

Welcome to PV 267 - UX workshop

Intro to UX and prototyping

Tereza Novotna
Interaction Designer

Overview

- ▶ The slides and recording of the class will be available to you in the GoogleClassroom.
- ▶ If you have any questions - feel free to reach out to me tnovotna@redhat.com
- ▶ Agenda of this workshop
 - Intro to ux design
 - Usability heuristics
 - Brainstorming and concepting
 - Wireframing
 - Prototyping
 - User testing
- ▶ Final assignment: an interactive prototype including
 - Detailed high fidelity prototype showcasing new feature or enhancement of messaging application
 - All packaged together, well-documented and presented on our last class

Overview

- ▶ A lot of topics to cover in one month
- ▶ We won't be able to cover each topic in depth
- ▶ I will be sharing real life work examples, struggles and wins
- ▶ I want to include your thoughts and discussion in this class.
- ▶ If possible -- please keep your cameras on, so we can leverage the best virtual experience

slido

Where are you located?

 Start presenting to display the poll results on this slide.

About me

- ▶ Tereza Novotna
- ▶ Interaction Designer at Red Hat for the past 3 years
- ▶ Graduated from Elon University in North Carolina with Master's in Interactive Media and Bachelor in International Business
- ▶ I like to run



slido

Got questions?

 Start presenting to display the poll results on this slide.

EXERCISE

Introduce yourself - create a stickie with your name and draw your favorite animal!

<https://app.mural.co/t/imperative3workshops1726/m/imperative3workshops1726/1607359508206/dca36c4d102f9df0efab991f7a11ad2d96e4d64b>

Roles in the UX



Red Hat User Experience Design

Interaction design

Visual design

User research

Front-end development (HTML/CSS, JavaScript, React)

Content/copy design

110 UXD members

Roles in the UX



Red Hat UI

Multiple web UIs - and multiple teams

Multiple frameworks

Custom component library - PatternFly

What are some programs/applications that designers use?

Try the free trials and get familiar with these!

1

Sketch / Figma

[Show example](#)

2

InVision / Marvel

3

Adobe (XD, Illustrator,
Photoshop)

4

Github / Gitlab

5

WebAIM (accessibility checker)

6

Google (Docs, Slides, Sheets, Forms, Jamboard)

Sketch is the main tool that designers on our team use!

What is user experience (UX)

The first requirement for an exemplary user experience is to meet the exact needs of the customer, without fuss or bother. Next comes simplicity and elegance that produce products that are a joy to own, a joy to use. True user experience goes far beyond giving customers what they say they want, or providing checklist features. In order to achieve high-quality user experience in a company's offerings there must be a seamless merging of the services of multiple disciplines, including engineering, marketing, graphical and industrial design, and interface design.

It's important to distinguish the total user experience from the user interface (UI), even though the UI is obviously an extremely important part of the design. As an example, consider a website with movie reviews. Even if the UI for finding a film is perfect, the UX will be poor for a user who wants information about a small independent release if the underlying database only contains movies from the major studios.

We should also distinguish UX and usability: According to the [definition of usability](#), it is a quality attribute of the UI, covering whether the system is easy to learn, efficient to use, pleasant, and so forth. Again, this is very important, and again total user experience is an even broader concept.

