Multi Page Apps

PROG31975 – Week 2 Part 1

Jawaad Sheikh

Jawaad.Sheikh@SheridanCollege.ca

Outline

- Introduction
- Design
- Design Considerations
- Adding New Pages
- Segues
- Unwinding Segues
- Dressing Up Your App
- Exercise

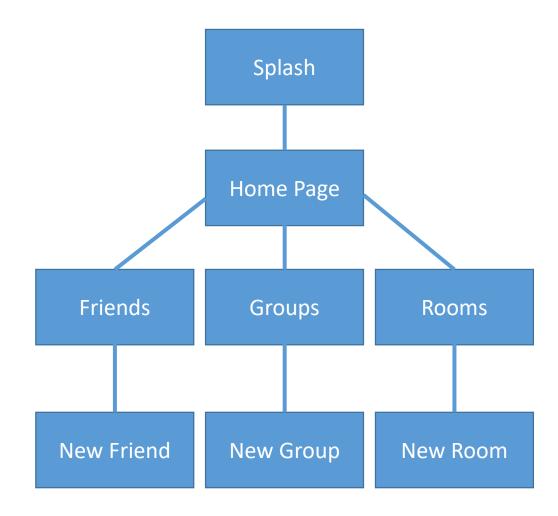
Introduction

- An app typically doesn't have just one page.
- By default, XCode auto generates the splash and home pages.
- From there, its up to you to add more pages

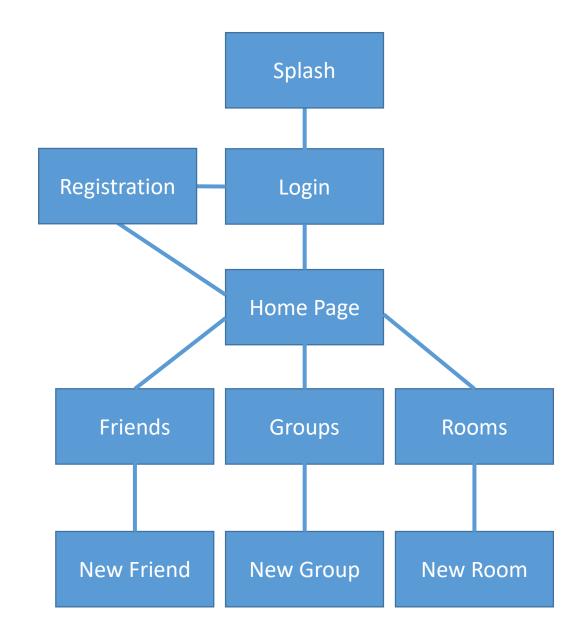
Introduction

- When designing your app, keep the following in mind:
 - Your flow needs to be intuitive!
 - If you can do it in 3 clicks instead of 4, do it!
 - Don't cram everything into one page!
 - Don't cram everything into one app!

- Design 1
 - Suppose you have an app that requires the following pages:
 - Splash
 - Home page
 - Friends list add, edit, delete
 - Groups list add, edit, delete
 - Rooms list add edit, delete



- Design 2
 - Suppose we wanted to add a login and registration to the app.



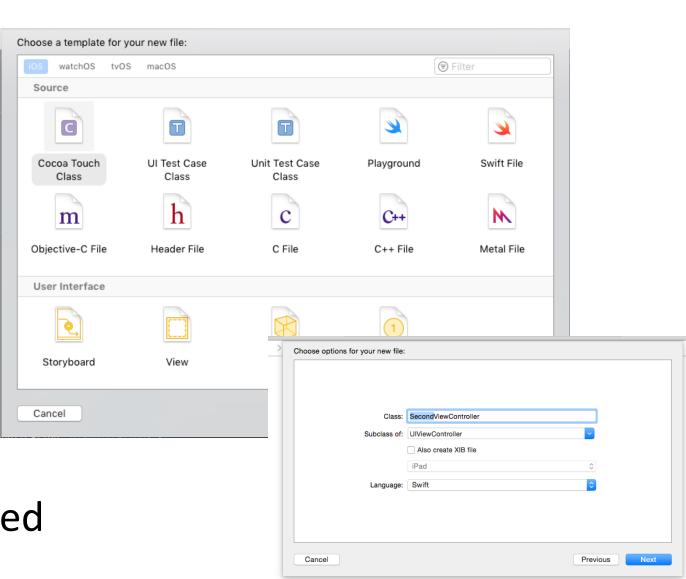
Design Considerations

- When designing multipage apps, consider the following:
 - Use fonts consistently (try using the same font everywhere)
 - Keep location of buttons and size of buttons consistent
 - Navigation bars
 - Back buttons
 - Overall keep pages looking consistent

View Controller In the View Controller Is Initial View Controller Layout Adjust Scroll View Insets storyboard, Hide Bottom Bar on Push Resize View From NIB Use Full Screen (Deprecated) drag a new Extend Edges V Under Top Bars Under Bottom Bars Under Opaque Bars View Transition Style | Cover Vertical Presentation Full Screen Controller = Defines Context object onto View Controller - A controller that manages a view. the screen: Storyboard Reference - Provides a placeholder for a view controller in an external storyboard. Navigation Controller - A controller that manages navigation through a hierarchy of views. ② 园 旧 旧 H 등 Filter hone 7 (wC hR) 60% 十

