



THE UNIVERSITY OF
MELBOURNE

v1.1

UI Development for Mobile Devices

Tilman Dingler

6-Aug-2019

COMP90018 - Mobile Computing Systems Programming

X

“The lecture slides are released in ppt format. It would be more convenient if they are in pdf format (which is more friendly for the mobile devices)”

Microblog: Feedback

“What does ‘innovative’ mean?”

Microblog: Question

“What does ‘innovative’ mean?”

The process of **translating an idea** or invention into a good or **service** that **creates value**

- satisfies one or more needs
- combines existing or non-existing services into something new
- (potentially) leads to wide adoption

Microblog: Question

“Prior to this lecture I only thought of mobile computing as referring to smartphones and wearables, but now I realise there are so many other types. The 4 challenges also intrigued me as I struggle with some of them on a daily basis but have not been aware of such. Ubiquitous computing is definitely going to be an important revelation in mobile computing, and will definitely change the way we live our lives at home, although I question whether most of us are open to this type of technology. For example, would my parents let our fridge buy groceries for us?”

Microblog: Reflection

1. Challenges when designing for mobile devices

2. Mobile devices and user input

3. User interface concepts and elements

4. Aspects of the *User Experience*

5. Getting the design right is inherently hard

Learning Outcomes

Richard Feynman

1918 - 1988



“The first principle is that you must not fool yourself — and you are the easiest person to fool.”

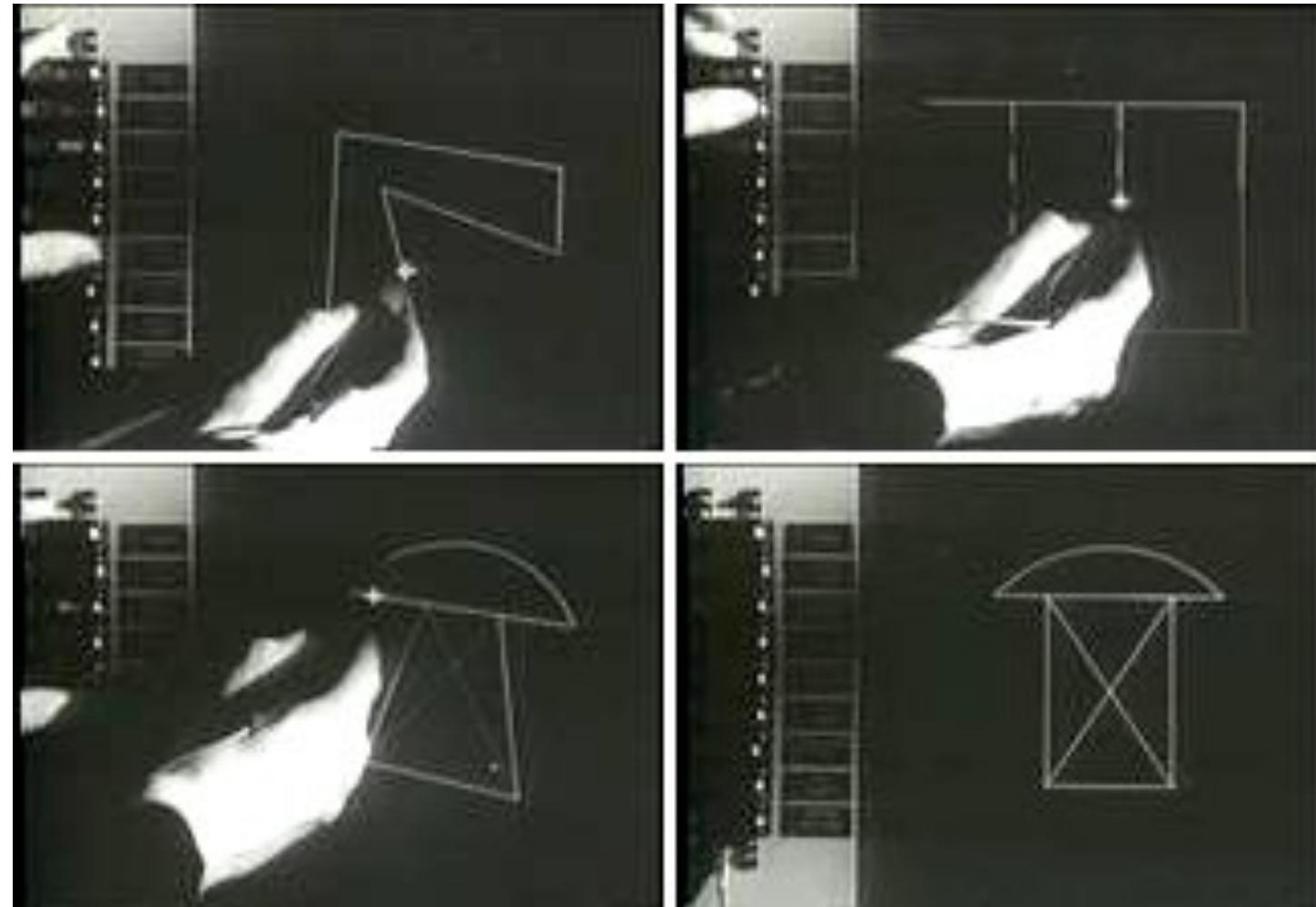
What do I know about

User Interfaces?

Collection by Gary Marsden
UCT, South Africa



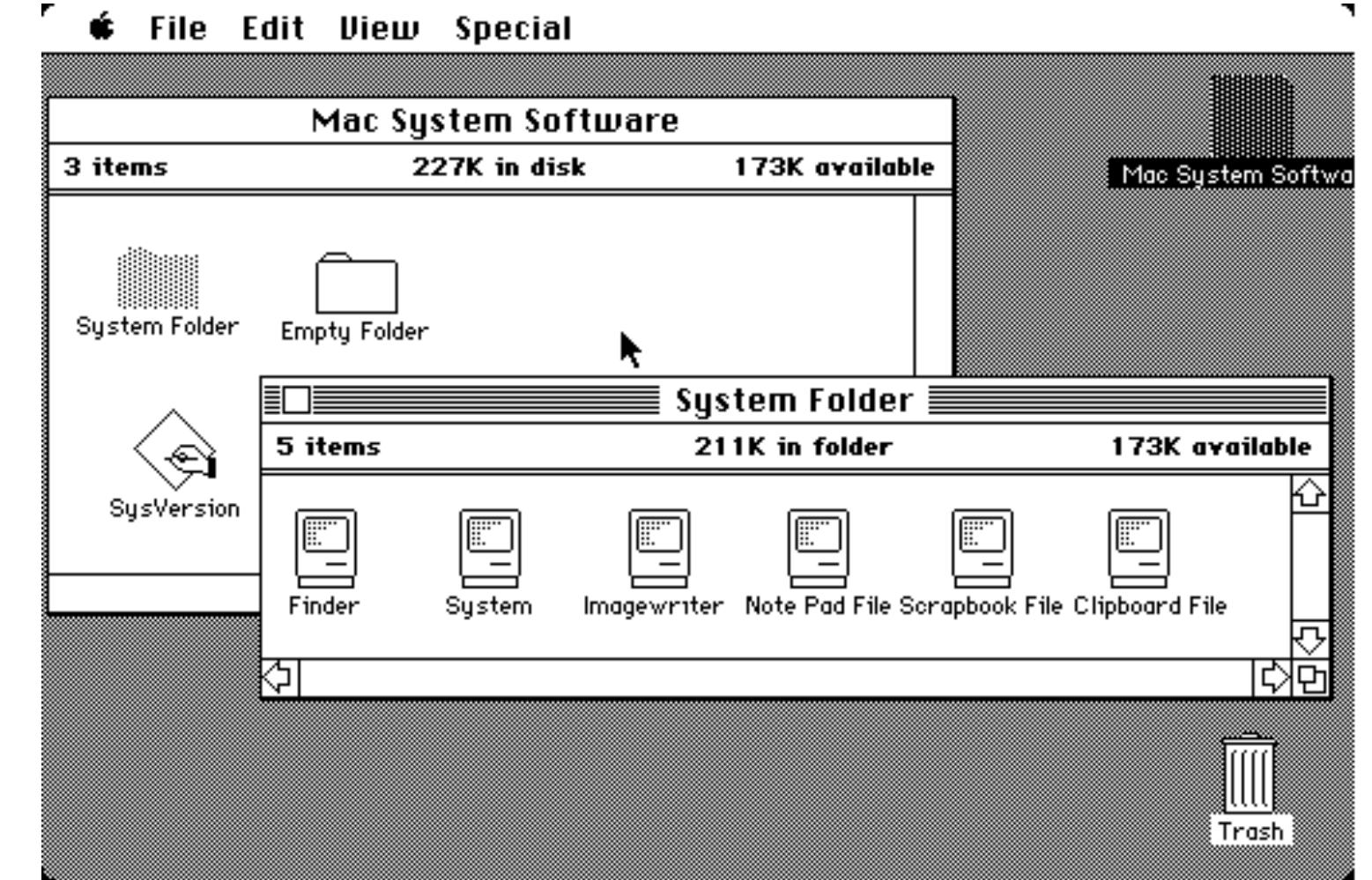
Mobile Computing Interfaces



Ivan Sutherland's Sketchpad, 1963



Xerox Star Workstation, 1973



Macintosh Desktop, 1984

Window, Icon, Menu, Pointer (WIMP)

How would you
improve this interface?



Nokia 6210

How would you
improve this interface?



How would you
improve this interface?

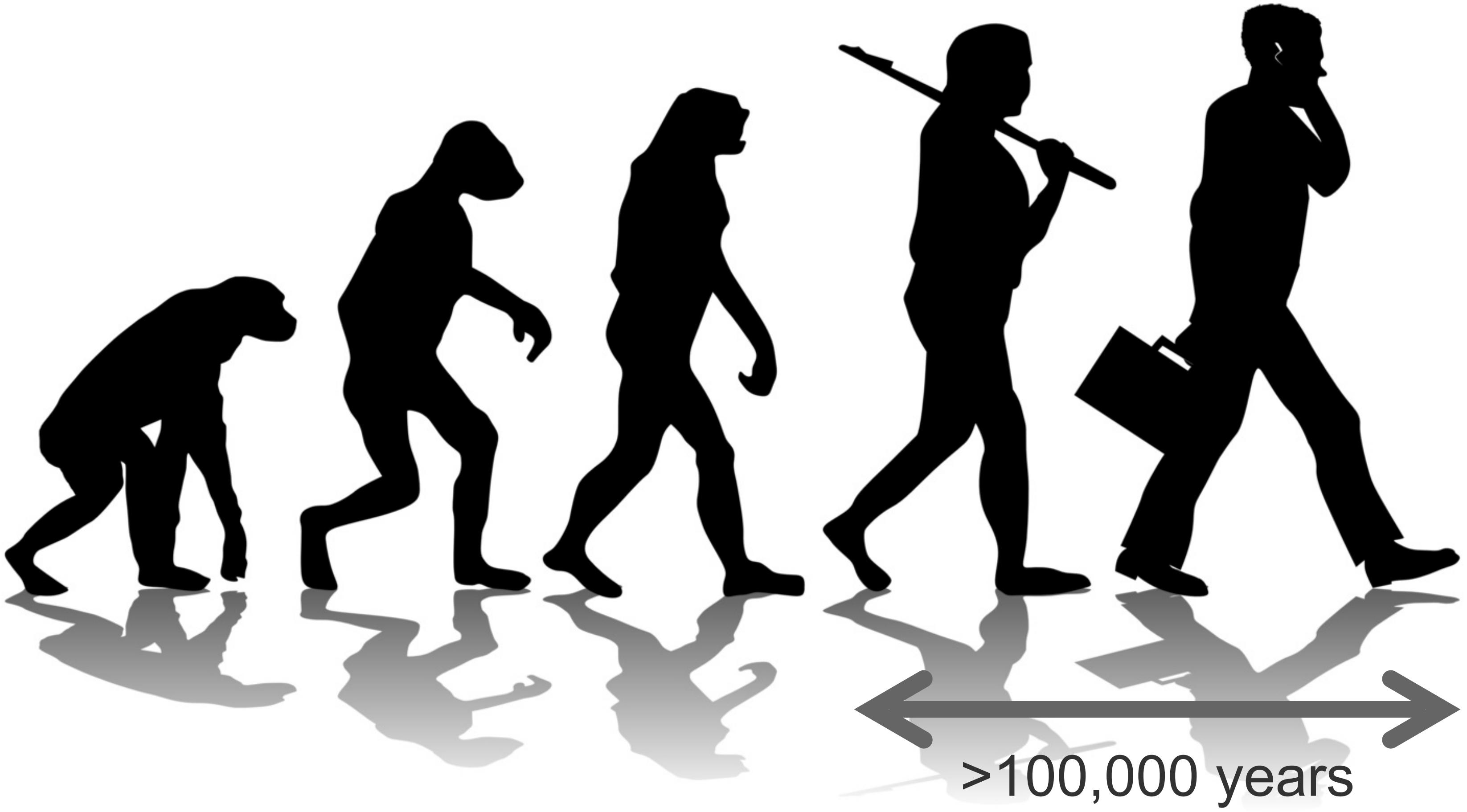




Computing power is not the limiting factor



The interface (and user experience) determines the size



Technical progress is fast – evolution isn't

2003



Nokia 6600

2006



Nokia N95

2007



iPhone

2014



iPhone 5S

2019



Huawei P30

2024

?

What's Next?

Transparent Phones?



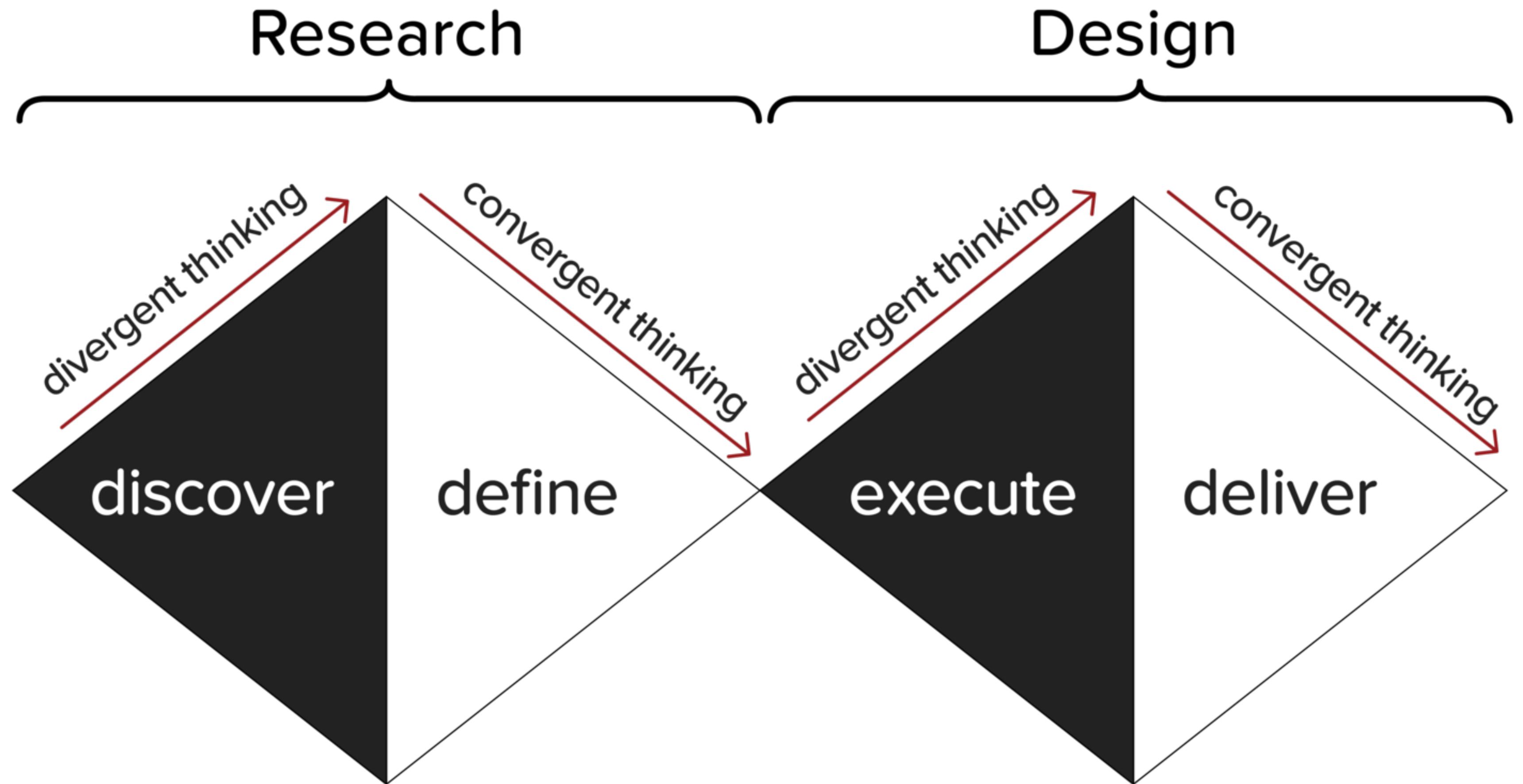
Concept by Seunghan Song:
<http://www.yankodesign.com/2009/08/18/phone-that-shames-the-weather-bureau/>

Flexible Phones?

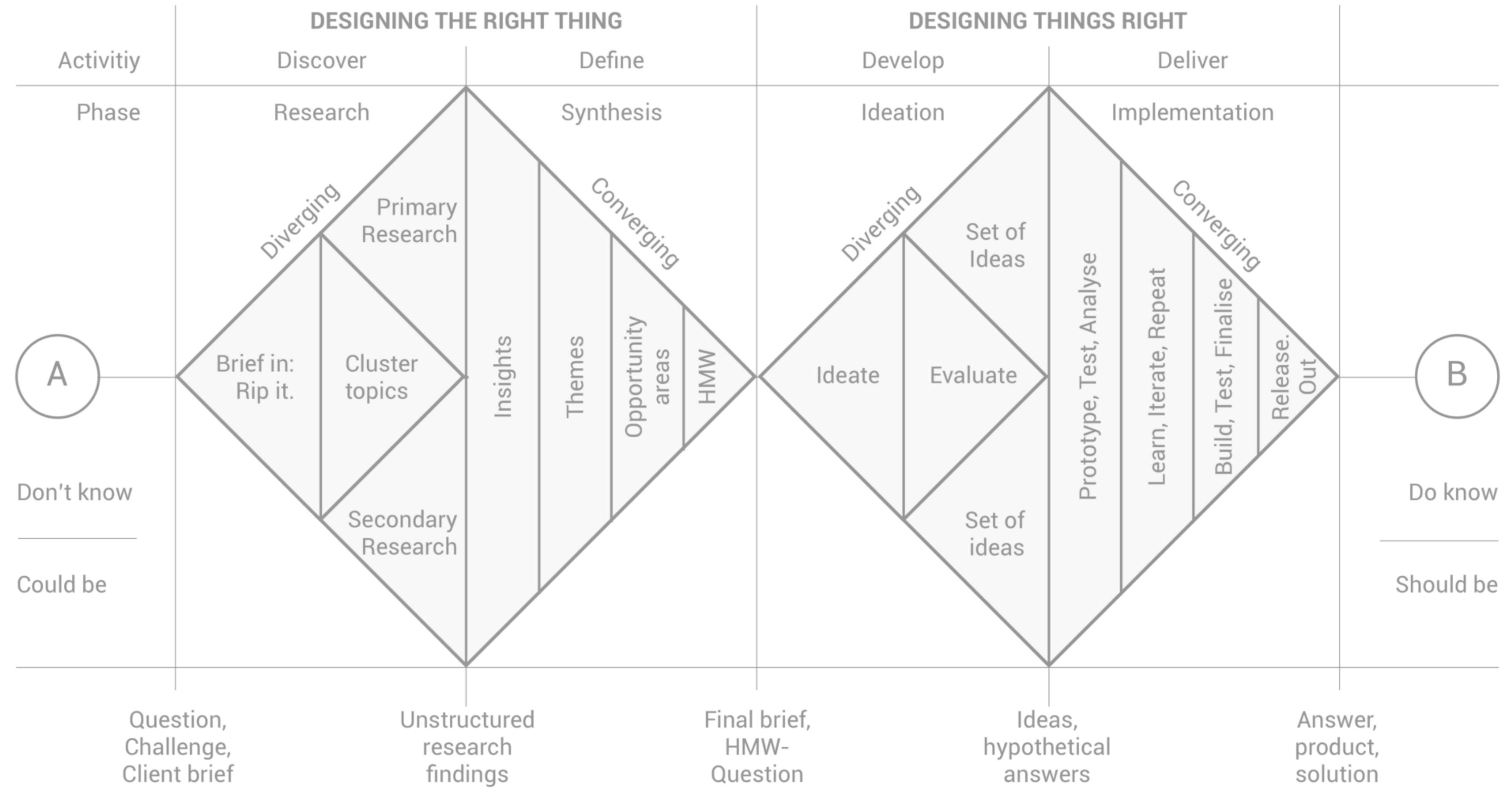


Concept by Dinard da Mata:
<http://www.behance.net/gallery/647321/Fluid>

What's Next?



