

# Designing across mobile platforms & screens

UC Berkeley iSchool 08.28.2012

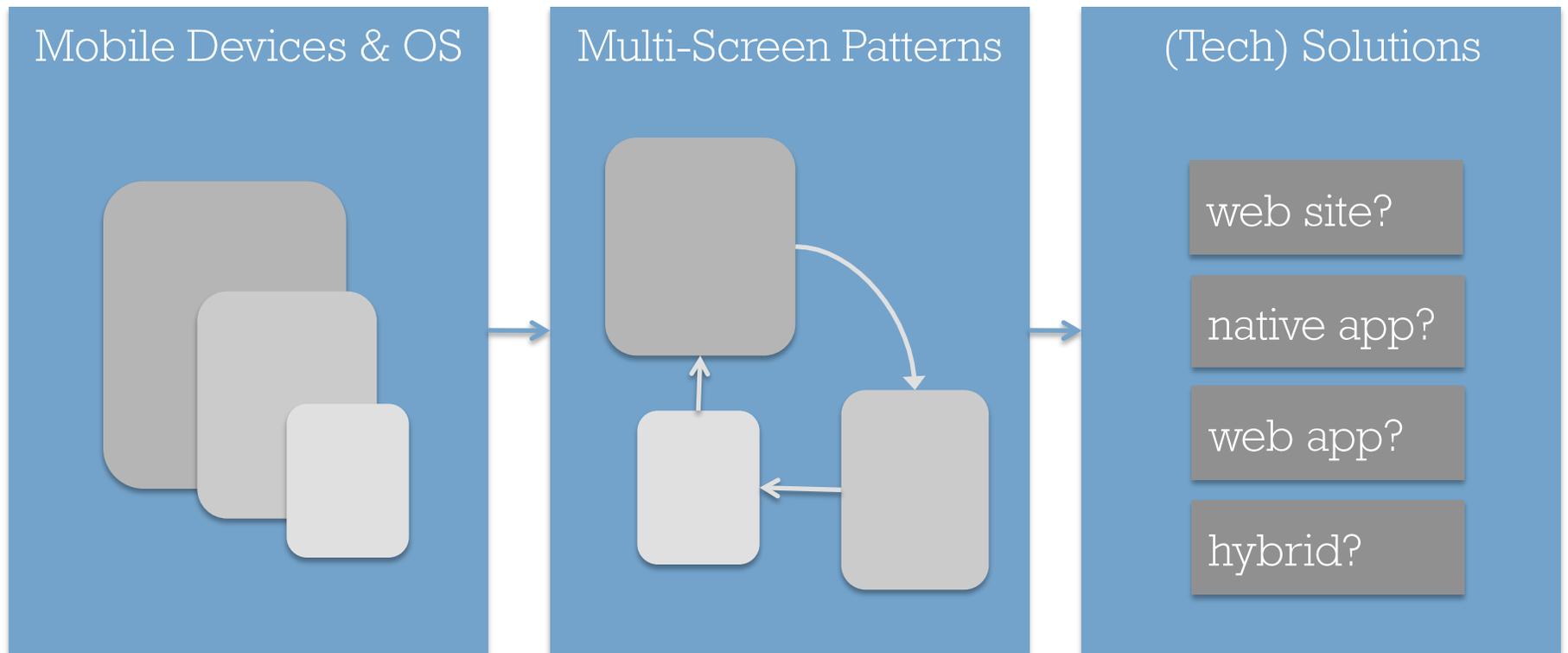


“**Convergence** is what enables experiences to shapeshift between different devices and environments. Instead of a user interacting with his mobile device as one isolated experience and then interacting with another device (such as a personal computer) as a totally separate, isolated experience, convergence allows these experiences to be connected and have continuity.”

“What is Convergence?”

*From Rachel Hinman's "The Mobile Frontier"*

# Lecture plan



# Mobile Devices & Operating Systems

10"

iPad 2 (2048 x 1536)

7"

Nexus 7 (1280 x 800)

Kindle Fire (1024 x 600)

iPad mini coming soon?

~4"

iPhone (960 x 640)

Galaxy (1280 x 720)

Lumia (480 x 800)

Common display sizes &  
their resolutions

# **What's inside:** features that impact the UX

## Common features that impact the UX (look up in tech specs)

Feature	What it does	Example app or feature
Camera*	Takes photos & often video.	Too many to list!
GPS	Provides the phone's location.	Maps
Accelerometer	Detects the phone's orientation.	Determines when to switch from portrait to landscape view.
Magnetometer	Detects the phone's direction.	Compass
Gyroscope	Detects 3-axis angular acceleration around the X, Y and Z axes.	Gaming
Proximity sensor	Detects proximity of user to phone.	Cue to dim screen when user speaking on phone.
Light sensor	Determines how much light is available in the area surrounding the phone.	Adjusts screen brightness to conserve battery.

**Barometer:** Measures atmospheric pressure. Used to determine your altitude and more accurately measure how quickly you're moving through an area.

*Galaxy Nexus*

**NFC:** Near field communication. Used to transfer data between devices. Most well-known Android feature is Google Wallet.

