

MOBILE DEVELOPMENT

LESSON 02 DEEPER DIVE INTO XCODE'S INTERFACE BUILDER

Arthur Ariel Sabintsev

Lead Mobile Architect, [ID.me](https://id.me)

GETTING STARTED

WWDC 2015

KEYNOTE

GETTING STARTED

WHAT DID APPLE ANNOUNCE

- OS X 10.11 (El Capitan)
- iOS 9
- watchOS 2
- Swift 2
 - Swift will be Open Sourced

GETTING STARTED

HOMework REVIEW

GETTING STARTED

WHAT DID WE LEARN IN LESSON

- Course Expectations
- Nomenclature
 - iOS, Swift, Cocoa Touch, etc.
- Overview of Developer Tools
 - Xcode, iOS Simulator, Github, Slack
- Git and Github (on Web and for Mac)
- Hands on Demo with Storyboards in Xcode's Interface Builder tool

GETTING STARTED

LEARNING OBJECTIVES

GETTING STARTED

LEARNING OBJECTIVES

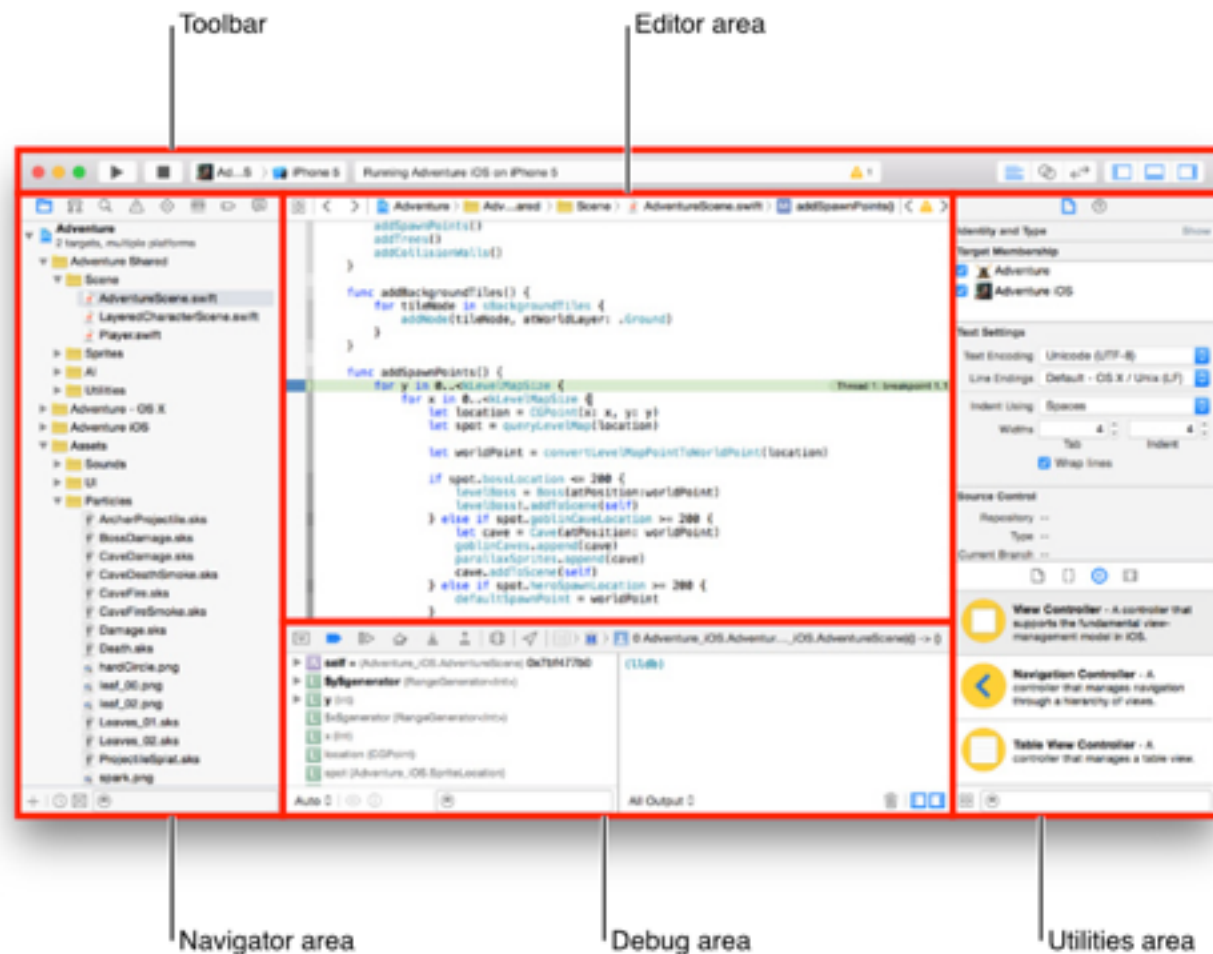
- Label everything in the Xcode window
- Storyboards vs. Xibs
 - Review of Storyboards
 - Review of Xibs (pronounced nibs)
- Learn about View Controllers (High-level)
 - Add multiple View Controllers to the storyboard
 - Link multiple View Controllers together with segues
- Learn about Navigation Controllers
 - Use Navigation Controller to link scenes

