Nielsen Norman Group

Jakob's Ten Usability Heuristics

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.

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.

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.

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.

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?".

7 Flexibility *and* Efficiency of Use

Shortcuts — hidden from 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.

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.

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.

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.

NN/g

www.nngroup.com/articles/ten-usability-heuristics/

Jakob's Heuristic

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.

Predictible 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.

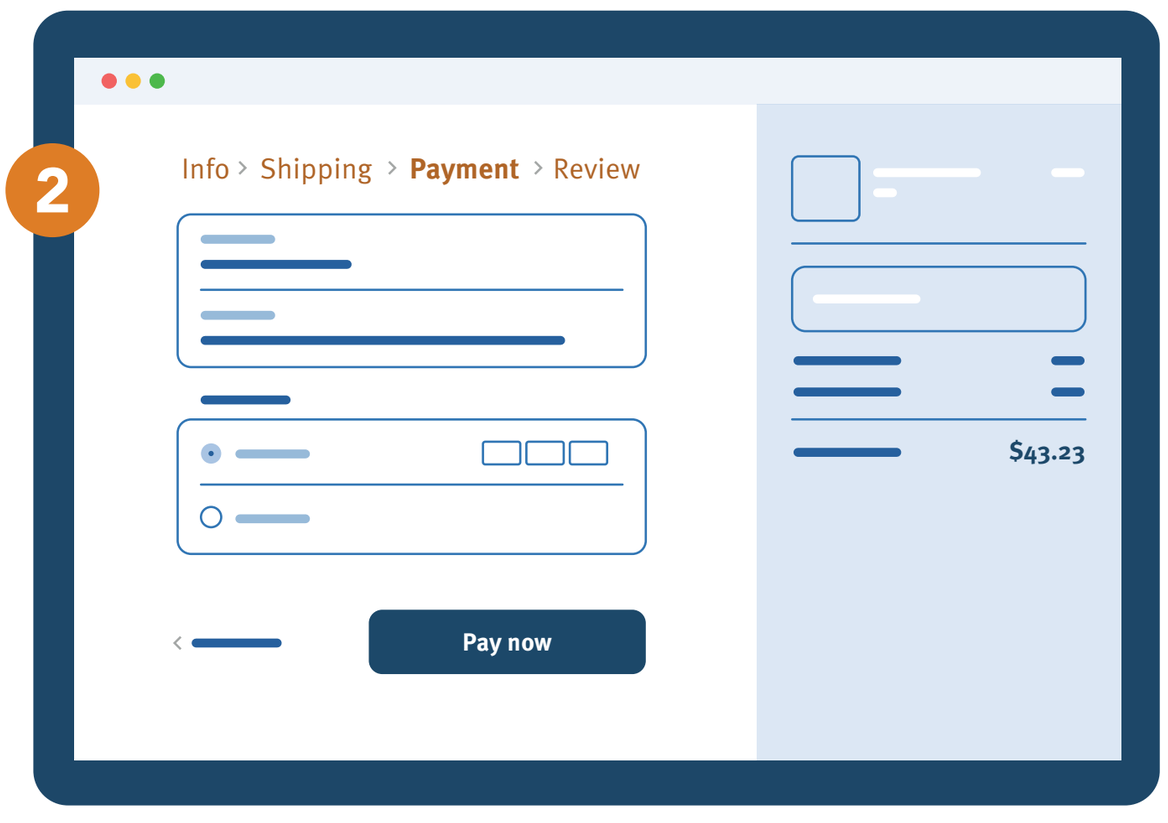
Tip: Present feedback to the user as *quickly* as possible.

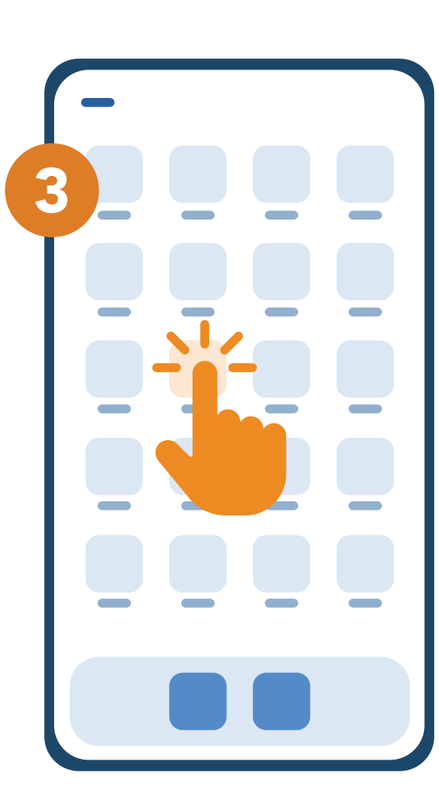
Tip: Build *trust* through open and continuous communication.

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.