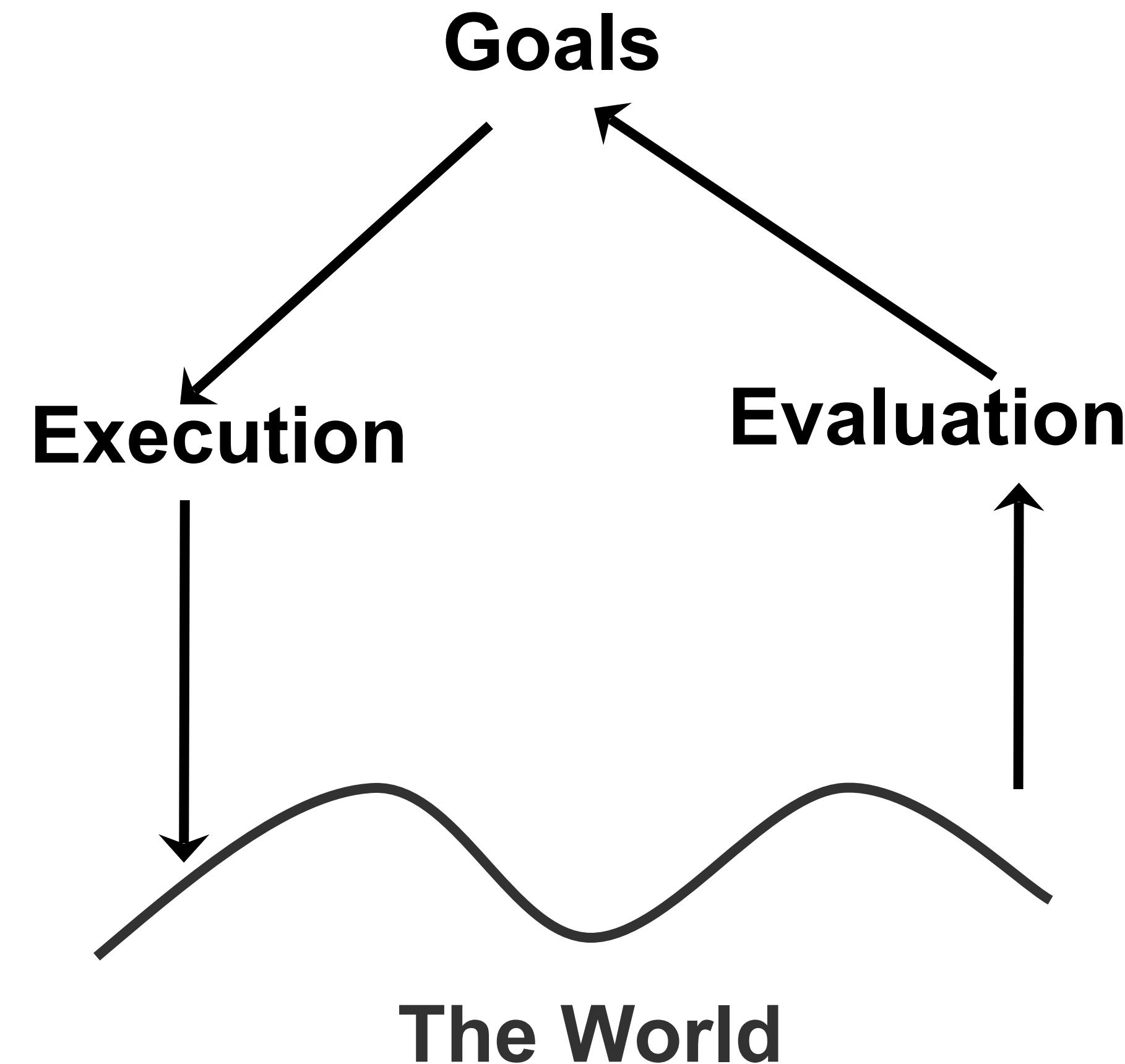
From Conceptual to Technical Design



From Conceptual to Technical Design

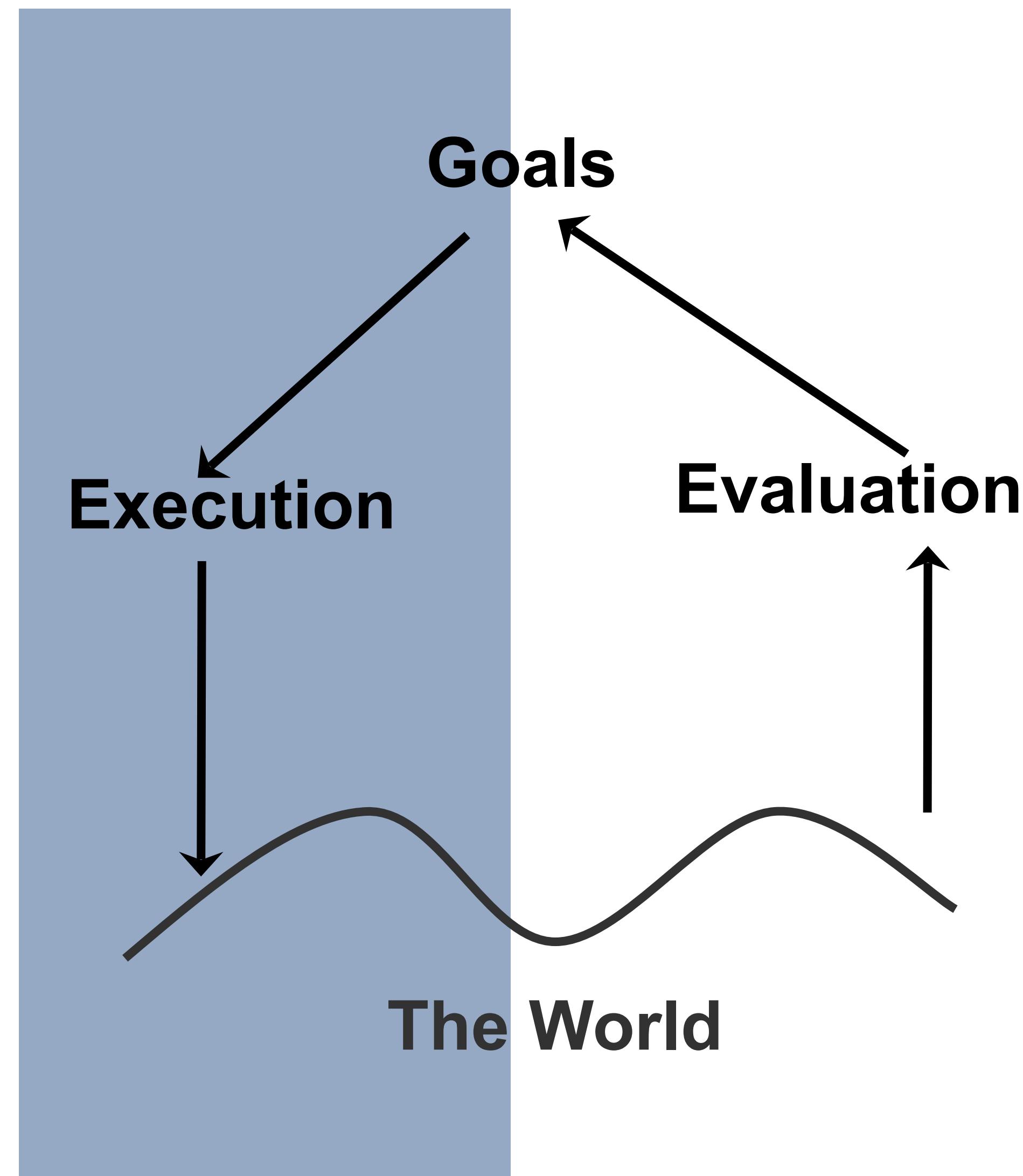
The Action Cycle

- The action is goal directed
 - What do we want to happen?
 - What is the desired state?
- Human action has two major aspects
 - Execution: what we do?
 - Evaluation: compare if what happens is what we want to happen



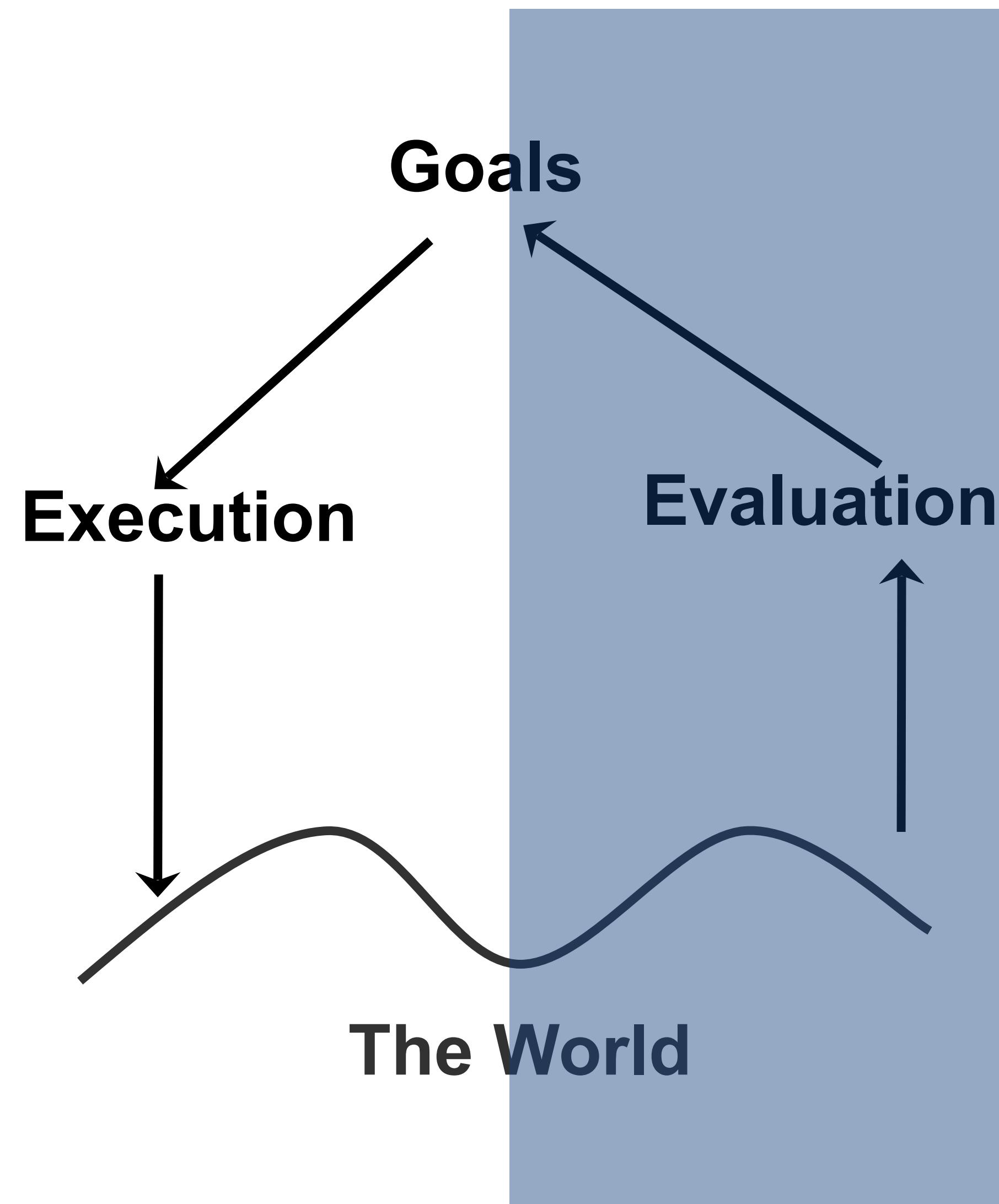
The Action Cycle

- Goal
 - translated into
- An intention to act as to achieve the goal
 - translated into
- The actual sequence of actions that we plan to do
 - translated into
- The physical execution of the action sequence

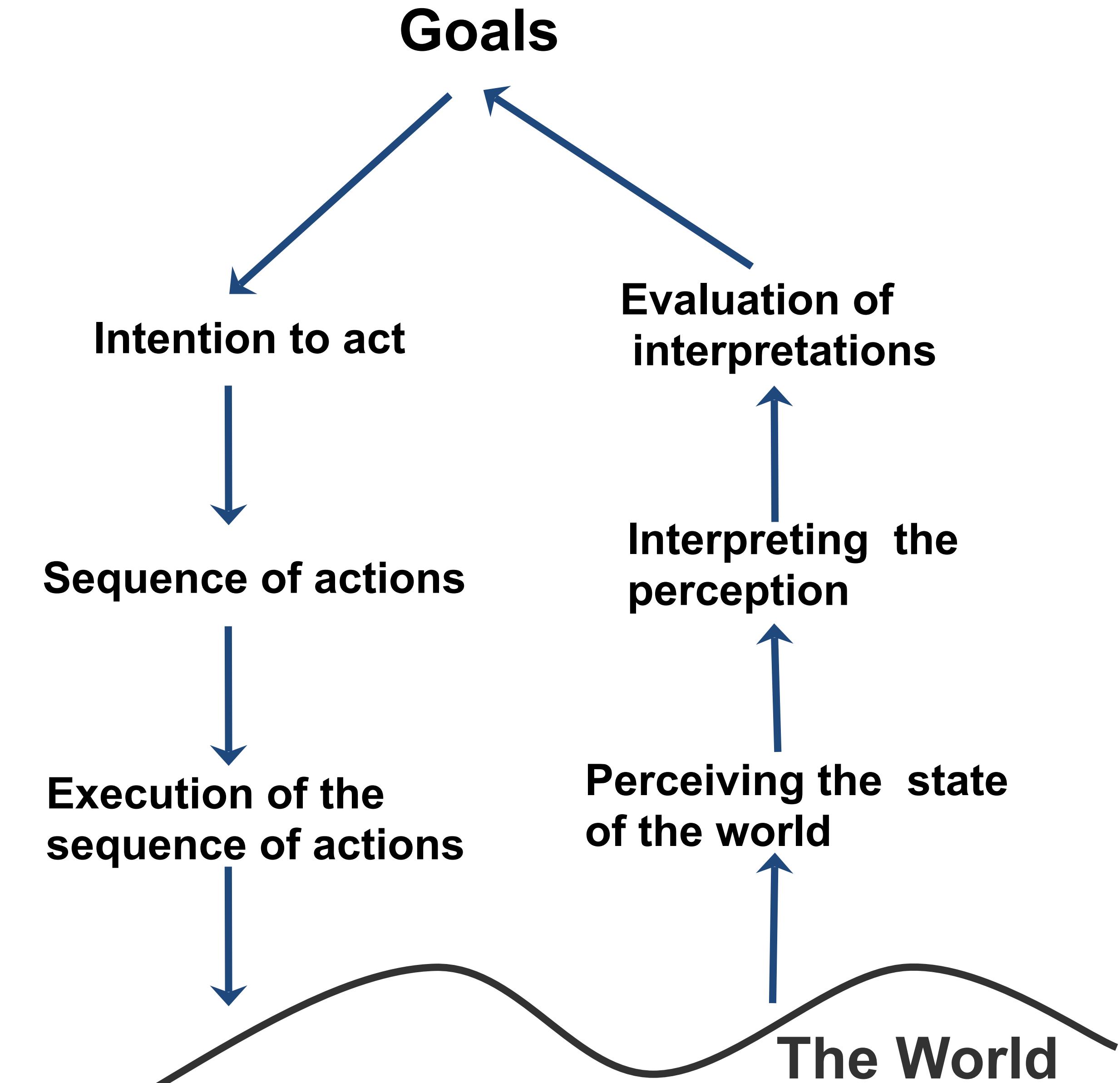


The Action Cycle

- **Perceiving** the state of the worlds
 - followed by
- **Interpreting** the perception according to our expectations
 - followed by
- **Evaluation** of the interpretations with what we expected to happen
 - compared to
- Goal

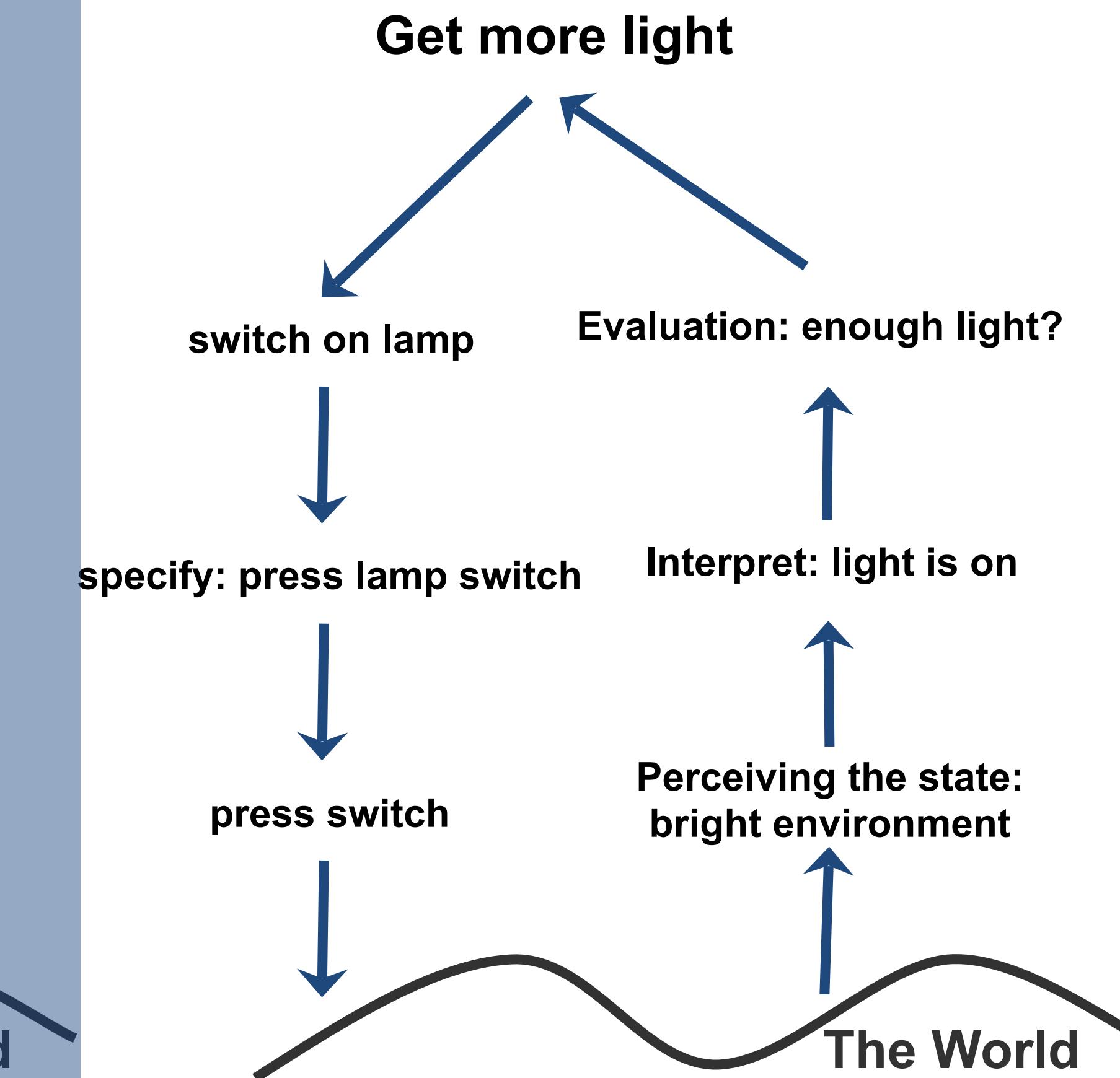
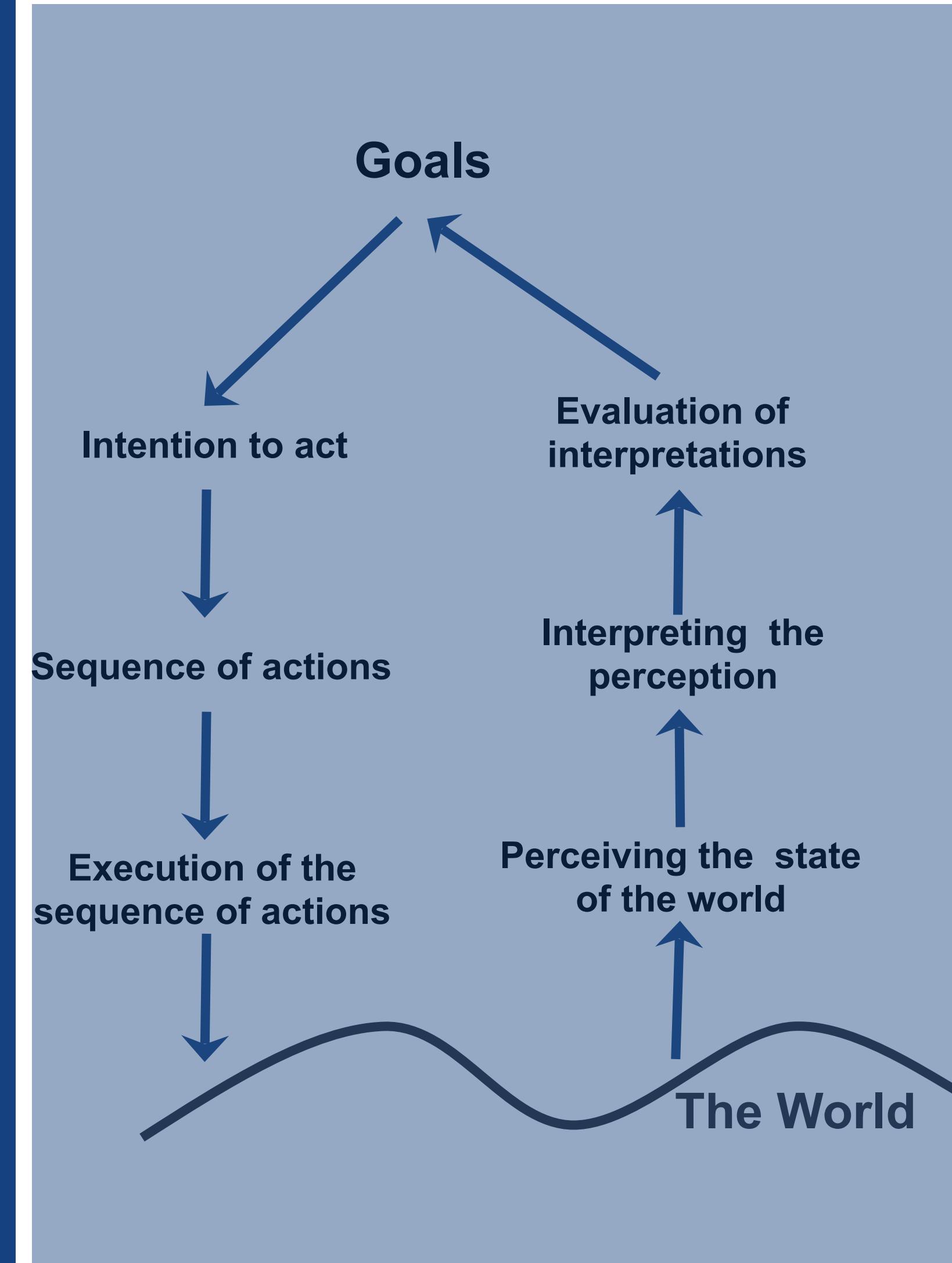


