

**HIT3329 / HIT8329**  
**Creating Data Driven Mobile Applications**

# Lecture 8

# Modeling Application Design

# Around Availability

Presented by Paul Chapman

Adjunct Lecturer F-ICT  
Director, Long Weekend LLC  
Swinburne University of Technology

# Last Lecture Reviewed

1. External Libraries & Dependencies
2. Concurrent Operations
3. UI Feedback
4. Timers & Event Loops

**Questions?**

# What's On For Today?

## Design Considerations:

1. User Interface (UI)
2. User Experience (UX)

# 1.0 UI Design Considerations

1. Smaller Screens
2. No WIMP Interfaces
3. Constrained User Interface Elements
4. Touches & Gestures
5. The "Just Noticeable Difference" (jnd)

# 1.1 Smaller Screens

## iPhone Screen Resolutions:

- 320x480 (iPhone 2G/3G/3GS) = 1x
- 640x980 (iPhone 4G/5G, iPod 4G) = 2x

## iPad Screen Resolutions:

- 1024x768 (iPad, iPad2) = HD
- 2048x1536 (iPad3?)

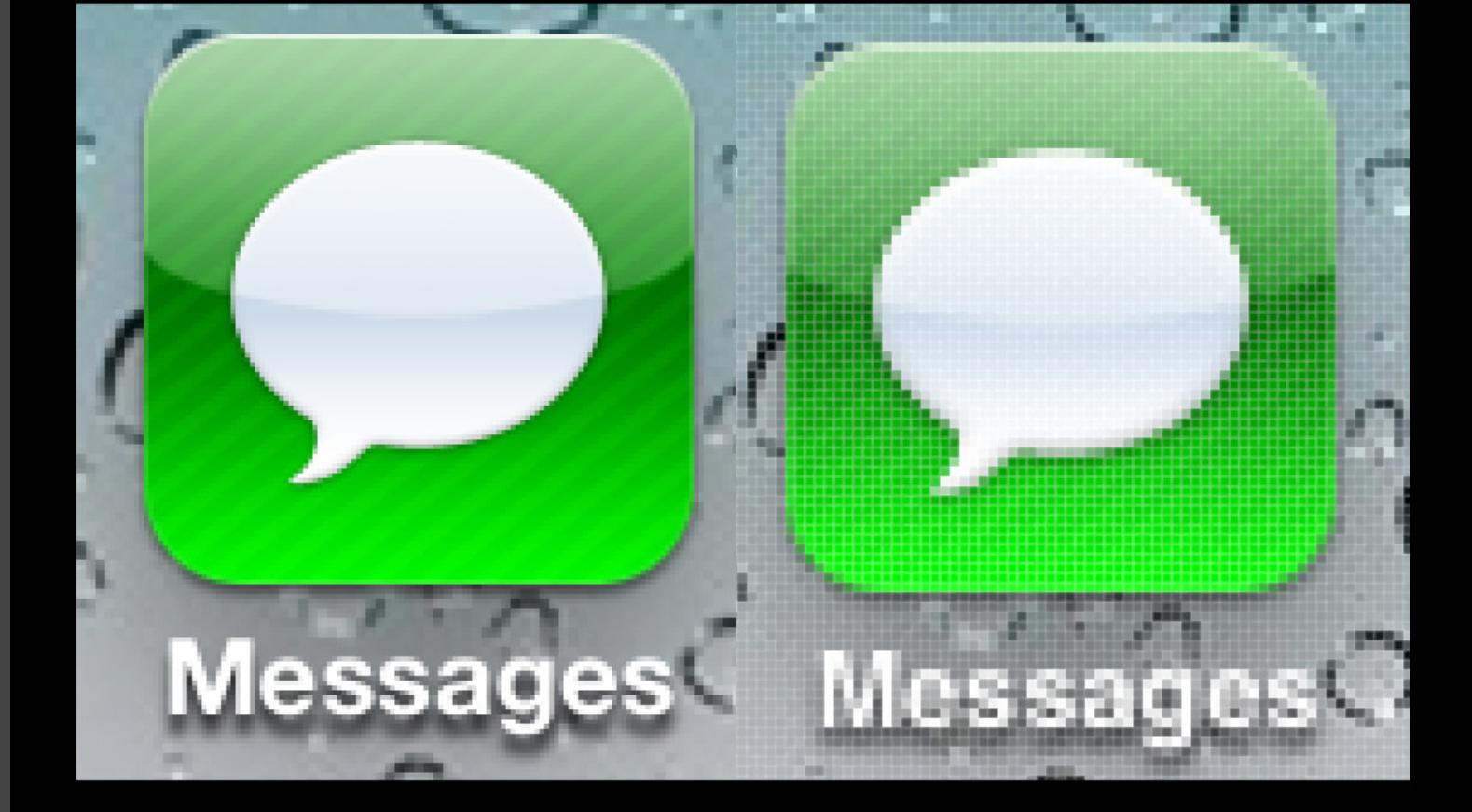
\*Apple named its 640x980 display a "Retina Display" based on the idea that it displays the maximum amount of detail a human eye can see from a distance of 12 inches (at 300dpi) **Source:** [http://en.wikipedia.org/wiki/IPhone\\_4](http://en.wikipedia.org/wiki/IPhone_4)