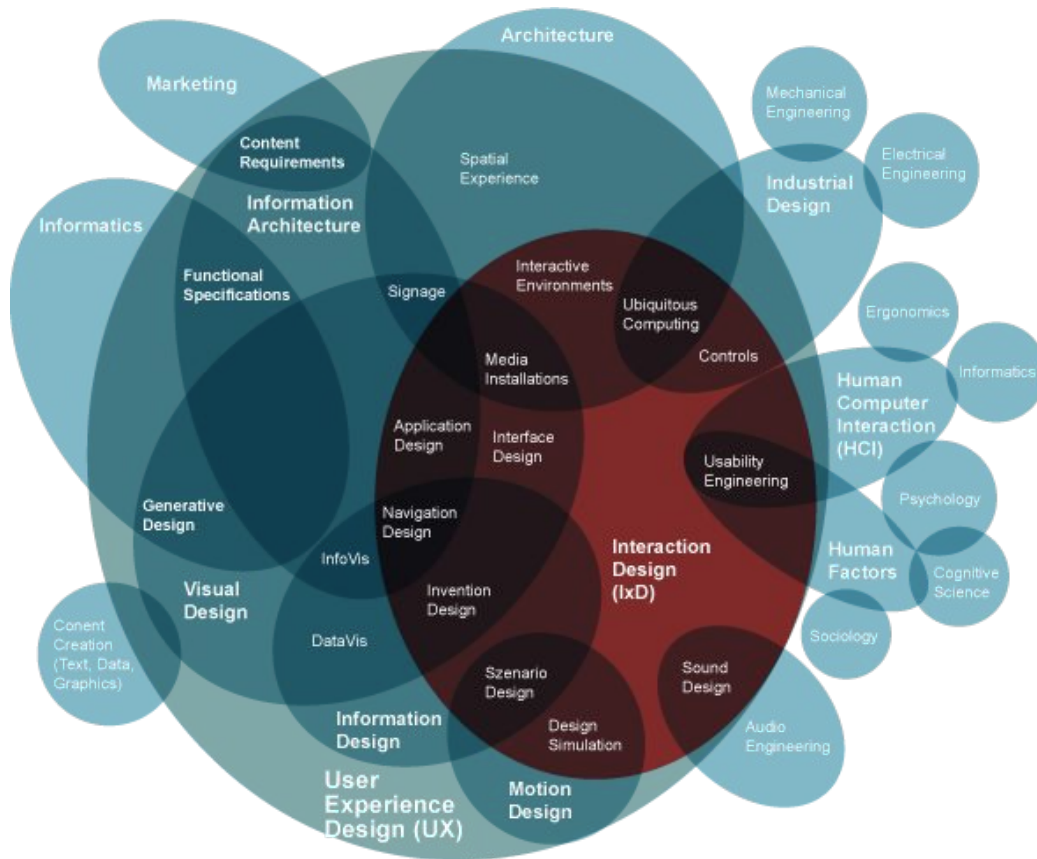
User experience discipline

“User experience is a discipline focused on designing the end-to-end experience of a certain product.” [or service, or system, etc.]

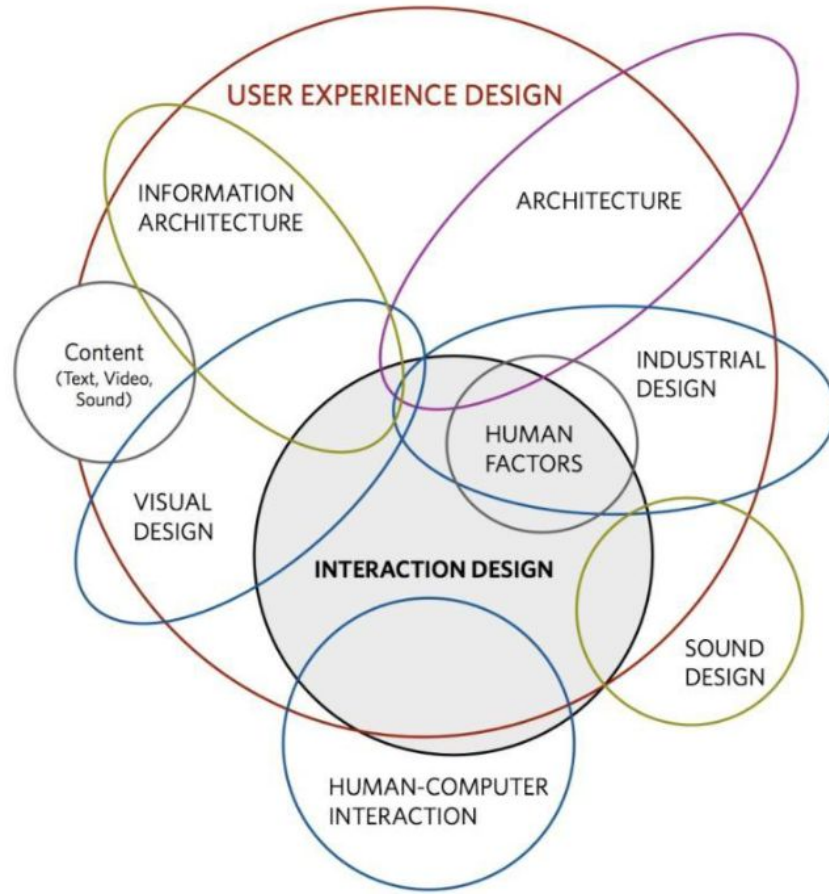
UX

User experience is the way a person feels about using a product, system or service.

The image shows a YouTube video player interface. The main video frame displays a close-up of an older man with a white beard and glasses, looking directly at the camera. Below the main video is a 'More videos' carousel with five thumbnails. The first thumbnail shows the same man in a dark suit against a blue background. The second shows him in a grey sweater in a home office. The third shows him in a dark suit on a stage with a blue screen behind him. The fourth shows him in a dark suit on a stage with a green screen. The fifth shows him in a dark suit on a stage with a white screen that has 'UX Rules' written on it. The video player controls at the bottom include a play button, a volume icon, a progress bar showing 0:17 / 1:49, a Creative Commons license icon, a settings gear icon, the YouTube logo, and a full-screen icon.



Copyright [envis precisely](#) (2009)
 based on «The Disciplines of User Experience» by Dan Saffer (2006)
www.kickerstudio.com/blog/2008/12/the-disciplines-of-user-experience



The background is a vibrant, abstract composition of swirling colors. On the left, there are wispy, ethereal clouds of light blue and cyan. On the right, there are more dynamic, splatter-like forms in shades of red, orange, and yellow. The colors blend and overlap, creating a sense of movement and depth. The overall effect is reminiscent of ink or paint being dropped into water and captured in a still frame.

