UX/UI Design for Mobile Applications

204311

Written by Thapanapong Rukkanchanunt

Last update: Dec 18, 2023

Brief Overview of UX/UI Design

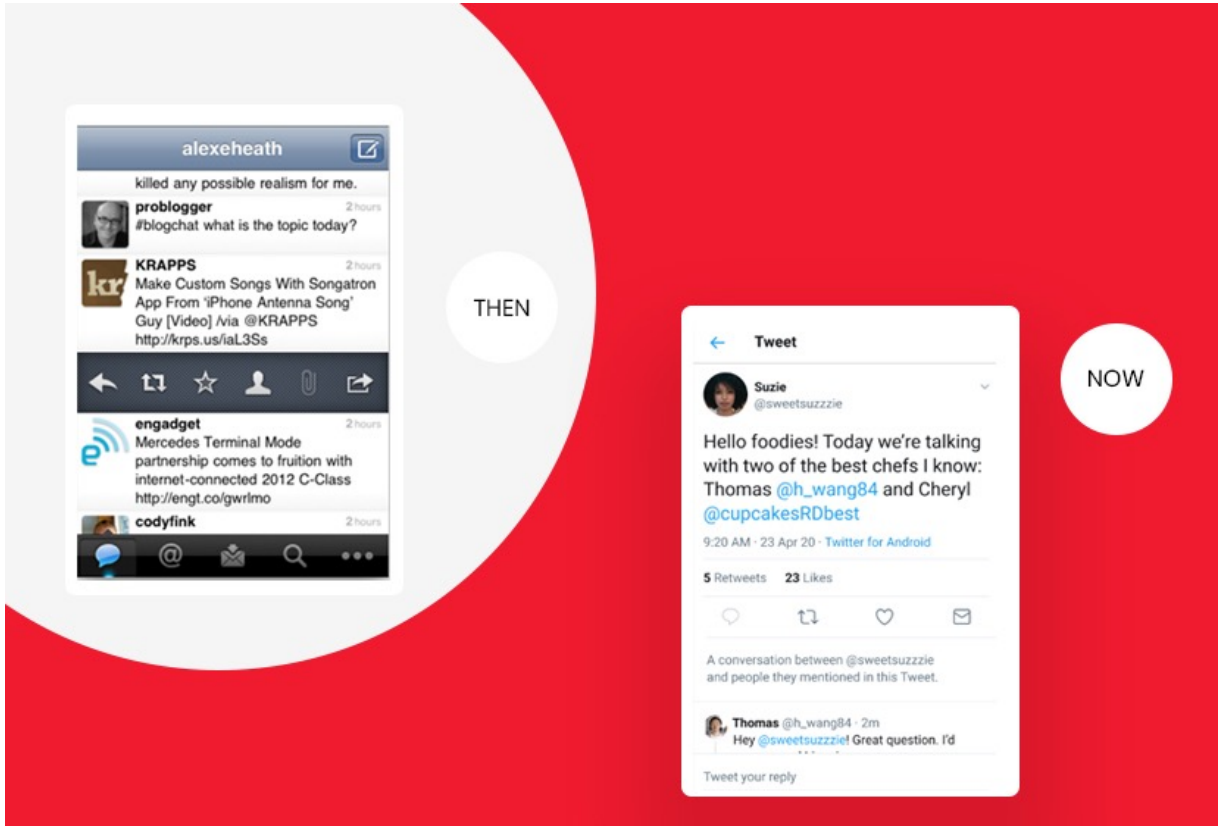
- User Experience (UX) is overall experience that a user has when using the applications.
- User Interface (UI) is an element that a user interact with.
- Understanding how UX/UI works will:
  - Make users happy.
  - Help users solve their problems.
  - Keep users engaged.
  - Expand the userbase.
  - Build reputation.

UX/UI Timeline

- Early vs Modern
- Major milestones

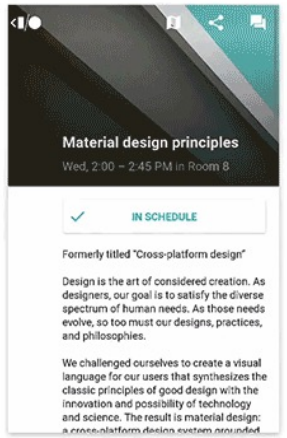
1992 IBM Simon

2002 BlackBerry

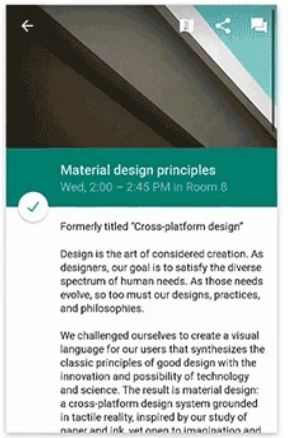


2007 iPhone

2014 Material Design



First release

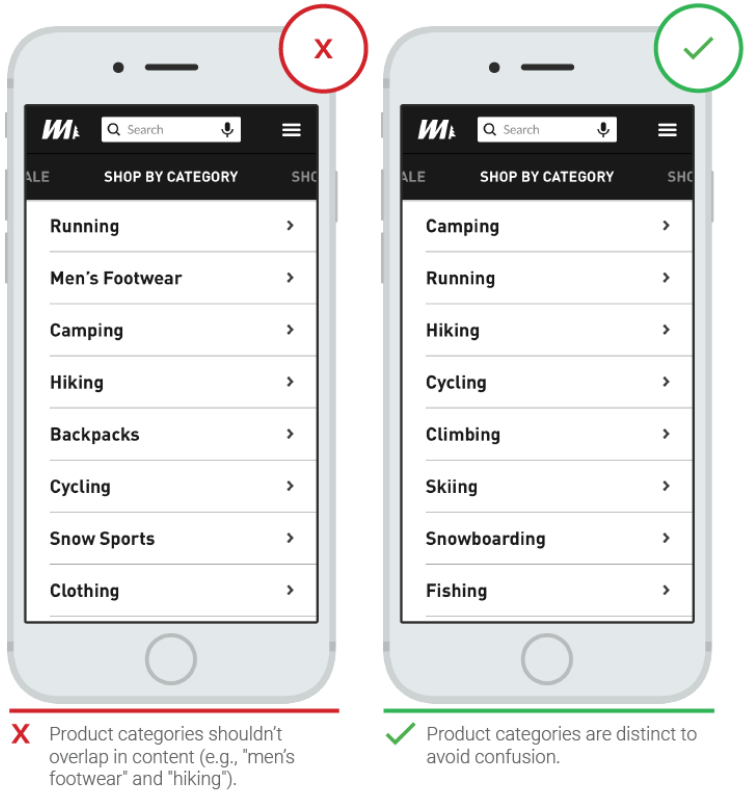


Updated version

Now

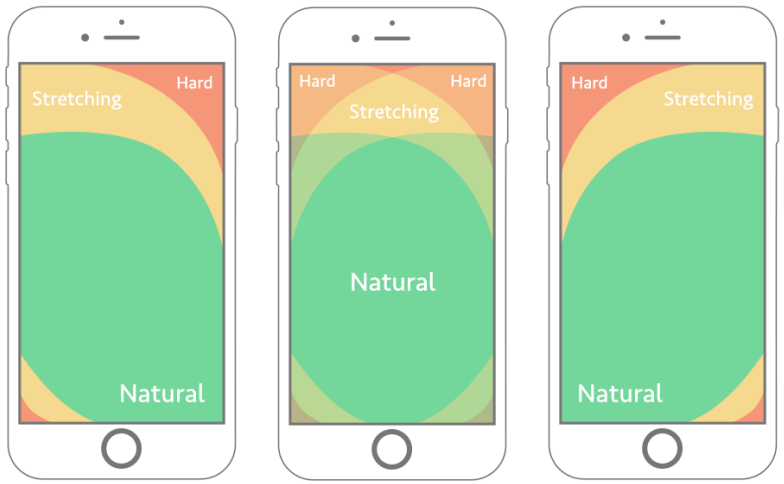
Key Concepts in UX

- Navigation – easy to move around app
- Personalization – keep unrelated contents away
- Usability – focus on one task
- Layout – follow thumb zone area
- Data Requirement – get necessary data as early as possible

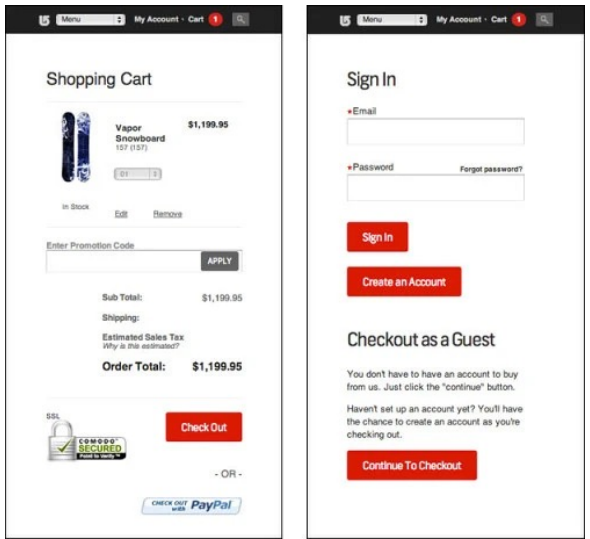


X Product categories shouldn't overlap in content (e.g., "men's footwear" and "hiking").

Product categories are distinct to avoid confusion.