Bottom Bar Inferred

- Next, add a new View
 Controller Object
- Right click in your project explorer.
 - Select New File
- Under iOS, choose Cocoa
 Touch Class
- Ensure subclass of "UIViewController" selected



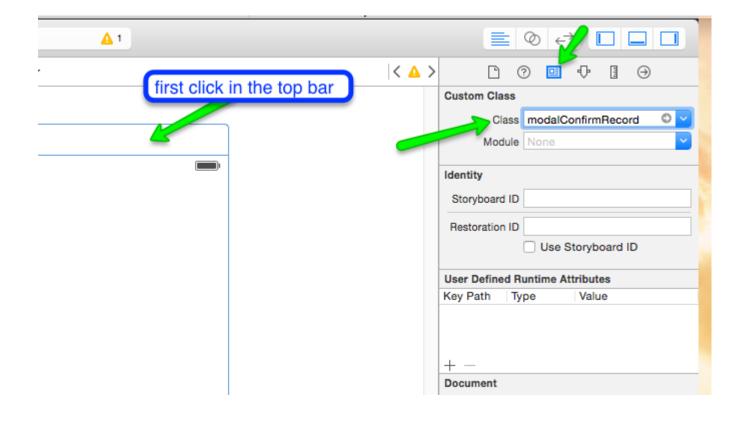
 Doing so results in a new empty swift file

```
A Test ) iPhone 7 Plus
                                                            Test: Ready | Today at 15:22
Test > Test > SecondViewController.swift > No Selection
                                           SecondViewController.swift
     AppDelegate.swift
                                           Created by Andy O'Sullivan on 14/03/2017.
     ViewController.swift
                                           Copyright @ 2017 Whatever. All rights reserved.
     Main.storyboard
     Assets.xcassets
                                       import UIKit

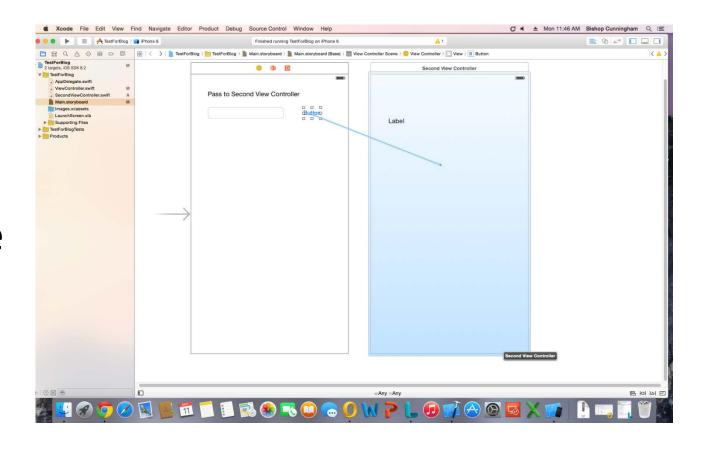
    LaunchScreen.storyboard

                                       class SecondViewController: UIViewController {
     Info.plist
                                           override func viewDidLoad() {
     SecondViewController.swift
                                               super.viewDidLoad()
 Products
                                               // Do any additional setup after loading the view.
                                           override func didReceiveMemoryWarning() {
                                               super.didReceiveMemoryWarning()
                                               // Dispose of any resources that can be recreated.
                                           // MARK: - Navigation
                                           // In a storyboard-based application, you will often want to do a little preparat
                                           override func prepare(for segue: UIStoryboardSegue, sender: Any?) {
                                               // Get the new view controller using segue.destinationViewController.
                                               // Pass the selected object to the new view controller.
```

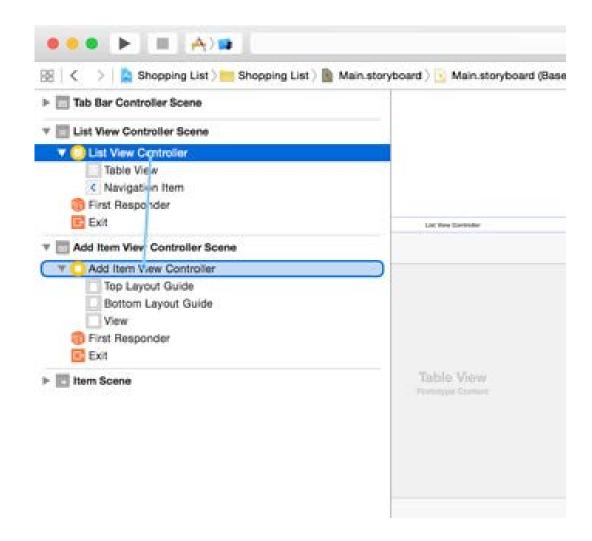
- Now to connect the new swift object to the storyboard
- Subclass the new screen in the storyboard as shown