iPhone 4

w/retina display

iPhone 3GS

w/out retina display



Source: <http://hardforum.com/showthread.php?t=1528190>

### 1.1.1 Retina vs Non Retina Displays

## 1.2 No WIMP Interfaces

- WIMP = *Windows Icons Menus Pointers*
- Developed by Xerox PARC in the 1980s
- Widely popularised by Apple and Microsoft operating systems
- The standard for desktop computing



## 1.2.1 Example WIMP Interface

# 1.3 No WIMP Interfaces

On mobile devices

- **Windows** = can only show one at a time
- **Icons** = at most 16\* fit on a screen at once,  
less if you include *button / tool bars*
- **Menus** = menus are shallow (not nested)
- **Pointers** = your finger is the pointer

\*Excluding tablets of course

# 1.4 Constrained User Interface Elements

- Less ***Screen Real Estate*** to exploit means:
  - Less data can be displayed at once
  - Fewer buttons at each level
  - Nested menus impractical & unwieldy
- The ***Pixel Density*** of iOS devices is higher than most PC screens. So elements that look "big enough" on a PC might be tiny on the device
- When in doubt, follow Apple's ***Human Interface Guidelines*** (HIG) and established conventions



## 1.4.1 Shallow Menus, Less Data, Fewer Buttons

# 1.5 Touches & Gestures

Touch computing is changing the face of modern computing: "fingers are the new mouse!"

## Pros:

- Highly accessible, even babies can use them!
- Proxy free, no keyboard no mouse
- Natural input

## Cons:

- They are less accurate
- Fingers are differently sized depending on user
- Highly dependant on a user's eye sight too



## 1.5.1 Touch Interfaces Are Natural

# 1.6 The Just Noticeable Difference (jnd)

"jnd" describes the sensory ability to detect differences or changes between two stimuli

Commonly used by marketers to determine how much to change a product:

- Too much change and effort is wasted
- Not enough users don't notice
- Also reduces impact of negative changes

Source: [http://en.wikipedia.org/wiki/Just-noticeable\\_difference#Marketing\\_applications\\_of\\_the\\_j.n.d](http://en.wikipedia.org/wiki/Just-noticeable_difference#Marketing_applications_of_the_j.n.d)



Source: <http://grahamaustin.pbworks.com/w/page/10895808/Differential%20Threshold>

### 1.6.1 The 'jnd' illustrated - Coca Cola



Source: <http://grahamaustin.pbworks.com/w/page/10895808/Differential%20Threshold>

## 1.6.2 The 'jnd' illustrated - Campbell's Soup



Source: <http://marketingamateurs.blogspot.com/>

### 1.6.3 The 'jnd' illustrated - Firefox

The new, faster MacBook Air

The ultimate everyday notebook.  
Now up to 2.5x faster, featuring high-speed Thunderbolt I/O and OS X Lion.

