

*Corso di Laurea Magistrale in Design, Comunicazione
Visiva e Multimediale - Sapienza Università di Roma*

Interaction Design

A.A. 2017/2018

Mobile Design Patterns

Andrea Marrella, Francesco Leotta

Last update : 16/3/2018

Good vs Bad Mobile Design

- ▶ An interesting and quick way to better understand how people are using apps is to *read reviews in mobile marketplaces*.
- ▶ In general, most 4- and 5-star reviews are not very specific.
- ▶ But the **1- and 2-star reviews are much more telling**; they tend to offer a truer picture of problems users are having with applications.
- ▶ Most common complaints are:

- ▶ Crashing
- ▶ Lack of key features (e.g., syncing, filtering, account linking)
- ▶ Confusing interface design
- ▶ Poor navigation (e.g., can't go back, can't find things)

To be fixed, they require user and device testing.

They can be fixed by following traditional mobile design patterns.

Design Patterns

- ▶ One way to approach the design of user interfaces is to **learn from examples** that have proven to be **successful** in the past.
- ▶ **Design Patterns** are **solutions** to a **recurrent problem** within a **specific application domain**.
 - ▶ They allow to *capture* and *reuse* the *knowledge* of what made a system – or paradigm – successful and apply it again in new situations.
- ▶ Mobile Design Patterns:
 - ▶ Navigation
 - ▶ Tables
 - ▶ Forms
 - ▶ Search
 - ▶ Tools
 - ▶ Helps and Tutorials
 - ▶ Feedback & Affordance

Balsamiq Mockups

Download Balsamiq Mockups from the following link:

- <https://balsamiq.com/download/>
- Install the “trial” version of the software.
- Send an email to leotta@dis.uniroma1.it for the license number



Roadmap

- ▶ **Navigation**
- ▶ Forms
- ▶ Search
- ▶ Tools
- ▶ Invitations
- ▶ Feedback & Affordance

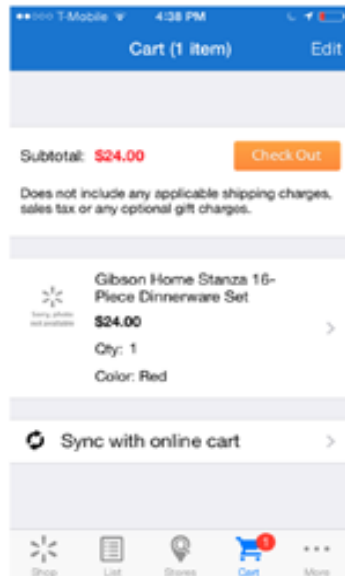
Navigation

- ▶ Navigation is about how users move through the views.
- ▶ Good navigation, like good design, is **invisible**.
 - ▶ Applications with good navigation just feel simple and make it easy to accomplish any task, from browsing through the web to post a comment on Facebook.
- ▶ **Primary Navigation patterns:** navigation from one primary category to another (as with the top-level menus of a desktop application).
- ▶ **Secondary Navigation patterns:** moving and navigating within a selected module.

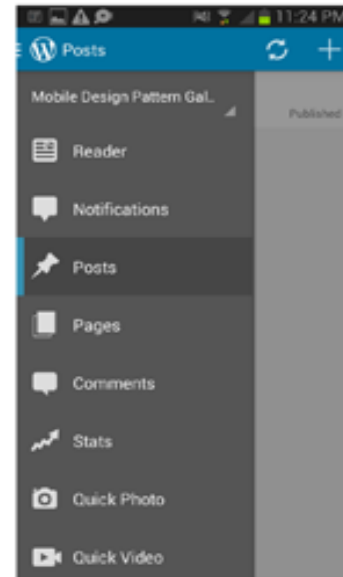
Navigation Patterns

- ▶ Two distinct types of primary navigation: **persistent** and **transient**.
- ▶ **Persistent navigation** encompasses interactive navigation components (for example, list menus) that are permanently visible.
- ▶ **Transient navigation** must be explicitly revealed with a tap or gesture.
 - ▶ These patterns arise from the constraints of smartphone screen sizes, which have pushed mobile designers to think “outside of the box”.

Walmart app (iOS)
Tab Bar for **persistent**
primary navigation.



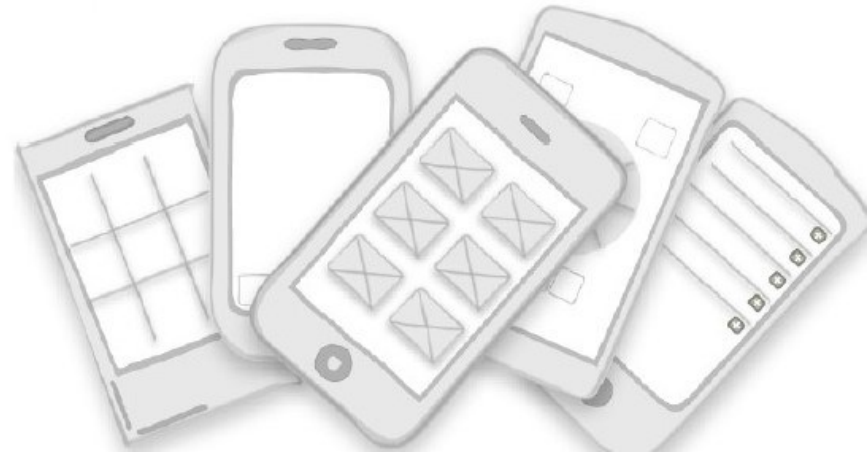
WordPress app (Android & iOS)
Transient navigation
through Side Drawer.



Primary Navigation

▶ Persistent Patterns:

- ▶ Springboard
- ▶ List Menu
- ▶ Expanding Lists
- ▶ Cards
- ▶ Gallery (Grid)
- ▶ Tabs
- ▶ Metaphor
- ▶ Infinite Area



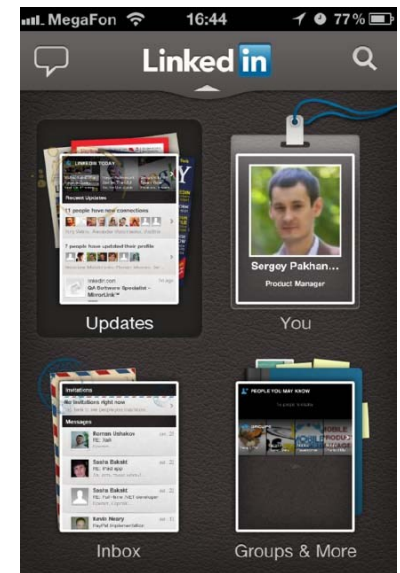
Springboard

- ▶ Springboard is a landing screen with options that act as launch points into the application.
 - ▶ Often used as navigation patterns at OS level (Windows Phone, iOS and Android).
- ▶ Use a **grid layout** for items of **equal importance**, or an **irregular layout** to **emphasize some items** more than others.



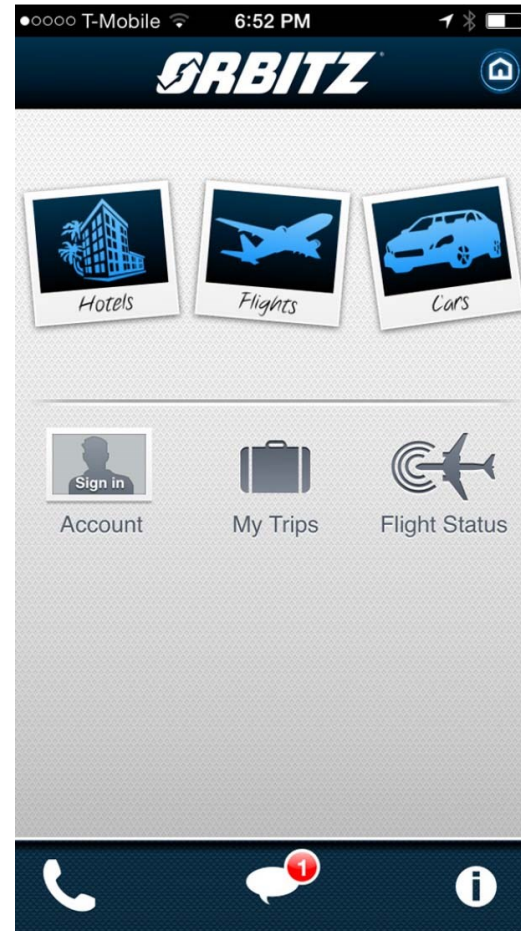
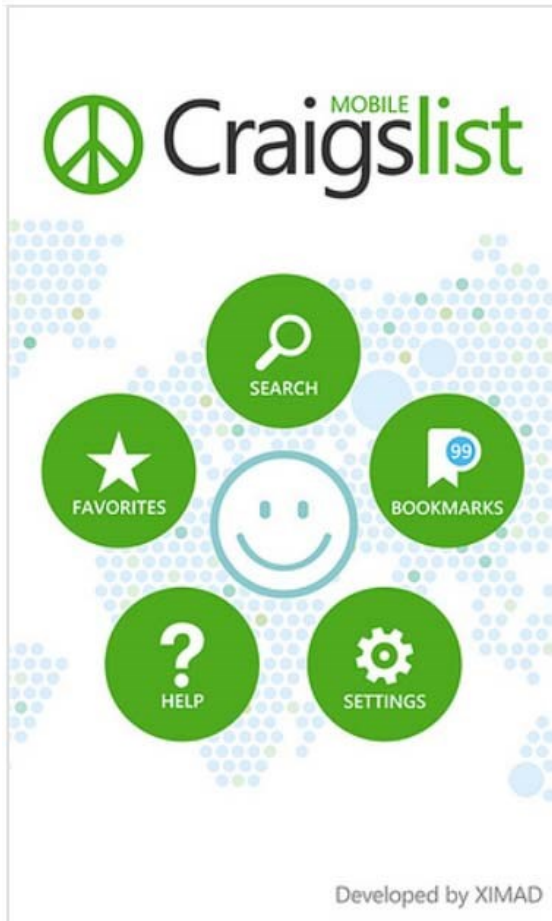
Home page of iOS phones.
Example of traditional
springboard

Springboard with irregular
layout to emphasize importance
(Windows phone) or hierarchy
(LinkedIn for iOS).



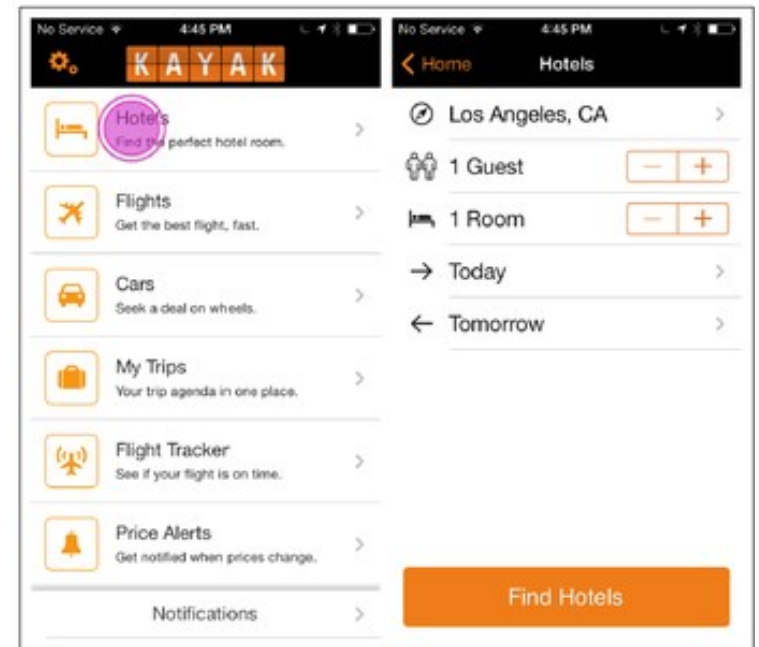
Customized Springboard

with irregular components



List Menu

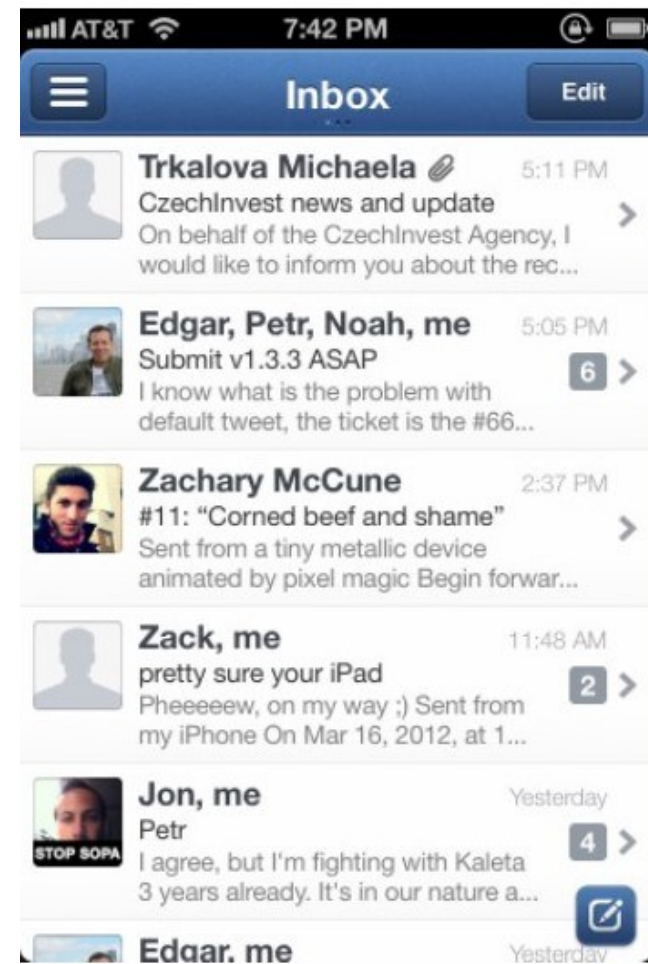
- ▶ Each list item is a launch point into the application, and switching modules requires navigating back to the list.
 - ▶ Users navigate by making **one choice per screen** until they reach their destination.
 - ▶ To navigate to another destination, users must **retrace** some of their steps (or **start over** from the beginning) and make different choices.
 - ▶ All internal screens should have a **way for returning back**. This is a kind of **hierarchical navigation**.
 - ▶ If a screen is the topmost one in an app (that is, the app's home), it should not present a 'back' button.



Works well for long (potentially infinite) lists with subtext [Kayak for iOS].

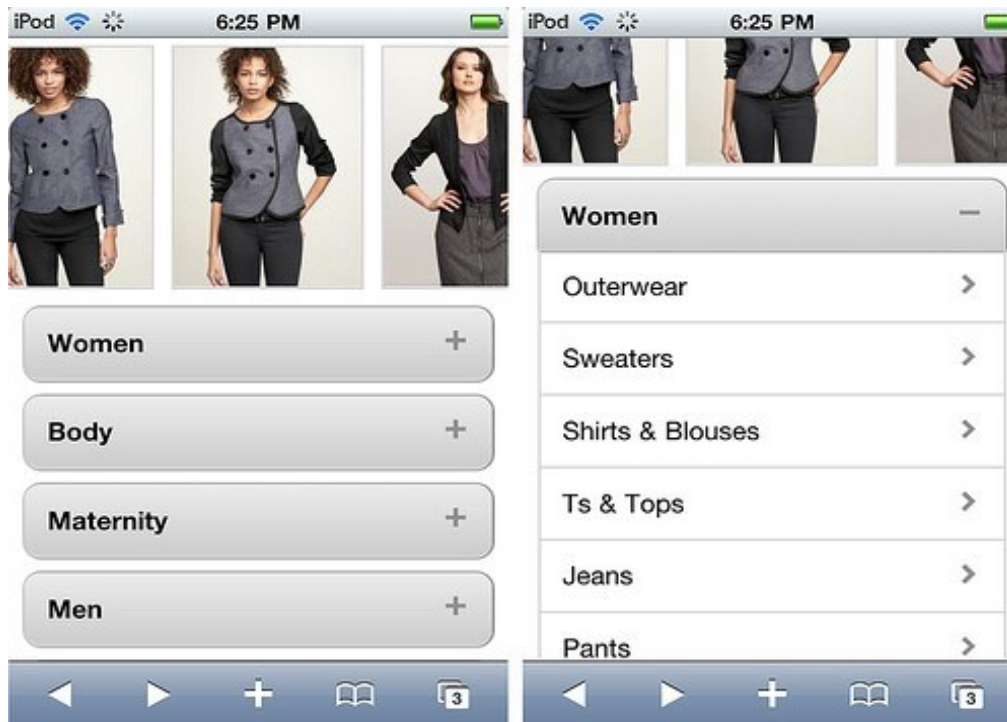
Customized List Menu

with thumbs and hierarchy



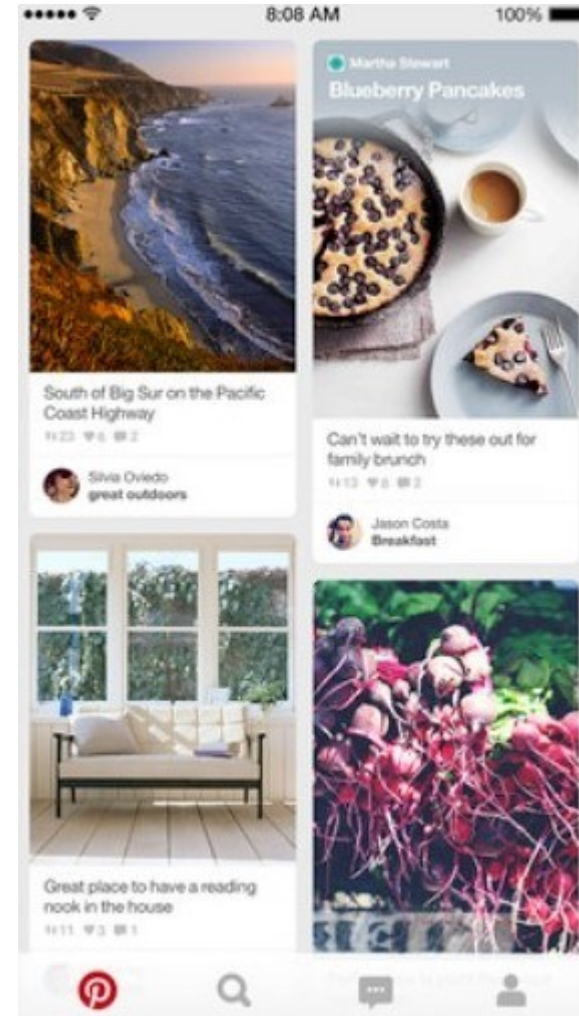
Expanding Lists

- ▶ Works best for progressively disclosing more details for an object.
 - ▶ Often used when a mobile app has been ported from an existing desktop's one.