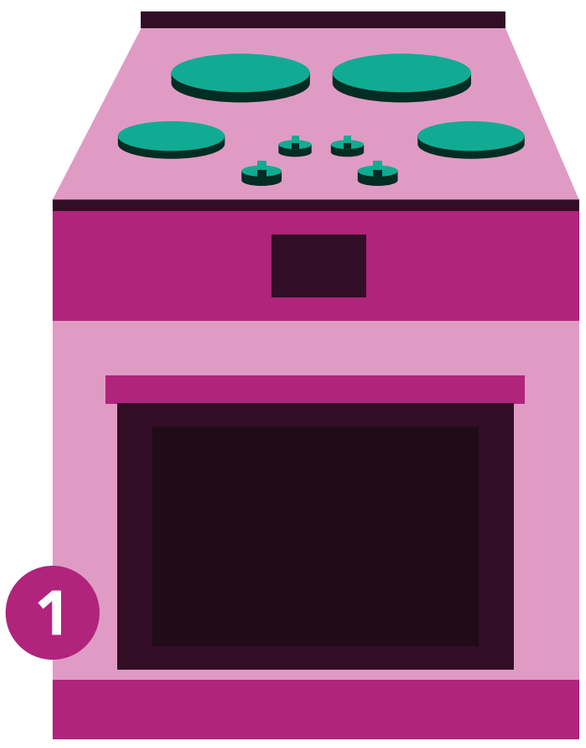




Jakob's Heuristic

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.

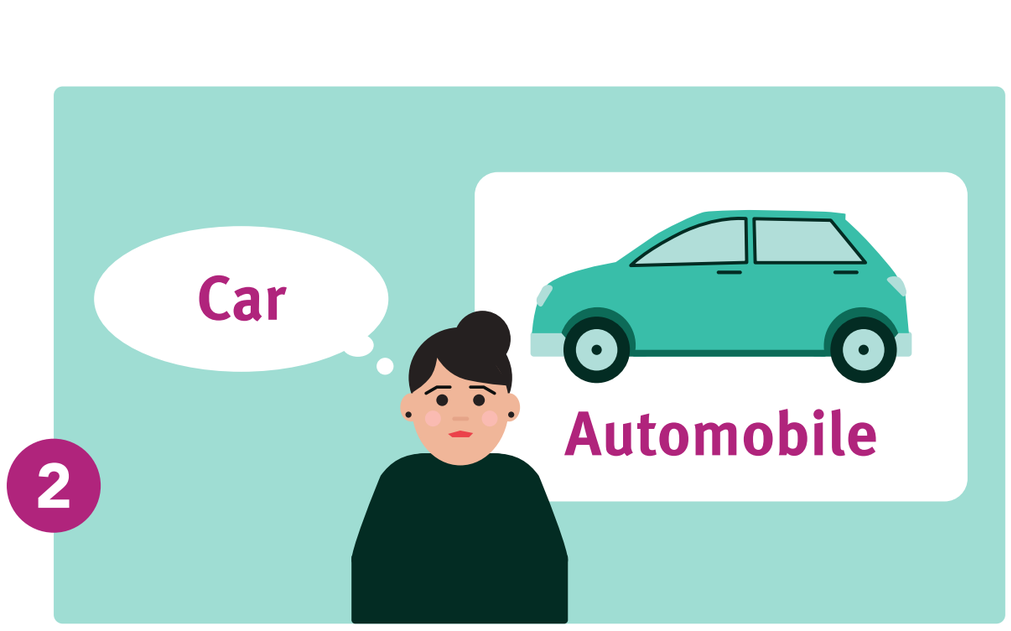
Tip: Never *assume* your understanding of words or concepts will match those of your users.

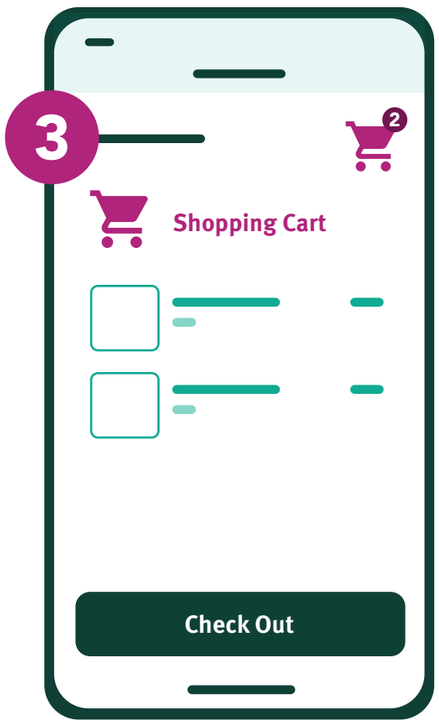
Tip: User research will help you uncover your users' familiar terminology, as well as their mental models around important concepts.

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.





Jakob's Heuristic

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*.

