iOS Dev (in Swift)

What I wish I had been told instead of “RTFM”

Alex Popov • @alexpopovjr • hello@alexpopov.ca

About Me

Principal iOS dev @ Shelfie

Coming Up

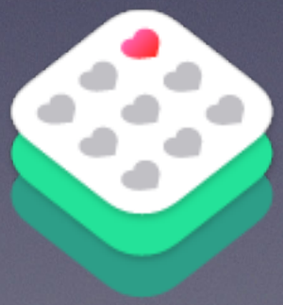
- Where to start
- High-level Concepts
- Best Practices

Getting Started

Developer Program

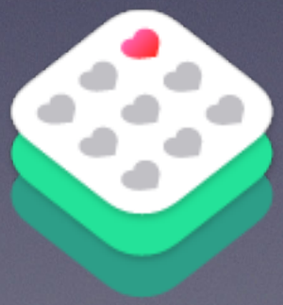
Getting Started

- Problem
- Fun
- WWDC



Getting Started

- Swift
- Objective-C



Getting Started

- Swift
- Objective-C
- Low-level
- AVFoundation
- Games

Note on Languages

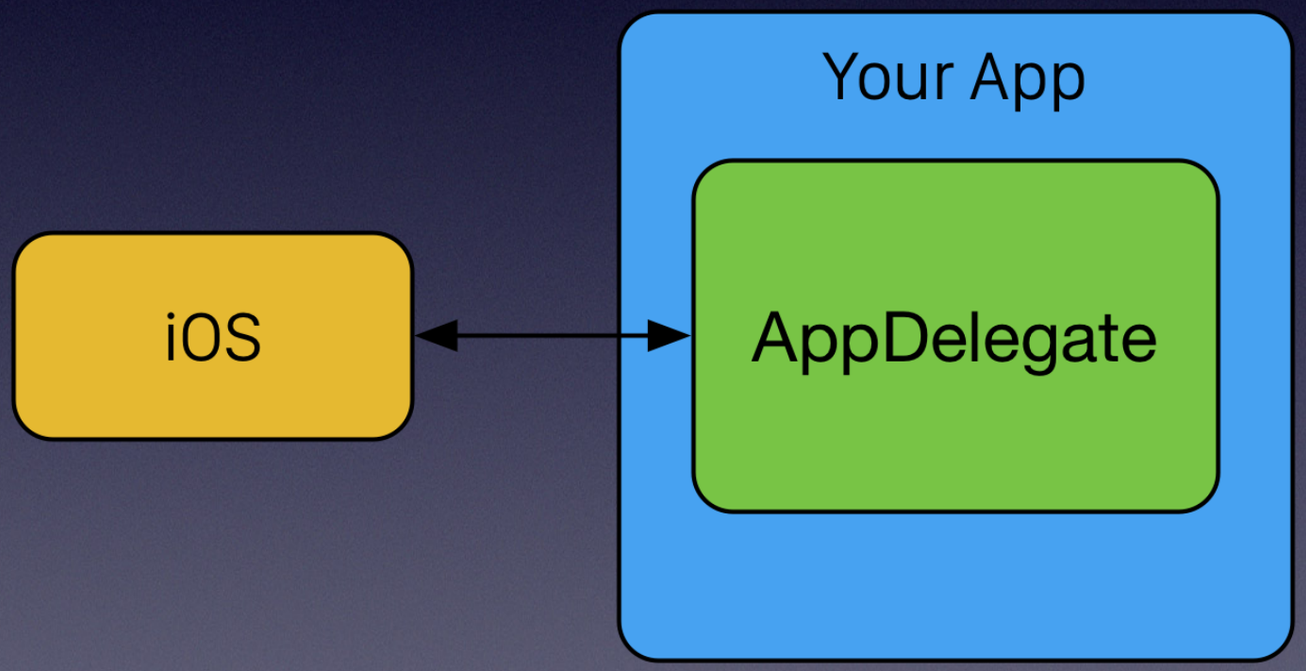
- Writing code is easy
- Software engineering is hard