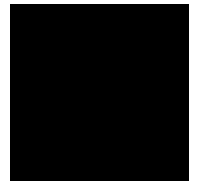
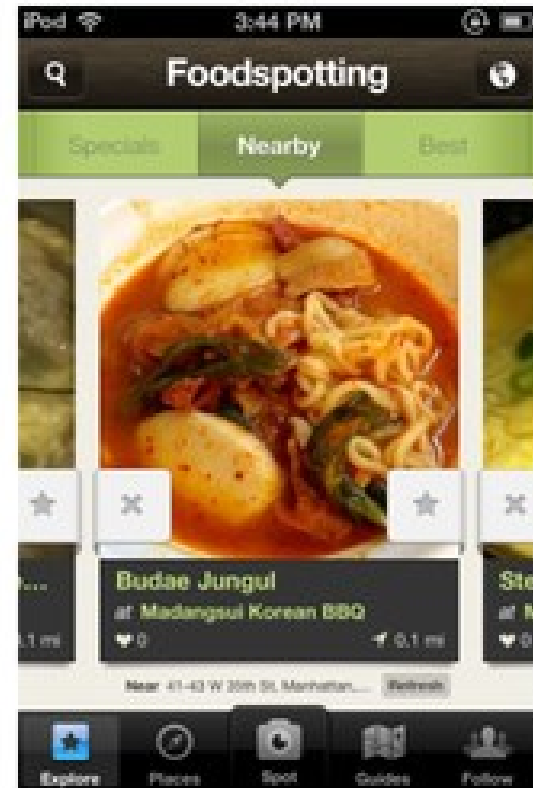
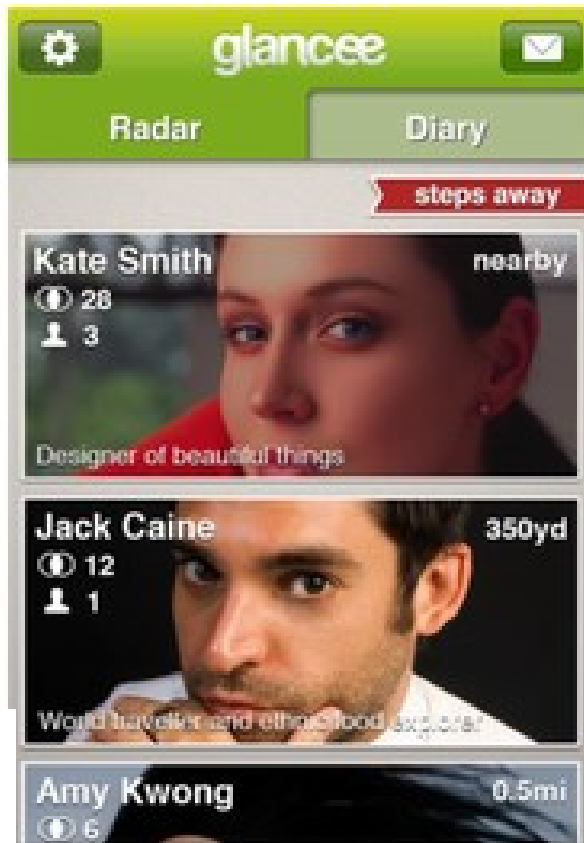


Cards

- ▶ **Cards** are small containers that logically encapsulates relevant information.
- ▶ Card navigation is based on a **card deck metaphor**, often including common card deck manipulations such as stacking, shuffling, discarding, and flipping.
- ▶ Good for presenting similar objects whose size or supported actions can vary considerably.



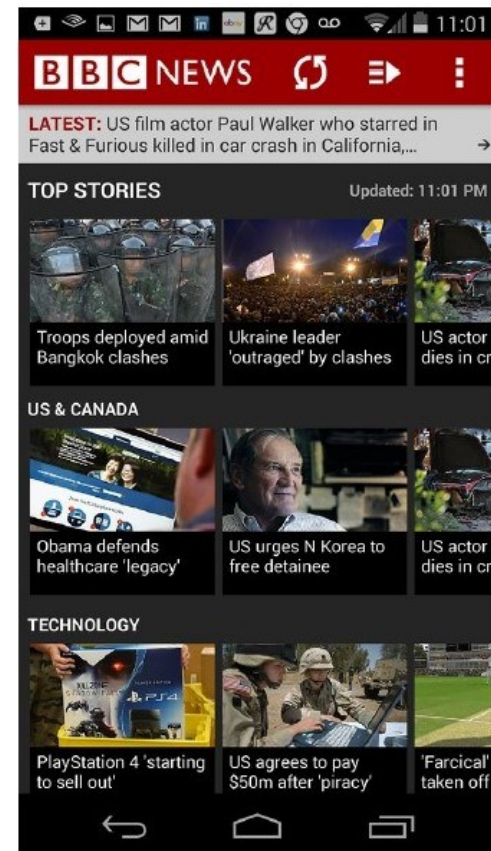
Example of Cards



Gallery

It works best for showing frequently updated, highly visual content where **no hierarchy** is implied.

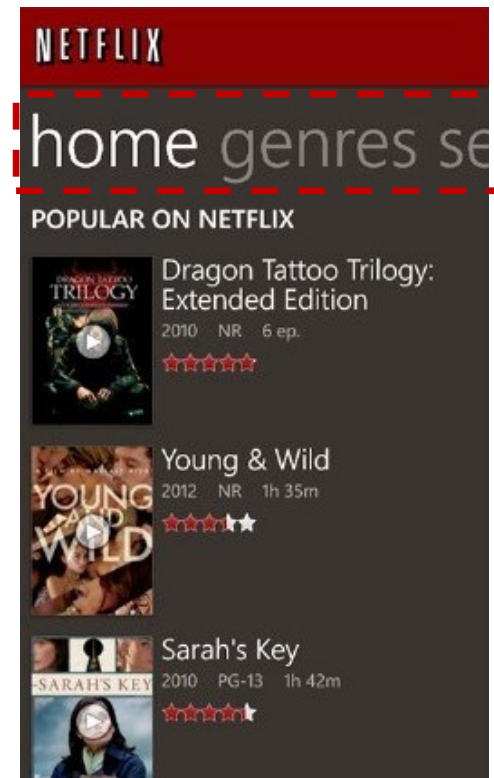
- ▶ The Gallery pattern displays live content (news, recipes, movies or photos) arranged in a **grid** (as with Recipeas app) or a **carousel** (as with BBC News)



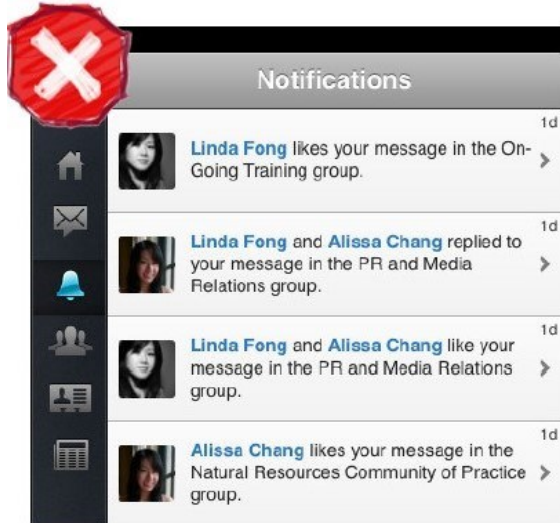
Tabs

Clearly differentiate the selected tab from the others!

- ▶ **Tab bars** are suggested to navigate **flat information structures**.
 - ▶ Users can navigate directly from one primary category to another because all primary categories are accessible from the main screen.



Tabs: further examples



Antipattern! Yammer has tabs, but **no labels**, making navigation more of a challenge.

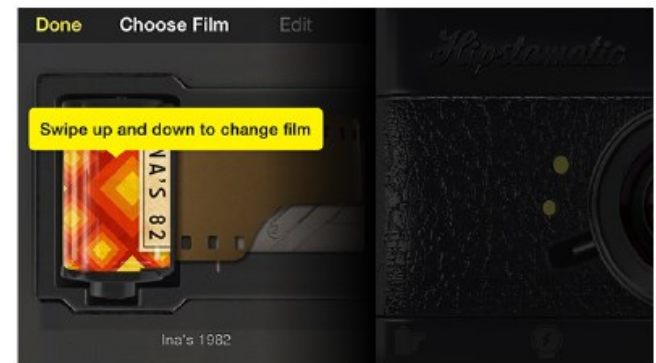
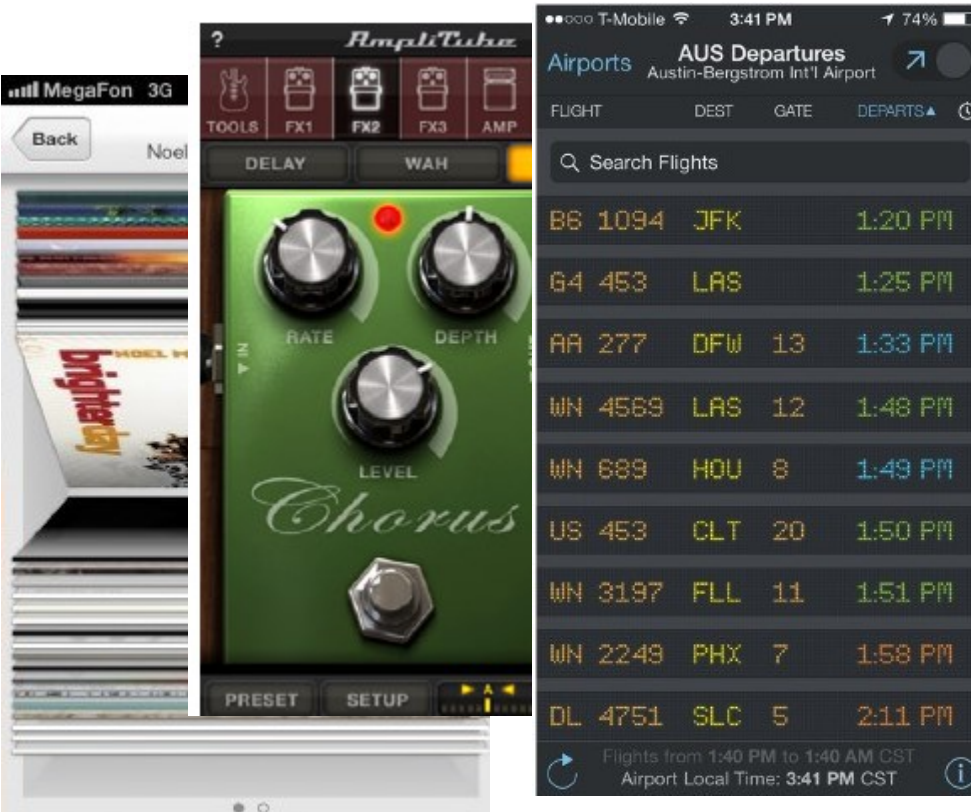
Do not confuse tab bar with toolbar!
The Tab Bar is for navigating the main categories of the app; the toolbar presents the tools and actions, for a specific screen.



Metaphor

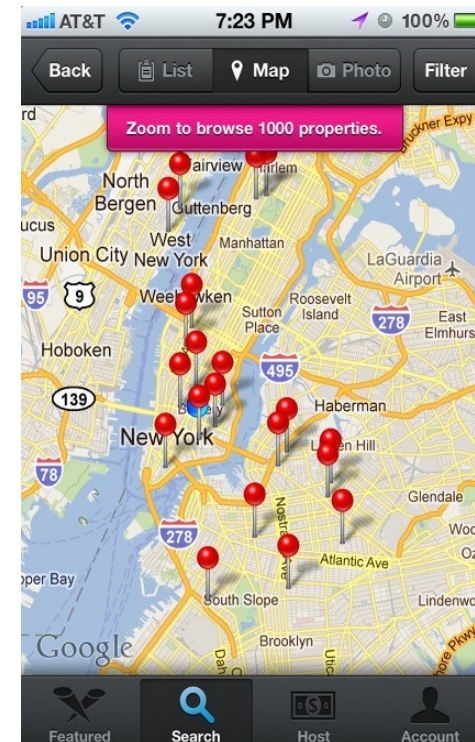
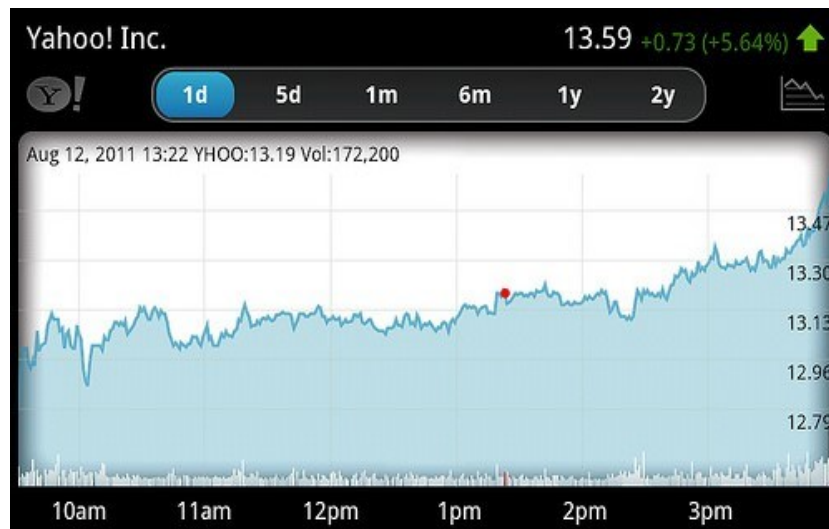
Hipstamatic for iOS.
Emulation of a real camera.

- ▶ This pattern is characterized by an interface designed to match its real-world counterpart.
 - ▶ Use the Metaphor pattern judiciously!



Infinite Area

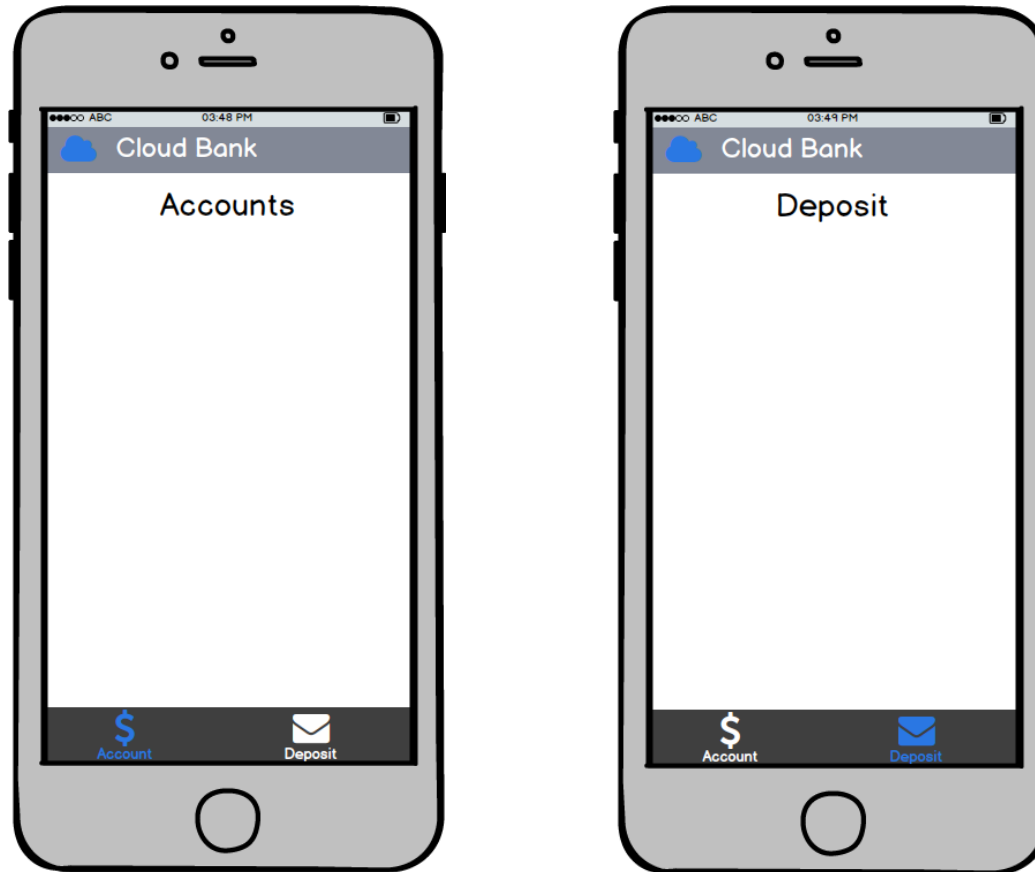
- ▶ The entire data set can be considered to be a large, navigable two dimensional graphic.
- ▶ The viewport shows only a small subset of the whole data set.



Exercise

- ▶ **We want to create a series of mockups for a mobile banking application for iOS devices.**
 - ▶ Our application will consist of two main navigation areas, called Accounts (Conto Corrente) and Deposit (Versamento). The Accounts page is the first that a user can navigate once the app is opened.
 - ▶ Please choose the most appropriate pattern to realize the navigation of the application and design a mockup through Balsamiq.

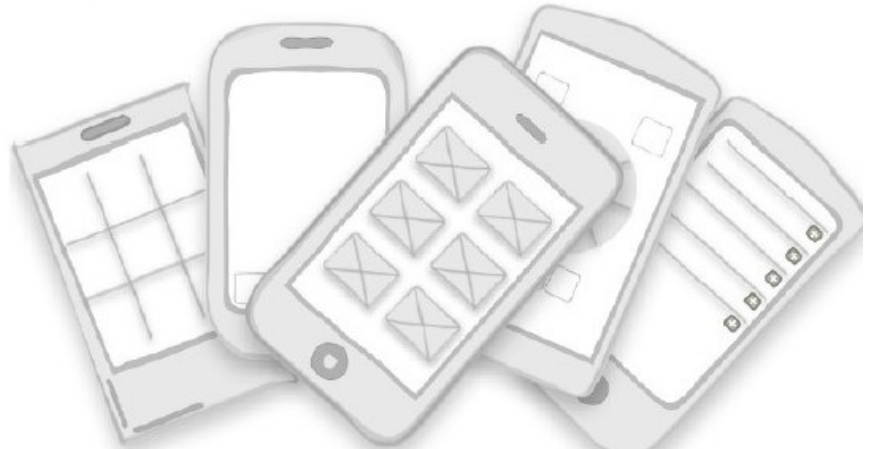
Solution



Primary Navigation

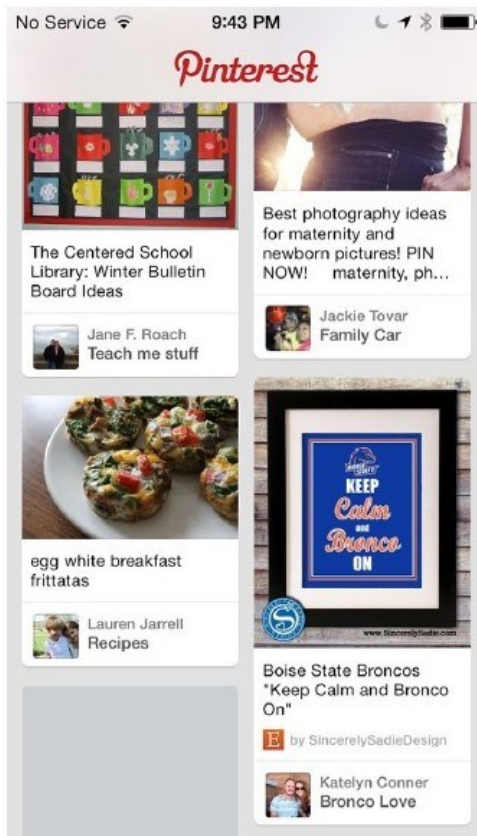
transient patterns

- ▶ The term **transient** means “staying a short time”, which is exactly how such navigation components work.
- ▶ They are **hidden** until we reveal them; then we make a selection and they **disappear again**.
- ▶ Transient Patterns:
 - ▶ Retracting Tab
 - ▶ Side Drawer
 - ▶ Toggle Menus



Retracting Tab

- ▶ Tab bar collapses or appears when the user is scrolling or swiping down through content.

