

Building User Interfaces

Platform-Specific Design

Mobile Design

Professor Yuhang Zhao

What we will learn today?

- Mobile input
- Microinteractions
- Mobile design patterns

Mobile Input

Mobile Device Capabilities

Mobile devices have unique capabilities, including input and sensing, that expand possibilities for interaction.

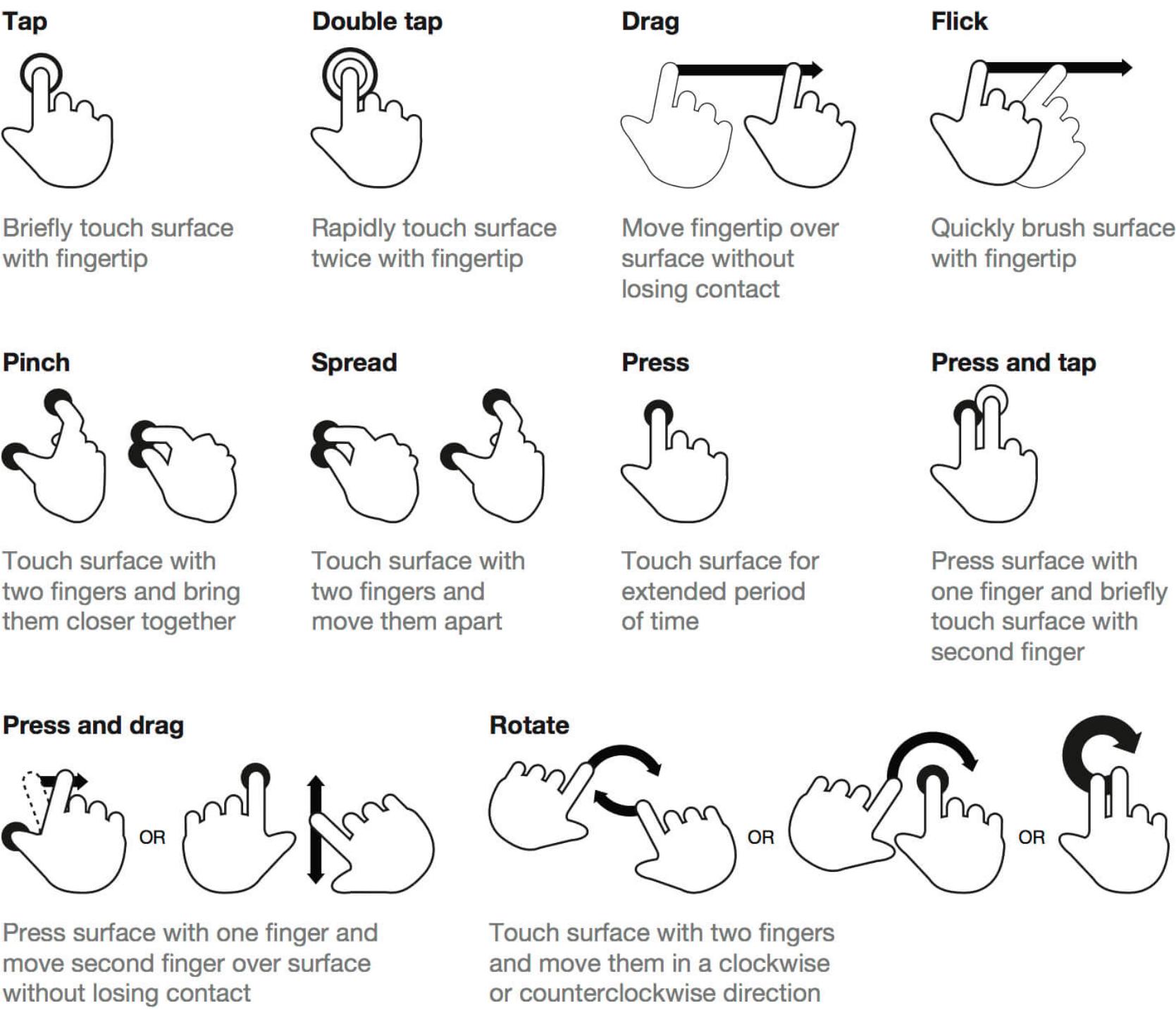
Mobile input primarily centers around the use of touch-sensitive screens that offer two capabilities: *direct manipulation input* and *multi-touch gestures*.

Direct Manipulation Input: In web and desktop interfaces, direct manipulation input involves mouse or trackpad input that is mapped to the screen using a *relative* mapping. On mobile devices, this mapping is *absolute*, and the user directly interacts with screen elements.¹



¹ Image source

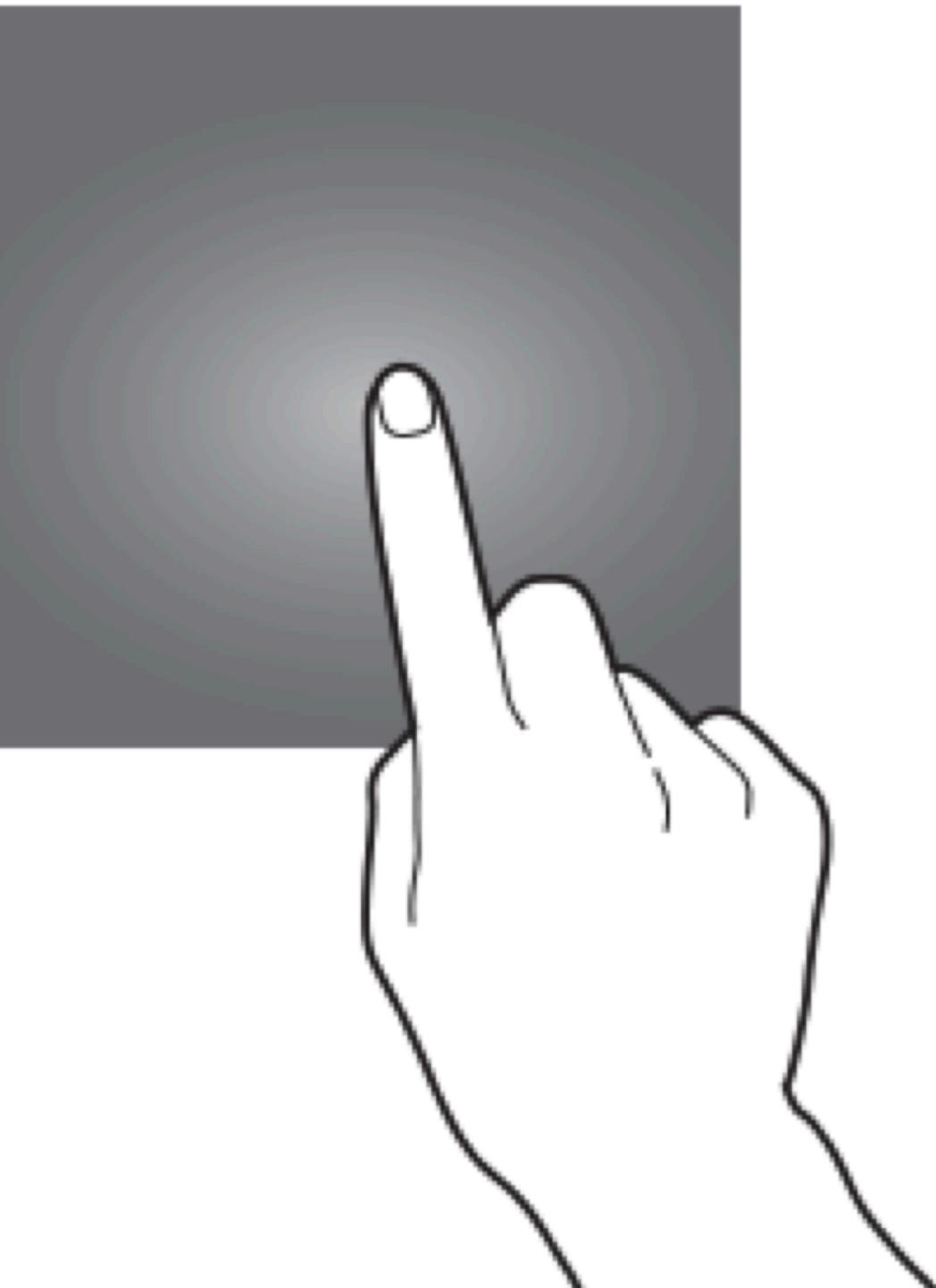
Multi-touch Gestures: Mobile input method include a number of idiomatic gestures dedicated to specific functions.²



² [Image source](#)

Gestures: Tap

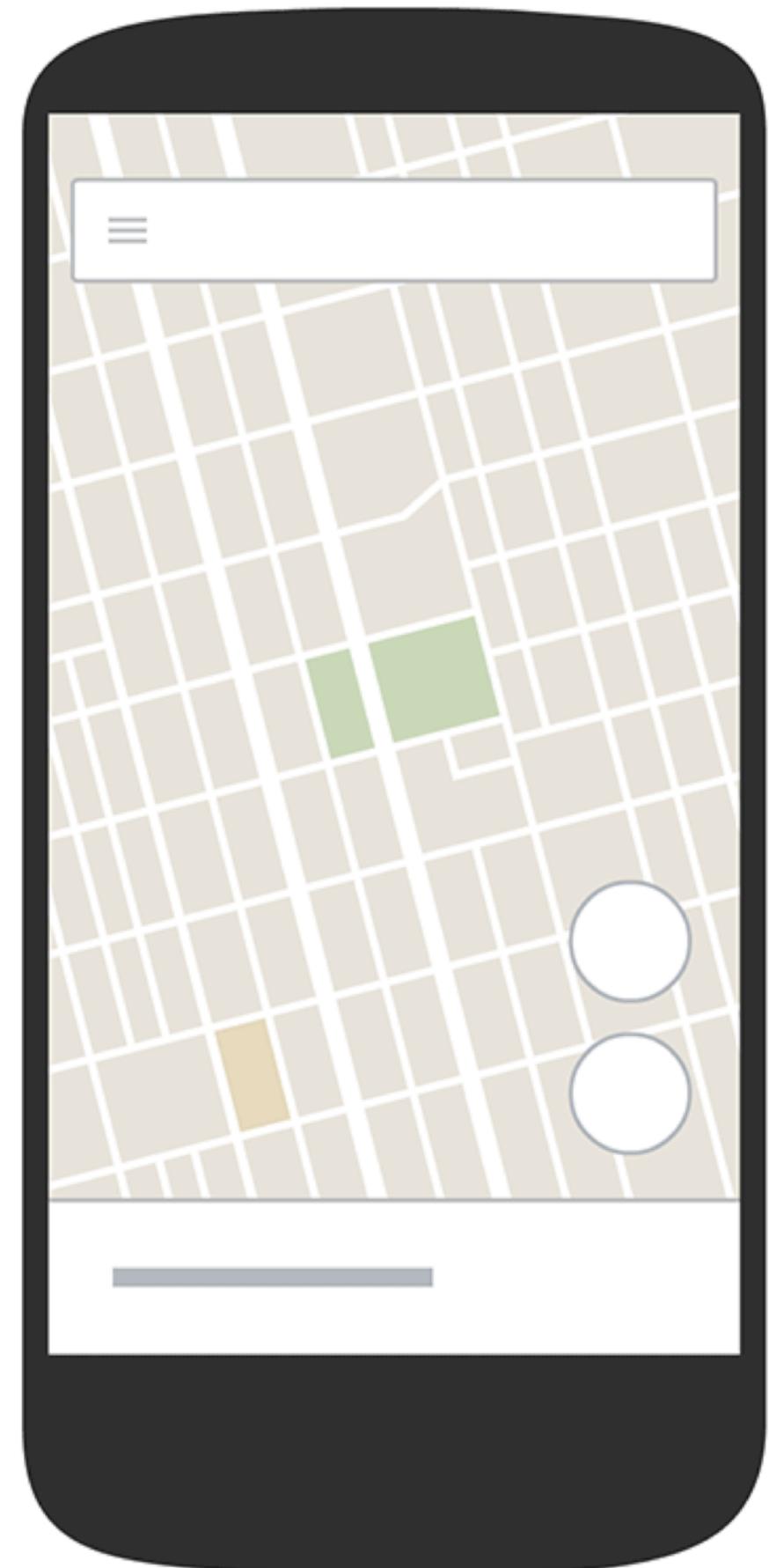
Usage: Maps to a "click" on the desktop computer to select objects or to activate/toggle the state of a control.³



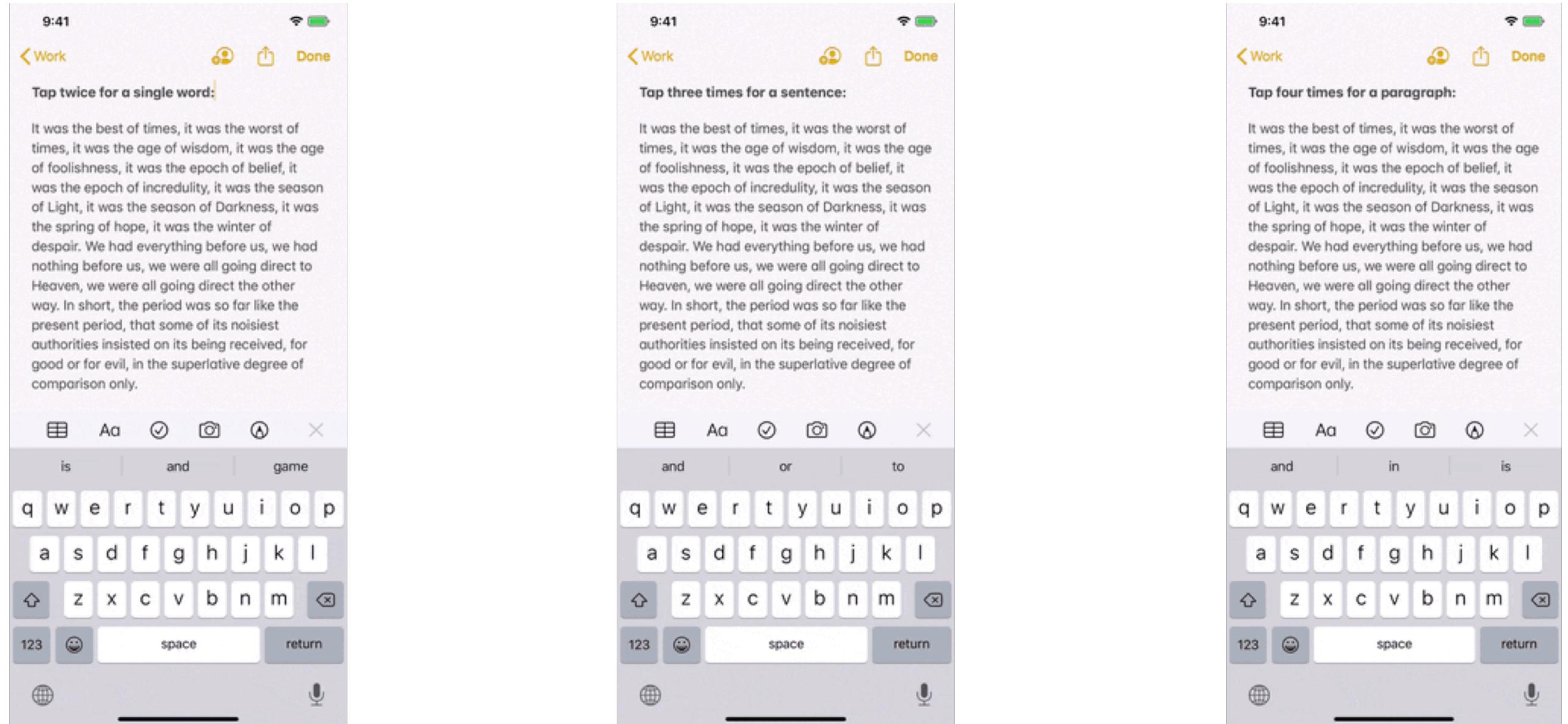
³Image source

Gestures: Double-tap

Usage: Zooming in/out content, selecting items in accessibility mode, or selecting text.⁴