7 Stages of Action



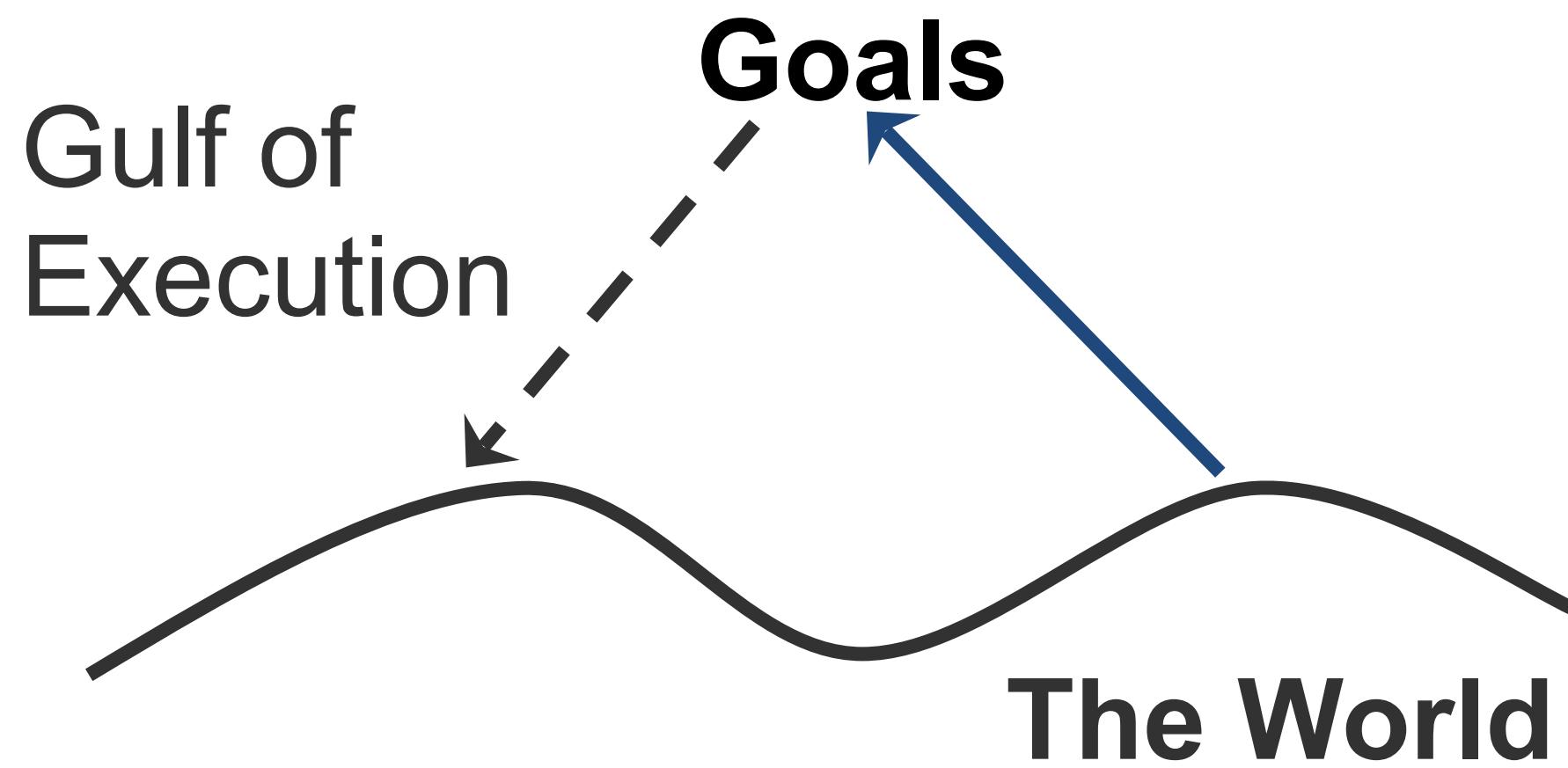
7 Stages of Action

Example



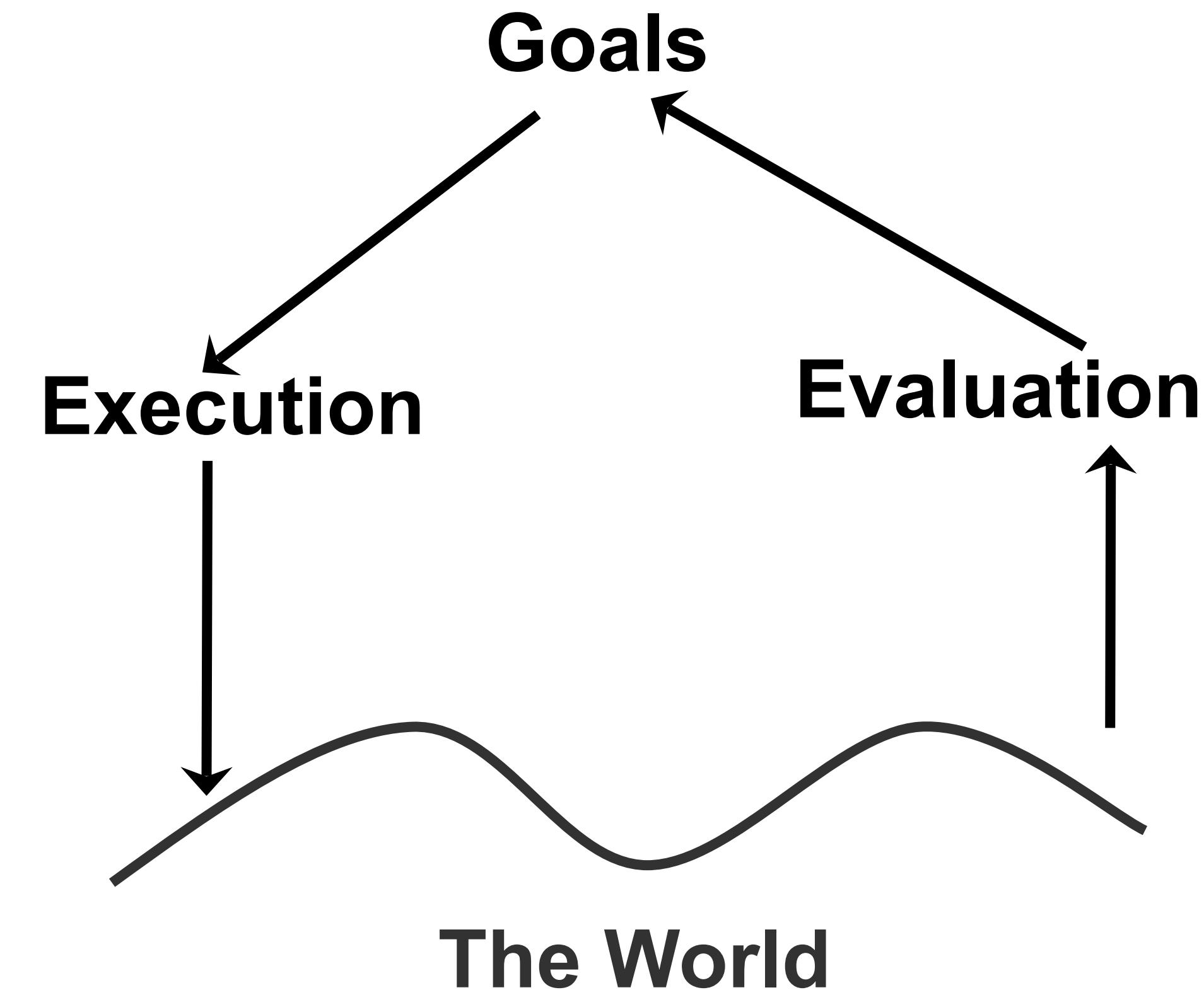
Gulf of Execution

- is the difference between the intentions and the allowable actions
- How directly can the actions be accomplished?
- Do the actions that can be taken in the system match the actions intended by the person?



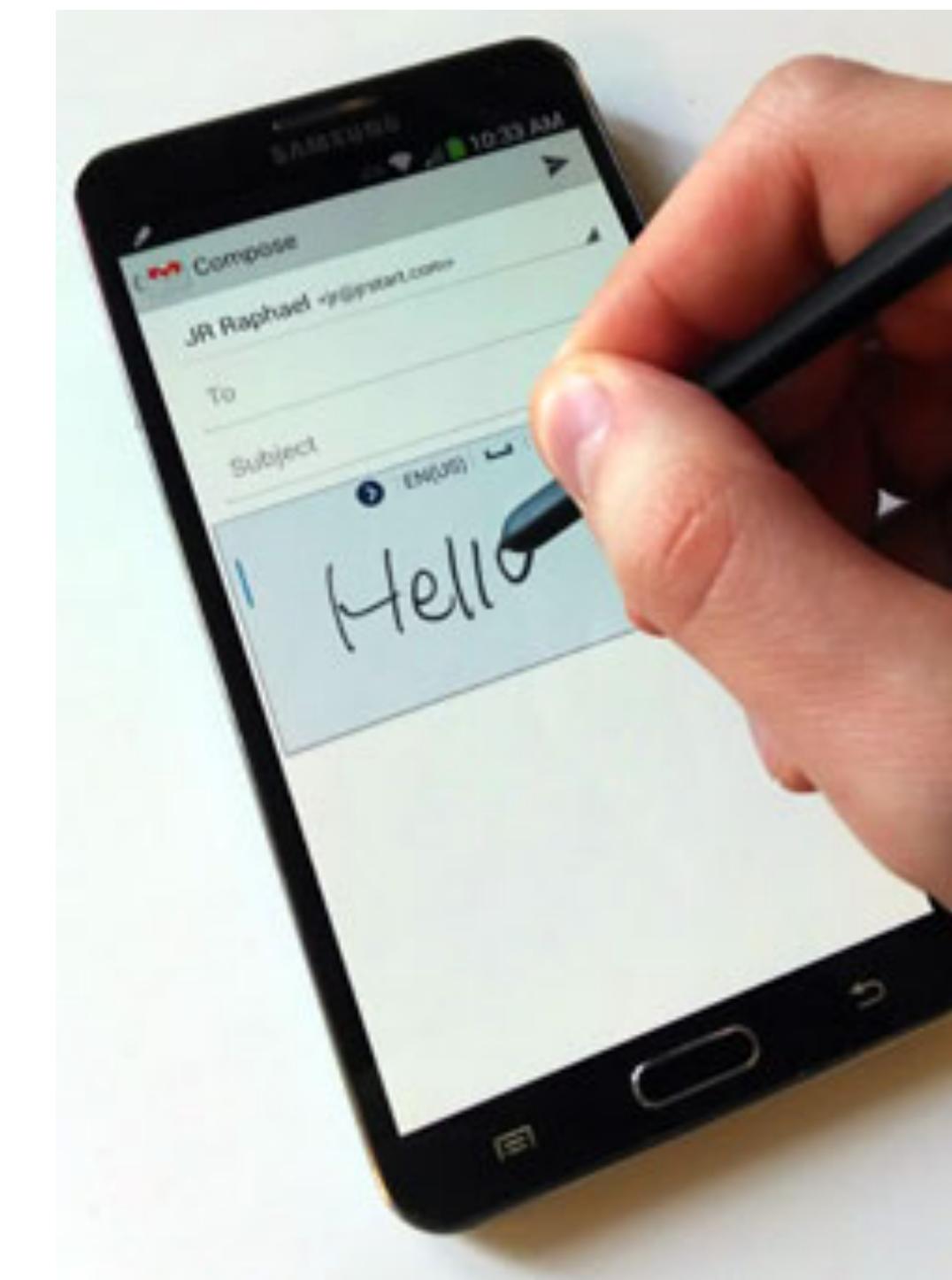
Design Aids

- Visibility
- A good conceptual model
- Good mappings
- Feedback



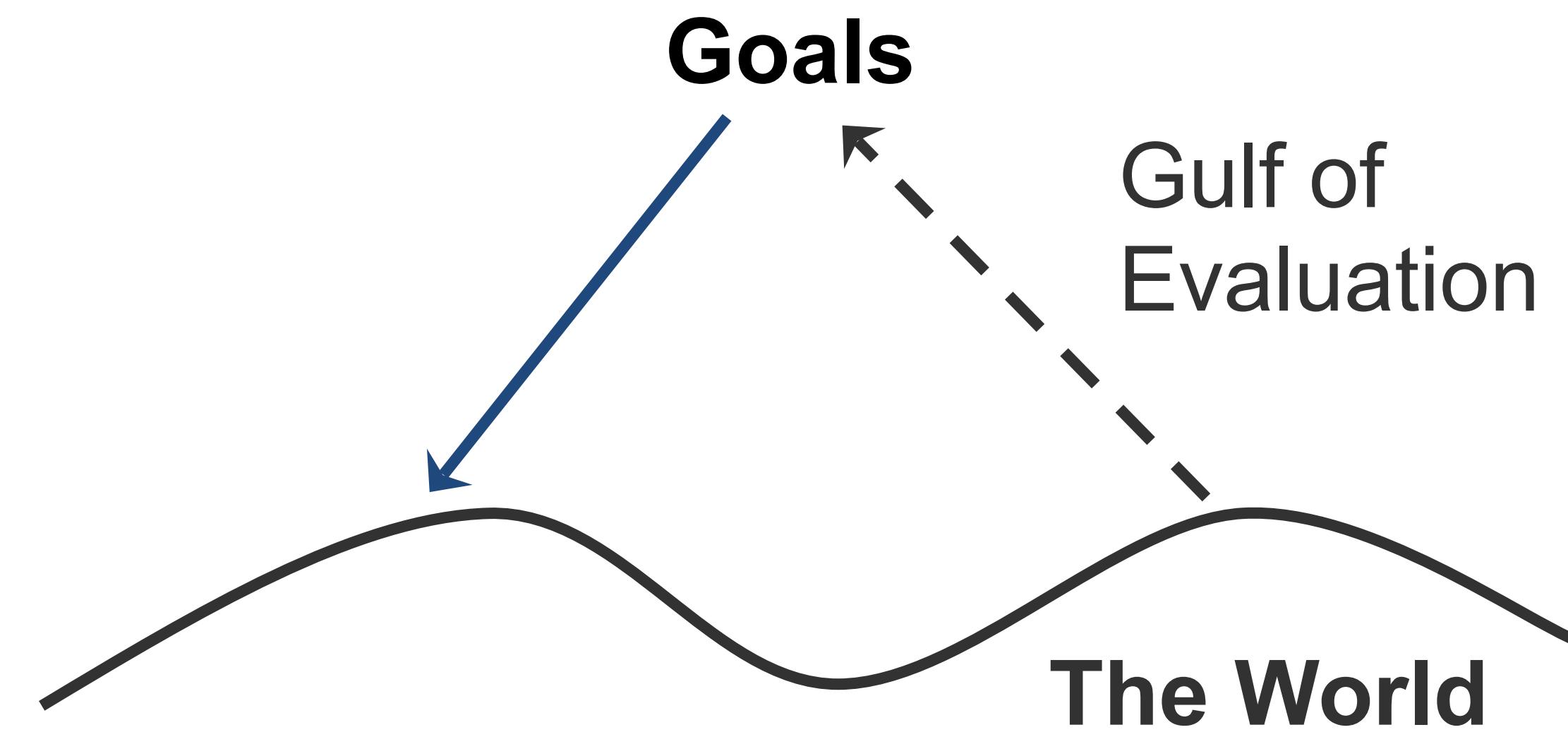
Gulf of Execution

- is the difference between the intentions and the allowable actions
- How directly can the actions be accomplished?
- Do the actions that can be taken in the system match the actions intended by the person?



Gulf of Evaluation

- reflects the amount of effort needed to interpret the state of the system
- Is the information about state of the system easily accessible?
- Is it represented to ease matching with intentions?



Gulf of Evaluation

- reflects the amount of effort needed to interpret the state of the system
- Is the information about state of the system easily accessible?
- Is it represented to ease matching with intentions?



Gulf of Evaluation

- reflects the amount of effort needed to interpret the state of the system
- Is the information about states of the system easily accessible?
- Is it represented to ease matching with intentions?

"The most profound technologies are those that disappear. They weave themselves into the fabric of everyday life until they are indistinguishable from it."

- Mark Weiser

Execution

- Can the user tell what actions are possible?
- Does the interface help with mapping from intention to physical action?
- Does the device easily support required actions?

Evaluation

- Can the user tell if the system is in the desired state?
- Can the user map from the system state to an interpretation?
- Can the user tell what state the system is in?

Design Questions

REVISED & EXPANDED EDITION

*The DESIGN
of EVERYDAY
THINGS*



DON
NORMAN

Recommended Reading

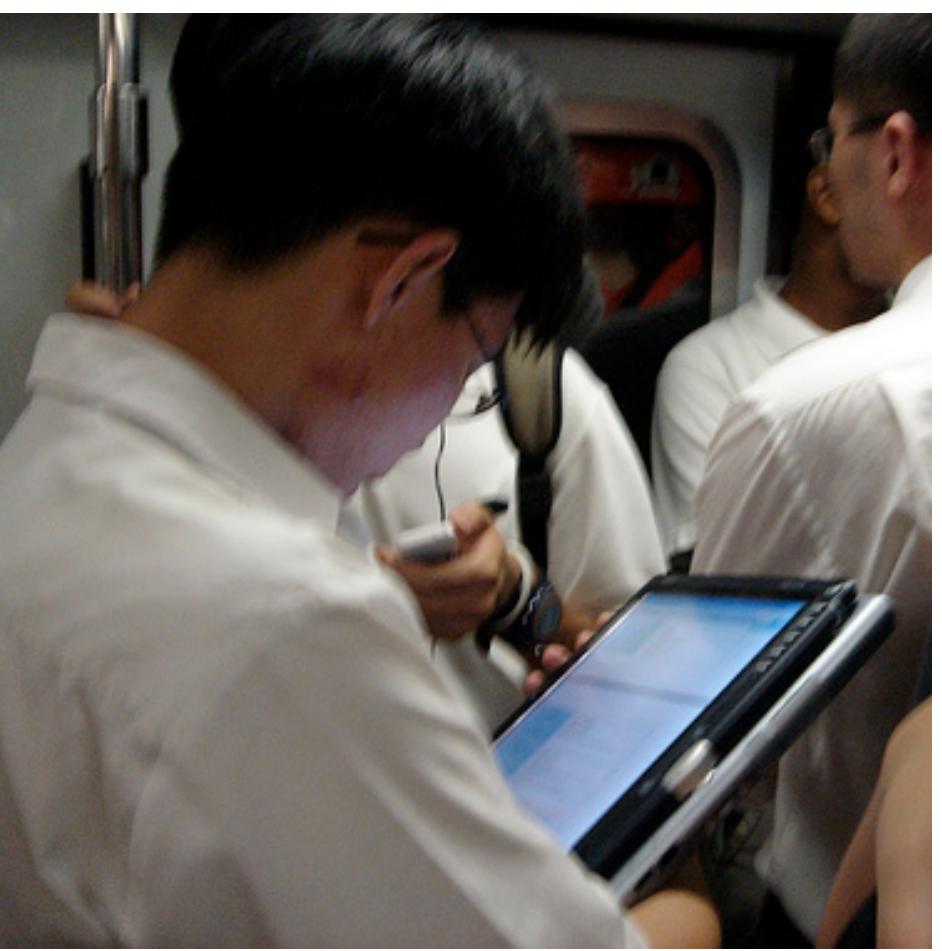
Communication

Mobility

Portability

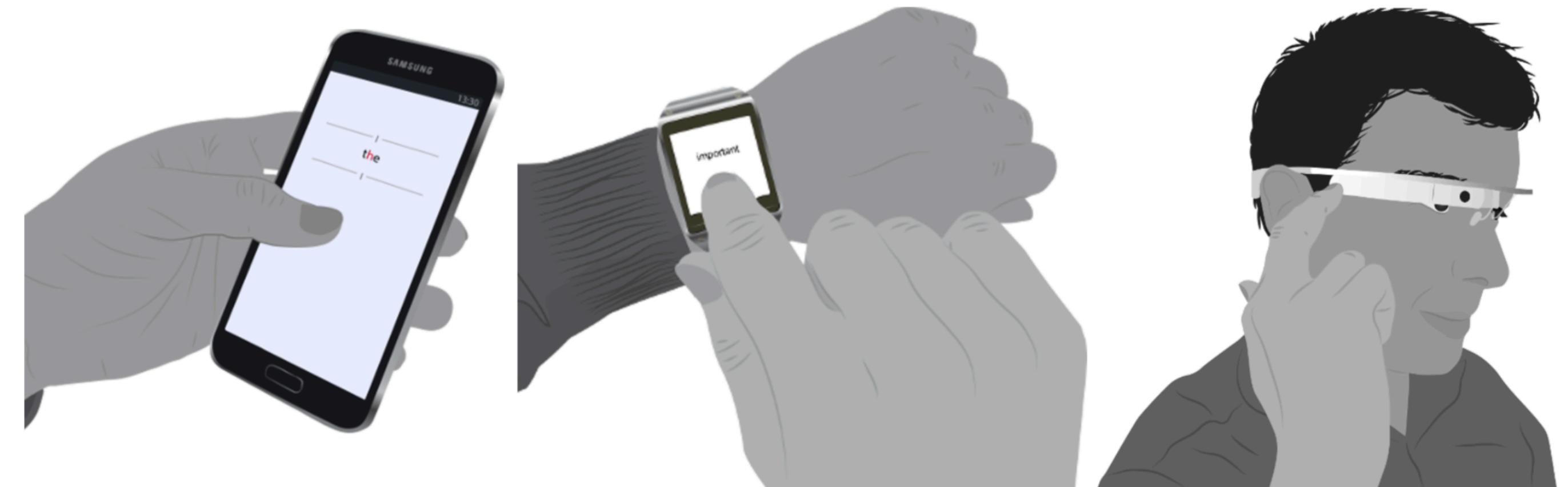
- More frequent disconnections
- Lower bandwidth
- Higher latency
- Variation in available bandwidth
- Greater network heterogeneity
- Increased security risks

Social impact



Designing for Mobility

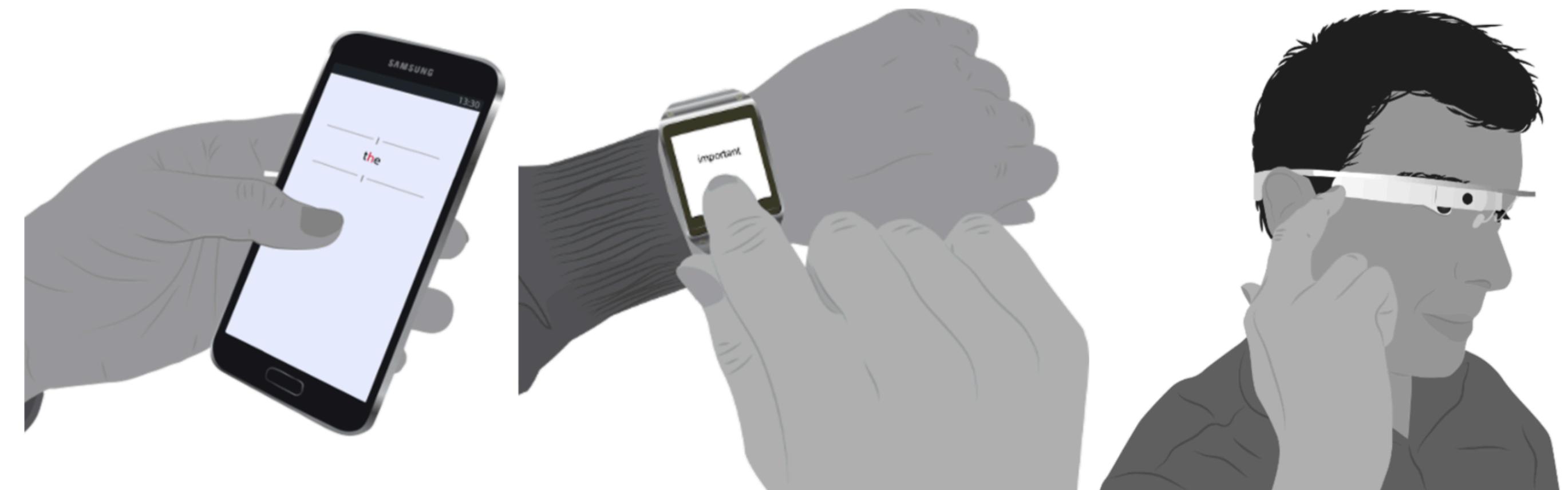
- Different **form factors**
 - e.g., screen size, handling
- Different **capabilities**
 - e.g., battery life, sensor availability,...
- **Accessibility**



