Stanford CS193p

Developing Applications for iOS
Fall 2013-14

Today

- Demo

  Polymorphism with Controllers in Matchismo

  How to change the class of a Controller in a storyboard

- Multiple MVCs in an Application

  `UINavigationController`

  `UITabBarController`

- Demo

  Attributor Stats

Demo

- Making a Generic Controller in Matchismo

  Polymorphism with Controllers in Matchismo

  Get rid of PlayingCardDeck in CardGameViewController.

  How to change the class of a Controller in a storyboard

Multiple MVCs

- Why?

  When your application gets more features than can fit in one MVC.

- How to add a new MVC to your storyboard

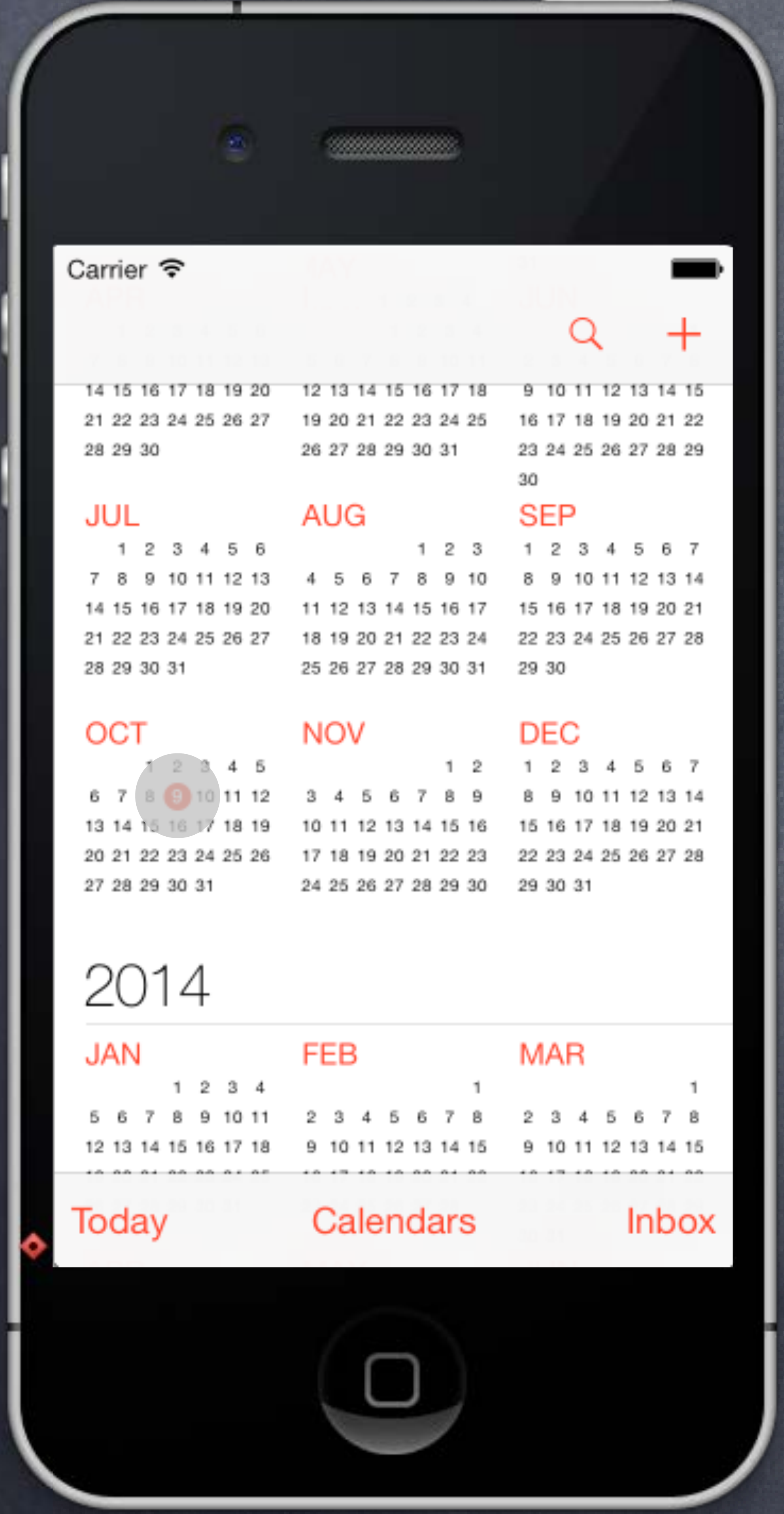
  Drag "View Controller" from Object Palette.
  Create a subclass of `UIViewController` using New File menu item.
  Set that subclass as the class of your new Controller in the Attributes Inspector.

- How to present this new MVC to the user

  `UINavigationController`

  `UITabBarController`

  Other mechanisms we'll talk about later in the course (popover, modal, etc.).



14 15 16 17 18 19 20
21 22 23 24 25 26 27
28 29 30

12 13 14 15 16 17 18
19 20 21 22 23 24 25
26 27 28 29 30 31

9 10 11 12 13 14 15
16 17 18 19 20 21 22
23 24 25 26 27 28 29
30

JUL
1 2 3 4 5 6
7 8 9 10 11 12 13
14 15 16 17 18 19 20
21 22 23 24 25 26 27
28 29 30 31

AUG
1 2 3
4 5 6 7 8 9 10
11 12 13 14 15 16 17
18 19 20 21 22 23 24
25 26 27 28 29 30 31

SEP
1 2 3 4 5 6 7
8 9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30

OCT
1 2 3 4 5
6 7 8 9 10 11 12
13 14 15 16 17 18 19
20 21 22 23 24 25 26
27 28 29 30 31

NOV
1 2
3 4 5 6 7 8 9
10 11 12 13 14 15 16
17 18 19 20 21 22 23
24 25 26 27 28 29 30

DEC
1 2 3 4 5 6 7
8 9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30 31

2014

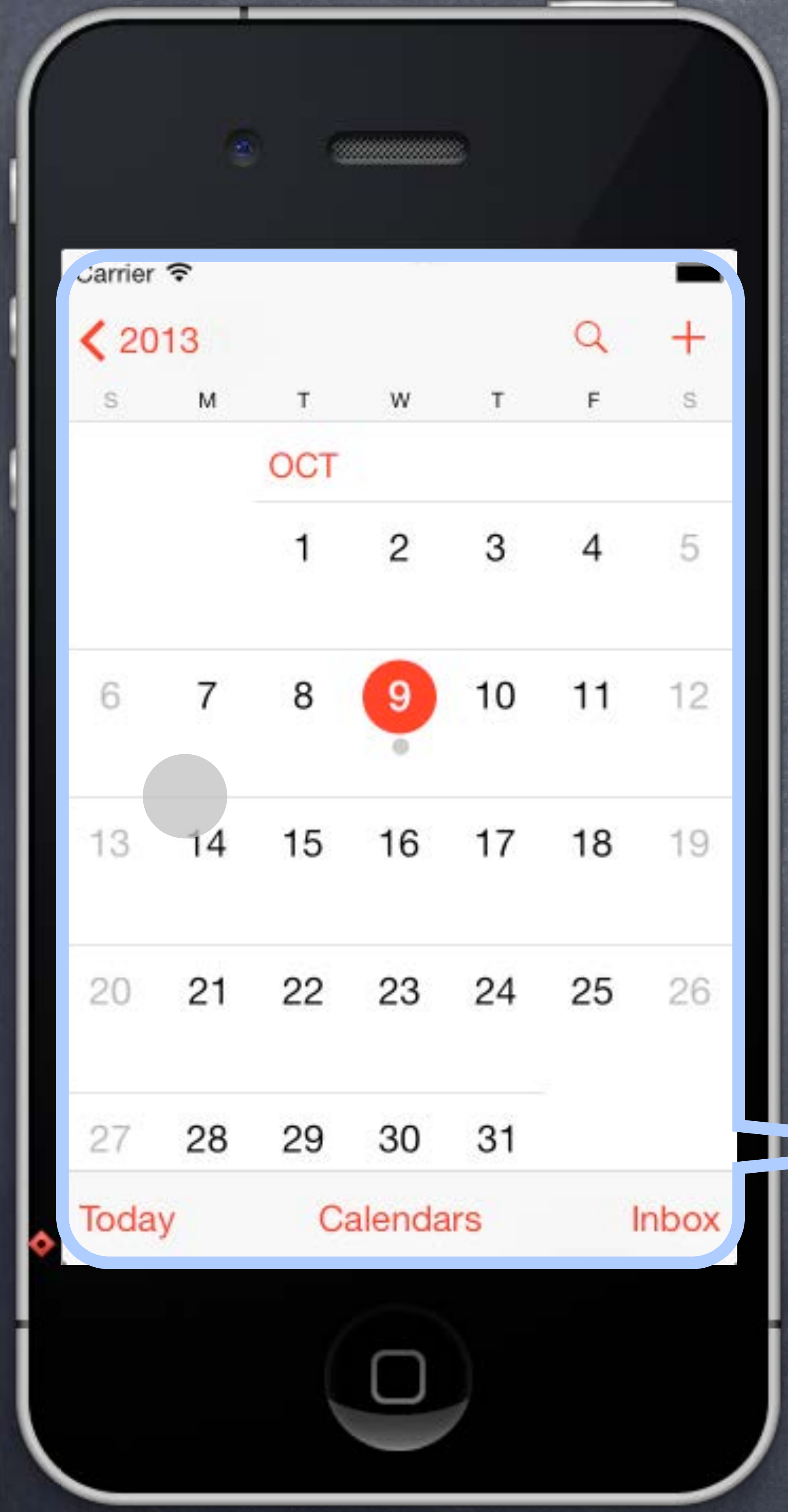
JAN
1 2 3 4
5 6 7 8 9 10 11
12 13 14 15 16 17 18