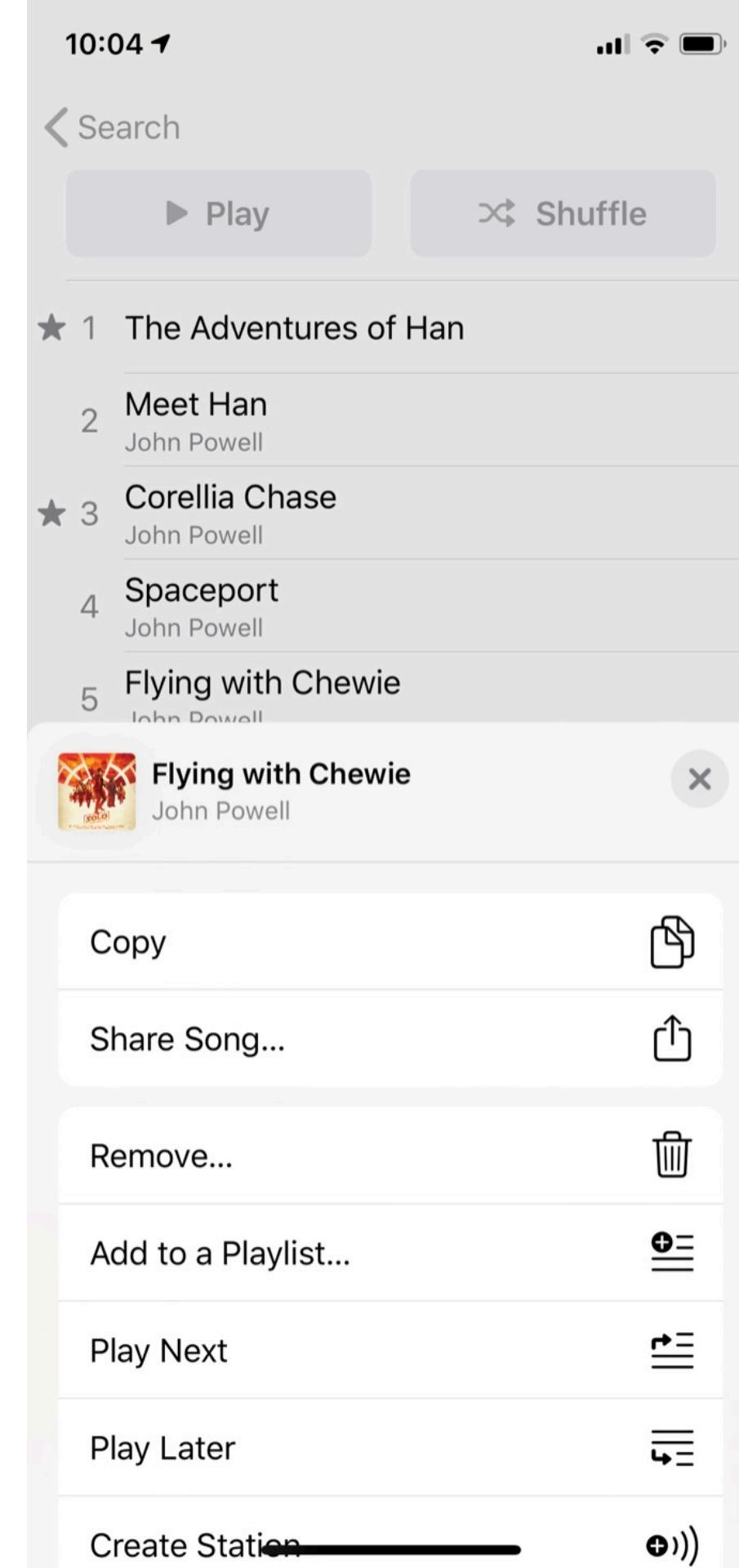
⁴Image source



⁵ Image source

Gestures: Long-press

Usage: Opening contextual menus, previews of content, or enabling editing modes.⁶



⁶[Image source](#)

Gestures: Drag/Swipe

Usage: Used to scroll through content, move objects, or adjust controls. These gestures are the most commonly used gestures on mobile devices.⁷

Drag involves a swipe and hold at the same time.



⁷Image source



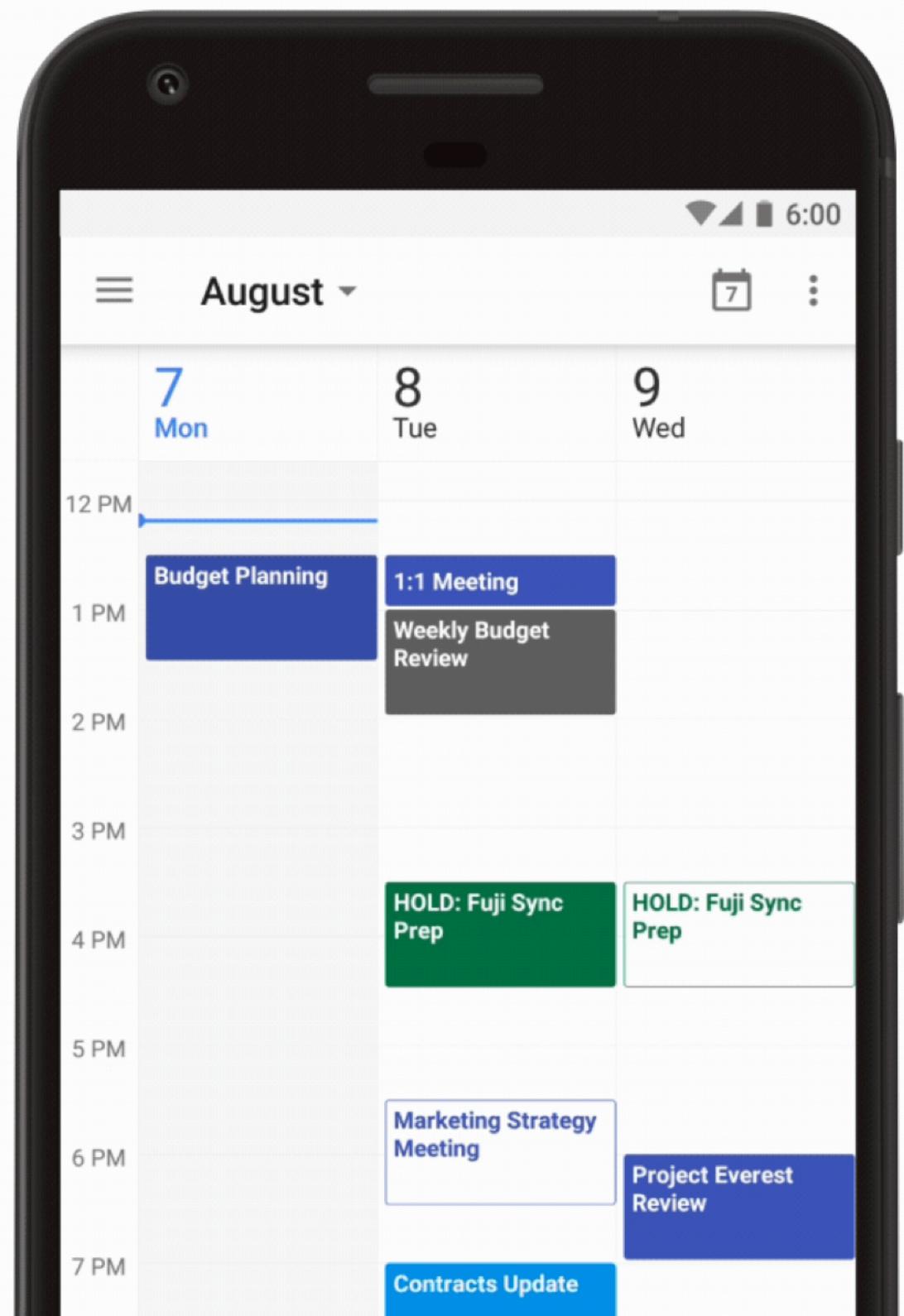
⁸ Image source

In scrolling, swiping vertically scrolls through large amounts of content, and swiping horizontally scrolls through carousels, opens drawers, and navigates to the previous/next screen.⁹



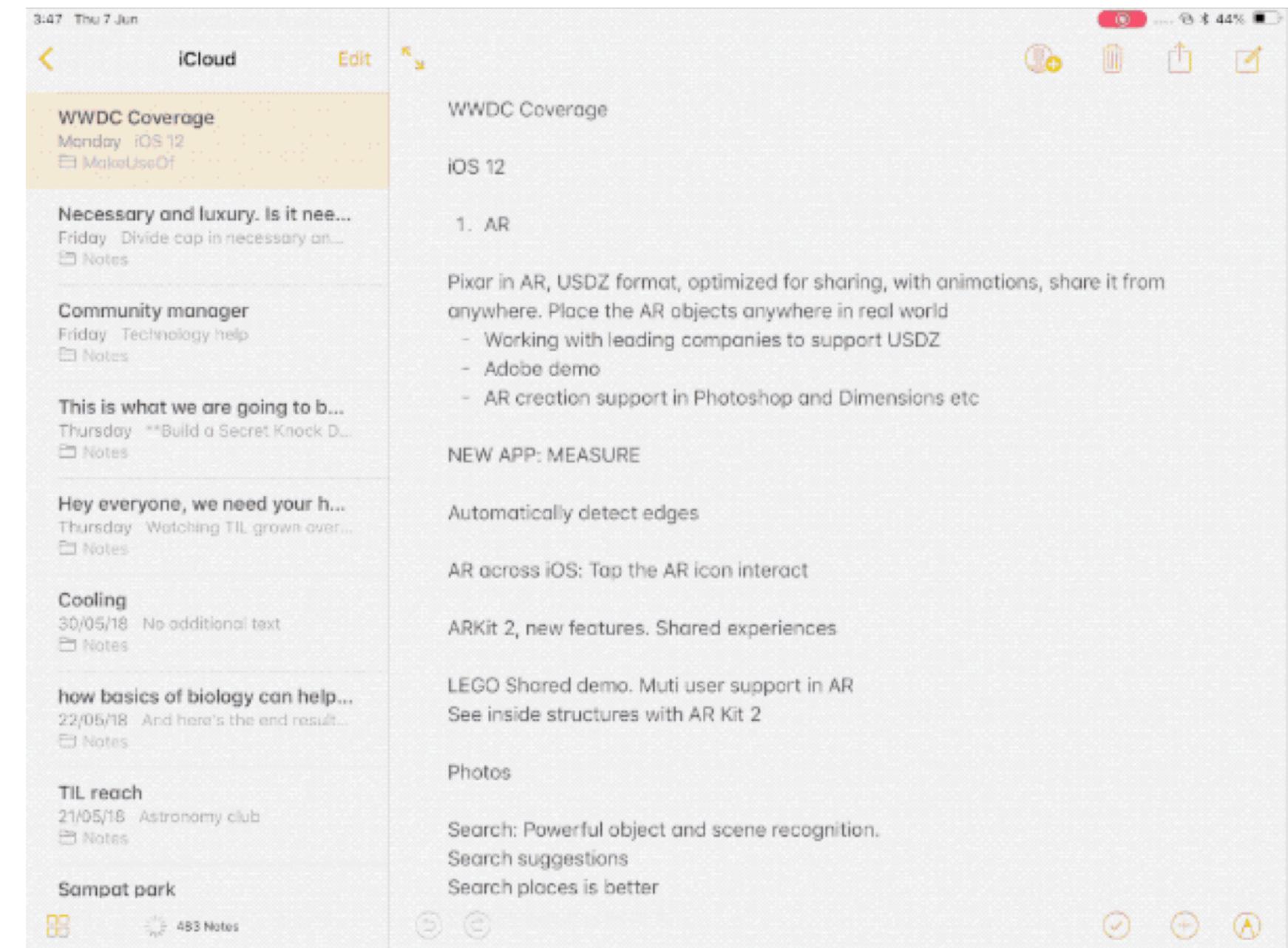
⁹ Image source: [left](#), [right](#)

Dragging can also move objects, such as shapes on a canvas, or operate controls, such as knobs, switches, sliders, and virtual control pads.¹⁰



¹⁰ [Image source](#)