**OS X Lion. Available Now.**  
Download it today from the Mac App Store.

**Watch the new iPad TV ad.**

**iCloud. Coming this fall.**  
iCloud stores your content and wirelessly pushes it to all your devices.

**Buy a new Mac for college and get \$100 in apps, music, and more.**

Shop the Apple Online Store (1-800-MY-APPLE), visit an Apple Retail Store, or find a reseller.

Site Map | Hot News | RSS Feeds | Media Info | Environment | Job Opportunities | Contact Us

Copyright © 2011 Apple Inc. All rights reserved. [Terms of Use](#) | [Privacy Policy](#) Choose your country or region ▾

The new, faster MacBook Air

The ultimate everyday notebook.  
Now up to 2.5x faster, featuring high-speed Thunderbolt I/O and OS X Lion.

**OS X Lion. Available Now.**  
Download it today from the Mac App Store.

**Watch the new iPad TV ad.**

**iCloud. Coming this fall.**  
iCloud stores your content and wirelessly pushes it to all your devices.

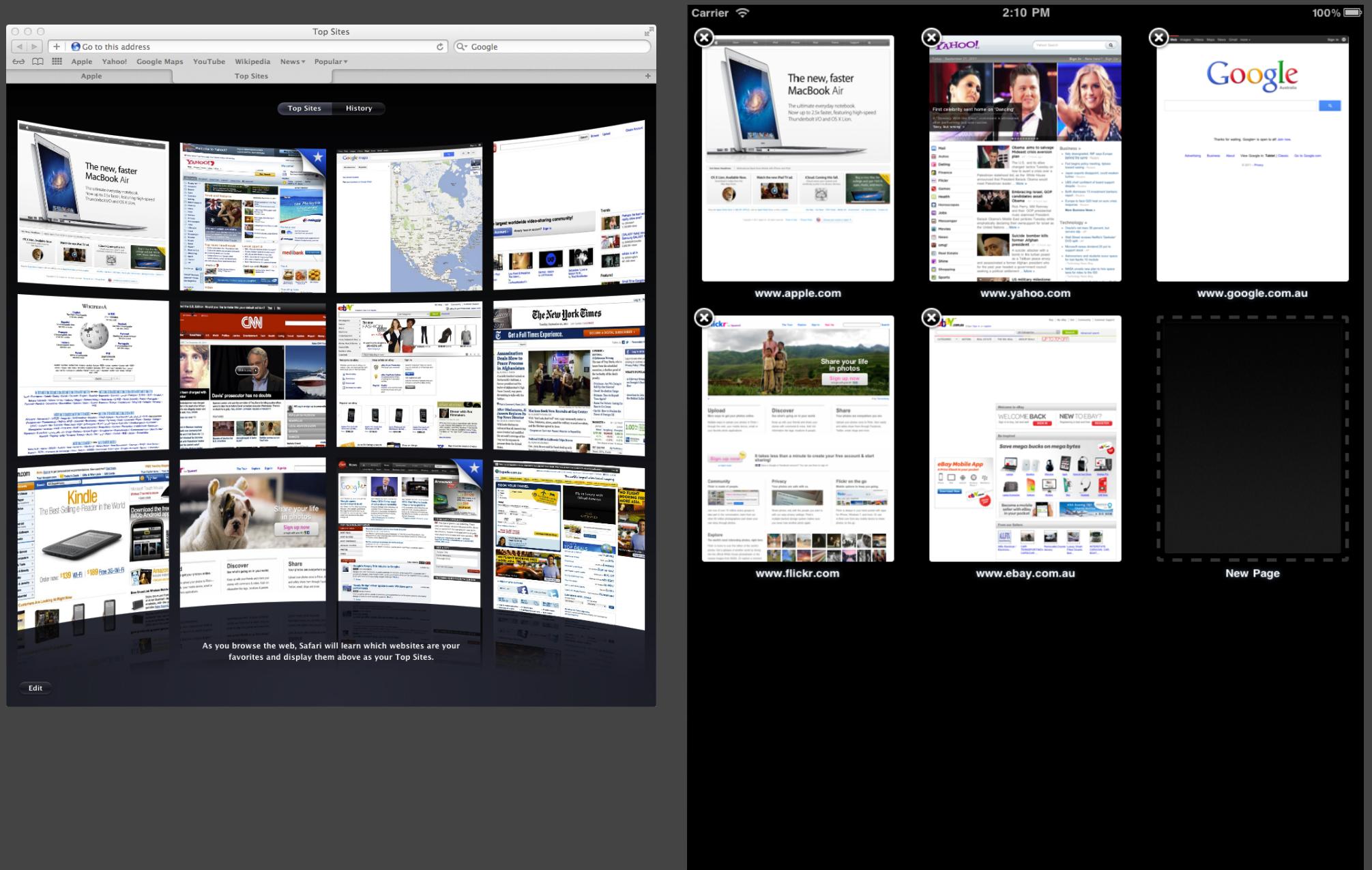
**Buy a new Mac for college and get \$100 in apps, music, and more.**