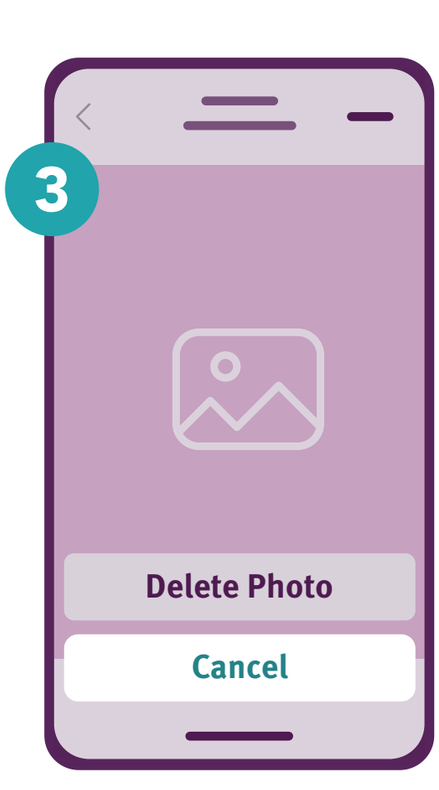
Tip: Show a clear way to *exit* the current interaction, like a "Cancel" button.

Tip: Make sure the exit is clearly *labeled* and discoverable.

1 ***Exit sign***
Digital spaces need quick "emergency" exits, just like physical spaces do.

2 ***Undo and redo***
These functions give users freedom because they don't have worry about their actions — everything is easily reversible.

3 ***Cancel button***
Users shouldn't have to commit to a process once it's started — they should be able to easily cancel and abandon.



NN/g

www.nngroup.com/articles/user-control-and-freedom

Jakob's Heuristic

4 Consistency *and* Standards

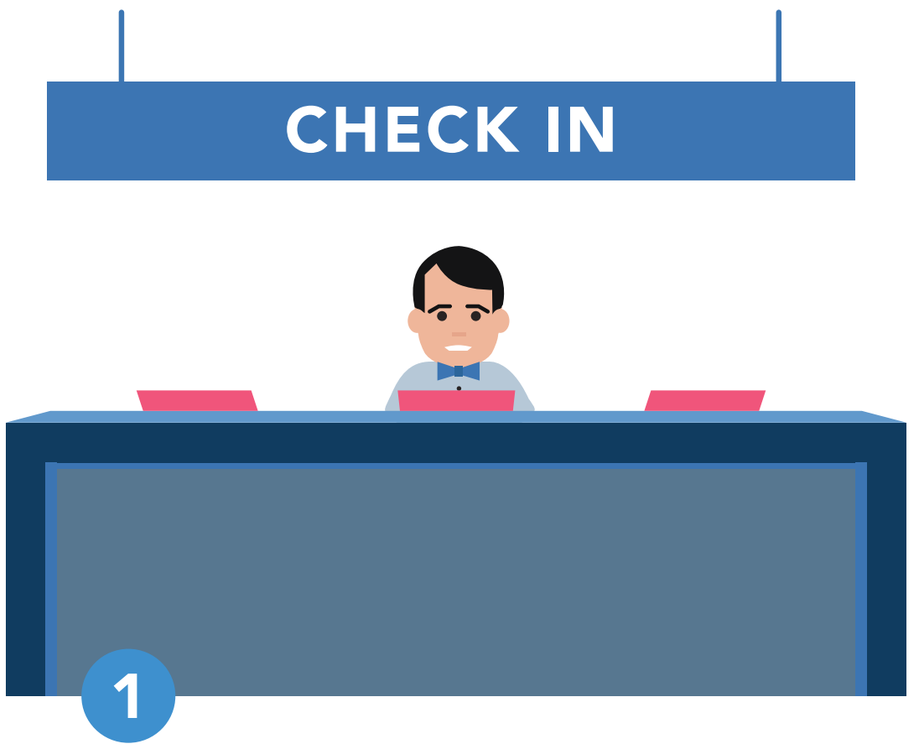
Definition Users should not have to wonder whether different words, situations, or actions mean the same thing. **Follow platform and industry conventions.**

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).

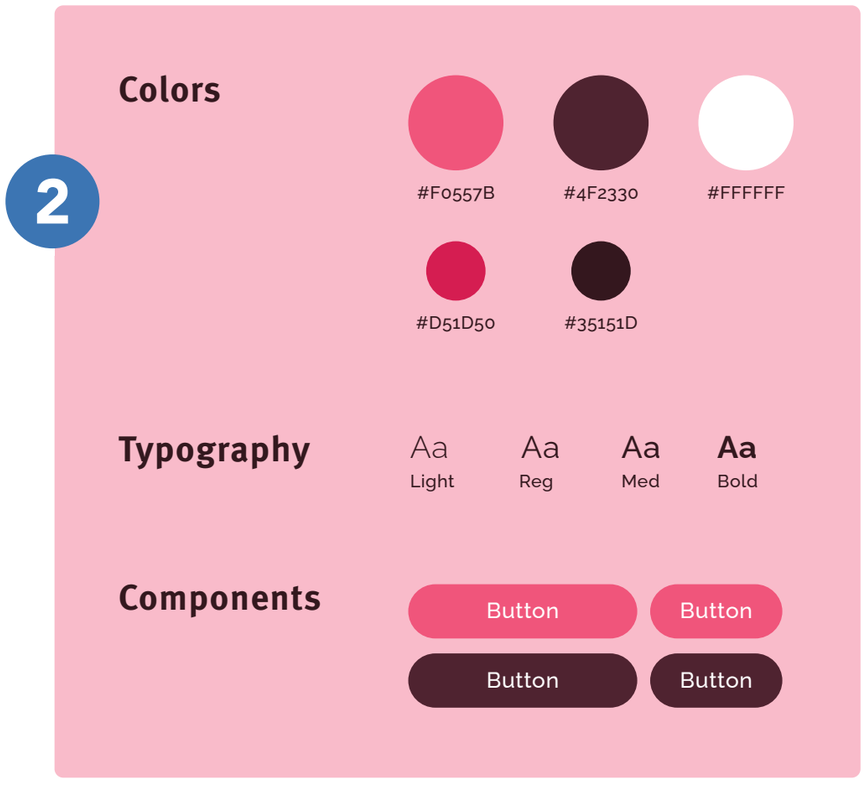
Tip: Follow established industry conventions (*external* consistency).