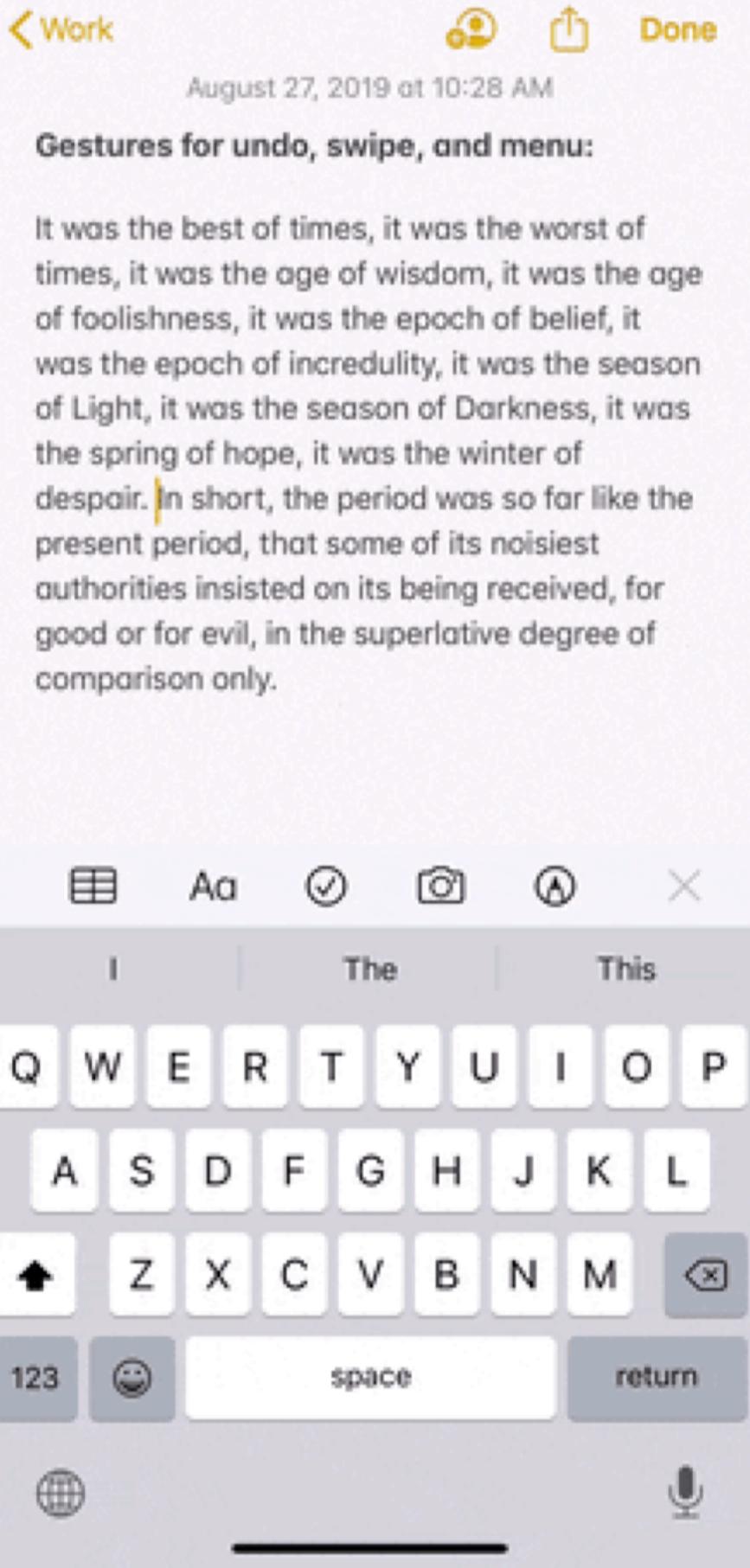
Finally, dragging or swiping with multiple fingers activate OS-level actions, e.g., switching between applications.¹¹



¹¹ [Image source](#)

Gestures: Triple-swipe

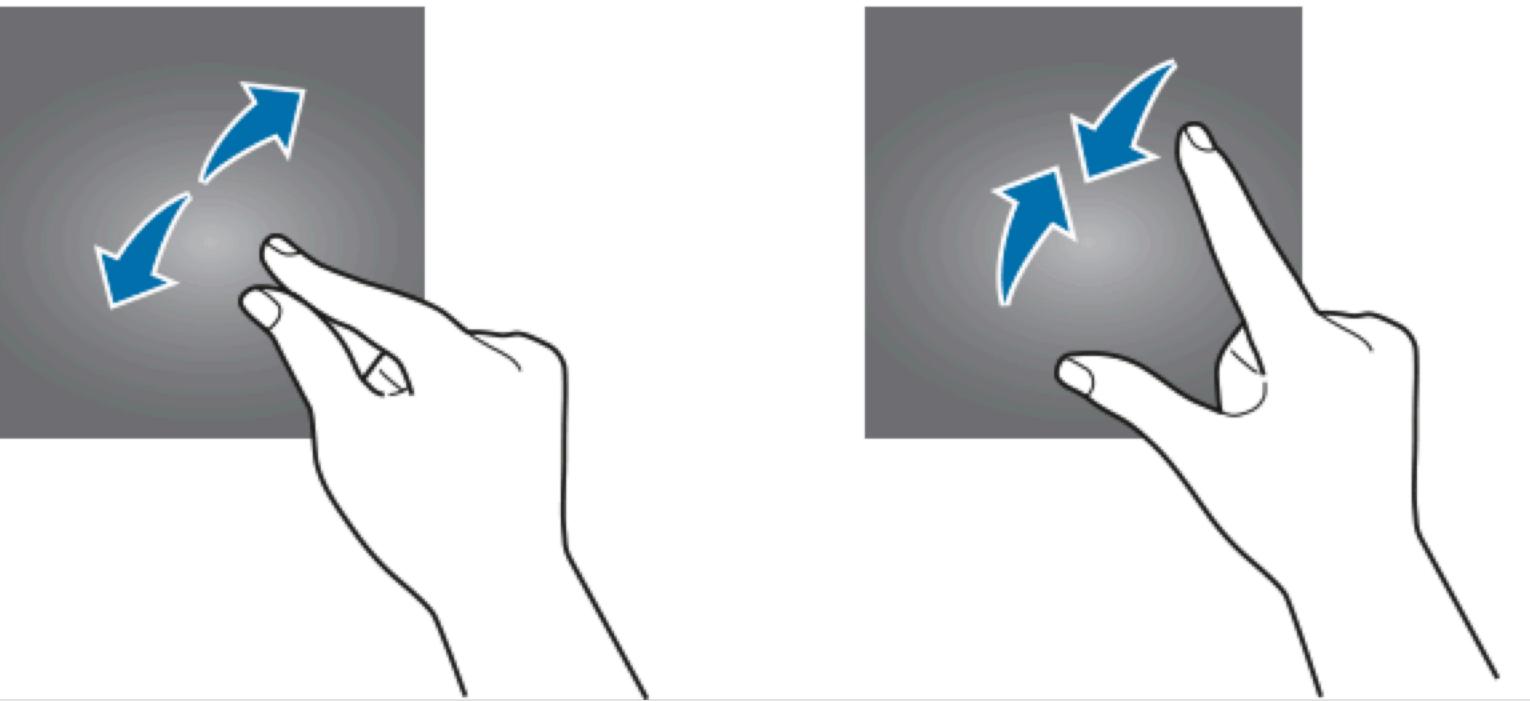
Usage: Mapped to specific OS-level activities, e.g., undo, redo.¹²



¹² [Image source](#)

Gestures: Two-finger Pinch and Spread

Usage: Used to shrink or expand visual elements, e.g., changing the scale of a map.¹³



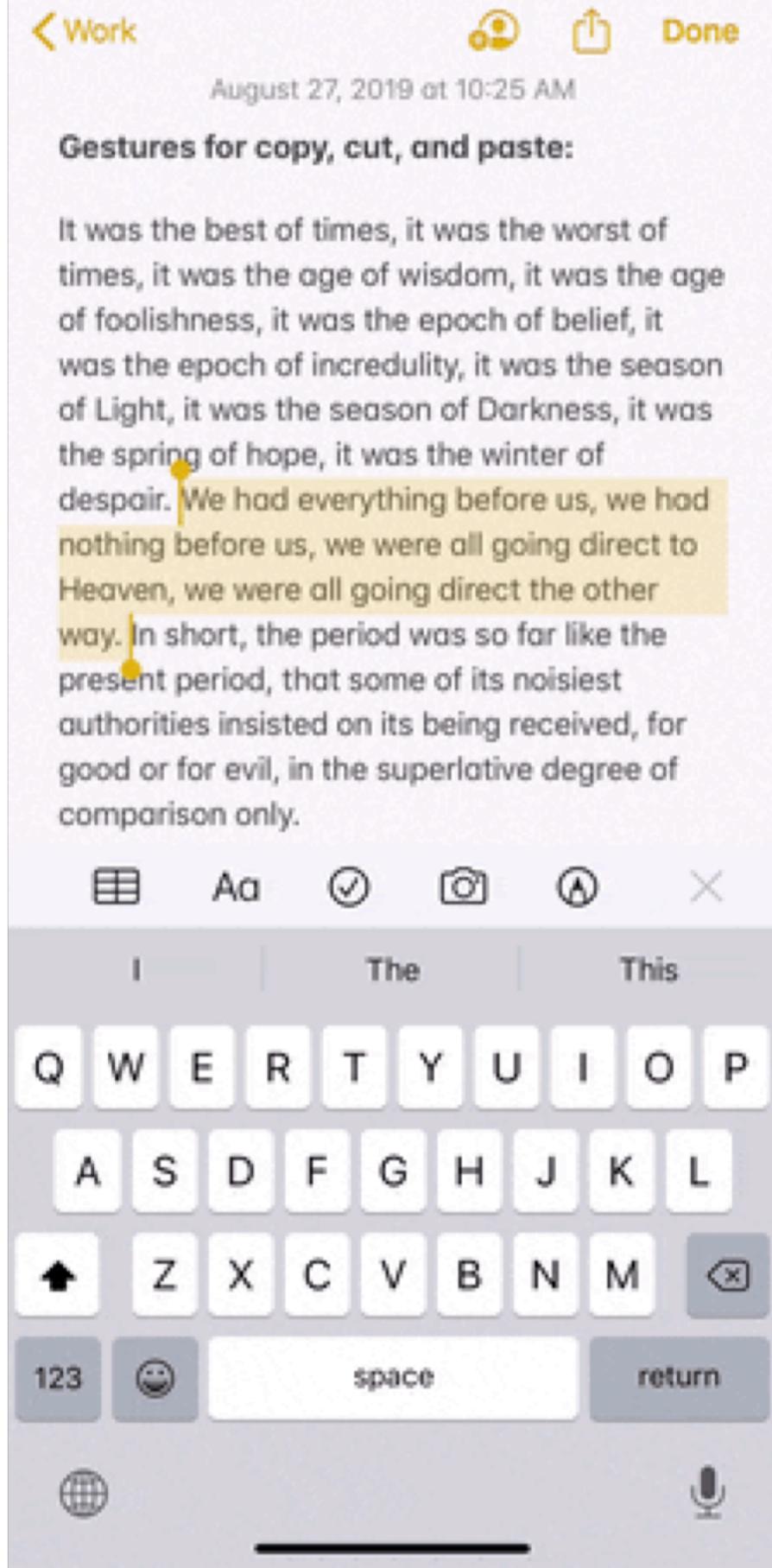
¹³ [Image source](#)



¹⁴ Image source

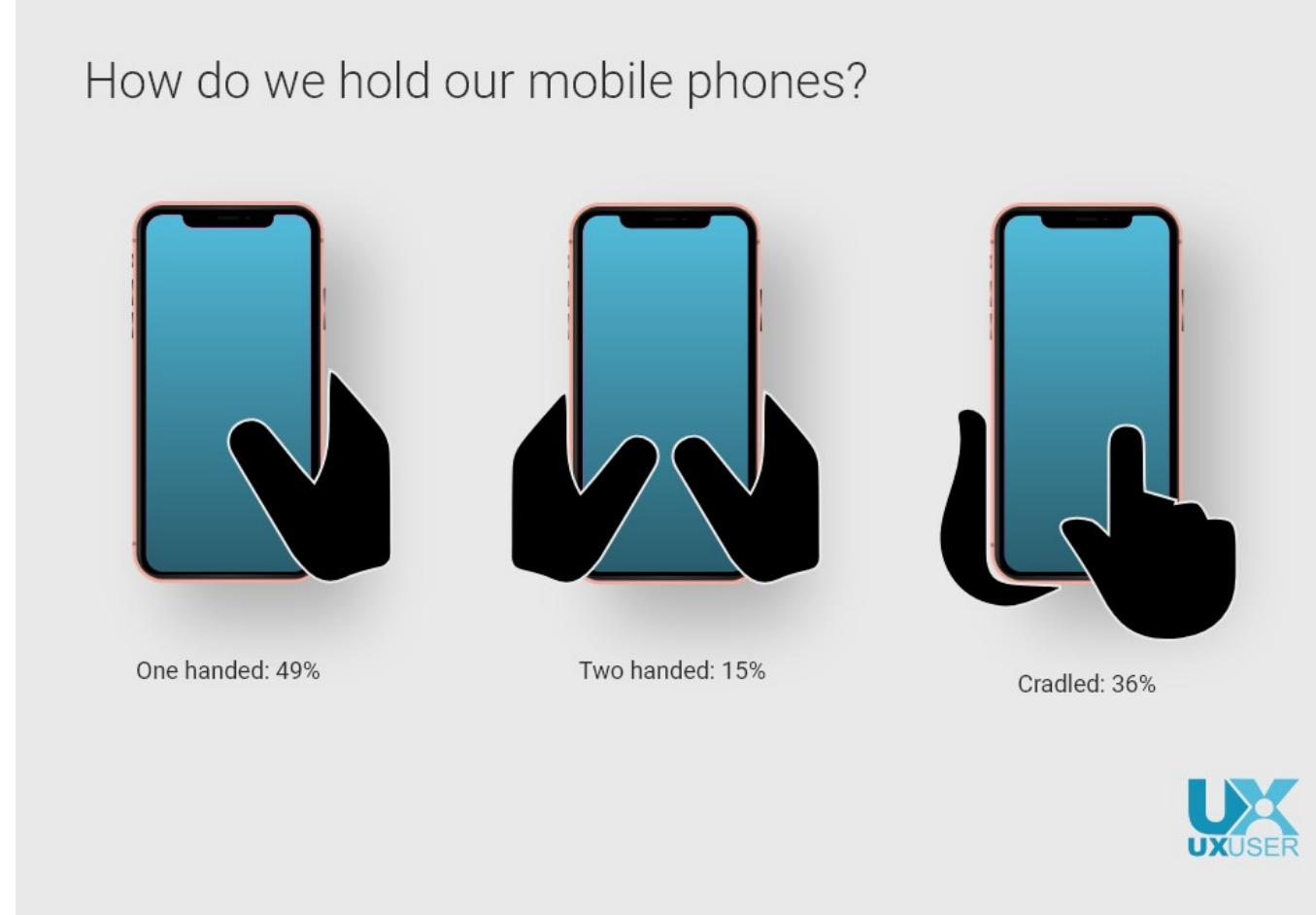
Gestures: Three-finger Pinch and Spread

Usage: Mapped to OS-level actions, such as copy, cut, and paste.¹⁵



¹⁵ [Image source](#)