GETTING STARTED

LABEL XCODE WINDOW

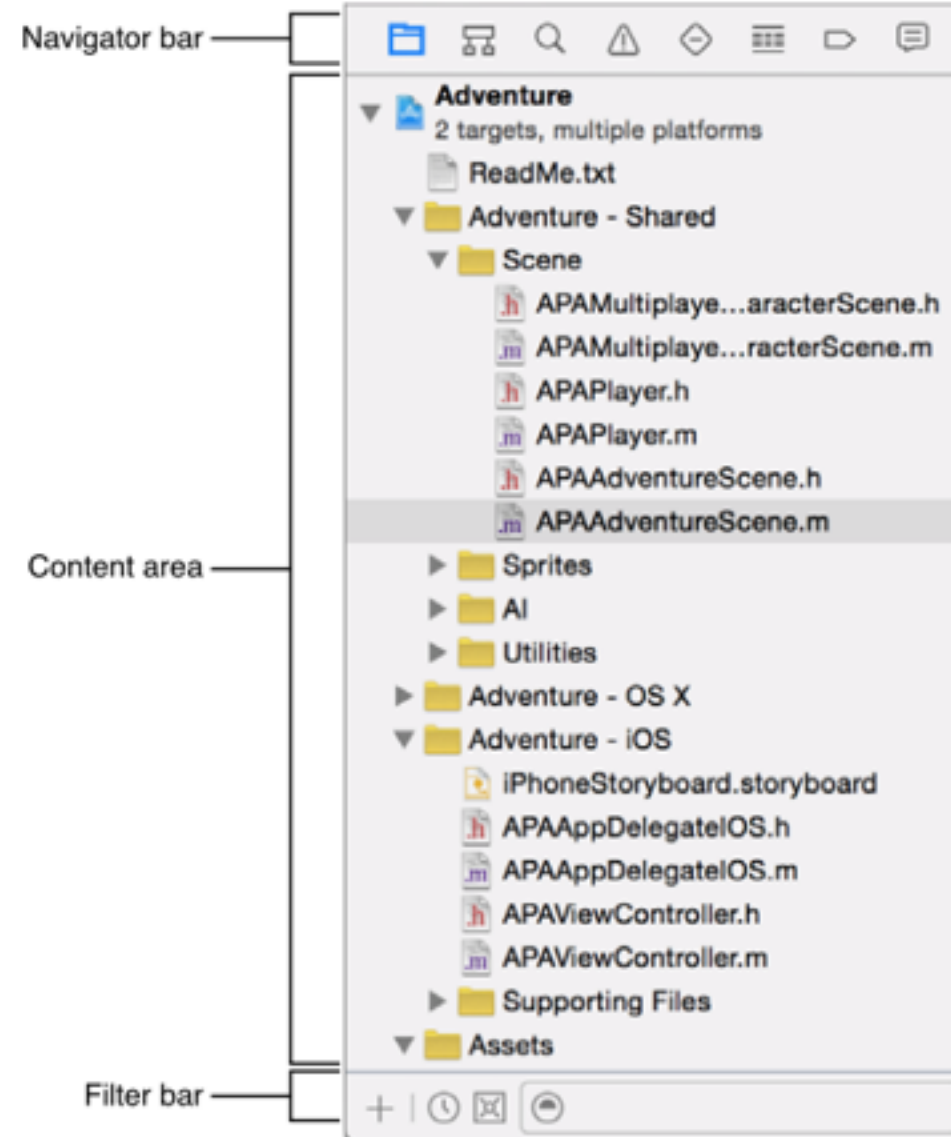
GETTING STARTED

NAVIGATING XCODE



GETTING STARTED

NAVIGATOR AF