**What is user
experience?**



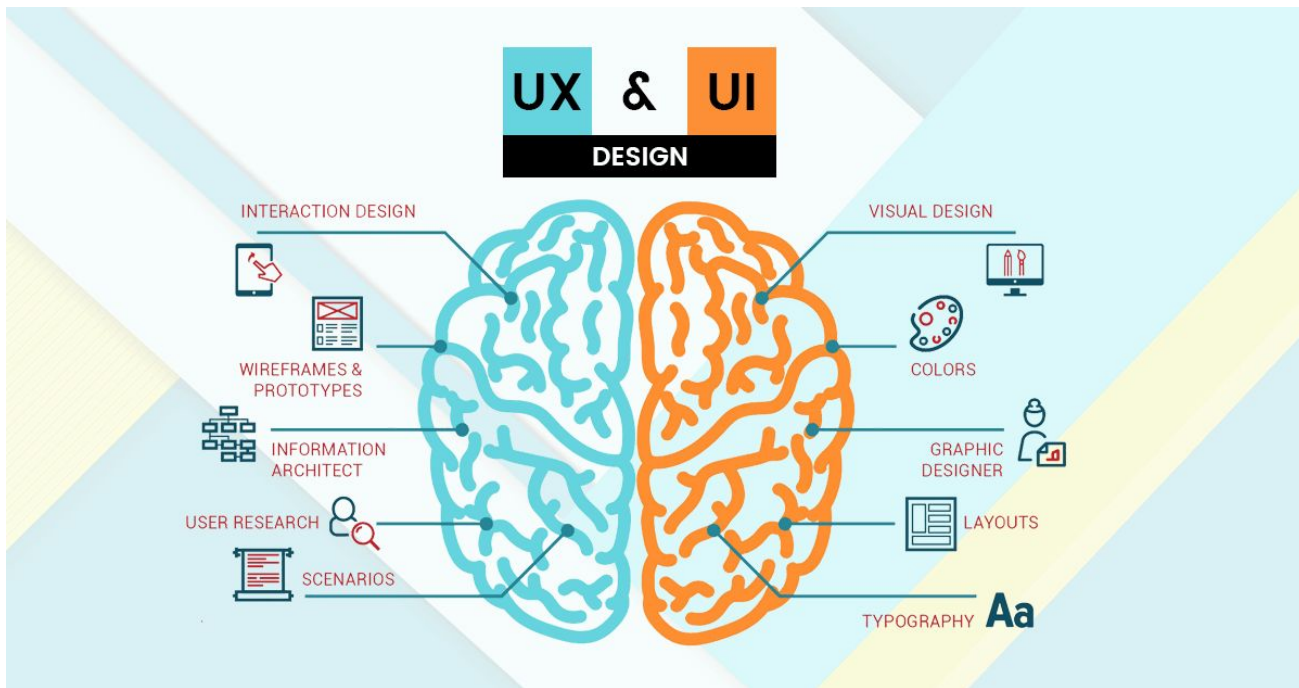
Almost everything you do in life - the purchases you make, the restaurants you eat in, the parks you explore, the cars you drive, the apps you download - is a user experience.

History

The term first appeared in acclaimed UX design expert Donald Norman's book, *The Design of Everyday Things*, which was first published in 1988. It marked a shift from the previous term "user centered system design" where instead of focusing on the system itself and the aesthetics of the interface, Norman concentrated on the needs of the user.

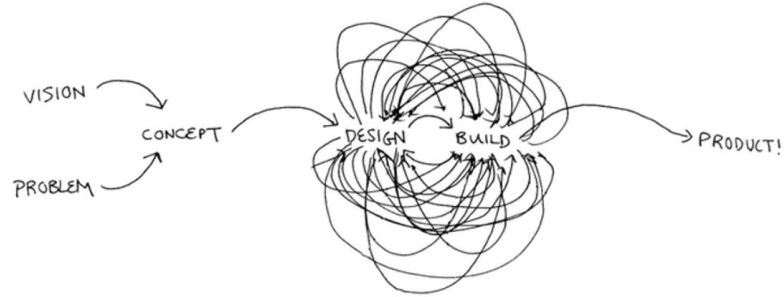
- ▶ Donald Norman together with Jakob Nielsen established the Nielsen Norman Group, active consultancy group