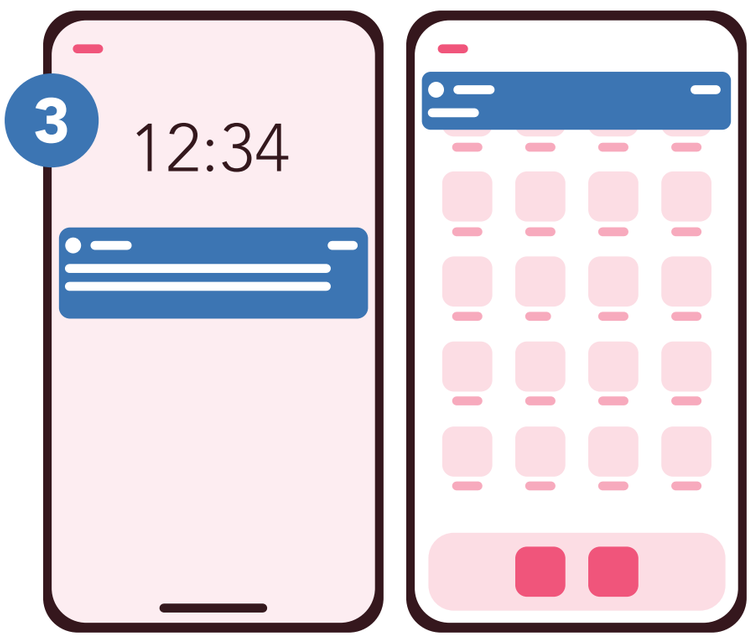


1 ***Check-in counter***
Check-in counters are usually located at the front of hotels. This consistency meets customers' expectations.

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.

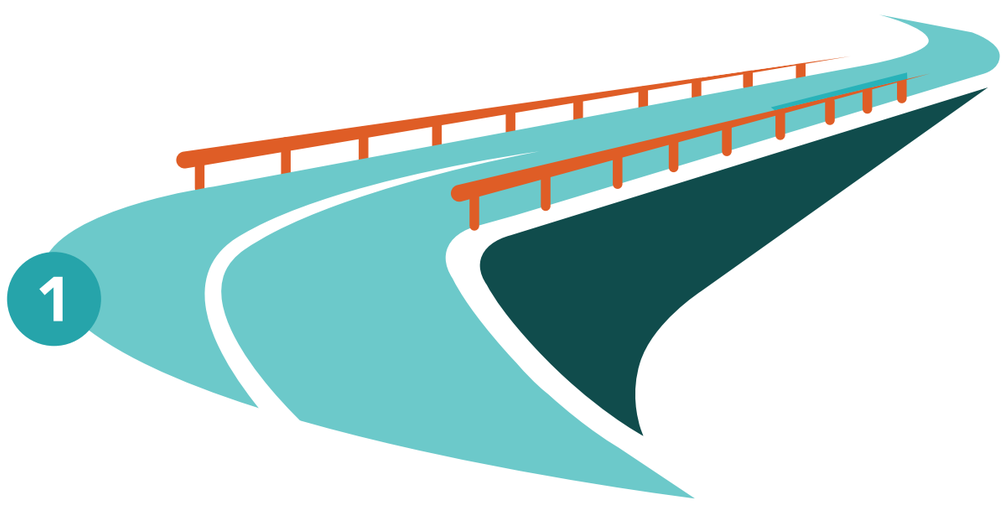




Jakob's Heuristic

5 Error Prevention

Definition Good error messages are important, but the best designs **carefully prevent problems** from occuring 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.