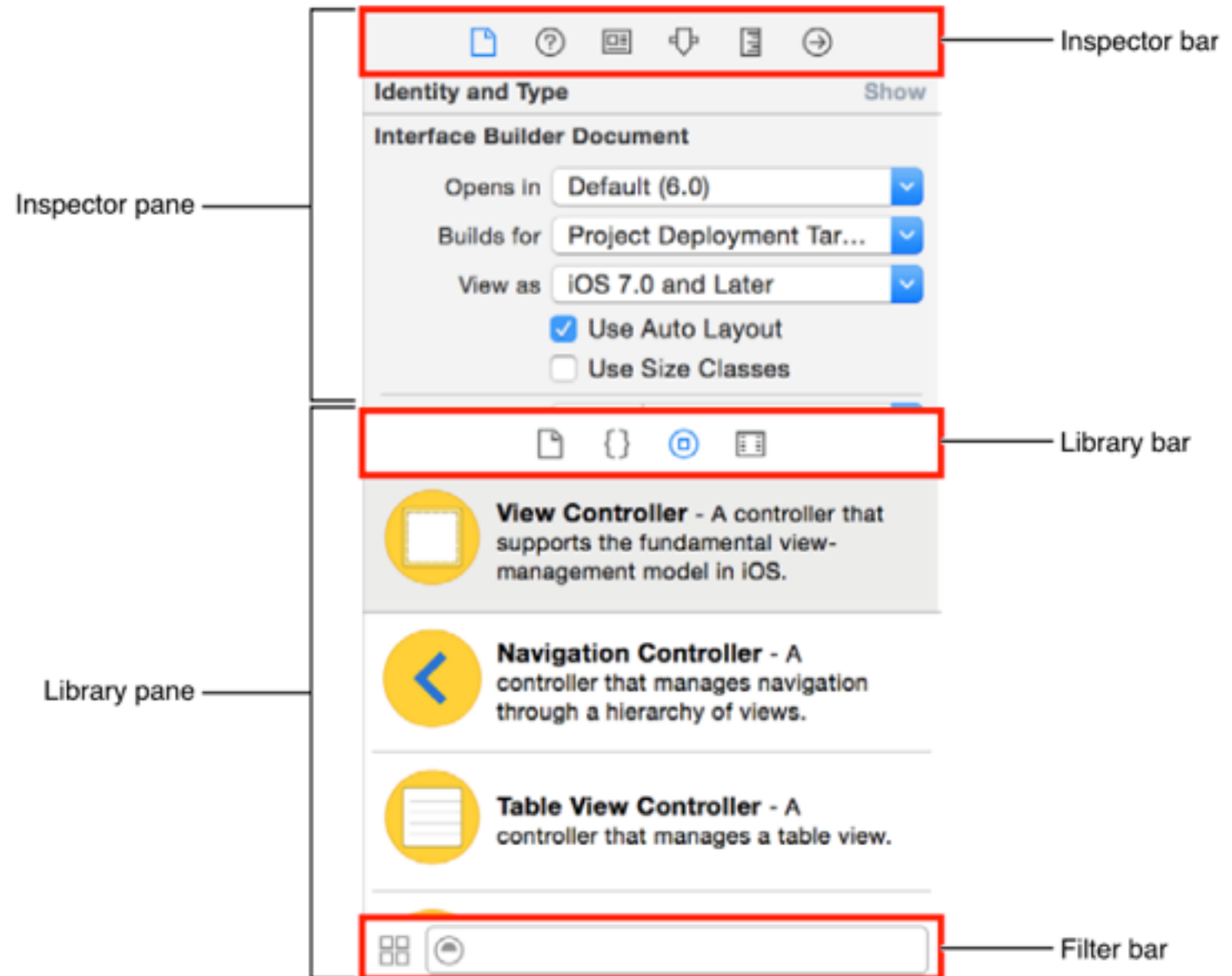
GETTING STARTED

JUMP BAR AND EDITOR PANES



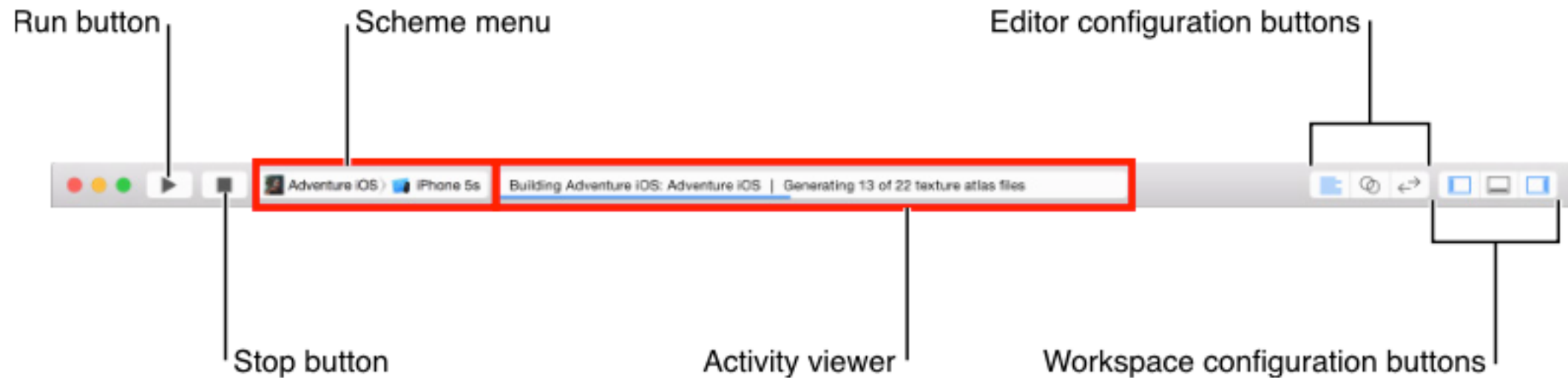
GETTING STARTED

UTILITIES A