The difference between UX and UI

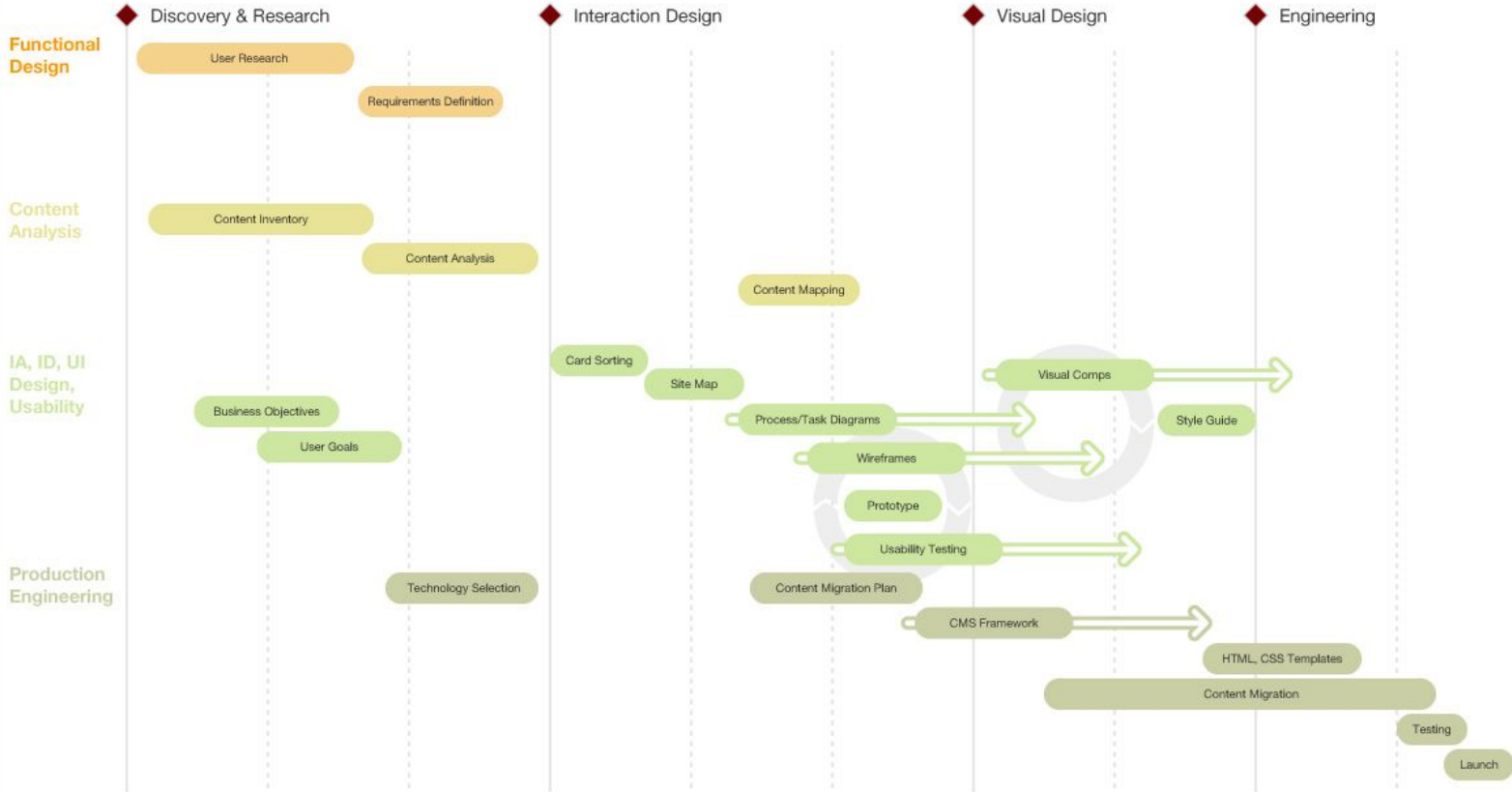


Source:
<https://www.uxui.com/tutorial/explaining-what-is-the-difference-between-ux-and-ui-design/>

Design process



Design Process Diagram



UX

An iceberg floating in dark blue water. The tip of the iceberg, which is above the water line, is white and contains the large letters 'UX'. The much larger part of the iceberg is submerged below the water line and is a dark blue color. To the left of the iceberg, there are four text blocks, each corresponding to a layer of the iceberg's structure.