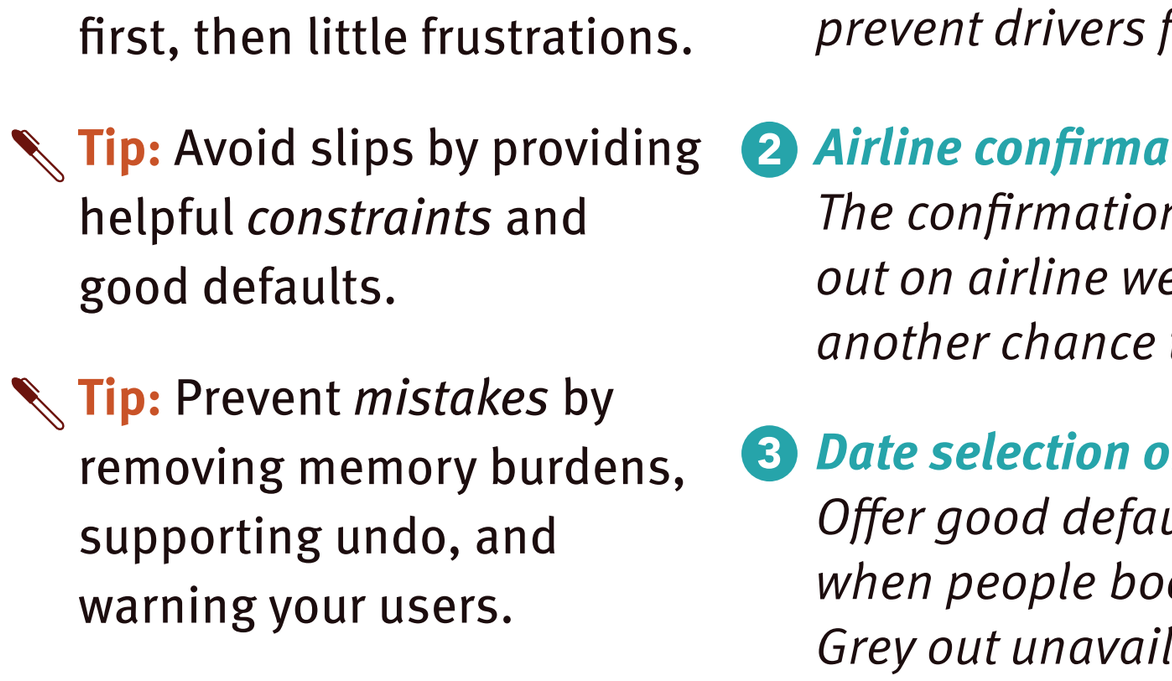
Tip: Avoid slips by providing helpful *constraints* and good defaults.

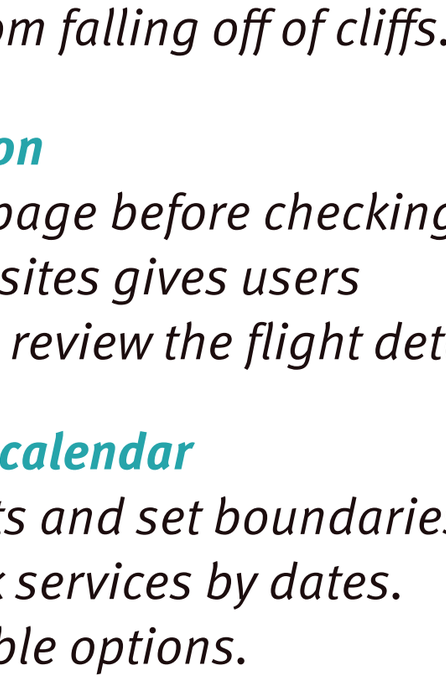
Tip: Prevent *mistakes* by removing memory burdens, supporting undo, and warning your users.

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.

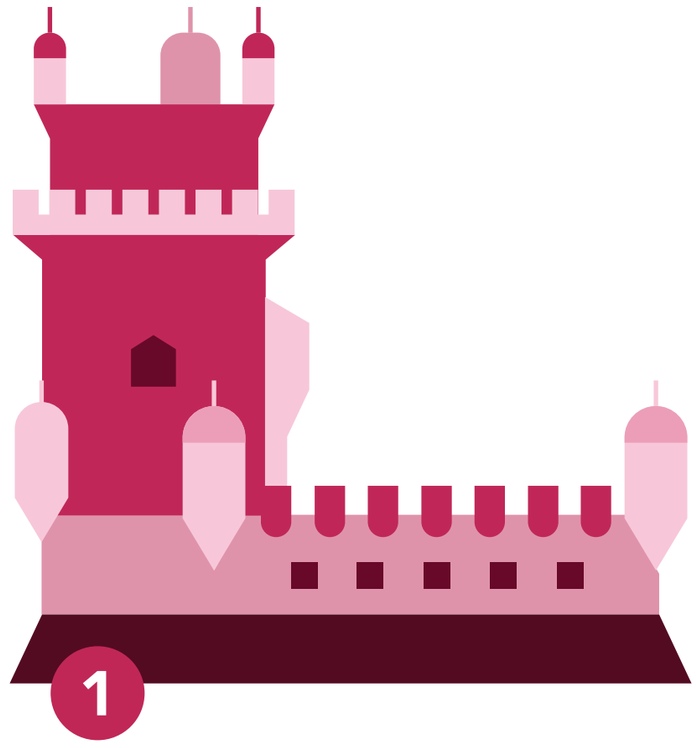




Jakob's Heuristic

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.