Designing for Cross-Device Usage

Ensure portability across devices

- High-level API
- Use platform-independent parts of low-level API
- Responsiveness: discover device capabilities



Ensure Portability across Devices

Web vs. Native

- Server-based approach
 - Create a web service
 - Client (the mobile device) accesses the content via a browser
- Device-based (native) approach
 - Develop application with an SDK
 - Deploy the application locally on the mobile device



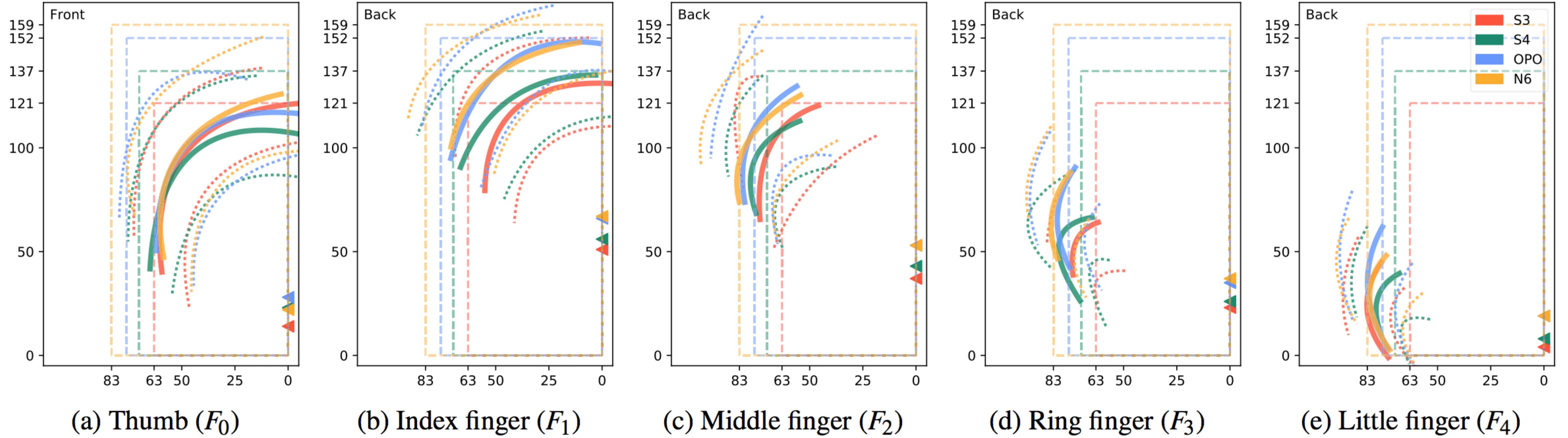
Web vs. Native

- Server-based approach
 - Create a web service
 - Client (the mobile device) accesses the content via a web browser
- Device-based approach
 - Develop application with an SDK
 - Deploy the application locally on the mobile device



"Responsive web design (RWD) is an approach to web design that makes web pages render well on a variety of devices and window or screen sizes. ."

- Wikipedia



Input on Mobile Devices

Keypad Input

- Mobile phones: 12-digit keypad
- Good for numbers, cumbersome for text
- Predictive input technology: T9 = text on 9 buttons; only one button press per letter required



Keyboard Input

- Bluetooth keyboards
- Thumb-based keyboards
(BlackBerry devices)
- Virtual keyboards



Keypads and Keyboards

Text Metrics

- Input speed: words per minute (wpm)
- Accuracy

Standard phones

- Multi-tap: 8 – 20 wpm,
world record: 29 wpm
- T9: approximately 20 wpm
- Touch: 22 - 56 wpm

Special hardware

- Twiddler, 26 to 47 wpm
- QWERTY keyboard: 34 wpm
(world record: 212 wpm)
- IBM SHARK: 60 – 80 wpm



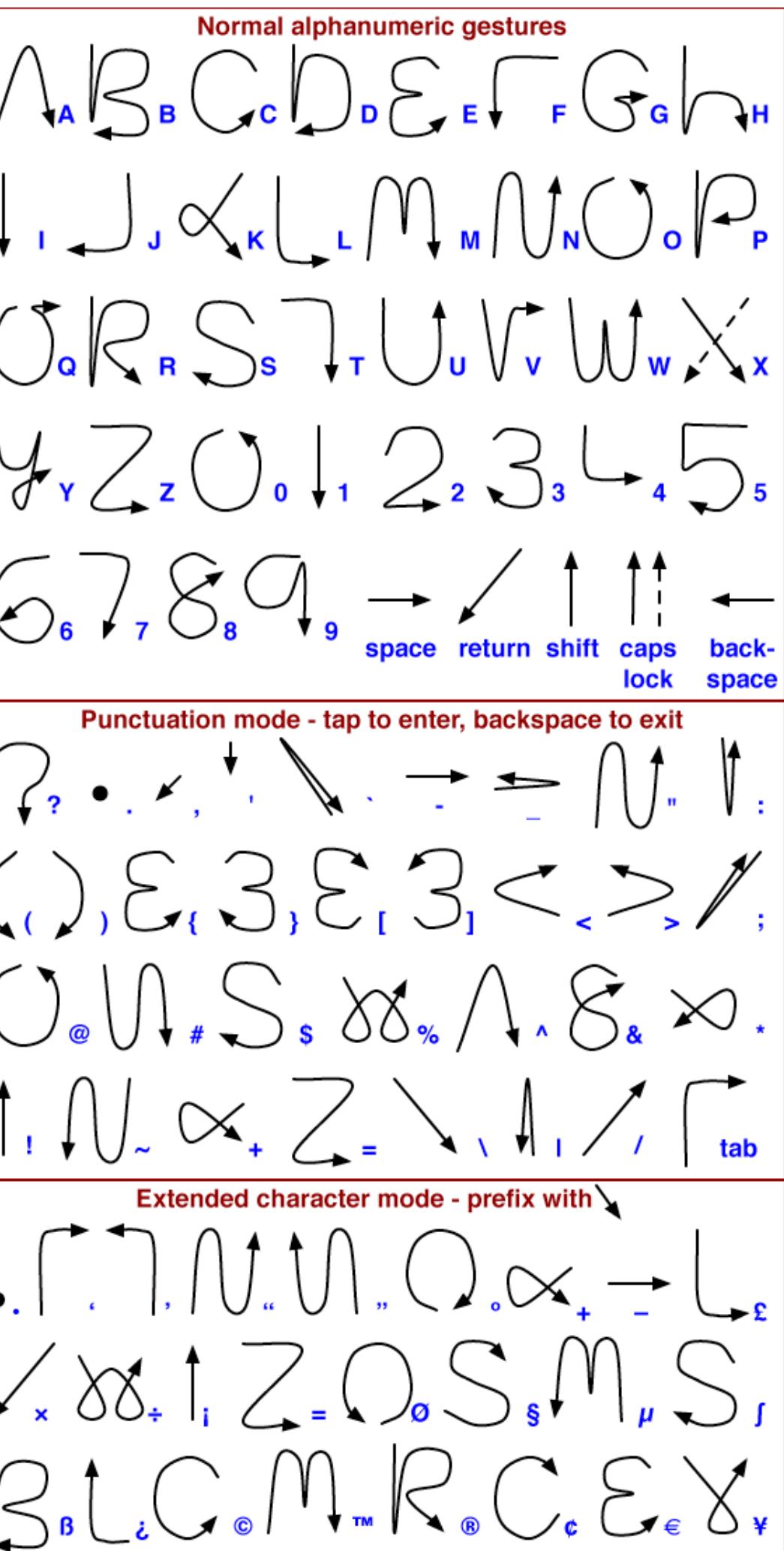
Keypads and Keyboards

Pen-based Input

- Touchscreen with a stylus
 - Soft keyboards
 - Character recognition
 - Handwriting recognition
 - Graffiti (Palm OS)

Voice input

- Simple commands
 - Earlier: VoiceXML where complex commands are recognised by a server
 - Now: Siri, Cortana, Alexa



Pens, stylus, voice

Text Entry on Smart Watches

- ZoomBoard

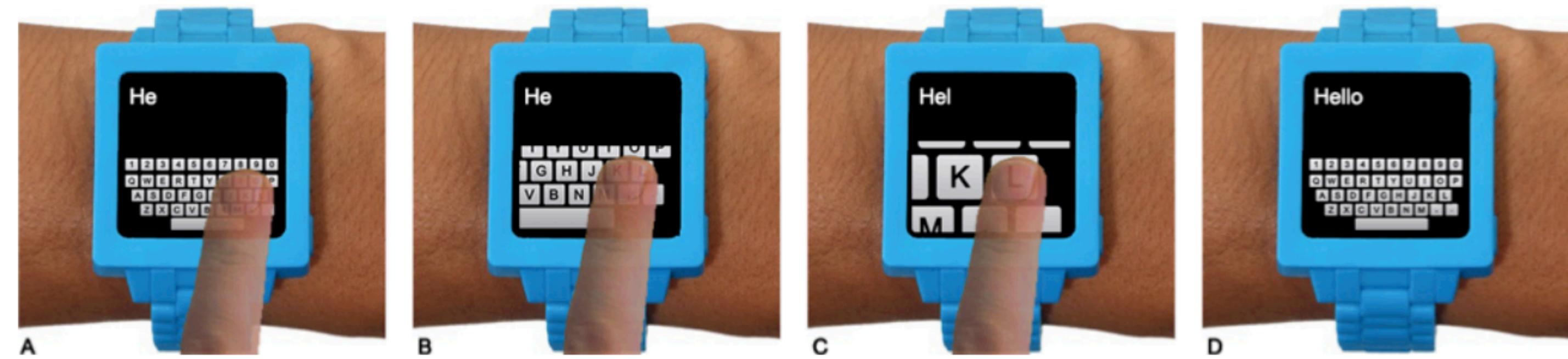


Figure 1: ZoomBoard on a watch-sized device. The keyboard is fully zoomed out by default (A). When users press a key, the keyboard iteratively zooms in (B & C), until the keys are a size that is comfortable and accurate (C). After the desired character is entered, the keyboard resets (D). Users may also swipe to the left to delete, to the right for a space, and up to switch to a symbols keyboard.

Text Entry on Smart Watches

- ZoomBoard
- Touch-Sensitive Wristbands



Figure 1: The prototypes with the two keyboard layouts. The multitap layout is shown on the left and the linear keyboard is shown on the right.

Omnitouch

- ZoomBoard
- Touch-Sensitive Wristbands
- Omnitouch



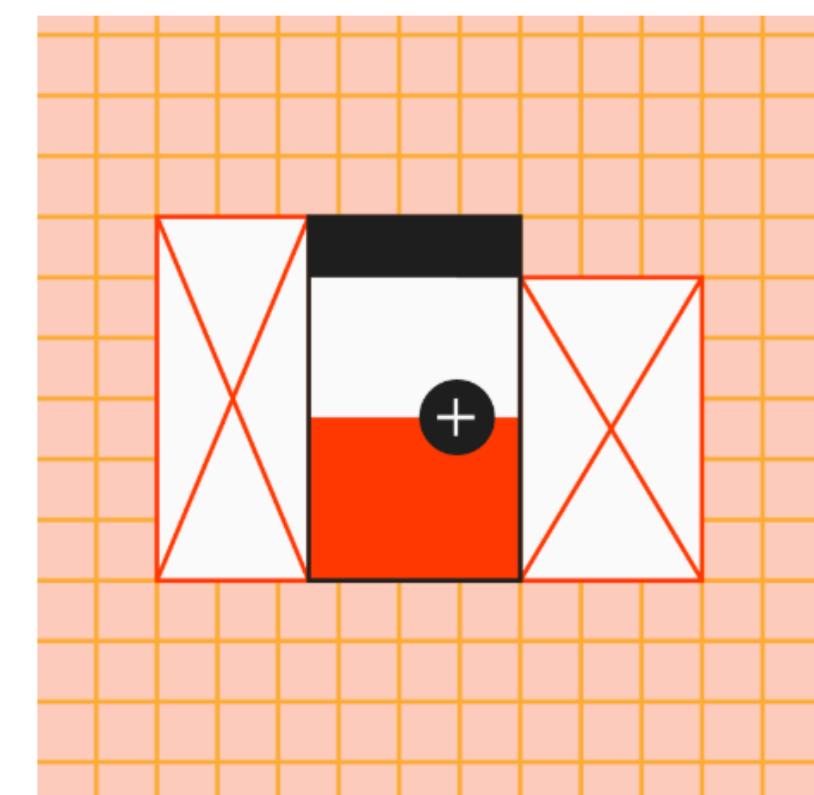
Harrison, C., Benko, H., and Wilson, A. D. 2011. OmniTouch: Wearable Multitouch Interaction Everywhere. In Proceedings of the 24th Annual ACM Symposium on User interface Software and Technology (Santa Barbara, California, October 16 - 19, 2011). UIST '11.



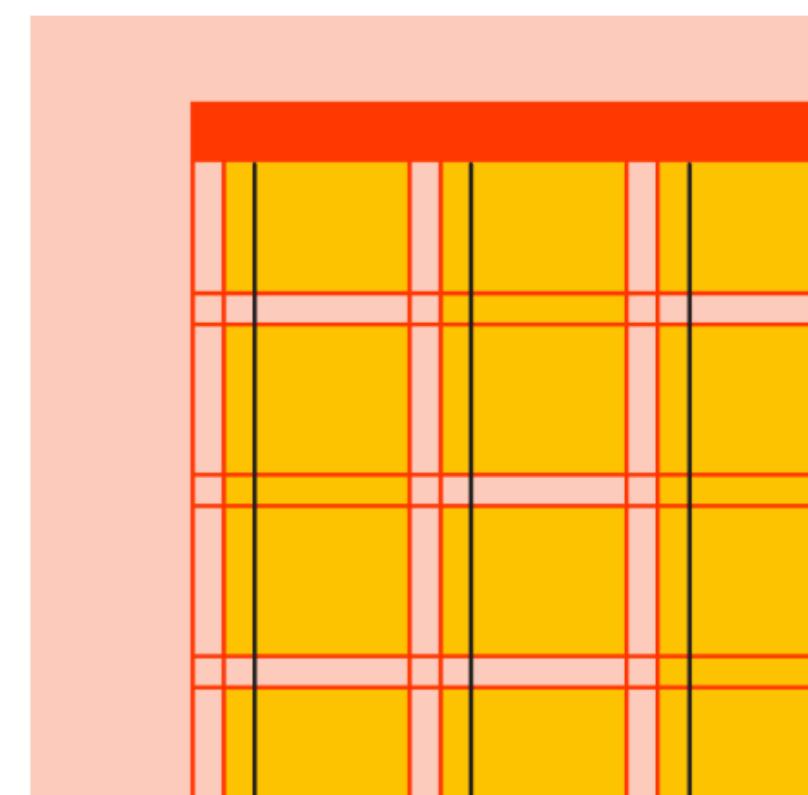
Tilt for Text Input?

KISS principle

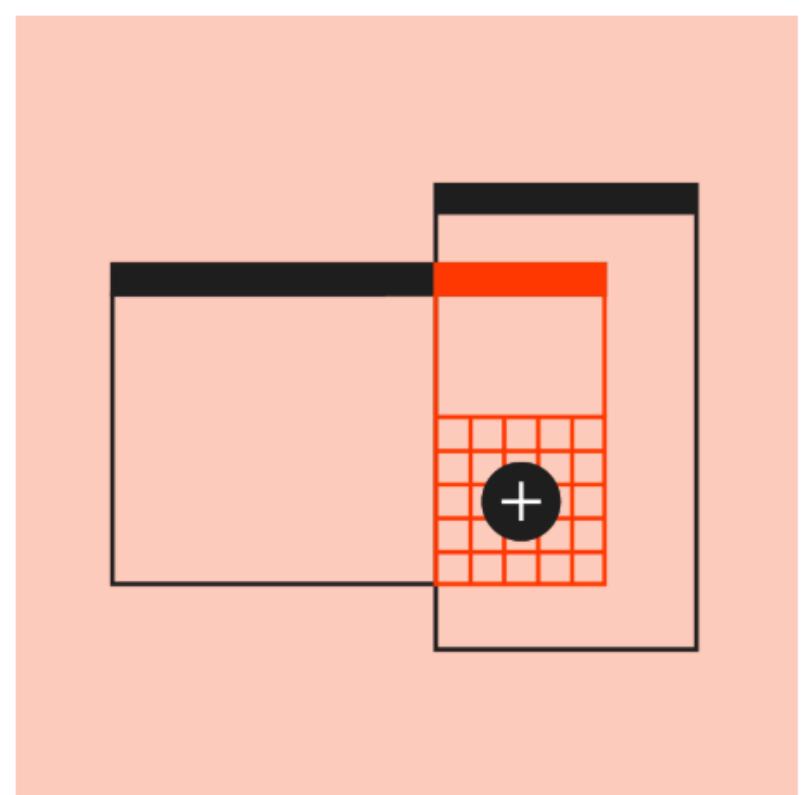
- Simple and easy-to-use UI
- Minimise user input
- Pre-select likely choices



Predictable



Consistent



Responsive

Mobile User Interface Elements

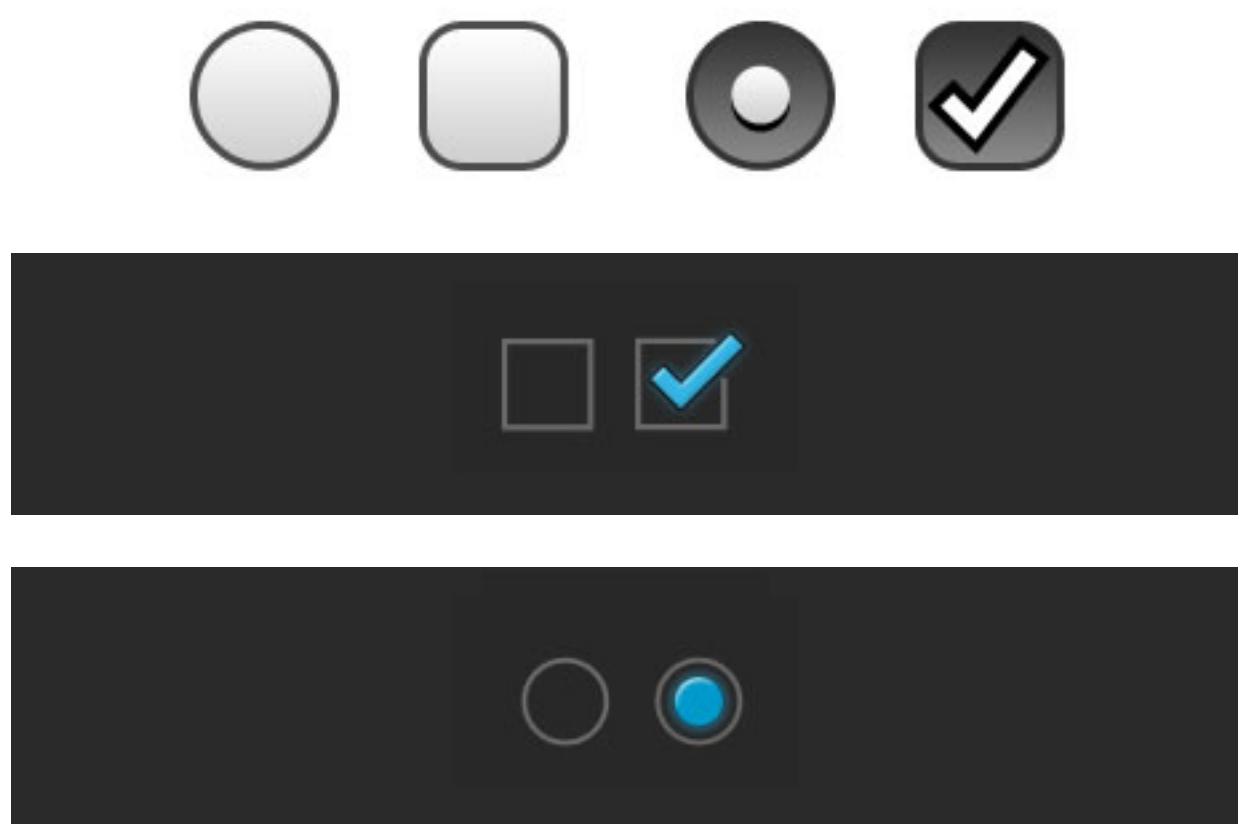
Buttons

- iOS Developer Library (UIButton): “Intercepts touch events and sends an action message to a target object when tapped”
- Android API (Button): “Communicates what action occurs when the user touches it”



Checkboxes & radio buttons

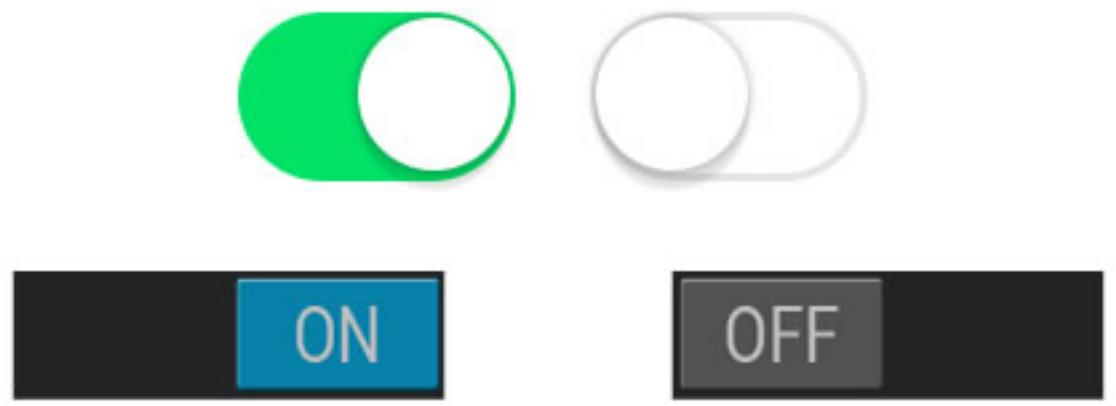
- *iOS Developer Library (UIButton)*
- Android API (Checkbox): “Select one or more options from a set”
- Android API (Radio Button): “Select one option from a set”; options are mutually exclusive