FEB
1
2 3 4 5 6 7 8
9 10 11 12 13 14 15

MAR
1
2 3 4 5 6 7 8
9 10 11 12 13 14 15

UINavigationController

- When to use it?
  When the user wants to “dive down” into more detail.



UINavigationController

- When to use it?

  When the user wants to "dive down" into more detail.

- How does it work?

  Encloses other MVCs (like the Year MVC and the Month MVC).
  Touches in one MVC "segue" to the other MVCs.

This is the UINavigationController's View.

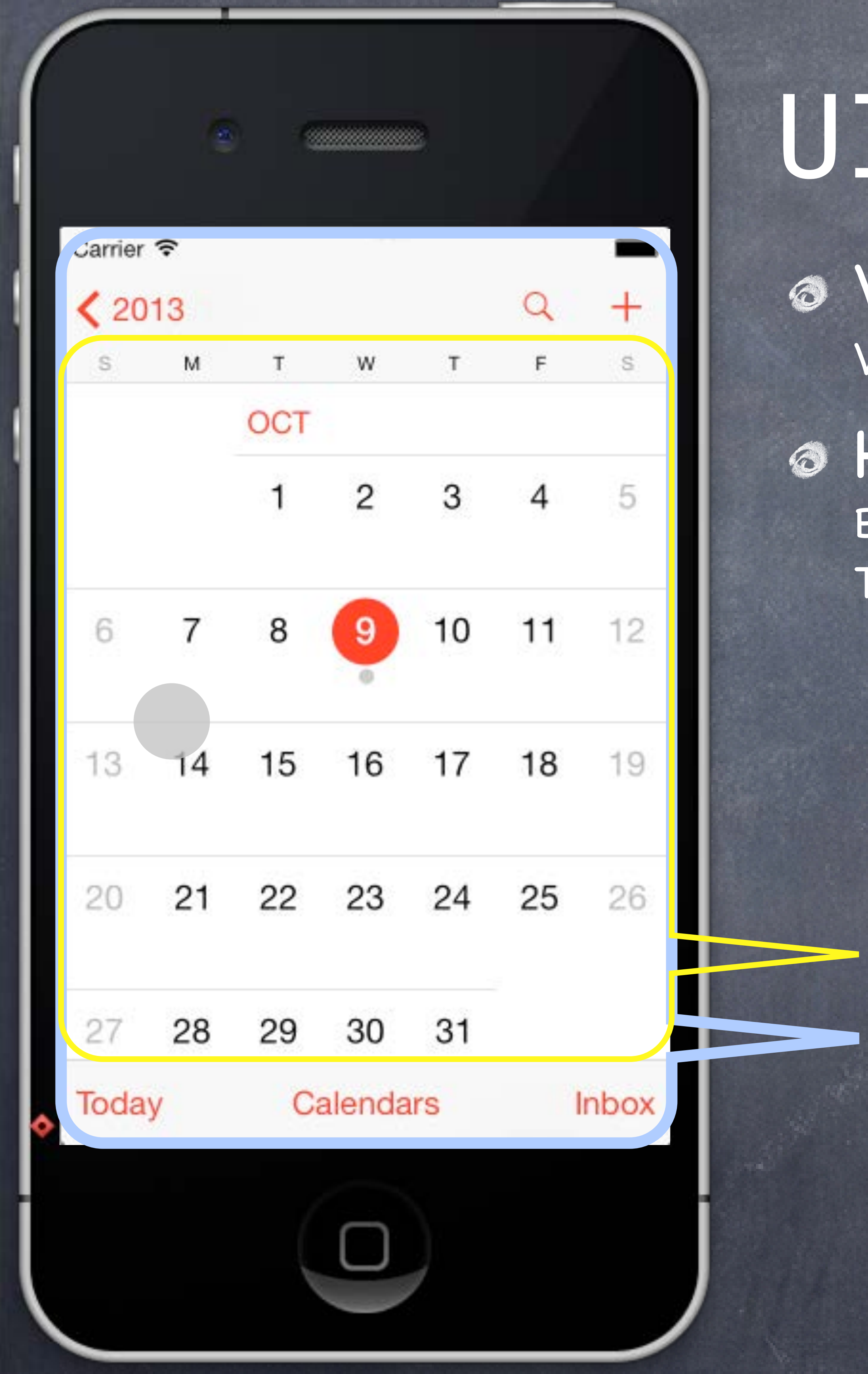
UINavigationController

- When to use it?

  When the user wants to "dive down" into more detail.

- How does it work?

  Encloses other MVCs (like the Year MVC and the Month MVC).
  Touches in one MVC "segue" to the other MVCs.



This is a Month MVC's View.

This is the UINavigationController's View.