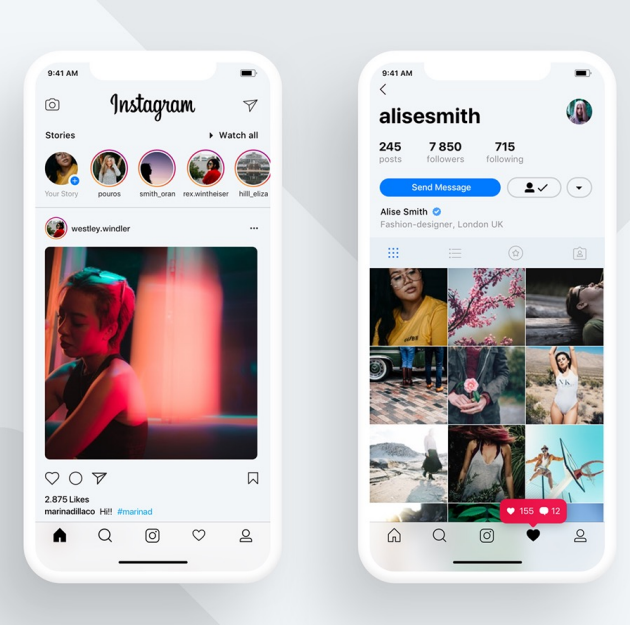


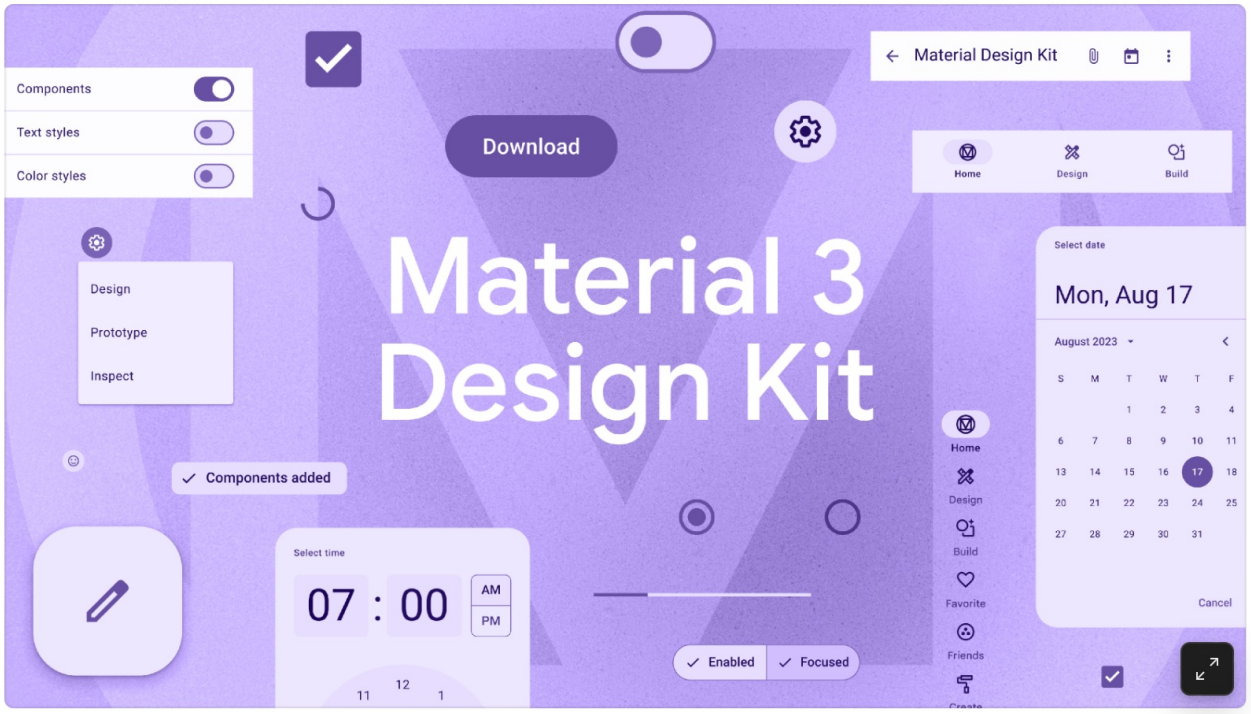
Left Hand *Combined* *Right Hand*



Key Concepts in UI

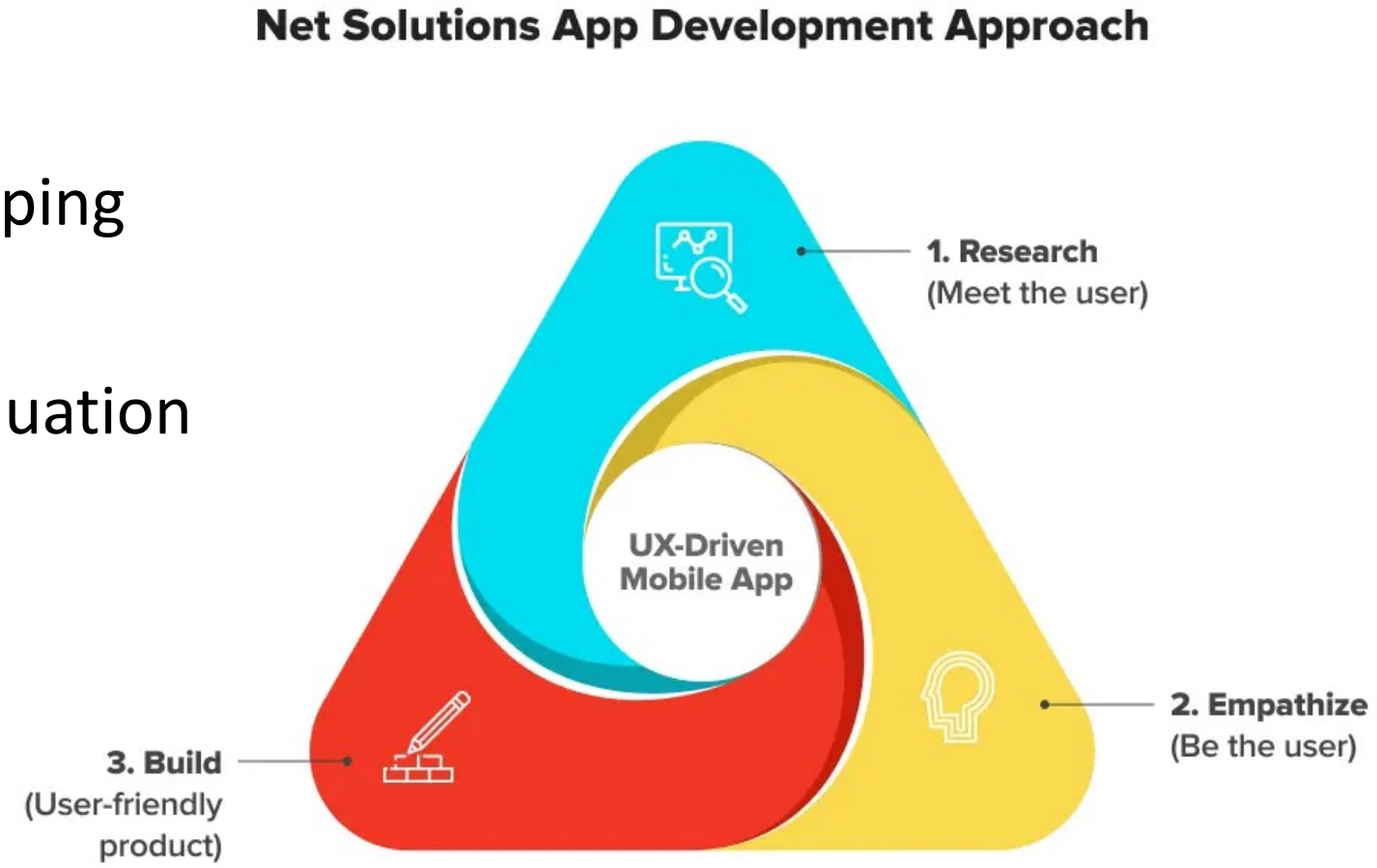
- Visual design elements – follow material design
- Consistency – maintain a uniform appearance and behavior
- Responsiveness – adapt to size
- Branding – color, typography





How Do We Do UX/UI?

- Research and analysis
- Wireframing and prototyping
- User testing
- Implementation and evaluation



net solutions

How Do We Do UX/UI?

- Research and analysis
  - Understanding users' needs (feature requirement)
  - Imagining how users would use your application (user flow)
- Wireframing and prototyping
- User testing
- Implementation and evaluation

Heuristic Evaluation

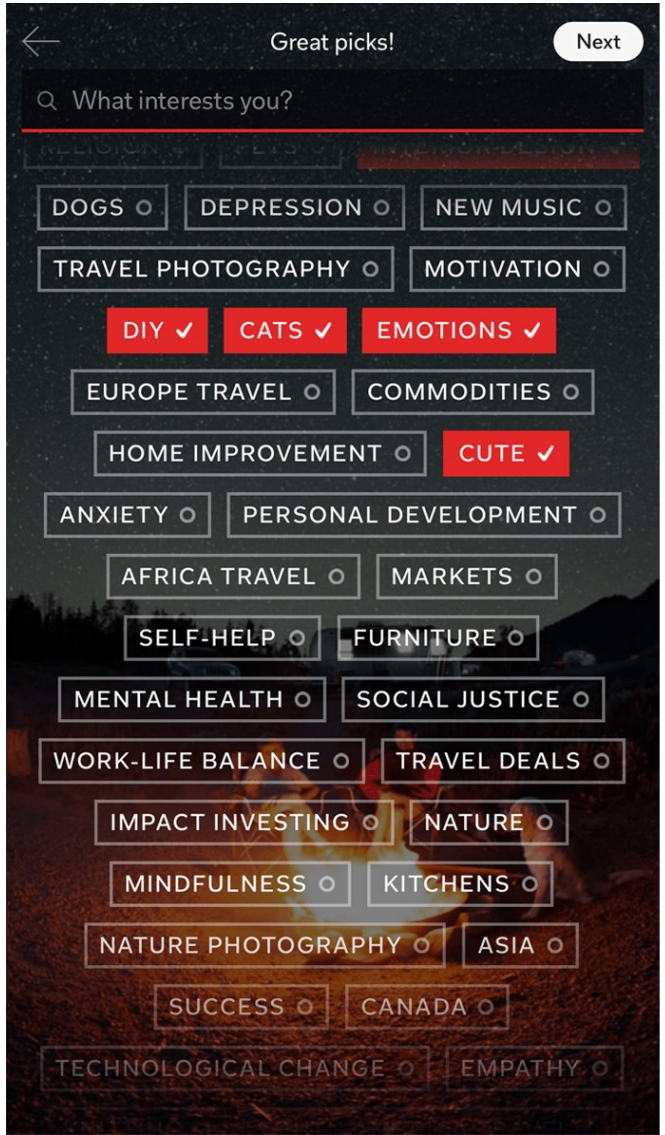
- A heuristic evaluation is a method for identifying design problems in a user interface.
- Evaluators judge the design against a set of guidelines (called heuristics) that make systems easy to use.
- “Heuristic” can also mean subjective so there are many such guidelines. We will follow Jakob Neilsen’s 10 general principles for interactive design.

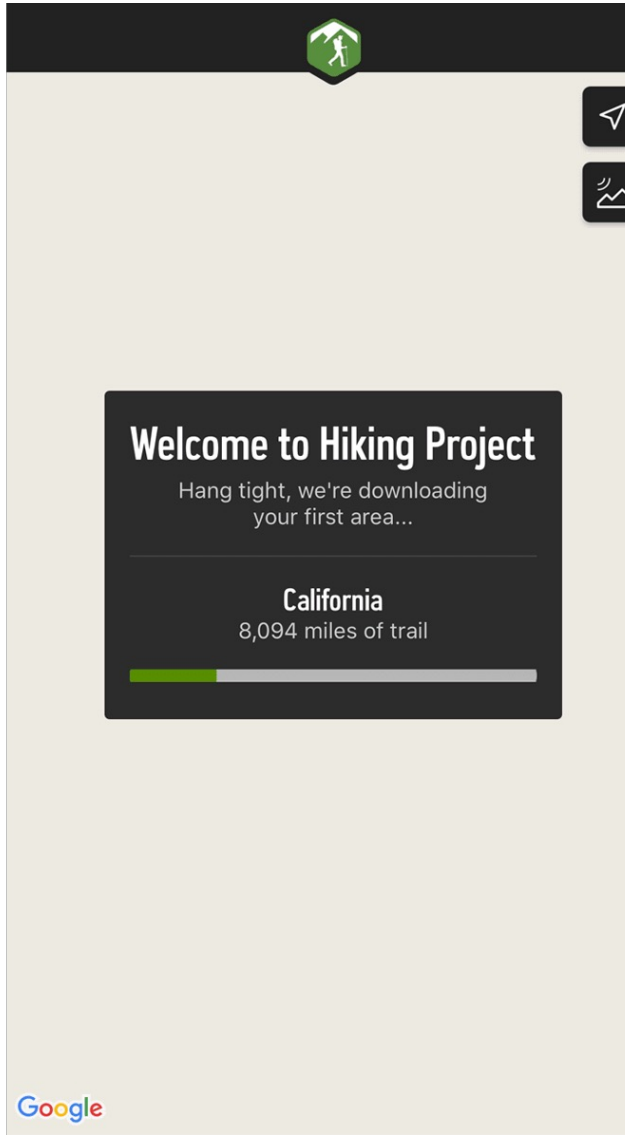
Jakob Neilson's 1st Principle

- The design should always keep users informed about what is going on, through appropriate feedback within a reasonable amount of time.
- When users know the current system status, they learn the outcome of their prior interactions and determine next steps. Predictable interactions create trust in the product as well as the brand.
- Example: You Are Here indicators on map.