UI Elements

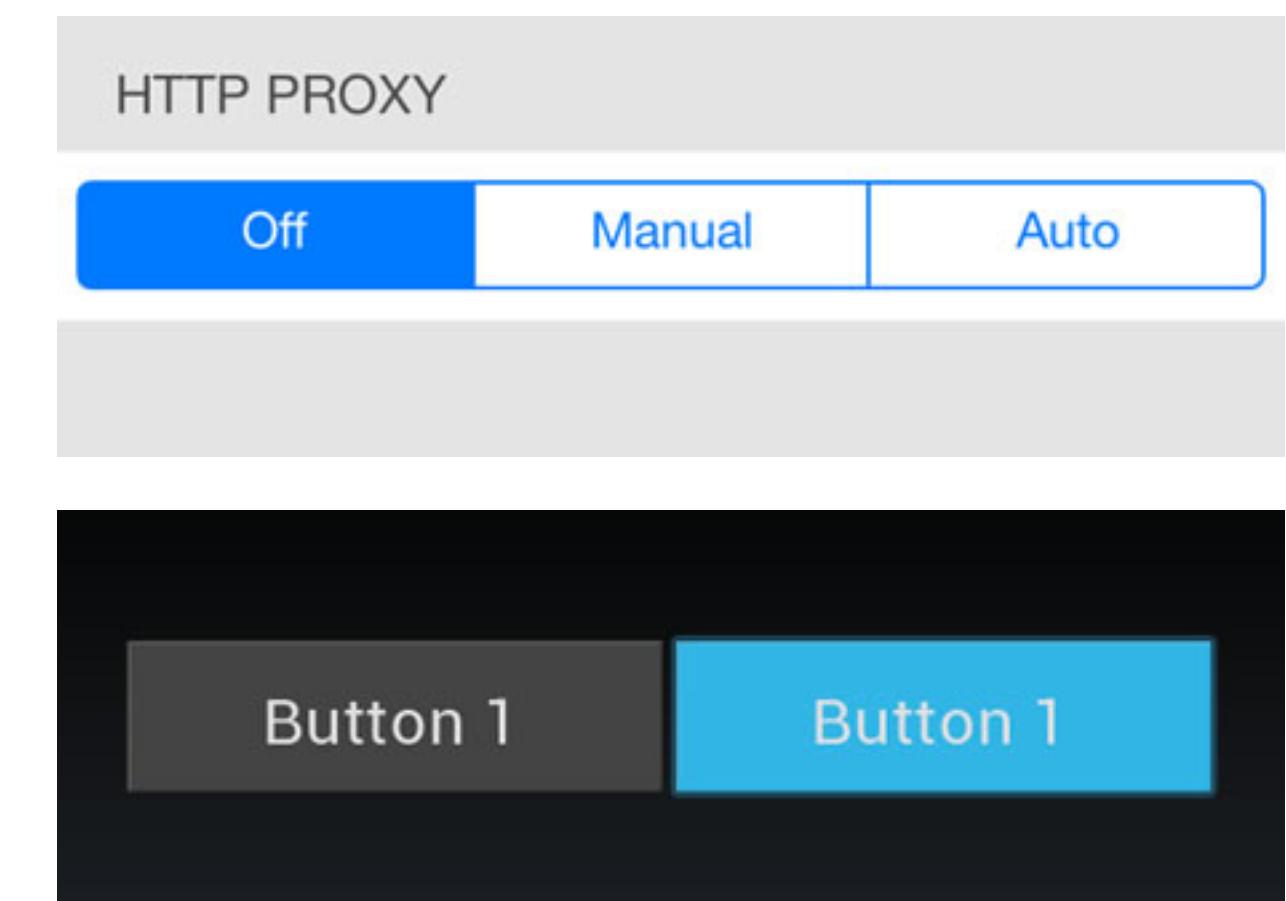
Switches

- iOS Developer Library (UISwitch): “Create and manage the On/Off buttons”
- Android API (Toggle Button): “Change a setting between two states”



Segmented controls

- iOS Developer Library (UISegmentedControl): “Horizontal control made of multiple segments, each segment functioning as a discrete button”
- *Android API (Button)*



UI Elements

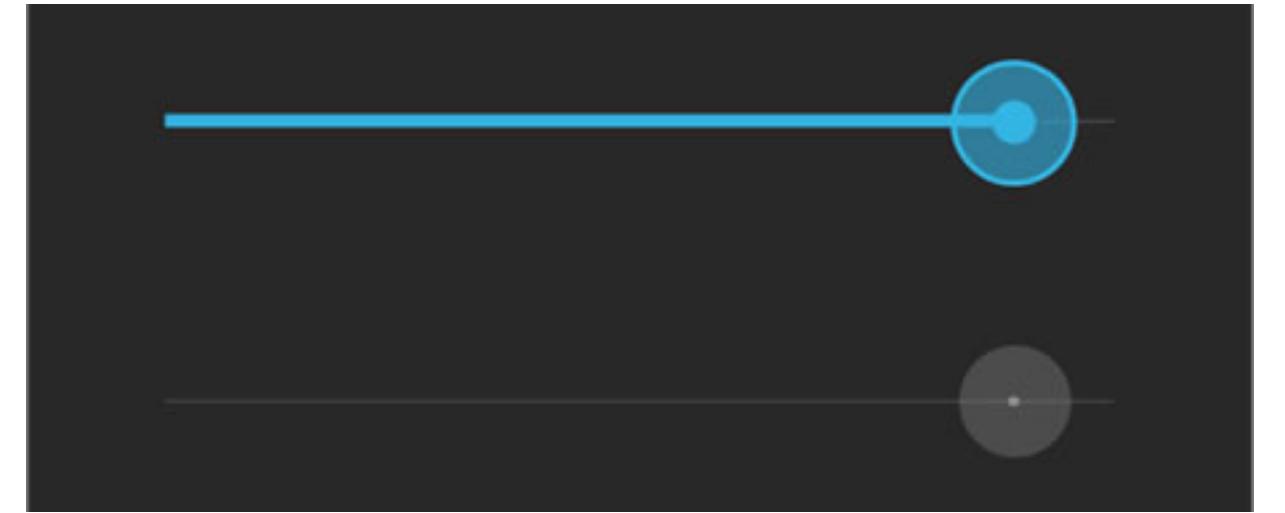
Stepper

- iOS Developer Library (UIStepper): “User interface for incrementing or decrementing a value”
- *Android API (Button)*



Slider

- iOS Developer Library (UISlider): “Select a single value from a continuous range of values”
- Android API (SeekBar): “Select a value from a continuous or discrete range of values”



UI Elements

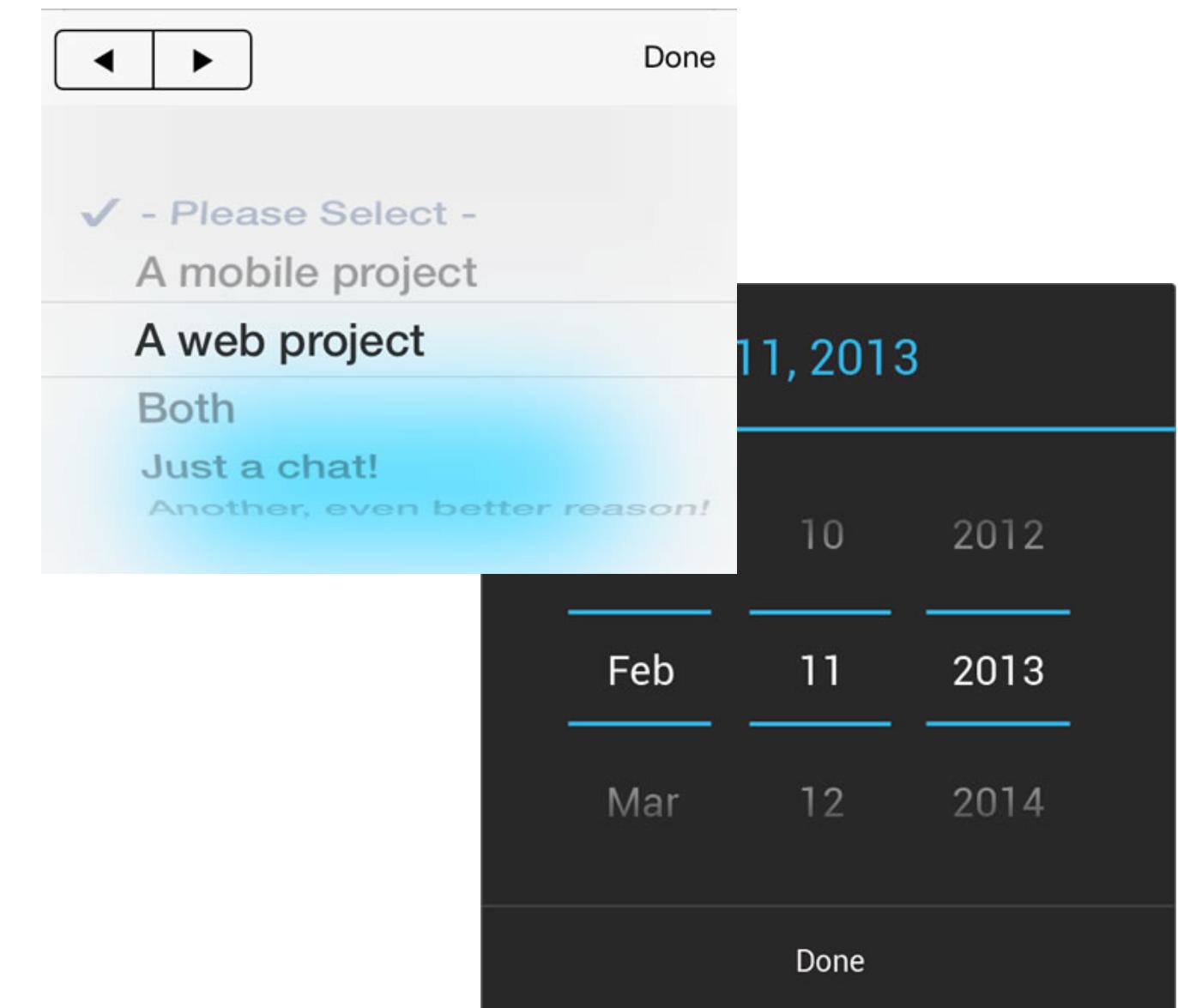
Popup menus

- iOS Developer Library (`UIViewControllerAnimated`): “Menu interface for the Cut, Copy, Paste, Select, Select All, and Delete commands”
- Android API (Popup Menu): “Modal menu anchored to a View”



Pickers

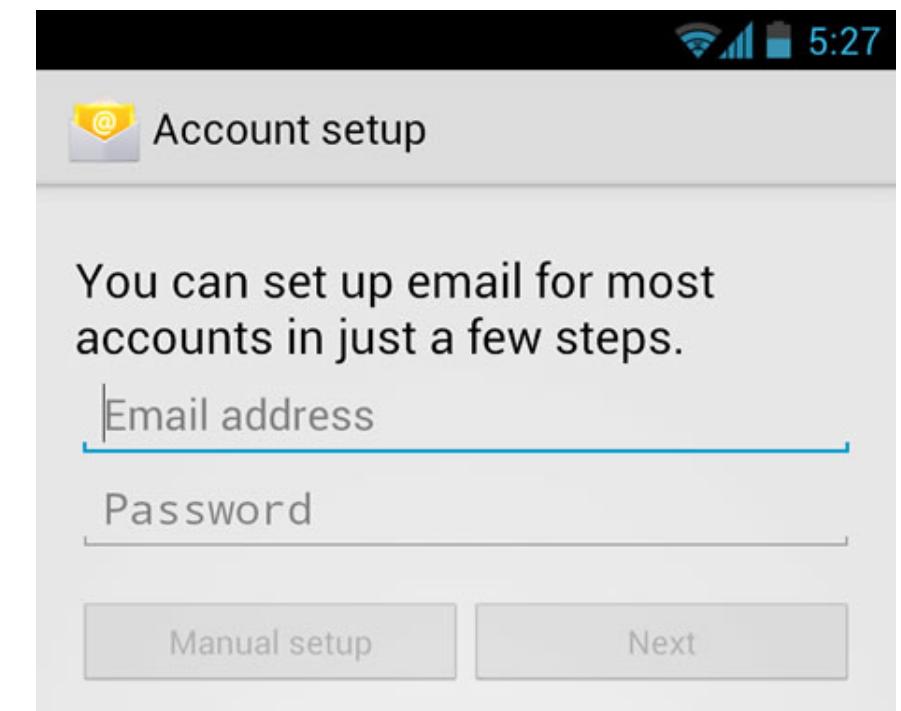
- iOS Developer Library (`UIPickerView`): “Spinning-wheel or slot-machine metaphor to show one or more sets of values”; there is also `UIDatePicker`
- Android API (Picker): “Pick a time or pick a date as ready-to-use dialogs”



UI Elements

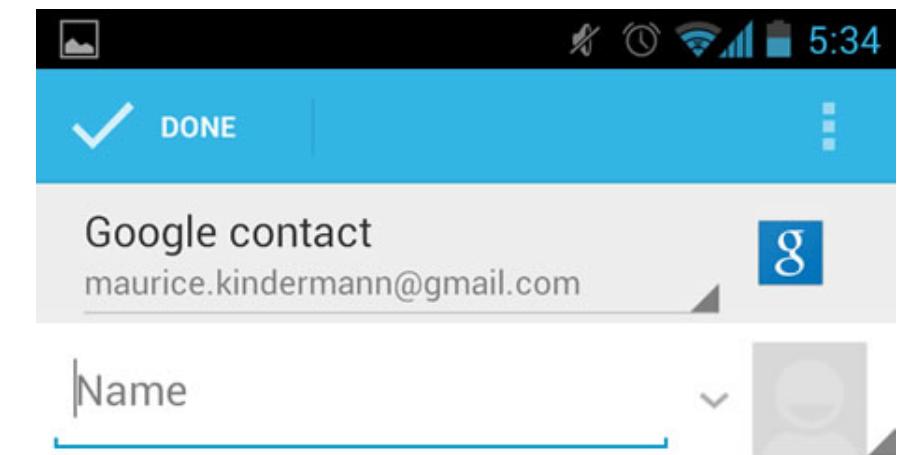
TextView

- “Displays text to the user and optionally allows them to edit it”
- Provides “a complete text editor, however the basic class is configured to not allow editing”



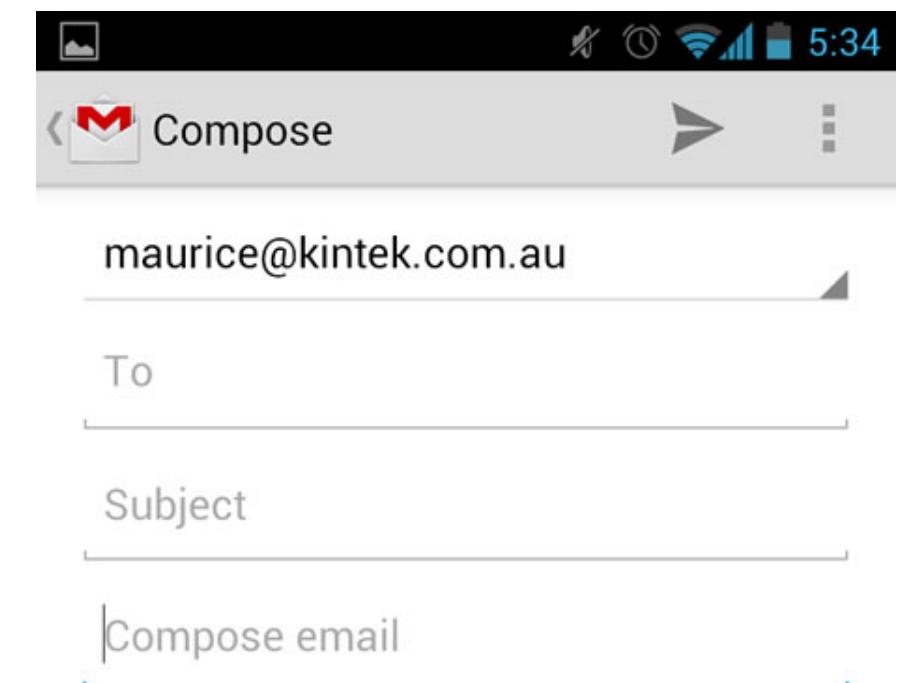
EditText

- A layer “over TextView that configures itself to be editable”

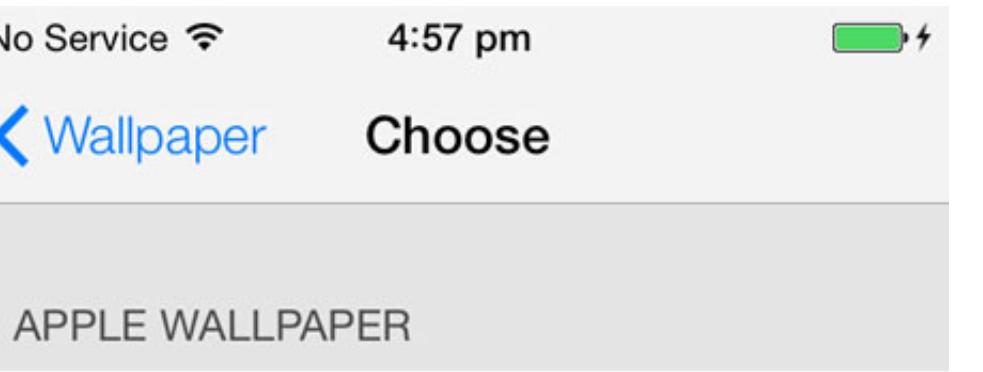


Attributes

- autoLink: “URLs and email addresses are ... converted to clickable links”
- autoText: “automatically correct ... common spelling errors”
- password: “characters are displayed as password dots instead of themselves”
- phoneNumber, capitalize, ...

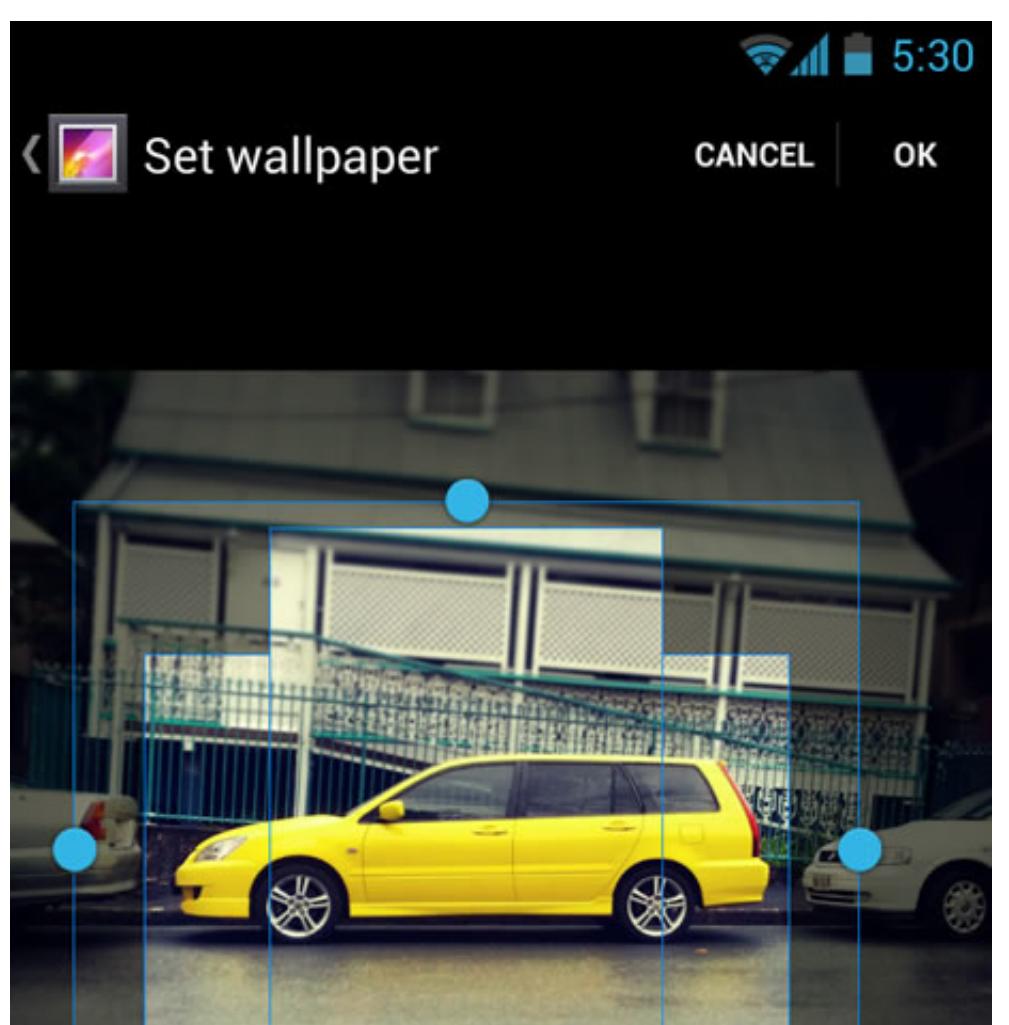


UI Elements



Images

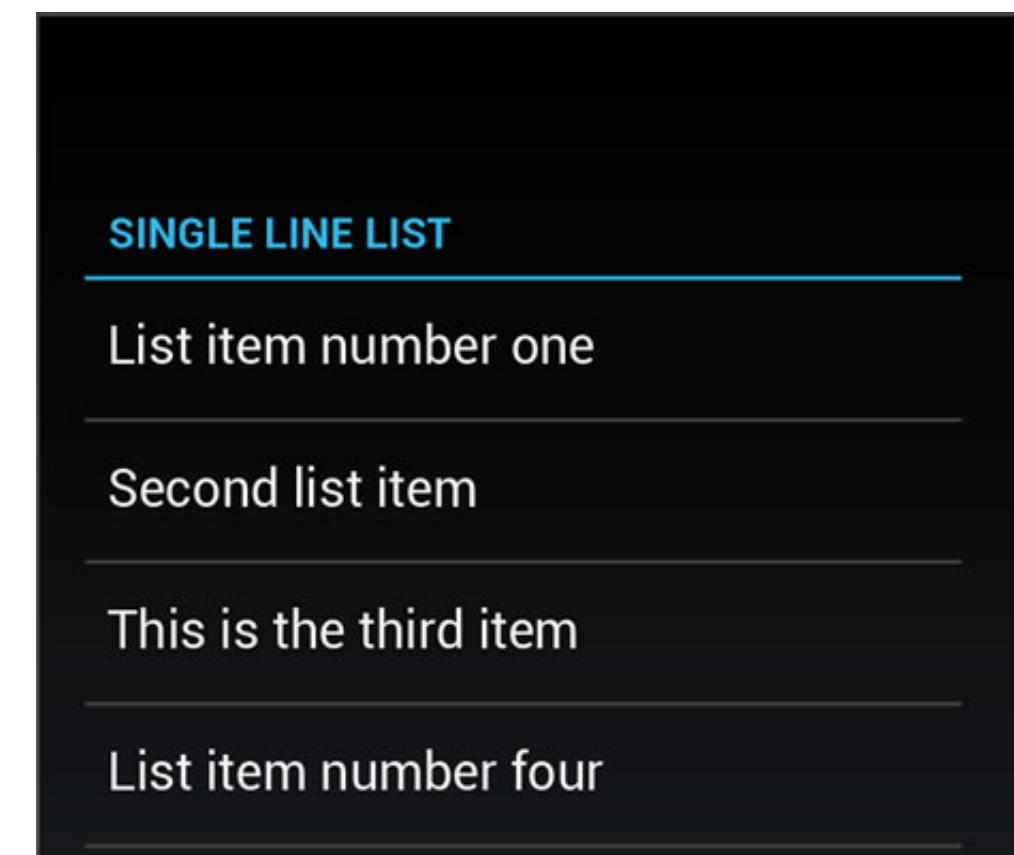
- iOS Developer Library (`UIImageView`): “a view-based container for displaying either a single image or for animating a series of images”
- Android API (`ImageView`): “displays an arbitrary image”