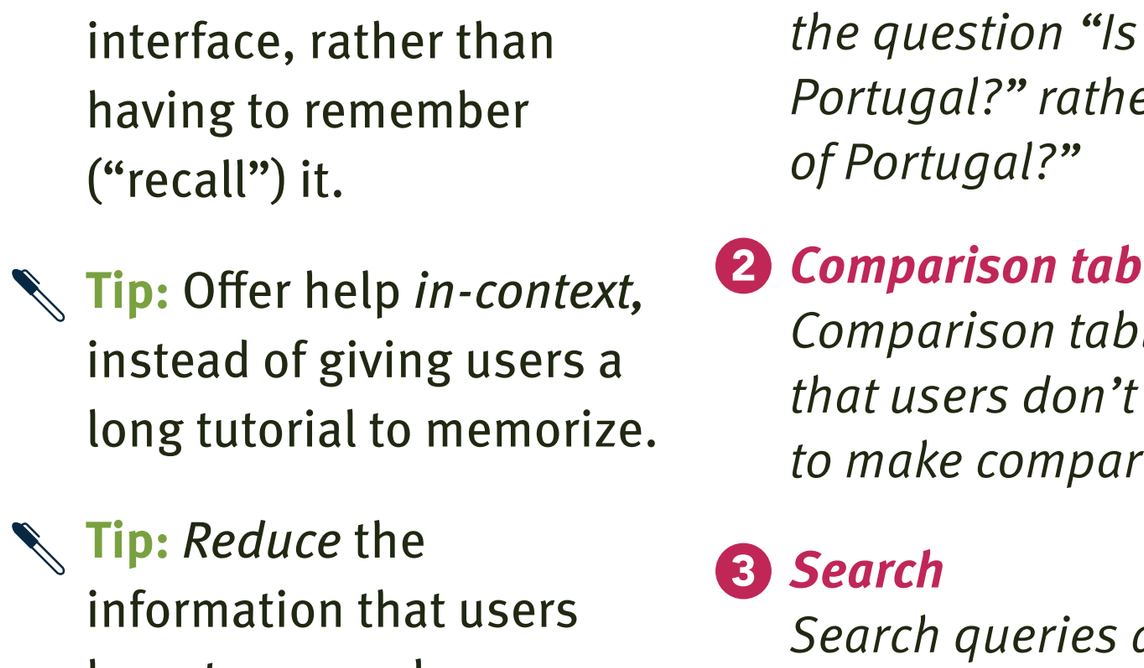
Tip: Offer help *in-context*, instead of giving users a long tutorial to memorize.

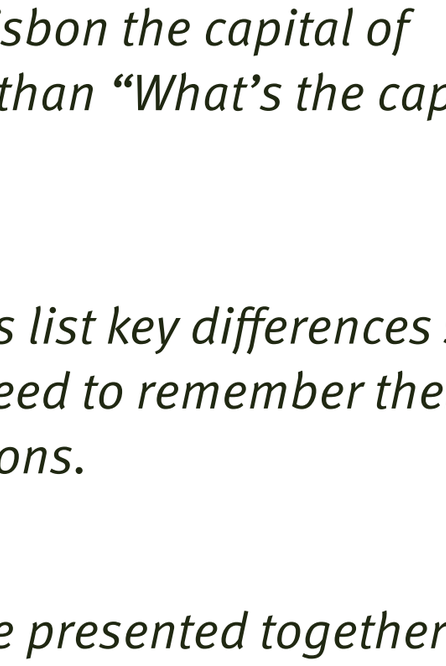
Tip: *Reduce* the information that users have to remember.

1 ***Lisbon***
People are more likely to correctly answer the question "Is Lisbon the capital of Portugal?" rather than "What's the capital of Portugal?"

2 ***Comparison table***
Comparison tables list key differences so that users don't need to remember them to make comparisons.

3 ***Search***
Search queries are presented together with the results as a reference.





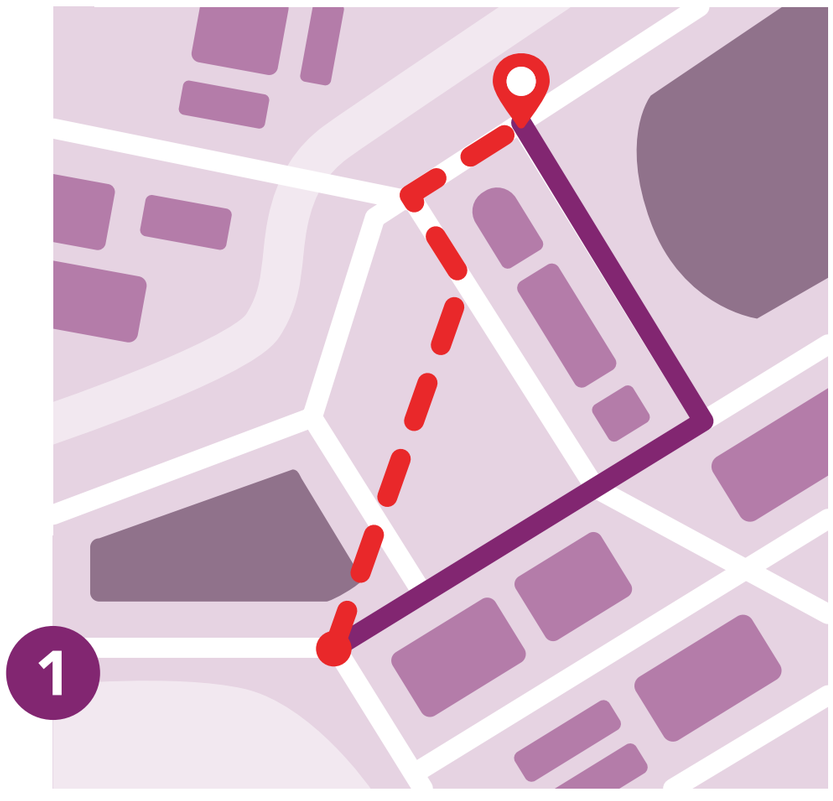
NN/g

www.nngroup.com/articles/recognition-and-recall/

Jakob's Heuristic

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.