Shop the Apple Online Store (1-800-MY-APPLE), visit an Apple Retail Store, or find a reseller.

Site Map | Hot News | RSS Feeds | Media Info | Environment | Job Opportunities | Contact Us

Copyright © 2011 Apple Inc. All rights reserved. [Terms of Use](#) | [Privacy Policy](#) Choose your country or region ▾

# 1.6.4 Safari OS X Lion vs. Safari iOS 4



# 1.6.5 Safari OS X Lion vs. Safari iOS 4



# 1.6.6 Safari iOS 4 vs. Safari iOS 5

# What's On For Today?

## Design Considerations:

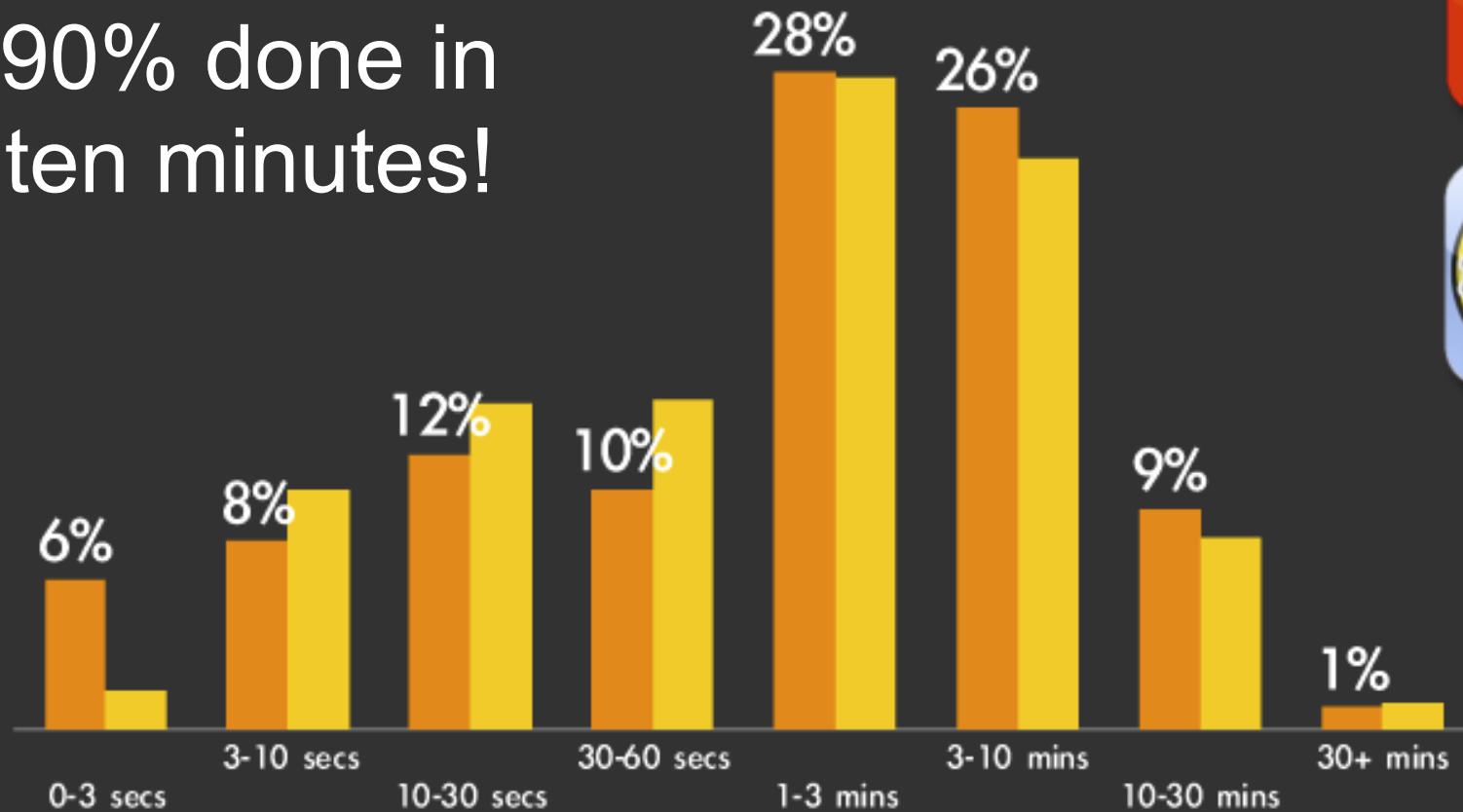
1. User Interface (UI)
2. User Experience (UX)