UI Elements

Lists

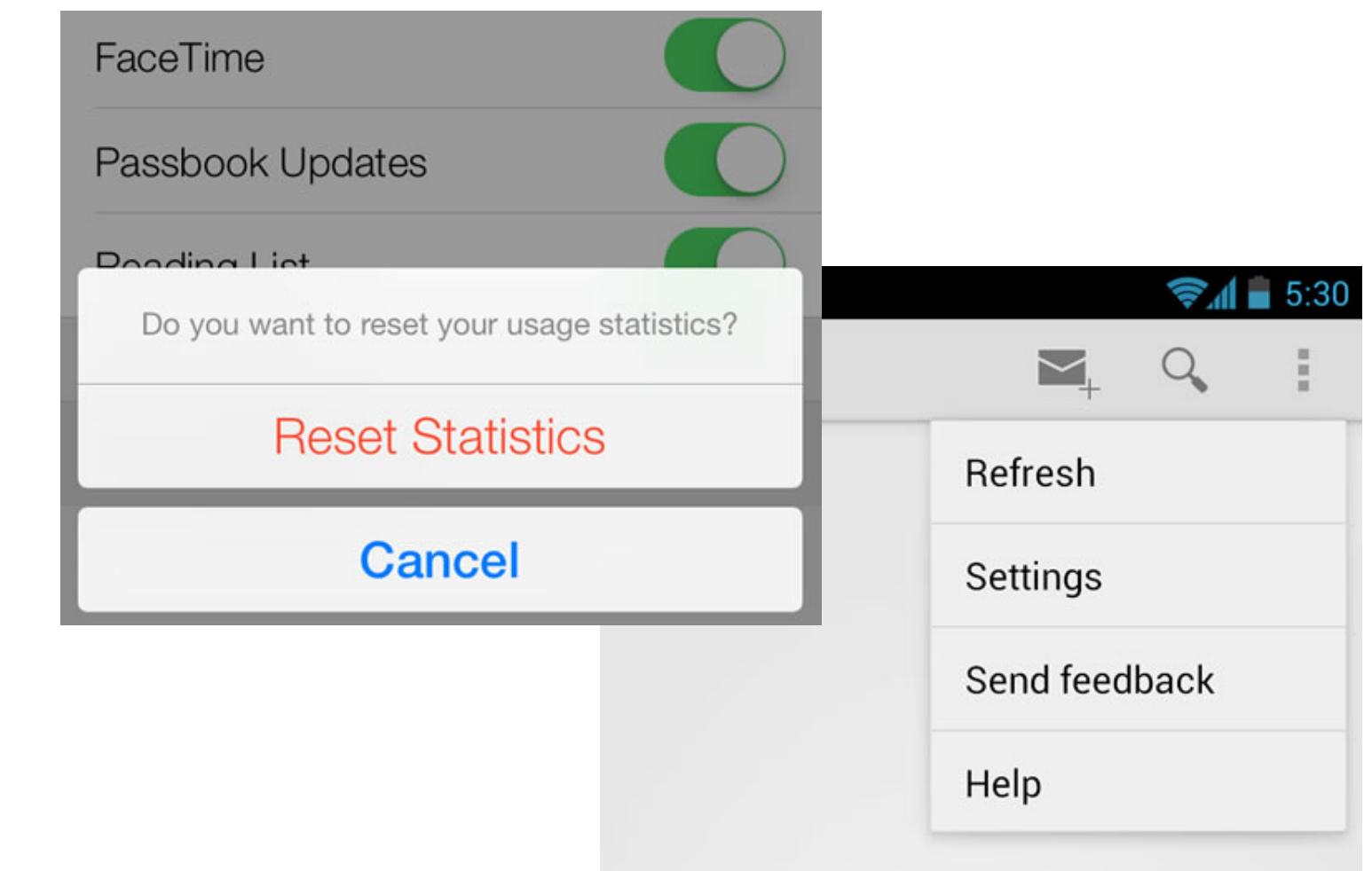
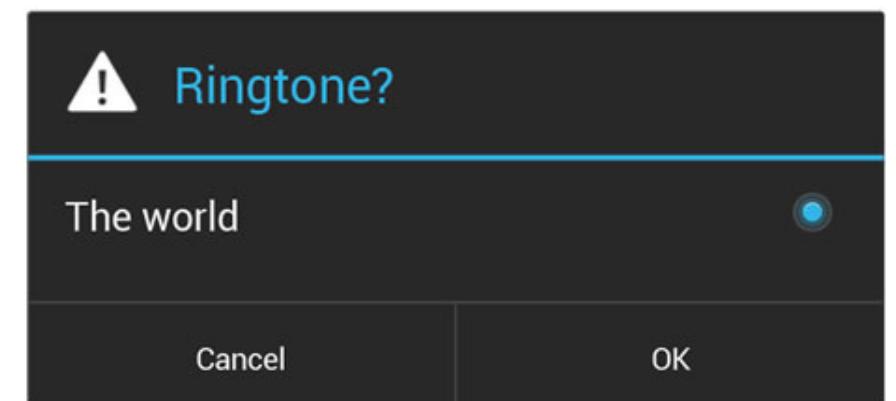
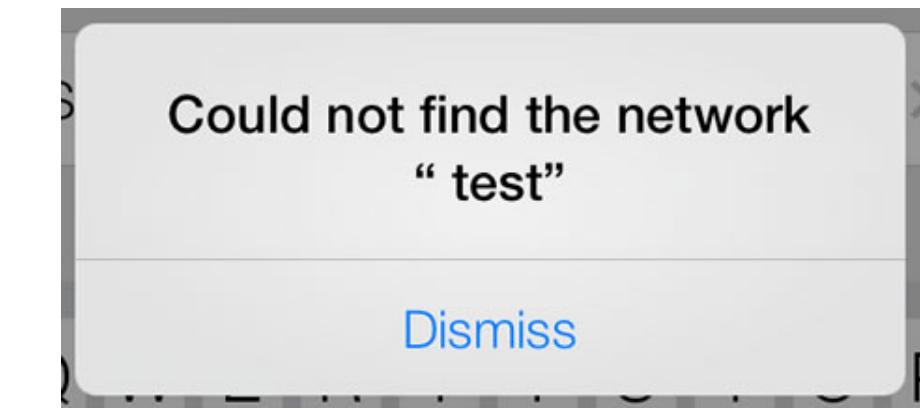
- iOS Developer Library (UITableView): “means for displaying and editing hierarchical lists of information”
- Android API (List View): “view group that displays a list of scrollable items”



UI Elements

Alerts & Dialogs

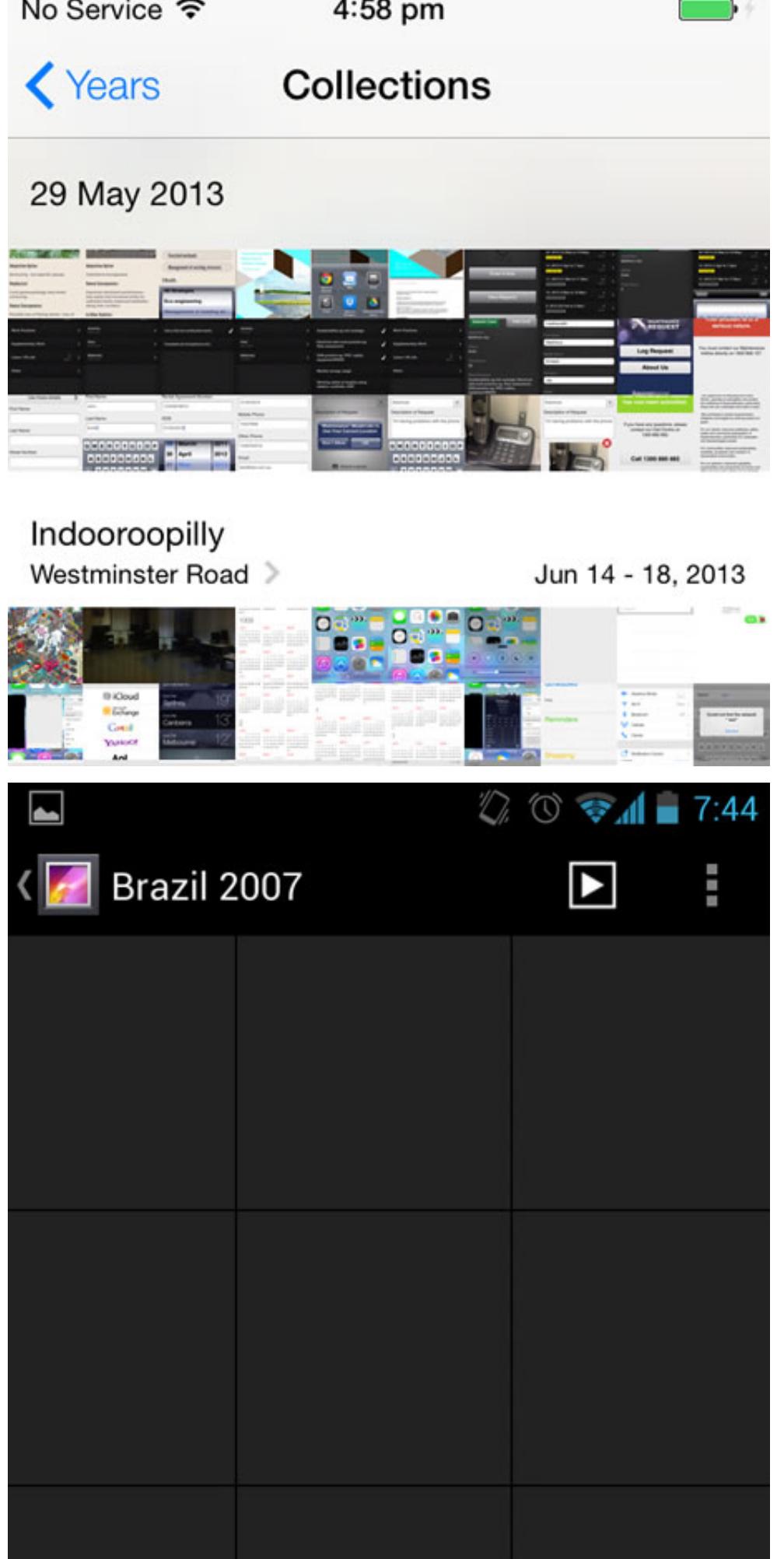
- iOS Developer Library (UIAlertView): “display an alert message to the user”
- Android API (Dialog): “small window that prompts the user to make a decision or enter additional information”
- iOS Developer Library (UIActionSheet): “set of alternatives for how to proceed with a given task”
- Android API (Spinner): “select one value from a set”



UI Elements

Collections

- iOS Developer Library (`UICollectionView`): “ordered collection of data items and presents them using customizable layouts”
- Android API (`GridView`): “shows items in [a] two-dimensional scrolling grid”



UI Elements

Scroll views

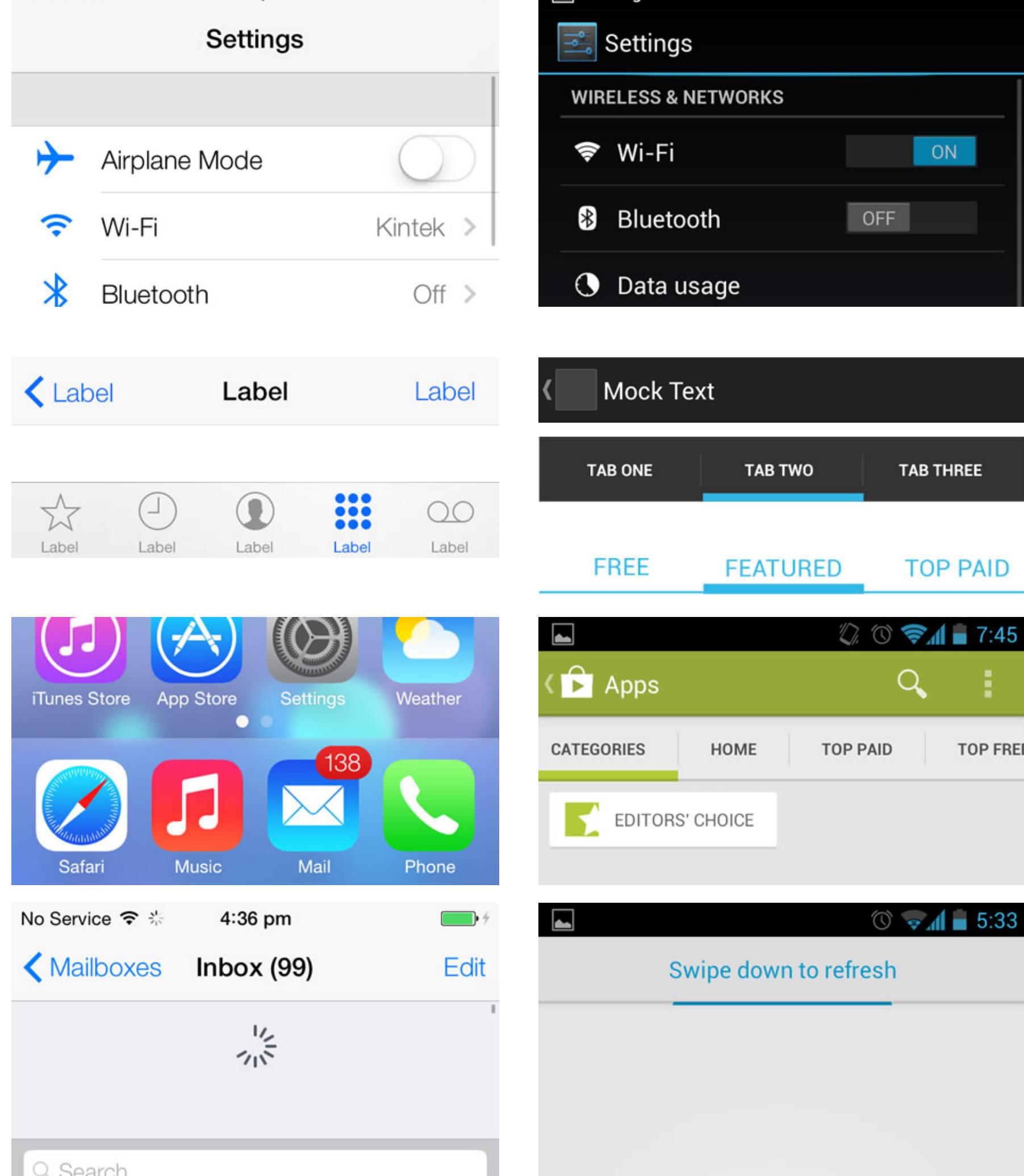
- iOS (UIScrollView) and Android (ScrollView)

Navigation

- iOS (UIBarButtonItem & UITabBar & UI Page Control) and Android (ActionBar)

Refresh

- iOS (UIRefreshControl)