Building Blocks

- iOS
- AppDelegate
- ViewController
- View
- Model

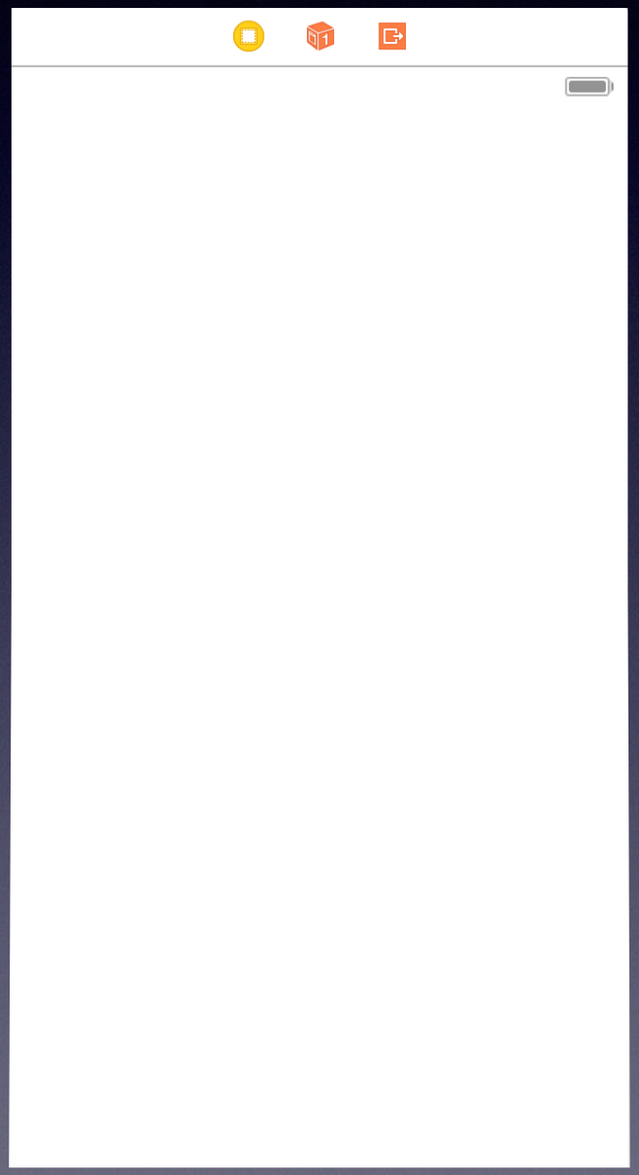
Building Blocks

- iOS
- AppDelegate
- ViewController
- View
- Model



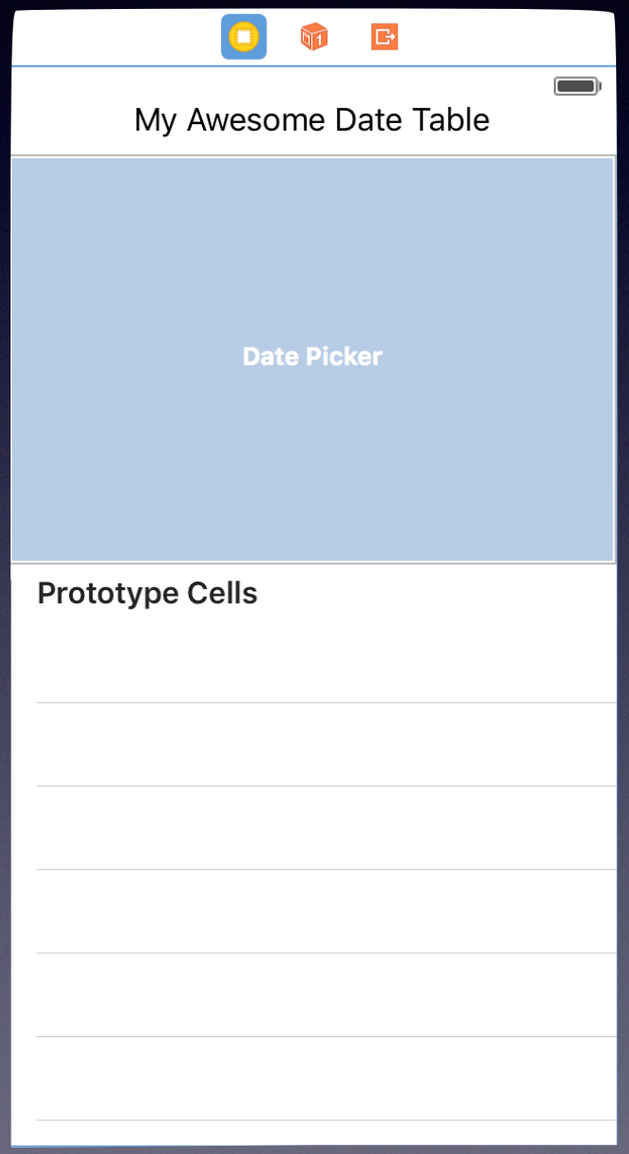
Building Blocks

- iOS
- AppDelegate
- ViewController
- View
- Model



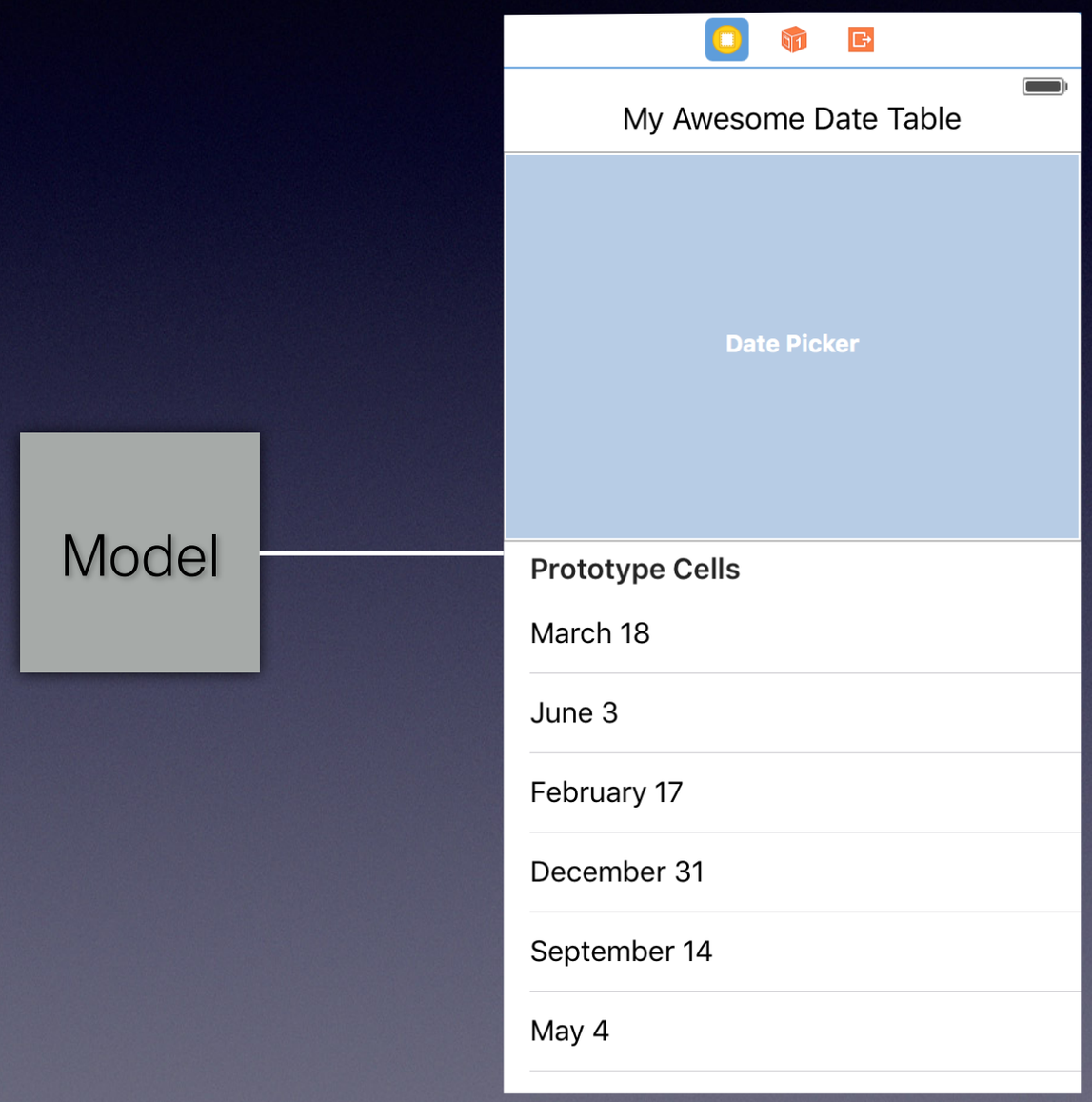
Building Blocks

- iOS
- AppDelegate
- ViewController
- View
- Model



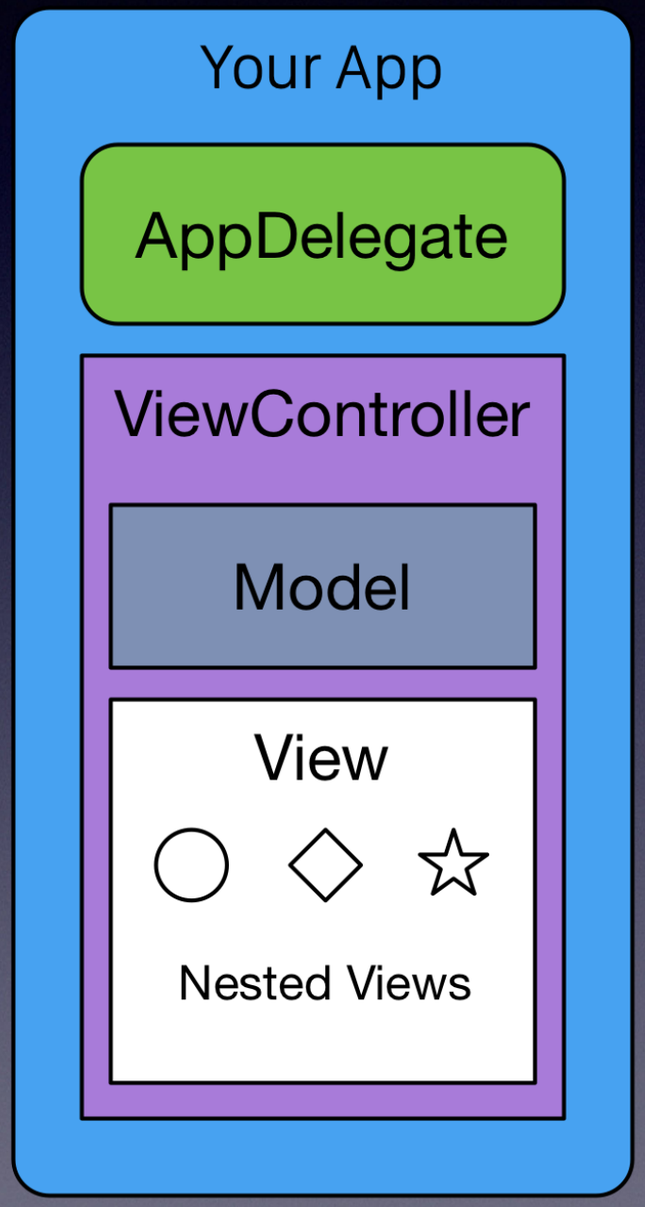
Building Blocks

- iOS
- AppDelegate
- ViewController
- View
- Model

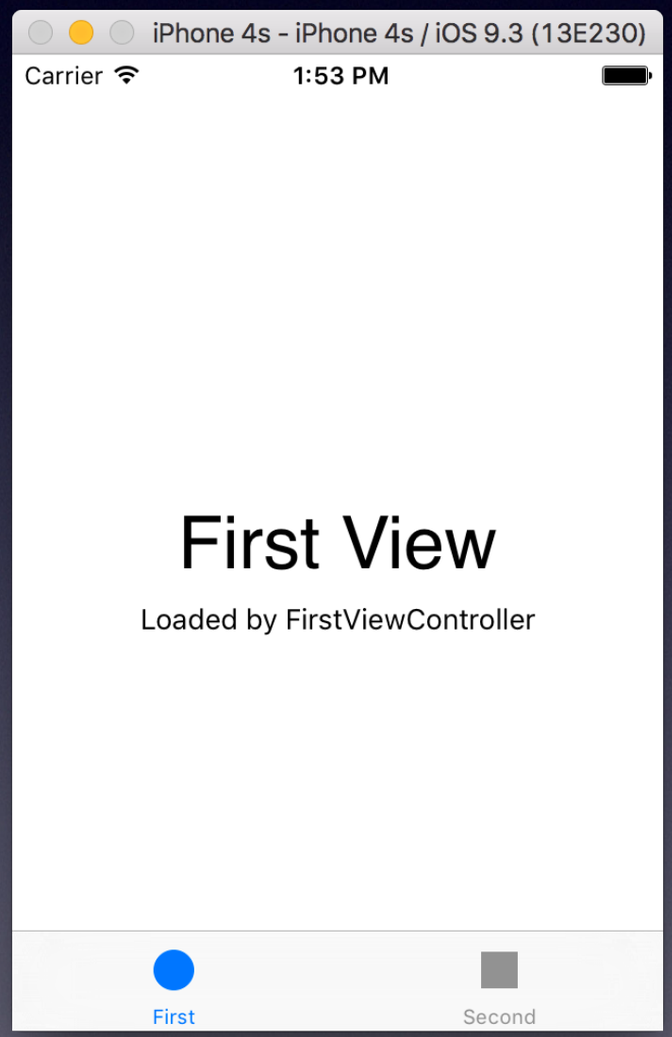


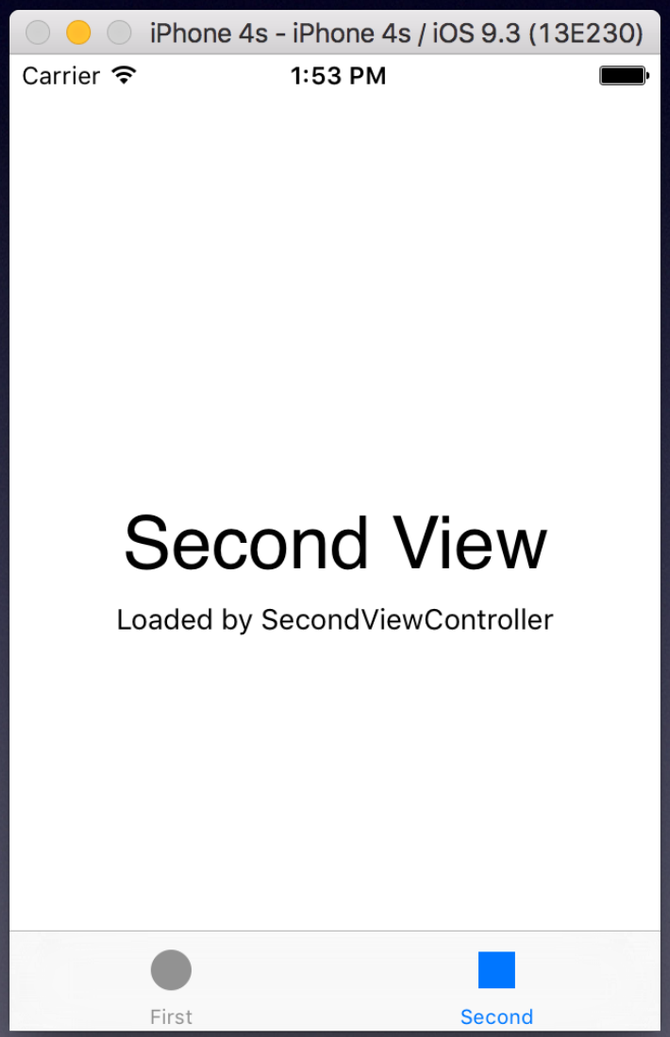
Building Blocks

- iOS
- AppDelegate
- ViewController
- View
- Model



Your First App





Your First App

Sw...alk 〉 iPhone 6s Plus | SwiftTalk: **Ready** | Today at 1:24 PM

AppDelegate.swift

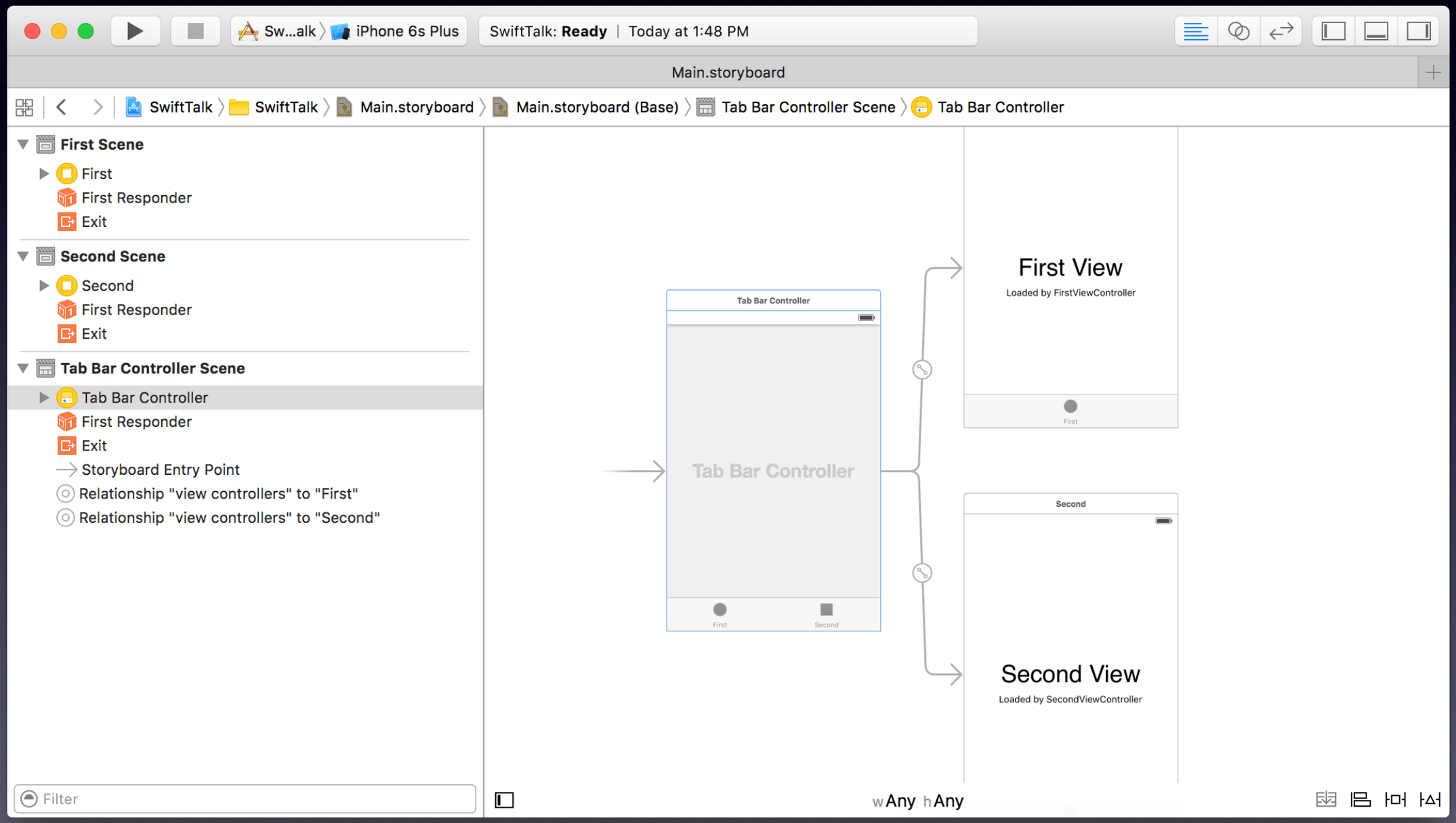
SwiftTalk 〉 SwiftTalk 〉 AppDelegate.swift 〉 No Selection

- SwiftTalk
  - SwiftTalk
    - AppDelegate.swift
    - FirstViewController.swift
    - SecondViewController.swift
    - Main.storyboard
    - Assets.xcassets