GETTING STARTED

WORKSPACE TOOLBAR



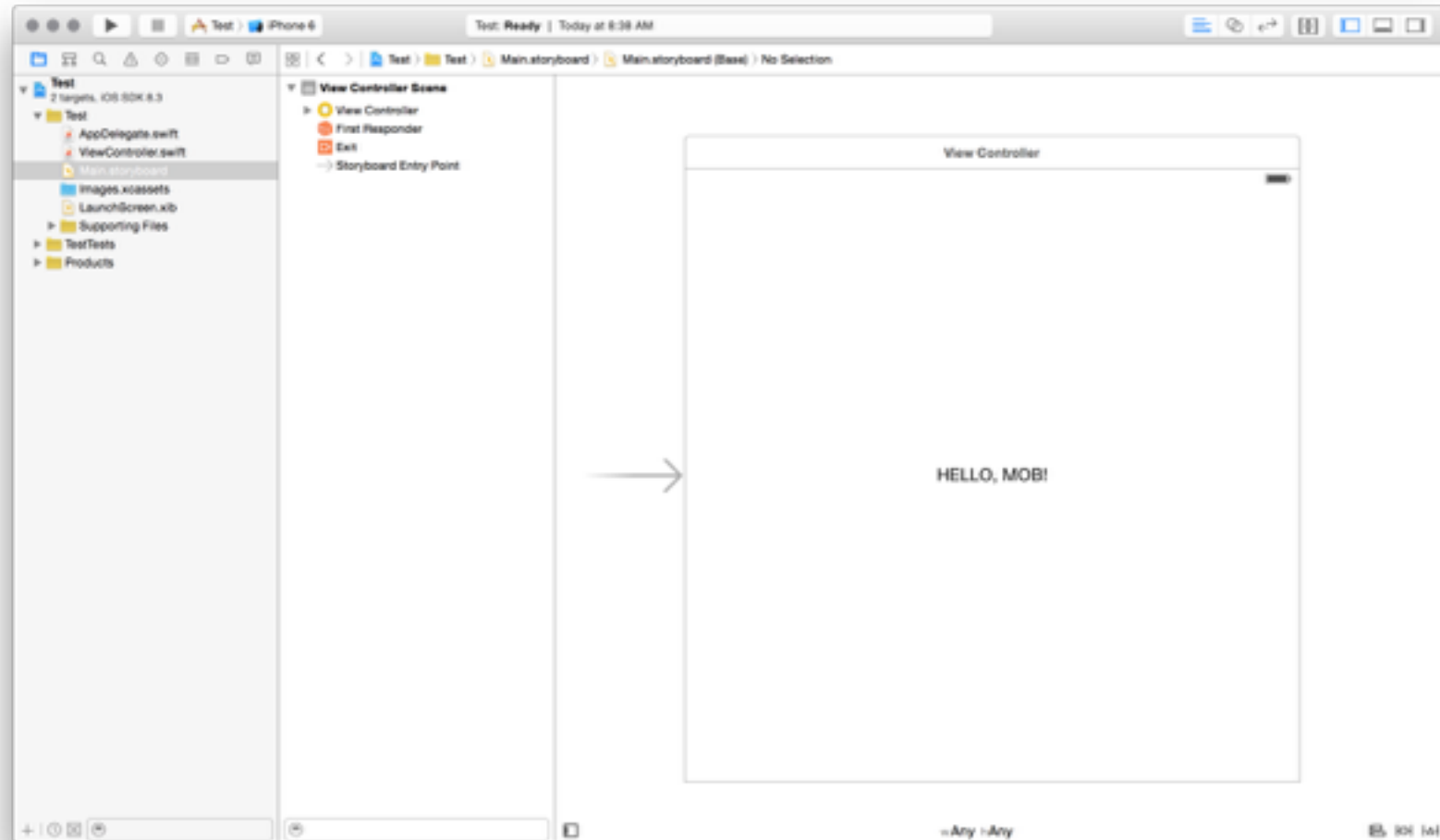
GETTING STARTED

VIEW CONTROLLERS

GETTING STARTED

VIEW CONTROLLERS (PT. 1)