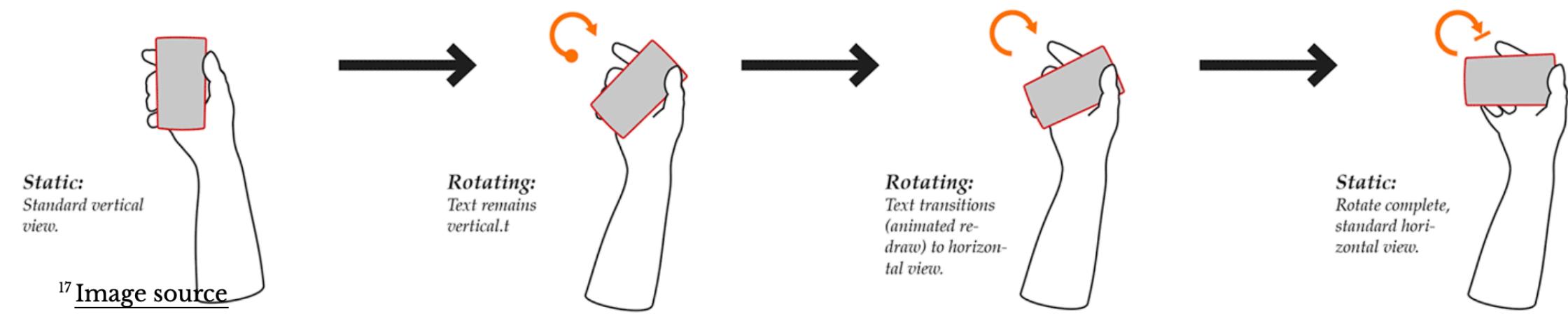
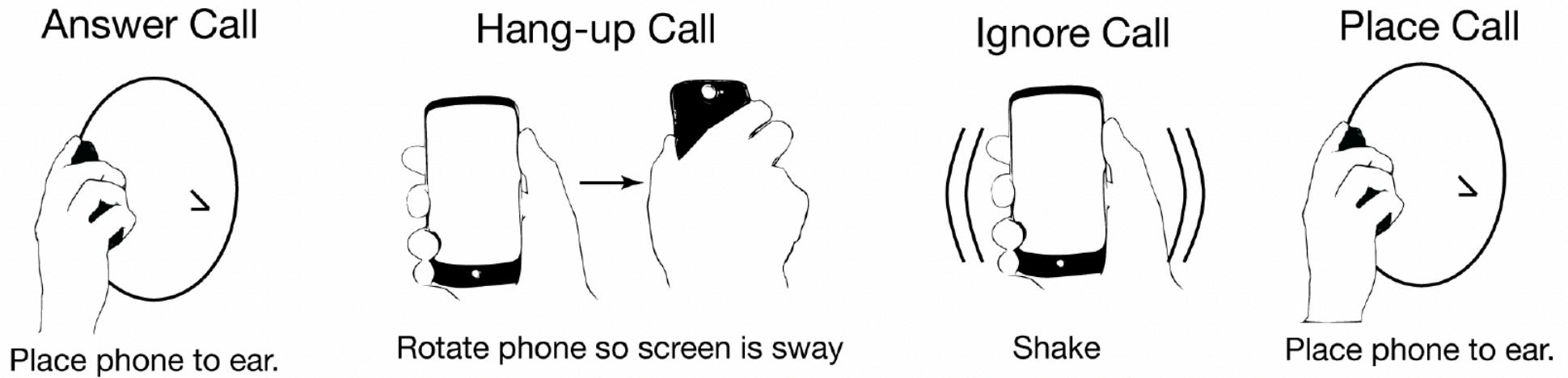
Pro Tip: In choosing which gestures to use, consider the context of use, e.g., how people hold their devices, how large the device is, what we are expecting to do.¹⁶



¹⁶ [Image source](#)

Motion Gestures

Definition: Gestures that involve moving the mobile devide in specific ways, e.g., shaking to enable/disable silent mode. These gestures are usually application specific or customizable at the OS level.



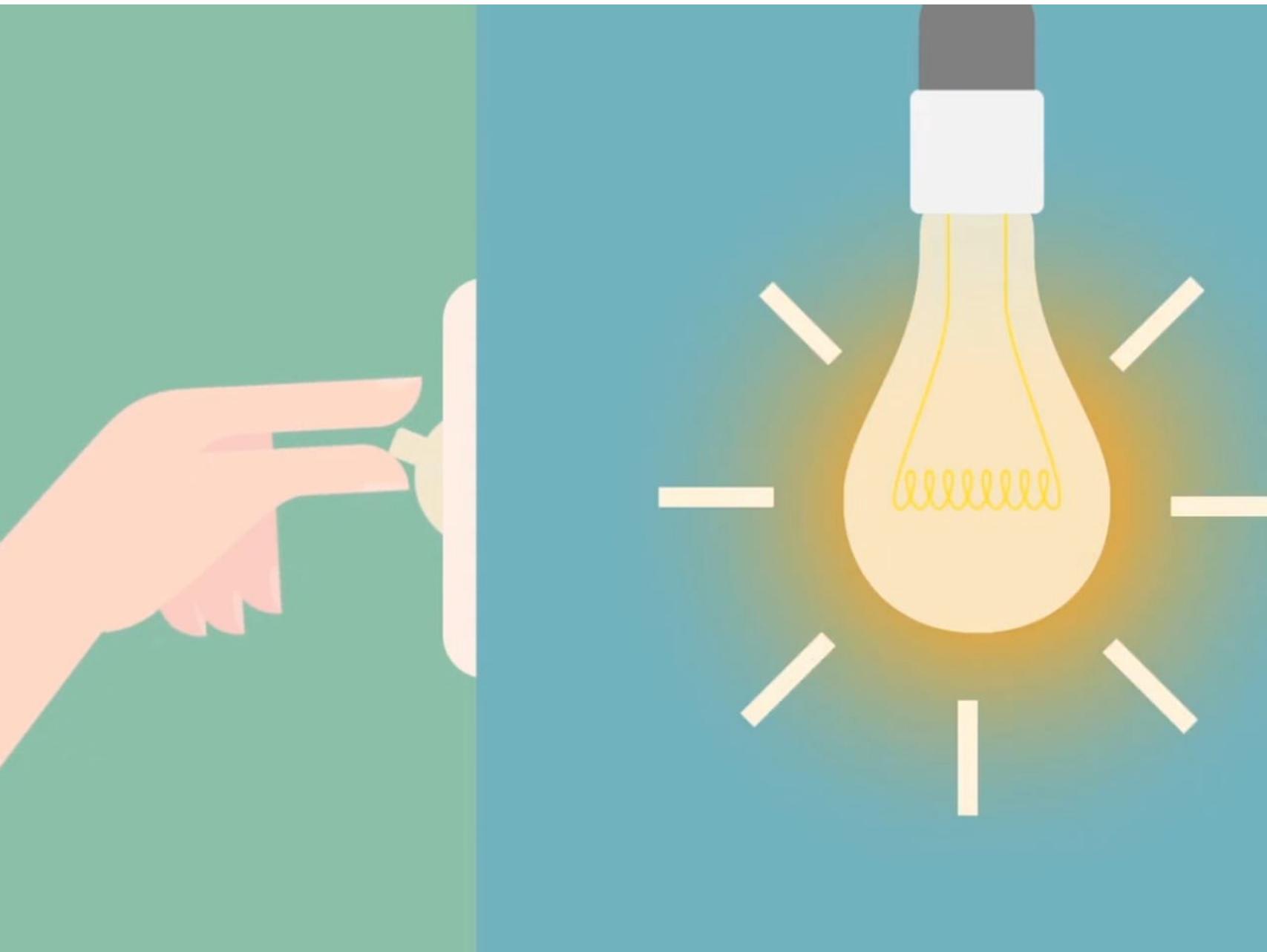
¹⁷ [Image source](#), Ruiz et al. 2011.

Microinteractions

What are *microinteractions*?

Definition: *Microinteractions* are contained product moments that revolve around a single use case.¹⁸

E.g., Facebook like button, pull-to-refresh action, flicking a notification to dismiss it, etc.



¹⁸ Dan Saffer, 2014, Microinteractions
Image source: Nick Babich

How microinteractions work²⁰

Each micro interaction has four parts:



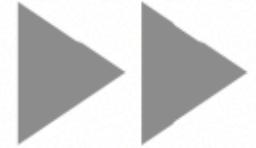
TRIGGER



RULES



FEEDBACK



LOOPS & MODES

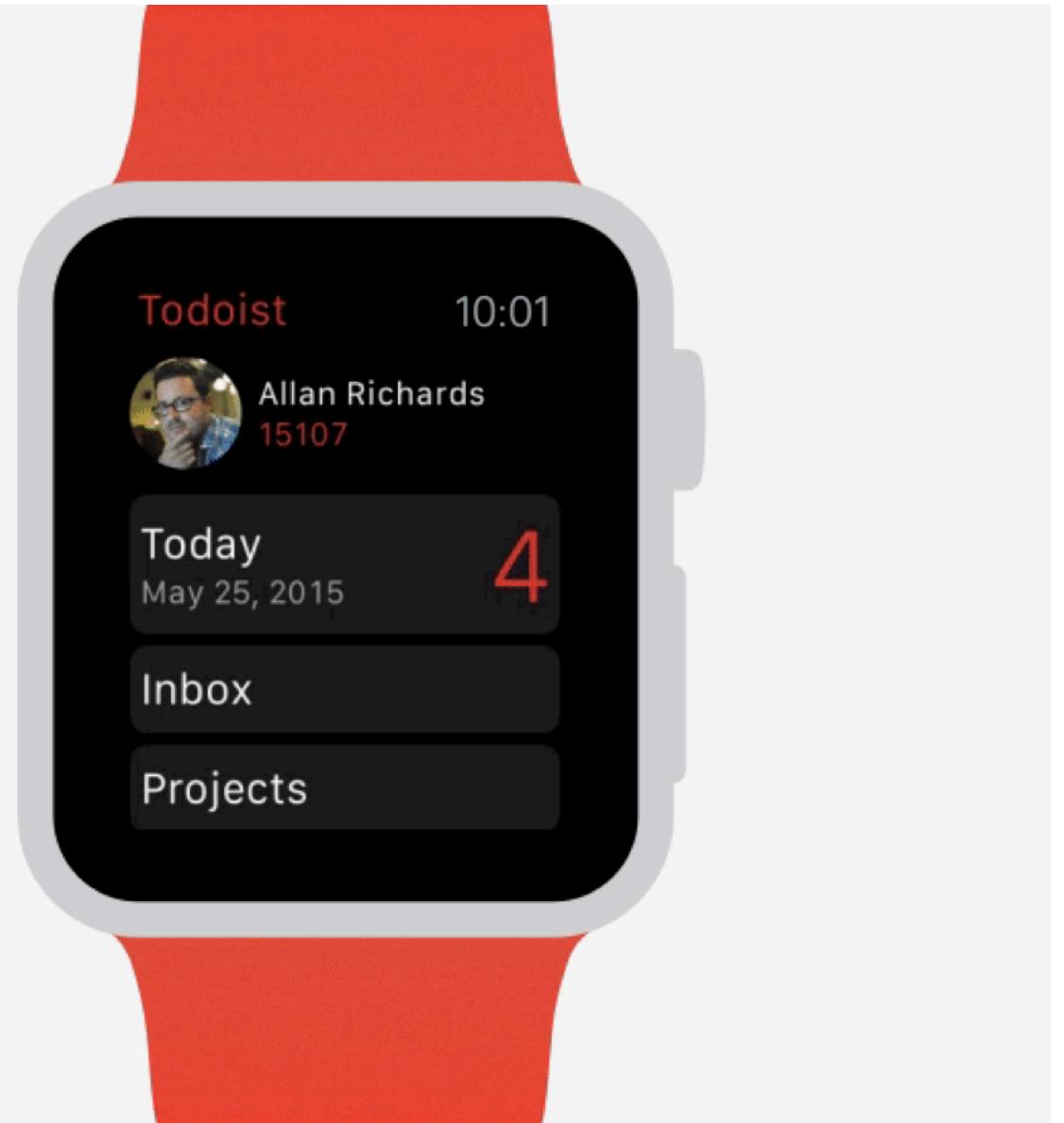
²⁰ [Image source](#)

Trigger

Definition: Events that initiate the microinteractions. Triggers can be *manual/user-initiated* or *automatic/system-initiated*.

Manual triggers involve intentional and explicit interaction with the system, e.g., by flipping a switch, pressing a button, speaking to the system.

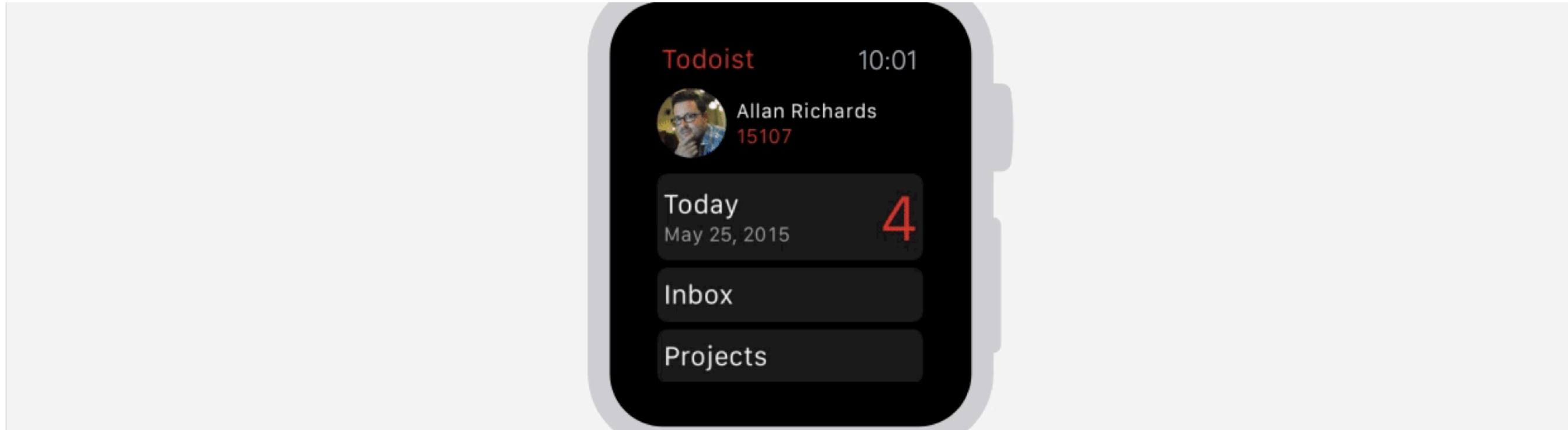
Automatic triggers occur when certain conditions are met, e.g., chime when a new text message arrives, silence the phone after 10 pm.