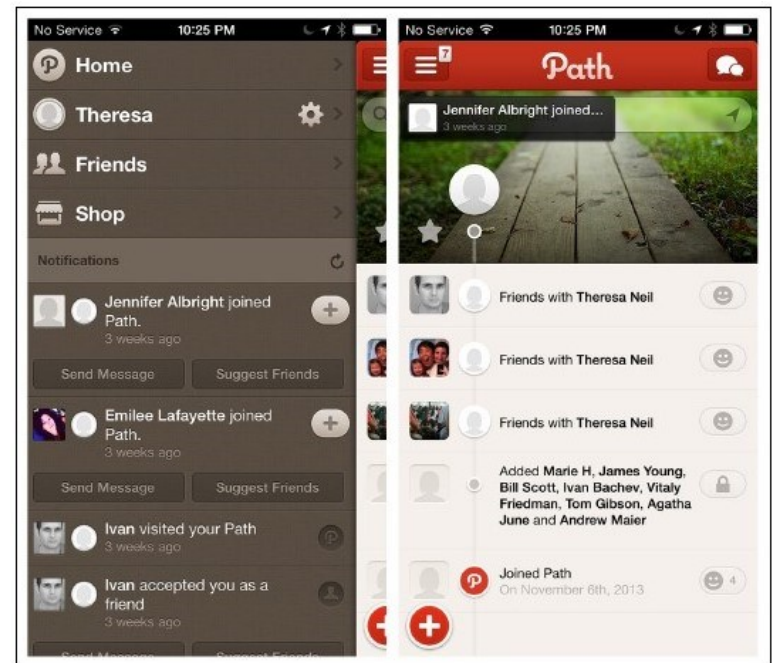
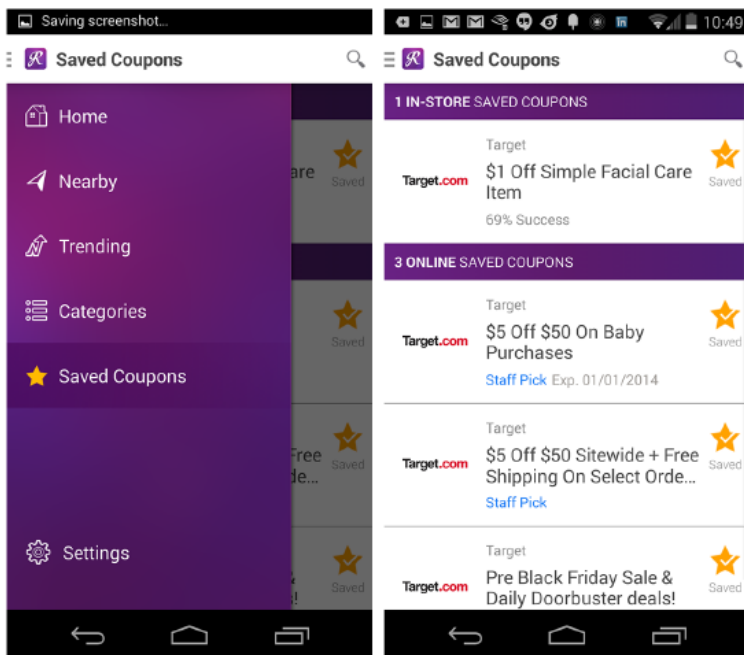


Pinterest for iOS:
scrolling down hides
the toolbar; scrolling
up reveals it!



Side Drawer

- ▶ There are two styles of Side Drawers.
 - ▶ The first is an **overlay**, meaning a **swipe gesture** will reveal a drawer that partially covers or overlaps the original screen content (as in RetailMeNot for Android).
 - ▶ The second style is an **inlay**, in which a **swipe gesture** will open a drawer that pushes the original screen content partially off-canvas (as in Path for iOS).



Tips for realizing side drawers

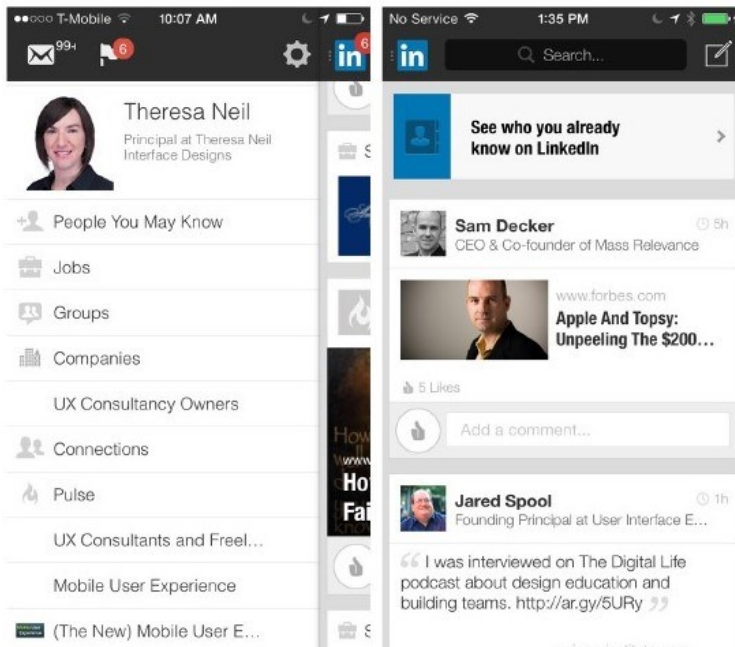
1. The Android design guidelines recommend having the **drawer automatically open on first use** so the user can see the menu and its structure, and learn how to close the drawer.
 - ▶ In the subsequent interactions, just launch the app with the side drawer closed.
2. The most popular orientation for the side drawer is **on the left**, but there are examples in which it is placed on the right (IfThisThenThat for iOS) or on both sides (Facebook for Windows Phone).
3. **Avoid to position** the side drawer on the bottom or on the top!
 - ▶ This positioning conflicts with the swipe up/down gesture that reveals the Control Center and Configuration features in iOS/Android devices.

Om finder for iOS. Almost every time we try to open the side drawer, the iOS Control Center opens instead.

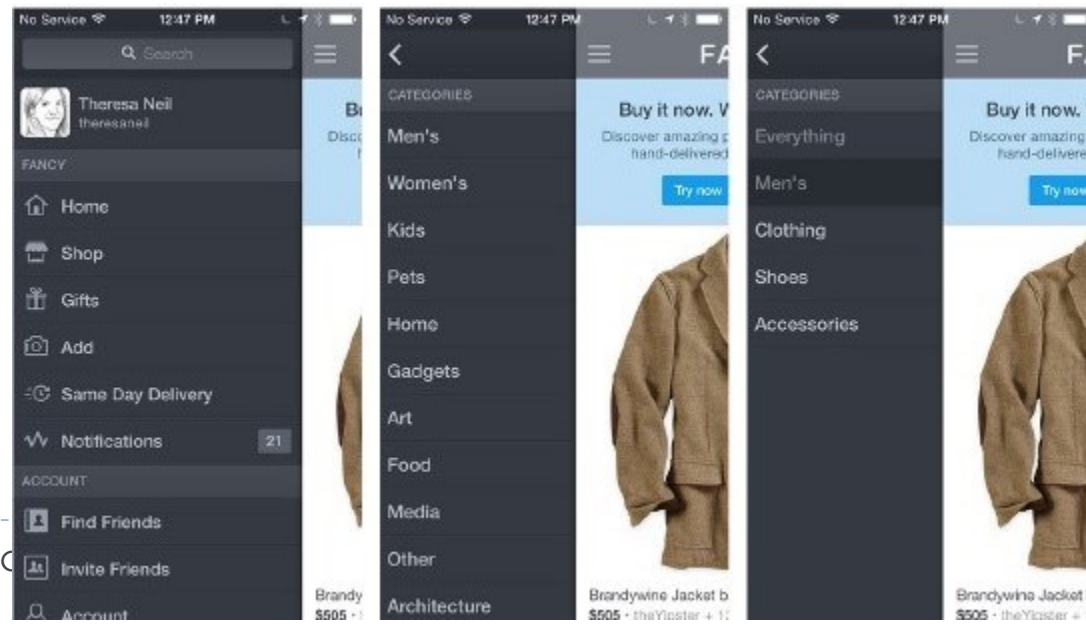


Tips for realizing side drawers

4. Side Drawer content **need not be limited** to only navigation options.
 - ▶ They may include profile information (LinkedIn for iOS)
5. Side Drawer can be more than one level deep. You can tap-tap-tap down the path until you reach the lowest level category.

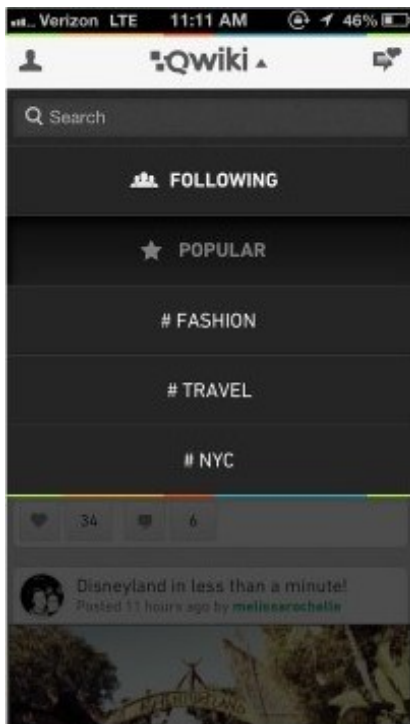


Do not overload side drawer with too many features. And **always include** a back button to the top-level navigation options.



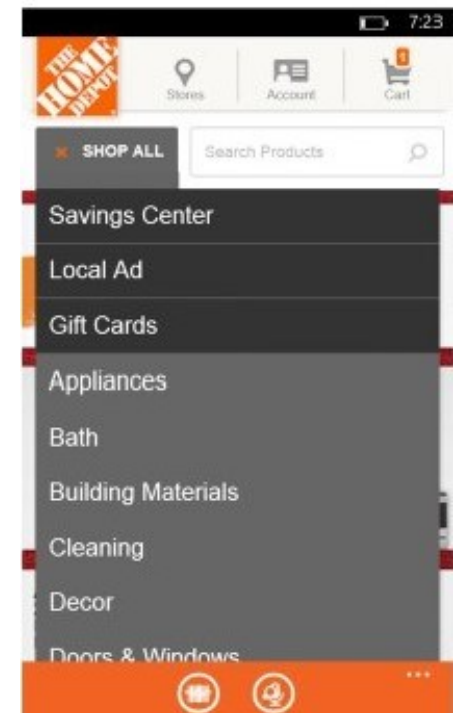
Toggle Menu

- ▶ The Toggle Menu can be an **inlay** that pushes the content down below the menu, or an **overlay** that appears as a layer above the content.
 - ▶ The overlay design is the **more common option** in native mobile apps.
 - ▶ **It should not cover the whole screen**, but instead let the background peek through. Tapping anywhere in the background should also hide the menu.



Qwiki for iOS.
Inlay toggle menu.

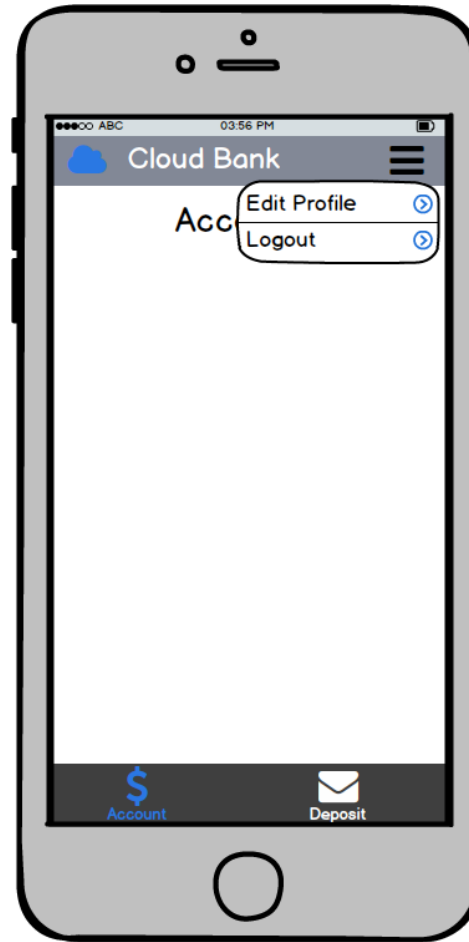
Home Depot for Windows phone
Overlay toggle menu.



Exercise

- ▶ **We want to create a series of mockups for a mobile banking application for iOS devices.**
 - ▶ Design a toggle menu that allows a logged user to see and edit her/his profile, and to perform the log out from the application. Design the mockup through Balsamiq.

Solution



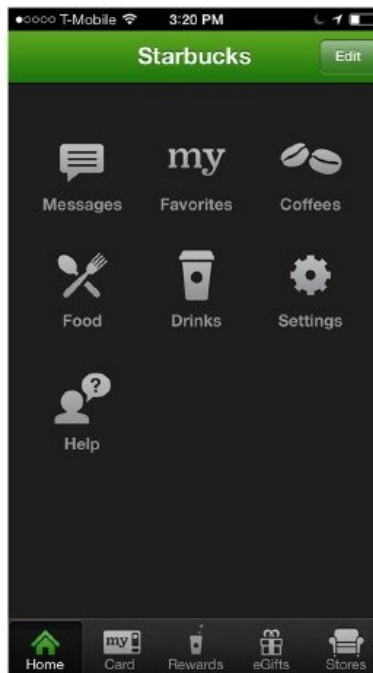
Persistent vs Transient Navigation

- ▶ When deciding between persistent and transient navigation, ask yourself a few questions:
 - ▶ Is your application “flat”?
 - ▶ Are the menu categories equivalent in hierarchy, and are there just a few primary categories (i.e., three to five) in the app?
 - ▶ Do your users need the menu to be always visible for quick access?
 - ▶ Do the menu categories have status indicators, like the number of unread emails, for instance?
- ▶ If you answered **yes** to one or more of these questions, it's probably best to stick with **persistent navigation**.

Secondary Navigation

National Geographic for iOS:
Side Drawer for primary navigation,
Gallery for secondary.

- ▶ It is about the navigation **within** a selected module.

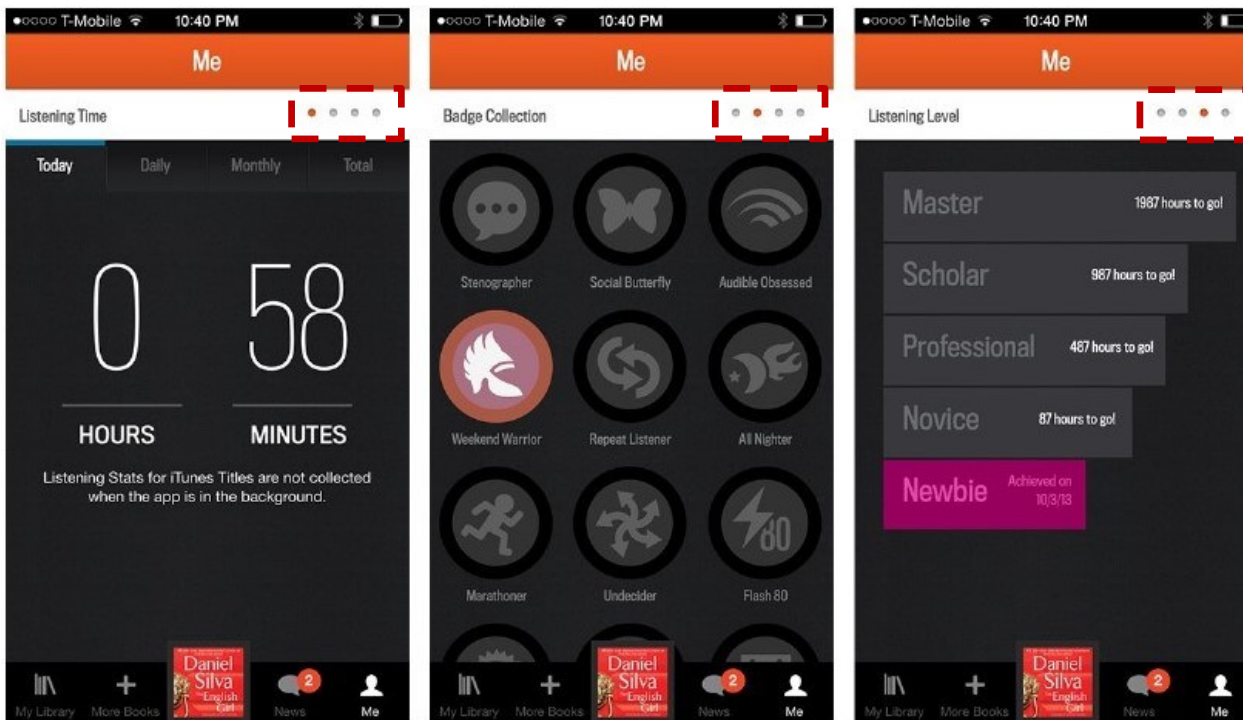


Starbucks for iOS:
Tabs for primary navigation,
Springboard for secondary.



Page Swiping

- ▶ This pattern for secondary navigation can be used to navigate quickly through content using the swipe gesture.
- ▶ The most common way to communicate this navigation pattern is via page indicators (the iOS term for the horizontal line of little dots) or with cards.

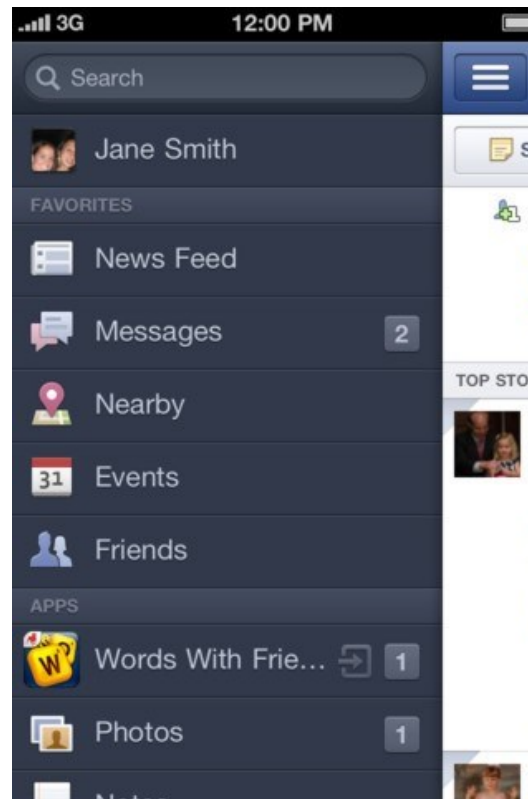
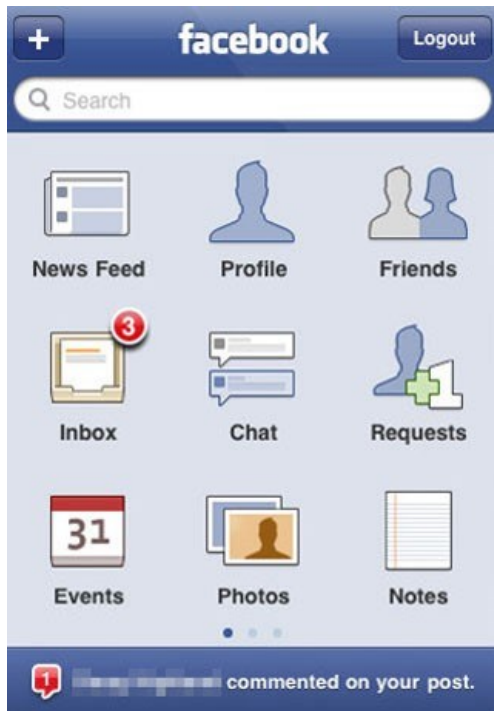


Audible for iOS.
Tabs for primary navigation,
Page Swiping for secondary.
Notice page indicators.