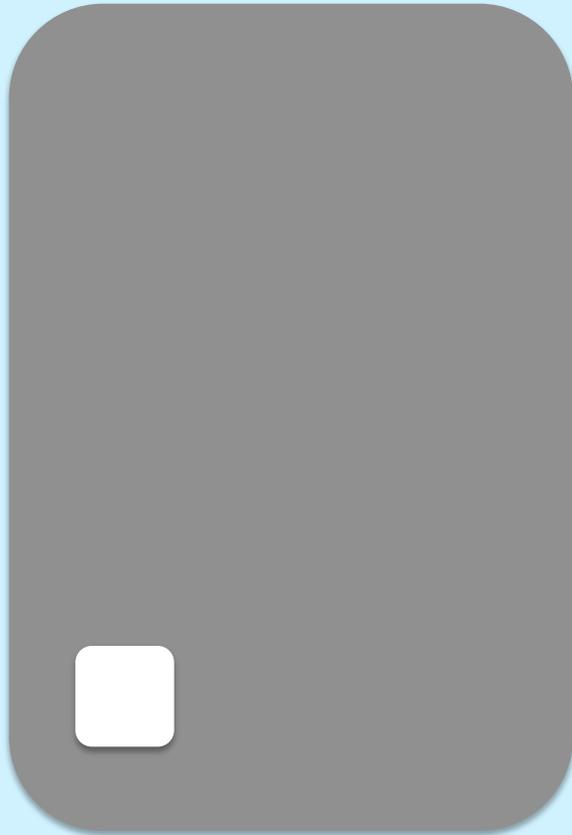
*Several Android devices*

Two noteworthy outliers

# Operating systems, specifically the *UI guideline* differences

## Global OS Differences:

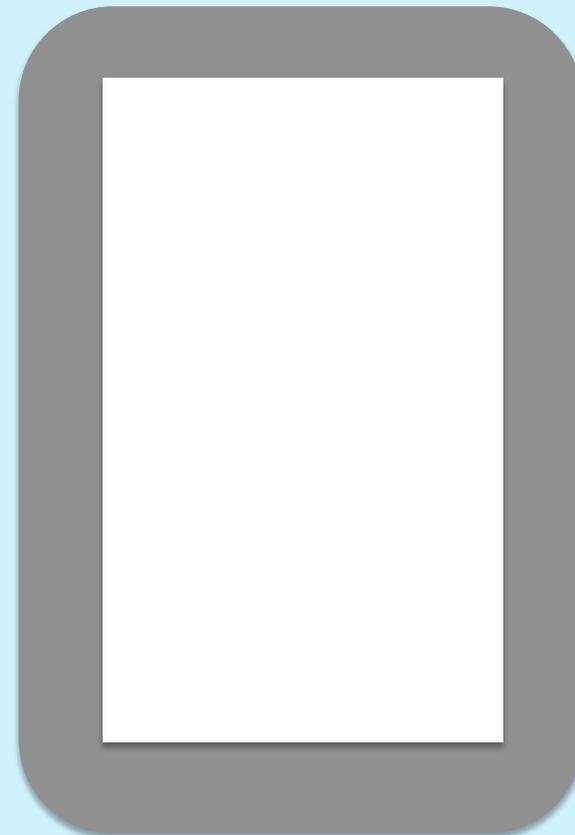
Interactions with other apps & the device