UINavigationController

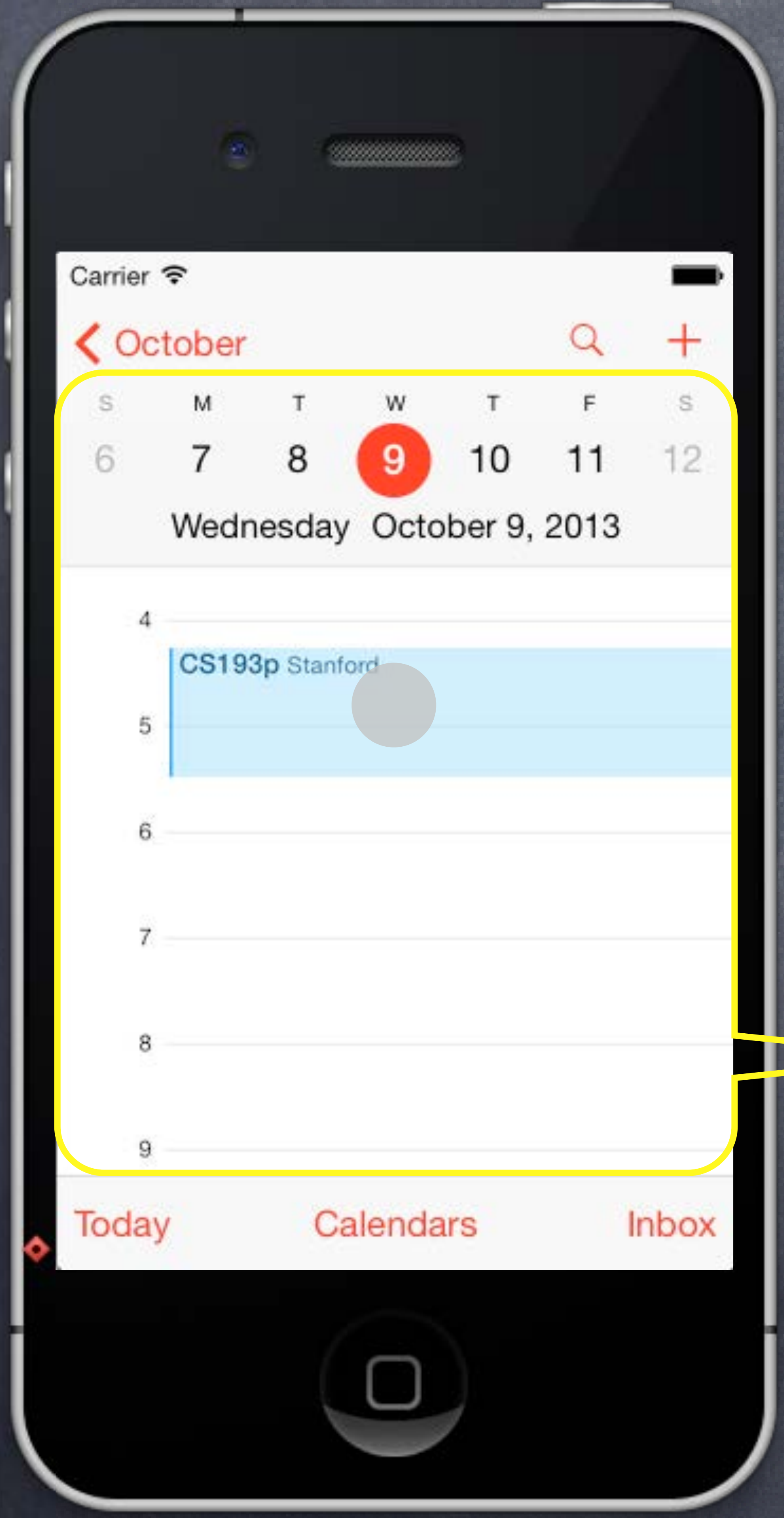
- When to use it?

  When the user wants to "dive down" into more detail.

- How does it work?

  Encloses other MVCs (like the Year MVC and the Month MVC).

  Touches in one MVC "segue" to the other MVCs.



This is a Day MVC's View.

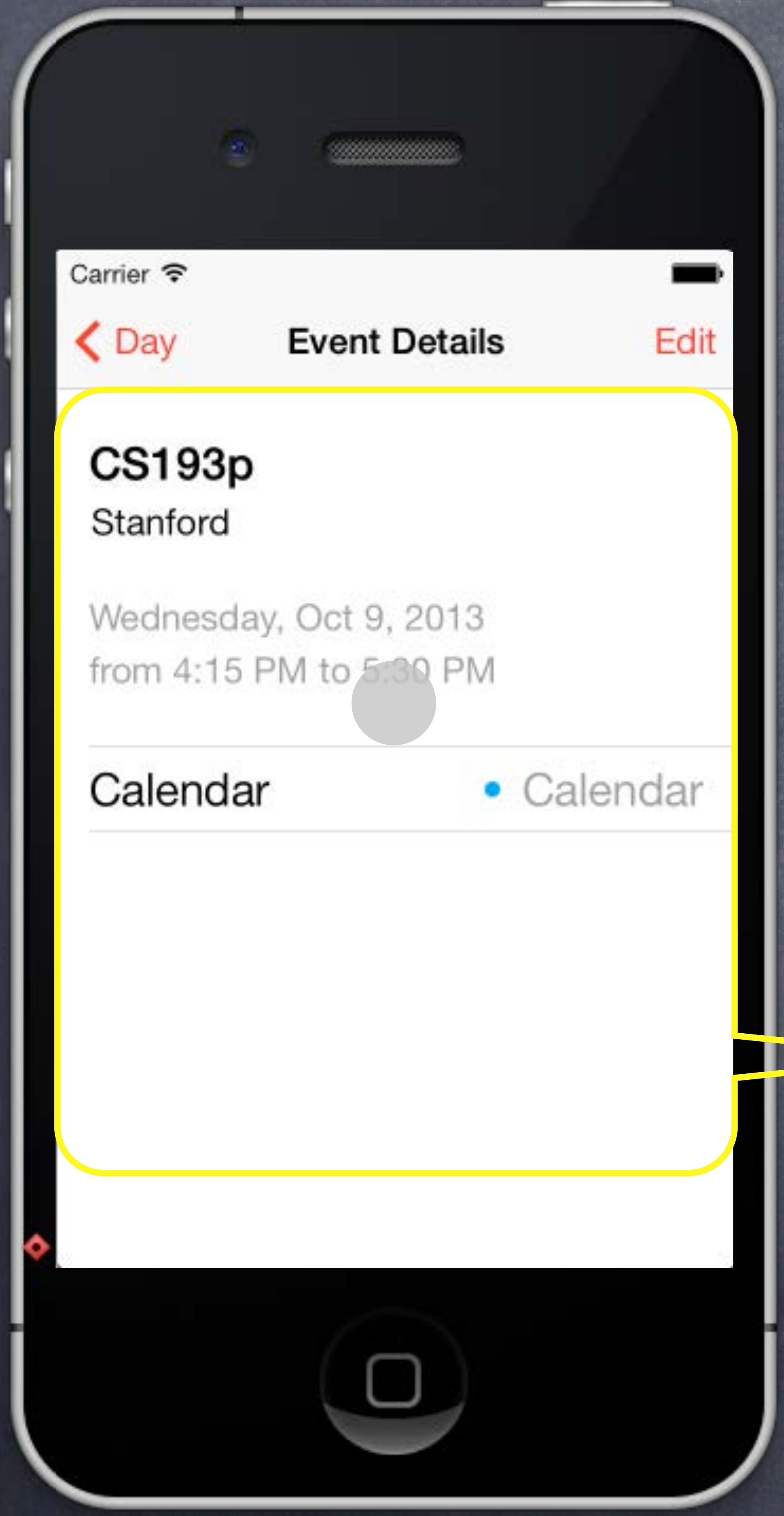
UINavigationController

- When to use it?

  When the user wants to "dive down" into more detail.

- How does it work?

  Encloses other MVCs (like the Year MVC and the Month MVC).
  Touches in one MVC "segue" to the other MVCs.