    - LaunchScreen.storyboard
    - Info.plist
  - Products

Filter

```
import UIKit

@UIApplicationMain
class AppDelegate: UIResponder, UIApplicationDelegate {

  var window: UIWindow?


  func application(application: UIApplication, didFinishLaunchingWithOptions launchOptions: [NSObject: AnyObject]?)
      -> Bool {
    // Override point for customization after application launch.
    return true
  }

  func applicationWillResignActive(application: UIApplication) {
    // Sent when the application is about to move from active to inactive state. This can occur for certain types of
        temporary interruptions (such as an incoming phone call or SMS message) or when the user quits the
        application and it begins the transition to the background state.
    // Use this method to pause ongoing tasks, disable timers, and throttle down OpenGL ES frame rates. Games should
        use this method to pause the game.
  }

  func applicationDidEnterBackground(application: UIApplication) {
    // Use this method to release shared resources, save user data, invalidate timers, and store enough application
        state information to restore your application to its current state in case it is terminated later.
    // If your application supports background execution, this method is called instead of applicationWillTerminate:
        when the user quits.
  }

  func applicationWillEnterForeground(application: UIApplication) {
    // Called as part of the transition from the background to the inactive state; here you can undo many of the
        changes made on entering the background.
  }

  func applicationDidBecomeActive(application: UIApplication) {
    // Restart any tasks that were paused (or not yet started) while the application was inactive. If the
        application was previously in the background, optionally refresh the user interface.
  }

  func applicationWillTerminate(application: UIApplication) {
    // Called when the application is about to terminate. Save data if appropriate. See also
        applicationDidEnterBackground:.
  }


}
```