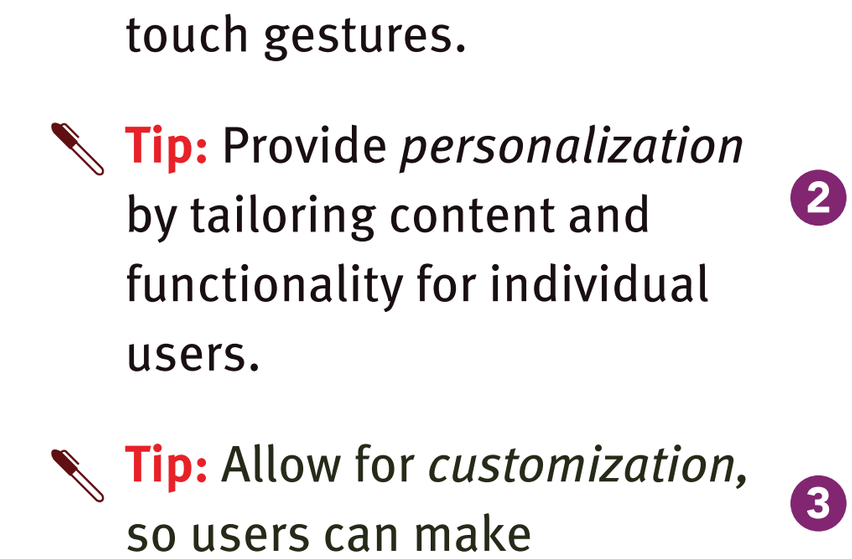
Tip: Provide *personalization* by tailoring content and functionality for individual users.

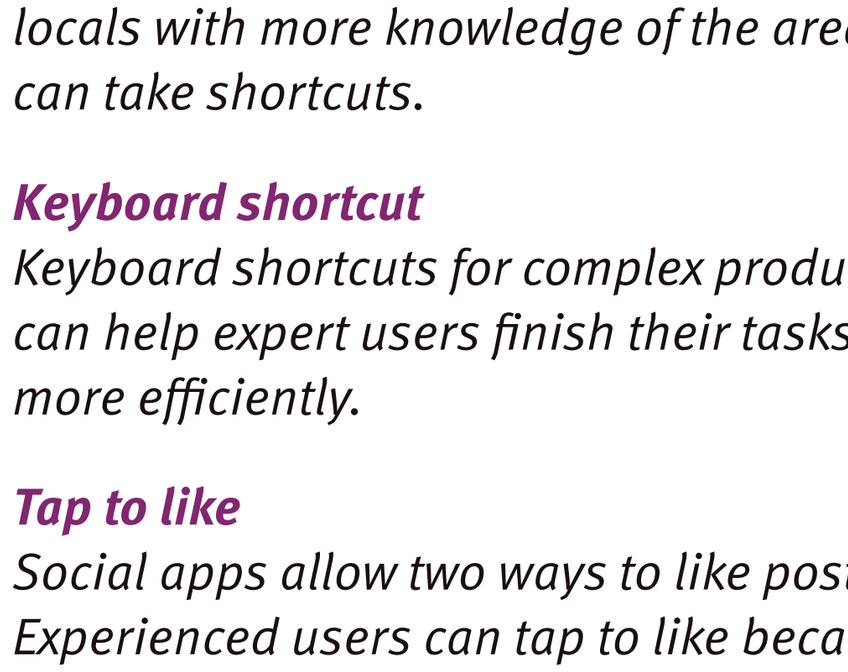
Tip: Allow for *customization*, so users can make selections about how they want the product to work.

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.