Exercise

- ▶ Can you compare these navigation designs?



Credits: Ivano Malavolta
Design Patterns for Mobile Apps

Roadmap

- ▶ Navigation
- ▶ **Tables**
- ▶ Forms
- ▶ Search
- ▶ Tools
- ▶ Helps and Tutorials
- ▶ Feedback & Affordance

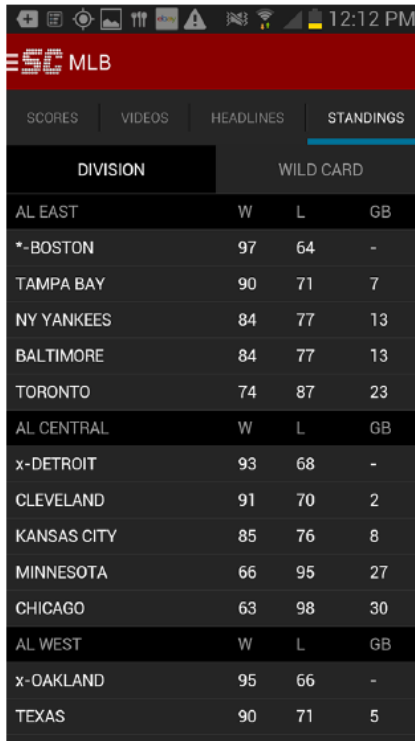
Tables

- ▶ Used for displaying (and interacting) with data:
 - ▶ Basic Table
 - ▶ Overview plus Data
 - ▶ Headerless Table
 - ▶ Fixed Column



Basic Table

- ▶ It is just a standard table with **fixed column headers** and a **grid layout**.
- ▶ Alternating table row colors, also called **zebra striping**, or placing a **thin horizontal line** between rows may enhance readability.



ESB MLB

SCORES VIDEOS HEADLINES STANDINGS

DIVISION	WILD CARD		
AL EAST	W	L	GB
*-BOSTON	97	64	-
TAMPA BAY	90	71	7
NY YANKEES	84	77	13
BALTIMORE	84	77	13
TORONTO	74	87	23
AL CENTRAL	W	L	GB
x-DETROIT	93	68	-
CLEVELAND	91	70	2
KANSAS CITY	85	76	8
MINNESOTA	66	95	27
CHICAGO	63	98	30
AL WEST	W	L	GB
x-OAKLAND	95	66	-
TEXAS	90	71	5

It's now common to **omit the vertical gridlines** to reduce visual noise.

Ant-Pattern. If skipping vertical gridlines, **keep columns aligned**

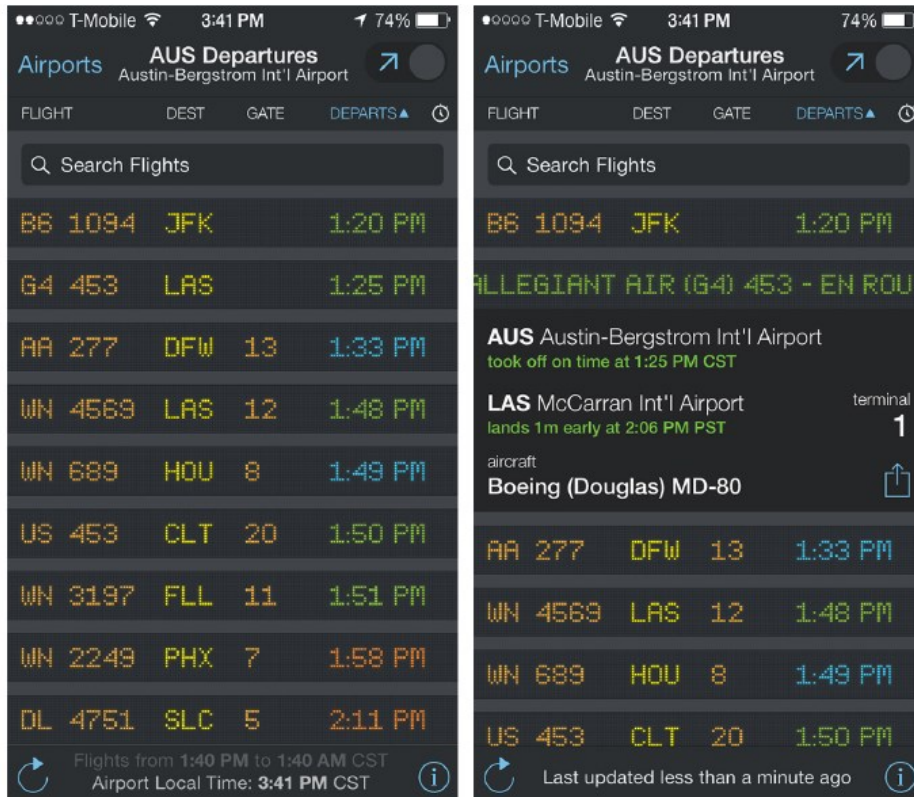


Stock Trainer

Ticker	Price	Cost	Num	Value
Equity Name	GL	\$ CHG	% CHG	
EZPW	\$16.90	\$18.68	20	\$338.00
EZCORP, Inc.		\$35.60		\$1.78 9.53%
AMAT	\$17.60	\$15.00	100	\$1760.00
Applied Materials, Inc.		\$260.00		\$2.60 17.33%
GOOG	\$876.39	\$886.43	15	\$13145.85
Google Inc.		\$150.60		\$10.04 1.13%
PFE	\$28.88	\$27.65	50	\$1444.00
Pfizer, Inc. Comm		\$61.50		\$1.23 4.45%
SSNLF	\$1300.00	\$1140.00	2	\$2600.00
ing Electronics Co. Ltd.		\$320.00		\$160.0 14.04% 0
TGT	\$63.94	\$69.38	15	\$959.10
Target Corporatio		\$81.60		\$5.44 7.84%

Basic Table (with Overview plus Data)

- ▶ It features a summary of the table's content displayed above the rows of data. Inline expand with tap.

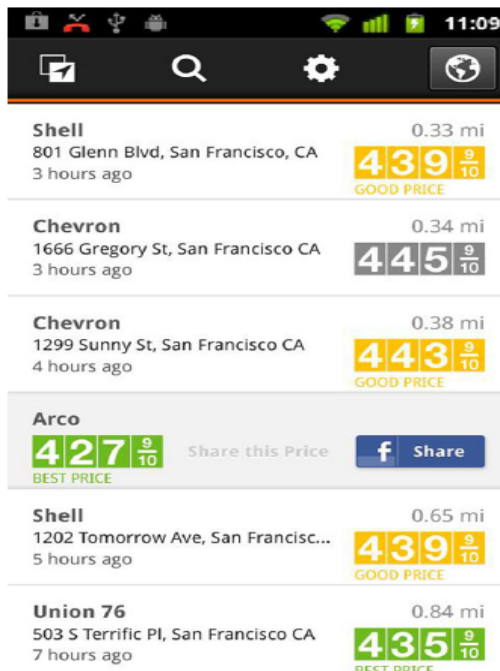


FlightBoard shows the flights for the selected airport. When users **tap on a row**, the flight details are shown **inline** instead of on a new screen.

Headerless Table

- ▶ It is characterized by **fat rows** displaying multiple variables for an object, and **no column labels**.
- ▶ It is common practice to **make the row identifier**—a key piece of information — **stand out**.

Ideal for displaying collections of items (like inventory, albums) and **search results**



Test Scores	
<input type="text" value="start typing to search"/>	
Amy	C
Test date: 08/10/2011	77 of 100
Bob	B-
Test date: 07/14/2011	84 of 100
Chris	A
Test date: 06/23/2011	95 of 100
Daniel	C
Test date: 08/01/2011	80 of 100
Eliza	F
Test date: 08/01/2011	64 of 100
Frank	C+

Test Scores	
<input type="text" value="Da"/>	
Damien	A-
Test date: 08/10/2011	94 of 100
Damon	C
Test date: 07/14/2011	78 of 100
Daniel	C
Test date: 06/23/2011	80 of 100
Danny	B+
Test date: 08/01/2011	93 of 100
Daphane	A+
Test date: 08/01/2011	99 of 100

Fixed Columns

- ▶ For larger tables, the Fixed Column pattern may be a viable solution.
- ▶ The **leftmost column is fixed** and the **other columns swipe side-to-side**.

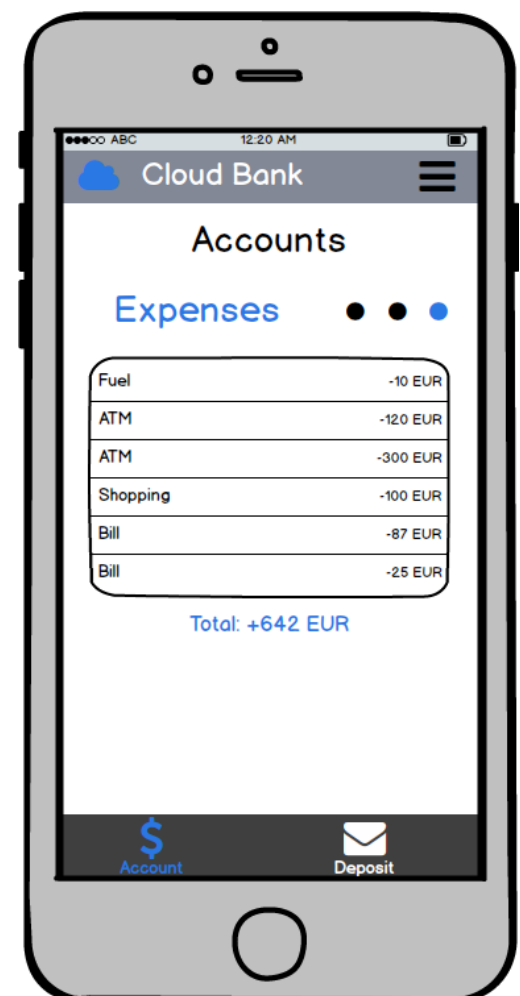
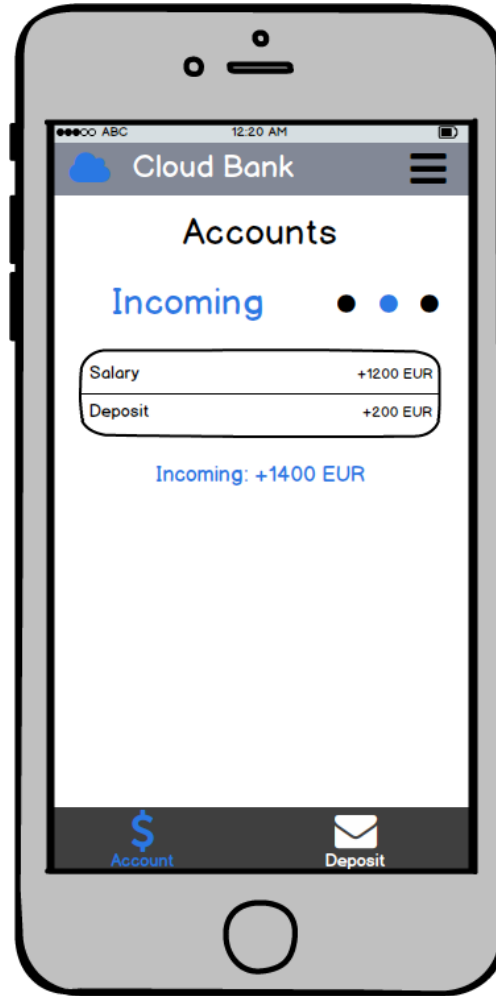
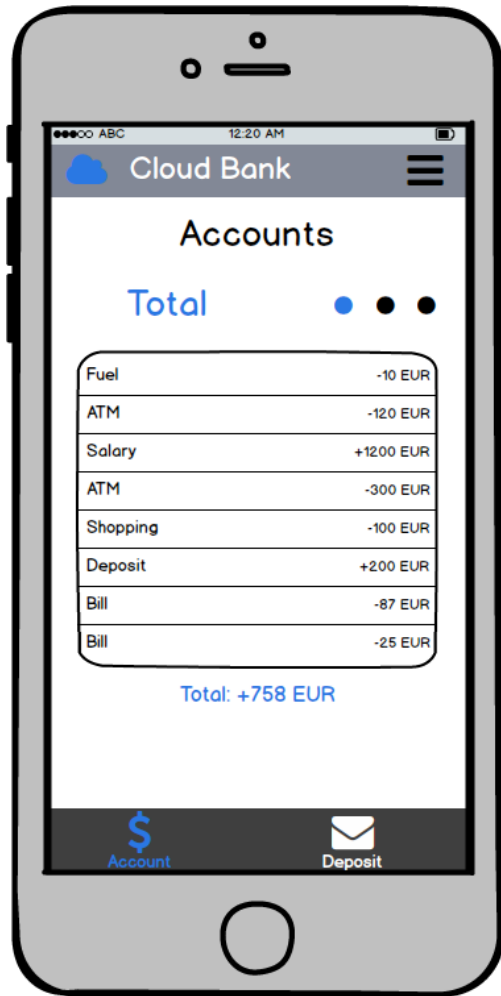
Sample - Prescriber Info				Sample - Prescriber Info			
Prescriber	Territory	Specialty	City	Prescriber	Roambitram CM vs PM	Roambitram CM vs P4	Prextrol CM
ABBOTT, LAURIE	12BA1	NRP	WAT	ABBOTT, LAURIE	32.7%	0.9%	
ABISALIH, JOHN	12BA1	CD	AL	ABISALIH, JOHN	15.2%	-6.8%	
ABOULEISH, PATRICIA	12BA1	IM	HC	ABOULEISH, PATRICIA	30.6%	-17.3%	
AL-ATRASH, MARGARET	12BA1	IM	BA	AL-ATRASH, MARGARET	10.4%	2.8%	1
ALBERT, WILLIAM	12BA1	NRP	LEV	ALBERT, WILLIAM	24.5%	16.5%	1,1
ALESSI, STEVEN	12BA1	FP	LII	ALESSI, STEVEN	15.2%	-34.6%	
ALEXANDER, STACY	12BA1	FP	PRE	ALEXANDER, STACY	33.6%	-17.3%	-1
ANDERSON, PETER	12BA1	IM	RO	ANDERSON, PETER	-	110.1%	
ANDRIANOV, JAN	12BA1	IM	MAD	ANDRIANOV, JAN	16.1%	62.5%	-
ANIEL, LESLEY	12BA1	IM	RU	ANIEL, LESLEY	28.0%	33.6%	-
AUGER, CLIFTON	12BA1	FP	FAR	AUGER, CLIFTON	-	-	
AXELSON, ROBERT	12BA1	NRP	WAT	AXELSON, ROBERT	-	-	
BARTLEY, ELIZABETH	12BA1	NRP	GU	BARTLEY, ELIZABETH	12.6%	-15.7%	-

It is important to clearly show that more information is available on the right.

Exercise

- ▶ **We want to create a series of mockups for a mobile banking application for iOS devices.**
 - ▶ Under the Account page, design three different views to keep track of your money.
 1. The first view shows the complete list of incoming (entrata) and expenses (uscite).
 2. The second view shows just the list of incoming.
 3. The third view shows just the list of expenses.
 - ▶ Exploit a secondary pattern for navigating the different views and tables to represent the data.

Solution



Roadmap

- ▶ Navigation
- ▶ Tables
- ▶ **Forms**
- ▶ Search
- ▶ Tools
- ▶ Helps and Tutorials
- ▶ Feedback & Affordance

Forms

- ▶ Used for data entry and configuration features:
 - ▶ Sign In
 - ▶ Registration
 - ▶ Check-in
 - ▶ Comments
 - ▶ Users Profile
 - ▶ Share
 - ▶ Empty Datasets
 - ▶ Multi-step
 - ▶ Settings



Sign In

- ▶ Sign In forms require a minimal number of inputs:
 - ▶ username
 - ▶ password
 - ▶ command button
 - ▶ password help
 - ▶ option to register
 - ▶ social sign in [optional]

Some apps do this
in a **single screen**
(Groupon for iOS)

◀ T-Mobile 4:01 PM

← Groupon

Sign in to your Groupon account

f Sign In with Facebook

or

theresaneil@gmail.com

Password

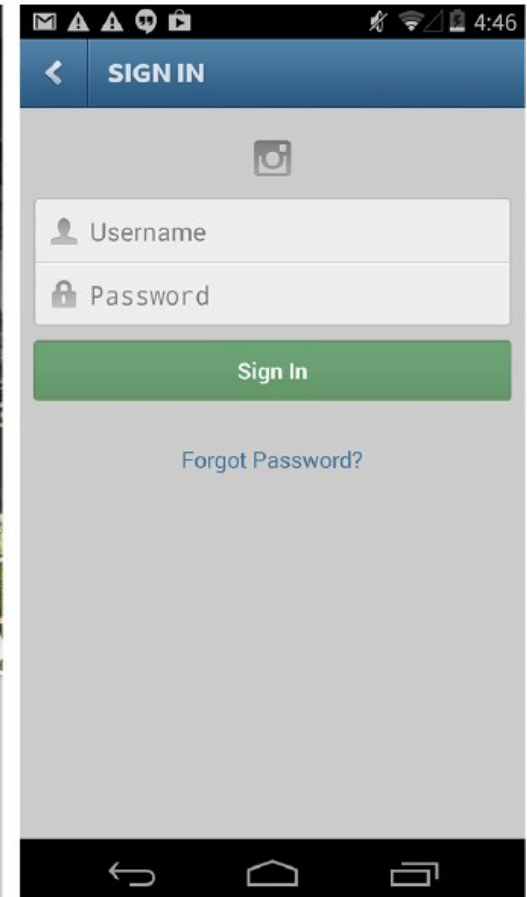
Sign In

Forgot Password?

Need a Groupon Account? Sign Up

Sign In

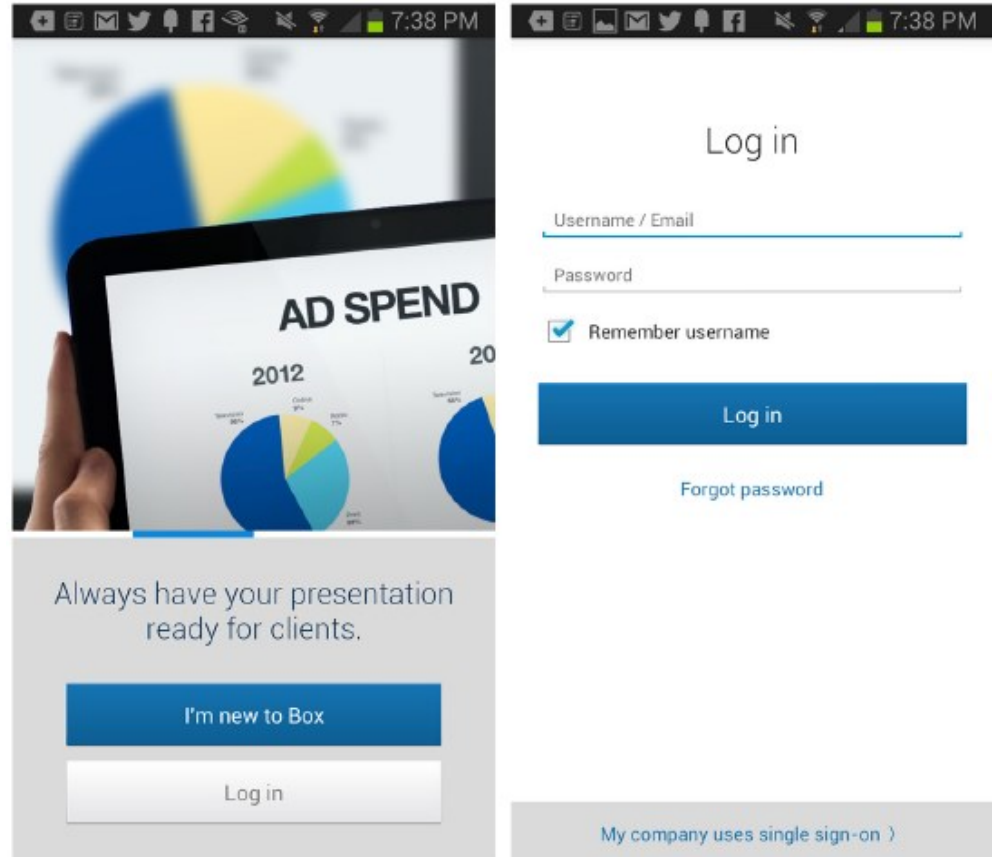
Other apps (Instagram for Android) present the Sign In and Register options **up front**, then take the user to the appropriate form.