Your First App



AppDelegate

Sw...alk 〉 iPhone 6s Plus | SwiftTalk: **Ready** | Today at 1:24 PM

AppDelegate.swift

SwiftTalk 〉 SwiftTalk 〉 AppDelegate.swift 〉 No Selection

- SwiftTalk
  - SwiftTalk
    - AppDelegate.swift
    - FirstViewController.swift
    - SecondViewController.swift
    - Main.storyboard
    - Assets.xcassets
    - LaunchScreen.storyboard
    - Info.plist
  - Products

Filter

```
9   import UIKit
10
11  @UIApplicationMain
12  class AppDelegate: UIResponder, UIApplicationDelegate {
13
14    var window: UIWindow?
15
16
17    func application(application: UIApplication, didFinishLaunchingWithOptions launchOptions: [NSObject: AnyObject]?)
          -> Bool {
18      // Override point for customization after application launch.
19      return true
20    }
21
22    func applicationWillResignActive(application: UIApplication) {
23      // Sent when the application is about to move from active to inactive state. This can occur for certain types of
          temporary interruptions (such as an incoming phone call or SMS message) or when the user quits the
          application and it begins the transition to the background state.
24      // Use this method to pause ongoing tasks, disable timers, and throttle down OpenGL ES frame rates. Games should
          use this method to pause the game.
25    }
26
27    func applicationDidEnterBackground(application: UIApplication) {
28      // Use this method to release shared resources, save user data, invalidate timers, and store enough application
          state information to restore your application to its current state in case it is terminated later.
29      // If your application supports background execution, this method is called instead of applicationWillTerminate:
          when the user quits.
30    }
31
32    func applicationWillEnterForeground(application: UIApplication) {
33      // Called as part of the transition from the background to the inactive state; here you can undo many of the
          changes made on entering the background.
34    }
35
36    func applicationDidBecomeActive(application: UIApplication) {
37      // Restart any tasks that were paused (or not yet started) while the application was inactive. If the
          application was previously in the background, optionally refresh the user interface.
38    }
39
40    func applicationWillTerminate(application: UIApplication) {
41      // Called when the application is about to terminate. Save data if appropriate. See also
          applicationDidEnterBackground:.
42    }
43
44
45  }
```

AppDelegate

Entry point:

```
application(_, didFinishLaunchingWithOptions:)
```

- Initial View Hierarchy
- Singletons
- Services

View Controllers

```
import UIKit

class FirstViewController: UIViewController {

  override func viewDidLoad() {
    super.viewDidLoad()
    // Do any additional setup after loading the view,
    // typically from a nib.
  }

}
```

View Controllers

```
import UIKit

class FirstViewController: UIViewController {

  override func viewDidLoad() {
    super.viewDidLoad()
    // Do any additional setup after loading the view,
    // typically from a nib.
  }

}
```

View Controllers

```
import UIKit

class FirstViewController: UIViewController {

  override func viewDidLoad() {
    super.viewDidLoad()
    // Do any additional setup after loading the view,
    // typically from a nib.
  }

}
```

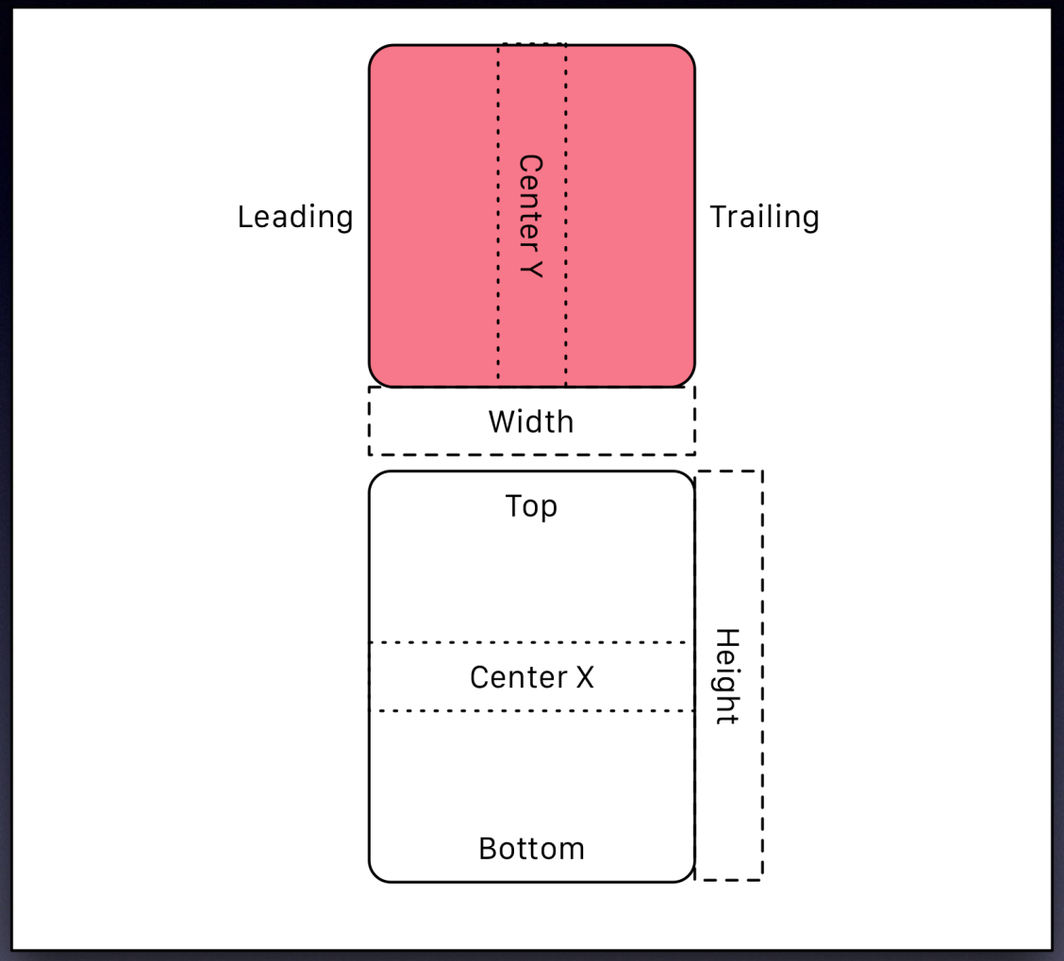
View Controllers

```
import UIKit

class FirstViewController: UIViewController {

  override func viewDidLoad() {
    super.viewDidLoad()
    // Do any additional setup after loading the view,
    // typically from a nib.
  }

}
```

