
Mobile Development

Lesson 2 - Jan 28, 2016

Plan today

1. Revisit XCode
 2. Github pointers
 3. Rebuild our sample app from lesson 1
 4. Put it on your devices!
 5. Announce your first assignment
 - a. Extra credit ideas
-

Toolbar area

Navigation area

Editor area

Inspector area

Library area

Debugging area

- 1 Show the Project navigator ⌘1
- 2 Show the Symbol navigator ⌘2
- 3 Show the Find navigator ⌘3
- 4 Show the Issue navigator ⌘4
- 5 Show the Test navigator ⌘5
- 6 Show the Debug navigator ⌘6
- 7 Show the Breakpoint navigator ⌘7
- 8 Show the Report navigator ⌘8
- 9 Show the Standard editor ⌘9
- 10 Show the Assistant editor ⌘10
- 11 Show the Version editor ⌘11
- 12 Hide/show the Navigator ⌘12
- 13 Hide/show Debug area ⌘13
- 14 Hide/show the Utilities ⌘14
- 15 Show the File inspector ⌘15
- 16 Show Quick Help inspector ⌘16
- 17 Show the identity inspector ⌘17
- 18 Show the Attributes inspector ⌘18
- 19 Show the Size inspector ⌘19
- 20 Show the Connections inspector ⌘20
- 21 Show the File Template library ⌘21
- 22 Show the Code Snippet library ⌘22
- 23 Show the Object library ⌘23
- 24 Show the Media library ⌘24

Apple Developer Site

<https://developer.apple.com/devcenter/ios/index.action>

The screenshot shows the Apple Developer website's iOS Dev Center. The header includes the Apple Developer logo, navigation links for Technologies, Resources, Programs, Support, and Member Center, and a search bar. Below the header is a dark navigation bar with 'iOS Dev Center' highlighted, along with links for Mac Dev Center and Safari Dev Center. The main content area is divided into three columns. The left column features a 'Resources for iOS 8' section with links to Downloads, the iOS Developer Library (including Getting Started, Guides, Reference, Release Notes, Sample Code, Technical Notes, and Technical Q&As), and Development Videos (including iOS Development and WWDC Videos). The middle column has a 'Featured Content' section listing links to iOS 8 for Developers, What's New in iOS, Adaptive User Interfaces, App Extensions, Apple Pay, CloudKit, Handoff, HealthKit, HomeKit, and a 64-Bit Transition Guide. The right column contains an 'iOS Developer Program' section with links to Certificates, Identifiers & Profiles, iTunes Connect, Apple Developer Forums, and the Developer Support Center. Below this is the 'App Store Resource Center' section, which includes links to Prepare for App Submission and the App Store Approval Process.


Developer Technologies Resources Programs Support Member Center


iOS Dev Center iOS Dev Center Mac Dev Center Safari Dev Center

[My Profile](#) [Sign out](#)


iOS 8.1 SDK

Resources for iOS 8

 **Downloads**
Download the latest build of iOS SDK











 **iOS Developer Library**

- Getting Started
- Guides
- Reference
- Release Notes
- Sample Code
- Technical Notes
- Technical Q&As

 **Development Videos**

- iOS Development
- WWDC Videos



Featured Content

-  [iOS 8 for Developers](#)
-  [What's New in iOS](#)
-  [Adaptive User Interfaces](#)
-  [App Extensions](#)
-  [Apple Pay](#)
-  [CloudKit](#)
-  [Handoff](#)
-  [HealthKit](#)
-  [HomeKit](#)
-  [64-Bit Transition Guide](#)