Carrier

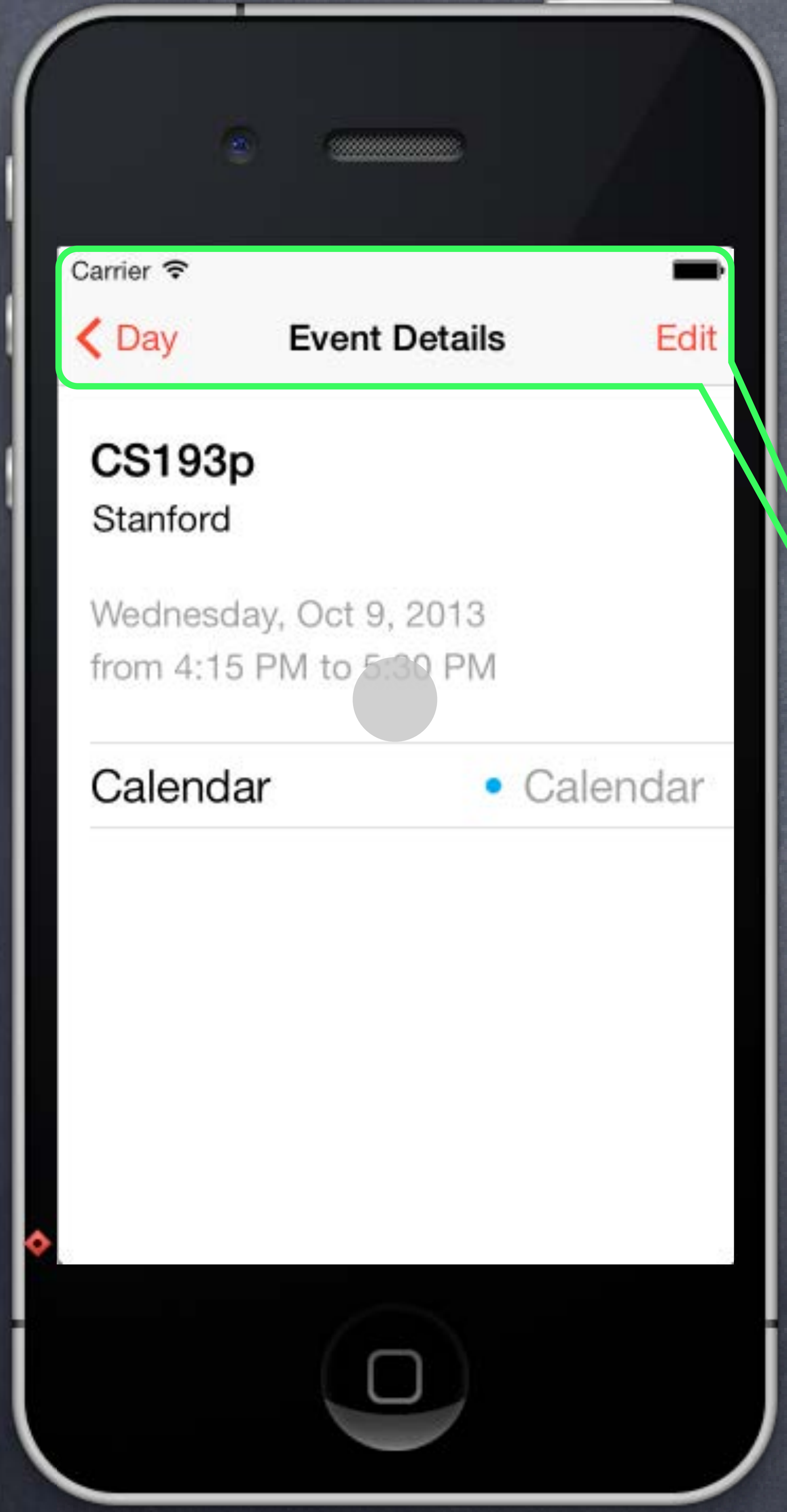
Day | Event Details | Edit

CS193p
Stanford

Wednesday, Oct 9, 2013
from 4:15 PM to 5:30 PM

Calendar • Calendar

This is a Calendar Event MVC's View.



Carrier

Day

Event Details

Edit

CS193p

Stanford

Wednesday, Oct 9, 2013
from 4:15 PM to 5:30 PM

Calendar • Calendar

UINavigationController

- When to use it?

  When the user wants to "dive down" into more detail.

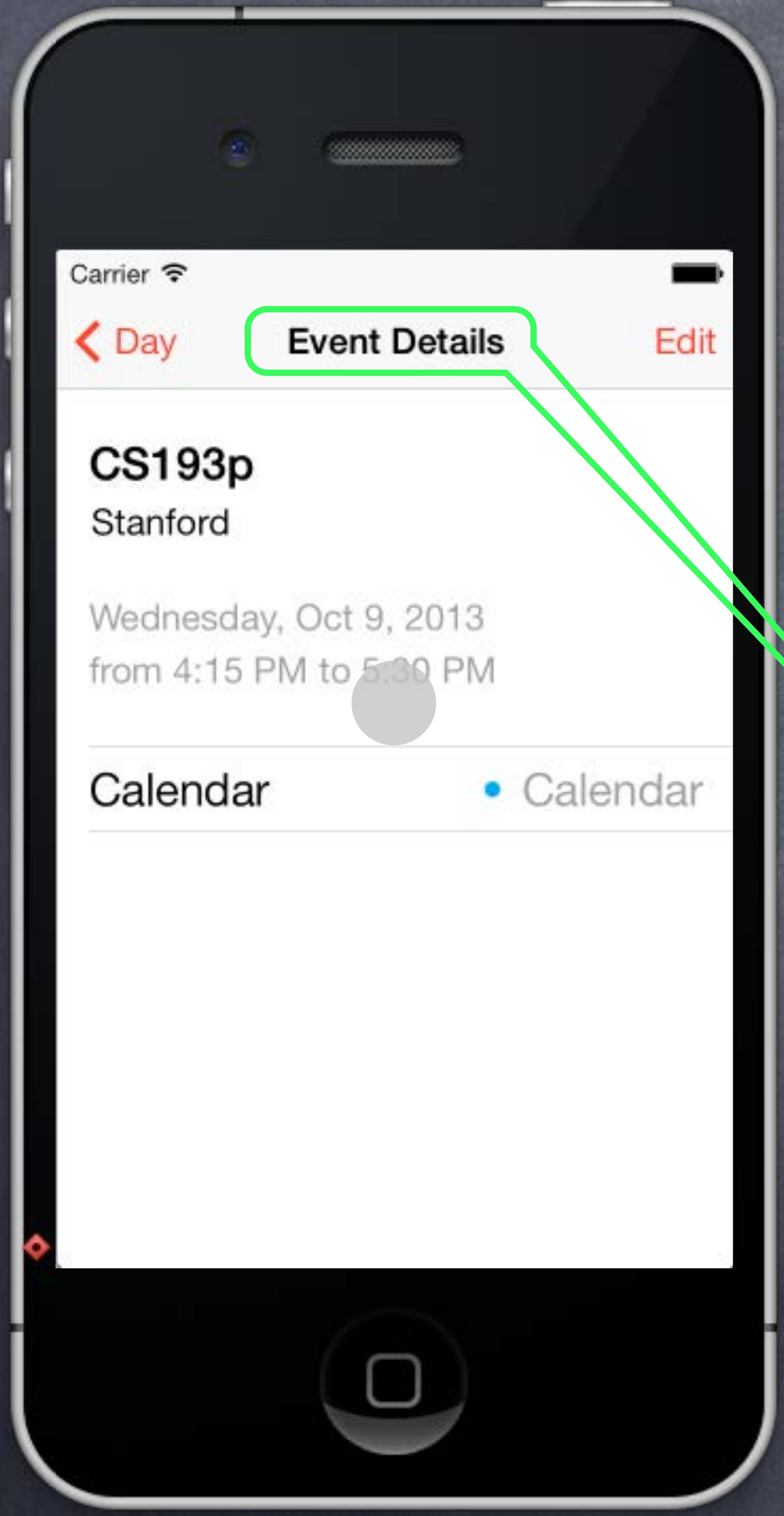
- How does it work?

  Encloses other MVCs (like the Year MVC and the Month MVC).

  Touches in one MVC "segue" to the other MVCs.

- Components of a UINavigationController

  Navigation Bar (contents determined by embedded MVC's navigationItem).



Carrier

Day

Event Details

Edit

CS193p

Stanford

Wednesday, Oct 9, 2013

from 4:15 PM to 5:30 PM

Calendar

Calendar

UINavigationController

- When to use it?

  When the user wants to "dive down" into more detail.

- How does it work?

