/off switch on projector.
 3. Discover which way to press the switch.
 4. Press the switch for power.
 5. Put on the slide and orientate correctly.
 6. Align the projector on the screen.
 7. Focus the slide.

What is a Tasks?

- «A **task** is a **goal** together with some ordered set of **actions**.» (Benyon)

Goal

- A state of the application domain that a work system (user + technology) wishes to achieve.
- Specified at particular levels of abstraction.

Task

- A structured set of activities required, used, or believed to be necessary by an agent (human, machine) to achieve a goal using a particular technology.
- The task is broken down into more and more detailed levels of description until it is defined in terms of actions.

Action

- An action is a task that has no problem solving associated with it and which does not include any control structure.
- Actions are 'simple tasks'.

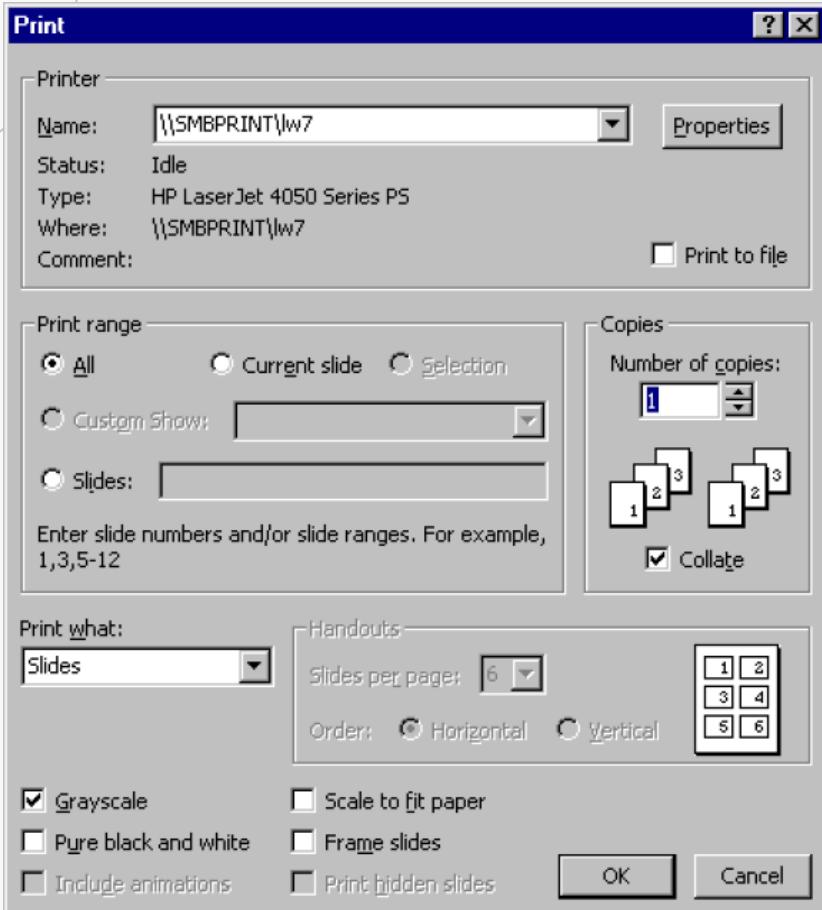
What You Learn with Task Analysis

- What your users' goals can be; what they are trying to achieve
- What users actually do to achieve those goals
- What experiences (personal, social, and cultural) users bring to the tasks
- How users are influenced by their physical environment
- How users' previous knowledge and experience influence:
 - How they think about their work
 - The workflow they follow to perform their tasks
 - The pain points they experience to perform the tasks

Why Is It Useful?

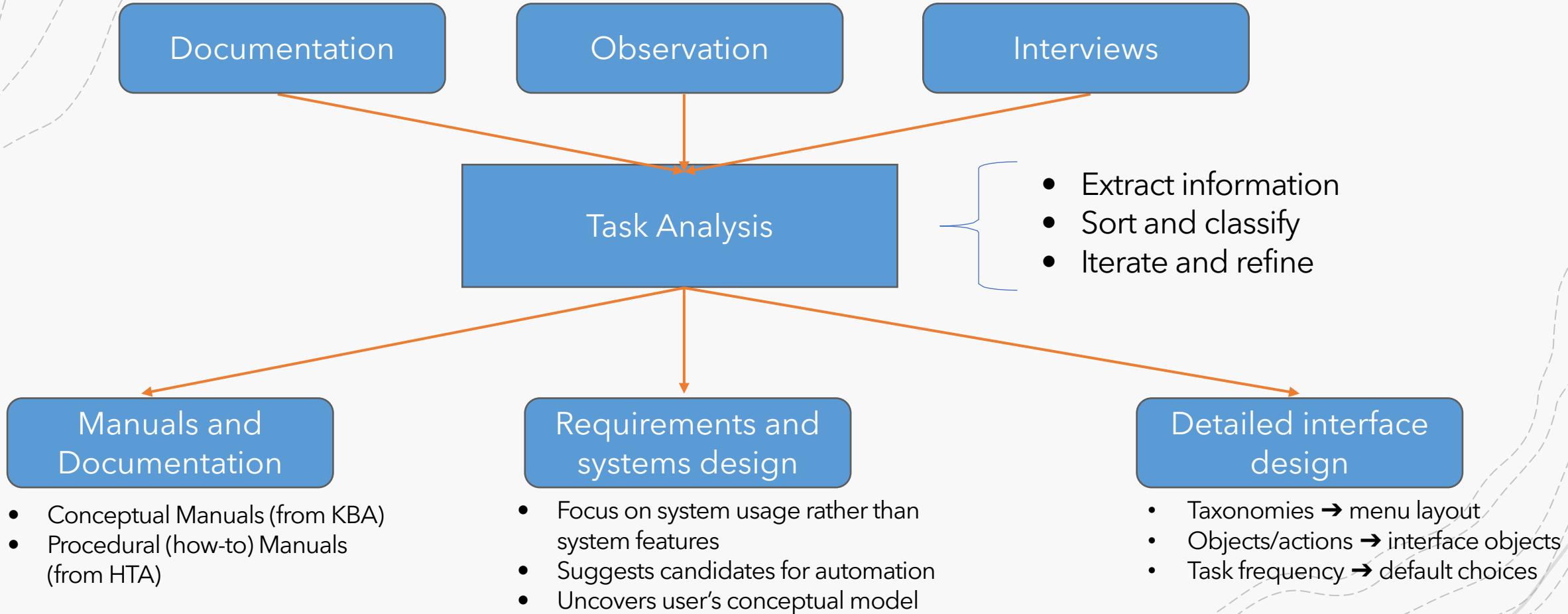
- Task analysis is the process of learning about ordinary users by observing them in action to understand in detail **how** they perform their **tasks** and achieve their intended **goals**
- Tasks analysis helps in:
 - Identifying the tasks that your application **must support**
 - Refining or re-defining your app's **navigation** or **search**
 - Application requirements gathering
 - Developing your content strategy and app **structure**
 - The initial stages of **Prototyping**
 - Performing **usability testing**