Appropriate Feedback

- Changing the color and adding a checkmark to buttons on a selection screen communicates that the system has registered the user's choices (left).
- Progress indicators reassure the user that a longer wait is normal, and that the system is still working (right).



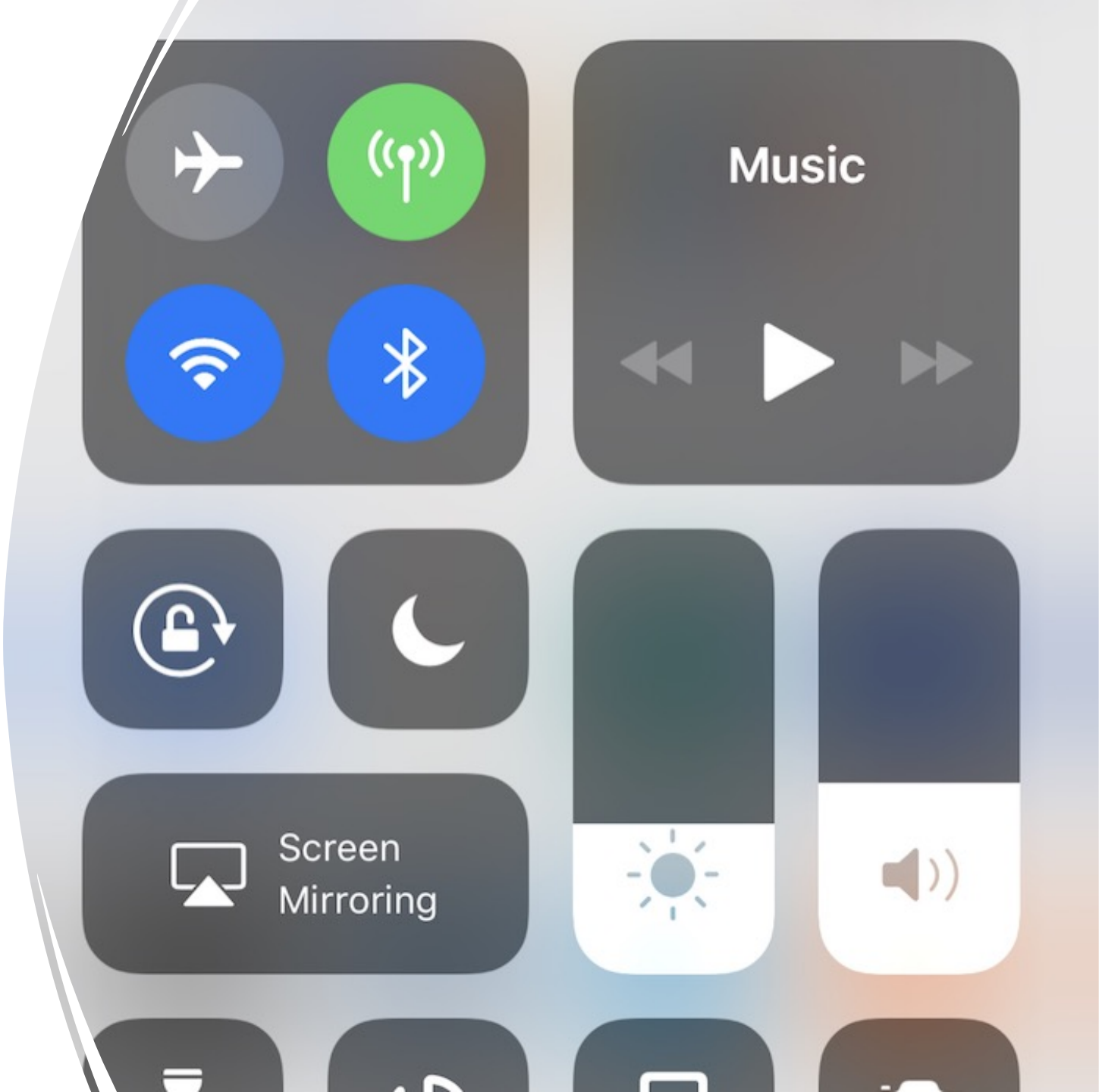


Jakob Neilson’s 2nd Principle

- 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.
- When a design’s controls follow real-world conventions and correspond to desired outcomes (called natural mapping), it’s easier for users to learn and remember how the interface works. This helps to build an experience that feels intuitive.

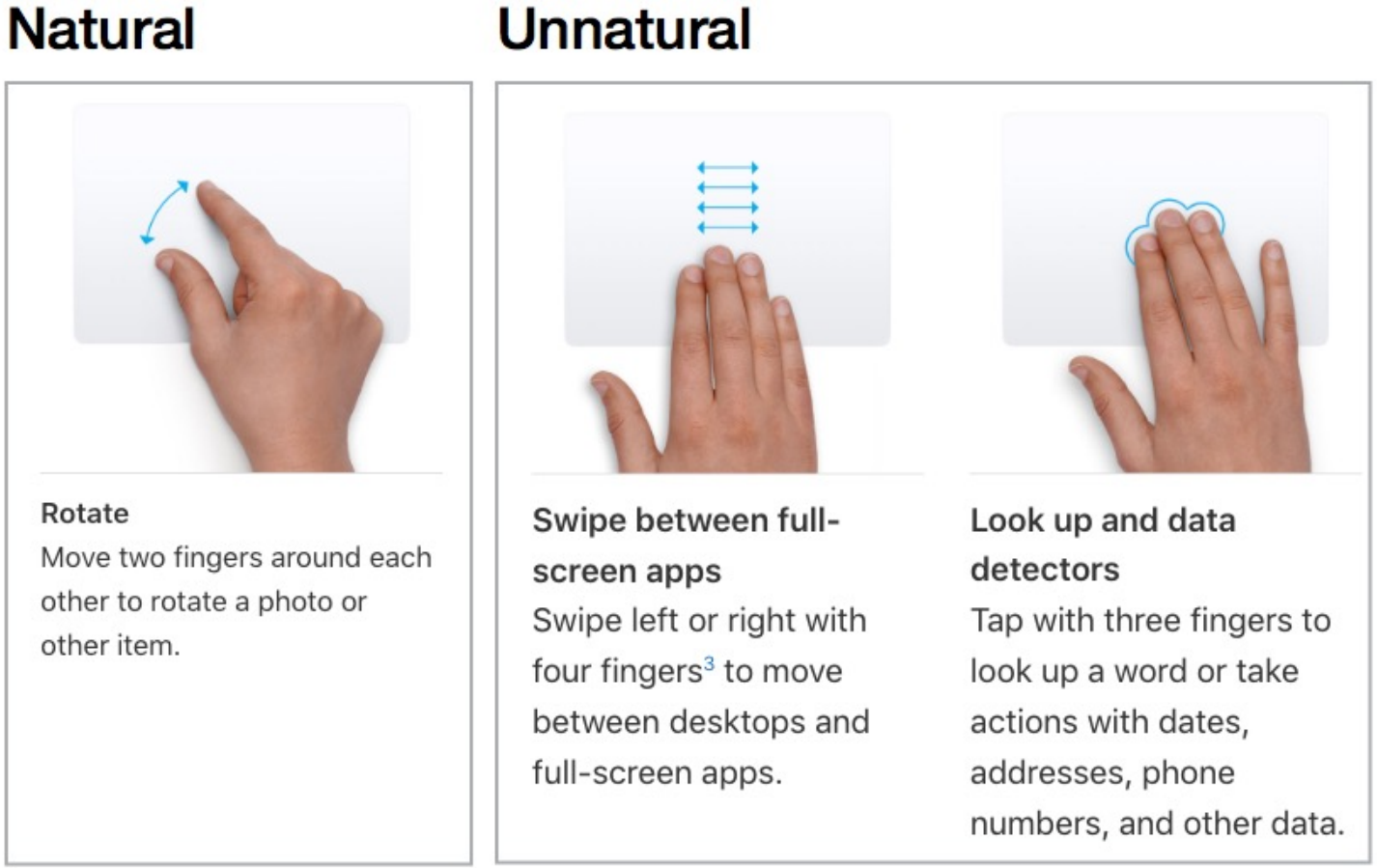
Conceptual and Metaphorical Similarity

- The brightness control in the iOS Control Center uses natural mapping by arranging the control so that swiping up raises the screen brightness (up is more) and swiping down darkens the screen (down is less).



Behavioral Similarity

- The rotate gesture on a trackpad (left) has a natural mapping to how a person might rotate an object in real life. However, many gestures, such as the four-finger swipe and three-finger tap, aren't clearly associated to the action being carried out, and therefore they are challenging for users to learn and execute

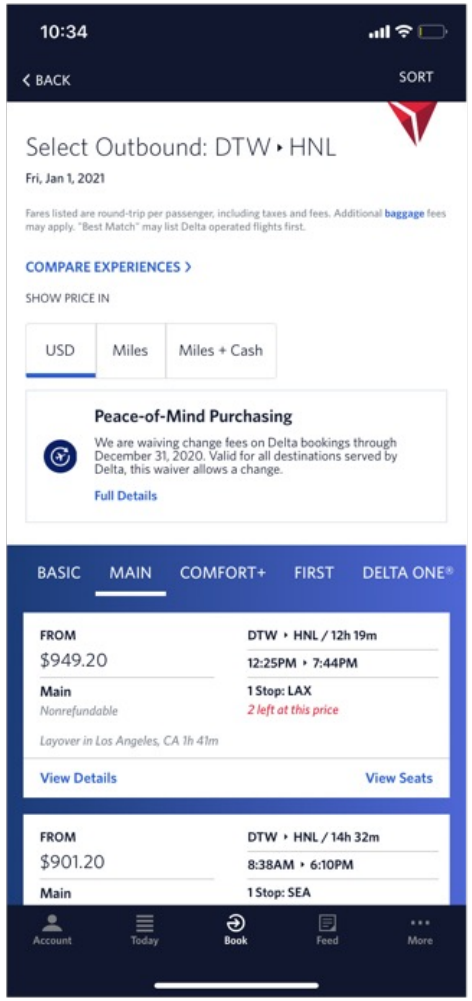


Jakob Neilson’s 3rd Principle

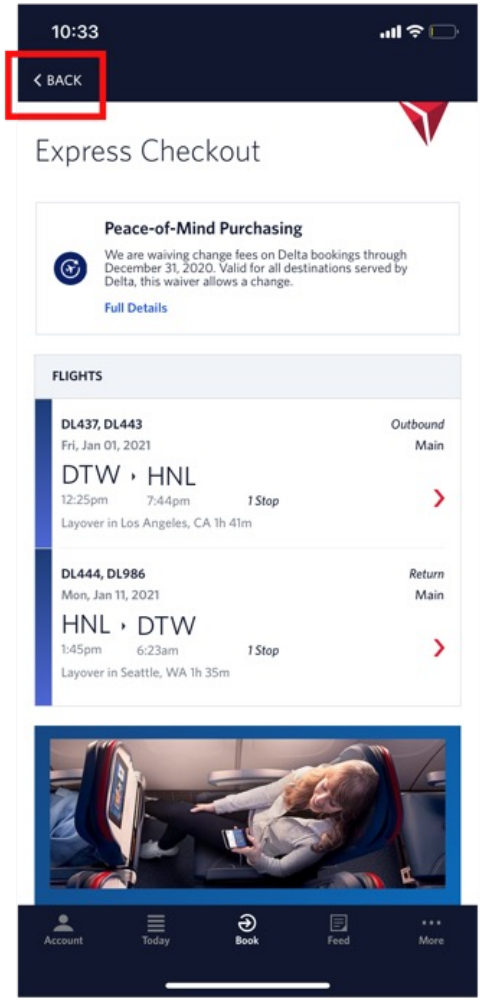
- Users often perform actions by mistake. They need a clearly marked "emergency exit" to leave the unwanted action without having to go through an extended process.
- When it's easy for people to back out of a process or undo an action, it fosters a sense of freedom and confidence. Exits allow users to remain in control of the system and avoid getting stuck and feeling frustrated.