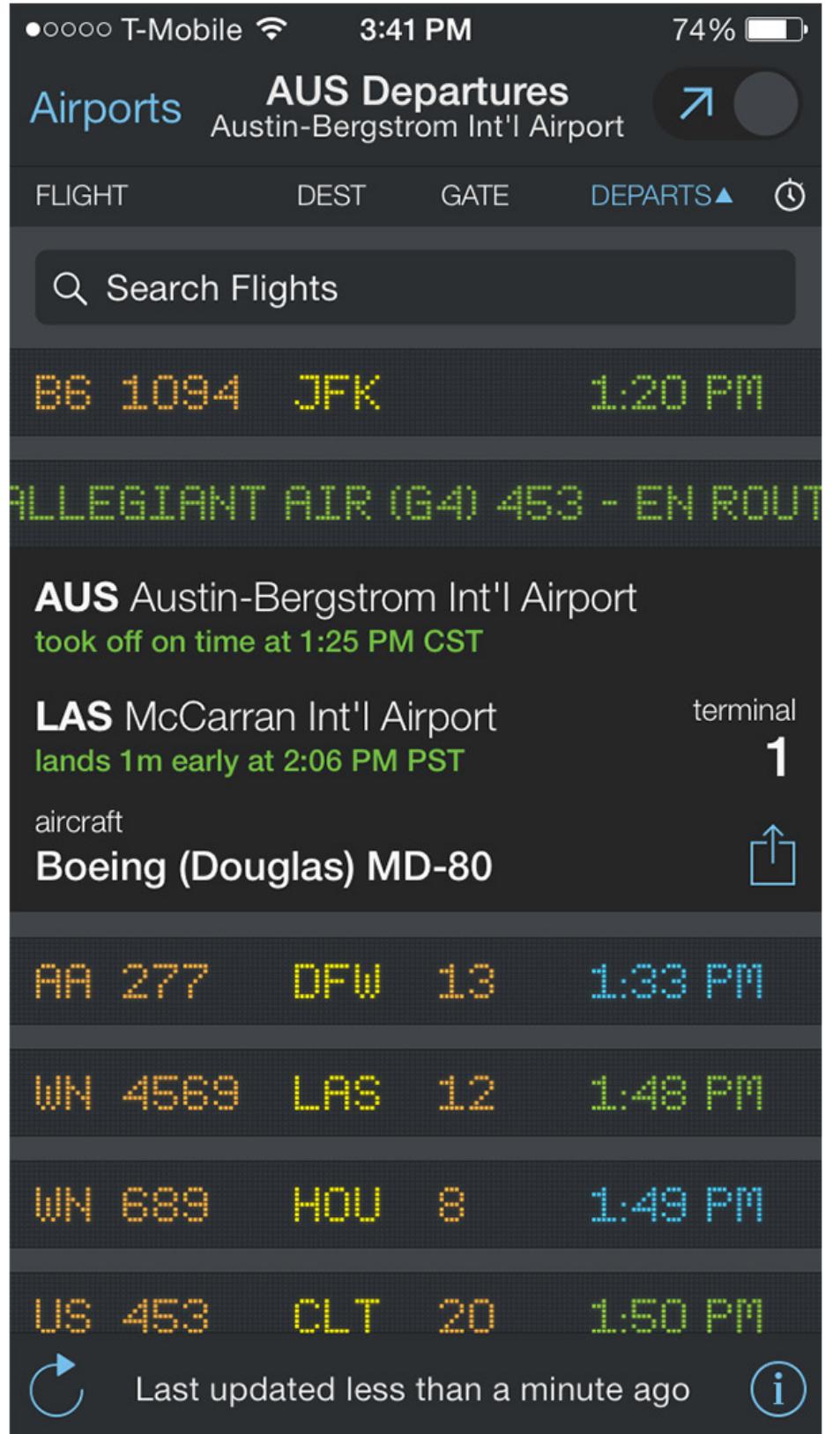
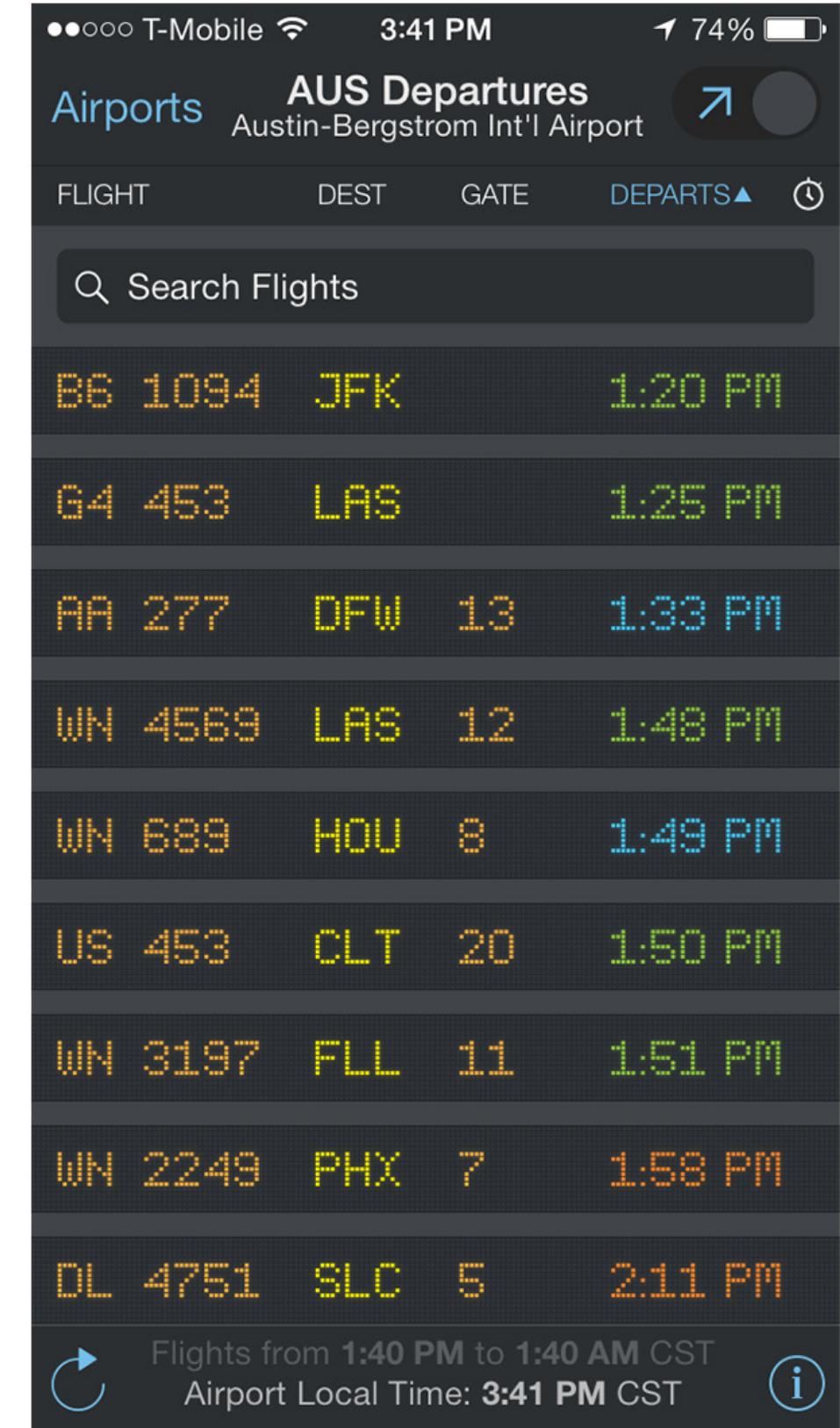
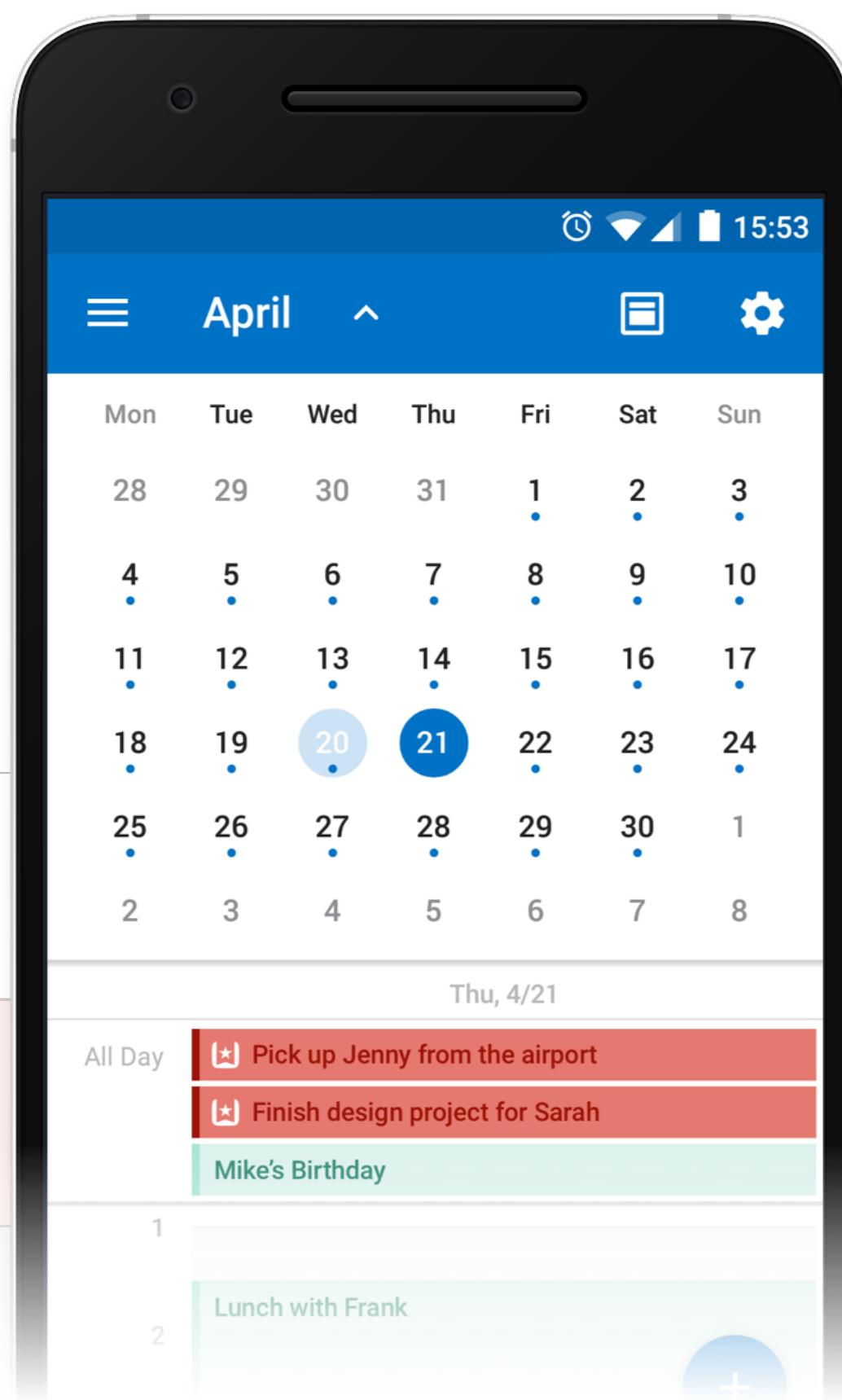
UI Elements



Please fill in this microblog.

<http://go.unimelb.edu.au/jb8r>

**Make the
unfamiliar
familiar.**

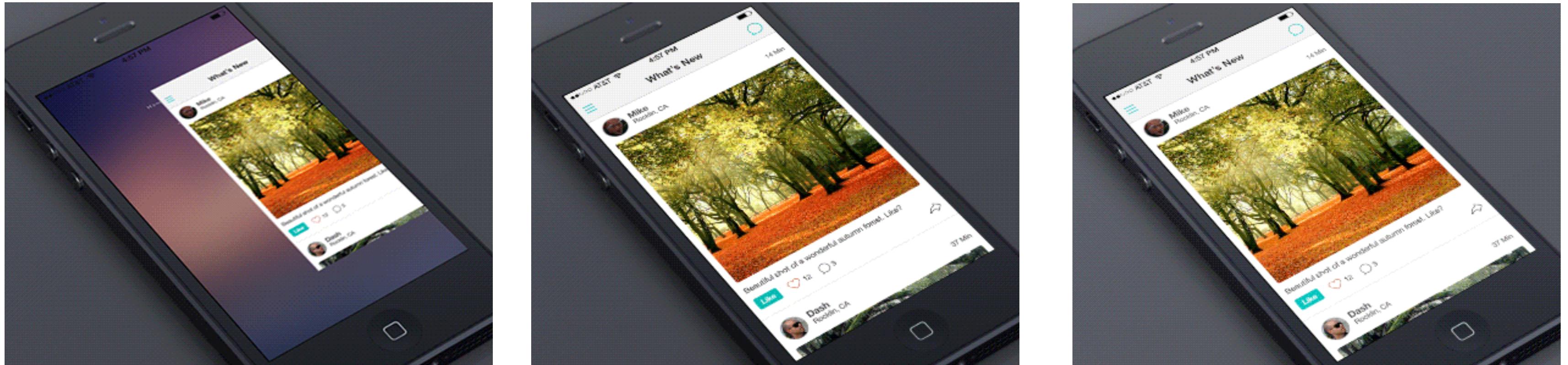


Metaphors and Skeuomorphism

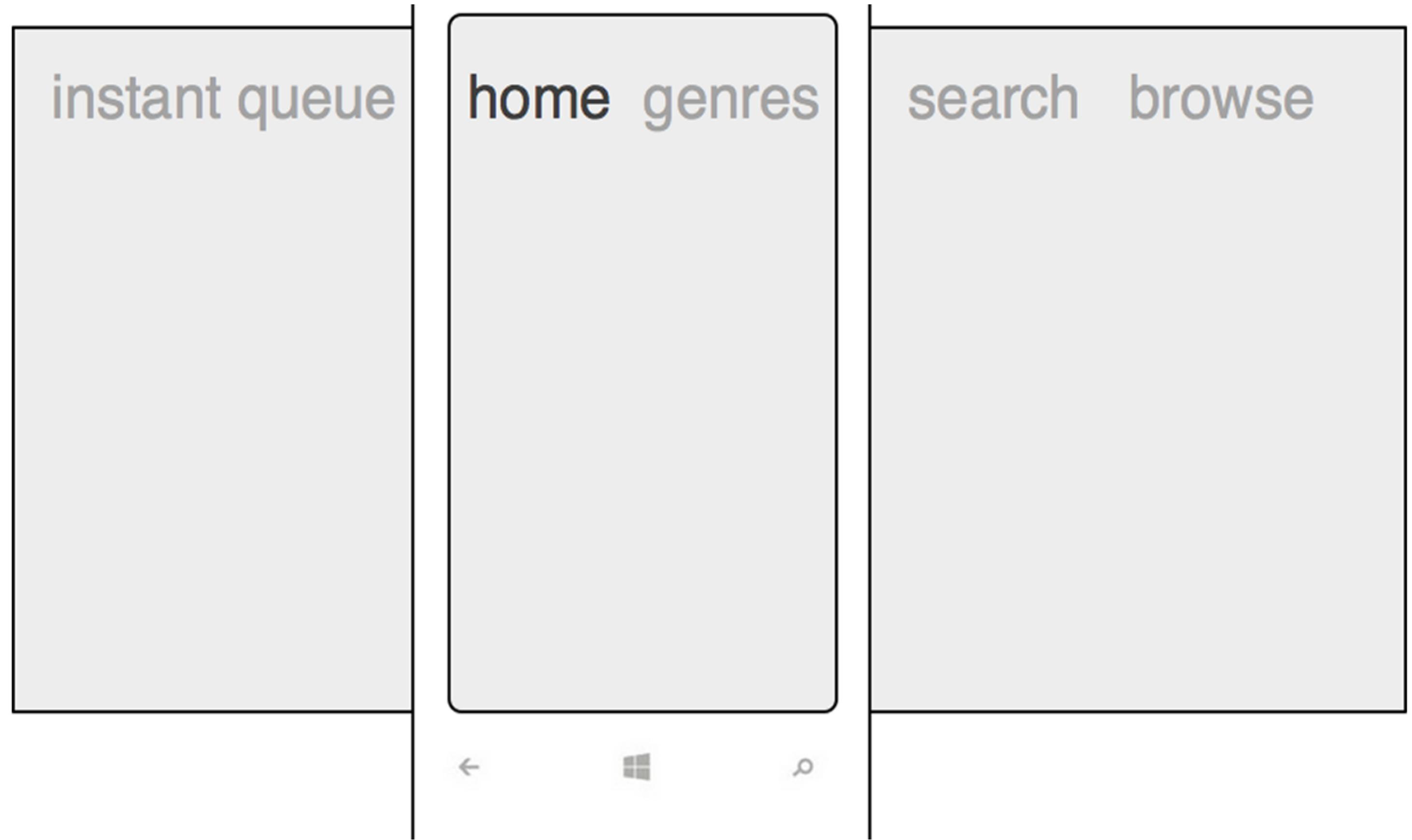
Mobile GUI

- Use side drawers

Think “Off Canvas”.