Example



- Tasks are used to plan for the layout of the application window
- Proximity and Boundaries reflect the decomposition of tasks
- Order of tasks is not mandatory

Where It Fits



Characteristics of Task Analysis

- Task analysis is easier when you have well-defined **workflows** (e.g., planning a trip somewhere)
 - or **repeated activities**, such as scheduling
- Challenge:
 - We **do not** design tasks, but interfaces
 - Tasks and objects do not map 1:1
 - e.g., a web app has multiple tasks
 - People use the same interface and application to achieve slightly different results or do things differently one another

[Some] Techniques for Task Analysis

- **Task decomposition** – Splitting tasks into sub-tasks and their ordering
- **Knowledge-based techniques** – Any information and instructions that users need to know, and how that knowledge is organized
- **Entity-relationship-based analysis** - identify actors, objects, relationships and their actions
- **Ethnography** – Observation of users' behavior in the use context
- **Protocol analysis** – Observation and documentation of actions of the user. This is achieved by authenticating the user's thinking. The user is made to think aloud so that the user's mental logic can be understood.

Hierarchical Task Analysis

A Task decomposition method

Hierarchical Task Analysis (HTA)

- One possible method for Task Decomposition
- Hierarchical Task Analysis is the procedure of **disintegrating tasks into subtasks** that could be analyzed using the logical sequence for execution
- This would help in achieving the goal in the best possible way

"A hierarchy is an organization of elements that, according to prerequisite relationships, describes the path of experiences a learner must take to achieve any single behavior that appears higher in the hierarchy". (Seels & Glasgow, 1990, p. 94)

Example HTA: How To Clean a House

0. in order to clean the house
 1. get the vacuum cleaner out
 2. fix the appropriate attachment
 3. clean the rooms
 - 3.1. clean the hall
 - 3.2. clean the living rooms
 - 3.3. clean the bedrooms
 4. empty the dust bag
 5. put the vacuum cleaner and attachments away

Plan 0: do 1 – 2 – 3 – 5 in that order.
when the dust bag gets full do 4

Plan 3: do any of 3.1, 3.2 or 3.3 in any order
depending on which rooms need cleaning

- A **hierarchy** of tasks and **sub-tasks**
 - Indentation and numbering denote the levels
- A set of **plans** describing in what **order** and under what **conditions** subtasks are performed
 - Plans are labeled by the task they describe

Notes

0. in order to clean the house
 1. get the vacuum cleaner out
 2. fix the appropriate attachment
 3. clean the rooms
 - 3.1. clean the hall
 - 3.2. clean the living rooms
 - 3.3. clean the bedrooms
 4. empty the dust bag
 5. put the vacuum cleaner and attachments away

Plan 0: do 1 – 2 – 3 – 5 in that order.
when the dust bag gets full do 4

Plan 3: do any of 3.1, 3.2 or 3.3 in any order
depending on which rooms need cleaning

- Not all tasks are mandatory
- E.g., task 4 is needed only if the bag is full.
- The order or operations may be free
- E.g., the rooms may be cleaned in any order
- Could be further refined with additional knowledge or context
- E.g., Plan 3: do 3.1 every day
3.2 once a week
when visitors are due 3.3

Expanding the Hierarchy (I)

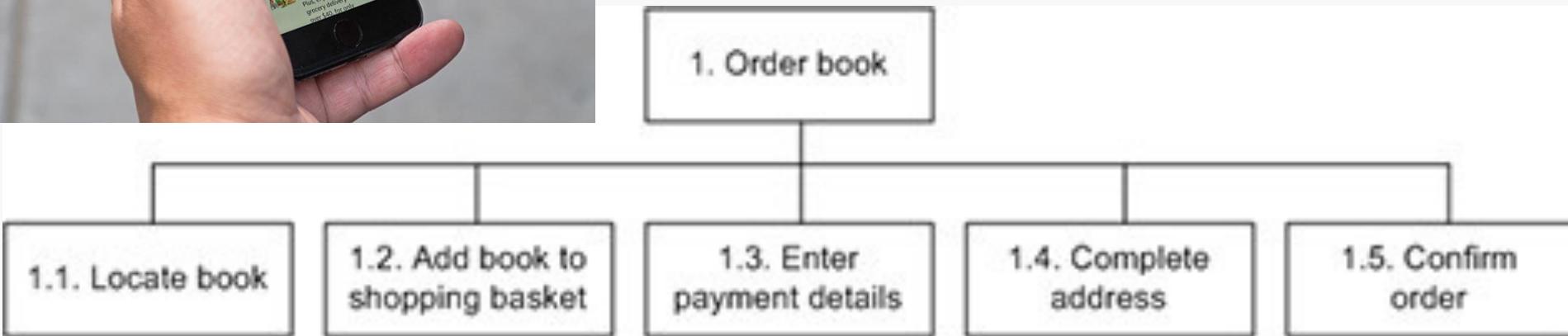
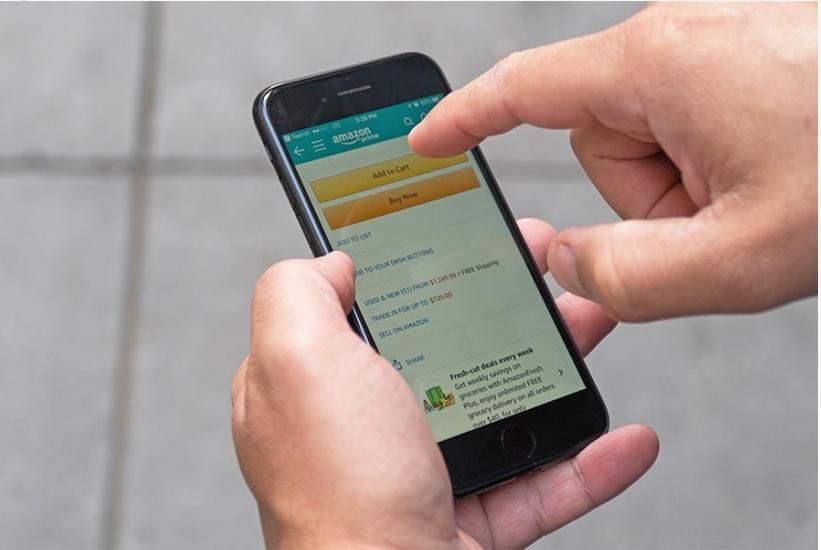
- Each task is de-composed in sub-tasks, iteratively and recursively
 - Answer to the question: «*what subtasks must be accomplished in order to perform the main task?*»
 - The answer will come from direct observation, expert opinion, documentation, ...