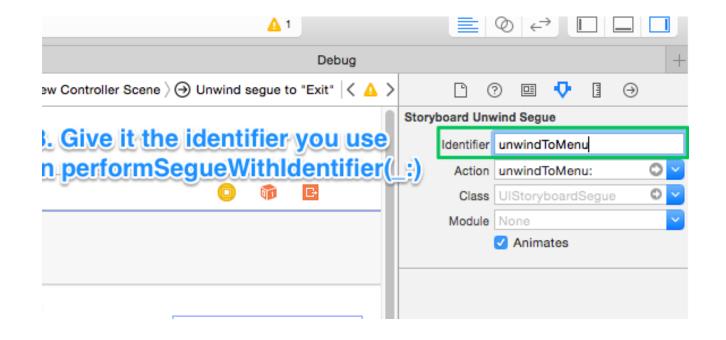
- Segues are used to transition from 1 page to another.
- Typically we connect a button to another page via a segue by:
 - Mousing over a button, hold down the control key and drag to the new page to create a segue



 Alternative way using document view.



- You would next need to give your segue an identifier.
- Highlight your segue, then add your new identifier in the attributes inspector.



Then you can invoke that segue using:

performSegue(withIdentifier: "segueName", sender: nil)

- When a segue is created, it enables the button to invoke an internal event handler which transitions to the new page upon clicking.
- When the new page is displayed on screen, its corresponding view controller is instantiated and put in memory.
- Upon which viewDidLoad is executed.

- There are a few view methods that exist in UIViewController
- Each executes at a certain time in the view lifecycle
 - viewDidLoad()
 - viewWillAppear()
 - viewDidAppear()
 - viewDidDisappear()
 - loadView()
 - awake()

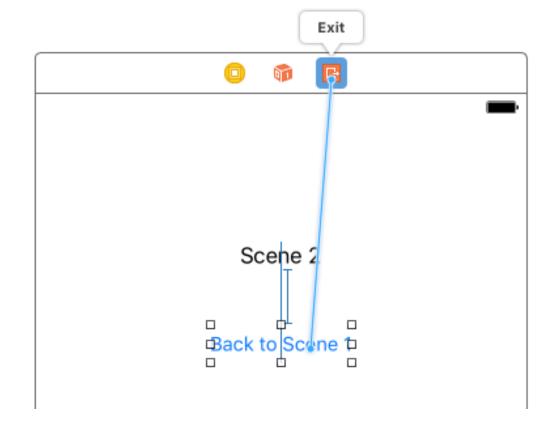
- Now that we know how to go forward to a new page.
- How about going back a page.
- Following the same process of creating a segue would work but it's a no-no!
- Why? Because segues instantiate new View Controller objects in memory and we want to free memory.

- The correct way is to unwind your segue.
- It's a 2 step process:
 - Create a special method in the View Controller file that the page will unwind to.
 - Control-drag your unwind segue in the storyboard.

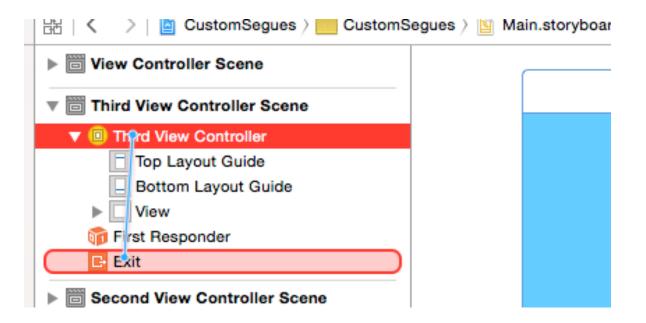
• First, create a special method:

```
@IBAction func unwindToThisView(sender : UIStoryboardSegue)
{
    // typically empty unless special code needed
}
```

- Second, control drag from your button to the exit icon at the top of the View Controller object in your storyboard.
- Choose the correct segue method name from the menu.



Alternative way –
 control dragging using
 the document view.



Dressing Up Your App

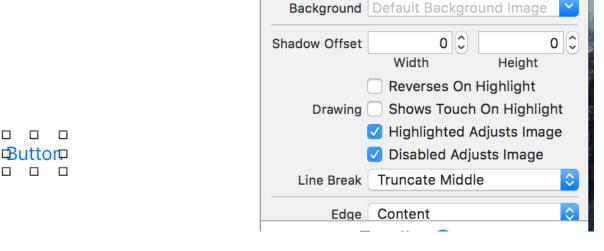
Examples:



THE PRIVENTURES OF

Dressing Up Your App

- After adding a button to the screen, you can set the image, setting here:
- Note that you should have added your image files to your project first.



Shadow Color

Image | Default Image

Default

Use Background! – Why?

Dressing Up Your App

- A couple of notes about images:
 - JPEG files do not support clear backgrounds but they have small file sizes
 - PNG & GIF files support transparent backgrounds
- Rule of thumb:
 - Used JPEG for backgrounds
 - Use PNG/GIF for button images

Exercise 1

- Create an iPhone app that has the following:
 - Splash page, home page and 3 sub pages
 - One sub page will have a web view
 - One sub page will have a slider that displays its adjusted value
 - One sub page will have a segmented controller that displays its updated value.
- Dress up your app with nice graphics, both backgrounds and buttons.