Sign In

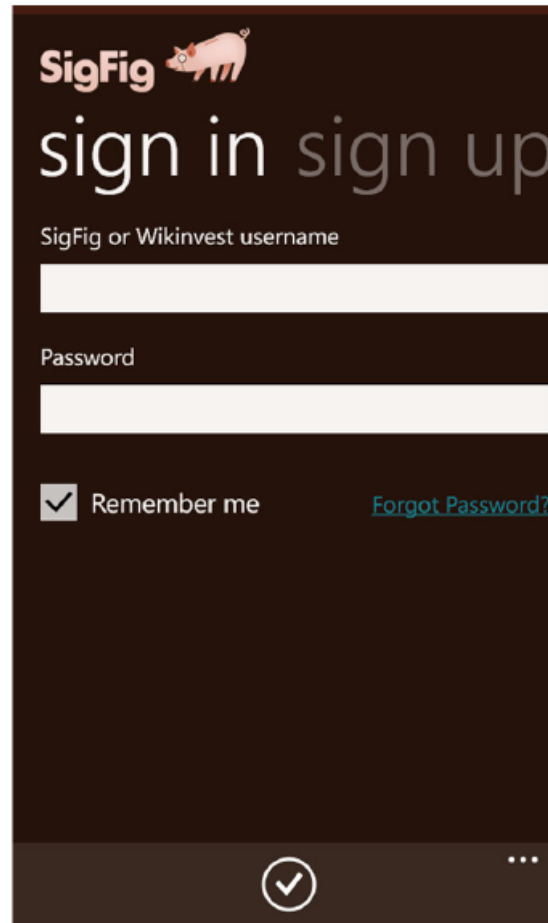
Other apps (Box for Android), place the Sign Up and Log In options in context **within the promotional tour**.

Prospective users may take action once they have been persuaded to use the app, and equally easy to simply log in.

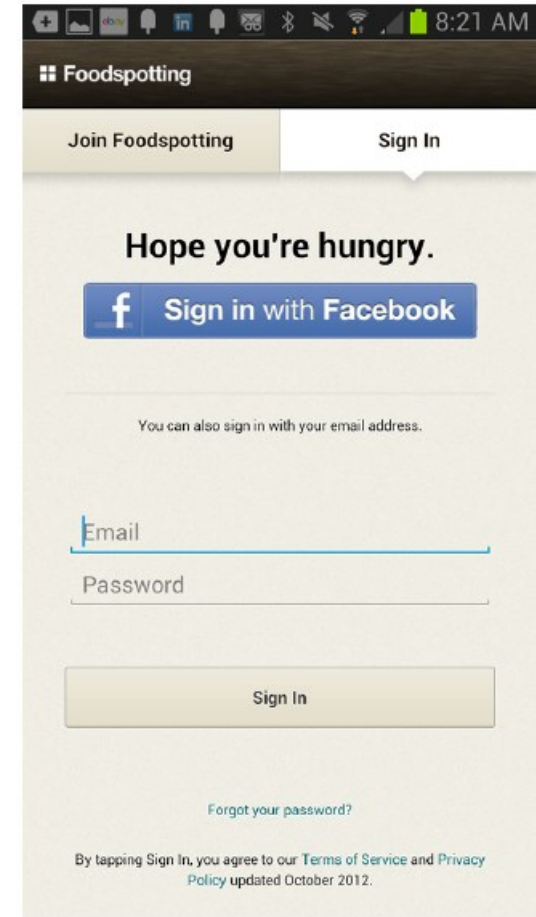


Sign In

Sign In and Sign Up on **tabs**.
(Foodspotting for Android and
SigFig for Windows Phone)



The image shows the sign-in screen of the SigFig mobile app. At the top, the SigFig logo (a pig) is displayed. Below it, the text "sign in sign up" is shown in a large, light font. The form includes a text input field for "SigFig or Wikinvest username", a text input field for "Password", and a checkbox labeled "Remember me" with a "Forgot Password?" link. A checkmark icon is visible at the bottom center of the screen.



The image shows the sign-in screen of the Foodspotting mobile app. At the top, the Foodspotting logo is displayed. Below it, the text "Join Foodspotting" and "Sign In" are shown. The main heading is "Hope you're hungry." followed by a blue button with the Facebook logo and the text "Sign in with Facebook". Below this, there is a note: "You can also sign in with your email address." followed by text input fields for "Email" and "Password". A "Sign In" button is located at the bottom. At the very bottom, there is a link for "Forgot your password?" and a footer note: "By tapping Sign In, you agree to our Terms of Service and Privacy Policy updated October 2012."

Registration . . .

Do not force users to register if it is not strictly required!

- ▶ **Keep it short**, preferably one screen. Register button well visible.
- ▶ Eliminate the redundant Confirm Email and Confirm Password fields.

Nook for Android.
Redundant Email and Password fields.

Tips: prompt users to double-check the email. Use Hide/Show options for checking the password.

iPod 6:58 PM

Back Create Account

Have a Facebook account?

Sign Up with Facebook

GOWALLA WILL NEVER POST TO YOUR ACCOUNT WITHOUT YOUR PERMISSION

or

First name

Last name

Email

Password

T-Mobile 6:28 PM

Email Address
theresaneil@gmail.com

Confirm Email Address
theresaneil@gmail.com

Password

Confirm Password

Done

anti-pattern

Q W E R T Y U I O P
A S D F G H J K L
Z X C V B N M
. ? 1 2 3 space Go

T-Mobile 11:18 PM

Create Account Next

theresa@tnid.com

Picture

Theresa Neil

Does your e mail address look right?
theresa@tnid.com

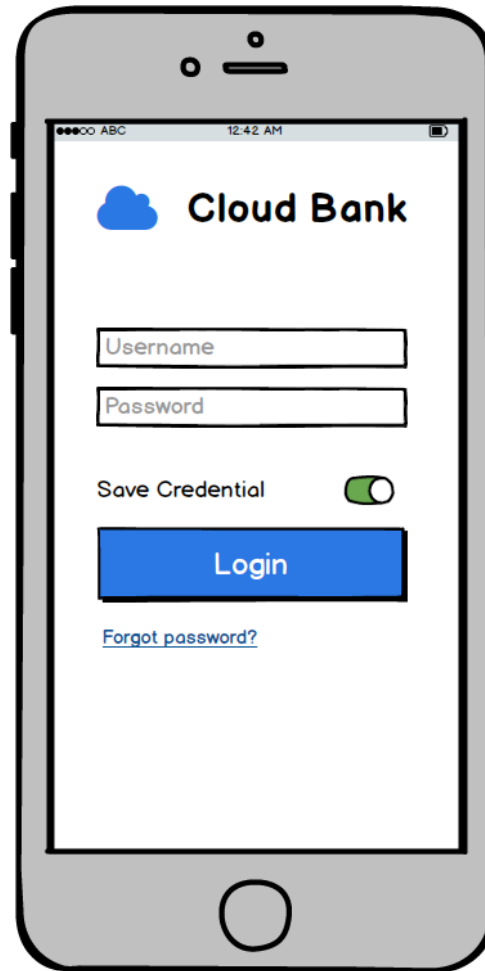
Change Looks Good!

By signing in, I accept Path's Terms of Service and Privacy Policy

Exercise

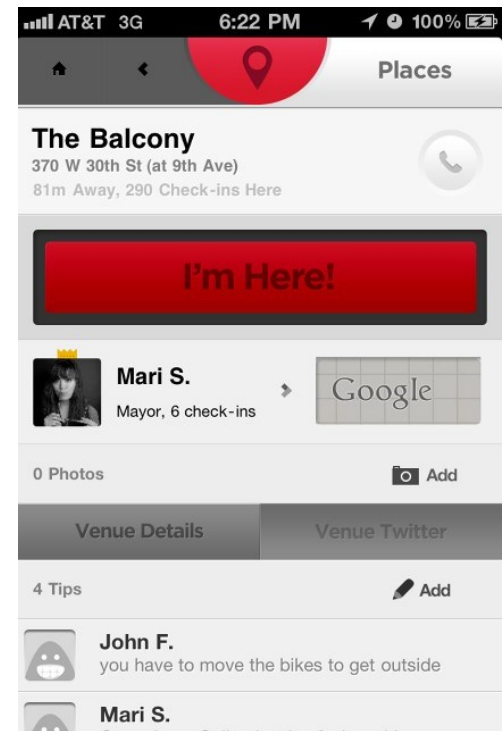
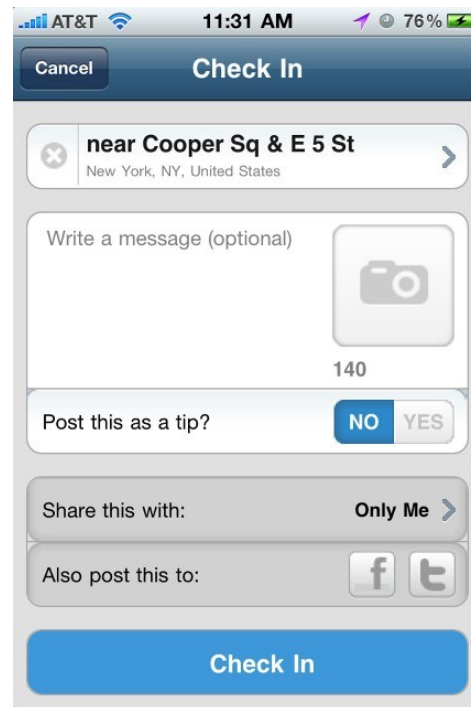
- ▶ **We want to create a series of mockups for a mobile banking application for iOS devices.**
 - ▶ Insert a login feature that is shown when the app is opened. It should include facilities to recover the password (if forgotten) and an option to save credential for future access.
 - ▶ Each user must perform the login before using the app.
 - ▶ Clicking on Logout in the Toggle Menu makes the user come back in the Login page.

Solution



Check-in

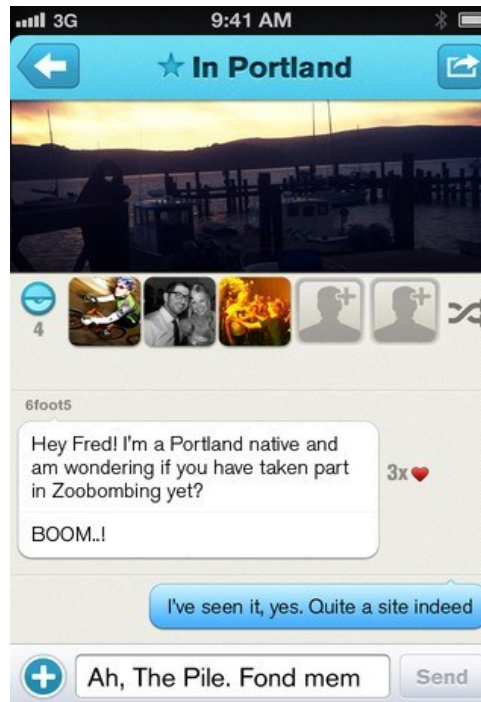
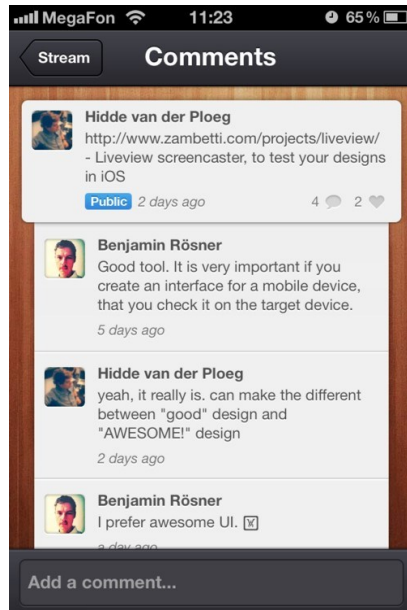
- ▶ It allows people to use the GPS on their mobile devices to let friends know exactly where they are.
- ▶ Keep it **ultra-short**. Design for speed, and efficiency.
- ▶ **Eliminate unnecessary fields.**



Credits: Ivano Malavolta
Design Patterns for Mobile Apps

Comments

- ▶ Feature used for inviting and allowing users to leave comment.
- ▶ **Always clarify** what is being commented!
- ▶ Show other people's **comments over time**.



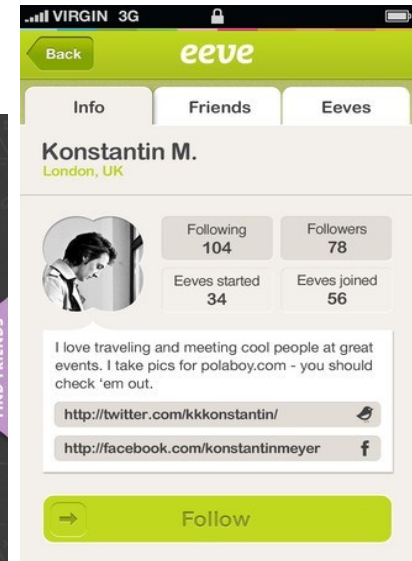
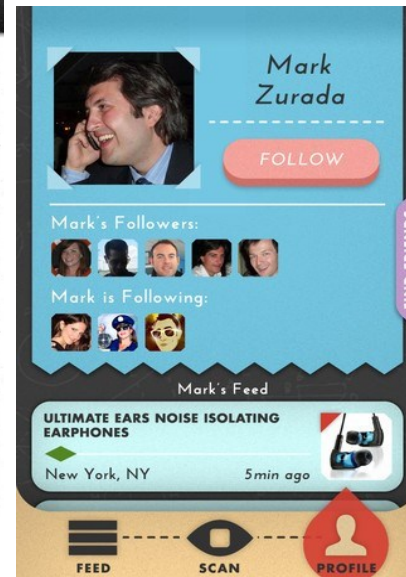
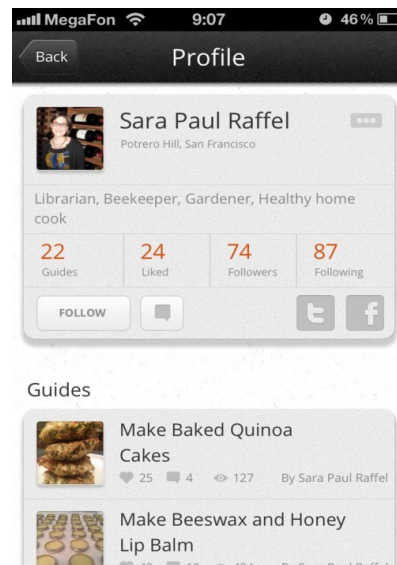
Credits: Ivano Malavolta
Design Patterns for Mobile Apps

How to show timeline?

- ▶ Within the current hour, show as *minutes ago*.
- ▶ Within the current day, show the time as number of *hours ago*.
- ▶ Within the last two days, shown the *day of the week*, and the *time range*, morning, afternoon, evening, night.
- ▶ Within the past week, show the *day of the week*.
- ▶ Within 12 months, show as *mm/DD*.
- ▶ Older than 12 months, show *only the year*.

User's Profile

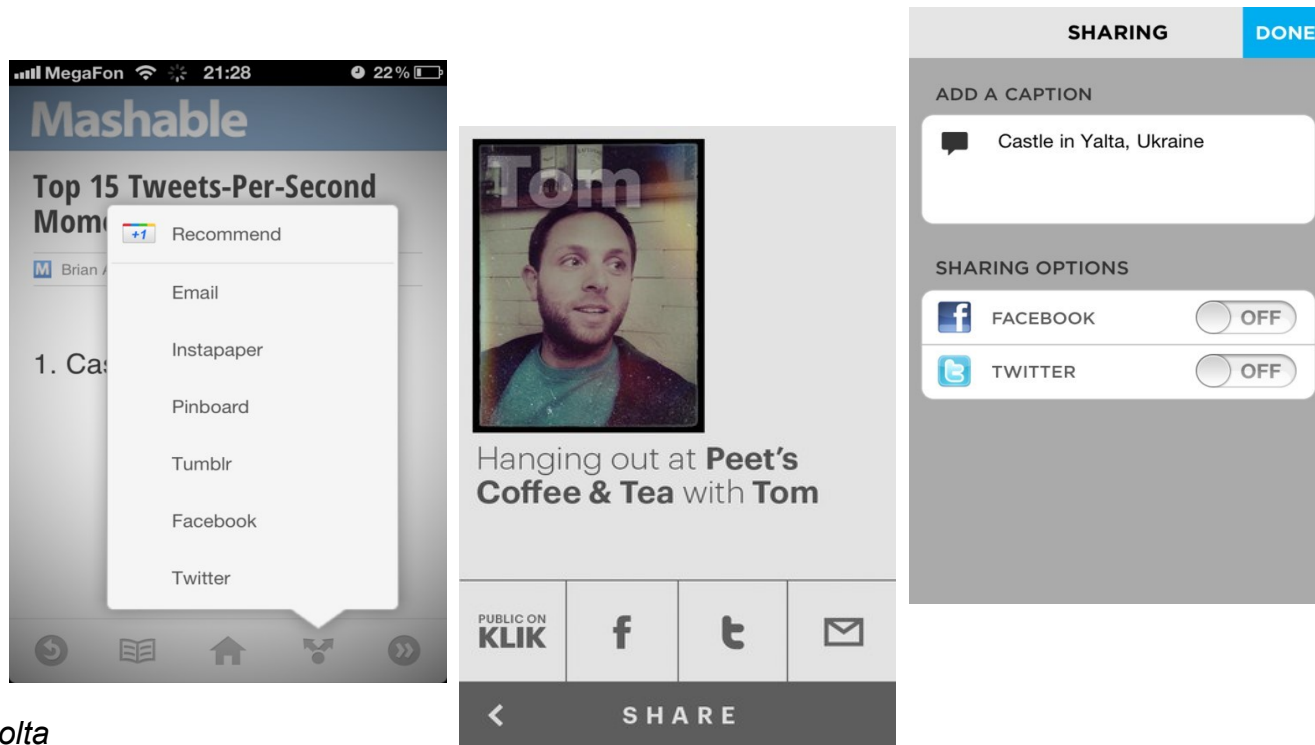
- ▶ Put the **name** and the **picture in evidence**.
- ▶ Show their **contribution** to the app or to the social network.
- ▶ Provide **action controls**.