- Procedural task knowledge elicitation techniques:
 - Observation, re-enactment
 - Ask about procedures and triggers (pre-conditions)
 - “What happens if X goes wrong?”
 - Sorting steps into appropriate orders

Expanding the Hierarchy (II)

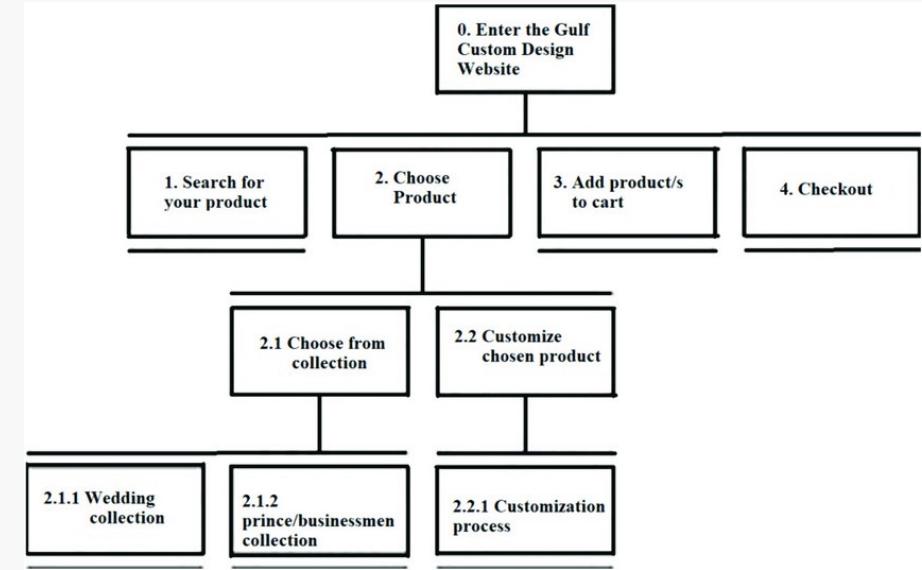
- When is this process stopped?
 - Depends on the intended usage of the HTA (design vs. documentation vs. troubleshooting vs. ...)
 - Expand only **relevant** tasks
 - “Simple” tasks should be **obvious** to the users, and they should not contain hidden **risks** of failure
 - **Motor** actions are the lowest level (not always needed)

Hierarchical Task Analysis (HTA)

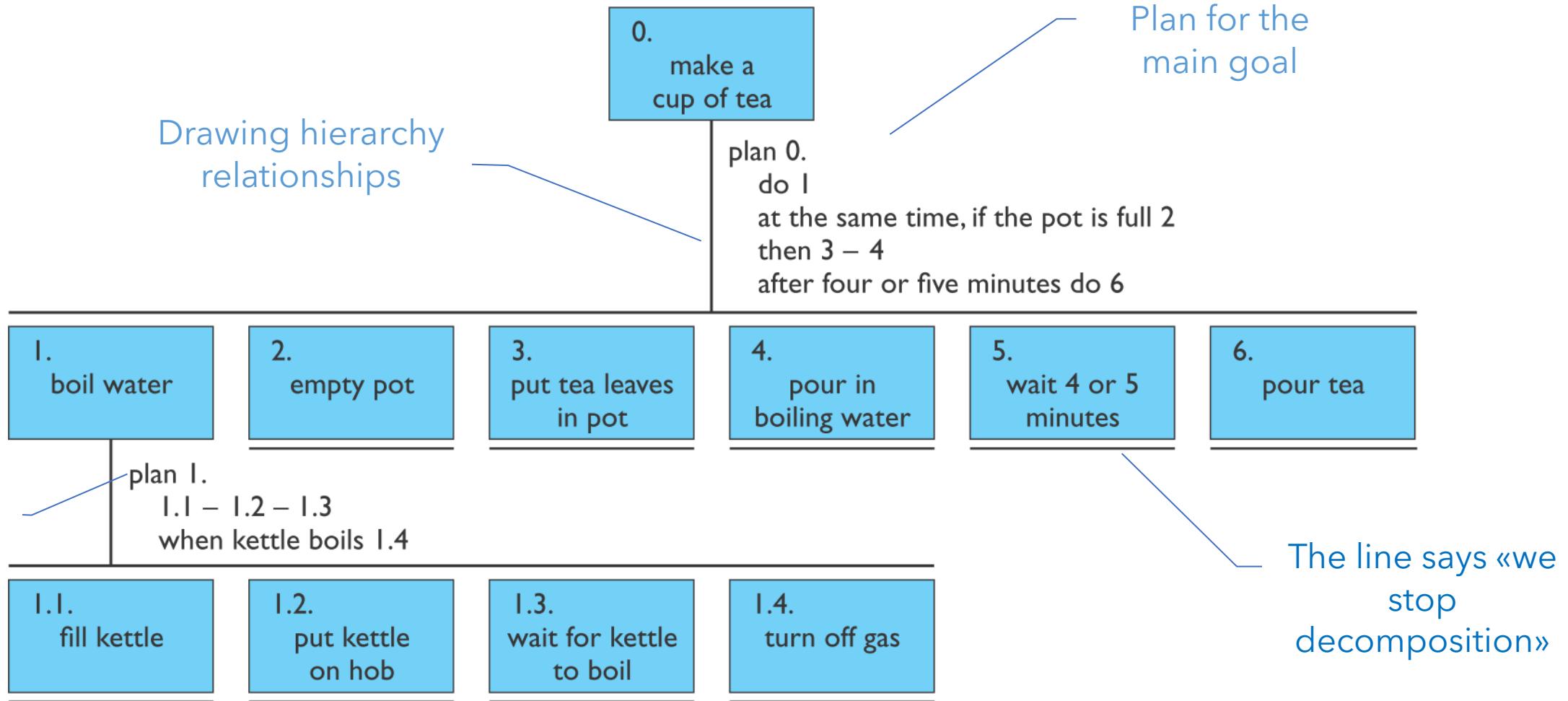


Creating a HTA diagram

- **Step 1:** Plan out your primary, top level pages
- **Step 2:** Add your secondary pages
- **Step 3:** Leave room for content expansion
- ***Important***
 - Keep the main navigation as simple as possible.
 - The more options you add the more distractions you're providing for your users.
 - Remember what your app's primary goal is. Use your top-level pages to push your users to that point.