- Navigation
- Widgets
- Multi-Tasking
- Notifications

## Local OS Differences:

Interactions within your app

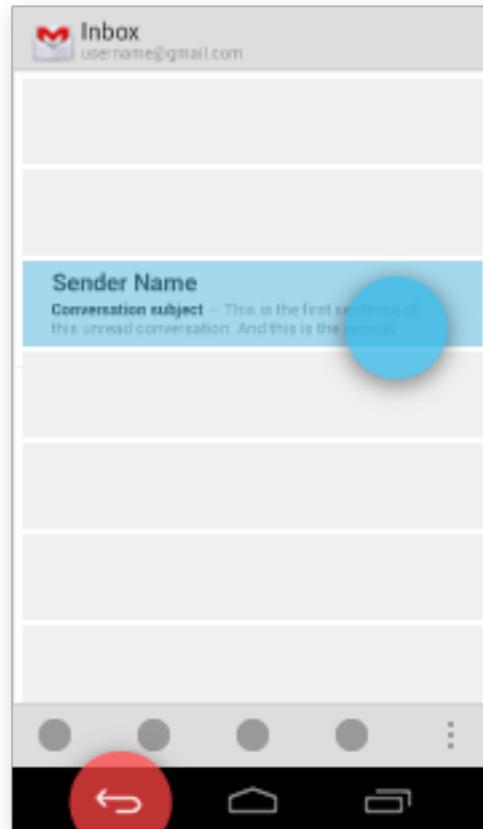


- Navigation
- UI Controls
- IA/Overflow
- Animations/Transitions

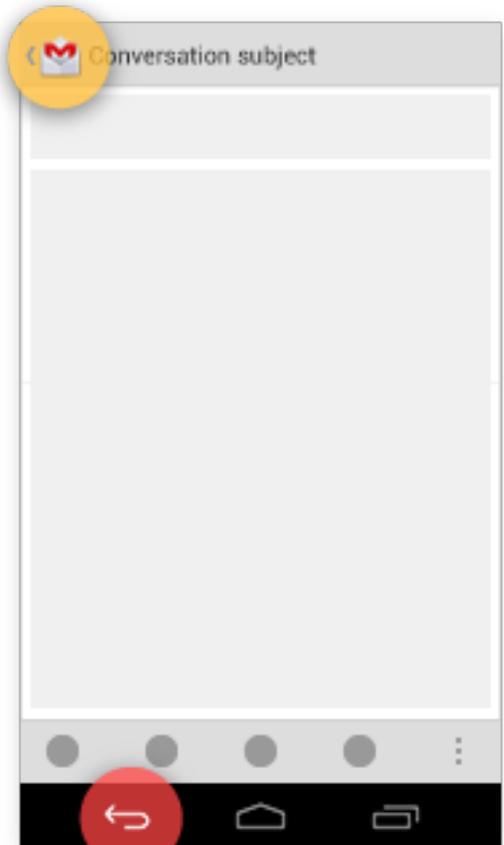
# Navigation on Android (Local & Global)



Home



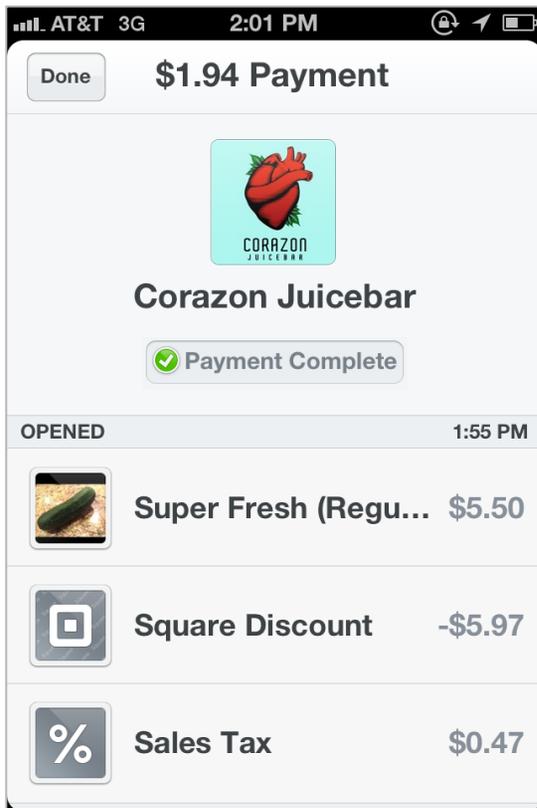
Conversation list



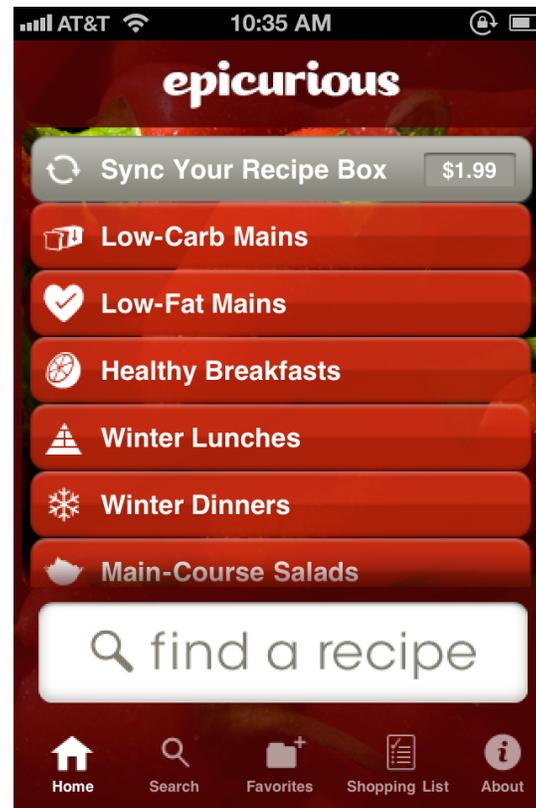
Conversation details



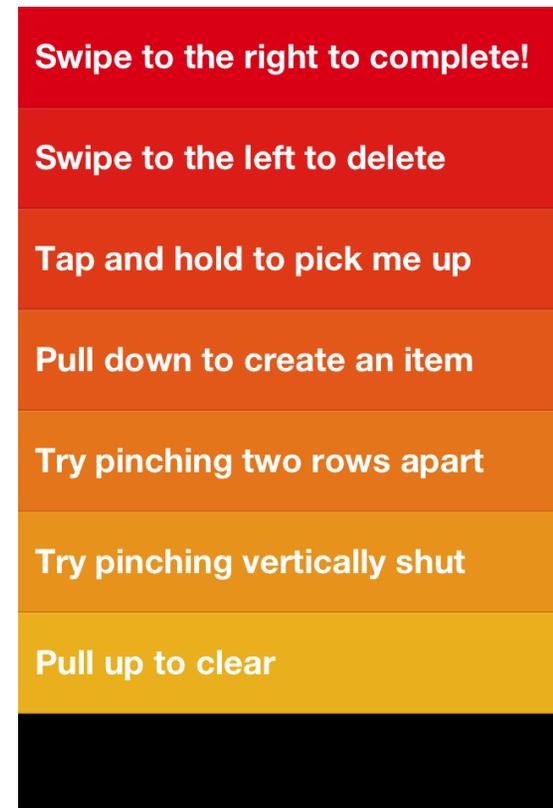
## Navigation on iPhone (Local)