Jakob's Heuristic

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

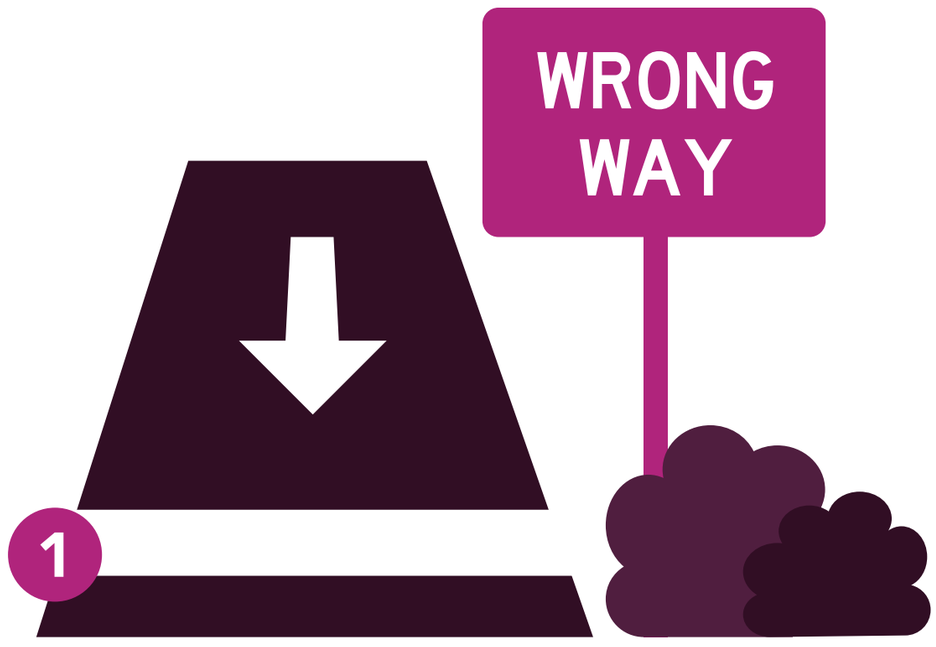
COMMUNICATE,
DON'T DECORATE

One of our favorite slogans

Jakob's Heuristic

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.

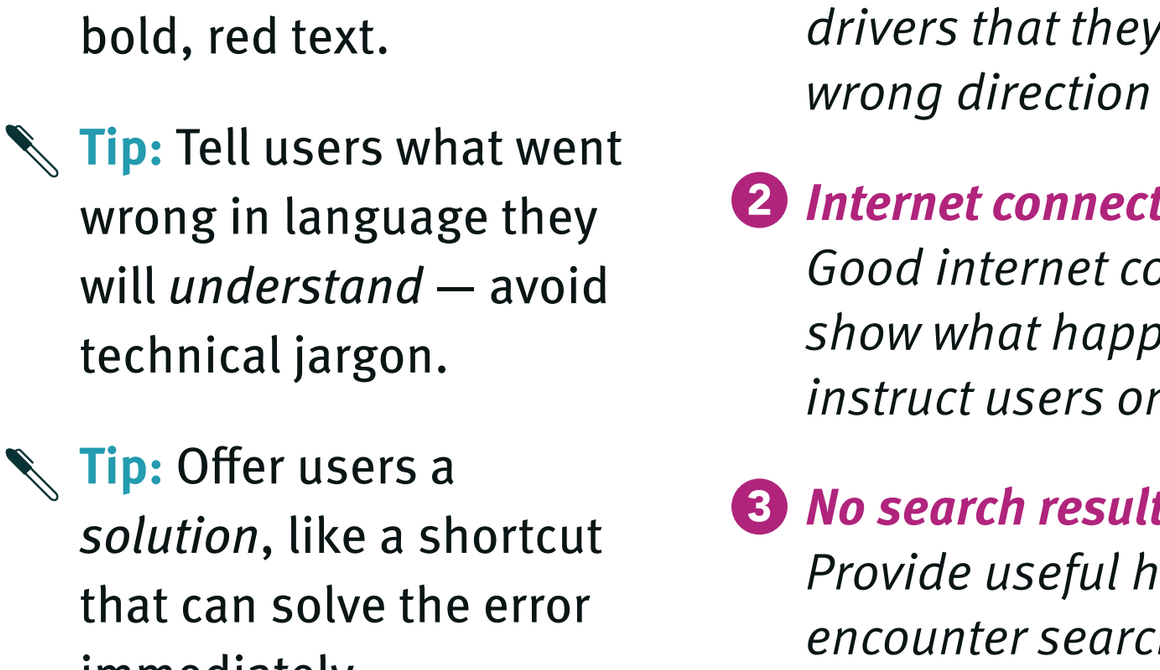
Tip: Tell users what went wrong in language they will *understand* — avoid technical jargon.

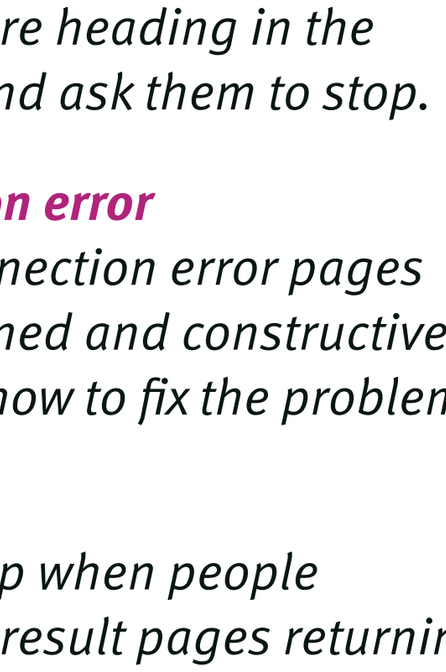
Tip: Offer users a *solution*, like a shortcut that can solve the error immediately.

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.





NN/g

www.nngroup.com/videos/usability-heuristic-recognize-errors/

Jakob's Heuristic

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.