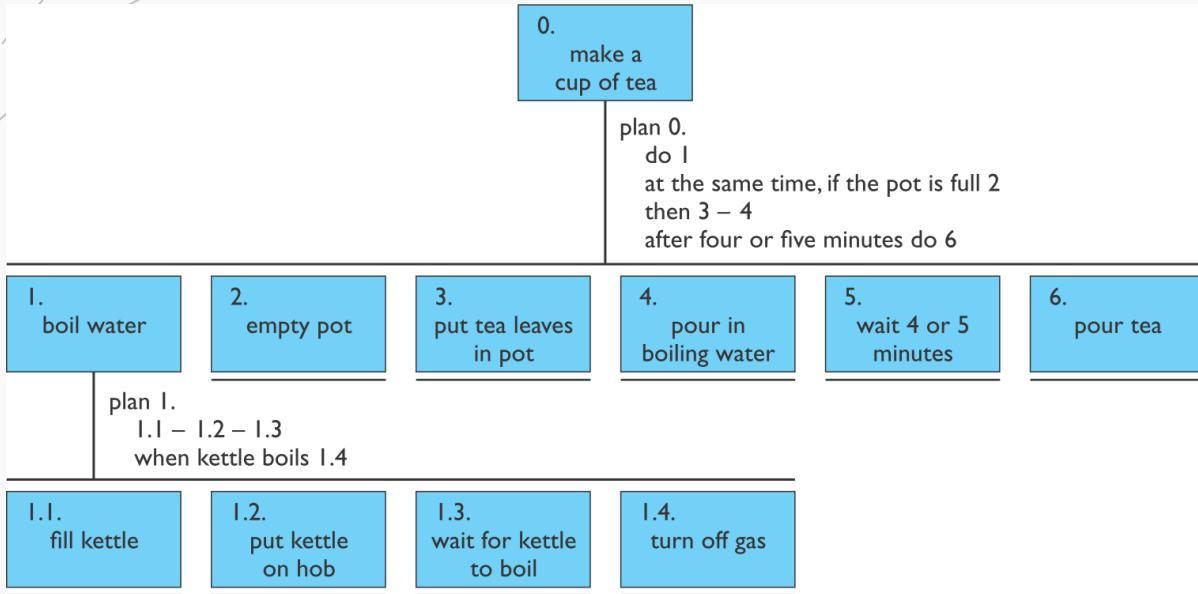
Example



Tasks as Explanation of Goals

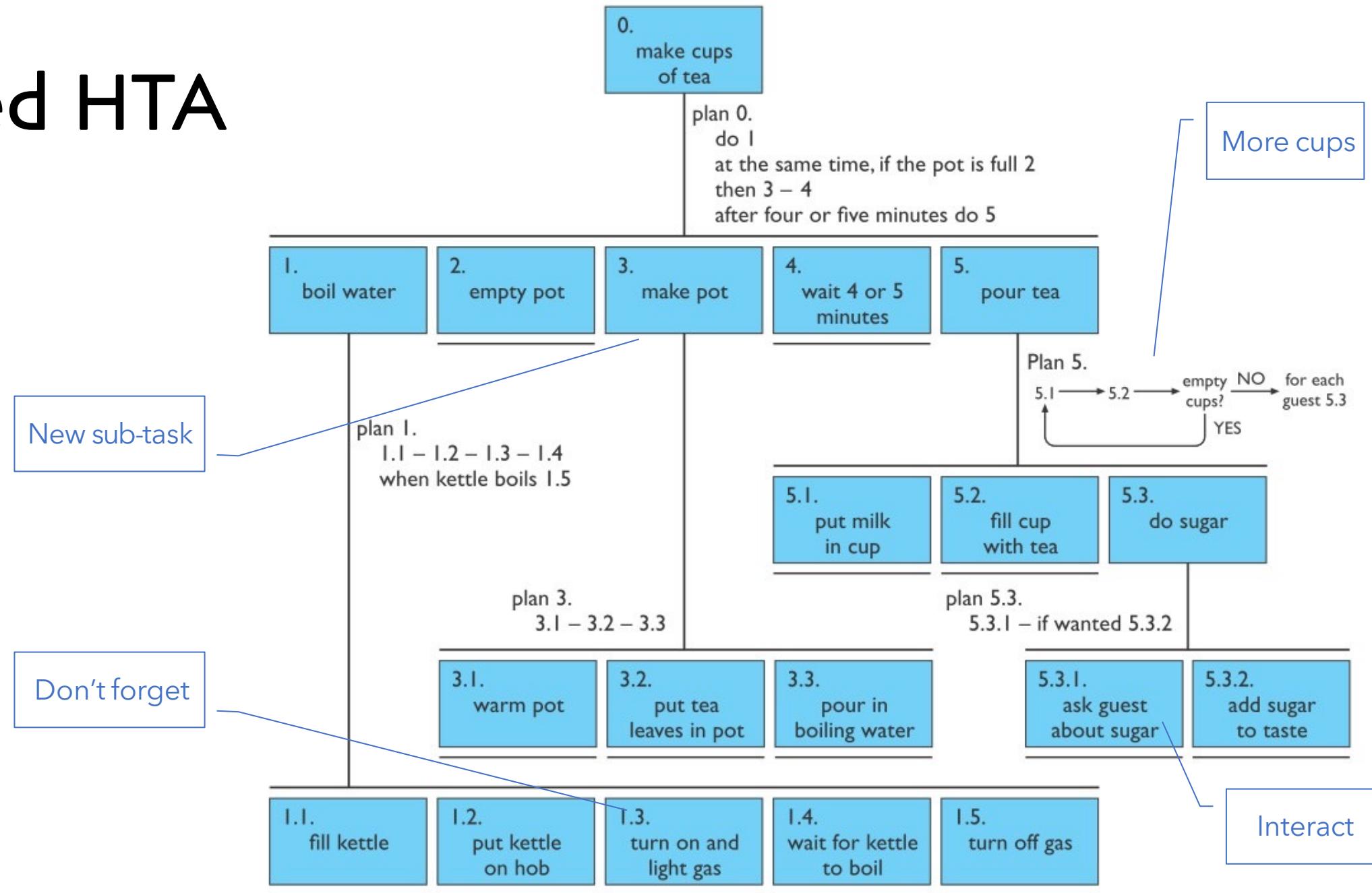
- Imagine asking the user the question:
 - **What are you doing now?**
- For the same action, the answer may be:
 - Typing ctrl-B
 - Making a word bold
 - Emphasizing a word
 - Editing a document
 - Writing a letter
 - Preparing a legal case

Refining the HTA



- Checking matched actions
 - Turn “off” without turning “on”?
- Restructuring
 - “Make pot” might be a meaningful task and group related actions
- Balancing complexity
 - Is “pour tea” simpler than “make pot”?
- Generalizing
 - If we want to make one or more cups?