Square



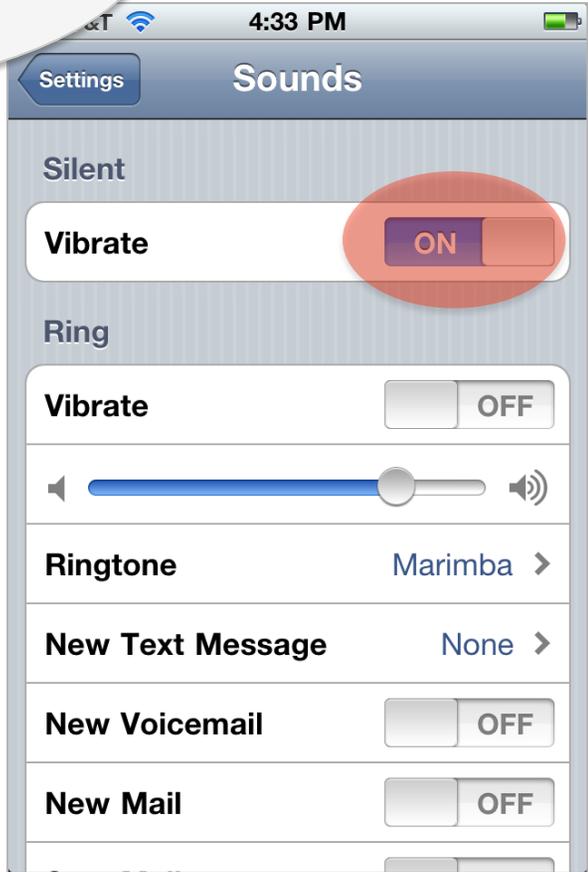
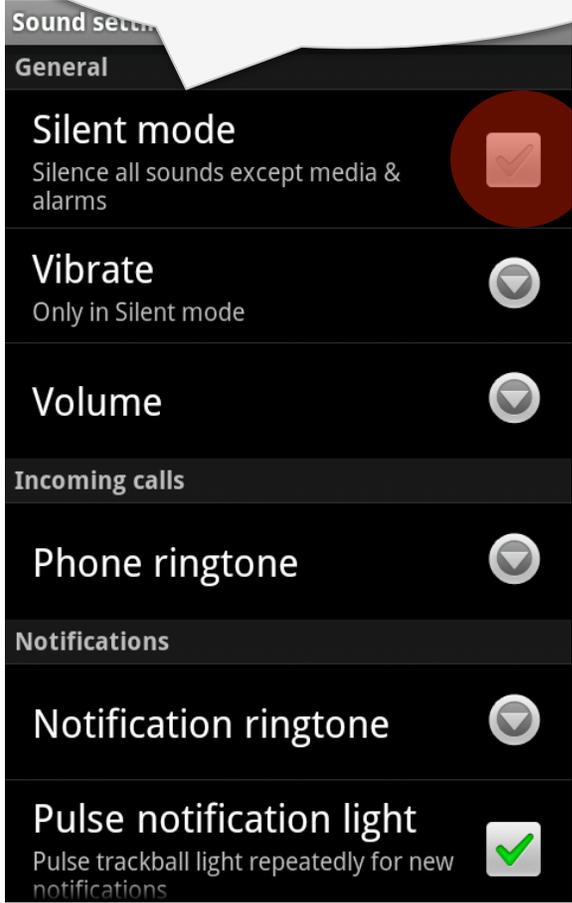
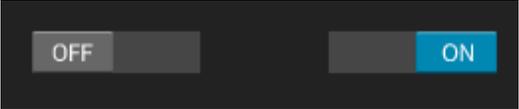
Epicurious