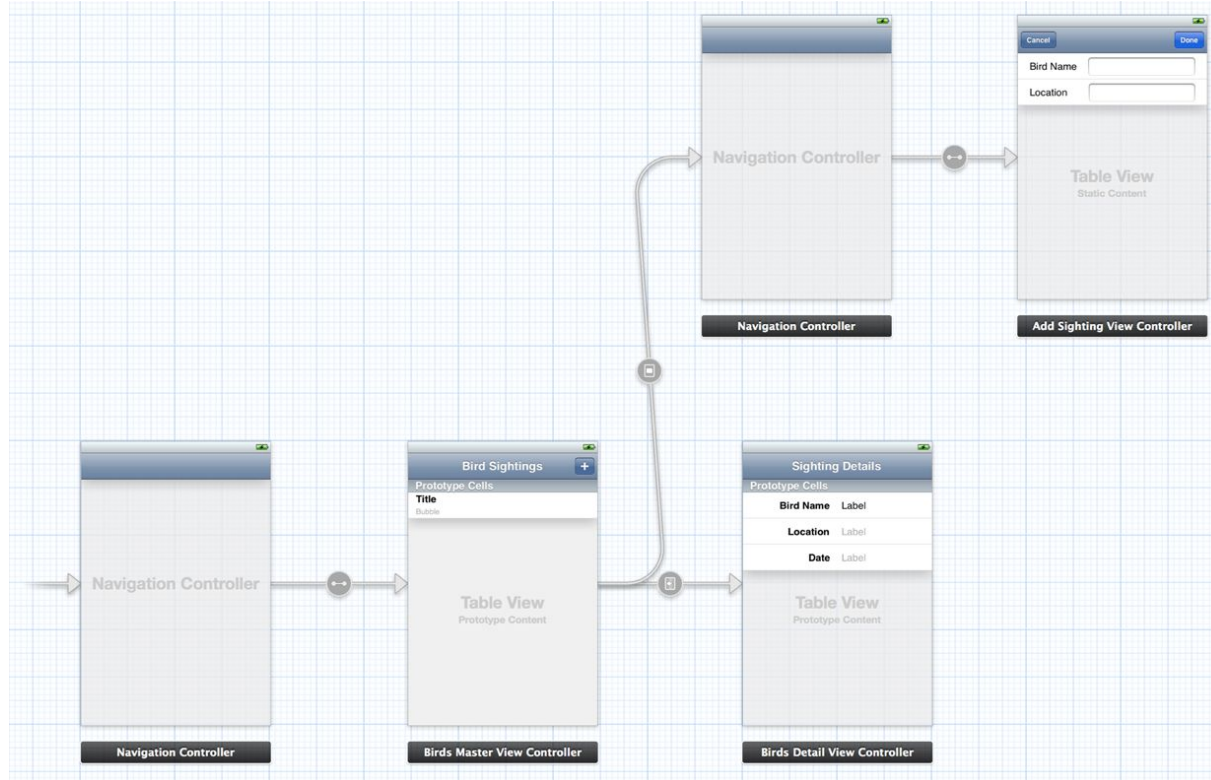
iOS Developer Program

- [Certificates, Identifiers & Profiles](#)
- [iTunes Connect](#)
- [Apple Developer Forums](#)
- [Developer Support Center](#)