True Meaning of Back Button

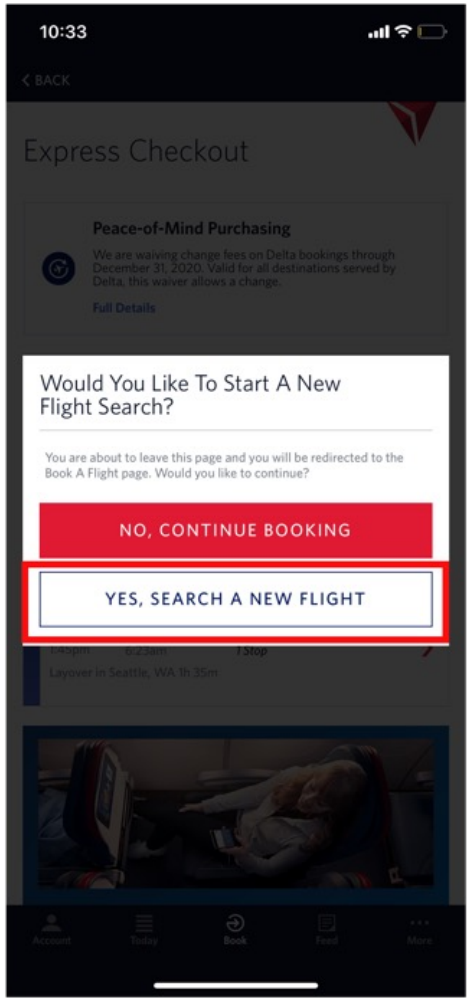
1



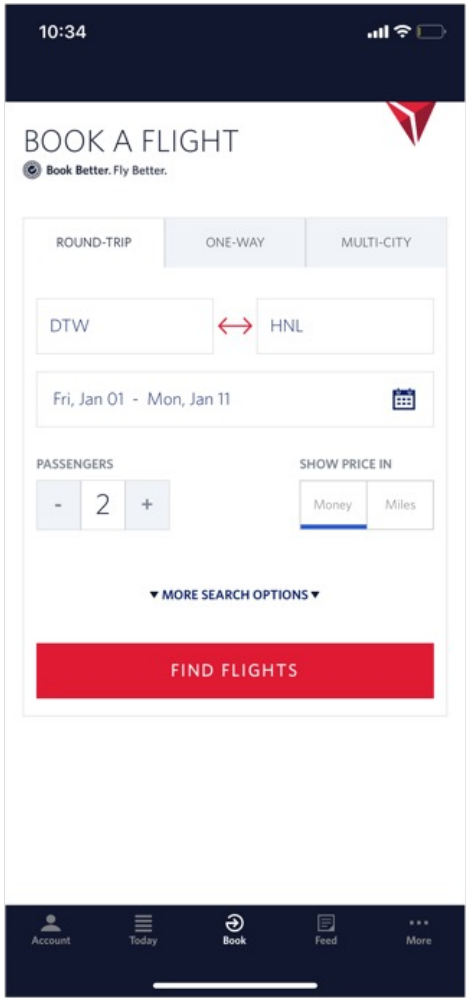
2



3

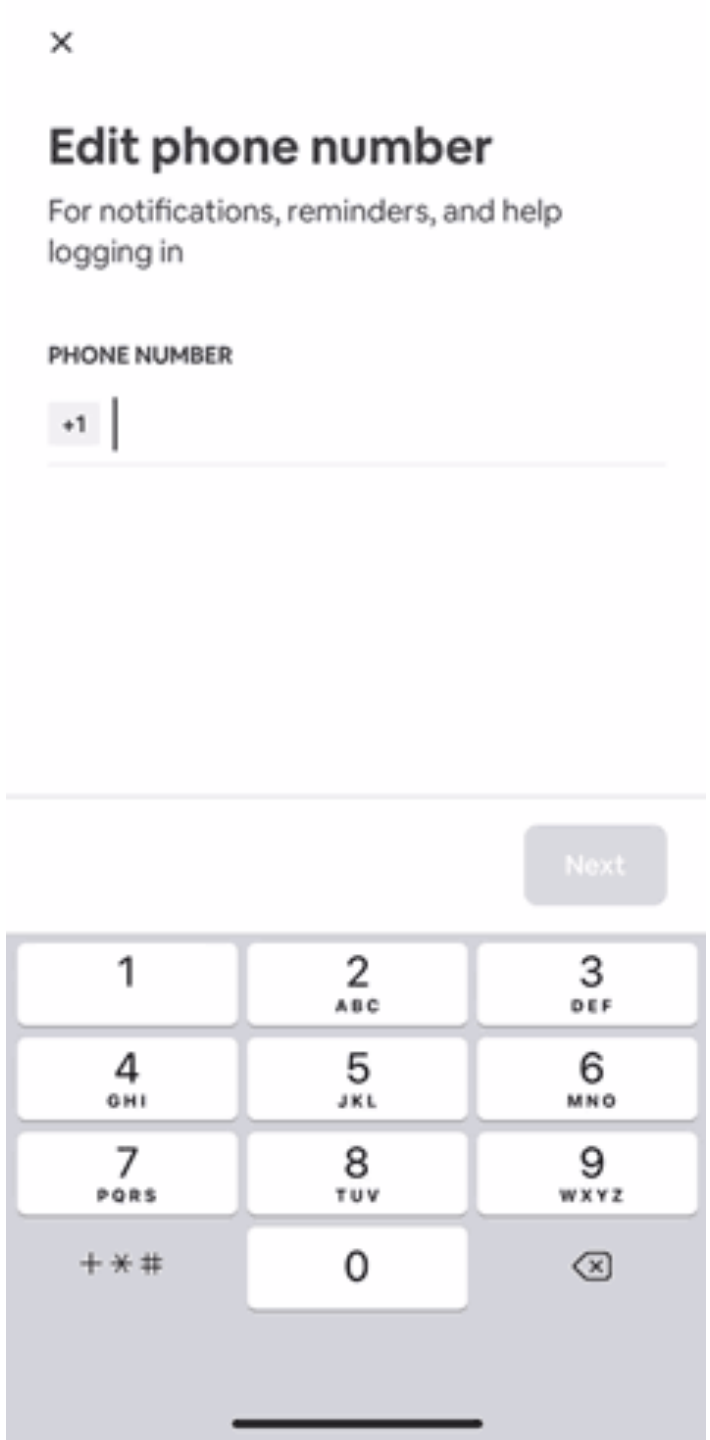


4

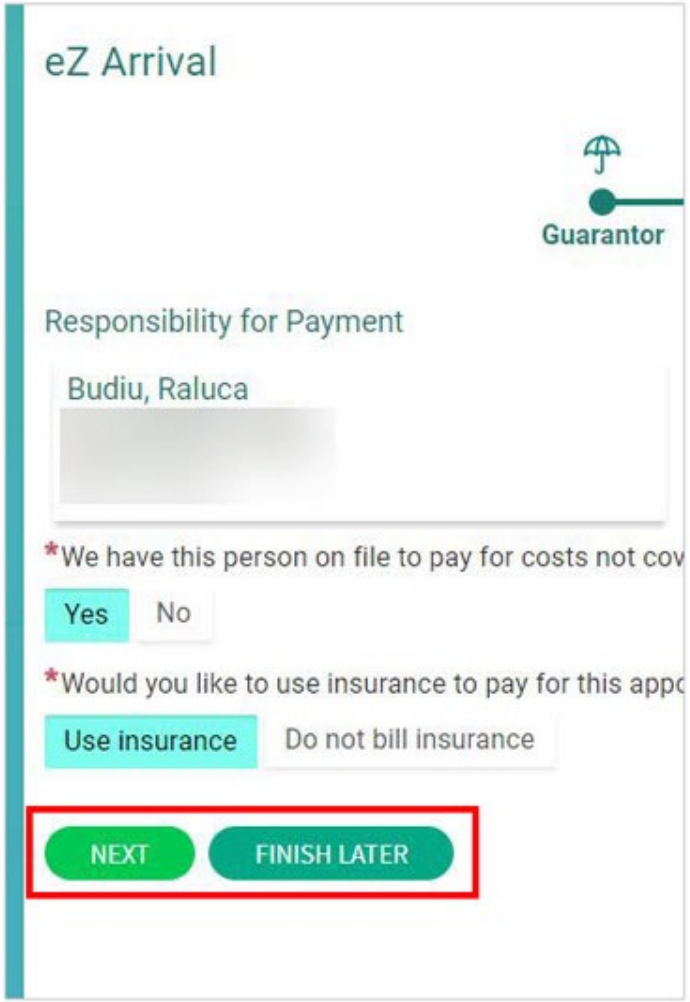


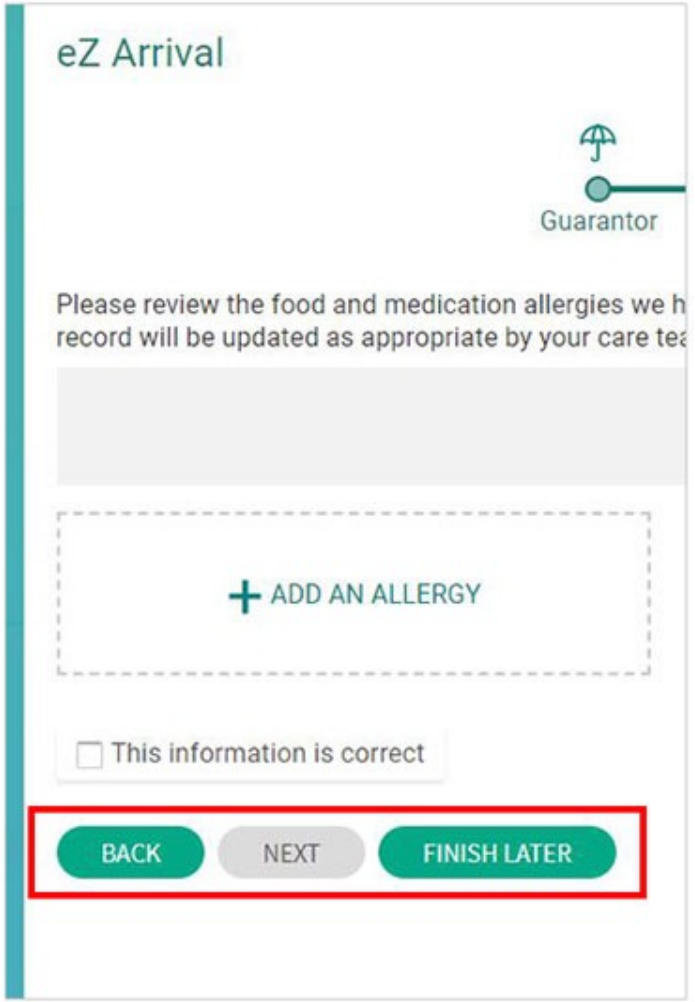
Jakob Neilson's 4th Principle

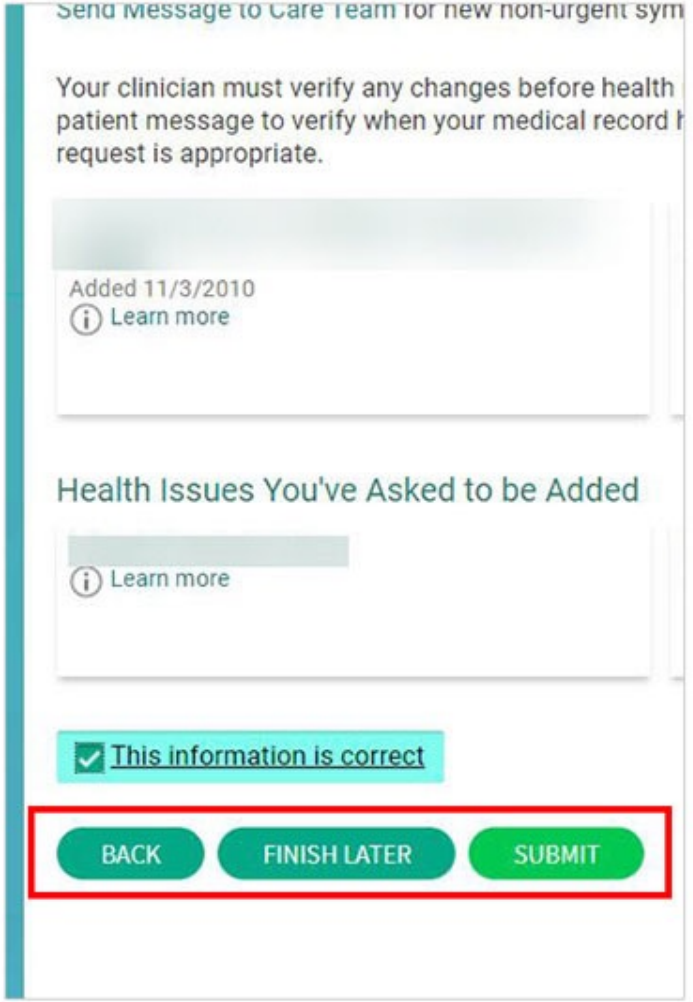
- Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform and industry conventions.
- Creating a design system for your application or family of applications can help teams maintain consistency across products and services.



Buttom Layout Example





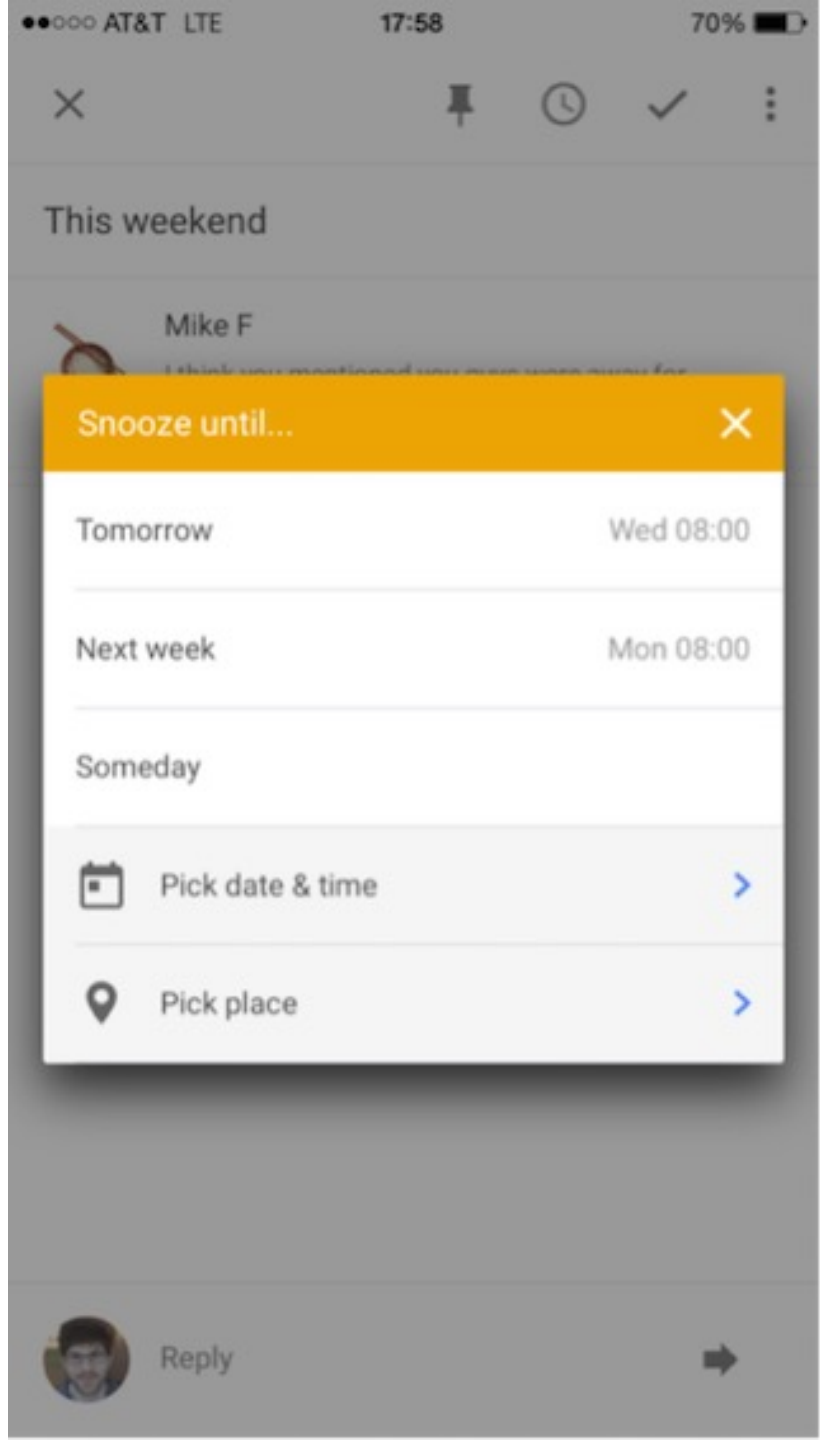


Jakob Neilson's 5th Principle

- 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.

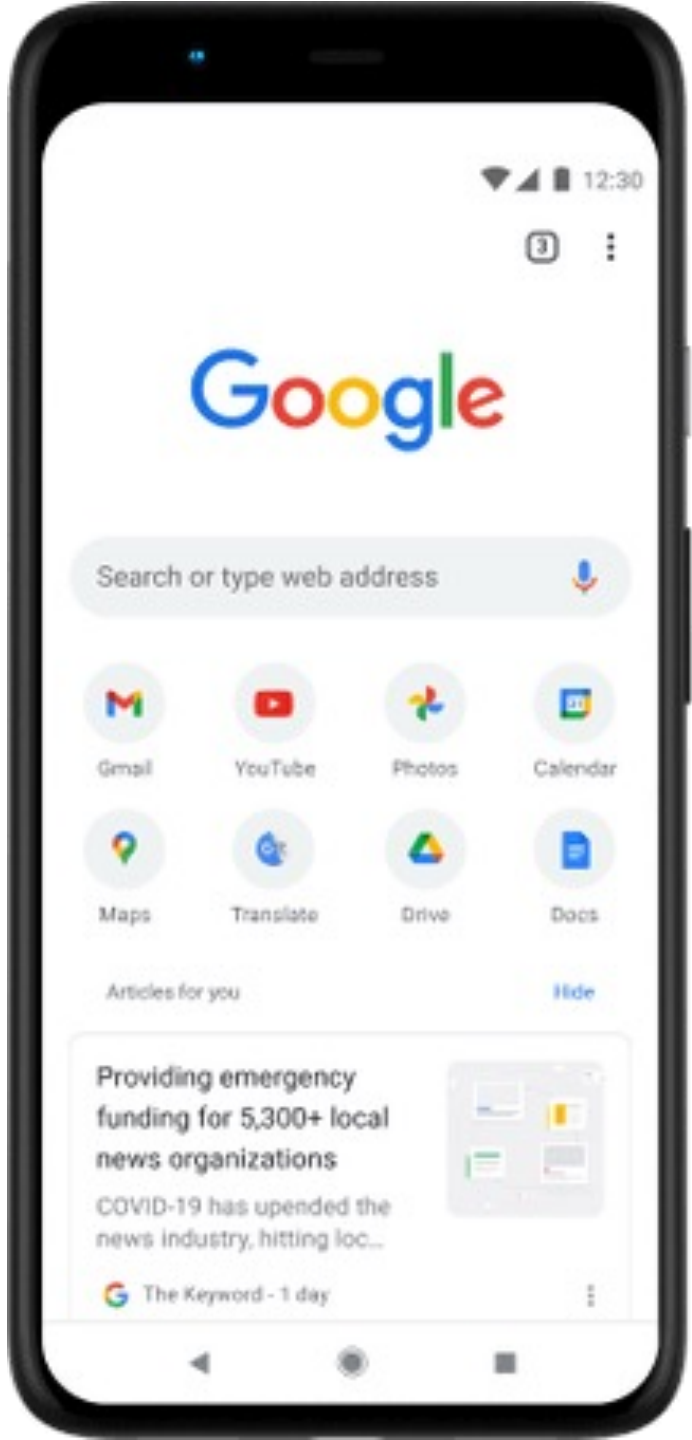
Choose Good Defaults