Clear

# UI Control Differences

**Correction:** latest version of Android now has switch like iOS

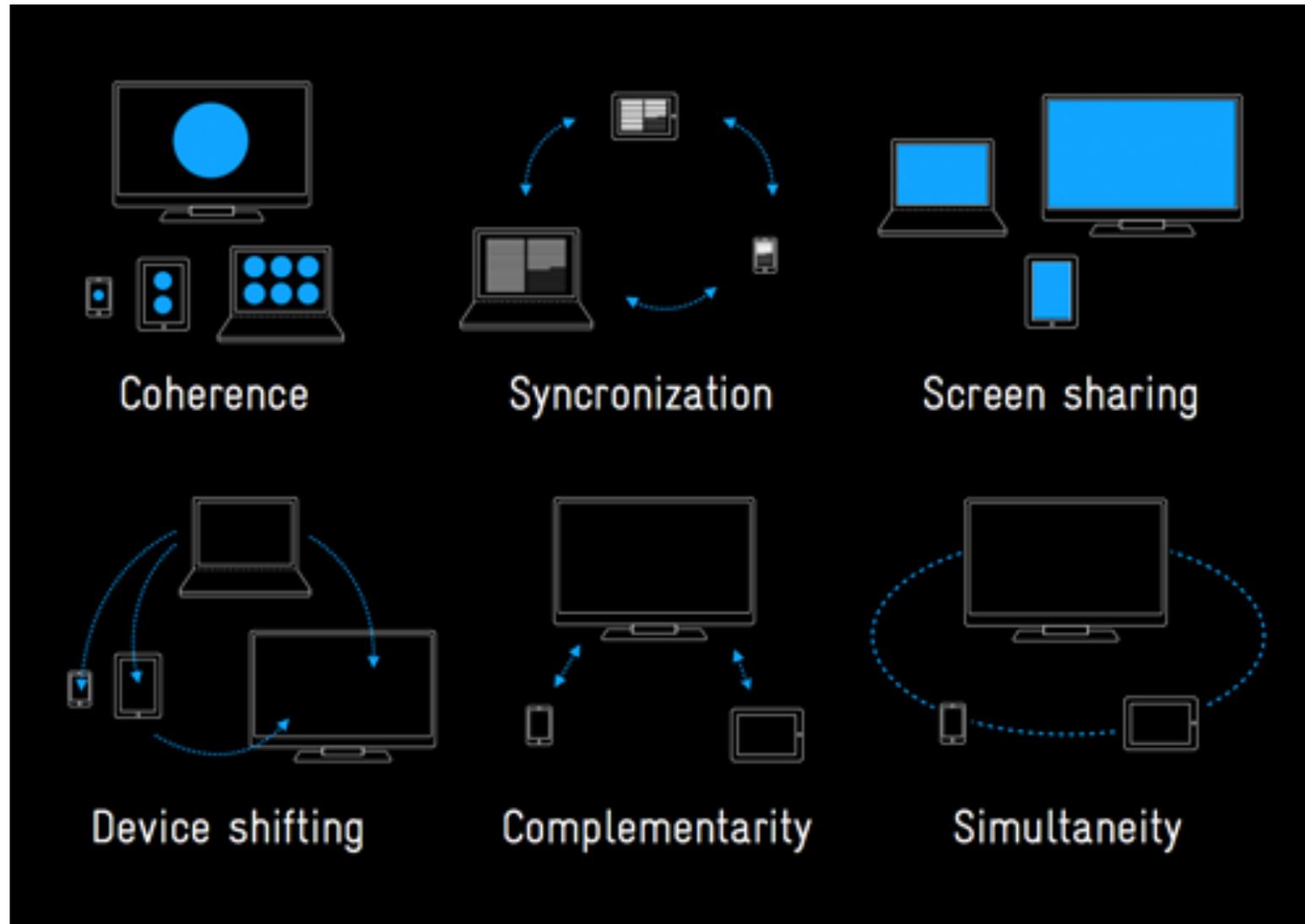


How to approach design when there are so many different devices & operating systems?

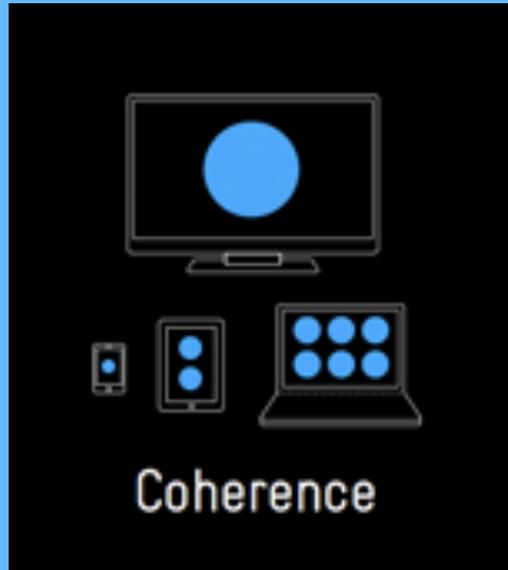
## Multi-screen Patterns

*From Precious Design in Hamburg, Germany*

*Explanatory text on next slides excerpted from "The Mobile Frontier" by Rachel Hinman*



## #1. Coherence

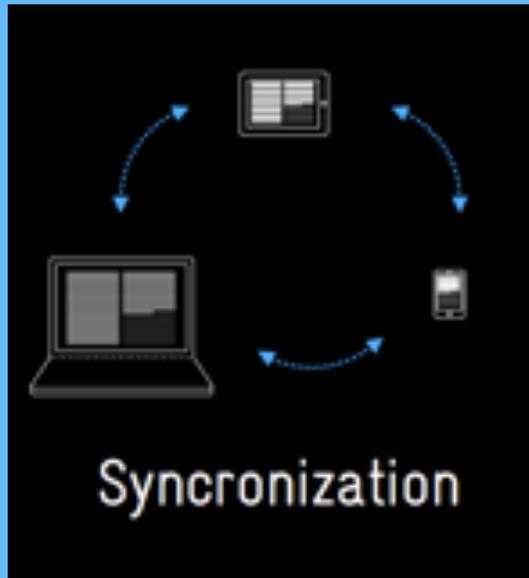


Optimizes digital experiences for specific device characteristics and usage scenarios while simultaneously ensuring there is a sense of continuity of the experience across all devices.