Views and Layout

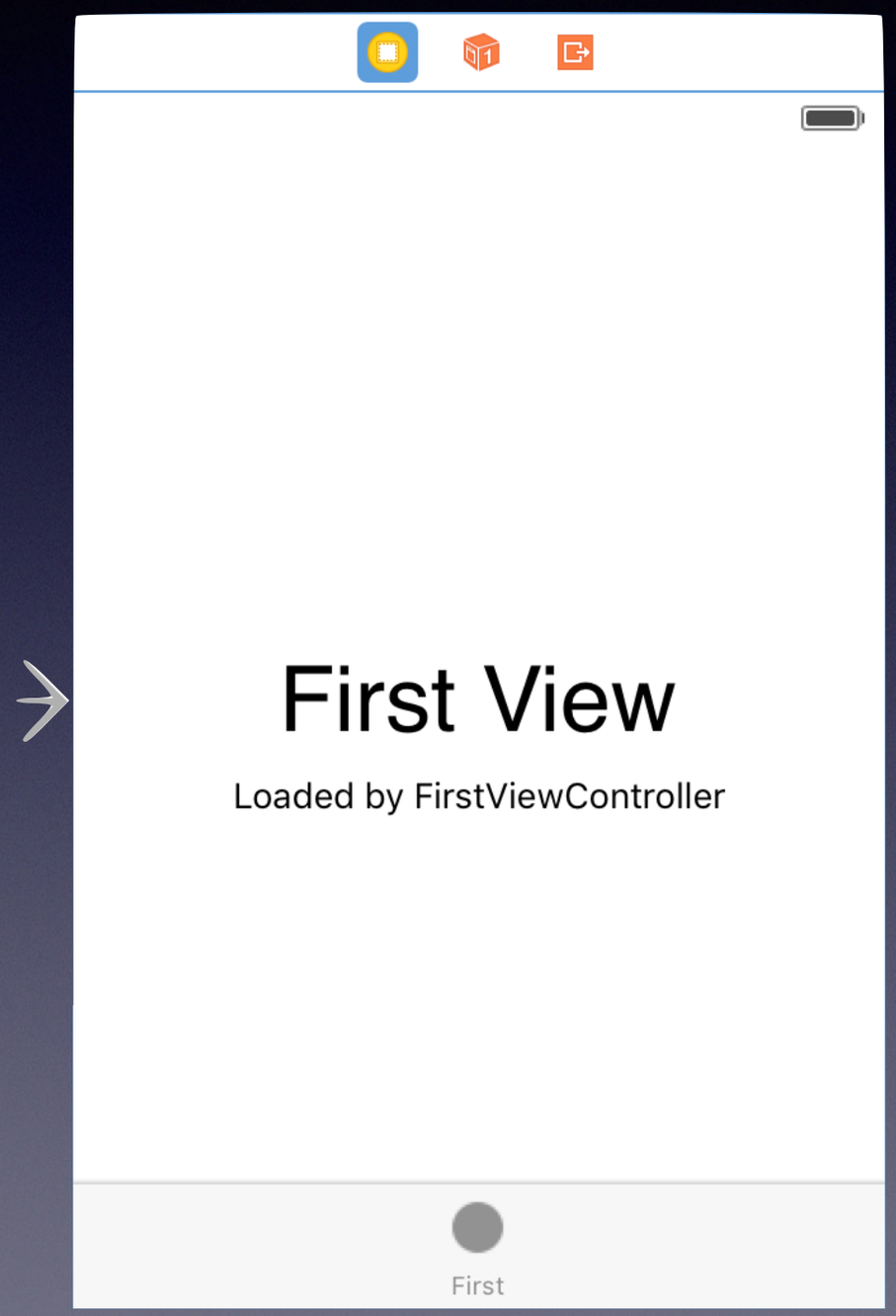
- Dumb
- Imported from UIKit
- Autolayout
- Constraints

Views and Layout




```
Y = m*X + B

WhiteBox.Top = 1.0 * PinkBox.Bottom + 20
```

Views and Layout



SwiftTalk 〉 iPhone 4s

Running SwiftTalk on iPhone 4s

Main.storyboard

SwiftTalk 〉 View 〉 L Loaded by FirstViewController

- SwiftTalk
  - SwiftTalk
    - AppDelegate.swift
    - FirstViewController.swift
    - SecondViewController.swift
    - Main.storyboard
    - Assets.xcassets
    - LaunchScreen.storyboard
    - Info.plist
  - Products

First View

Loaded by FirstViewController

First

wAny hAny

SwiftTalk

Constraints

Ag

All This Size Class

Align Center X to: First View **Edit**

Top Space to: First View
Equals: Default **Edit**

Showing 2 of 2

Content Hugging Priority

Horizontal 251

Vertical 251

Content Compression Resistance Priority

Horizontal 750

Vertical 750

Intrinsic Size Default (System Defined)