²¹ Image source: [left](#), [right](#)

Rules

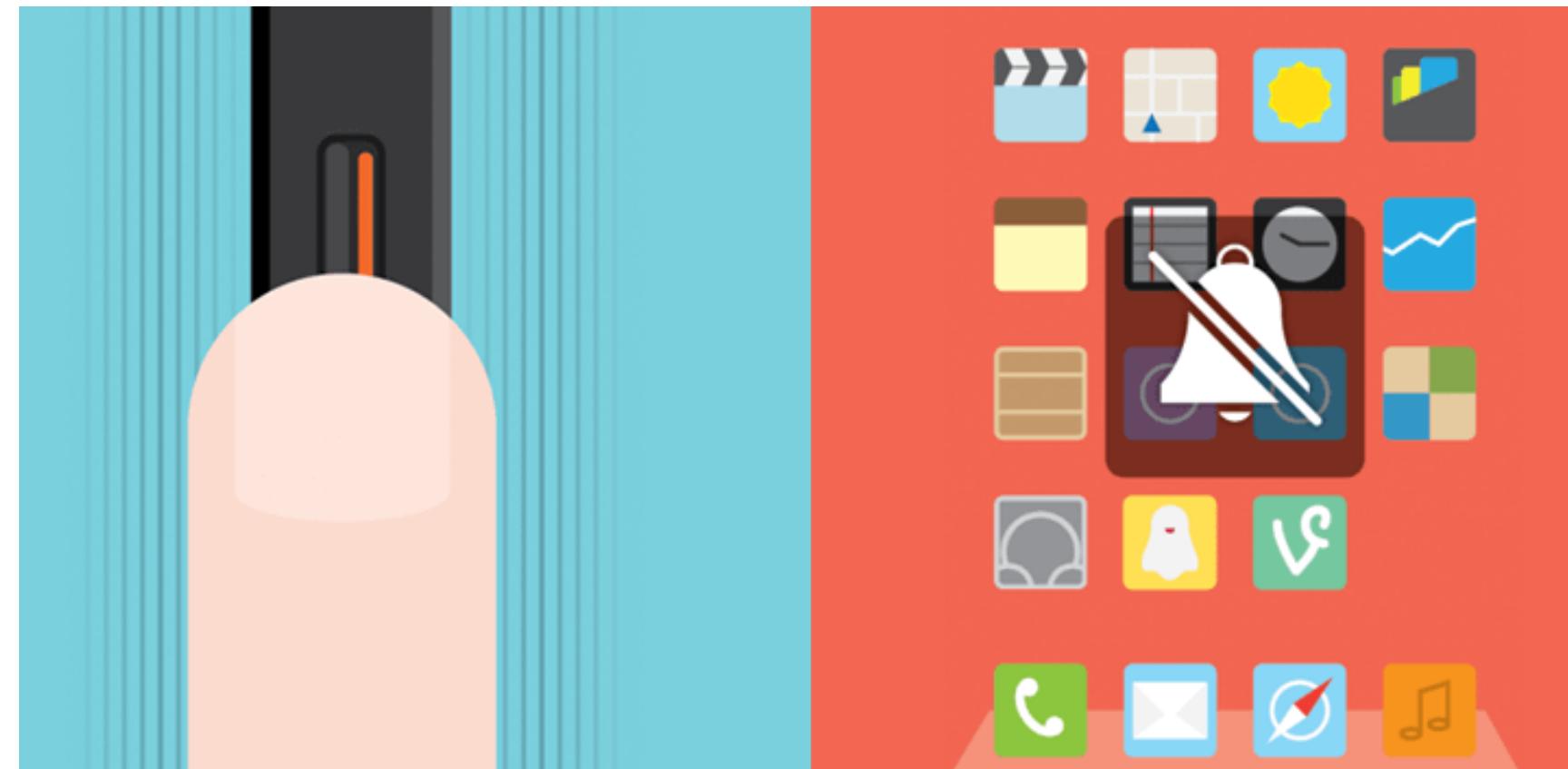
Definition: Rules determine what happens (and doesn't happen) in the system when a microinteraction is triggered.²²



²² [Image source](#)

Feedback

Definition: Information that the user sees, hears, or feels to reflect what is happening. Feedback can take many forms: visual, aural, haptic, etc.²³



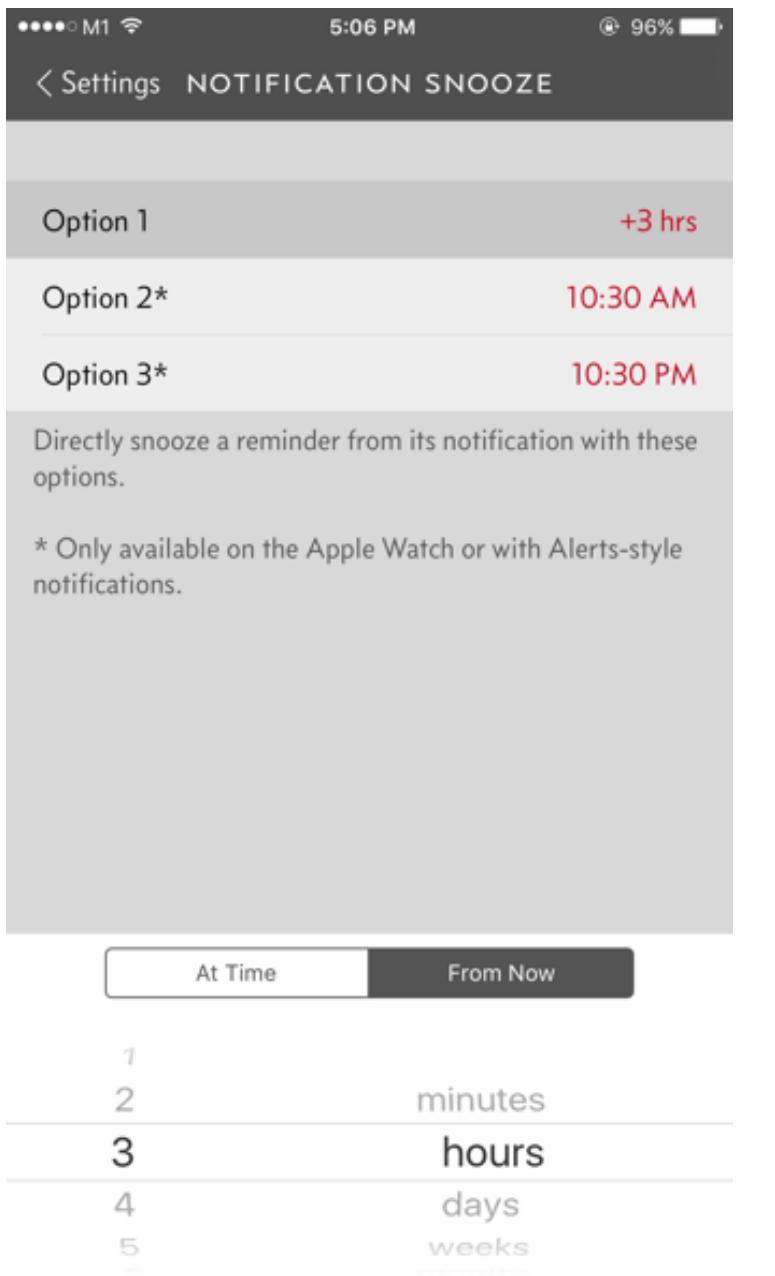
²³ [Image source](#)

Loops & Modes

Definition: Meta-rules that, depending on *context*, change microinteraction rules and feedback (e.g., "snoozing" a reminder; chime/vibration feedback when silent mode is off/on).

Loops determine the length of the micro interaction and whether the interaction repeats (e.g., related beeping when you leave the fridge door open) or changes over time (e.g., microwave oven reminder to pick up food changing over time).

Modes switch the functioning or operation of the system. E.g., "do not disturb" mode that changes system behavior.



²⁴ Image source

Pro Tips: Minimize the scope of the microinteraction and make sure that it doesn't turn into a feature. Best microinteractions support single task, single action.

Simpler way of thinking about microinteractions is *action-feedback pairs for a single purpose*.

In-class Activity

Mobile Microinteractions

On your mobile device or tablet, identify a microinteraction and describe its four elements.

[Assignment template](#)

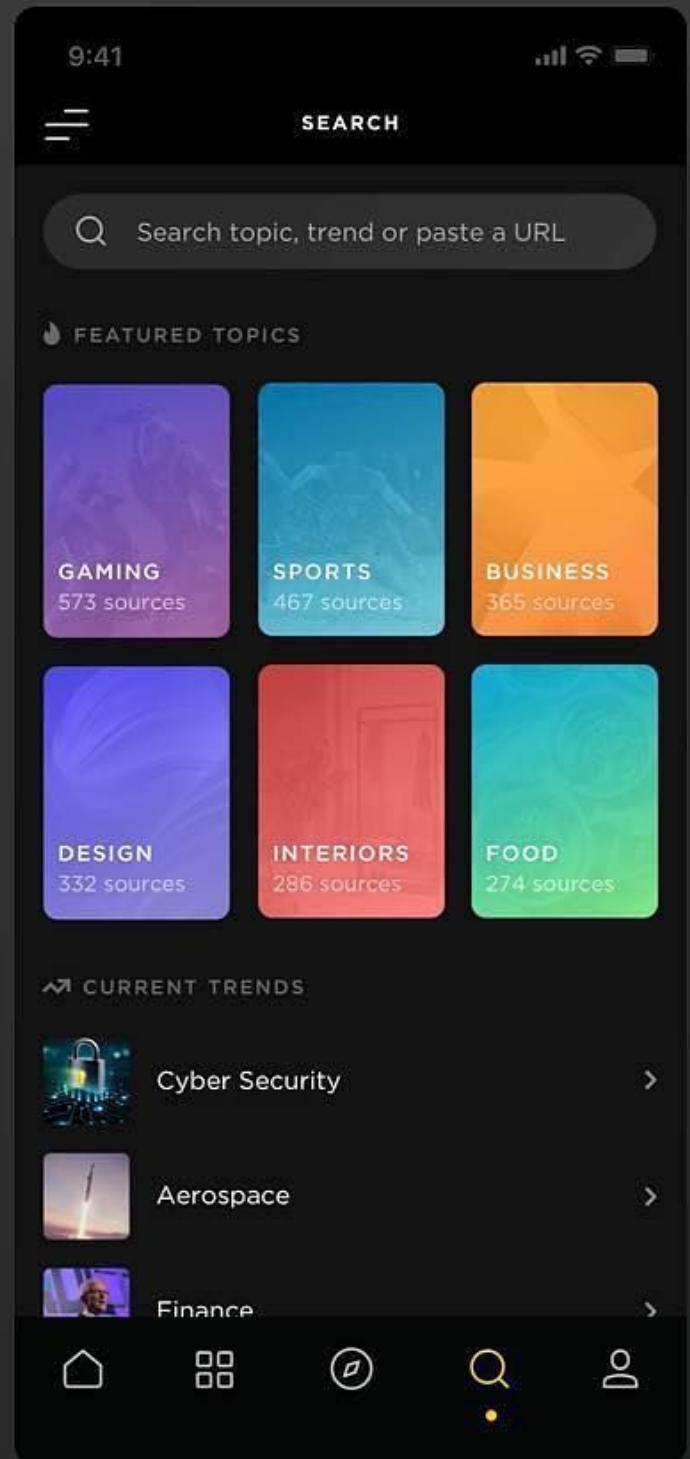
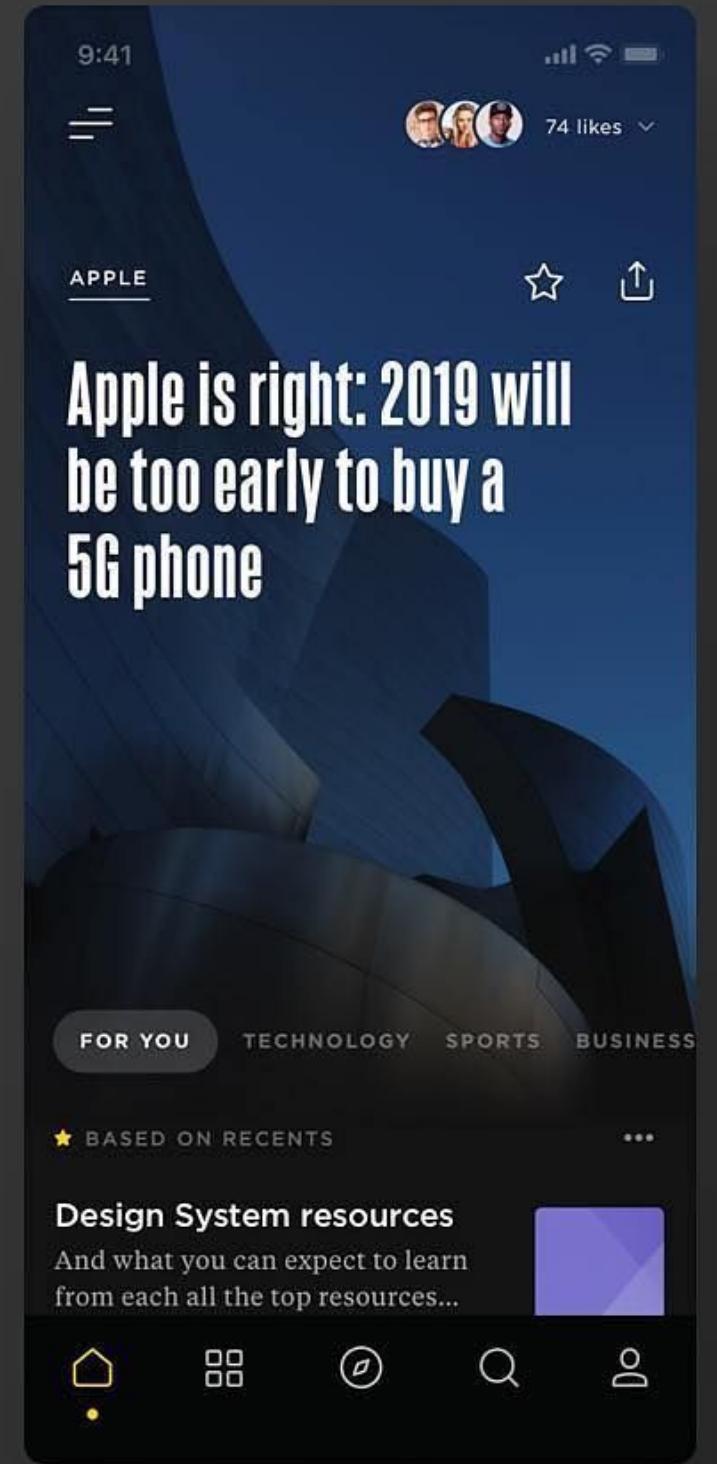
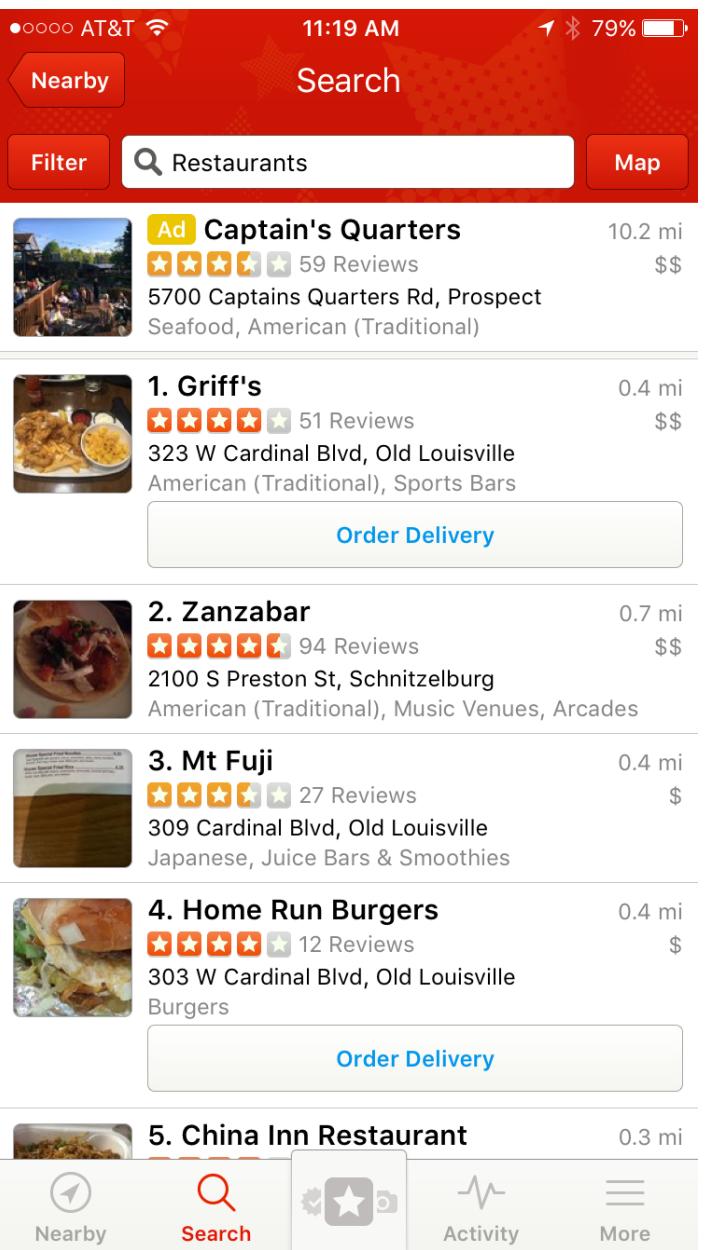
Mobile Design Patterns