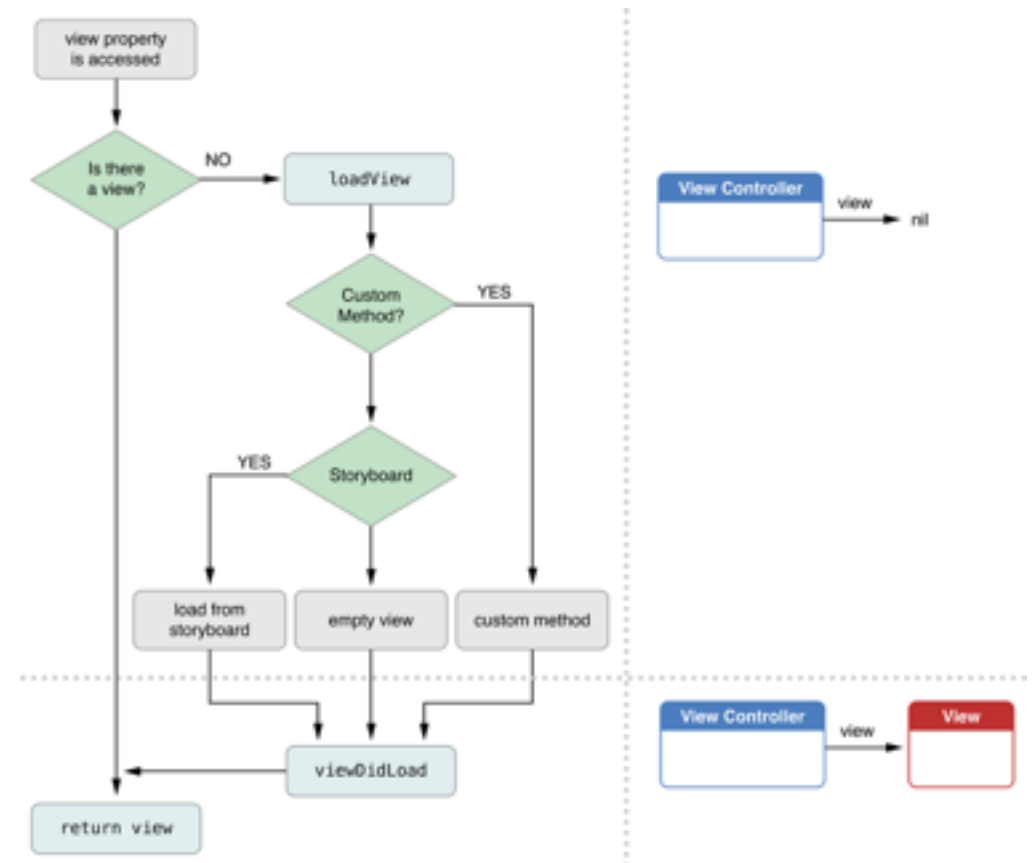
- Each *View Controller* represents one screen in our app
- *Each View Controller* represents one scene in our Storyboard