Surface
- Visual Design

Skeleton
- Interface Design
- Navigation Design
- Information Design

Structure
- Interaction Design
- Information Architecture

Scope
- Functional Specs
- Content Requirements

Strategy
- User Needs
- Site Objectives

START WITH MEANING

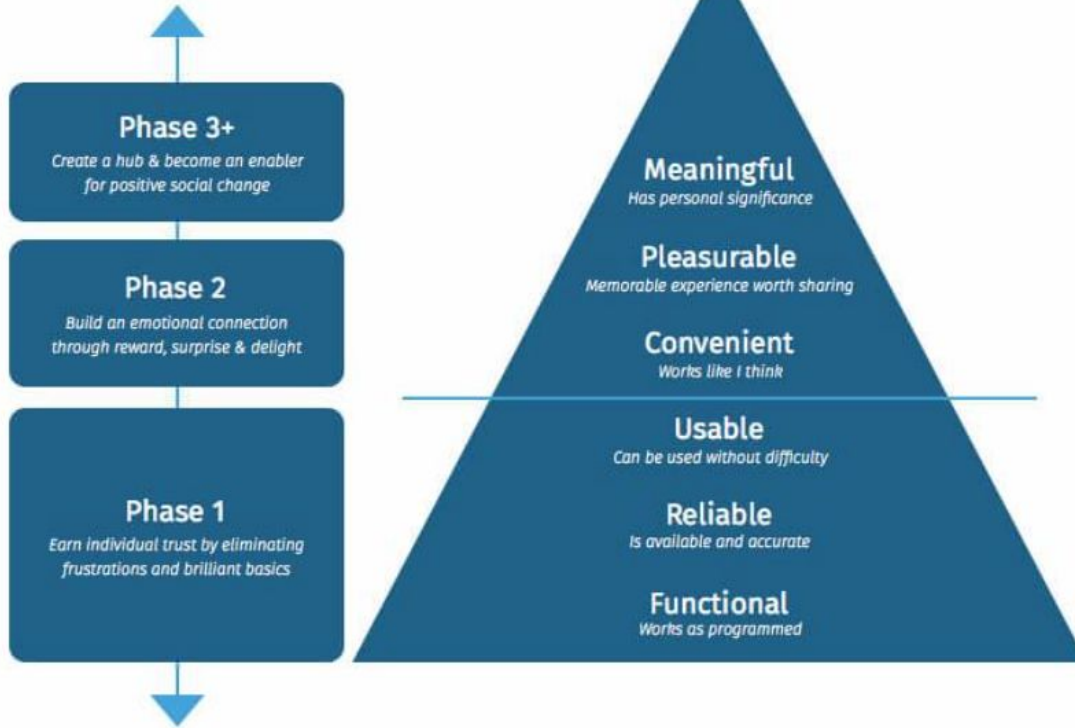


Illustration of Stephen Anderson's UX Hierarchy of Needs from [Simon Pan](#) site.

slido

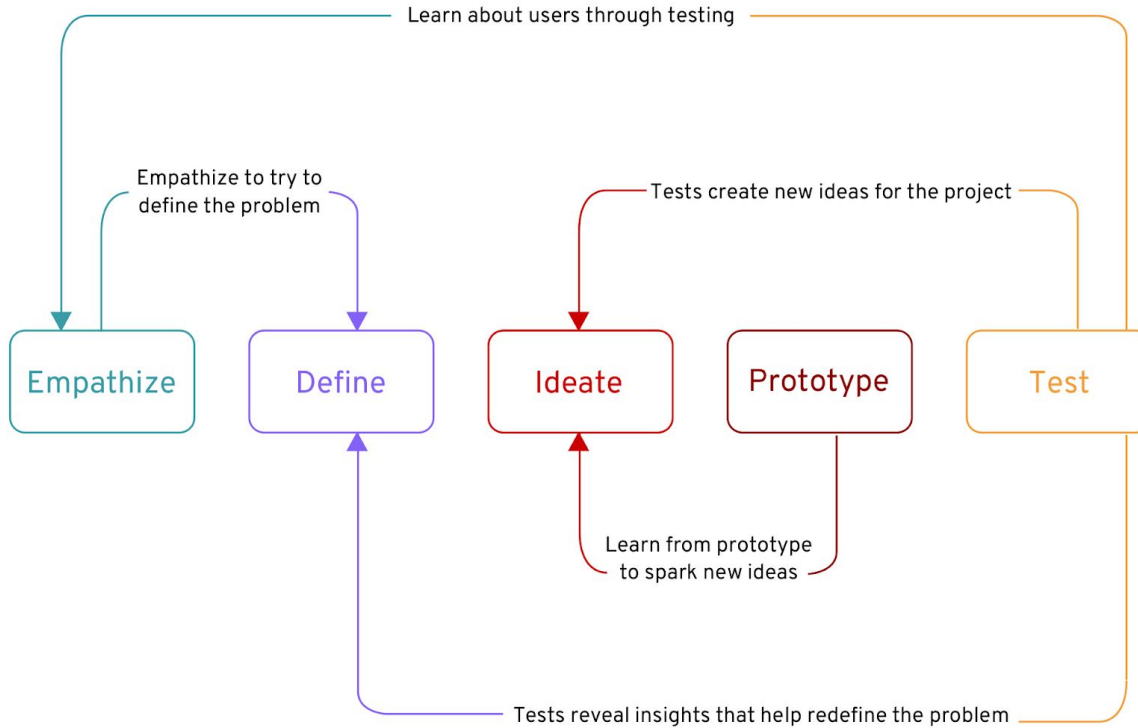
Co je to UX?

 Start presenting to display the poll results on this slide.