Why mobile design patterns?

Mobile platforms are highly constrained design environments.
Mobile design patterns help designers overcome these limitations by
expanding capabilities for input, display, and navigation.

Patterns: Stacks

Usage: Used to vertically organize design elements such as a toolbar, content panes, and a navigation bar to maximally utilize the vertical space in mobile devices.



²⁵ Image source: [left](#), [right](#)

Patterns: Screen carousels

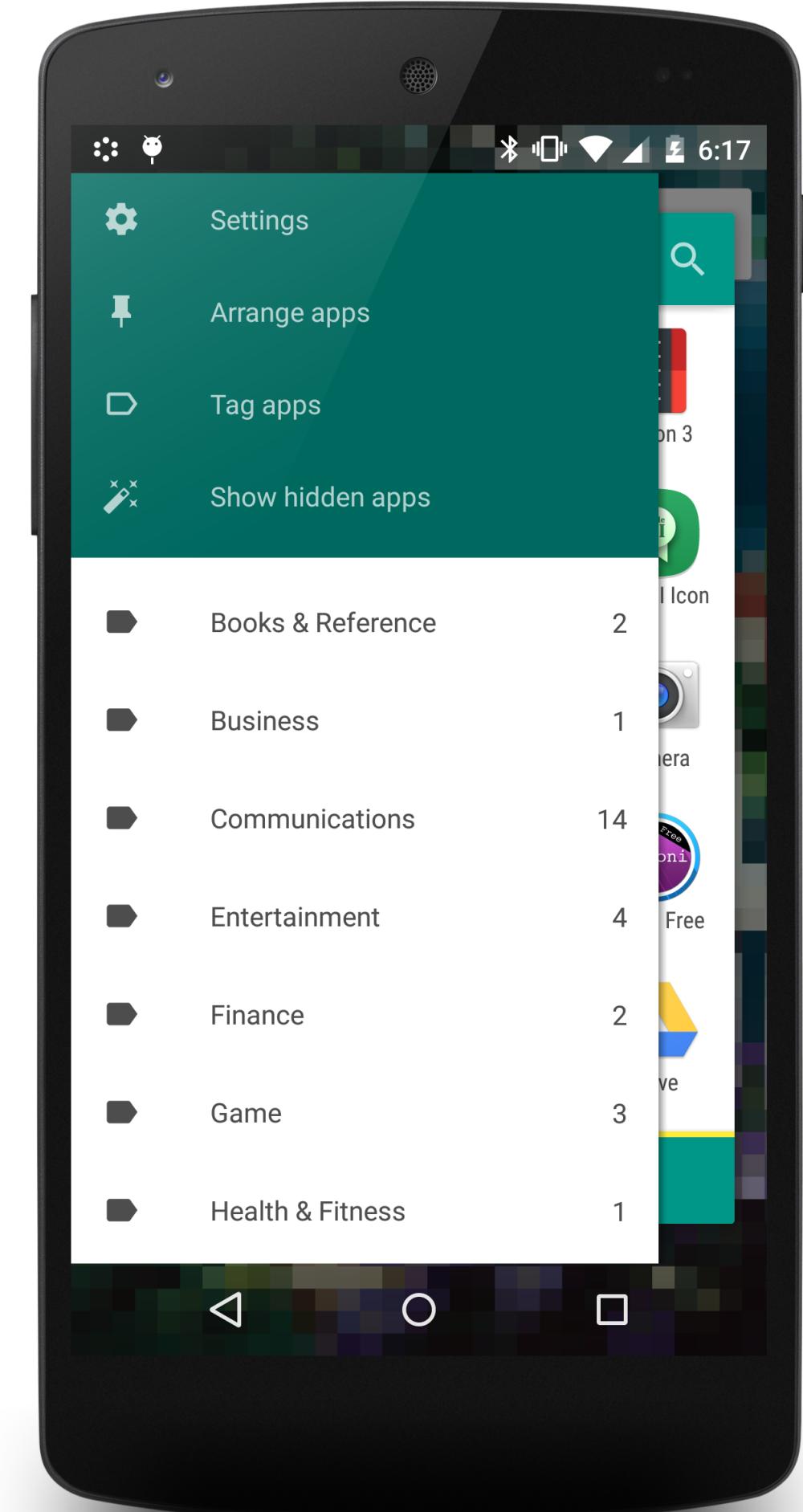
Usage: Full-screen content panes that can be placed on a horizontal array to display different instances of the same kind of information, such as weather information for different cities.²⁶



²⁶ [Image source](#)

Patterns: Drawers

Usage: Drawers provide links for navigation or controls for the various settings of the application.²⁷



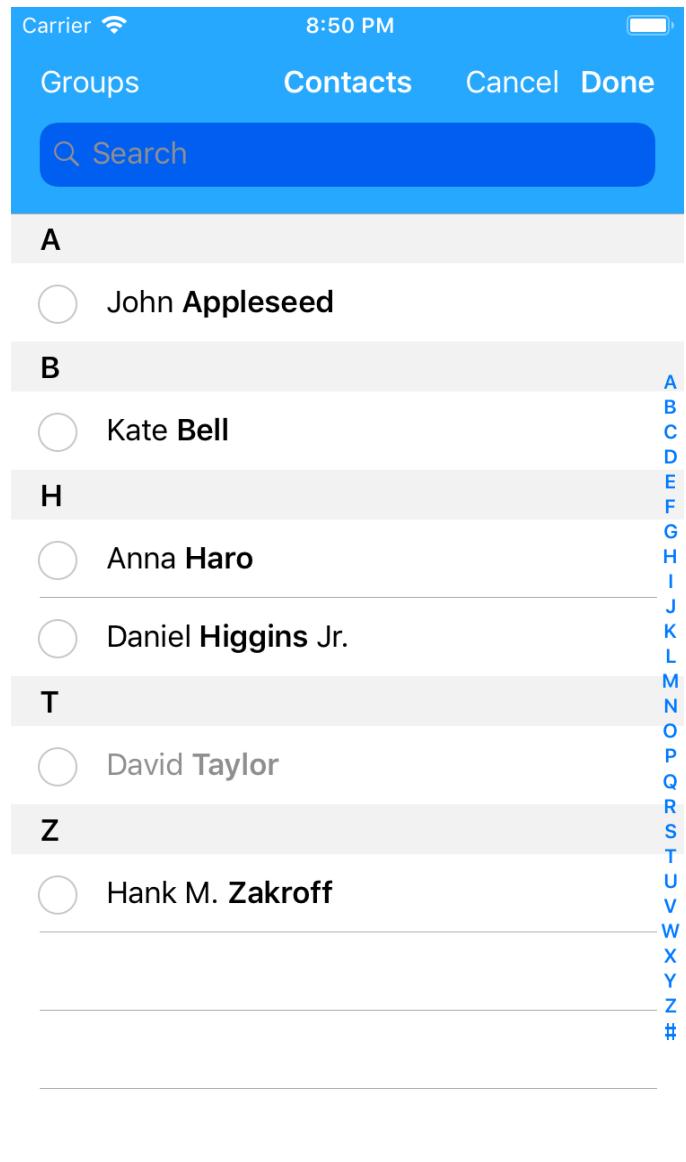
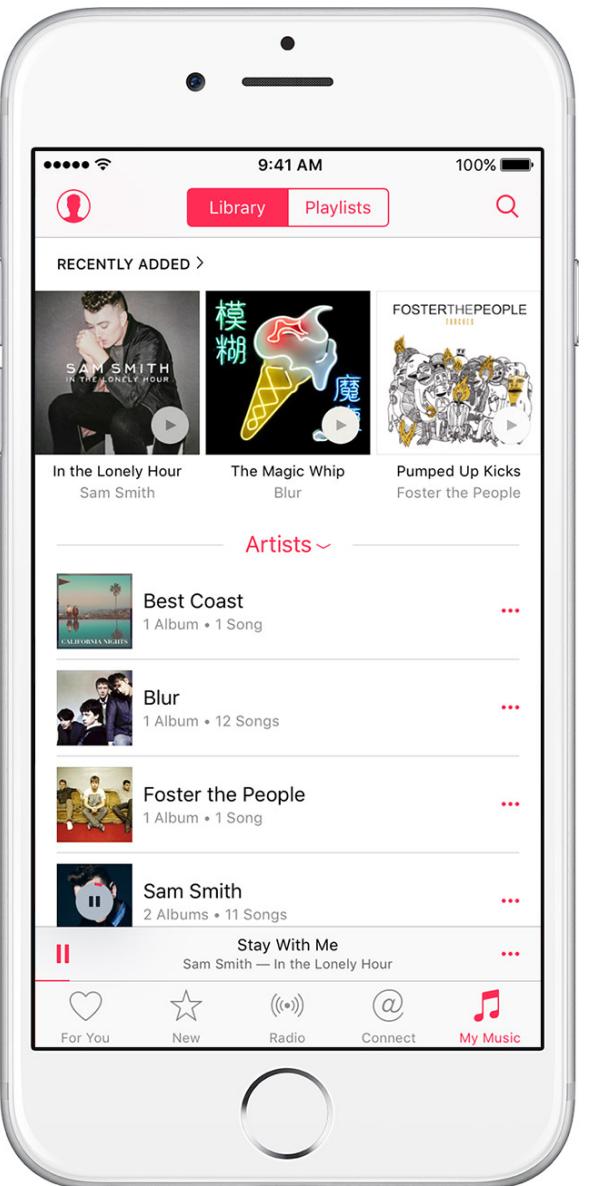
²⁷[Image source](#)