GETTING STARTED

VIEW CONTROLLERS (PT. 2)

- A view controller manages the lifecycle of its views
- Resource management in View Controllers:
<https://developer.apple.com/library/ios/featuredarticles/ViewControllerPGforiPhoneOS/ViewLoadingandUnloading/ViewLoadingandUnloading.html>



GETTING STARTED

VIEW CONTROLLERS (PT. 3)

- A view is a User Interface (UI) element (e.g., label, button, etc.)
 - A view has two intrinsic properties:
 - **Size**: Width and height
 - **Point**: The specific location (x & y coordinates) of where it's drawn on the screen
 - Combined, the **size** and **point** make the **frame** of the View Controller.
- Views can contain other views; **subviews**
- View controllers can contain other view controllers; **child view controllers**

GETTING STARTED

NAVIGATION CONTROLLERS

GETTING STARTED

NAVIGATION CONTROLLERS

- Navigation controllers are view controllers that manage a stack of view controllers
- One view (the top of the stack) is presented at a time
- By default, navigation controllers come with a customizable navigation bar



GETTING STARTED

STORYBOARDS VS XIBS

GETTING STARTED

STORYBOARDS

- A way to visually represent your entire app
- Each scene in a storyboard can be connected to another scene
- See how data flows between the screens in your app

GETTING STARTED

XIBS

- A way to visually represent one view controller
- A way to visually represent one view and its subviews
- Allows for reusability (DRY Principle)
- Historical note:
 - XML Interface Builder files (.xib) were introduced in Xcode 3 in 2007.
 - .xib files are turned into .nib files when your application is compiled
 - More info here:
 - <http://www.speirs.org/blog/2007/12/5/what-are-xib-files.html>
 - https://developer.apple.com/library/ios/documentation/General/Conceptual/

GETTING STARTED

STORYBOARDS VS XIBS

PROS (STORYBOARDS)

- Simplify the prototyping process
- Easy to see application flow

PROS (XIBS)

- Easier to version control than storyboards
- Reusability

CONS (STORYBOARDS)

- Lacks Reusability
- Hard to version control
and have multiple developers
work on a single storyboard

CONS (XIBS)

- Lack of a way to represent
graphical relationships between screens
- Takes longer to load into than storyboards

GETTING STARTED

WHEN TO USE STORYBOARDS

- Scoping out the application flow
- When creating one-time use view controllers

GETTING STARTED

WHEN TO USE NIBS

- Creating reusable view controllers
- Creating custom views
 - For example:
 - Pop up windows
 - Menus
 - Buttons
 - Settings screens

GETTING STARTED



KEY OBJECTIVE(S)

Learn the flow of building a new project. Add UI elements to a project and modify their properties. Use storyboards.

TIMING

10 min 1. Code with partner

5 min 2. Debrief

DELIVERABLE

A new project with a label displaying student bio, an image of that student and a button labeled “Goals”. Use storyboards.

Bonus: Make your bio, button and picture appear when the device is rotated

GETTING STARTED

STORYBOARD MULTIPLE SCENES EXAMPLE

GETTING STARTED

IN-CLASS EXERCISE

GETTING STARTED



EXERCISE

KEY OBJECTIVE(S)

Create a view with your bio (including image).
Create multiple views inside of a Navigation Controller

TIMING

30 min 1. Work in pairs.

10 min 2. Debrief

DELIVERABLE

- Pair up and build an app with multiple scenes linking
- Add UI Elements and appropriately link them to other slides.

GETTING STARTED

HOMEWORK

GETTING STARTED

HOMework

- Homework 1 (Lessons 1-2)
 - Link: [HW 1 for Lessons 1-2](#)
 - Due: Sunday, June 14, 2015 at 6pm.
 - Read Chapter 2 in the Gitbook:
 - Link: [Chapter 2 in MOB Gitbook](#)
 - Read The Basics Chapter in Apple's Swift iBook
 - Link: m

GETTING STARTED

Q&A