# 2.0 UX Design Considerations

1. Short User Sessions
2. Focus Your Features
3. The Phone Can Ring
4. Optimisation & Caching
5. Visual Feedback
6. Building for Availability

## 2.1 Short User Sessions

90% done in  
ten minutes!



## 2.2 Focus Your Features

LESS SCREEN SPACE  
SHORTER SESSIONS 

---

Difficult to Get the User's Attention

---

1. Do one thing really well
2. Remove non-core features\*
3. Decide your MVP early on

*\*Remember, you're making an app, not a new operating system*

## 2.3 Know Your MVP

### Minimum Viable Product

The minimum set of features and level of "polish" you can reasonably release to users.

A clearly defined MVP gets your app to users sooner and helps avoid features no one wants.

Popularised by Eric Ries in web app development

Further Reading: [http://en.wikipedia.org/wiki/Minimum\\_viable\\_product](http://en.wikipedia.org/wiki/Minimum_viable_product)

## 2.4 The Phone Can Ring

Unlike a PC chained to a desk and plugged into a wall, your app's normal operation is regularly interrupted by:

- Incoming phone calls
- Notifications
- Network drop outs
- Low battery
- GPS unavailability
- Insufficient memory

## 2.5 Dealing with the Unexpected

- Confirm network connections before use
- Implement delegate methods for entering foreground/background
- Don't expect GPS to always work
- Consider battery consumption (e.g GPS, data streaming, CPU intensive tasks)
- **Always** consider the usage context of your app and the "typical" user's expectations

## 2.6 Optimisation & Caching

- Simulator Testing  $\neq$  Device Testing
- Elegant Code vs. Premature Optimisation
- For now, assume all your assumptions about performance are incorrect
- This is not an excuse to write leaky code

## 2.7 Where to Optimise

### Caching Downloaded Data

- Test your data source for responsiveness
- Test to find optimal download batch size  
(e.g. up to 50 tweets at a time)

### Database Access

- Limit result set size
- Avoid wasteful search conditions
- Use indexes and keys correctly

## 2.8 Where to Stabilise

- Manage memory correctly, avoid leaks
- Avoid unnecessary object creation
- Find processing bottlenecks and refactor
- Tools to help you stabilise your app:
  - Leaks
  - Allocations
  - Time Profiler
- Run on simulator and tethered devices

## 2.9 Visual Feedback\*

- During lengthy loading or processing cycles, communicate it visually:
  - Modal notification overlay
  - Loading spinner in sub-view
  - Network access spinner
- Build in loading timeouts or add a cancel button to avoiding "spinning forever"

\*This was discussed previously in lecture 7

## 2.10 Building For Availability\*

- Avoid blocking the UI thread
- Use background processing as needed, confirm it is performant on devices
- Cancel objects' delegate relationships before they are deallocated (e.g. when pressing a view controller's back button)

\*This was discussed previously in lecture 7

# What We Covered Today

## Design Considerations:

1. User Interface (UI)
2. User Experience (UX)

# End of Lecture 8

1. Lab
2. Assignments
3. Dinner time