Evernote

## #2. Synchronization

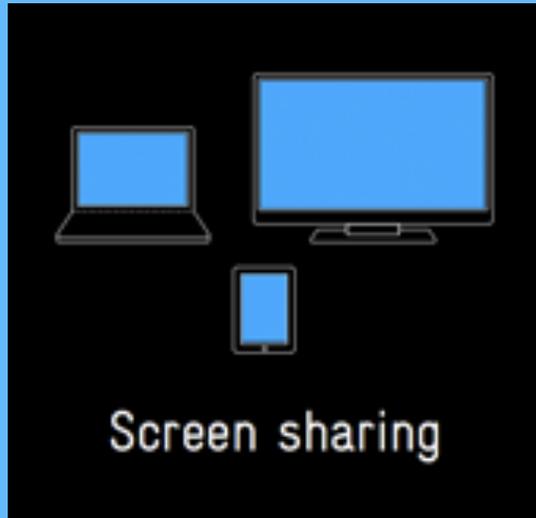


Keeps content in sync, regardless of the device for the sake of task continuation.



Kindle

### #3. Screen sharing

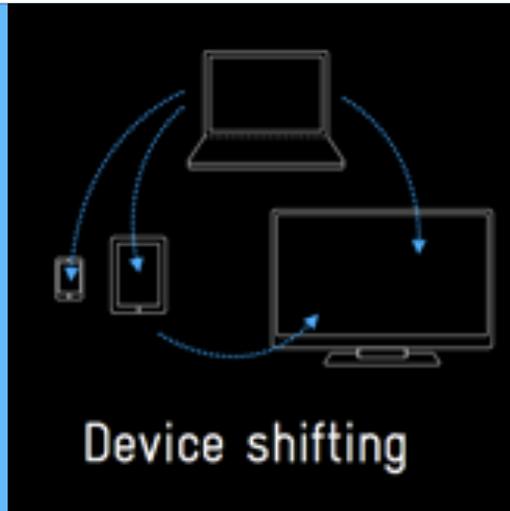


Multiple devices share the same content source. Similar to a patchwork quilt, each device in the ecosystem displays parts of the whole. Only when all the devices are together can the complete picture emerge.



Junkyard Jumbotron

## #4. Device Shifting



Shifts experience from one device to another seamlessly. In the case of Airplay, users may “broadcast” media from one device to another.

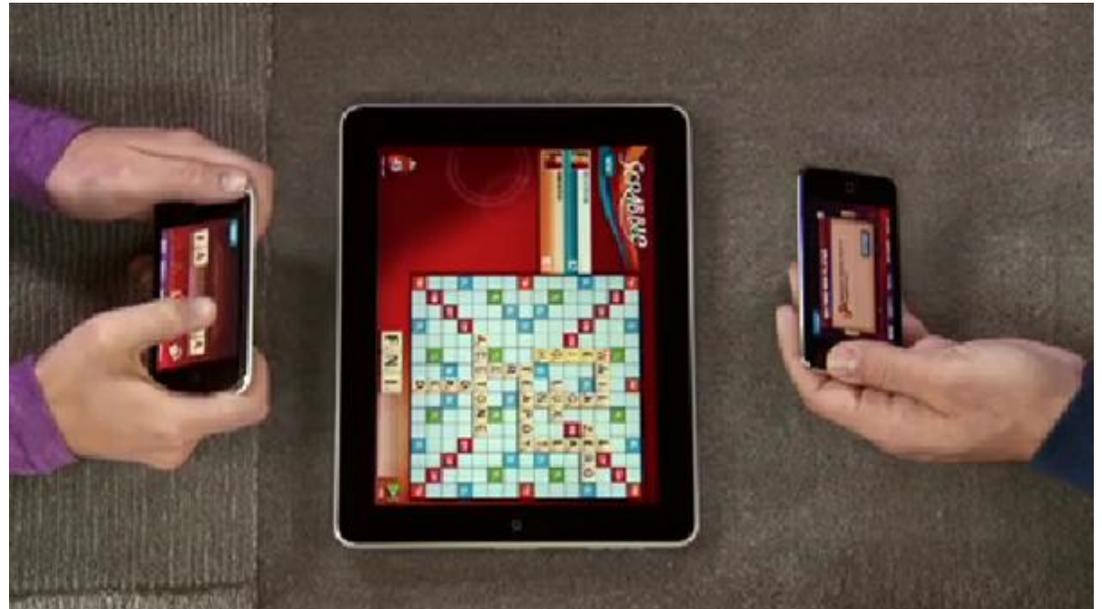


Apple's Airplay

## #5. Complimentary

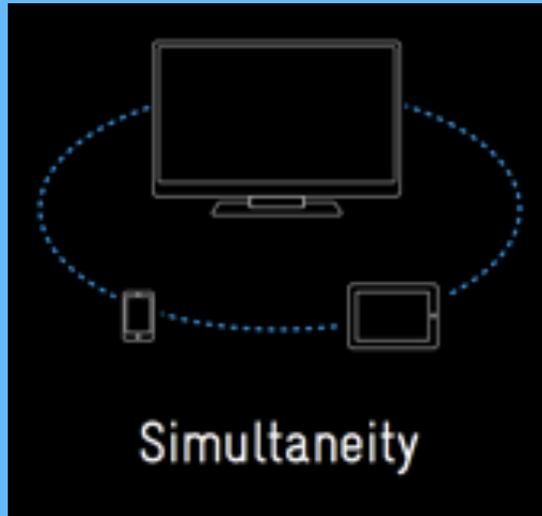


Two or more screens work together in concert to elevate an experience. Each device plays a specific role, but the roles support each other in a complimentary fashion.