Modified HTA



Main Constructs To Define Plans

- **Fixed sequence** - 1.1 then 1.2 then 1.3
- **Optional tasks** - if the pot is full 2
- **Wait for events** - when kettle boils 1.4
- **Cycles** - do 5.1 5.2 while there are still empty cups
- **Time-sharing** - do 1; at the same time ...
- **Discretionary** - do any of 3.1, 3.2 or 3.3 in any order
- **Mixtures** - most plans involve several of the above

Knowledge-based Analysis

Knowledge-based Analysis

- Aim to understand knowledge required for a task
 - Provide training material, how-to manuals
 - Take advantage of common knowledge across tasks
 - Organize information and Navigation in the application
- Focus on:
 - **Objects** used in task
 - **Actions** performed

Capturing Knowledge

- Use taxonomies
 - Represent levels of abstraction
 - Organization (grouping) depends on purpose

Example: Taxonomy of Car Controls

motor controls

steering *steering wheel, indicators*

engine/speed

 direct *ignition, accelerator, foot brake*

 gearing *clutch, gear stick*

lights

 external *headlights, hazard lights*

 internal *courtesy light*

wash/wipe

 wipers *front wipers, rear wipers*

 washers *front washers, rear washers*

heating *temperature control, air direction, fan, rear screen heater*

parking *hand brake, door lock*

radio

numerous!

How To Generate Concepts?

- Declarative knowledge elicitation techniques:
 - Established convention, existing documentation
 - Asking users to list objects; card sorting
 - Structured interviews, listing nouns and verbs
- Group concepts according to general-specific relationships

Modeling Depends On Your Goals

- Functional decomposition (what they do)

wash/wipe

wipers *front wipers, rear wipers*

washers *front washers, rear washers*

- Positional decomposition (where they are located)

wash/wipe

front *front wipers, front washers*

rear *rear wipers, rear washers*

- Both views are correct, both are useful in different contexts

Task Descriptive Hierarchy (TDH)

- Task Analysis for Knowledge Description (TAKD) uses three types of branches in TDH taxonomies:
 - XOR – object in exactly one branch
 - AND – object must be in both
 - OR – can be in one, many or none

wash/wipe **AND**

function **XOR**

wipers *front wipers, rear wipers*

position **XOR**

washers *front washers, rear washers*

front *front wipers, front washers*

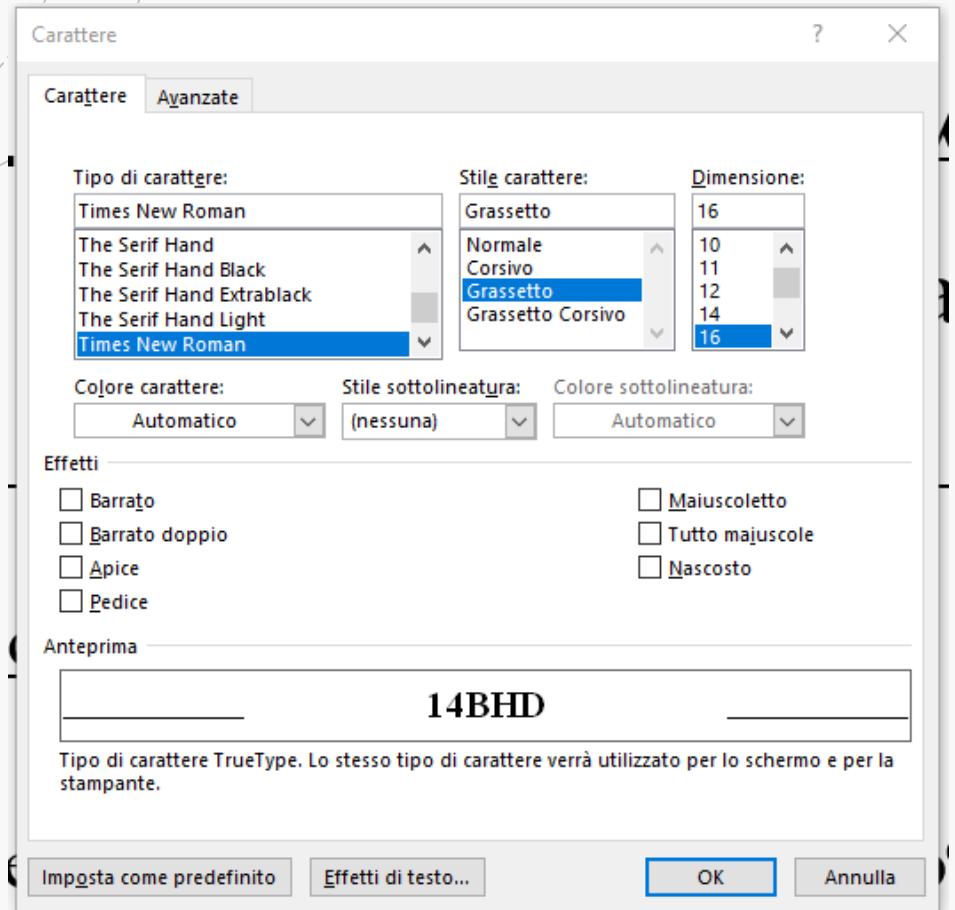
rear *rear wipers, rear washers*

Task Descriptive Hierarchy

- Task Analysis for Knowledge Description (TAKD) uses three types of branches in TDH taxonomies:
 - XOR – object in exactly one branch
 - AND – object must be in both
 - OR – can be in one, many or none
- More complex formalisms (not studied here) aim at capturing several (all?) points of view in a unique model

wash/wipe **AND**
function **XOR**
wipe
front wipers, rear wipers
wash
front washers, rear washers
position **XOR**
front
front wipers, front washers
rear
rear wipers, rear washers

From Concept Taxonomies To User Interfaces



- A Typeface is described by Font, Style, Size, Color
- Different “effects” may be applied
- Moving in the dialog window will explore different related concepts in the taxonomy

Storyboards

Comic book - like representation of user scenarios, with emphasis of how the system supports users in the development of the task