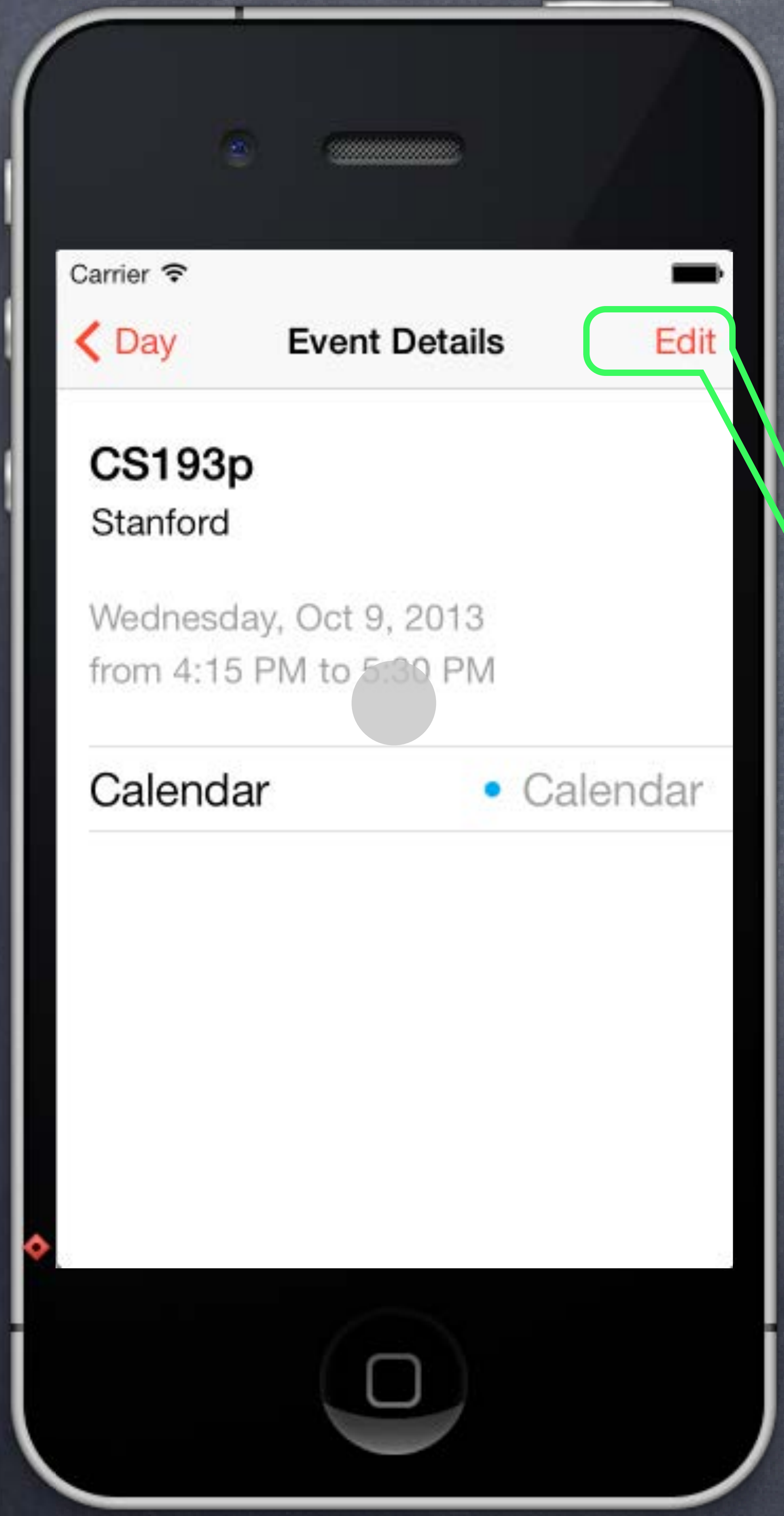
  Encloses other MVCs (like the Year MVC and the Month MVC).

  Touches in one MVC "segue" to the other MVCs.

- Components of a UINavigationController

  Navigation Bar (contents determined by embedded MVC's navigationItem).

  Title (by default is title property of the embedded MVC)



UINavigationController

- When to use it?

  When the user wants to "dive down" into more detail.

- How does it work?

  Encloses other MVCs (like the Year MVC and the Month MVC).

  Touches in one MVC "segue" to the other MVCs.

- Components of a UINavigationController

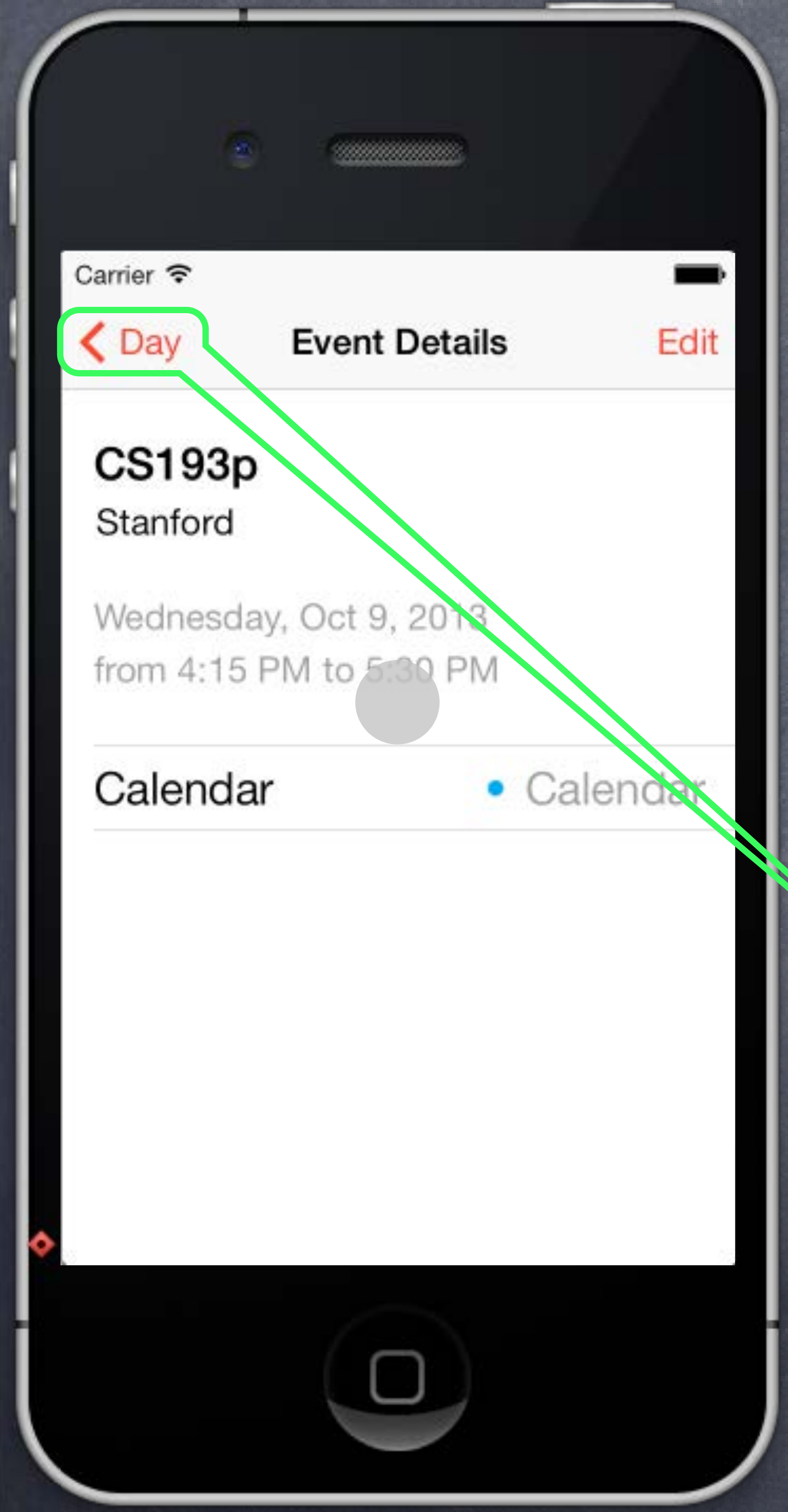
  Navigation Bar (contents determined by embedded MVC's navigationItem).

  Title (by default is title property of the embedded MVC)

  Embedded MVC's navigationItem.rightBarButtonItems

  (an NSArray of UIBarButtonItems)

UINavigationController



Carrier

Day Event Details Edit

CS193p

Stanford

Wednesday, Oct 9, 2013

from 4:15 PM to 5:30 PM

Calendar • Calendar

- **When to use it?**

  When the user wants to "dive down" into more detail.

- **How does it work?**

  Encloses other MVCs (like the Year MVC and the Month MVC).

  Touches in one MVC "segue" to the other MVCs.