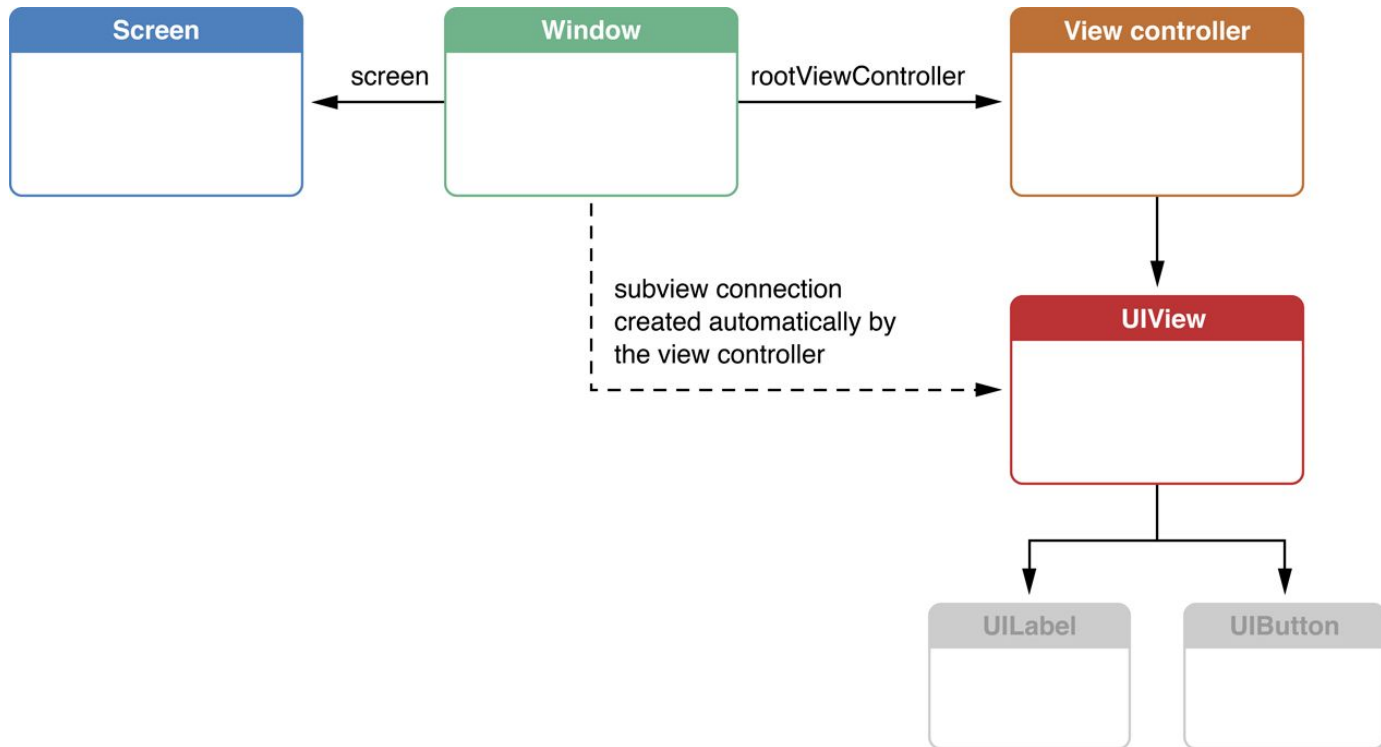
App Store Resource Center

-  [Prepare for App Submission](#)
-  [App Store Approval Process](#)