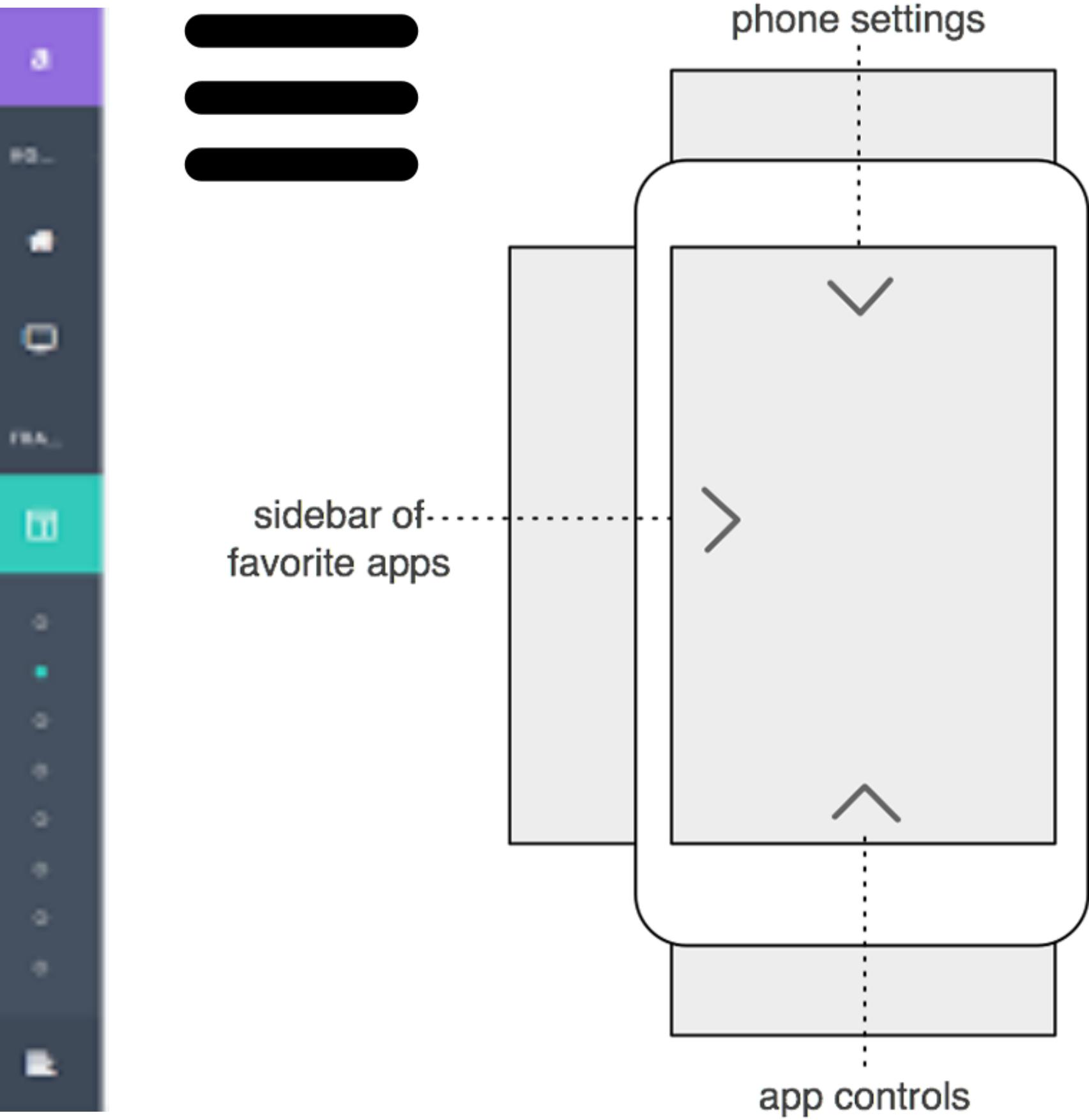


Navigation

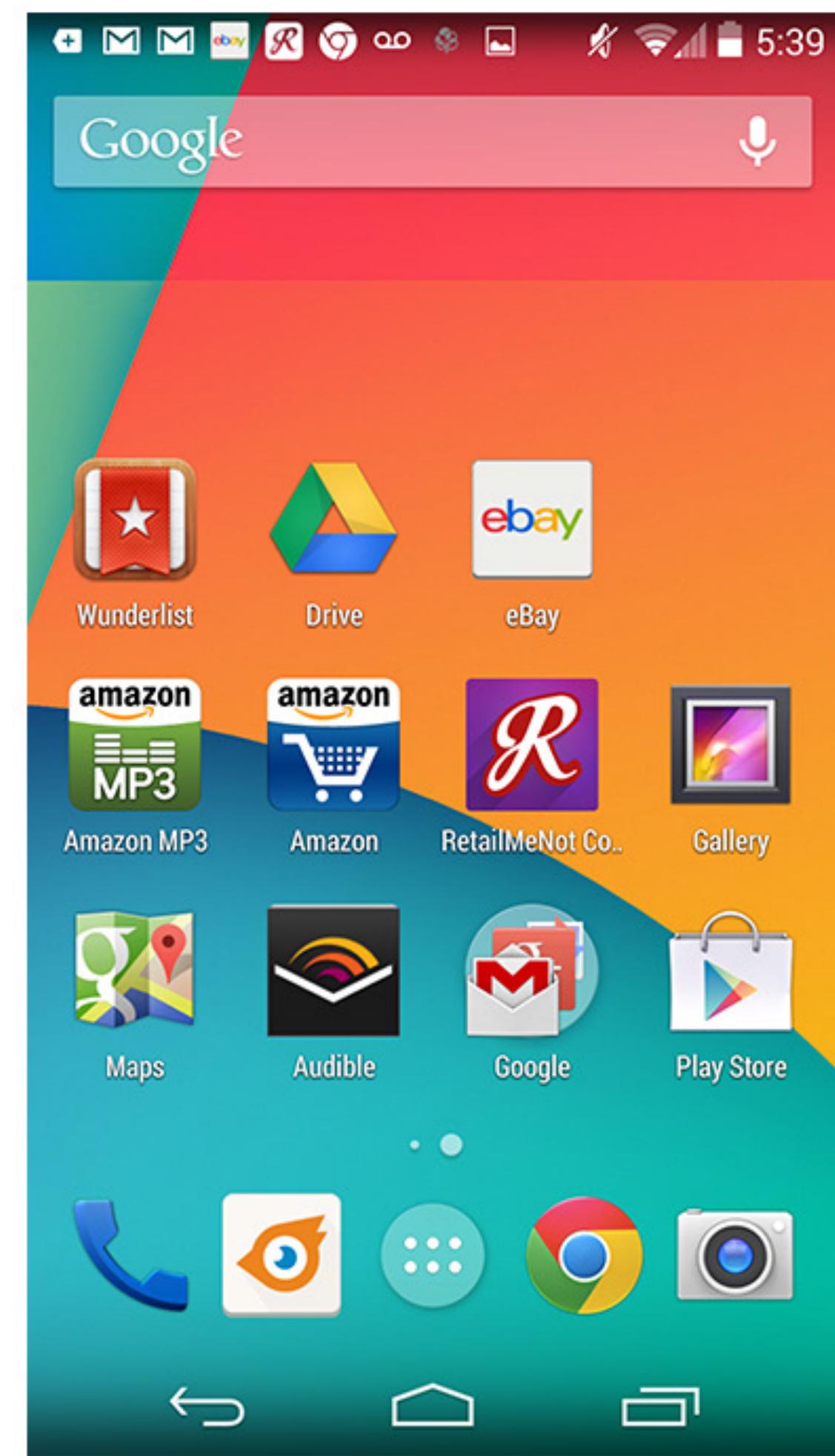


Windows Phone: Panorama Control

Leave the screen entirely for the application



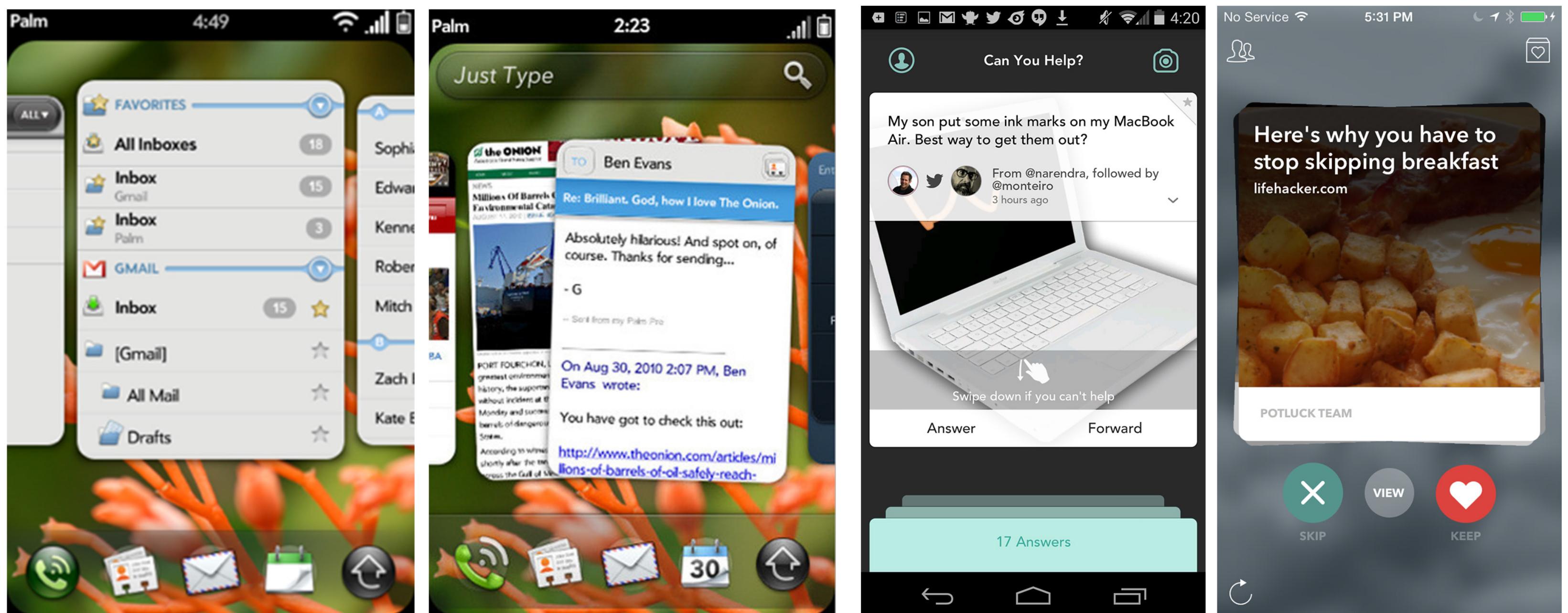
Ubuntu / Android



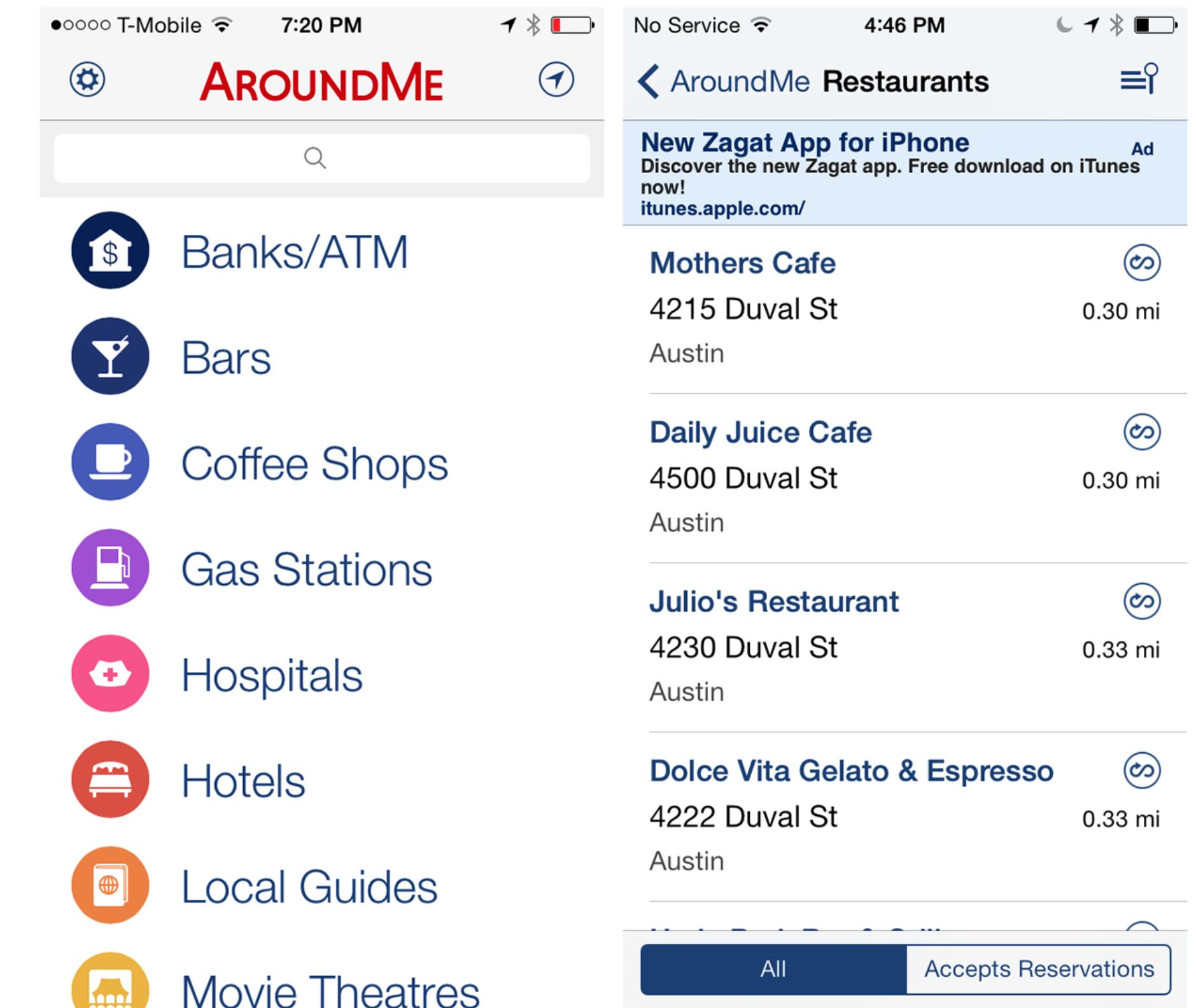
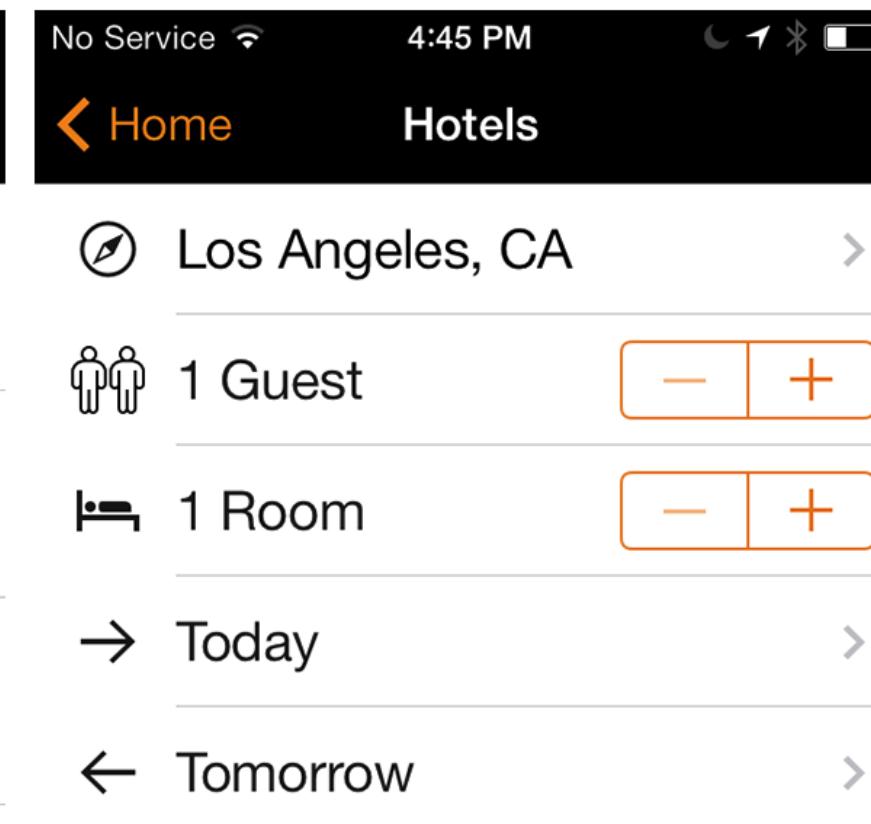
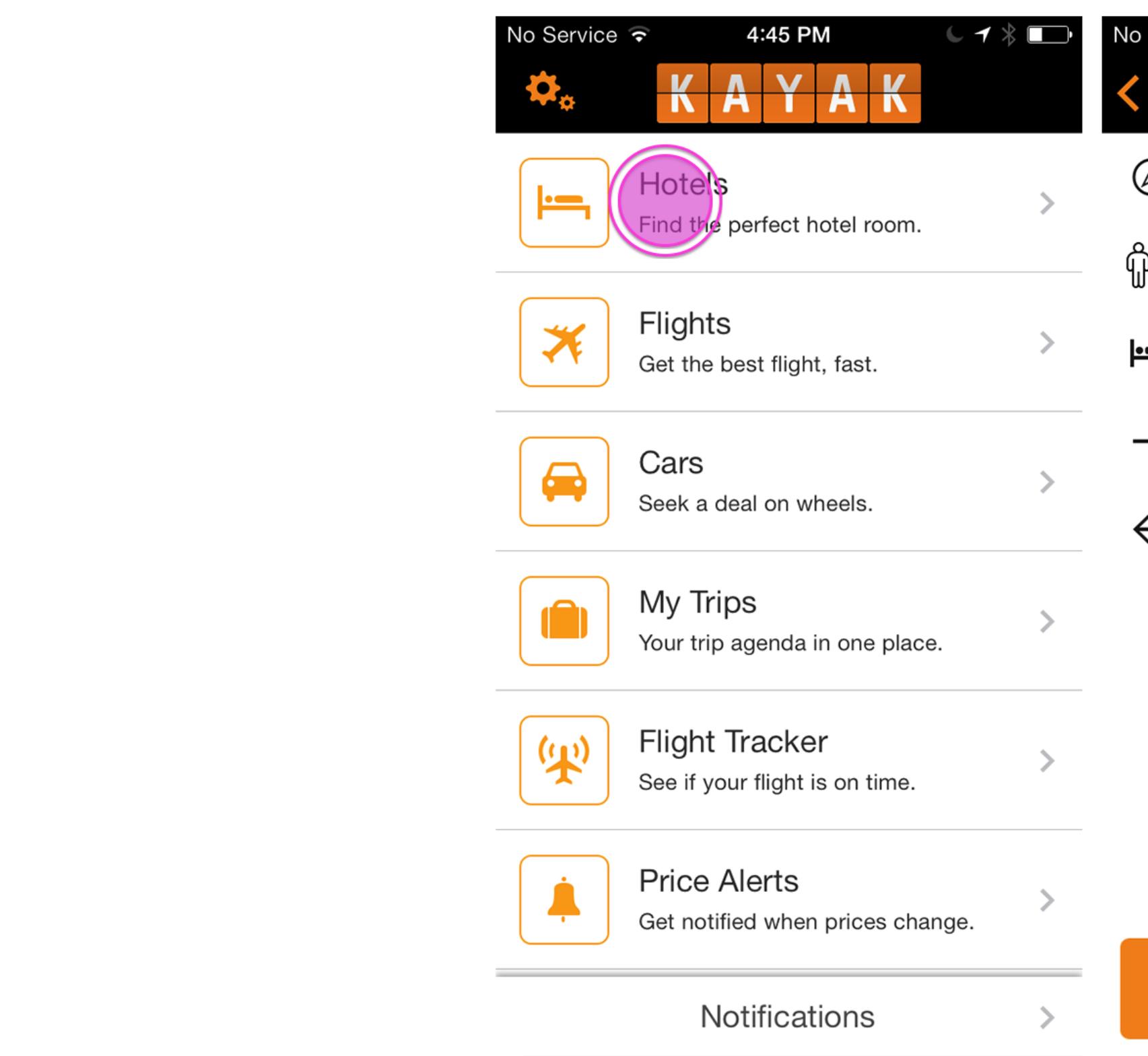
Springboards

Card deck metaphor

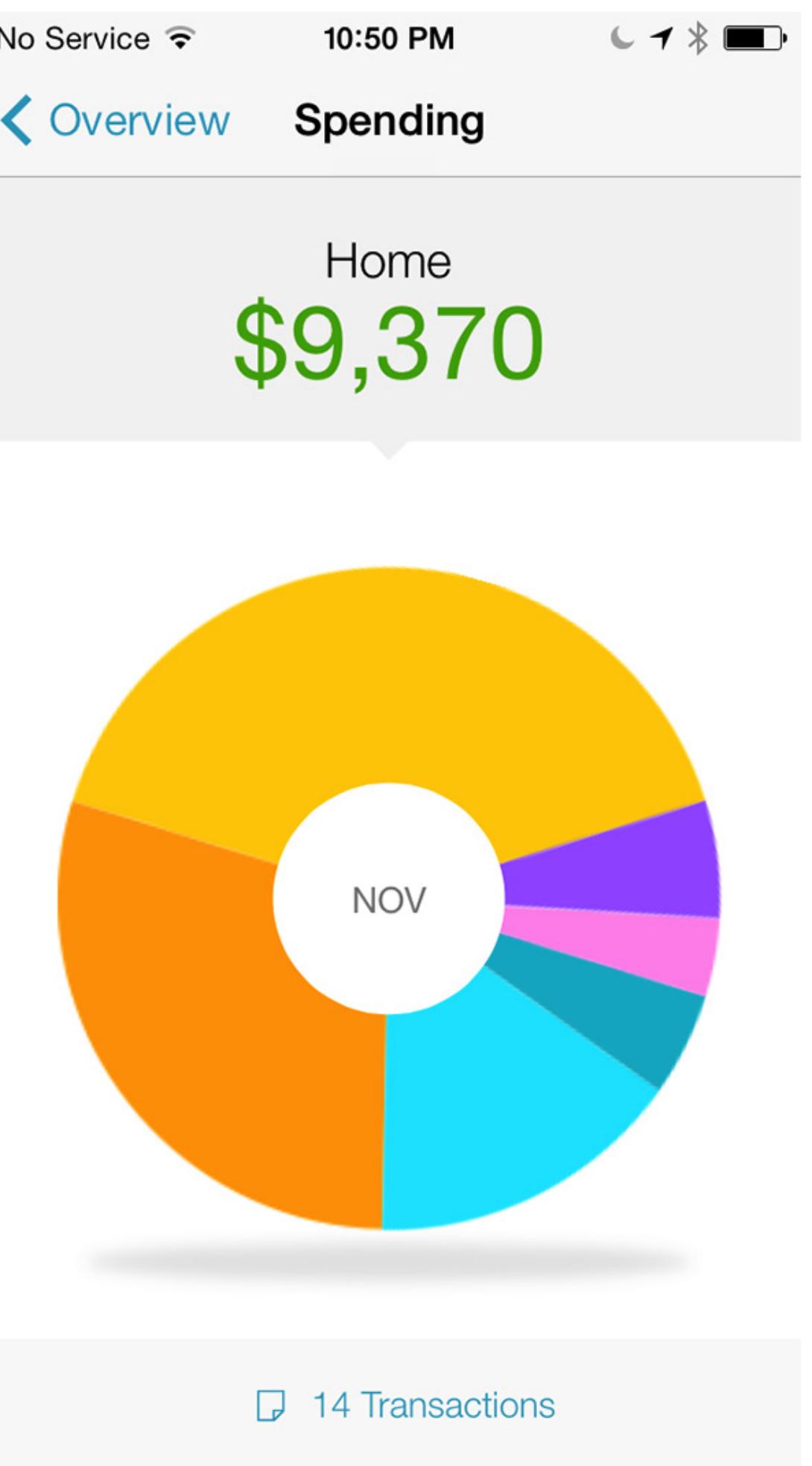
- Stack, shuffle, discard, flip
- Used in Palm webOS (2010/2011) or Jelly in iOS



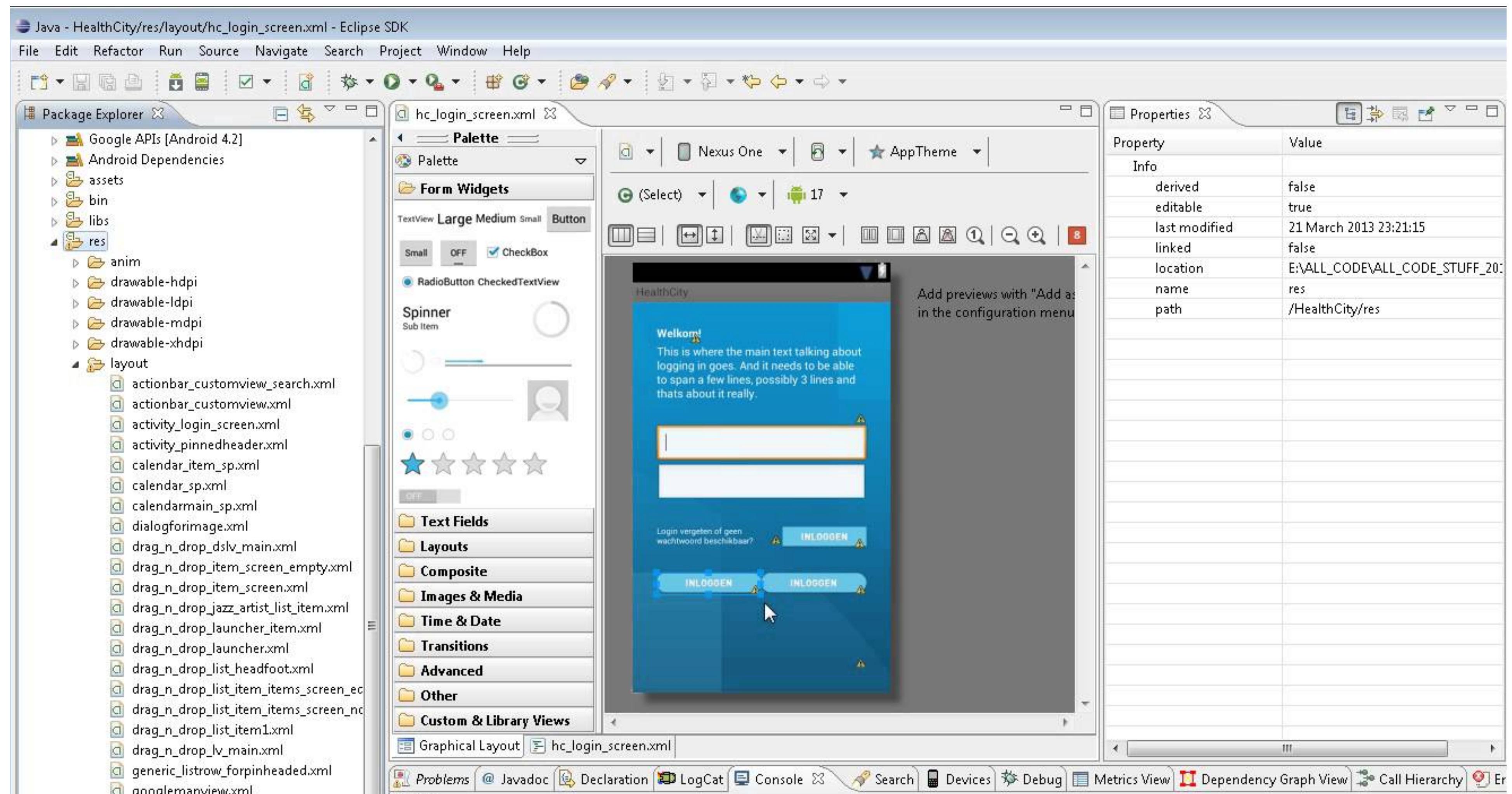
Cards



Lists



Dashboards

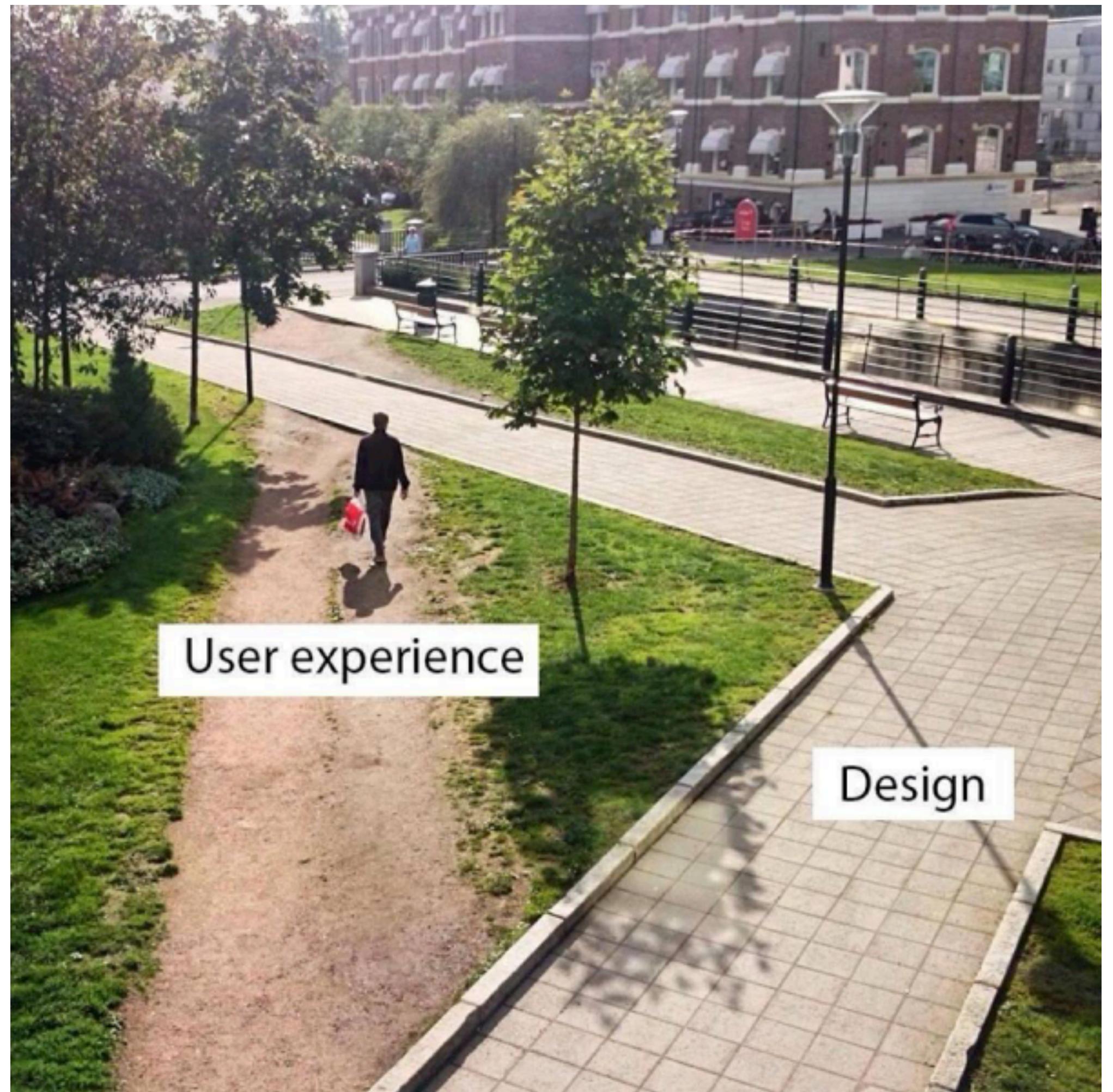


Android Interface Builder

User experience

encompasses all aspects of the end-user's interaction with the company, its services, and its products.

- Nielsen Norman Group



User Experience



User Experience

Usability

Usability is a quality attribute that assesses how easy user interfaces are to use. The word "usability" also refers to methods for improving ease-of-use during the design process.

Usability has five quality components:

- **Learnability:** How easy is it for users to accomplish basic tasks the first time they encounter the design?
- **Efficiency:** Once users have learned the design, how quickly can they perform tasks?
- **Memorability:** When users return to the design after a period of not using it, how easily can they reestablish proficiency?
- **Errors:** How many errors do users make, how severe are these errors, and how easily can they recover from the errors?
- **Satisfaction:** How pleasant is it to use the design?

1. Match between system and the real world
2. Help and documentation
3. Consistency and standards
4. User control and freedom
5. Error prevention
6. Recognition rather than recall
7. Flexibility and efficiency of use
8. Aesthetic and minimalist design
9. Help users recognize, diagnose, and recover from errors
10. Provide help and documentation



Nielsen's Usability Heuristics for User Interface Design

Meet expectations

1. Match the real world
2. Consistency & standards
3. Help & documentation

The user is the boss

4. User control & freedom
5. Visibility of system status
6. Flexibility & efficiency

Handle errors

7. Error prevention
8. Recognition, not recall
9. Error reporting, diagnosis, and recovery

Keep it simple

10. Aesthetic & minimalist design



Chunking the Heuristics

So... What do I know NOW about

User Interfaces?

1. Challenges when designing for mobile devices

2. Mobile devices and user input

3. User interface concepts and elements

4. Aspects of the *User Experience*

5. Getting the design right is inherently hard

Summary