Credits: Ivano Malavolta
Design Patterns for Mobile Apps

Share

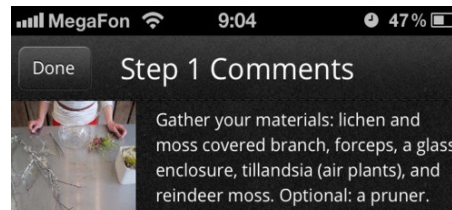
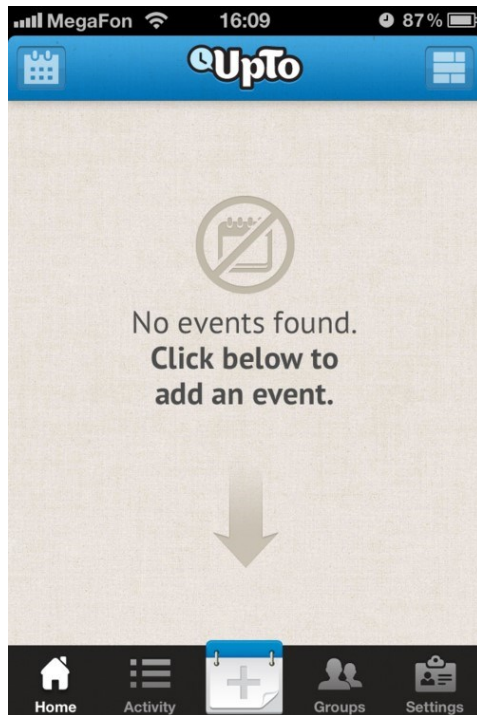
- ▶ Always provide an **“off social” way to share** (e.g., by email).
- ▶ **Remark what is being shared** and keep track of past logins.



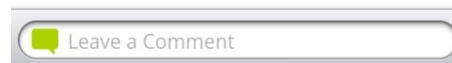
Credits: Ivano Malavolta
Design Patterns for Mobile Apps

Empty Datasets

- ▶ **Avoid white-screens**, explain why the dataset is empty
- ▶ Call for action, only one!
- ▶ **Avoid error messages.**



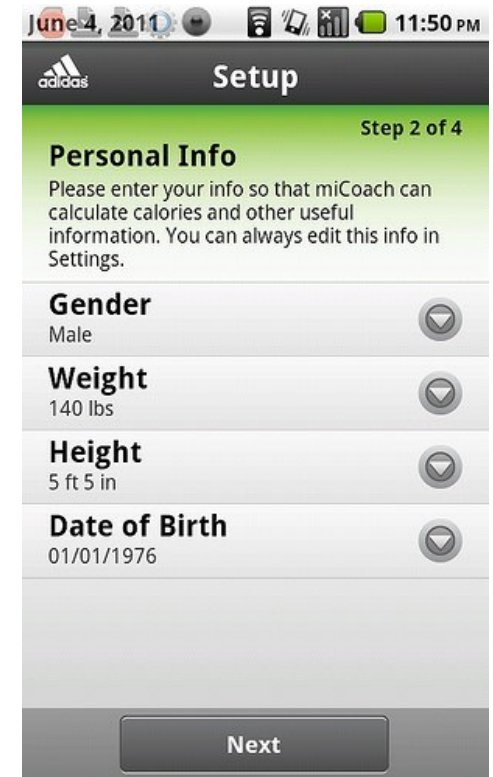
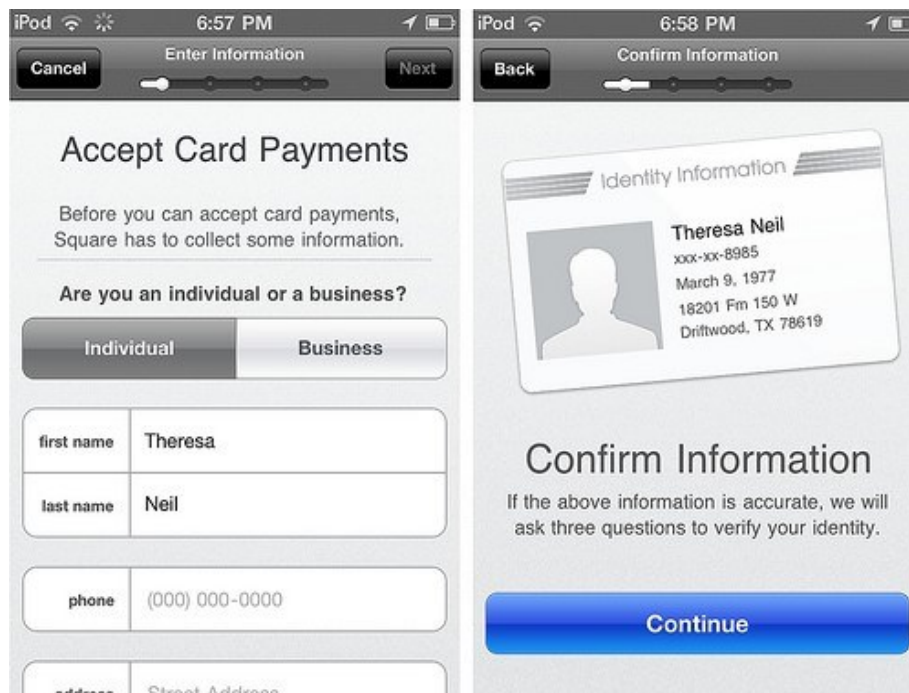
Be the first
to comment on
this step



*Credits: Ivano Malavolta
Design Patterns for Mobile Apps*

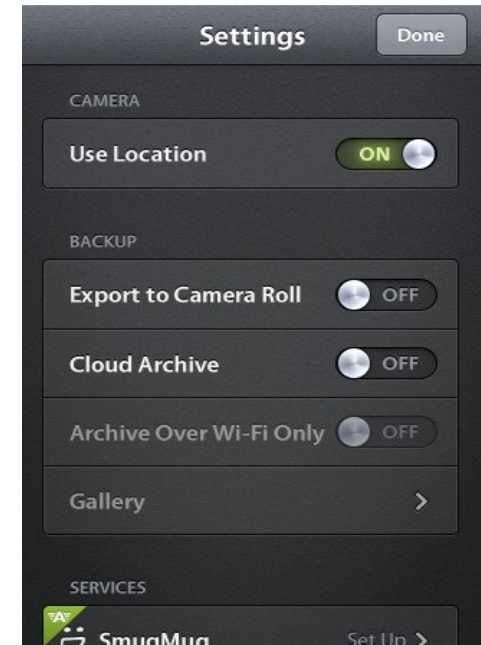
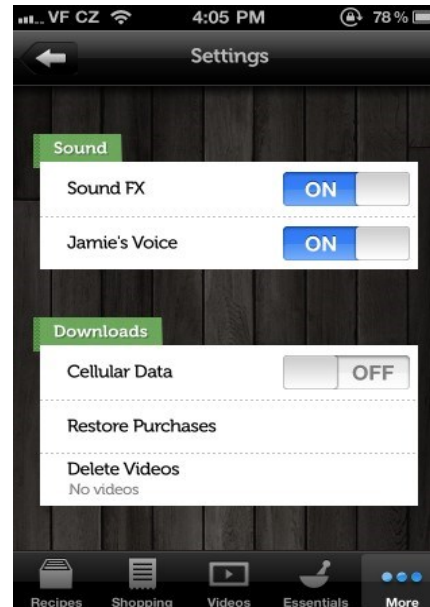
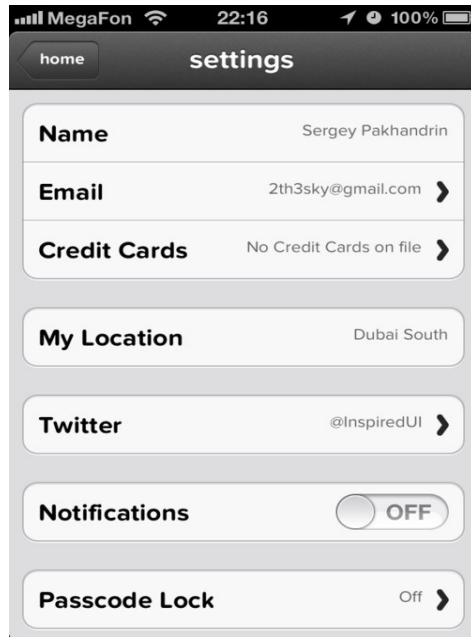
Multi Steps

- ▶ Show the user **where they are** and **where they can go**.
- ▶ Often organized as **sequential workflows**.
- ▶ Minimize the number of pages and steps.



Settings

- ▶ Put them inside the app.
- ▶ **Clear and grouped!** Easy to be understood!



*Credits: Ivano Malavolta
Design Patterns for Mobile Apps*

Roadmap

- ▶ Navigation
- ▶ Tables
- ▶ Forms
- ▶ **Search**
- ▶ Tools
- ▶ Helps and Tutorials
- ▶ Feedback & Affordance

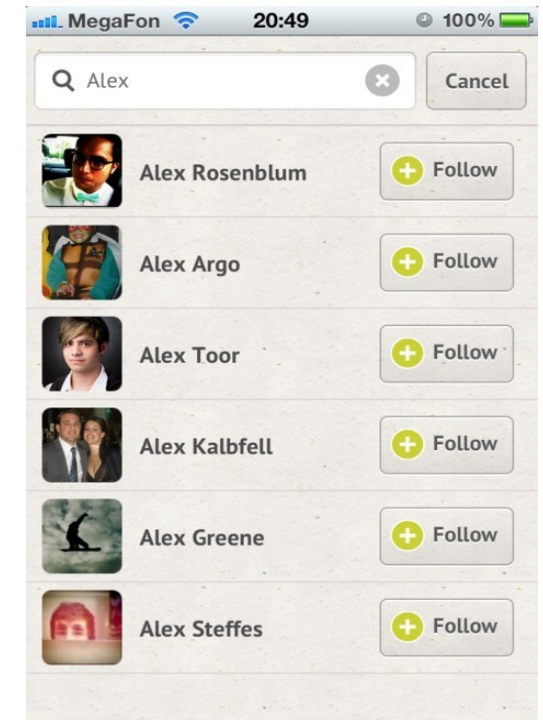
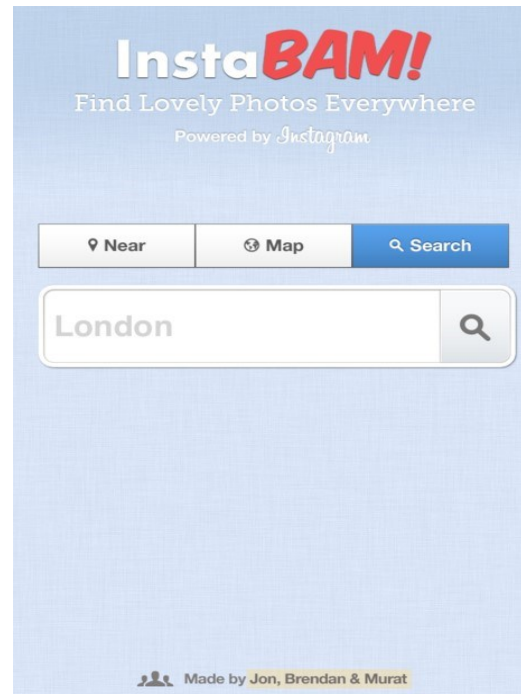
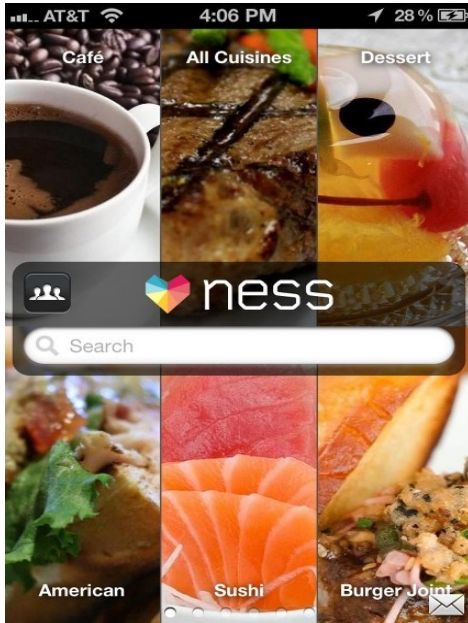
Search

- ▶ Explicit Search
- ▶ Dynamic Search
- ▶ Search form
- ▶ Search Results



Explicit Search

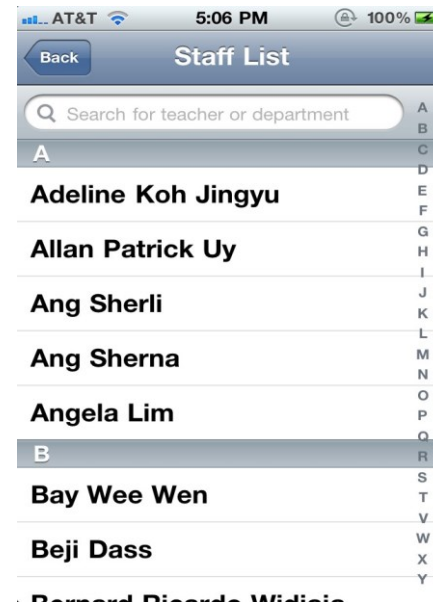
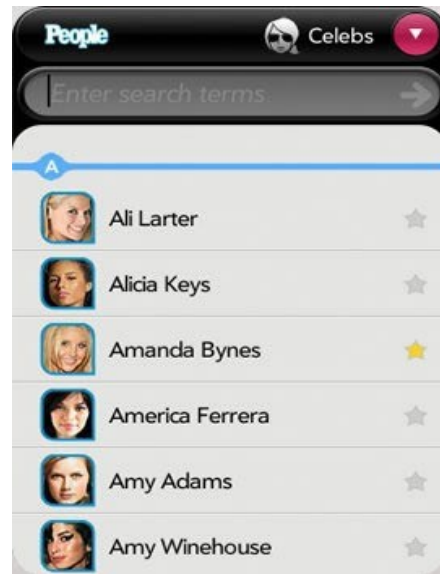
- ▶ Offer a **clear button** in the field.
- ▶ Provide an **option to cancel** the search.
- ▶ **Use feedback** to show the search is being performed.



Credits: Ivano Malavolta
Design Patterns for Mobile Apps

Dynamic Search

- ▶ **Automatically filters** a given list of items dynamically, during typing.
- ▶ Works well for constrained data sets, like an address book or personal media library.



Credits: Ivano Malavolta
Design Patterns for Mobile Apps

Search form

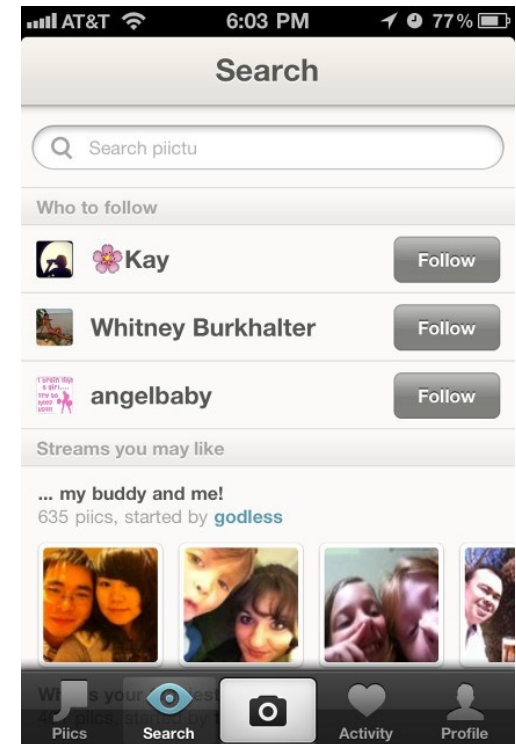
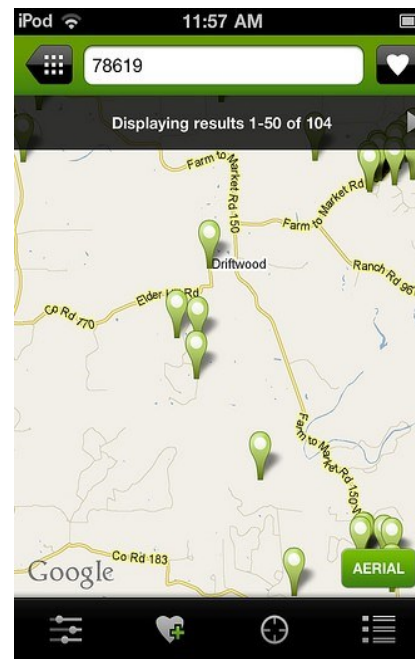
- ▶ **Minimize** the number of input fields.
- ▶ Follow form design best practices (alignment, labels, size)
- ▶ Use only when strictly needed!



Credits: Ivano Malavolta
Design Patterns for Mobile Apps

Search Results

- ▶ **Scroll down** to analyze the results.
- ▶ Apply a **reasonable default sort order**.
- ▶ Call for action.



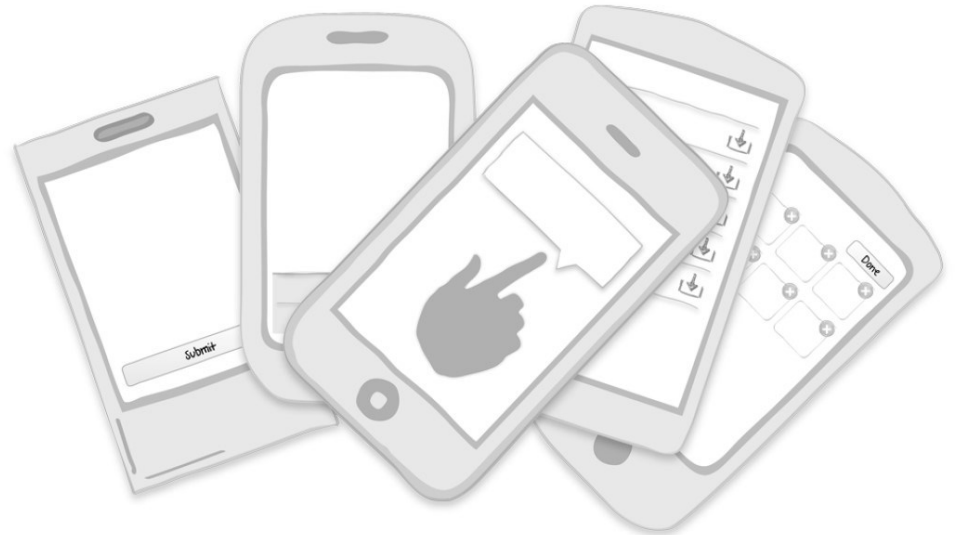
Credits: Ivano Malavolta
Design Patterns for Mobile Apps

Roadmap

- ▶ Navigation
- ▶ Tables
- ▶ Forms
- ▶ Search
- ▶ **Tools**
- ▶ Helps and Tutorials
- ▶ Feedback & Affordance

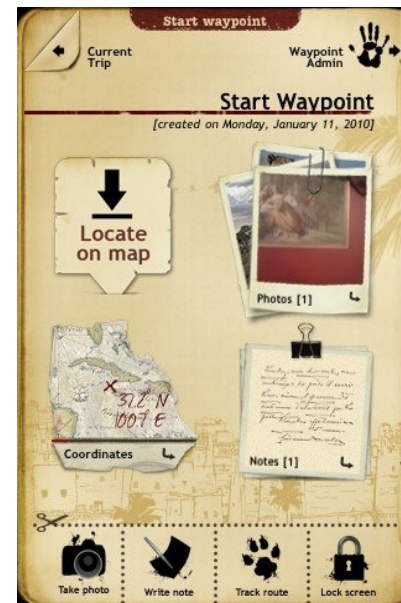
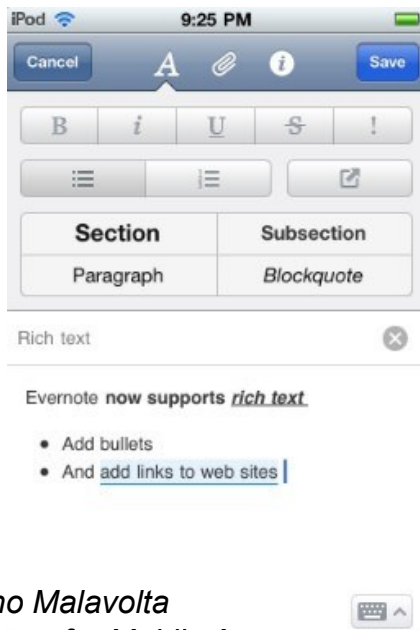
Tools

- ▶ **Buttons & Controls**
 - ▶ Toolbar
 - ▶ Contextual Buttons
 - ▶ Inline Actions
 - ▶ Call to Action Buttons
 - ▶ Multi-state Button
 - ▶ Actions on Maps



Toolbar

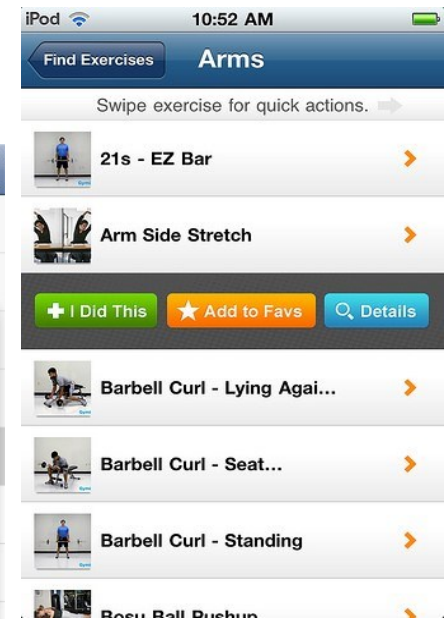
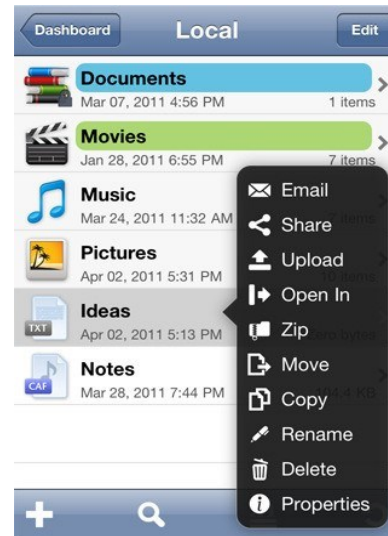
- ▶ Contain **screen level actions**, they are generally displayed at the bottom of the screen. **Different** from the tab bar!
- ▶ Choose icons that are easy to recognize, or use labels plus icons!