- **Components of a UINavigationController**

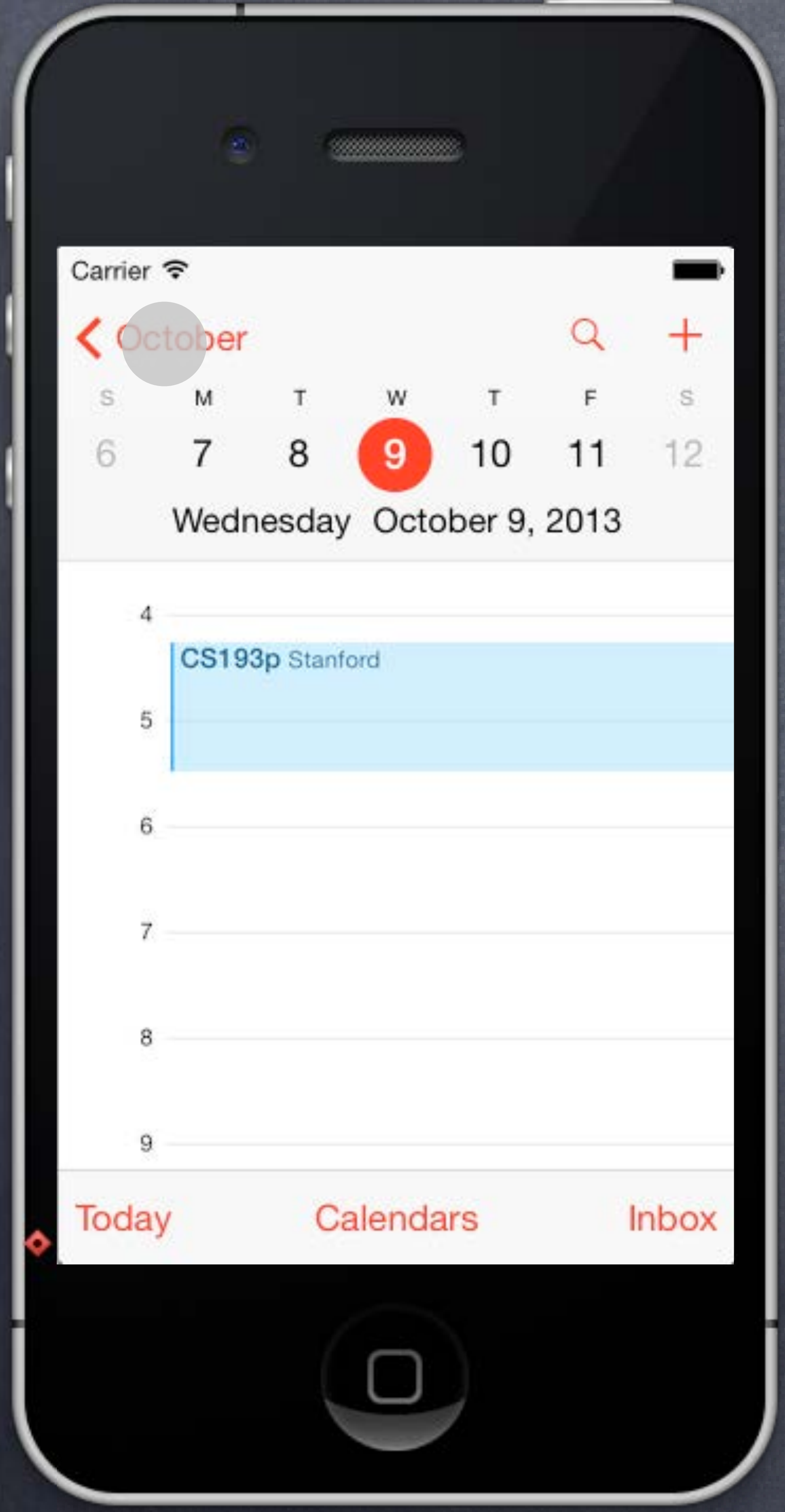
  Navigation Bar (contents determined by embedded MVC's `navigationItem`).

  Title (by default is `title` property of the embedded MVC)

  Embedded MVC's `navigationItem.rightBarButtonItems`

  (an `NSArray` of `UIBarButtonItems`)

  Back Button (automatic)



UINavigationController

- When to use it?

  When the user wants to "dive down" into more detail.

- How does it work?

  Encloses other MVCs (like the Year MVC and the Month MVC).

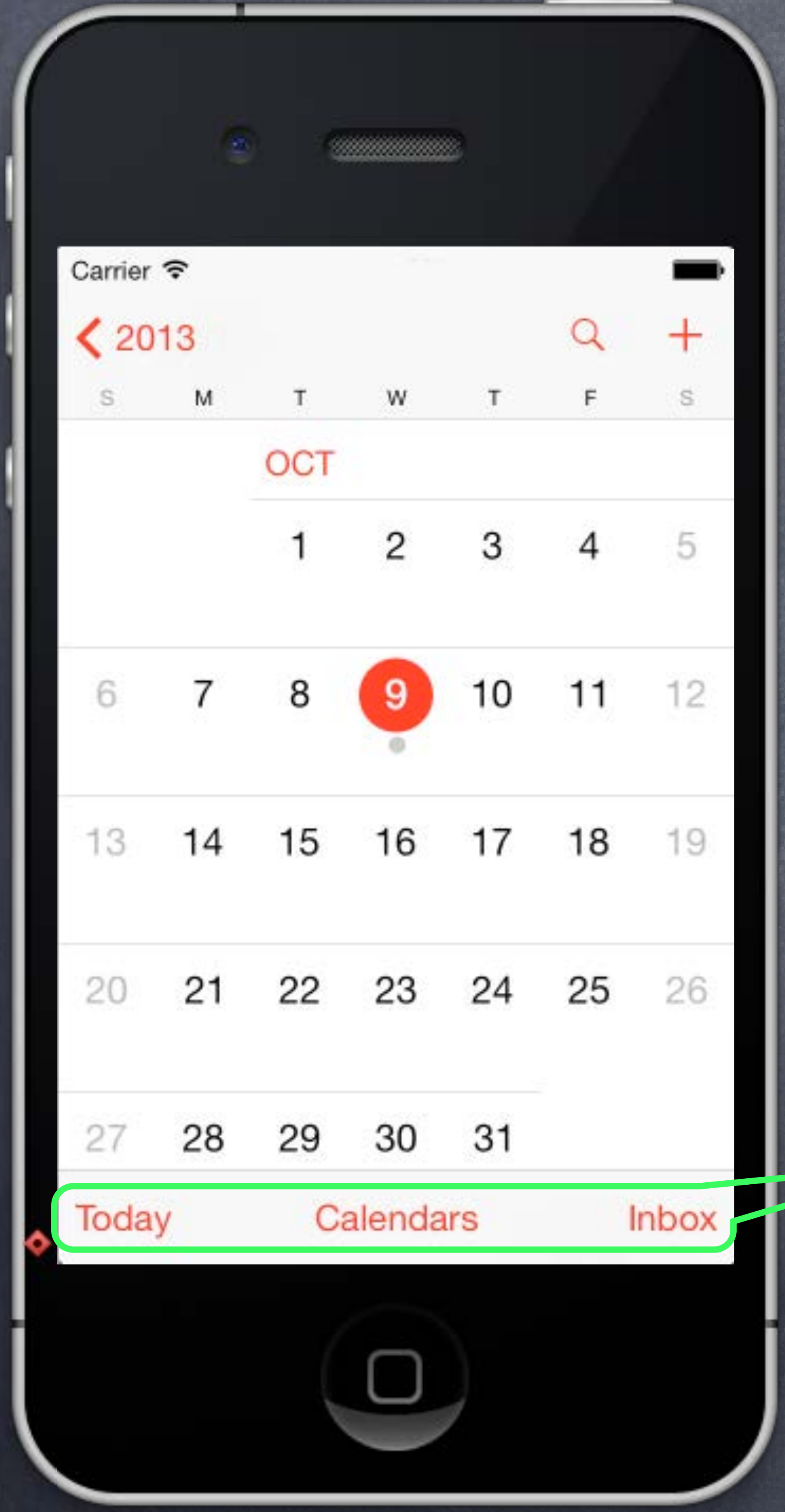
  Touches in one MVC "segue" to the other MVCs.

- Components of a UINavigationController

  Navigation Bar (contents determined by embedded MVC's navigationItem).

  - Title (by default is title property of the embedded MVC)
  - Embedded MVC's navigationItem.rightBarButtonItems
    - (an NSArray of UIBarButtonItems)
  - Back Button (automatic)

UINavigationController



- When to use it?

  When the user wants to "dive down" into more detail.

- How does it work?

  Encloses other MVCs (like the Year MVC and the Month MVC).

  Touches in one MVC "segue" to the other MVCs.

- Components of a UINavigationController

  Navigation Bar (contents determined by embedded MVC's navigationItem).