Storyboard

- “A graphical depiction of the outward appearance of the intended system, without any accompanying system functionality”
- A hand-drawn comic that features the execution of a task (like a concrete scenario)
- With a few panels (sequence of sketches) it conveys what a person may accomplish
 - Always include people
- They communicate **flow**, showing what happens **at key points** in time
- No artistic skills are required
 - Not about “nice pictures”
 - About communicating ideas



What To Find In a Storyboard

- Illustrate a goal (for the task)
- How a task unfolds (people interacting among themselves and with devices)
 - Repeated for all significant steps
- At the end, how they accomplish their goals (satisfactory outcome)
- Storyboards are **all about tasks**

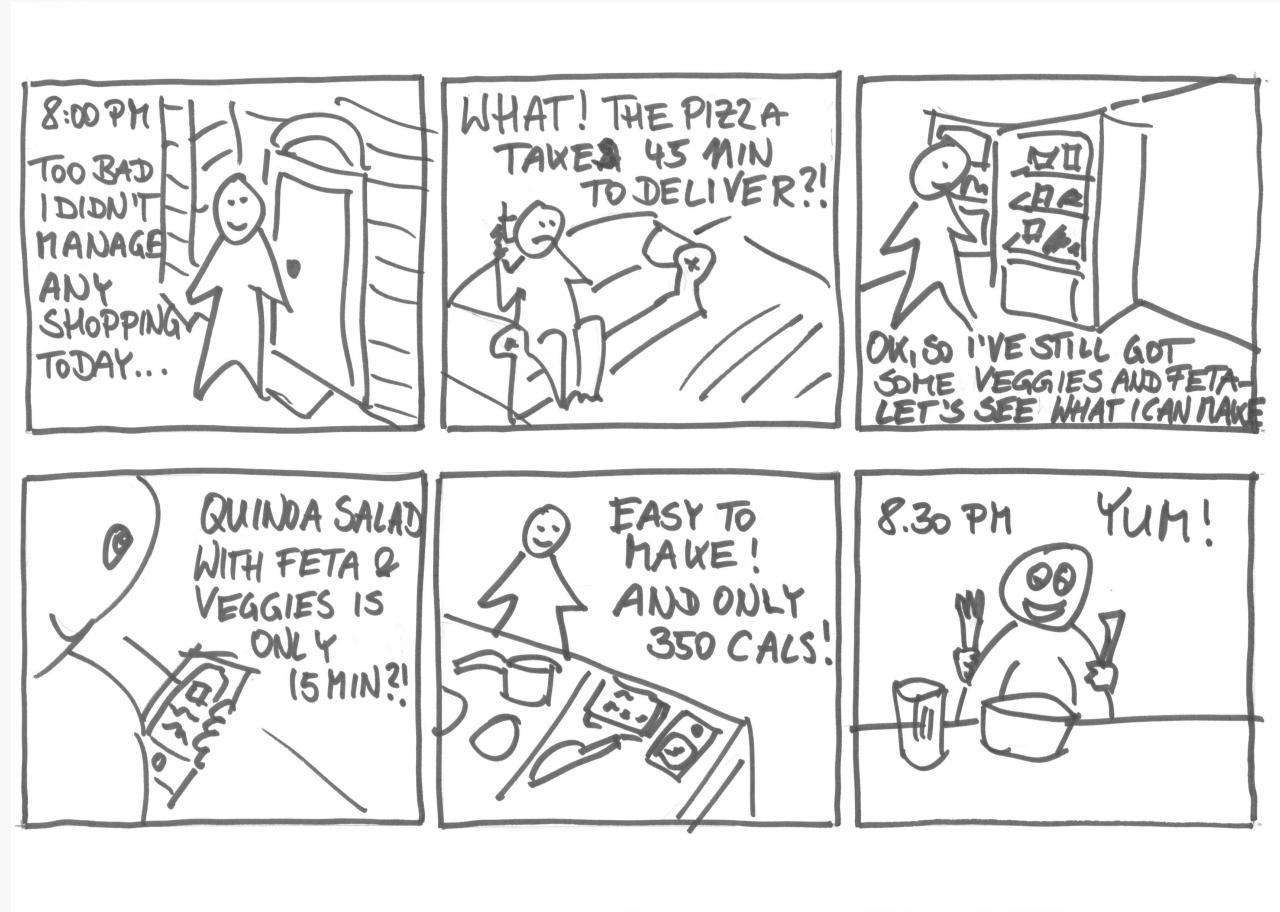
Example

This storyboard illustrates how the app had already downloaded the daily recipe to the user's smartphone, so he could look it up and check the shopping list while on the underground, before shopping for ingredients and making a healthy meal.



Example

This storyboard illustrates how the app can show the user that a home cooked meal can be quicker than ordering food delivery, using left over ingredients in the fridge.



Storyboards Should Convey...

- Setting
 - People involved
 - Environment
 - Task being accomplished
- Sequence
 - What steps are involved?
 - Not the detailed UI
 - What role the UI plays in helping users achieve their goal?
 - What leads someone to use the system?
 - The “trigger” for the task
 - What task is being illustrated?
- Satisfaction
 - What's the motivation for the user?
 - The end point to reach after all the steps
 - What's the end result?
 - What need are you “satisfying”?

Handling Dynamicity In Storyboards

- Traditional storyboarding
 - “Comic book” conventions: actors, speech bubbles, background
 - Notes attached to each scene explaining what is happening
- Scored storyboards
 - When the user interface is highly dynamic, or contains specific media elements
 - Add specific annotations focusing on movement, colors, sounds, ...
- Text-only storyboards
 - When the interaction behavior is too complex to compact into an illustration, use a longer text description

Why Hand-drawn?

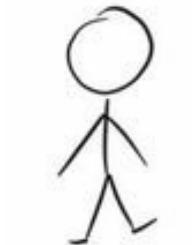
- Quick

- No need to spend time in graphics tools (they would “push” you to focus on details, too)
- Able to experiment different scenarios

- Imprecise

- Users feel free to express more comments and suggestions w.r.t. a more “polished” version
- Focus on the content (the graphics is obviously ignored)
- No distraction by fonts, colors, icons, ...

Drawing Sketching People



Stick People



Block People



Blob People



Star People



Triangle People



Use Your
Imagination



neutral



pointing



ballet

Star man versatility

Benefits of Storyboards

- Emphasize how an interface accomplishes a task
- Focus the conversation and feedback on user tasks
- Get everyone on same page about the app's goals
- Avoid nitpicking about user interface details (buttons, etc.)

Scenarios

Possible sequences of actions for reaching user goals

Scenario

- Scenarios are stories for design: rich stories of interaction
- Description of how the user engages the interactive system to solve a specific task/goal
- Formats:
 - Written summary, Use Case
 - Graphical sketches (→ Storyboard)
 - Flowcharts, Transition Diagram...