Credits: Ivano Malavolta
Design Patterns for Mobile Apps

Contextual Buttons

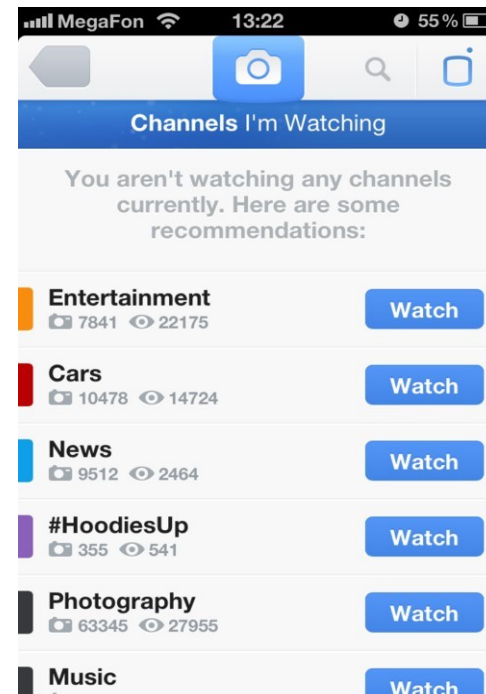
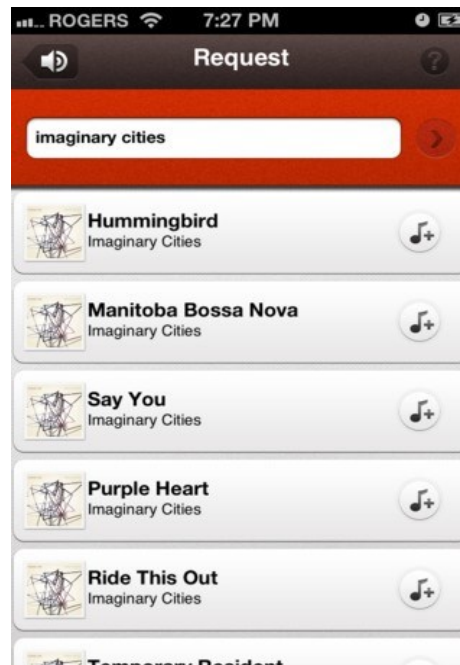
- ▶ If buttons are necessary, they should be displayed in **proximity to the actionable object**.
- ▶ Choose a familiar icon or use a text label.



*Credits: Ivano Malavolta
Design Patterns for Mobile Apps*

Inline Actions

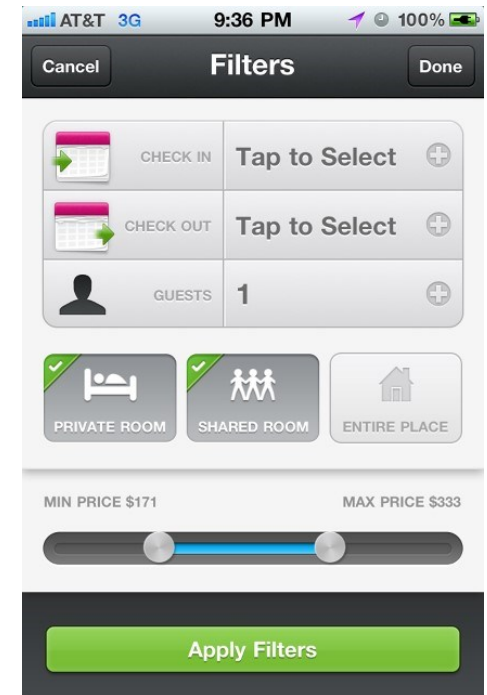
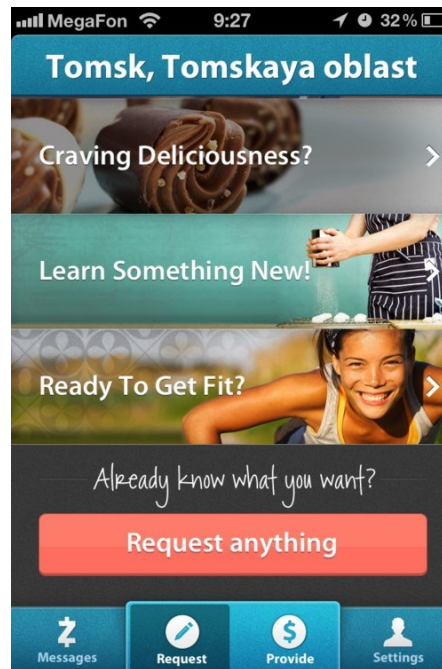
- ▶ They should be in **proximity to the actionable object**.
- ▶ Choose a familiar icon or use a text label.
- ▶ Max 1 to 2 Inline Actions per object!



Credits: Ivano Malavolta
Design Patterns for Mobile Apps

Call to Action Buttons

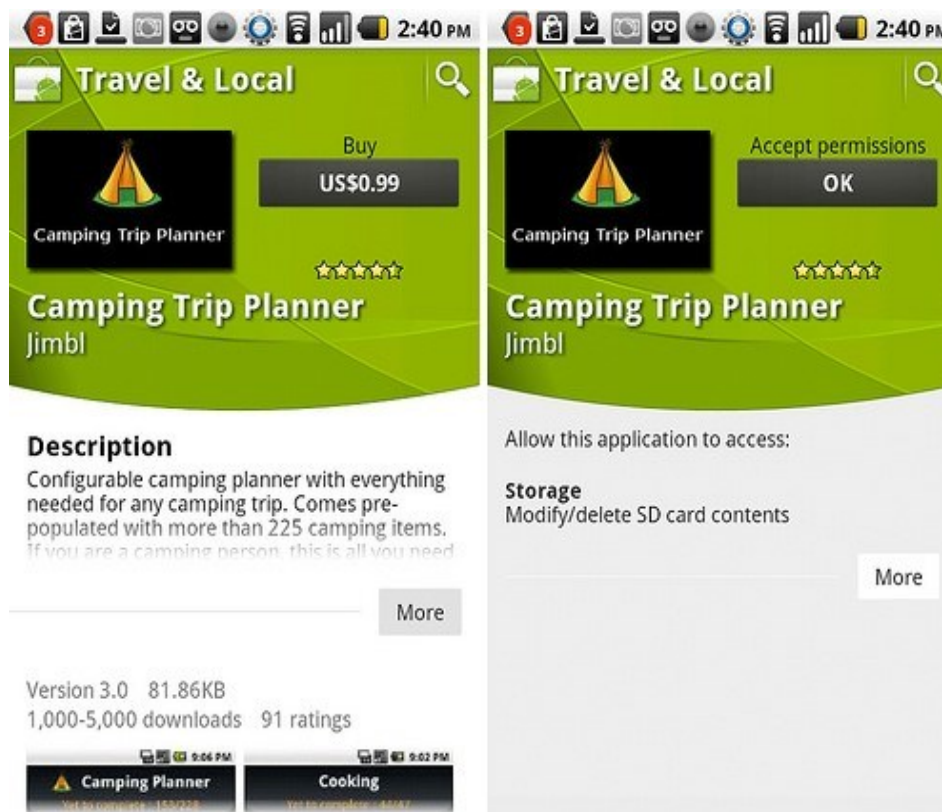
- ▶ **Don't hide the main call to action** in a menu or disguise it as an unrecognizable icon in a toolbar.
- ▶ **Good contrast and clear label!**



*Credits: Ivano Malavolta
Design Patterns for Mobile Apps*

Multi-state Buttons

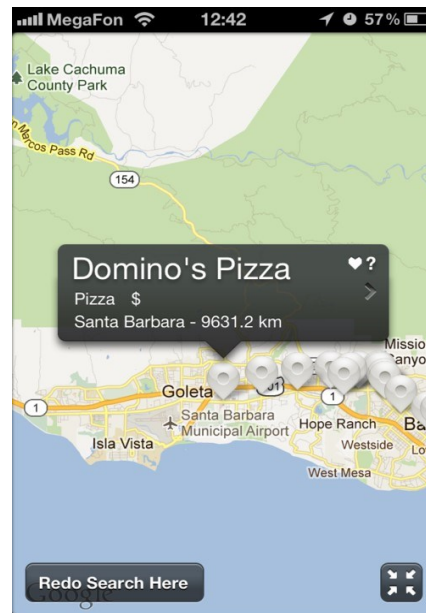
- ▶ Multi-State Buttons work well for a series of tightly correlated actions that will be performed in succession.



Credits: Ivano Malavolta
Design Patterns for Mobile Apps

Actions on Maps

- ▶ Provide visible markers (avoiding “terrain-mode”).
- ▶ Use as **much screen** as possible
- ▶ Can be small if the location is not the main item.



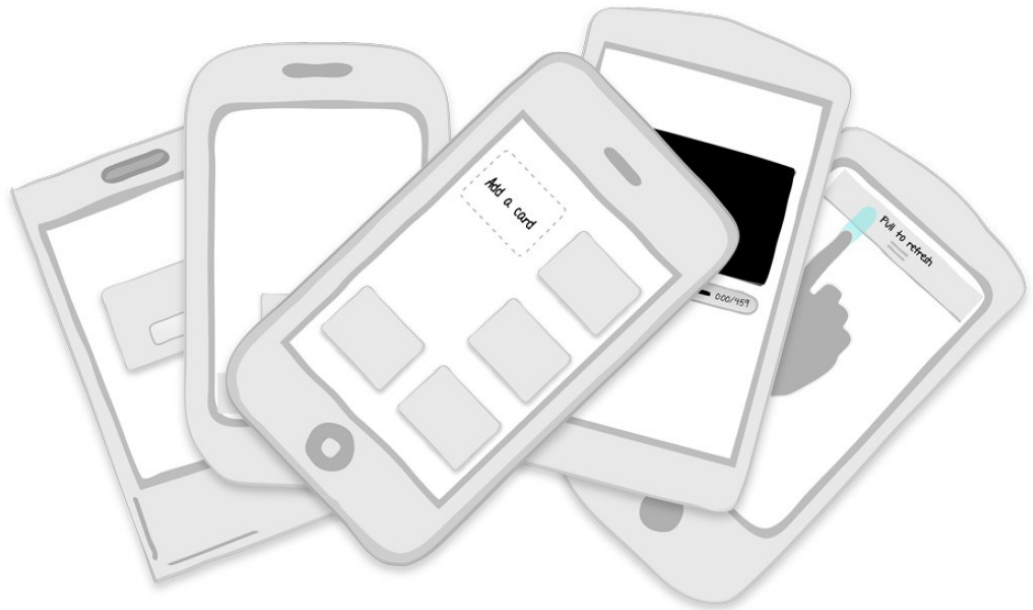
Credits: Ivano Malavolta
Design Patterns for Mobile Apps

Roadmap

- ▶ Navigation
- ▶ Tables
- ▶ Forms
- ▶ Search
- ▶ Tools
- ▶ **Helps and Tutorials**
- ▶ Feedback & Affordance

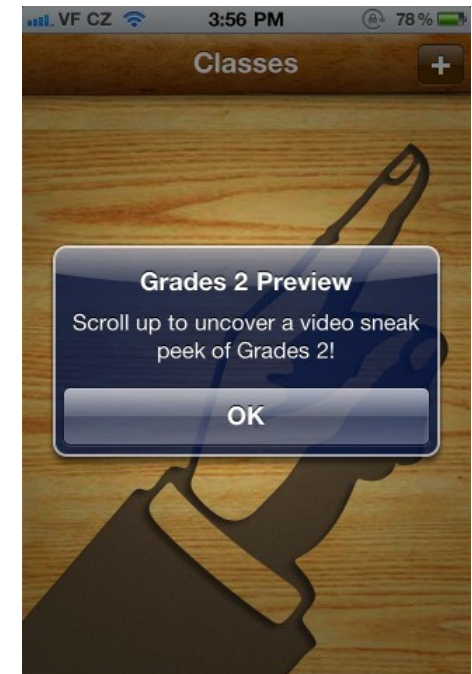
Helps and Tutorials

- ▶ Helpful tips displayed the first time a user launches an app.
 - ▶ Dialog
 - ▶ Tips
 - ▶ Tour
 - ▶ Video
 - ▶ Transparency



Dialog

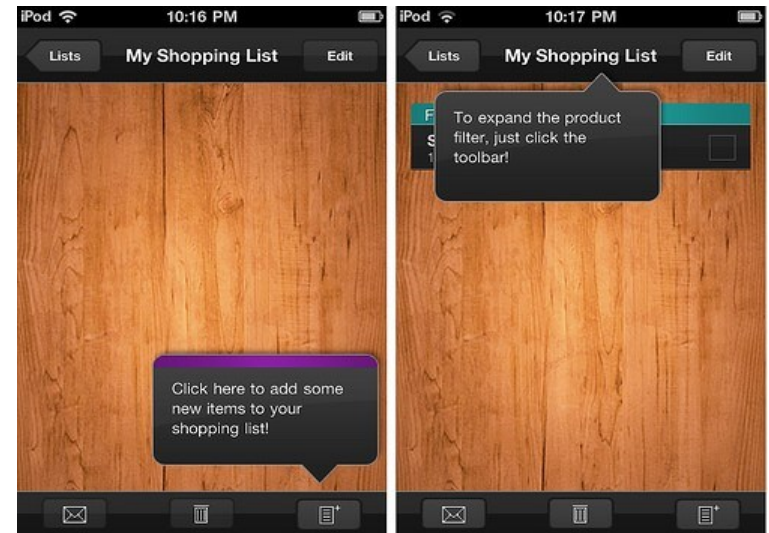
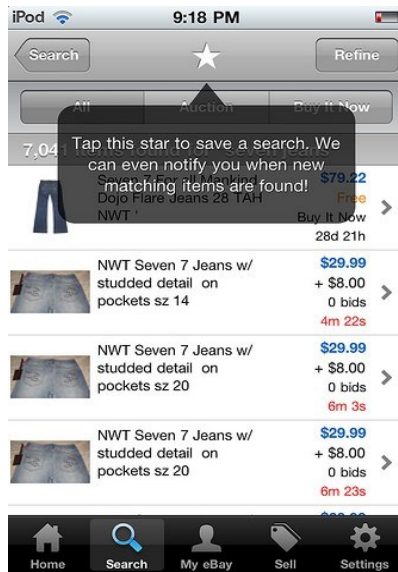
- ▶ Keep **dialog content short**.
- ▶ Make sure there is an alternate way to access instructions from within the application.



Credits: Ivano Malavolta
Design Patterns for Mobile Apps

Tips

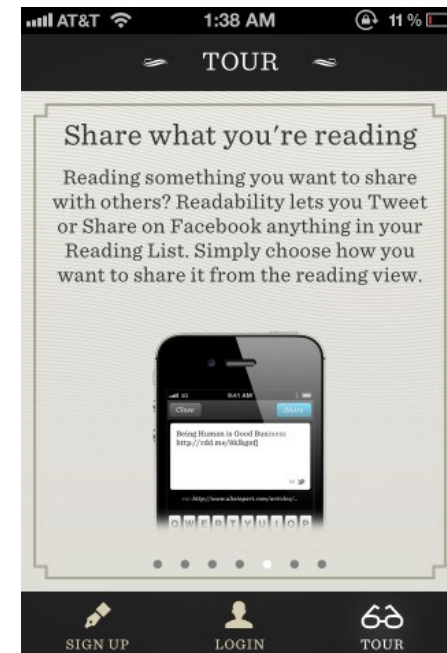
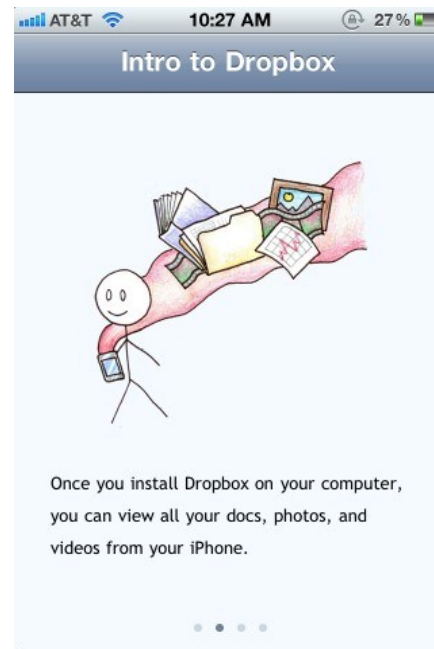
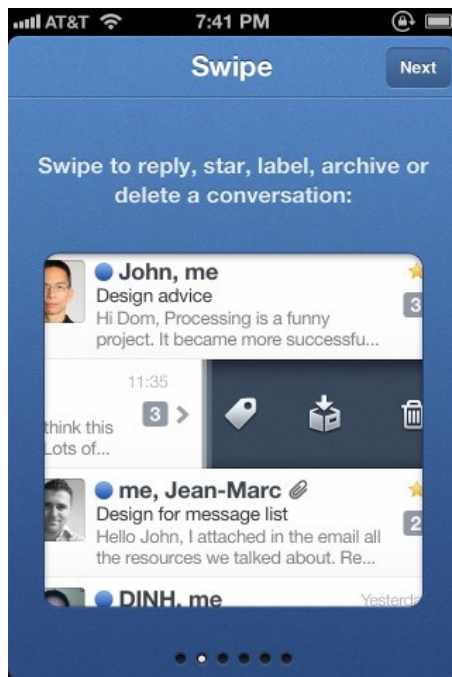
- ▶ Place tips in **proximity to the feature** they refer to.
- ▶ Keep the content short.
- ▶ **Remove the tip once interaction begins.**



*Credits: Ivano Malavolta
Design Patterns for Mobile Apps*

Tour

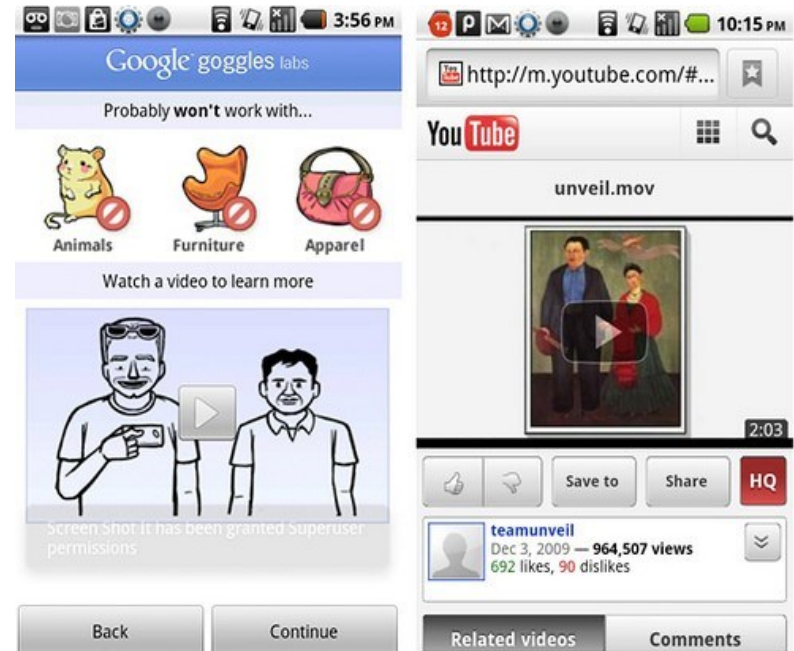
- ▶ A tour should **highlight key features** of the application, preferably from a (user) goal perspective.
- ▶ Keep it short and visually engaging.