  Title (by default is title property of the embedded MVC)

  Embedded MVC's navigationItem.rightBarButtonItems

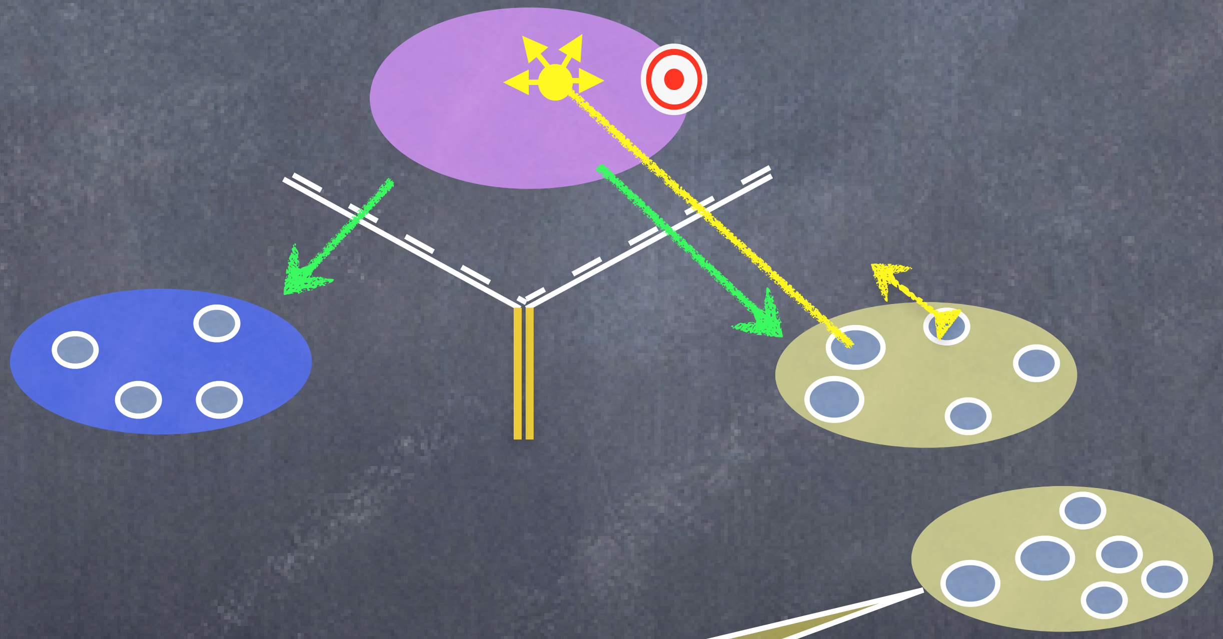
  (an NSArray of UIBarButtonItems)

  Back Button (automatic)

  Embedded MVC's toolbarItems property

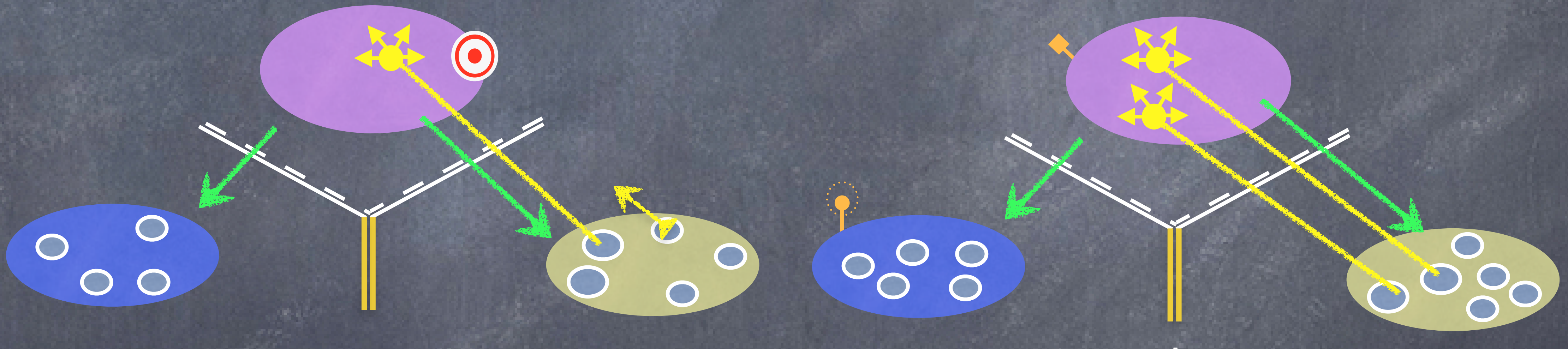
  (also an NSArray of UIBarButtonItems)

MVCs working together



I want more features, but it doesn't make sense to put them all in one MVC!

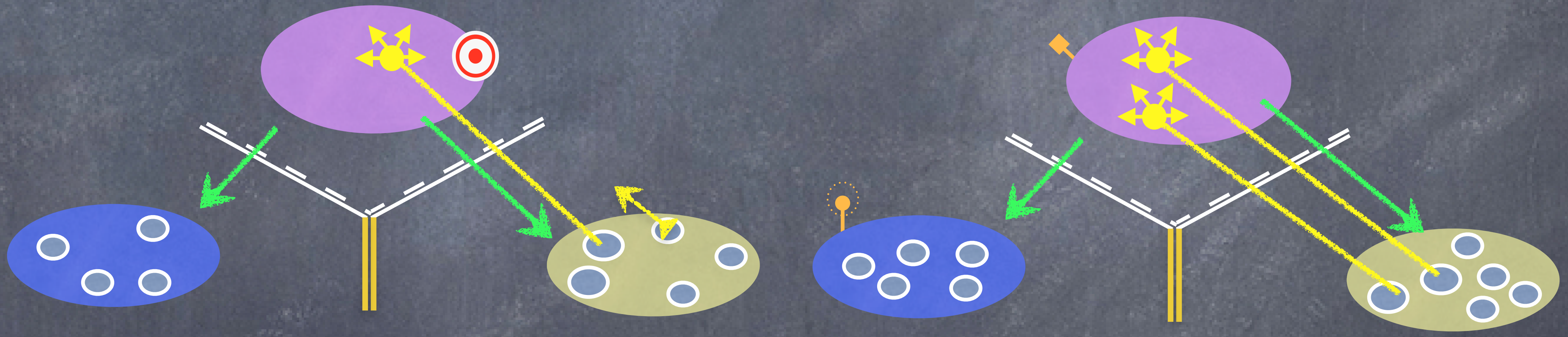
MVCs working together



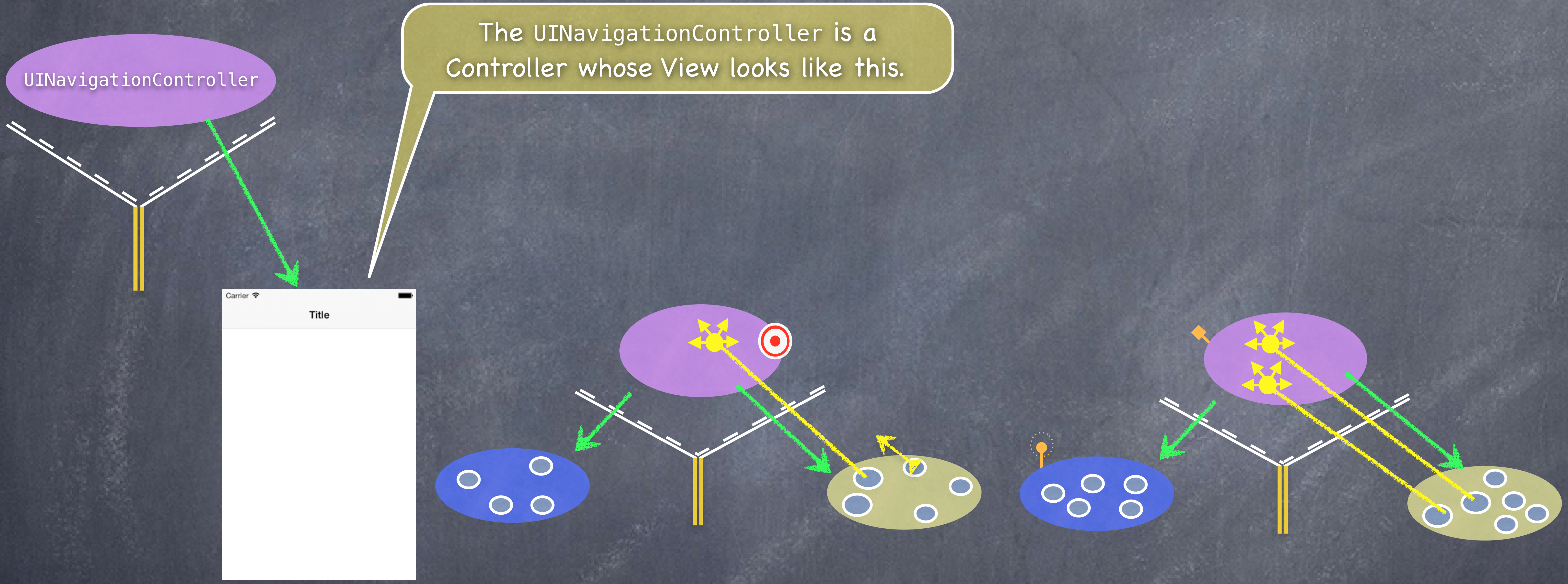
So I create a new MVC to encapsulate that functionality.