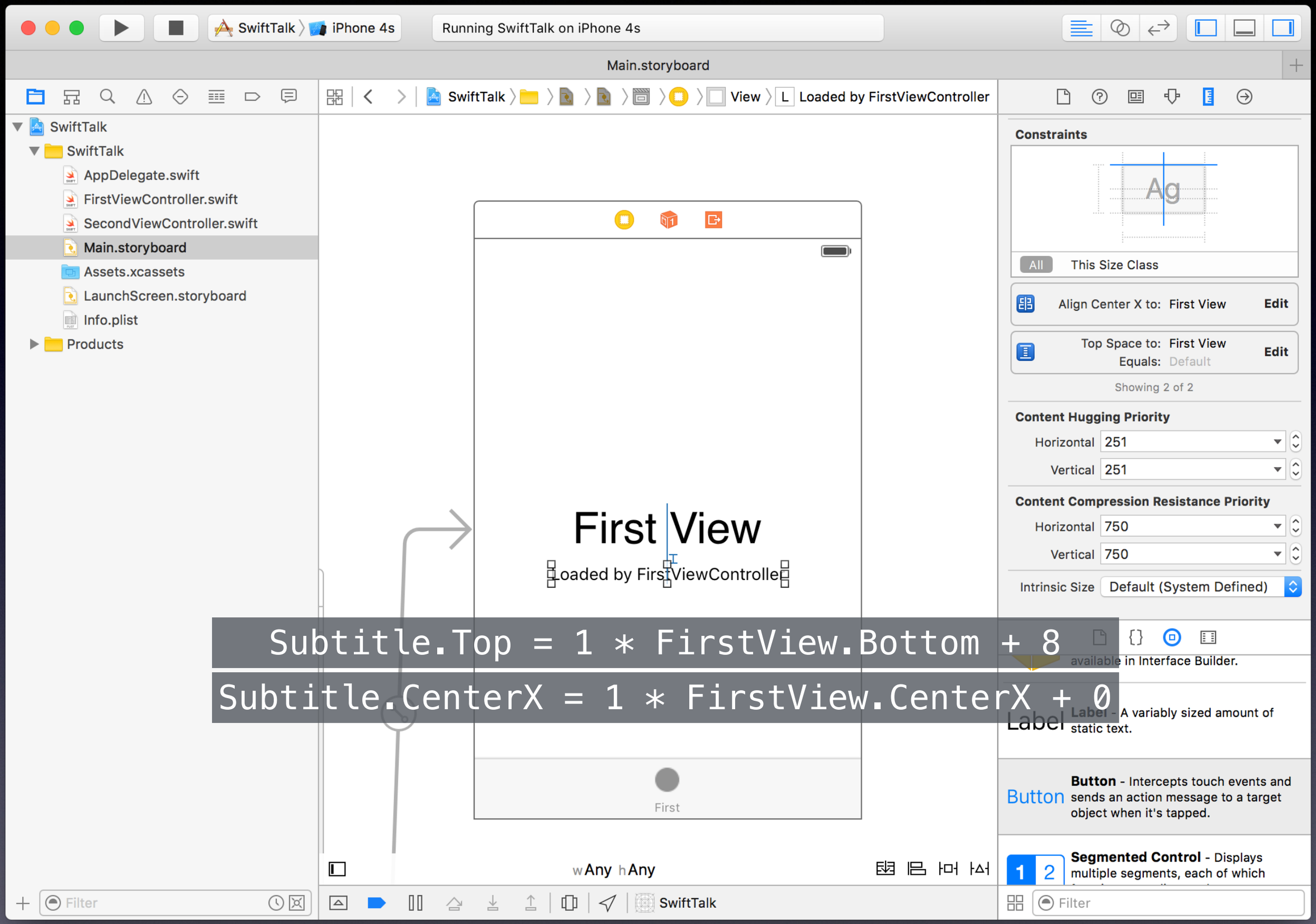
available in Interface Builder.

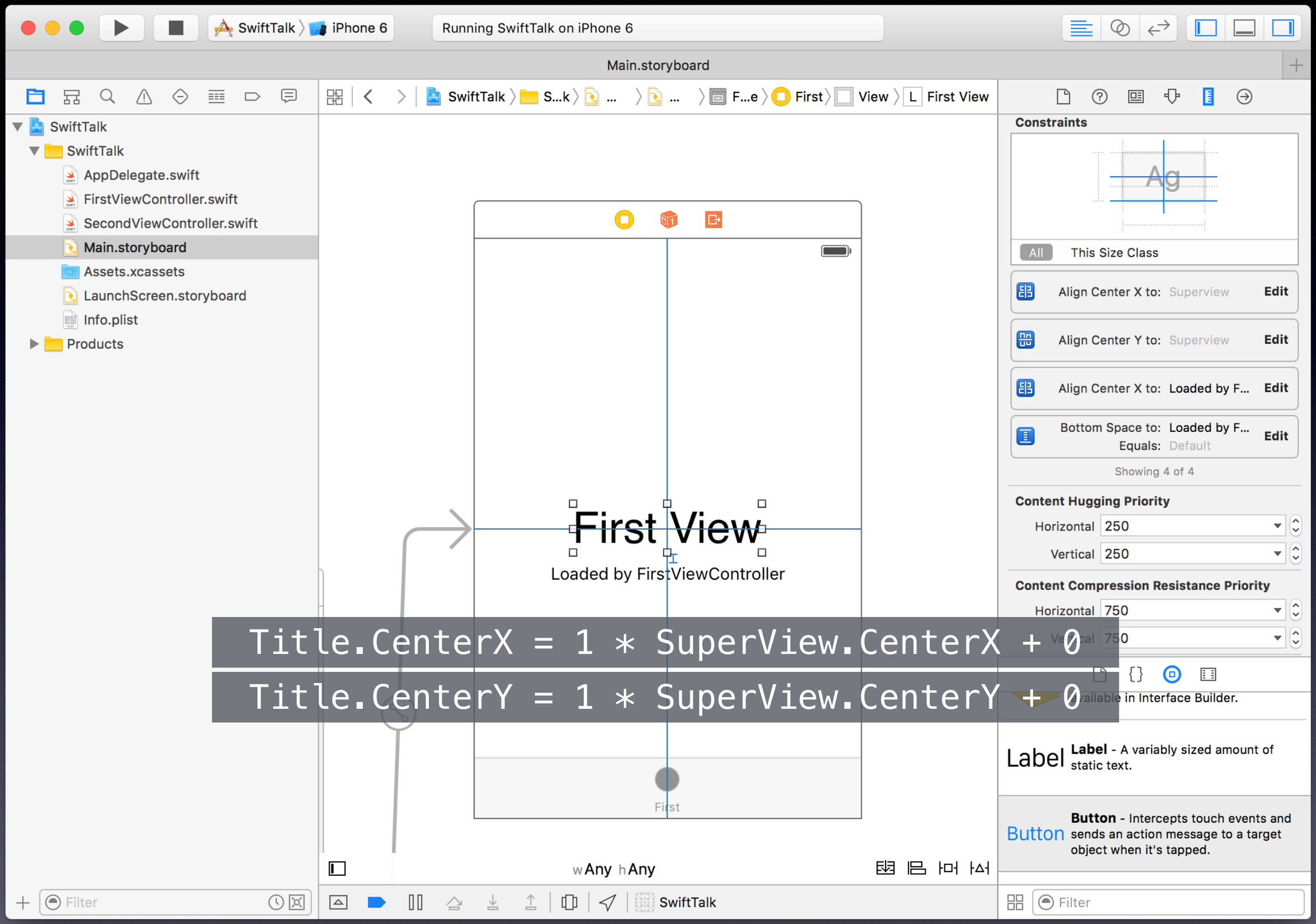
Label **Label** - A variably sized amount of static text.