- Google's Inbox app for iOS allows you to "snooze" an email until a later time. The default options are sensible and prevent typing errors for common choices.



Jakob Neilson's 6th Principle

- 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 (e.g. field labels or menu items) 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.

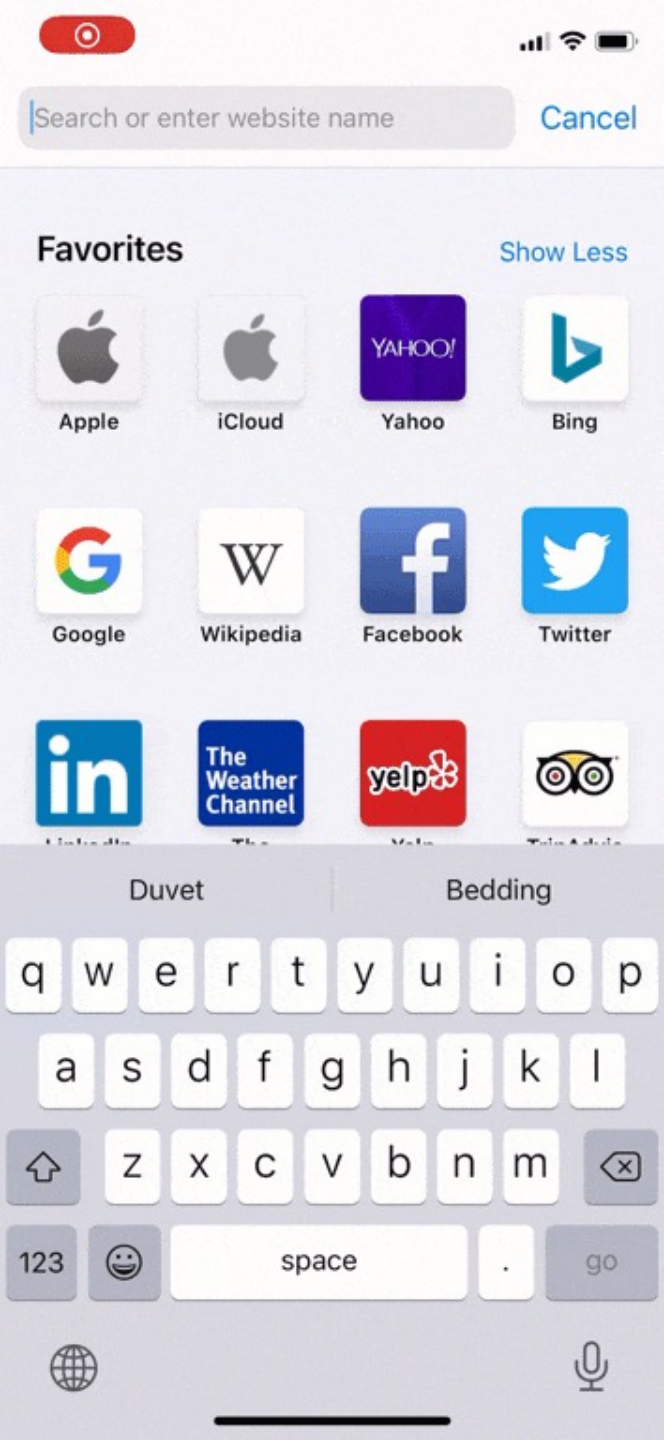


Jakob Neilson's 7th Principle

- Shortcuts — hidden from novice users — may speed up the interaction for the expert user so that the design can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.
  - Provide accelerators like keyboard shortcuts and touch gestures.
  - Provide personalization by tailoring content and functionality for individual users.
  - Allow for customization, so users can make selections about how they want the product to work.