Design thinking

- an approach used for practical and creative problem-solving
- Design Thinking can also be applied to any field; it doesn't necessarily have to be design-specific
- is extremely user-centric
- understand people's needs and come up with effective solutions to meet those needs

DESIGN THINKING : A NON-LINEAR PROCESS



Use case

- ▶ Usability.gov
 - A written description of how users will perform tasks on your website. It outlines, from a user's point of view, a system's behavior as it responds to a request. Each use case is represented as a sequence of simple steps, beginning with a user's goal and ending when that goal is fulfilled.
 - Project teams can then negotiate which functions become requirements and are built

✓ What uses cases include

- Who is using the website
- What the user want to do
- The user's goal
- The steps the user takes to accomplish a particular task
- How the website should respond to an action

✗ What use cases do NOT include

- Implementation-specific language
- Details about the user interfaces or screens.

How to write a use case

1. Identify who is going to be using the website.
2. Pick one of those users.
3. Define what that user wants to do on the site. Each thing the use does on the site becomes a use case.
4. For each use case, decide on the normal course of events when that user is using the site.
5. Describe the basic course in the description for the use case. Describe it in terms of what the user does and what the system does in response that the user should be aware of.
6. When the basic course is described, consider alternate courses of events and add those to "extend" the use case.
7. Look for commonalities among the use cases. Extract these and note them as common course use cases.
8. Repeat the steps 2 through 7 for all other users.

Example Use Cases

Below, are examples of three use cases with increasing levels of complexity. For our purposes we have defined them as Simple, Middleweight and Heavyweight use case for doing the laundry. In each of these types of uses cases you will see that:

- A housekeeper does laundry on a Wednesday
- She washes each load.
- She dries each load.
- She folds certain items.
- She irons some items.
- She throws away certain items
- Example of use cases and alternatives.

EXERCISE

You'll be divided into groups

- (1) think of as many as possible use cases a messaging app can have (Messenger, imessage, whatsapp etc).
- (2) Select an application, and include screenshots how the app performs specific use cases
- (3) Make sure you notice what some applications cannot do.


Add it all into Mural.

Heuristic principles

10 general principles for interaction design


1 Visibility of System Status

Designs should *keep users informed* about what is going on, through appropriate, timely feedback.

 Interactive mall maps have to show people where they currently are, to help them understand where to go next.


2 Match between System and the Real World

The design should speak the users' language. Use words, phrases, and concepts *familiar to the user*, rather than internal jargon.

 Users can quickly understand which stovetop control maps to each heating element.


5 Error Prevention

Good error messages are important, but the best designs *carefully prevent problems from occurring in the first place*.

 Guard rails on curvy mountain roads prevent drivers from falling off cliffs.

8 Aesthetic and Minimalist Design

Interfaces should not contain information which is irrelevant. Every extra unit of information in an interface *competes* with the relevant units of information.


 A minimalist three-legged stool is still a place to sit.

Nielsen Norman Group

Jakob's Ten Usability Heuristics


3 User Control and Freedom

Users often perform actions by mistake. They *need a clearly marked "emergency exit"* to leave the unwanted action.

 Just like physical spaces, digital spaces need quick "emergency" exits too.


6 Recognition Rather Than Recall

Minimize the user's memory load by making elements, actions, and options visible. Avoid making users remember information.

 People are likely to correctly answer "Is Lisbon the capital of Portugal?".


9 Recognize, Diagnose, and Recover from Errors

Error messages should be expressed in plain language (no error codes), precisely indicate the problem, and constructively suggest a solution.

 Wrong-way signs on the road remind drivers that they are heading in the wrong direction.


4 Consistency and Standards

Users should not have to wonder whether different words, situations, or actions mean the same thing. *Follow platform conventions*.

 Check-in counters are usually located at the front of hotels, which meets expectations.


7 Flexibility and Efficiency of Use

Shortcuts – hidden by novice users – *may speed up the interaction for the expert user*.

 Regular routes are listed on maps, but locals with more knowledge of the area can take shortcuts.

10 Help and Documentation

It's best if the design *doesn't need* any additional explanation. However, it may be necessary to provide documentation to help users complete their tasks.

 Information kiosks at airports are easily recognizable and solve customers' problems in context and immediately.

Watch Heuristic Evaluation of User Interfaces

https://www.youtube.com/watch?v=6Bw0n6Jvwk&feature=emb_title&ab_channel=NNgroup

Presented by Jakob Nielsen

Source:

<https://www.mockplus.com/blog/post/what-is-a-wireframe-what-is-a-prototype>

1 Visibility of System Status

Definition The design should **always keep users informed** about what is going on, through appropriate feedback within a reasonable amount of time.



Knowing what the current system status is can help users learn the outcome of their prior interactions and determine next steps.

Predictable interactions create trust in the product as well as the brand.

Tip: Communicate *clearly* to users what the system's state is — no action with consequences to users should be taken without informing them.

Tip: Present feedback to the user as *quickly* as possible.

Tip: Build *trust* through open and continuous communication.