User scenarios

- User scenarios are **short stories** which UX designers create to show how users might act or achieve a goal when **interacting with a digital product**.
- **User scenarios help define:**
 1. The main **needs and motivations** of your users.
 2. The most **important points to consider** during the UX design process.



User scenarios

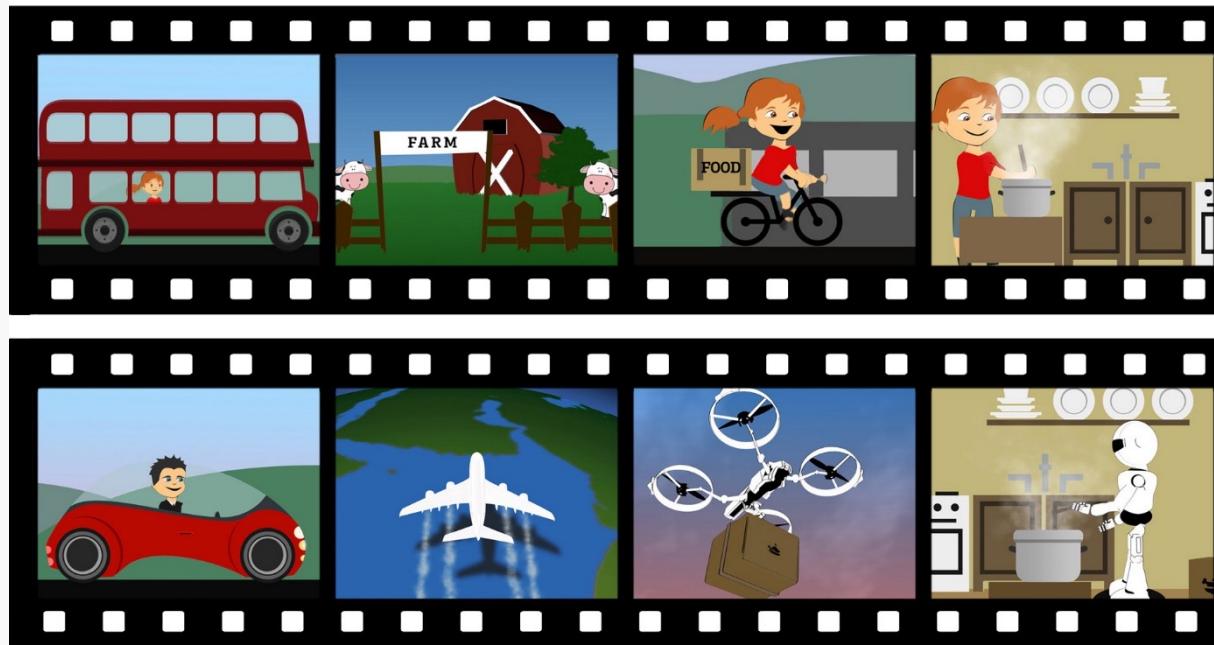
- "Cheryl is a Marketing Manager in Birmingham, and she has a business trip coming up in Northern California.
 - The trip is half pleasure, half work, and she wants to do some hiking in the mountain trails. She knows the roads can be pretty bumpy, so she wants to rent a car that is suitable for off-road driving.
 - Cheryl discovers [\[the mobile app\]](#) to find the perfect car. She is unfamiliar with these types of vehicles, so she compares some models and reads the reviews to get a better idea of her best option.
 - When she finds one, she likes in the budget allotted by her company, she books it."

In this short scenario, you can find many important points to consider when designing for Cheryl:

- **Context** (of use)
- **Goals**
- **Problems**
- **Motivations** (for using the app)
- **Personal traits**
- **Relevant habits**
- **Limitations/Frustrations**

Scenarios

- May be textual descriptions, animations, audio or video
- Example animation scenarios



Level of Details In Scenarios

- **Stories**

- From needfinding
- Used for understanding what people do and what they want

- **Conceptual (abstract) Scenarios**

- Used for generating ideas and specifying requirements
- Abstracts tasks from stories
- No reference to technology
- May lead to different concrete scenarios

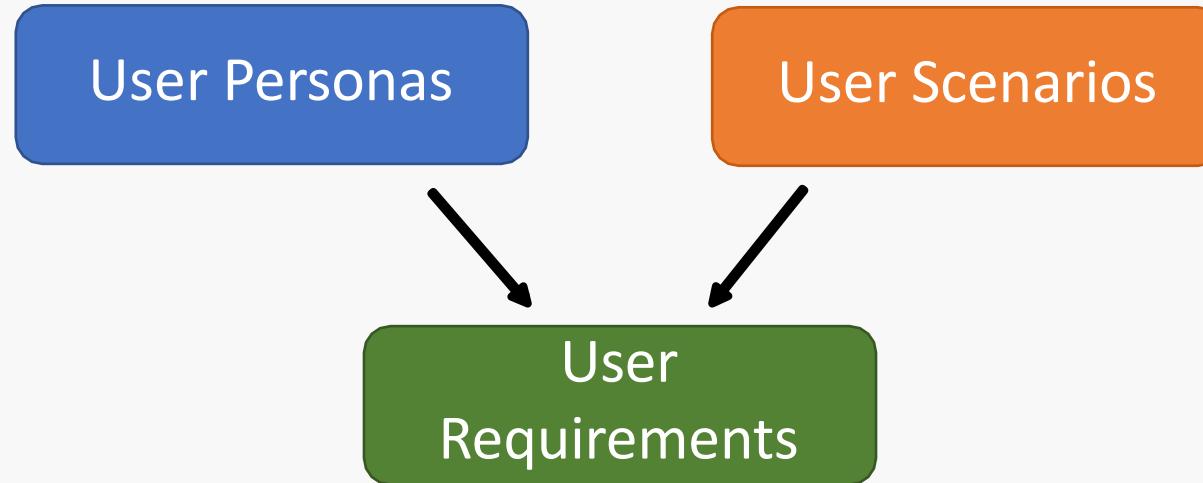
- **Concrete Scenarios**

- Used for envisioning ideas and evaluation
- One possible solution to a Conceptual Scenario (may try many alternatives)
- Shows how technologies are used in the user context
- Key design features are included