Accelerators Improve Repetitious Usage

- Accelerators are secondary ways of accomplishing the same task that function as faster (but typically less obvious) methods.
- An example of an accelerator is how modern mobile keyboards allow users to perform a swipe gesture over the letters to input text, rather than tapping each individual letter.
- This gesture is an enhancement: it doesn't get in the way of a new user (who very likely is not aware of it at all) but can save a lot of time to a seasoned user.

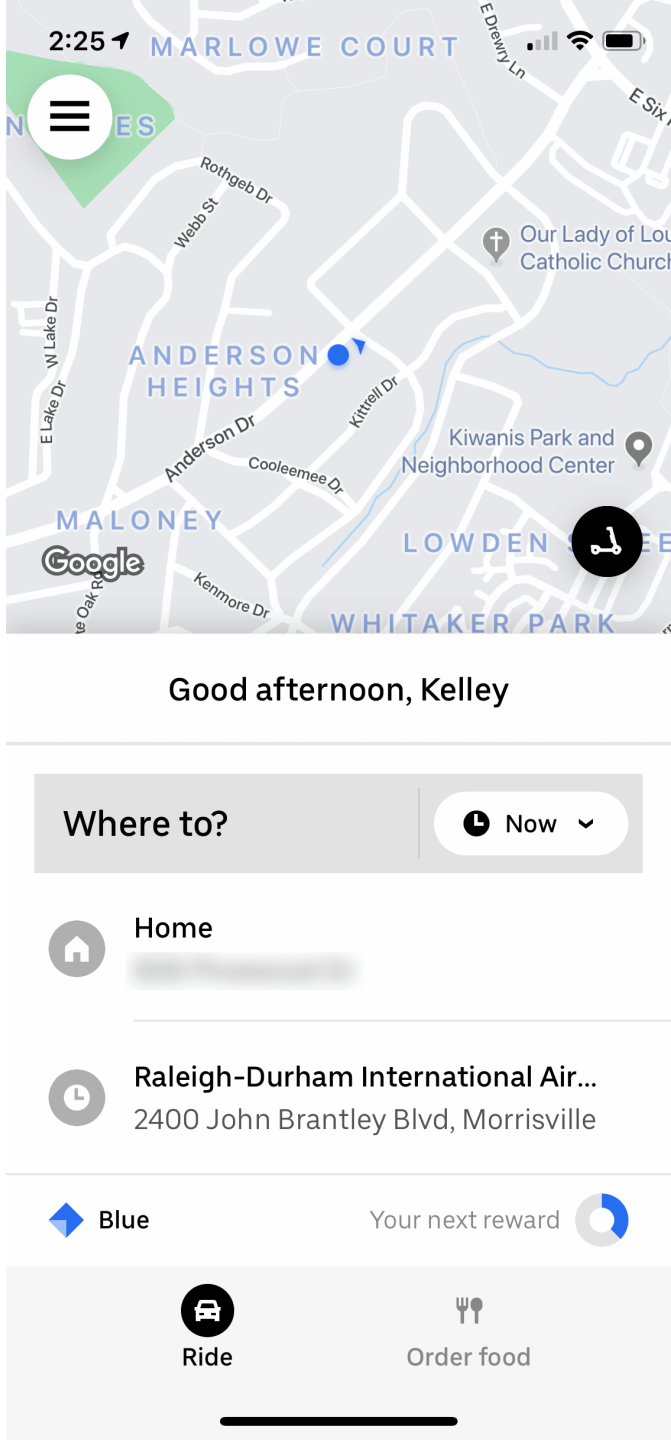


Jakob Neilson’s 8th Principle

- Interfaces should not contain information that 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 heuristic 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 interface support the user's primary goals.

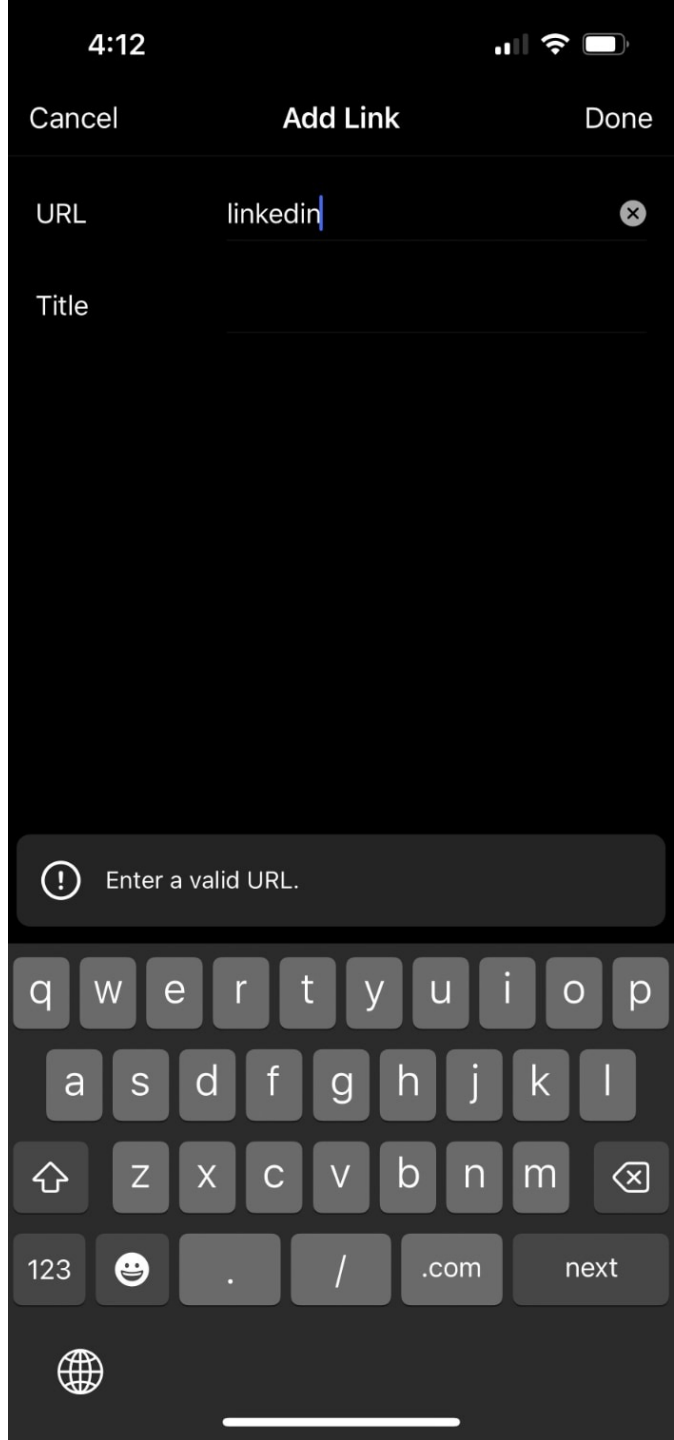
Visual Hierarchy

- Uber mobile app: The visual hierarchy is clear in Uber's mobile app. The screen is split in half between the map and input form (bottom half of screen), which enforces the thought that these components are equally important to the user. The eye is immediately drawn to the Where to? field because of its gray background, then to the recent locations below it, which are slightly smaller in font size.