1 "You Are Here" maps

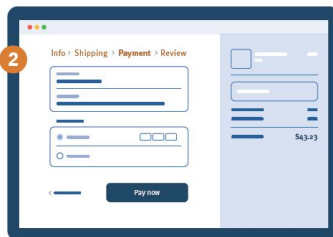
Interactive mall maps have to show people where they currently are, to help them understand where to go next.

2 Checkout flow

Multistep processes show users which steps they've completed, they're currently working on, and what comes next.

3 Phone tap

Touchscreen UIs need to reassure users that their taps have an effect — often through visual change or haptic feedback.





2 Match between System and the Real World

Definition The design should speak the users' language. Use words, phrases, and concepts **familiar to the user**, rather than internal jargon. Follow real-world conventions, making information appear in a natural and logical order.



The language you should use depends very much on your specific users.

- ✎ **Tip:** Ensure users can *understand* meaning without having to go look up a word's definition.
 - ✎ **Tip:** Never *assume* your understanding of words or concepts will match those of your users.
 - ✎ **Tip:** User research will help you uncover your users' familiar terminology, as well as their mental models around important concepts.
- 1 **Stovetop controls**
 When stovetop controls match the layout of heating elements, users can quickly understand which control maps to each heating element.
 - 2 **"Car" vs. "automobile"**
 If users think about this object as a "car," use that as the label instead.
 - 3 **Shopping cart icon**
 A shopping cart icon is easily recognizable because that feature serves the same purpose as its real-life counterpart.



3 User Control and Freedom

Definition Users often perform actions by mistake. They **need a clearly marked "emergency exit"** to leave the unwanted action without having to go through an extended process.

When it's easy for people to back out of a process or undo an action, it fosters a sense of freedom and confidence.

Exits allow users to remain in control of the system and avoid getting stuck and feeling frustrated.

- Tip: Support *Undo* and *Redo*.
- Tip: Show a clear way to *exit* the current interaction, like a "Cancel" button.
- Tip: Make sure the exit is clearly *labeled* and discoverable.



- Exit sign**
Digital spaces need quick "emergency" exits, just like physical spaces do.
- Undo and redo**
These functions give users freedom because they don't have worry about their actions – everything is easily reversible.
- Cancel button**
Users shouldn't have to commit to a process once it's started – they should be able to easily cancel and abandon.

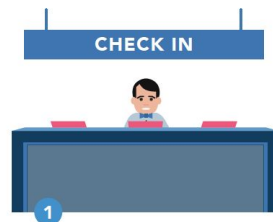


4 Consistency and Standards

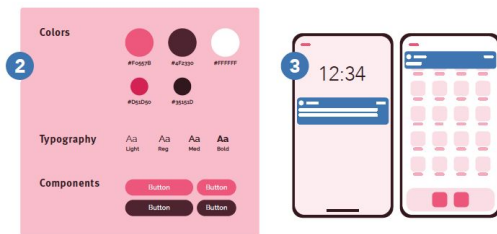
Definition Users should not have to wonder whether different words, situations, or actions mean the same thing. **Follow platform and industry conventions.**

Jakob's Law states that people spend most of their time on products other than yours. Failing to maintain consistency may increase the users' cognitive load by forcing them to learn something new.

- Tip: Improve learnability by maintaining *both* types of consistency: internal and external.
- Tip: Maintain consistency within a single product or a family of products (*internal* consistency).
- Tip: Follow established industry conventions (*external* consistency).



- 1 Check-in counter**
Check-in counters are usually located at the front of hotels. This consistency meets customers' expectations.
- 2 Design system**
Using elements from the same design system across the product lines lowers the learning curve of users.
- 3 Notifications**
A standardized notification design provides a similar but distinguishable look and feel for different app pop-ups.



5 Error Prevention

Definition Good error messages are important, but the best designs **carefully prevent problems** from occurring in the first place. Either eliminate error-prone conditions, or check for them and present users with a confirmation option before they commit to the action.



There are two types of errors: slips and mistakes.

Slips are unconscious errors caused by inattention.

Mistakes are conscious errors based on a mismatch between the user's mental model and the design.

Tip: *Prioritize* your effort: Prevent high-cost errors first, then little frustrations.

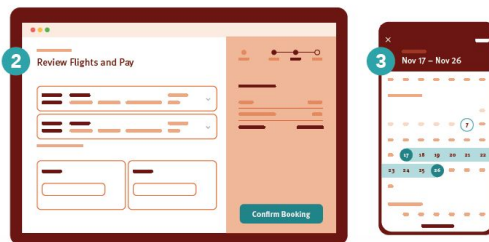
Tip: Avoid slips by providing helpful *constraints* and good defaults.

Tip: Prevent *mistakes* by removing memory burdens, supporting undo, and warning your users.

1 Guard rails
Guard rails on curvy mountain roads prevent drivers from falling off of cliffs.

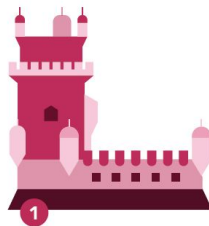
2 Airline confirmation
The confirmation page before checking out on airline websites gives users another chance to review the flight details

3 Date selection on calendar
Offer good defaults and set boundaries when people book services by dates. Grey out unavailable options.