*Credits: Ivano Malavolta
Design Patterns for Mobile Apps*

Video

- ▶ Demos and screencasts should showcase **key features** or show **how to use the application**.
 - ▶ Common video features (pause, volume controls, etc.) should be provided.



Transparency

- ▶ Transparencies are not meant to compensate for poor screen designs!
- ▶ Remove the Transparency once interaction begins.



*Credits: Ivano Malavolta
Design Patterns for Mobile Apps*

Roadmap

- ▶ Navigation
- ▶ Tables
- ▶ Forms
- ▶ Search
- ▶ Tools
- ▶ Helps and Tutorials
- ▶ **Feedback & Affordance**

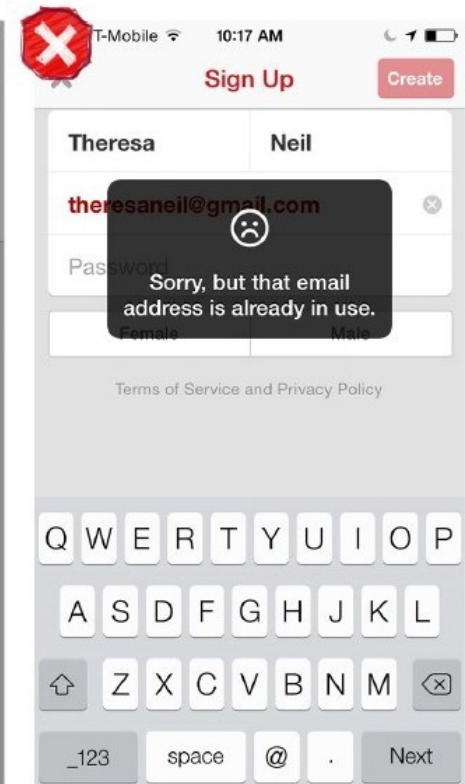
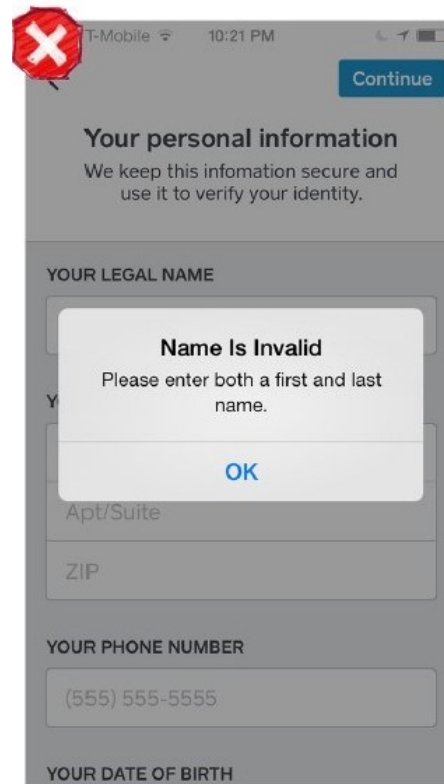
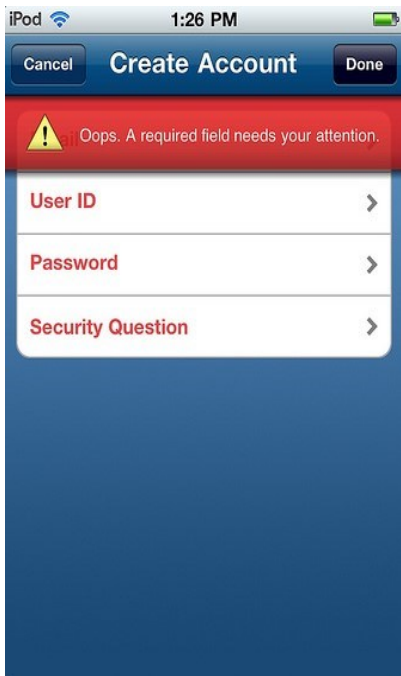
Feedback & Affordance

- ▶ Feedback
 - ▶ Errors
 - ▶ Confirmation
 - ▶ System Status
- ▶ Affordance
 - ▶ Tap
 - ▶ Flick
 - ▶ Drag



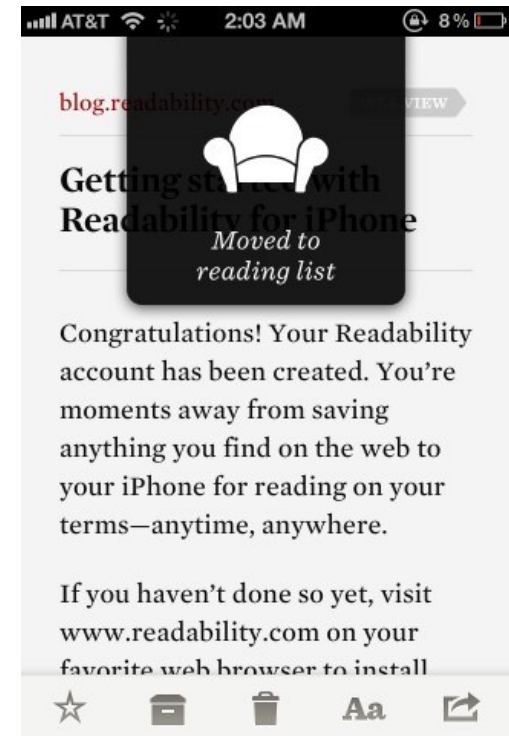
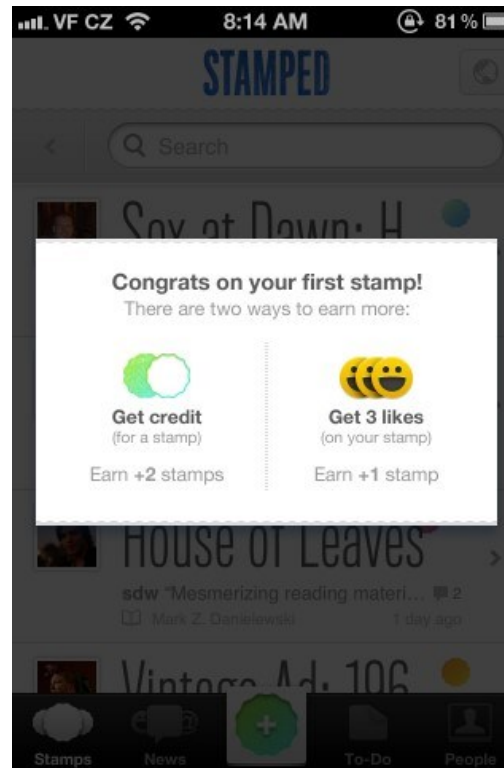
Feedback: Errors

- ▶ Use **plain language** that offers a solution for resolving the issue.
 - ▶ make the error **visible**.
 - ▶ use **in-screen messaging** instead of modal dialogs.



Feedback: Confirmation

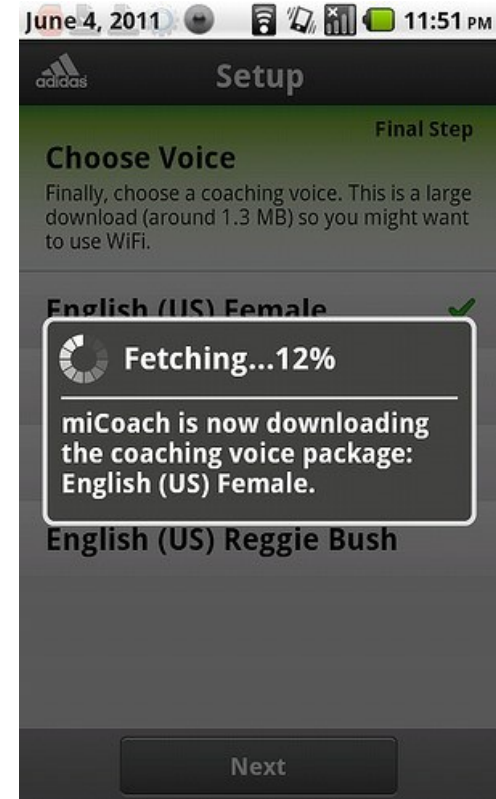
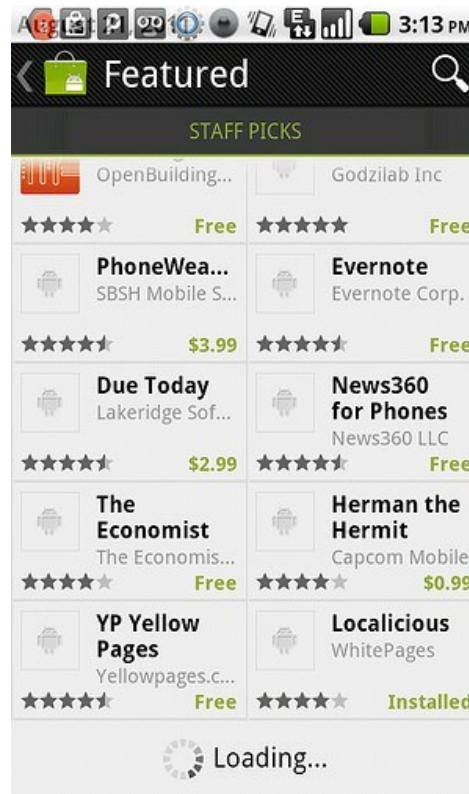
- ▶ Provide confirmation when an action is taken.
 - ▶ don't break the user's flow!



Credits: Ivano Malavolta
Design Patterns for Mobile Apps

Feedback: System Status

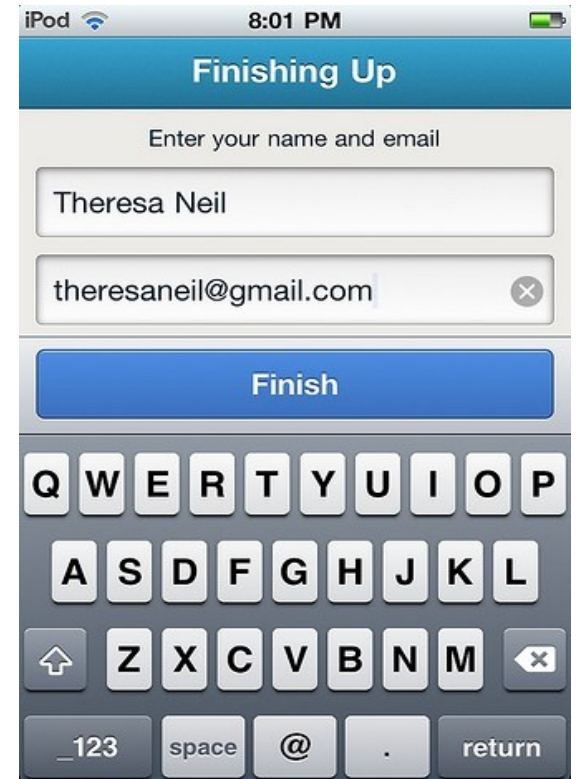
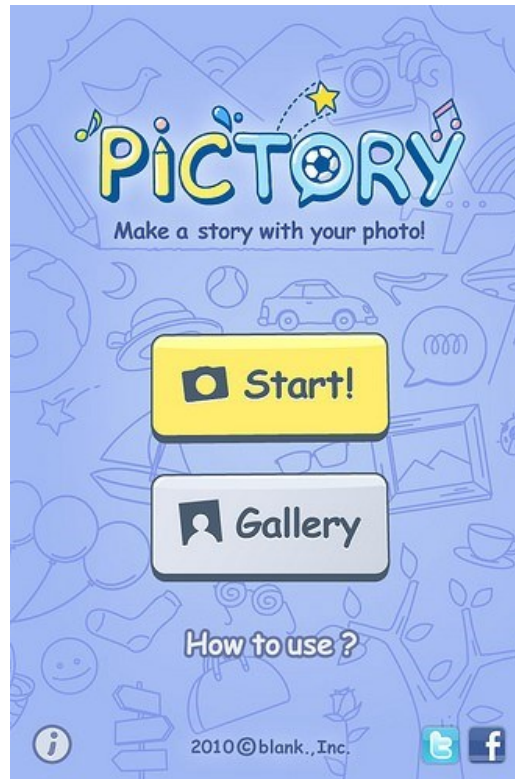
- ▶ Provide feedback about the system's status.
 - ▶ Offer a cancel option for potentially lengthy operations.



Credits: Ivano Malavolta
Design Patterns for Mobile Apps

Affordance: Tap

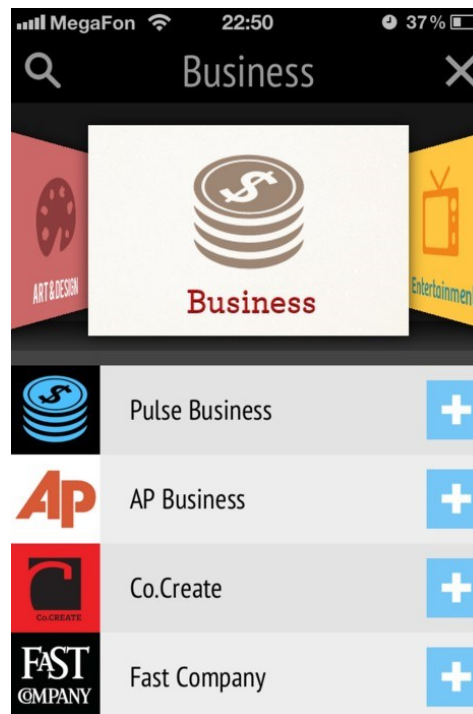
- ▶ Use common visual design techniques to indicate tappable controls.
- ▶ Apply 3D effects (for example, shadows) judiciously.



Credits: Ivano Malavolta
Design Patterns for Mobile Apps

Affordance: Flick

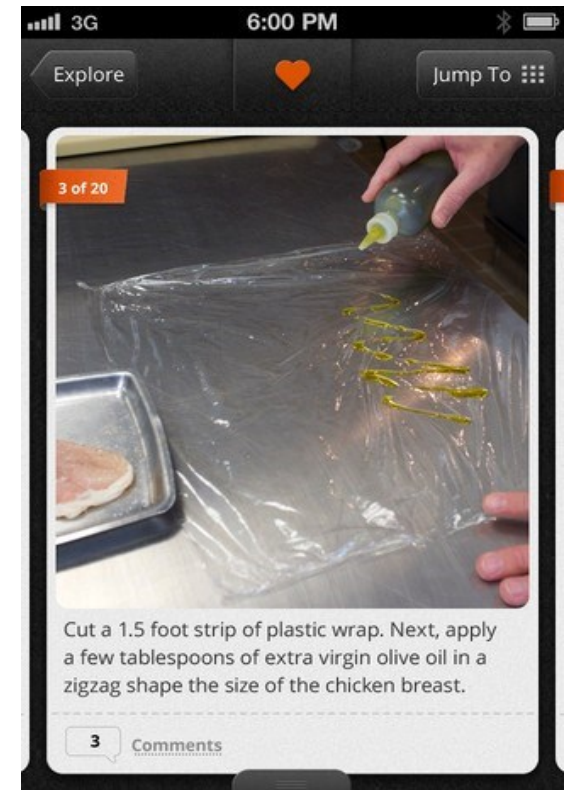
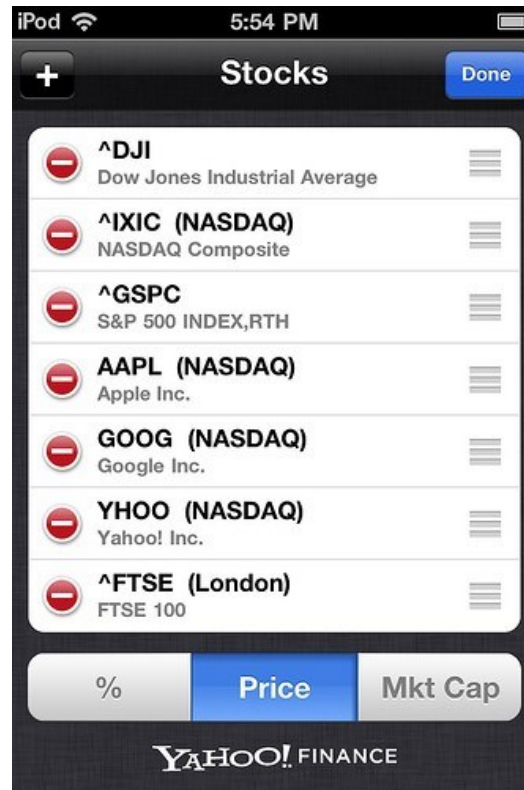
- ▶ Use a page indicator.
- ▶ Show the edge of the next item.
- ▶ Avoid heavy weight scroll bars.



Credits: Ivano Malavolta
Design Patterns for Mobile Apps

Affordance: Drag

- ▶ Use a **recognizable icon** for the handle.
- ▶ Consider to help the users to let them know this feature is available.



Credits: Ivano Malavolta
Design Patterns for Mobile Apps

Exercise

- ▶ **We want to create a series of mockups for a mobile banking application for iOS devices.**
 - ▶ Starting from the previous exercises, design a complete prototype of the banking application through Balsamiq.
 - ▶ Include a view for the deposit and a page for modifying the profile of logged users.

Bibliography

1. T. Neil: *Mobile Design Pattern Gallery* , 2nd Edition (2015)
2. I. Malavolta: Design Patterns for Mobile Apps (slides)