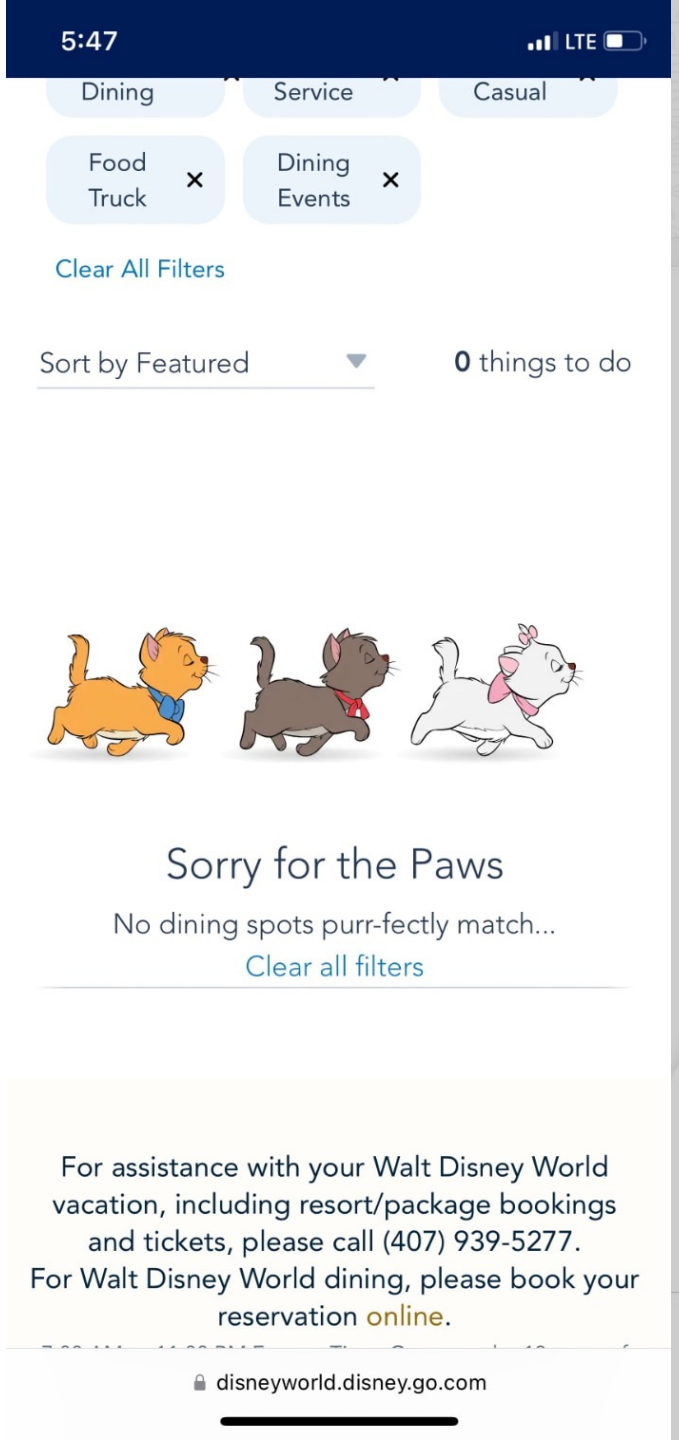
Jakob Neilson's 9th Principle

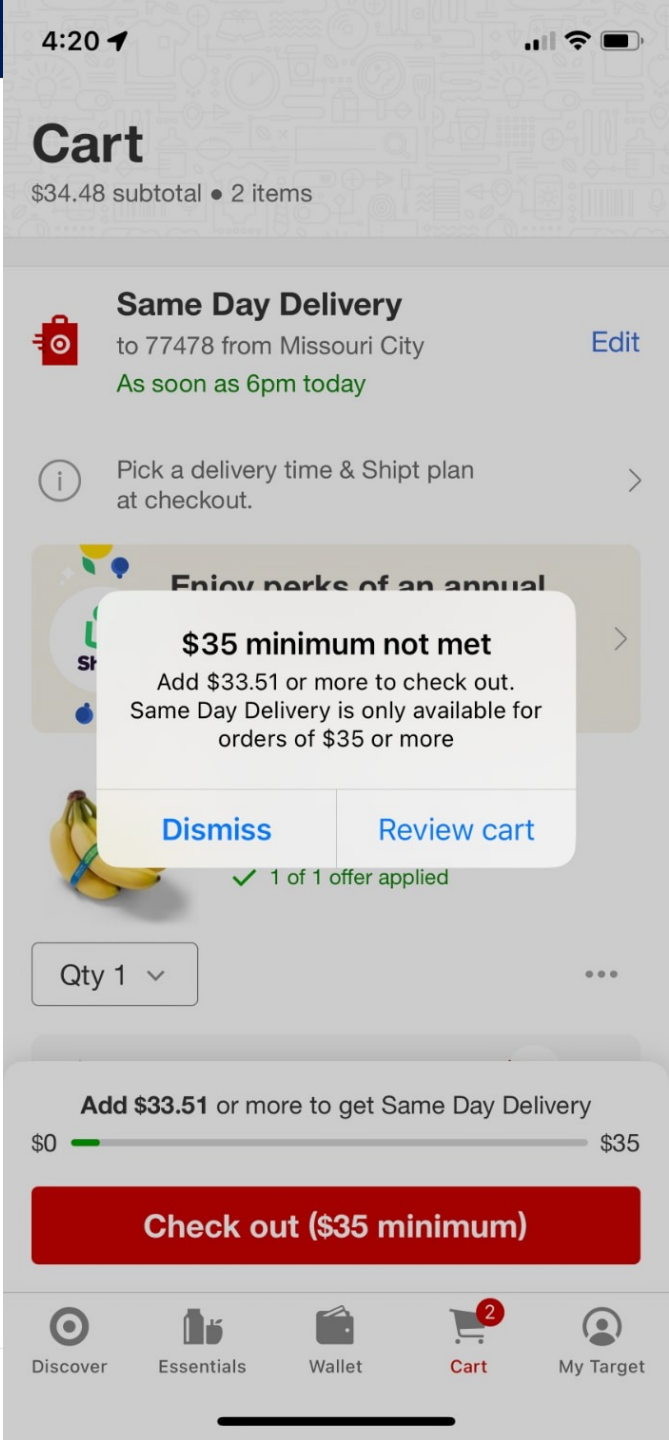
- Error messages should be expressed in plain language (no error codes), precisely indicate the problem, and constructively suggest a solution.
- These error messages should also be presented with visual treatments that will help users notice and recognize them.
- Bad: When the user adds an unrecognizable URL to their Instagram profile, the resulting error message is subtly styled and positioned far away from the URL field.



Bad and Good

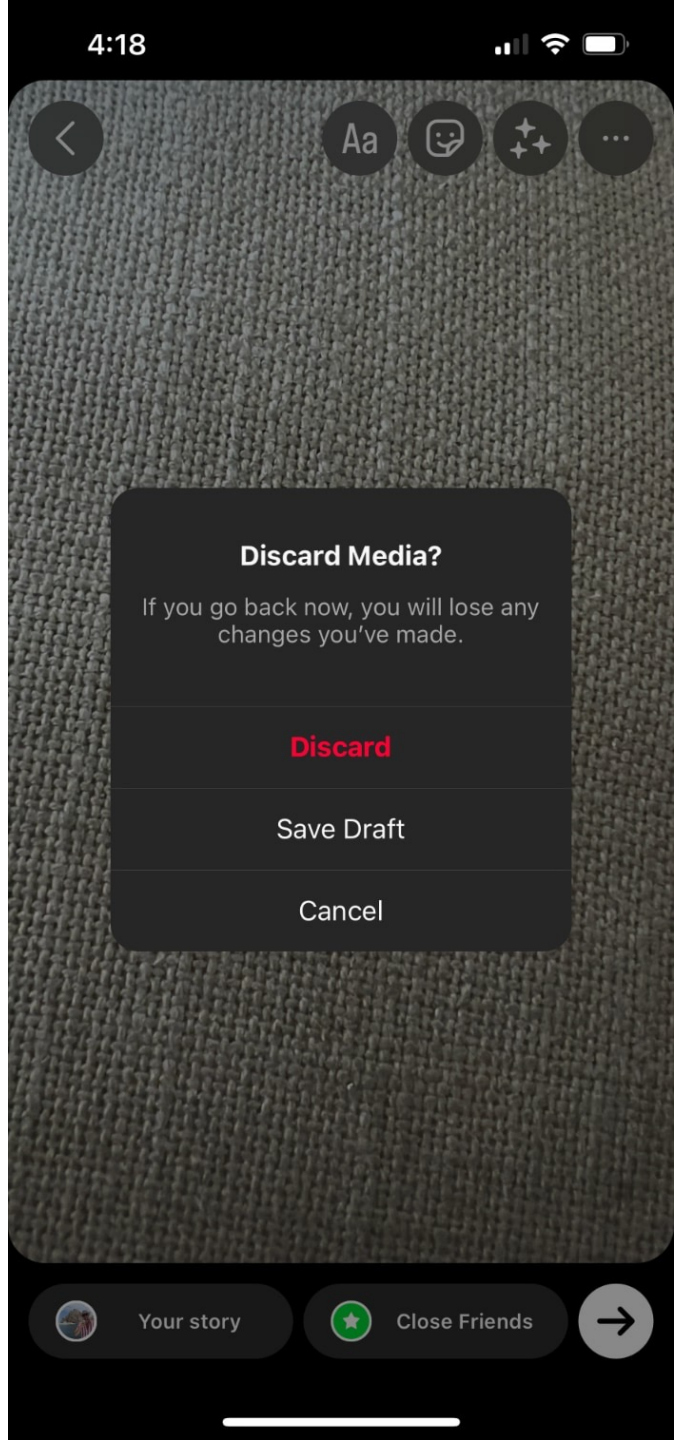
- Bad: When searching for dining locations with very narrow filters, Disneyworld obfuscates the lack of results with puns instead of clearly conveying the situation.
- Good: Target gives clear feedback that users must spend more to qualify for same-day shipping. Note how the message avoids blaming the user for not purchasing enough and focuses instead on the threshold.





Preserve The User's Input

- Let users correct errors by editing their original action instead of starting over.
- For example, display the original text entered a text field even if it does not match the requirements for that field and allow the user to modify it.
- Good: If the user navigates back while adding an image to an Instagram story, the system provides an option to save their work as a draft to avoid losing it.

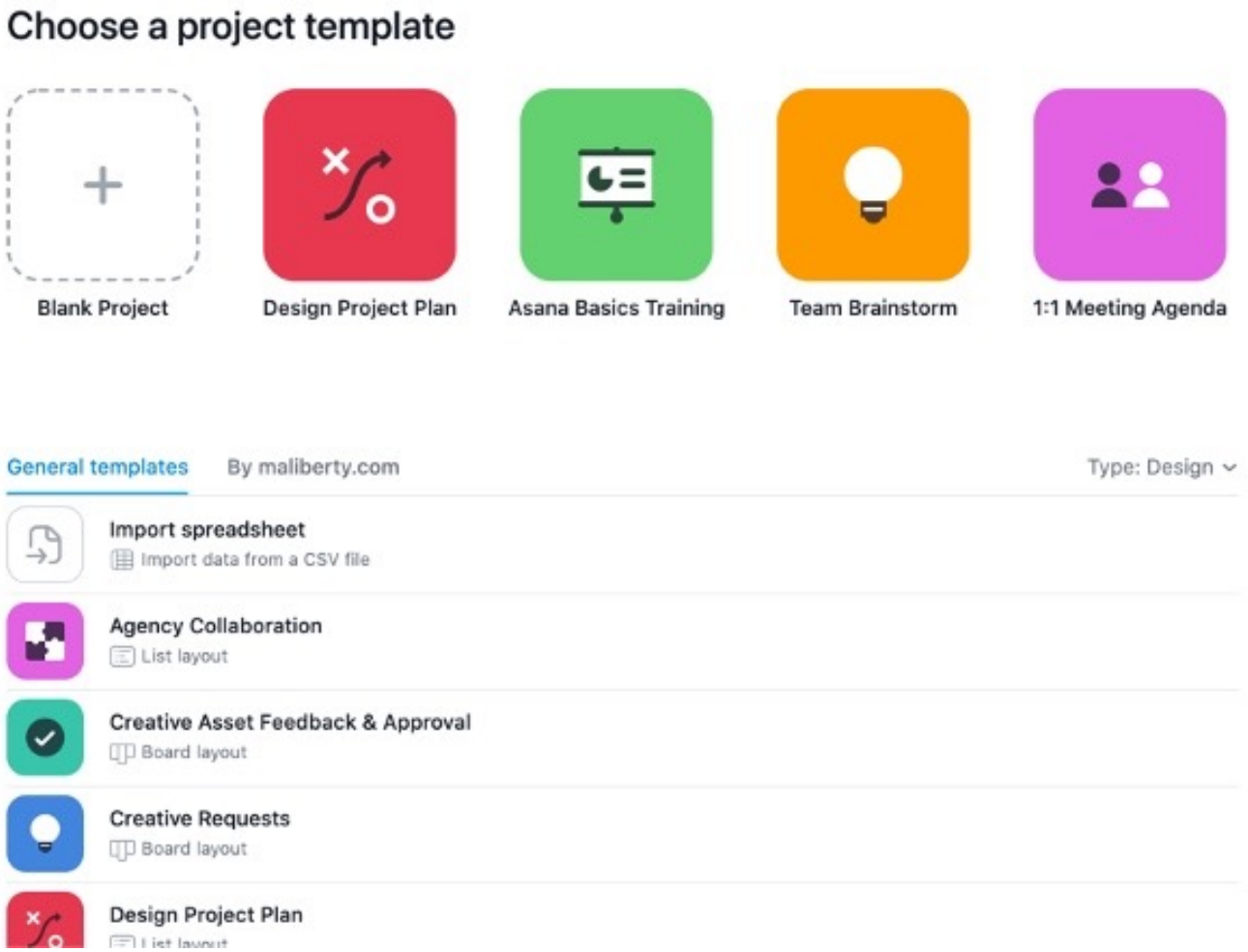


Jakob Neilson’s 10th Principle

- It’s best if the system 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.