- **Use Cases**

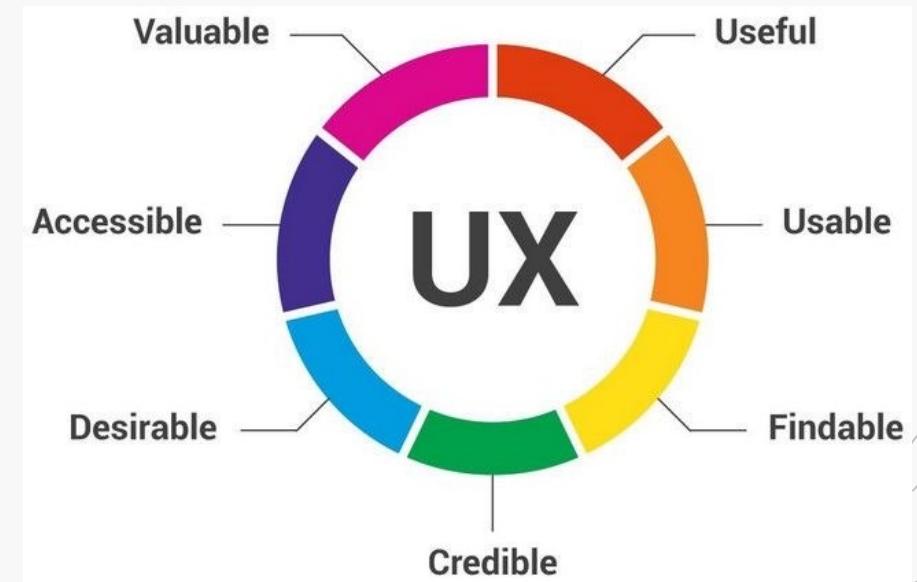
- Used for specification and implementation (→ software engineering)

User Requirements



User Requirements

- In brief, user requirements are what is **needed** for a digital product or service **to be successful**.
- **Value**: Is it useful?
- **Usability**: Is it easy to use?
- **Adoptability**: Is it easy to start using?
- **Desirability**: Is it fun or engaging?



User Requirements

- How do I define the user requirements for my mobile app?

MoSCoW method

Must have

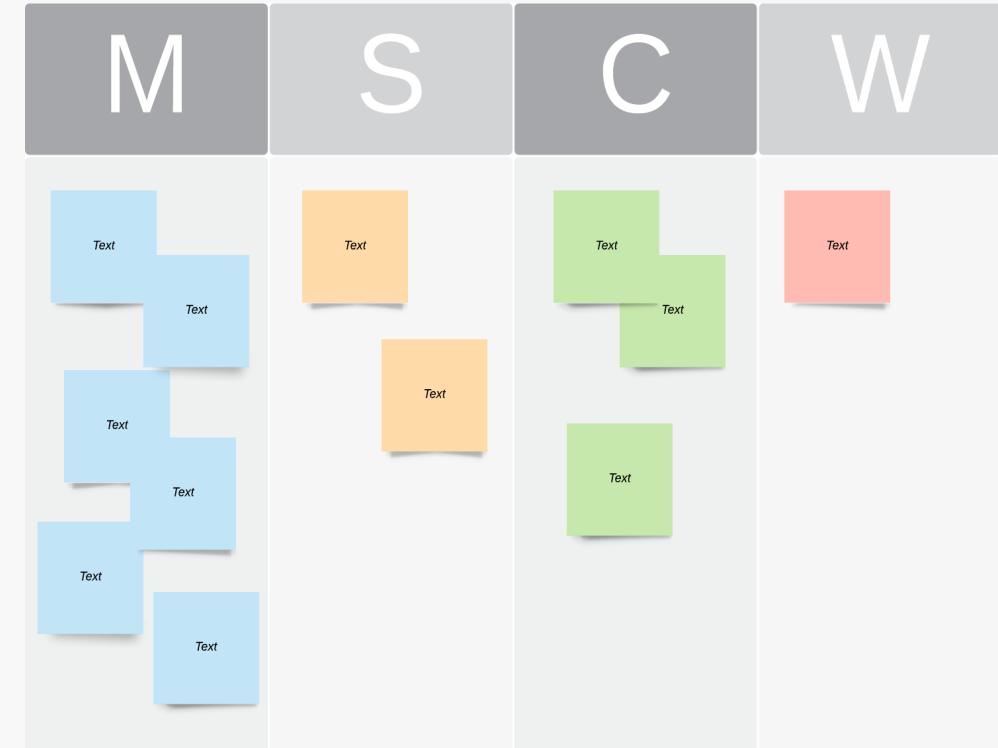
Should have

Could have

Won't have (this time)

MoSCoW method

- **MUST HAVE:** All requirements that are necessary for a successful user experience.
- **SHOULD HAVE:** Requirements that are important but not necessary.
- **COULD HAVE:** Requirements that are nice to have, but have a smaller impact on the user experience.
- **WON'T HAVE (this time):** All other requirements that have not been recognised as a priority this time.



This week's practical activity

Now you have the theme and idea of your project, this week you should:

1. Create 3 user personas:

- One persona should be a user with an impairment or disability, your app should be inclusive and accessible.

2. Define 3 user scenarios:

- Short stories that describe important points to consider for your mobile app, one scenario should focus on a premium feature.

3. Define user requirements:

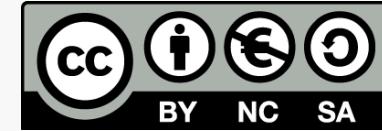
Using the MoSCoW method, create a list of features/functionalities to be included in your app. Remember to add the premium feature to the list.

4. Create initial HTA flowcharts:

Use HTA flowcharts to visualise the steps involved in accessing the main free and premium features in your mobile app.

References

- Alan Dix, Janet Finlay, Gregory Abowd, Russell Beale: Human Computer Interaction, 3rd Edition, Chapter 15 "Task Analysis"
- David Benyon: Designing Interactive Systems, Chapter 11 "Task Analysis"
- <http://www.usabilitybok.org/task-analysis>
- <https://www.usability.gov/how-to-and-tools/methods/task-analysis.html>



License

- These slides are distributed under a Creative Commons license "**Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)**"
- **You are free to:**
 - **Share** – copy and redistribute the material in any medium or format
 - **Adapt** – remix, transform, and build upon the material
 - The licensor cannot revoke these freedoms as long as you follow the license terms.
- **Under the following terms:**
 - **Attribution** – You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.
 - **NonCommercial** – You may not use the material for commercial purposes.
 - **ShareAlike** – If you remix, transform, or build upon the material, you must distribute your contributions under the same license as the original.
 - **No additional restrictions** – You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.
- <https://creativecommons.org/licenses/by-nc-sa/4.0/>



Vietnam National University of HCMC
International University
School of Computer Science and Engineering



THANK YOU

Dr Vi Chi Thanh - vcthanh@hcmiu.edu.vn

<https://vichithanh.github.io>



SCAN ME