Scrabble

## #6. Simultaneity



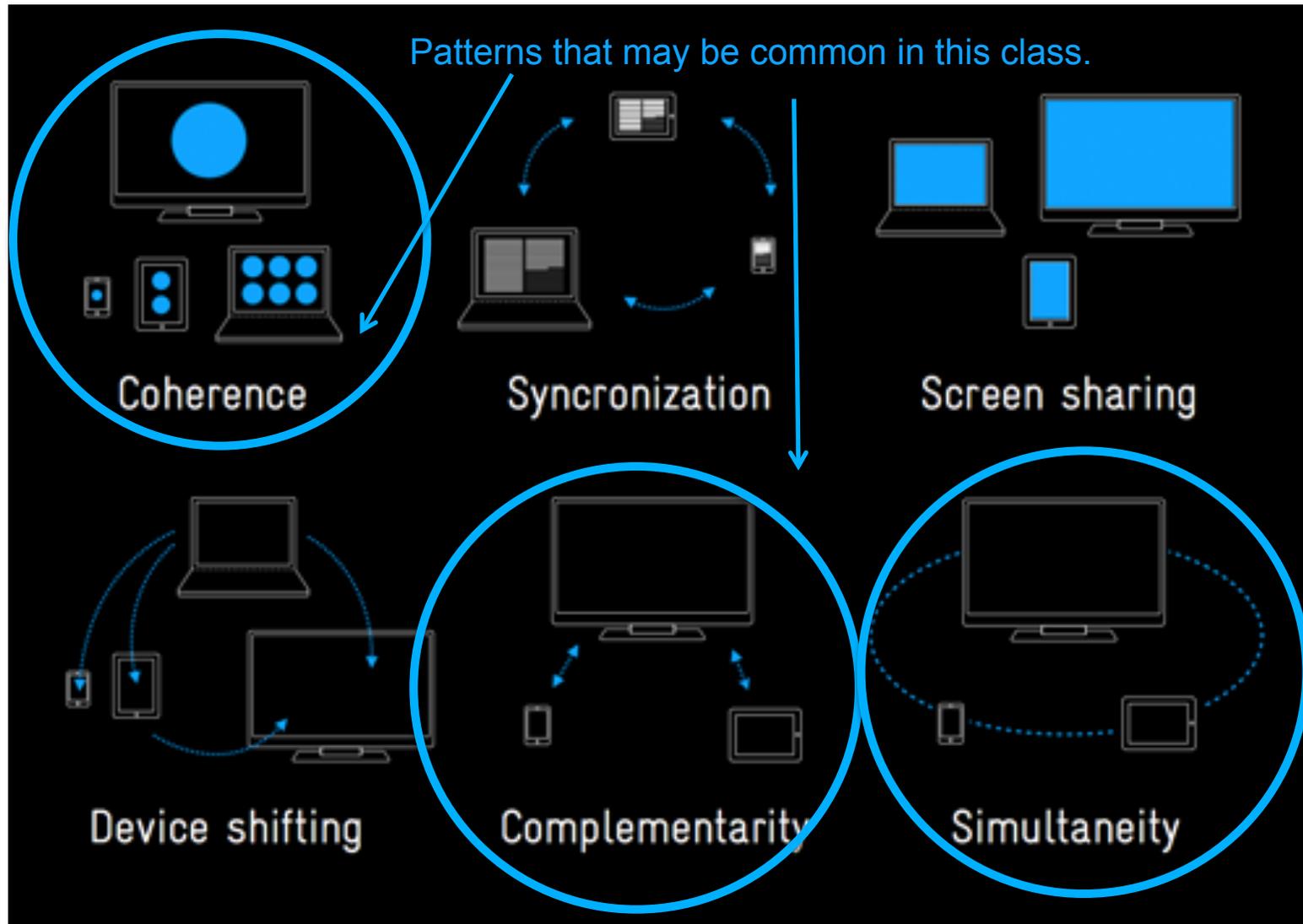
Provides users with two separate but connected experiences that can occur simultaneously.



Heineken Star Player

# Multi-screen Patterns

*From Precious Design in Hamburg, Germany*



Let's say based on your research, you plan to support Android & iOS tablets and phones as well as a web site for desktop access. So that's potentially **5 different UIs!**

Often the next big questions are:

“Should I design native, web, or hybrid apps?”

OR...

“Should I build a (*responsive*) web site for all?”

# Differences between Native & Web apps:

	Native	Web
Capabilities	Can interface with the device's native features, information and hardware (camera, accelerometer, etc.)	Mobile web apps can access a limited amount of the device's native features and information (orientation, geolocation, media, etc.)
Monetization	Developers have the ability to charge a download price and app stores will typically handle the payment process	Charging users to use the mobile web app requires you to set up your own paywall or subscription-based system
Method of Delivery	Downloaded onto a mobile device	Accessed through a mobile device's web browser
Performance	Typically perform faster than mobile web apps	Performance may be less optimal depending on features

From <http://sixrevisions.com/mobile/native-app-vs-mobile-web-app-comparison/>

# Development Differences:

Native	Web app
Each mobile application development platform (e.g. iOS, Android) requires its own development process	Runs in the mobile device's web browser and each may have its own features and quirks
Each mobile application development platform has its own native programming language: Java (Android), Objective-C (iOS), and Visual C++ (Windows Mobile), etc.	Mobile web apps are written in HTML5, CSS3, JavaScript and server-side languages or web application frameworks of the developer's choice (e.g. PHP, Rails, Python)
Standardized software development kits (SDKs), development tools and common user interface elements (buttons, text input fields, etc.) are often provided by the manufacturer of the platform	There are no standard software development kits (SDKs) that developers are required to use to make a mobile web app

There are tools and frameworks to help in developing apps for deployment on multiple mobile OS platforms and web browsers (e.g. PhoneGap, Sencha Touch 2, Appcelerator Titanium, etc.)