Patterns: Lists & Grids

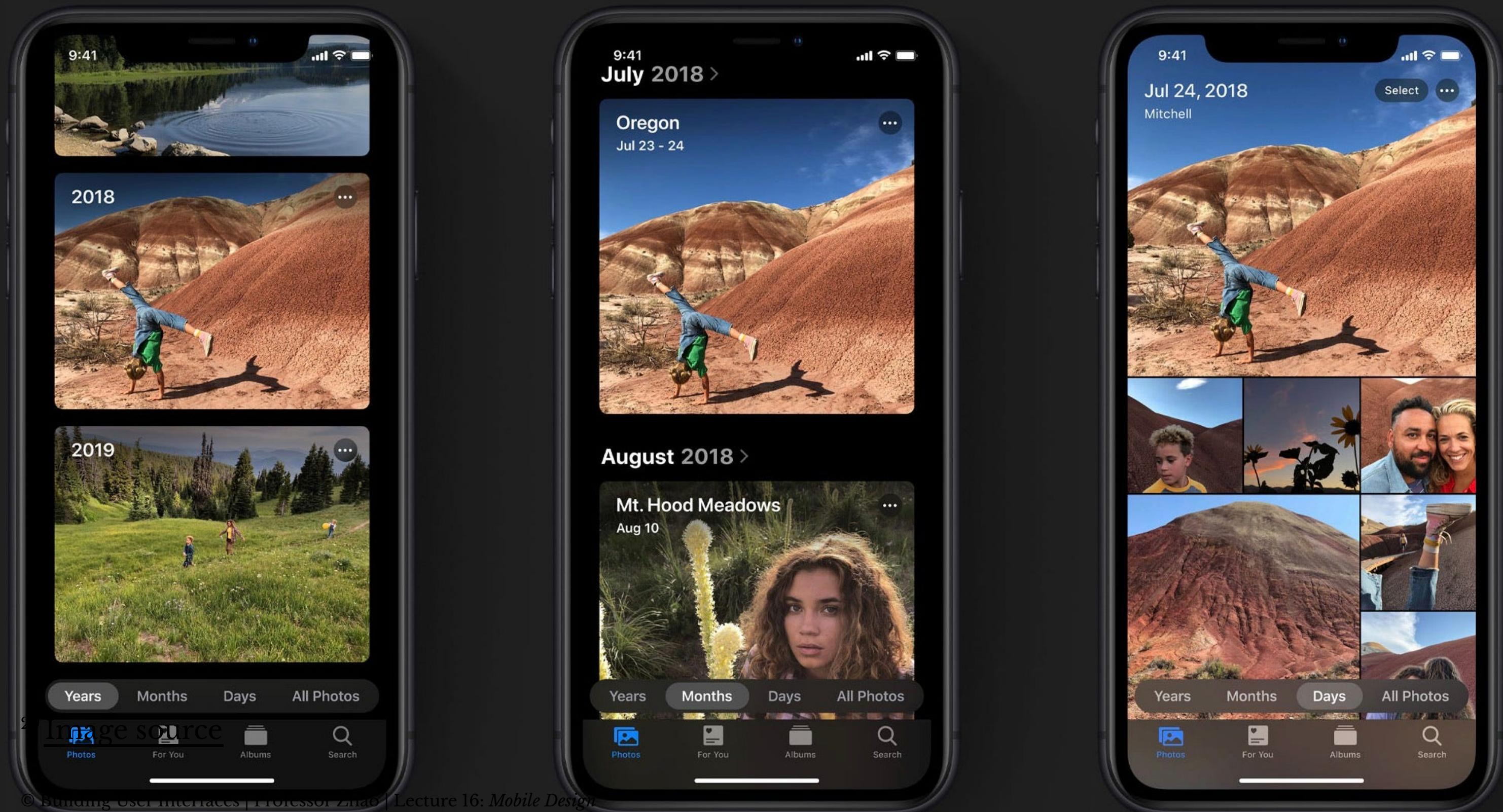
Usage: Lists involve vertically stacking a large number of items, including text, controls, and thumbnails, and supporting navigation through vertical scrolling.

Grids involve a large continuous grid or multiple panes of grids that users can scroll through vertically or horizontally.

Grids are more commonly used for information with more visual content, such as media thumbnails, icons, and photos.

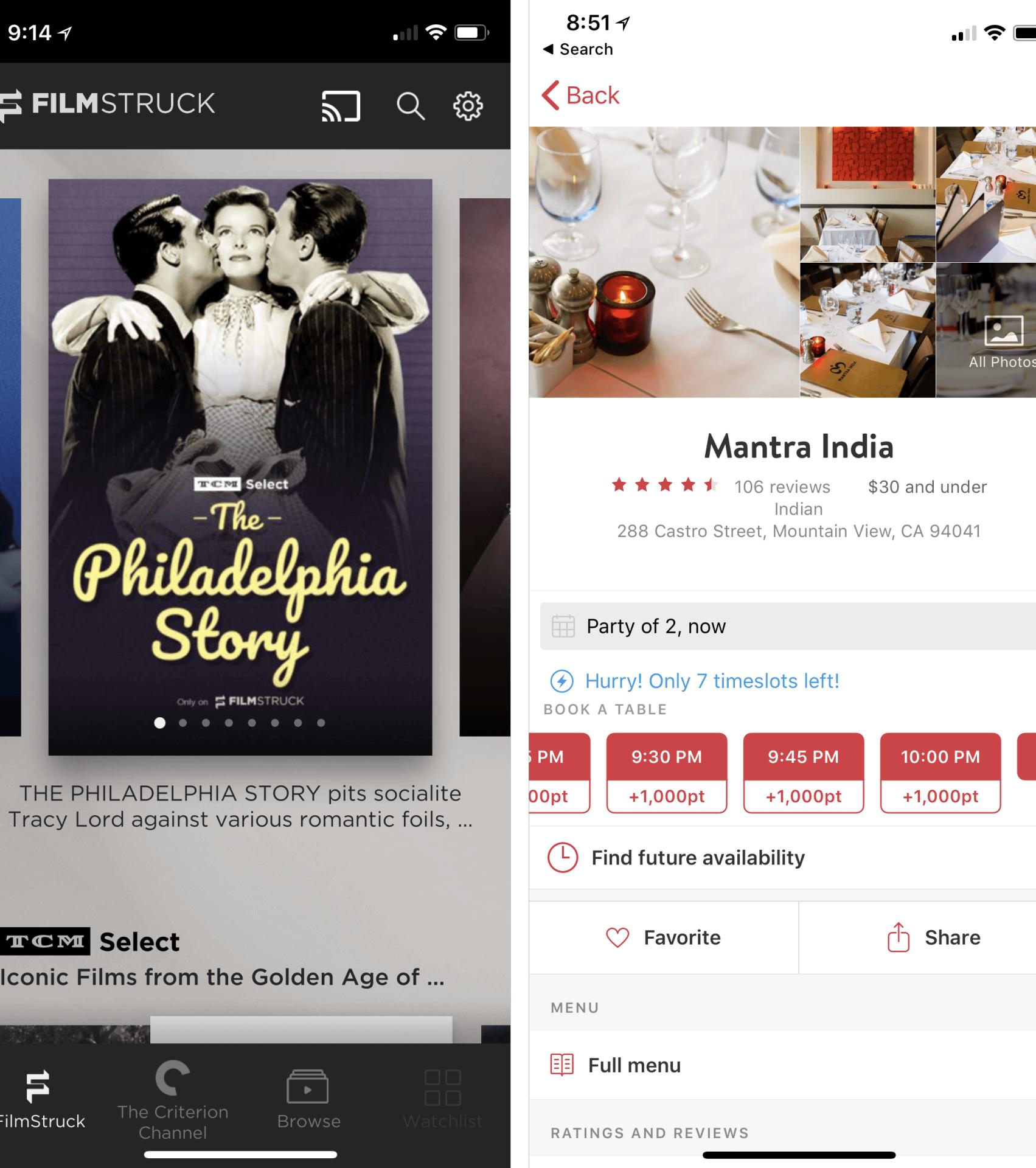


²⁸ Image source: [left](#), [right](#)



Patterns: Carousels

Usage: Content carousels provide a row of content items including images or textual cards that users can navigate through by swiping left and right.³⁰



³⁰ [Image source](#)

Patterns: Swimlanes

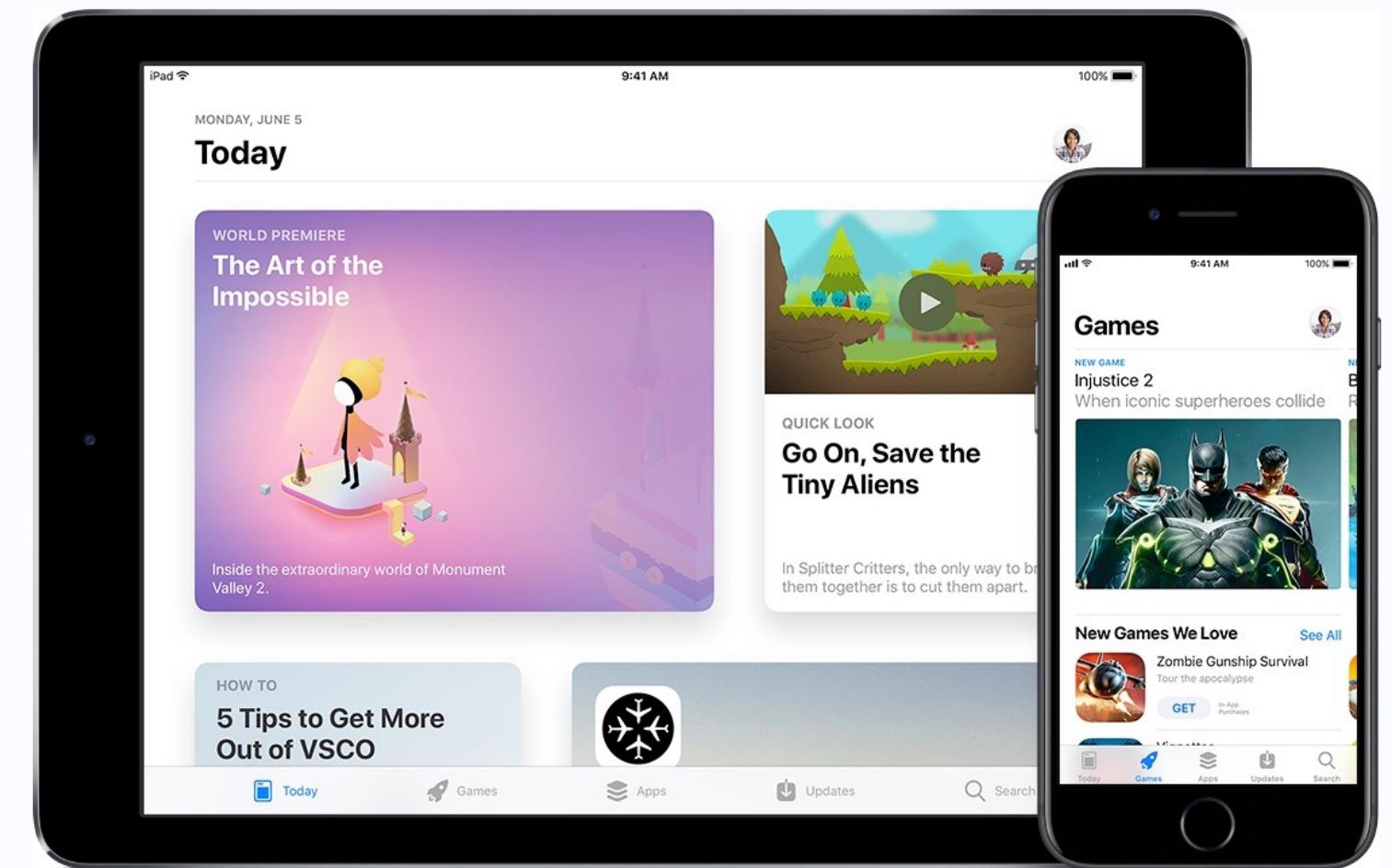
Usage: Swimlanes are stacked content carousels that each show a row of items, enabling visual browsing through several different lists with minimal navigation.³¹



³¹ [Image source](#)

Patterns: Card

Usage: Cards are little rectangles full of images and text that serve as entry points to more detailed information. They can be organized by lists and grids, but they put together different compositions of multimedia content, including images, text, and links, on a column, row, or grid that users navigate through by swiping horizontally and vertically.³²

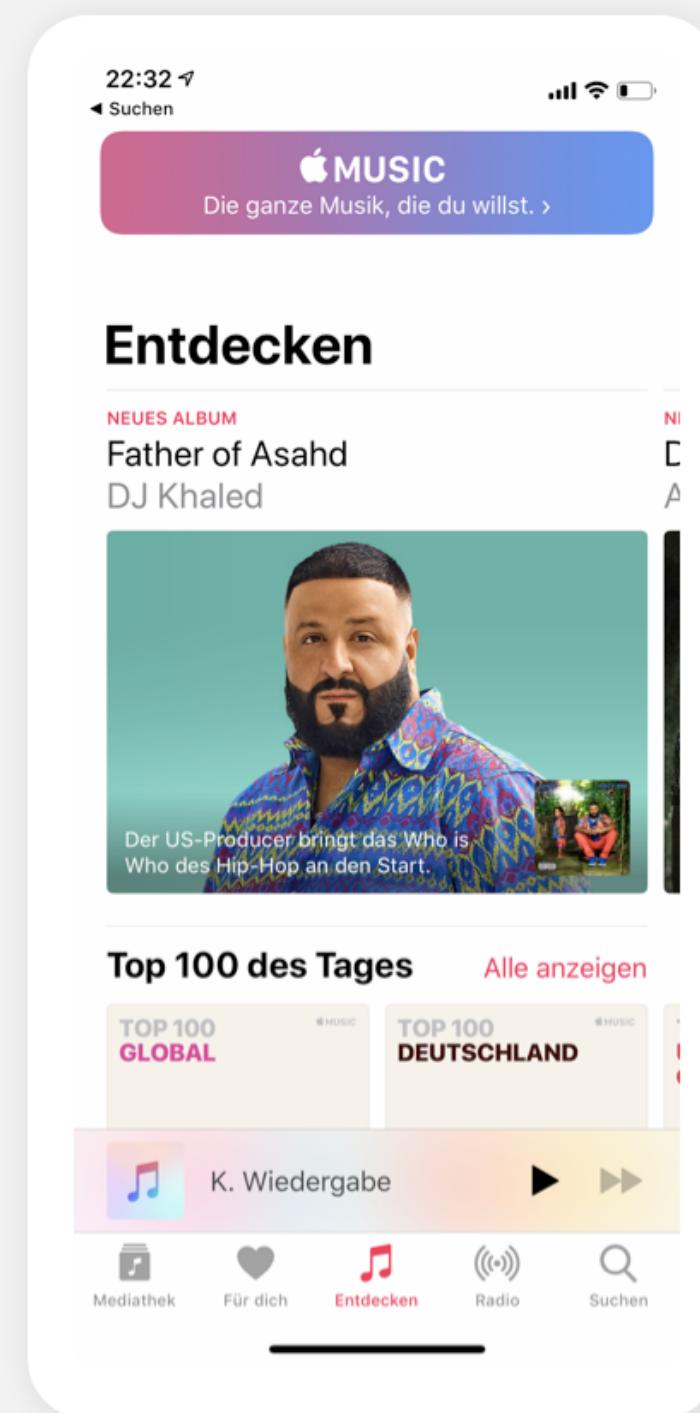


³² [Image source](#)