Button **Button** - Intercepts touch events and sends an action message to a target object when it's tapped.

1 2 **Segmented Control** - Displays multiple segments, each of which

Filter

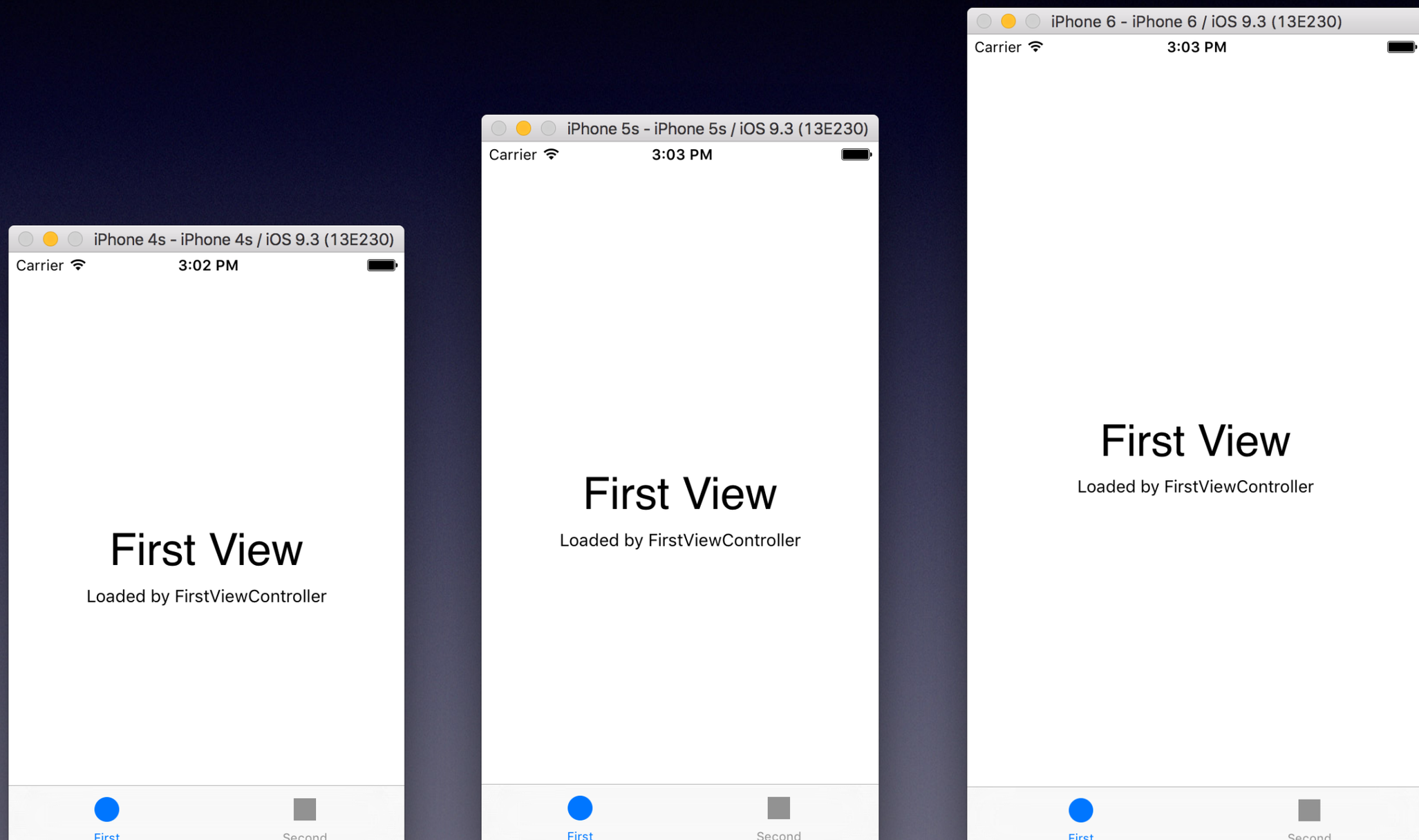

SwiftTalk
iPhone 4s
Running SwiftTalk on iPhone 4s
Main.storyboard
SwiftTalk
View
Loaded by FirstViewController
SwiftTalk
SwiftTalk
AppDelegate.swift
FirstViewController.swift
SecondViewController.swift
Main.storyboard
Assets.xcassets
LaunchScreen.storyboard
Info.plist
Products
Constraints
Ag
All
This Size Class
Align Center X to: First View
Edit
Top Space to: First View
Equals: Default
Edit
Showing 2 of 2
Content Hugging Priority
Horizontal 251
Vertical 251
Content Compression Resistance Priority
Horizontal 750
Vertical 750
Intrinsic Size Default (System Defined)
First View
Loaded by FirstViewController
First
wAny hAny
Subtitle.Top = 1 * FirstView.Bottom + 8
Subtitle.CenterX = 1 * FirstView.CenterX + 0
available in Interface Builder.
Label
Label - A variably sized amount of static text.
Button
Button - Intercepts touch events and sends an action message to a target object when it's tapped.
Segmented Control - Displays multiple segments, each of which
Filter
SwiftTalk
Filter


SwiftTalk
iPhone 6
Running SwiftTalk on iPhone 6
Main.storyboard
SwiftTalk
SwiftTalk
AppDelegate.swift
FirstViewController.swift
SecondViewController.swift
Main.storyboard
Assets.xcassets
LaunchScreen.storyboard
Info.plist
Products
First View
Loaded by FirstViewController
First
wAny hAny
Constraints
All
This Size Class
Align Center X to: Superview Edit
Align Center Y to: Superview Edit
Align Center X to: Loaded by F... Edit
Bottom Space to: Loaded by F...
Equals: Default
Edit
Showing 4 of 4
Content Hugging Priority
Horizontal 250
Vertical 250
Content Compression Resistance Priority
Horizontal 750
Vertical 750
Label - A variably sized amount of static text.
Button - Intercepts touch events and sends an action message to a target object when it's tapped.
Filter
Title.CenterX = 1 * SuperView.CenterX + 0
Title.CenterY = 1 * SuperView.CenterY + 0

Views and Layout

- Must be unambiguous
  - X
  - Y
  - Width
  - Height

Views and Layout



Building Blocks

- iOS
- AppDelegate
- ViewController
- View
- Model
- AutoLayout

Best Practices & Tips

- Structs
- Protocols
- Strongly-typed
- map-filter-reduce
- Futures

Questions

Alex Popov • @alexpopovjr • hello@alexpopov.ca