Let's imagine the worst case scenario—the **5**  
**different native UIs!**

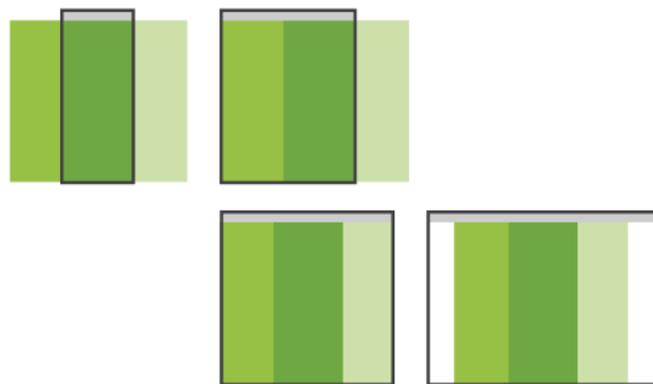
# Key differences that will impact your designs & how to deal with them:

**Display size & resolution:** Most apps will require different layouts and navigation schemes for the 3 key display sizes on the earlier slide: 4" vs. 7" vs. 10". (Some ideas from Luke W on next slide.)

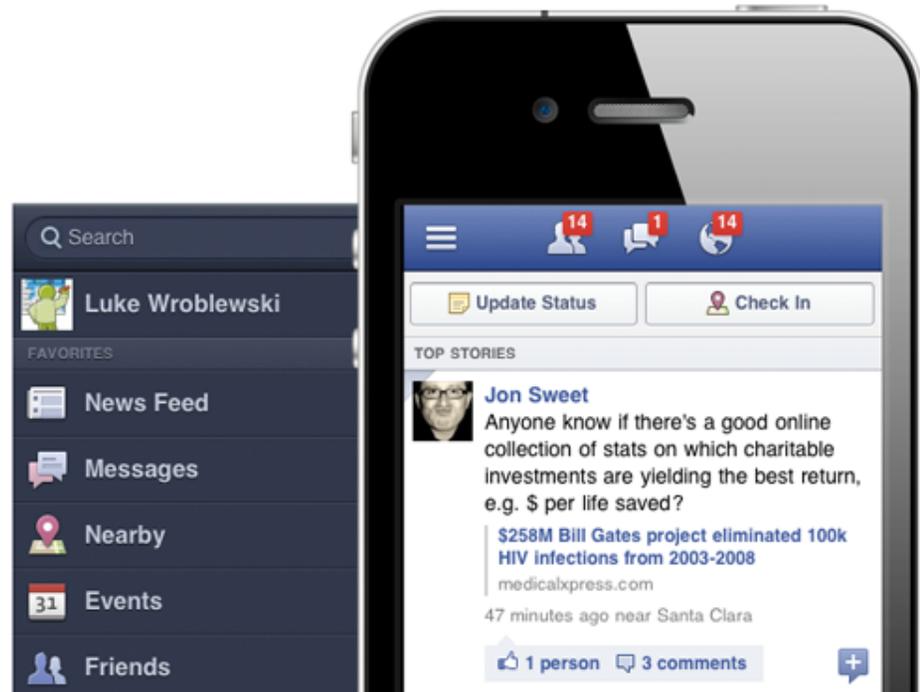
**OS differences:** Typically apps that closely follow platform-specific UI guidelines will need to use a distinct navigation approach for each OS they support. However, there are many custom navigation solutions that can work across platforms.

*(Notes: there are other differences, e.g., UI controls, that are easier to address. We will cover all of the above in more depth when we start to get into your apps' information architecture & screen level designs.)*

# Ways to deal with different display sizes & local navigation:



Off canvas



See more examples in “Multi-device layouts”: <http://www.lukew.com/ff/entry.asp?1514>

Or here from Brad Frost: <http://bradfrostweb.com/blog/web/responsive-nav-patterns/>

In addition to “pure” native or web apps, there are also a variety of “hybrid” tools that allow you to create web apps that look like native apps. They all use HTML for layout, JavaScript for device access, and CSS for look & feel.

# Hybrid Tools:

<http://www.phonegap.com>

<http://www.appcelerator.com/>

<http://www.rhobile.com/>

In summary, you have at least 4 options to consider in terms of tech solutions:

1. Responsive web site (*less likely in this class*)
2. Web app
3. Native apps
4. Hybrids

Please see the course web site for related readings & assignments from “The Mobile Frontier,” Lukew, Brad Frost & more...

<http://blogs.ischool.berkeley.edu/i290-dme-f12/>

Email: [ginsburg@ischool.berkeley.edu](mailto:ginsburg@ischool.berkeley.edu)

Twitter: [@suzanneginsburg](https://twitter.com/suzanneginsburg)