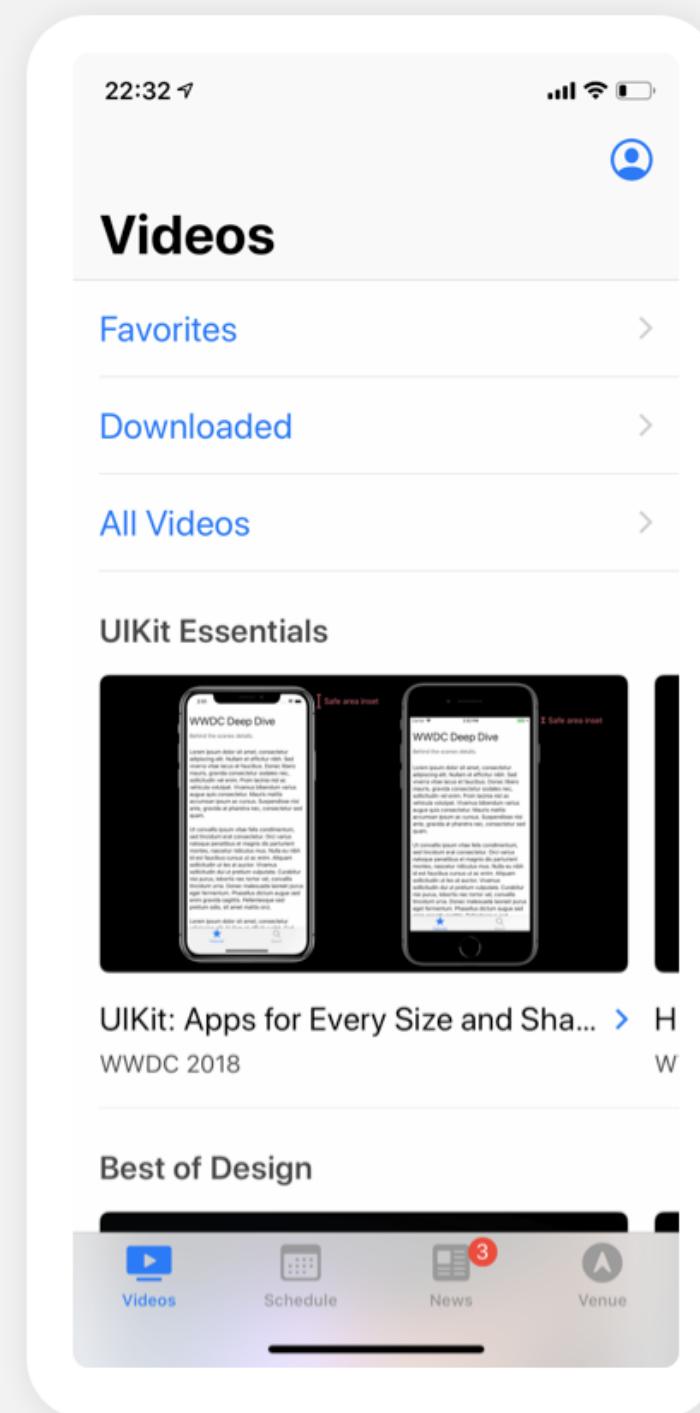
Patterns: Bars

Usage: These are vertical or horizontal strips containing buttons or tabs dedicated to different components or tools.

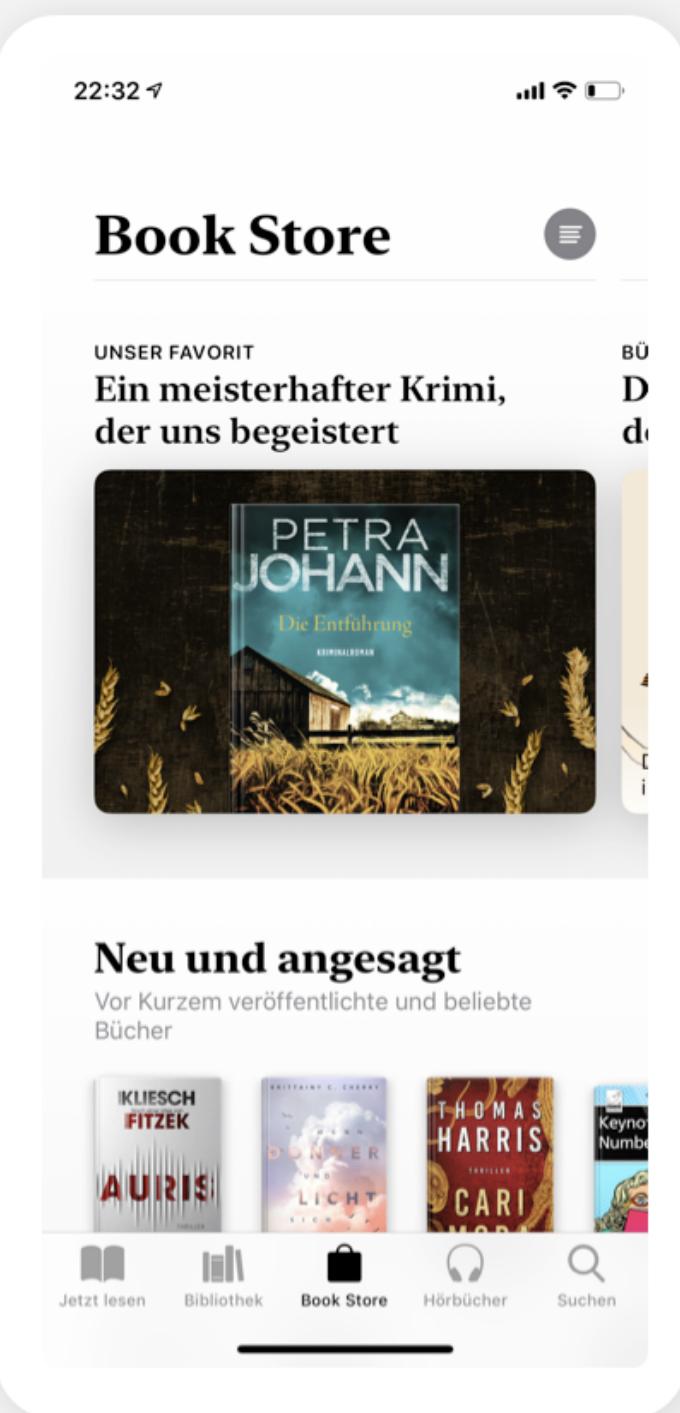
1. *Tab bars* that are placed at the top, bottom, or the side of the screen enable navigation between different components.
2. Bars can also serve as *toolbars* to activate various application or operating-system-level functions.
3. Bars help *navigation* by linking to previously viewed content or to view the previous/next item among multiple screens.



³³ Image source
Apple Music



WWDC



Books



³⁴ Image sources: [left](#), [right](#)

Patterns: Search, sorting, filtering

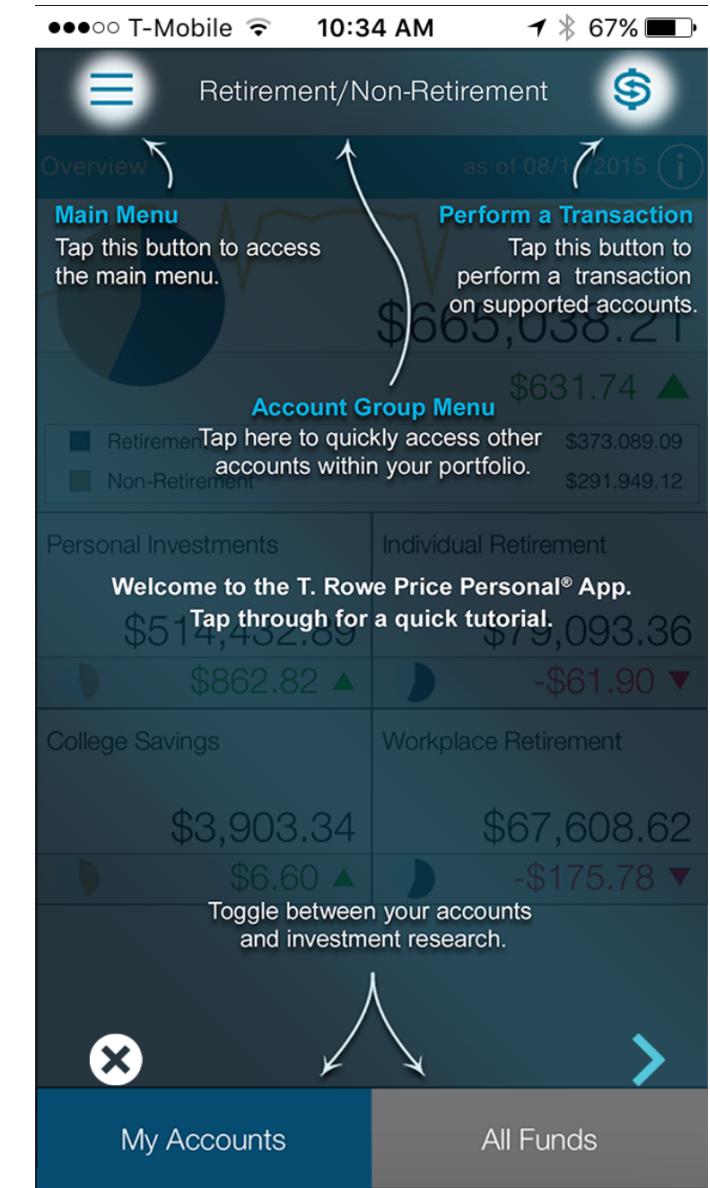
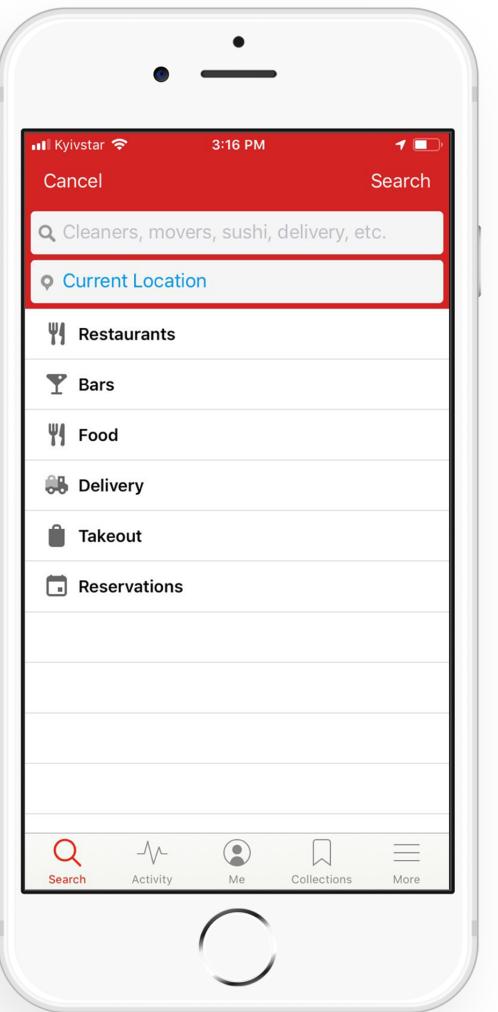
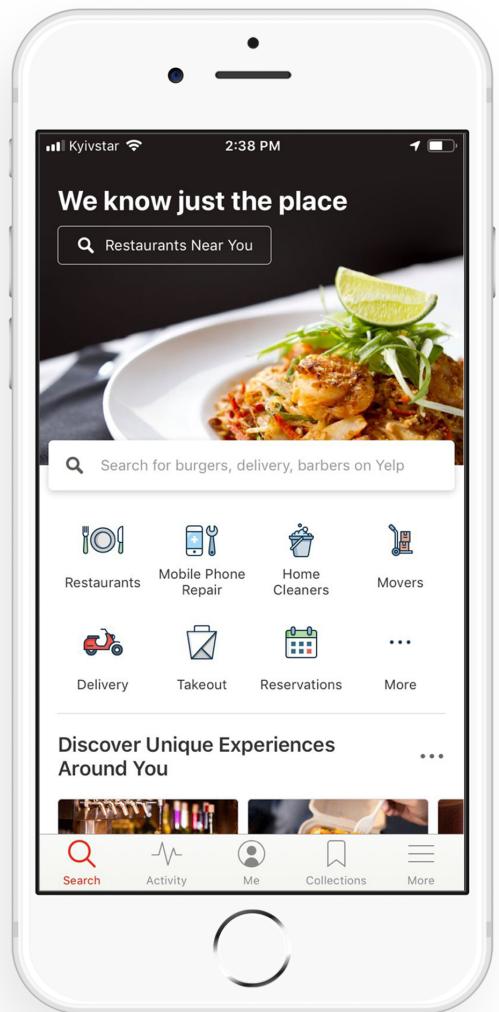
Usage: Used to enable search and filtering to navigate through a large body of content that may be distributed across the entire navigation structure of the application.

These patterns usually provide a search box to enter a search query either by typing text, voice input, or selecting from among a history of searches.

Patterns: Landing pages, guided tours

Usage: Includes a landing, welcome, or home screen that can serve as a portal to the most frequently used functions or as a guide to the next action in the task.

This pattern can also serve as *guided tours* for new users or serve as help screens by overlaying instructions or tooltips on the screen.



³⁵ Image source: [left](#), [right](#)

Advanced Direct Manipulation

Usage: Applications, such as image editors, drawing or presentation tools, or media players, enable direct-manipulation-based controls for content creation or editing.

These patterns provide customized user-input methods, such as control handles for object manipulation or scrollbars to navigate through video content.



³⁶ Image source

Patterns: Panes and panels

Usage: Multi-pane structures and pop-up panels and tools are commonly used in tablets to provide secondary application windows in a way that's similar to desktop applications.³⁷



³⁷ [Image source](#)

Patterns: AR/VR

Usage: Augmented and virtual reality, a.k.a. *mixed reality*, are used to overlay objects or information on images or video of the user's real environment that is captured using the mobile device's camera.³⁸



³⁸ [Image source](#)

What did we learn today?

- Mobile input
- Microinteractions
- Mobile design patterns