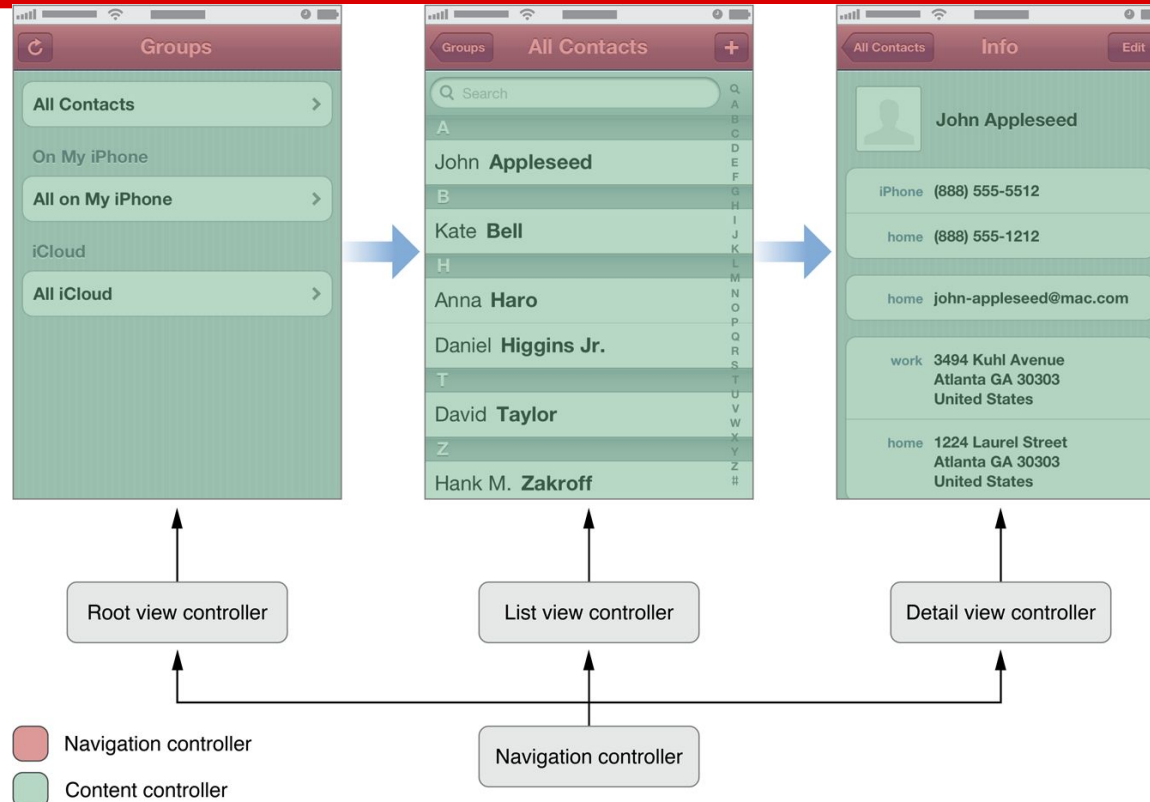
Interface Builder



View Controllers



View Controllers



HW

MULTIPLE VIEW CONTROLLERS

In class we learned quite a bit about view controllers. We learned that view controllers are “smart”, and we also learned how to display one view controller from another using segues. UITabBarController and UINavigationController are both useful when managing several UIViewControllers, creating a user interface that the end user of your app will be familiar with. For this homework create an app with the following:

- A tab bar controller containing a navigation controller in each tab
- Each navigation controller should have a view controller with a button. Be sure to make the background color of each different.
- When the button is pressed in each view controller, push a new view controller that contains a label that says, “hello world”
- Challenge: In one view controller instead of the button pushing a new view controller, have the button change the background color of the view

SOME ADDITIONAL READING

If you'd like to do some additional reading to supplement some of the class material we've covered, you may find the links below useful.

UIViewController:

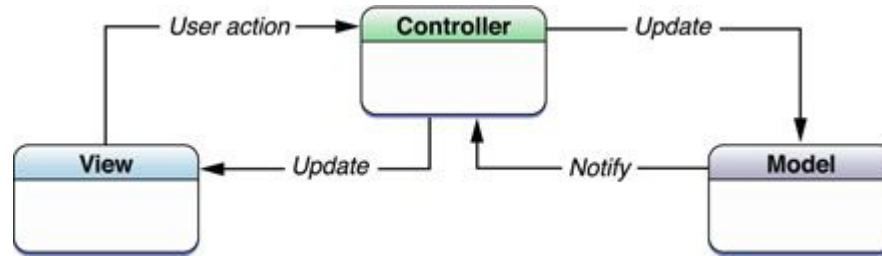
https://developer.apple.com/library/ios/documentation/UIKit/Reference/UIViewController_Class/

UIView:

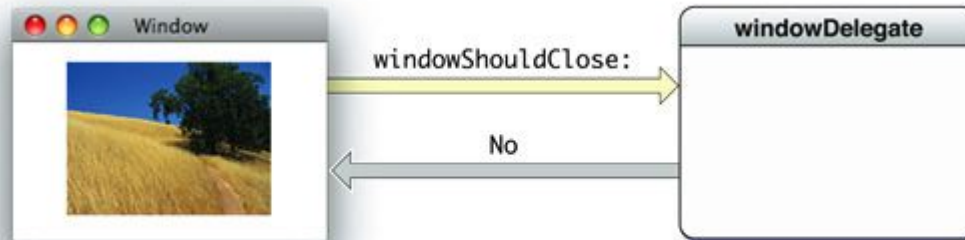
https://developer.apple.com/library/ios/documentation/UIKit/Reference/UIView_Class/

Challenge (we'll cover this topic in a few weeks!): <http://roadfiresoftware.com/2015/01/ios-essentials-the-uiviewcontroller-lifecycle/>

Model View Controller



Delegation



UIKit flow