6 Recognition Rather Than Recall

Definition Minimize the user's memory load by making elements, actions, and options visible. The user should **not have to remember information** from one part of the interface to another. Information required to use the design should be visible or easily retrievable when needed.



Humans have limited short-term memories. Interfaces that promote recognition reduce the amount of cognitive effort required from users.

- Tip: Let people *recognize* information in the interface, rather than having to remember ("recall") it.
- Tip: Offer help *in-context*, instead of giving users a long tutorial to memorize.
- Tip: *Reduce* the information that users have to remember.

- Lisbon**
People are more likely to correctly answer the question "Is Lisbon the capital of Portugal?" rather than "What's the capital of Portugal?"
- Comparison table**
Comparison tables list key differences so that users don't need to remember them to make comparisons.
- Search**
Search queries are presented together with the results as a reference.



7 Flexibility *and* Efficiency of Use

Definition Shortcuts — hidden from novice users — may speed up the interaction for the expert user such that the design can **cater to both inexperienced and experienced users**. Allow users to tailor frequent actions.

Flexible processes can be carried out in different ways, so that people can pick whichever method works for them.

- Tip: Provide *accelerators* like keyboard shortcuts and touch gestures.
- Tip: Provide *personalization* by tailoring content and functionality for individual users.
- Tip: Allow for *customization*, so users can make selections about how they want the product to work.



1 Shortcuts

Regular routes are listed on maps, but locals with more knowledge of the area can take shortcuts.

2 Keyboard shortcut

Keyboard shortcuts for complex products can help expert users finish their tasks more efficiently.

3 Tap to like

Social apps allow two ways to like posts. Experienced users can tap to like because it speeds up their browsing.



8 Aesthetic and Minimalist Design

Definition Interfaces should not contain information which is irrelevant or rarely needed. Every extra unit of information in an interface **competes** with the relevant units of information and diminishes their relative visibility.



This doesn't mean you have to use a flat design — it's about making sure you're keeping the content and visual design focused on the essentials. Ensure that the visual elements of the UI support the user's primary goals.

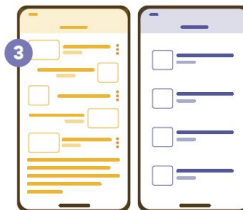
- Tip: Keep the content and visual design of UI focus on the *essentials*.
- Tip: Don't let unnecessary elements distract users from the information they really need.
- Tip: *Prioritize* the content and features to support primary goals.

- 1 Ornate vs. simple teapot**
Excessive decorative elements can interfere with usability.
- 2 Communicate, don't decorate**
Over-decoration can cause distraction and make it harder for people to get the core information they need.
- 3 Messy vs organized UI**
Messy UI increases the interaction cost for users to find their desired content; Organized UI lowers the cost.

2

**COMMUNICATE,
DON'T DECORATE**

One of our favorite slogans



9 Help Users Recognize, Diagnose, and Recover from Errors

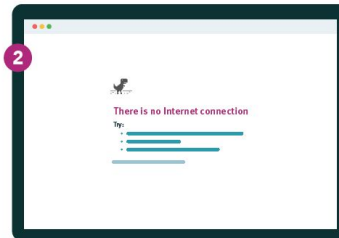
Definition Error messages should be expressed in **plain language** (no error codes), precisely indicate the problem, and constructively suggest a solution.

Error messages should be presented with visual treatments that will help users notice and recognize them.

- Tip:** Use *traditional* error message visuals, like bold, red text.
- Tip:** Tell users what went wrong in language they will *understand* — avoid technical jargon.
- Tip:** Offer users a *solution*, like a shortcut that can solve the error immediately.



- 1 Wrong way sign**
Wrong-way signs on the road remind drivers that they are heading in the wrong direction and ask them to stop.
- 2 Internet connection error**
Good internet connection error pages show what happened and constructively instruct users on how to fix the problem.
- 3 No search results**
Provide useful help when people encounter search-result pages returning zero results, such as popular topics.



10 Help and Documentation

Definition It's best if the design **doesn't need** any additional explanation. However, it may be necessary to provide documentation to help users understand how to complete their tasks.



Help and documentation content should be easy to search and focused on the user's task. Keep it concise, and list concrete steps that need to be carried out.

- Tip: Ensure that the help documentation is easy to search.
 - Tip: Whenever possible, present the documentation in-context right at the moment that the user requires it.
 - Tip: List *concrete* steps to be carried out.
- 1 Airport information center**
Information kiosks at airports are easily recognizable and solve customers' problems in context and immediately.
 - 2 Frequently asked questions**
Good frequently-asked-question pages anticipate and provide the helpful information that users might need.
 - 3 Information icon**
Information icons reveal tooltips to explain jargon when users touch or hover over them, which provides contextual help.