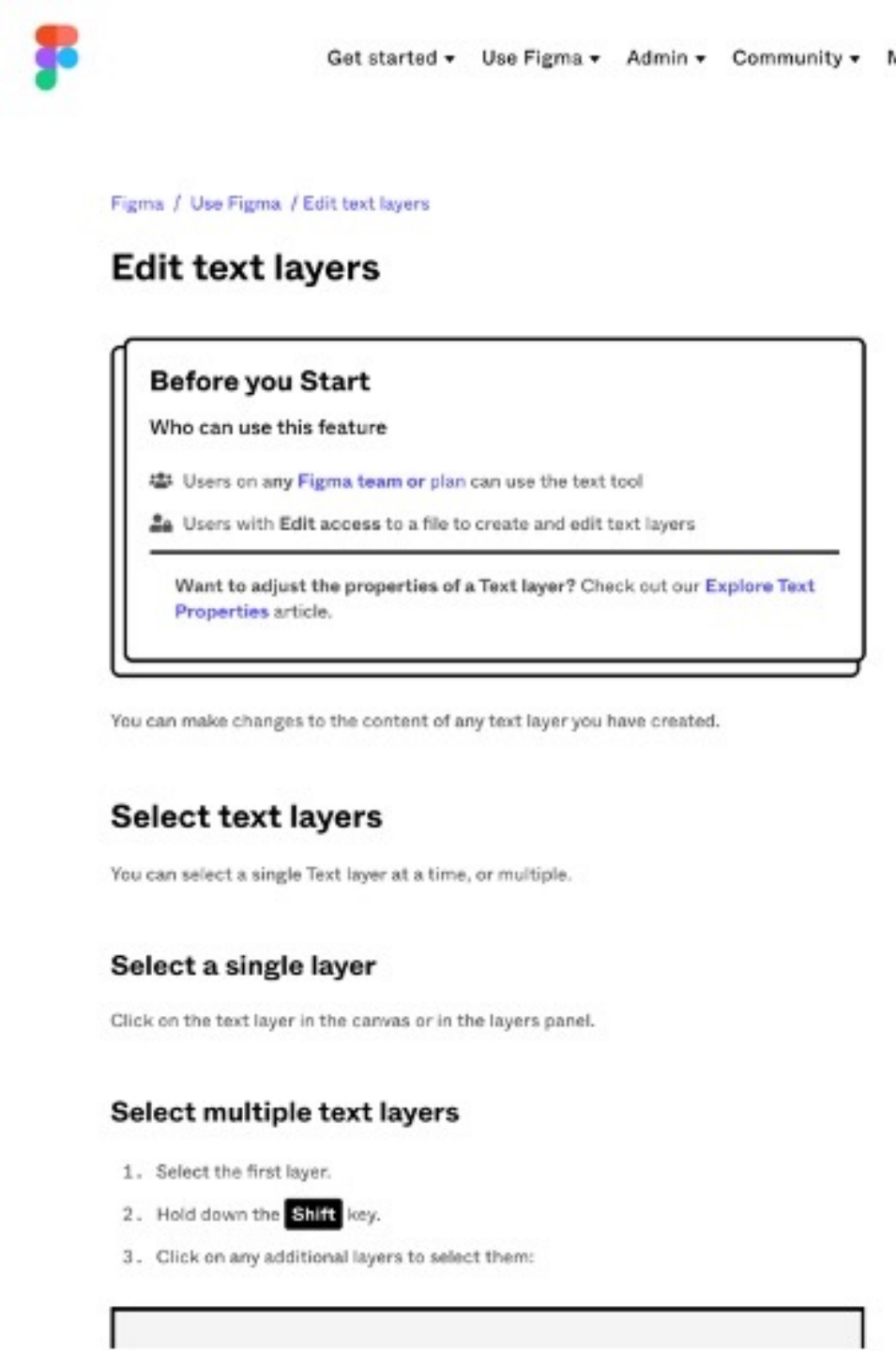
Proactive Help

- Proactive help can be implemented through tutorials, instructional overlays, templates, contextual help, tooltips, and wizards.

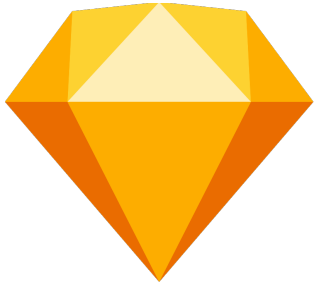


Reactive Help

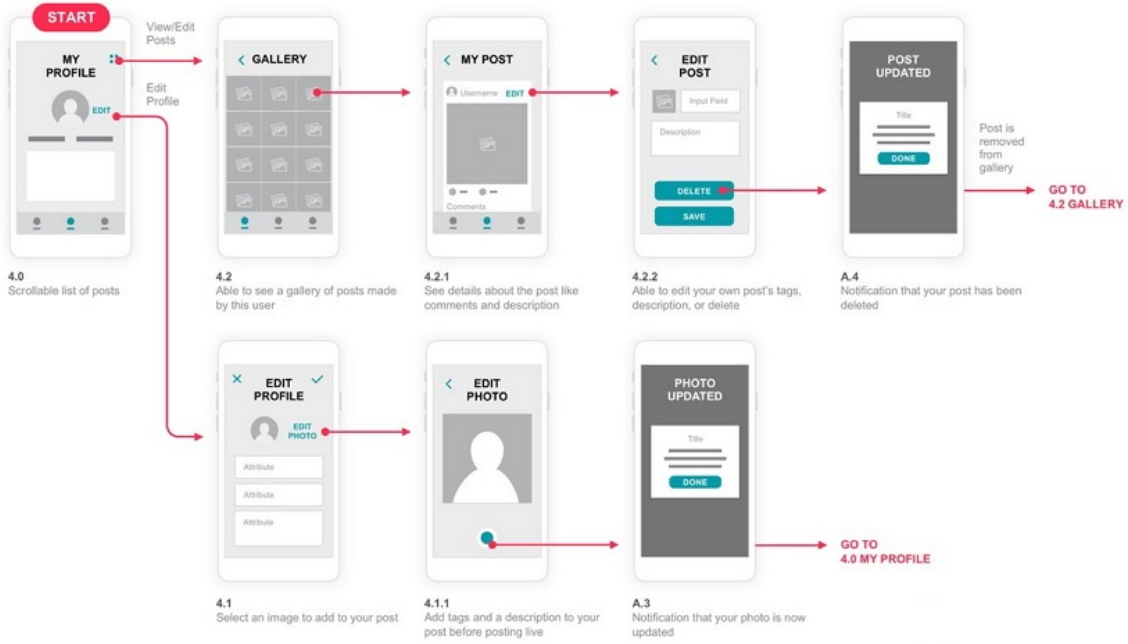
- The goal of reactive help is to answer questions, troubleshoot user problems, or provide detailed documentation and materials for people who want to become expert users.
- The support pages for Figma, a prototyping tool, were easy to scan. They had clear headings and numbered lists.



How Do We Do UX/UI?



- Research and analysis
- Wireframing and prototyping
  - Sketch
  - Adobe XD (discontinued as of June 22, 2023)
  - Figma
- User testing
- Implementation and evaluation



How Do We Do UX/UI?

- Research and analysis
- Wireframing and prototyping
- User testing
  - Find target audiences
  - Brief participants
  - Record the process
- Implementation and evaluation

In-Class Activity

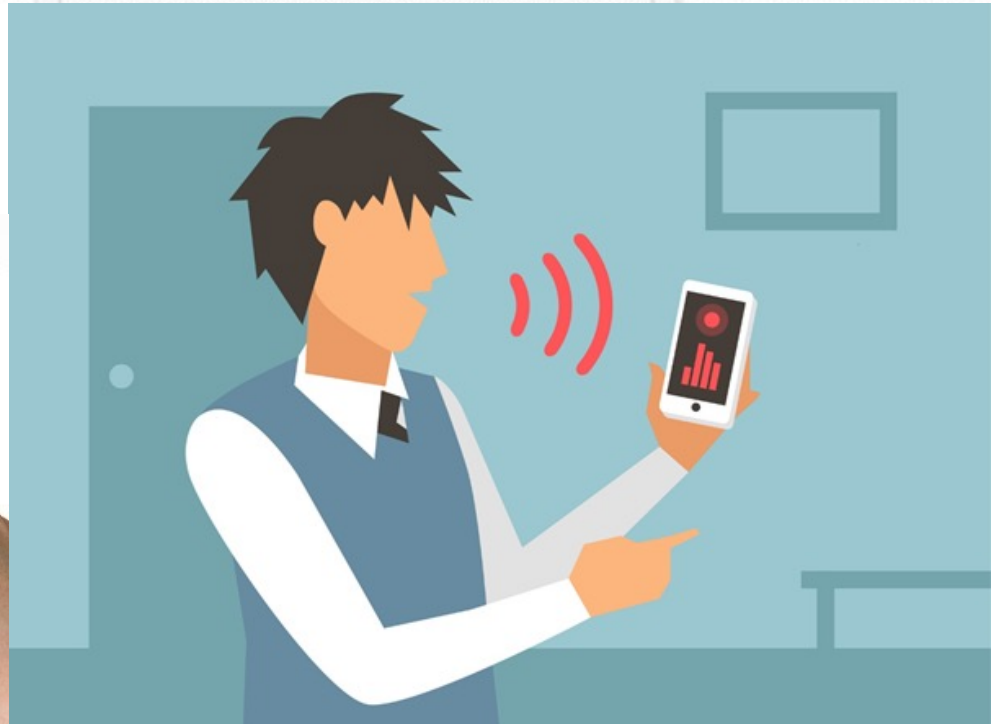
- In a group of 4, use Jakob's general principles to evaluate user interface of the following applications:
  - Headspace
  - Shazam
  - Opentable
  - Lemonade
- Which principles are violated?
- How can you fix them?

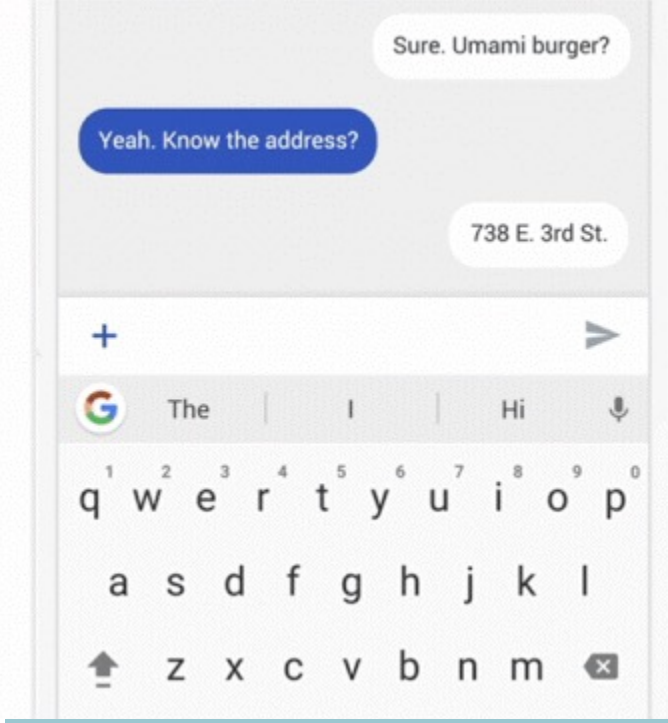
How Do We Do UX/UI?

- Research and analysis
- Wireframing and prototyping
- User testing
- Implementation and evaluation
  - Use the result of user testing to fine-tune app UI.
  - Re-evaluate the new design for same target audiences as well as new audiences.

The Future

- AI and machine learning in design
- Voice user interface (VUI)
- Augmented Reality (AR)
- Virtual Reality (VR)





References

- https://www.nngroup.com/articles/ten-usability-heuristics/
- https://www.netsolutions.com/insights/mobile-ux-design/