MVCs working together

If the relationship between these two MVCs is “more detail,” we use a UINavigationController to let them share the screen.

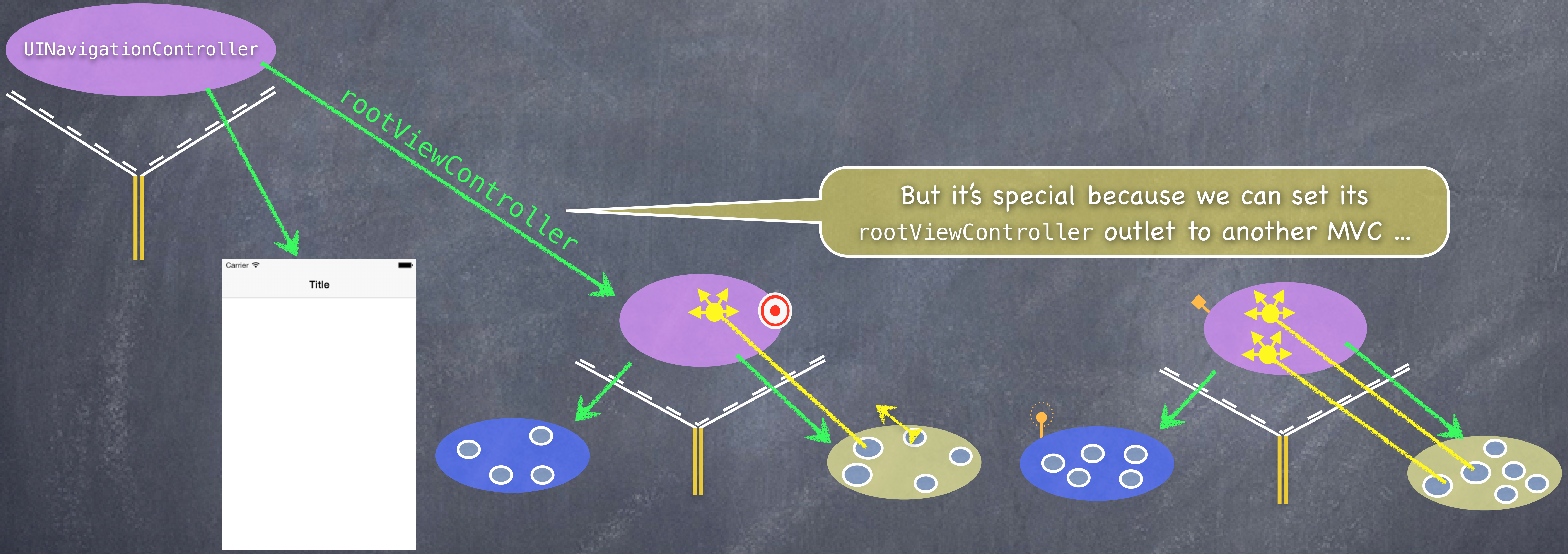


MVCs working together



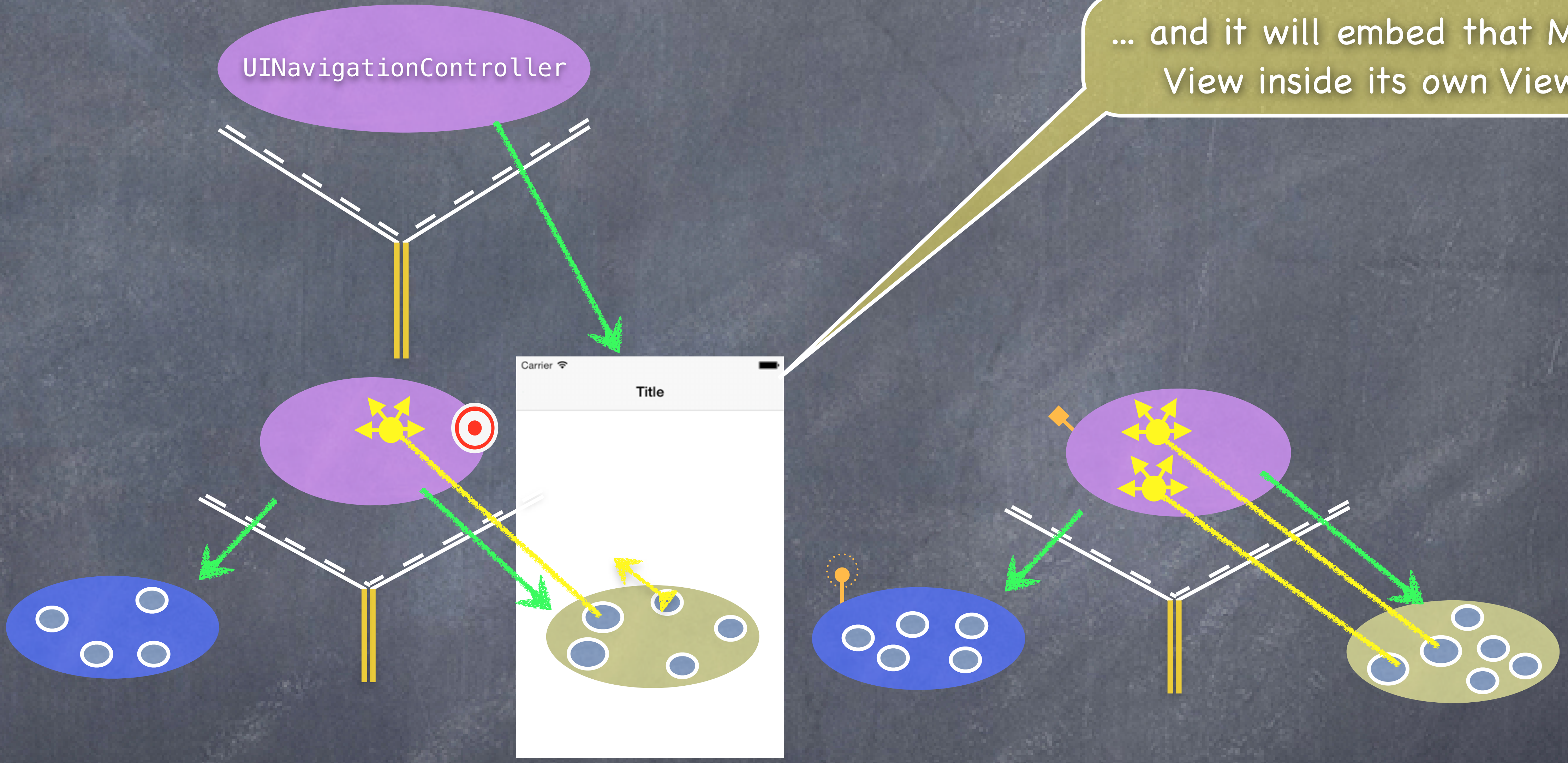
The UINavigationController is a Controller whose View looks like this.

MVCs working together



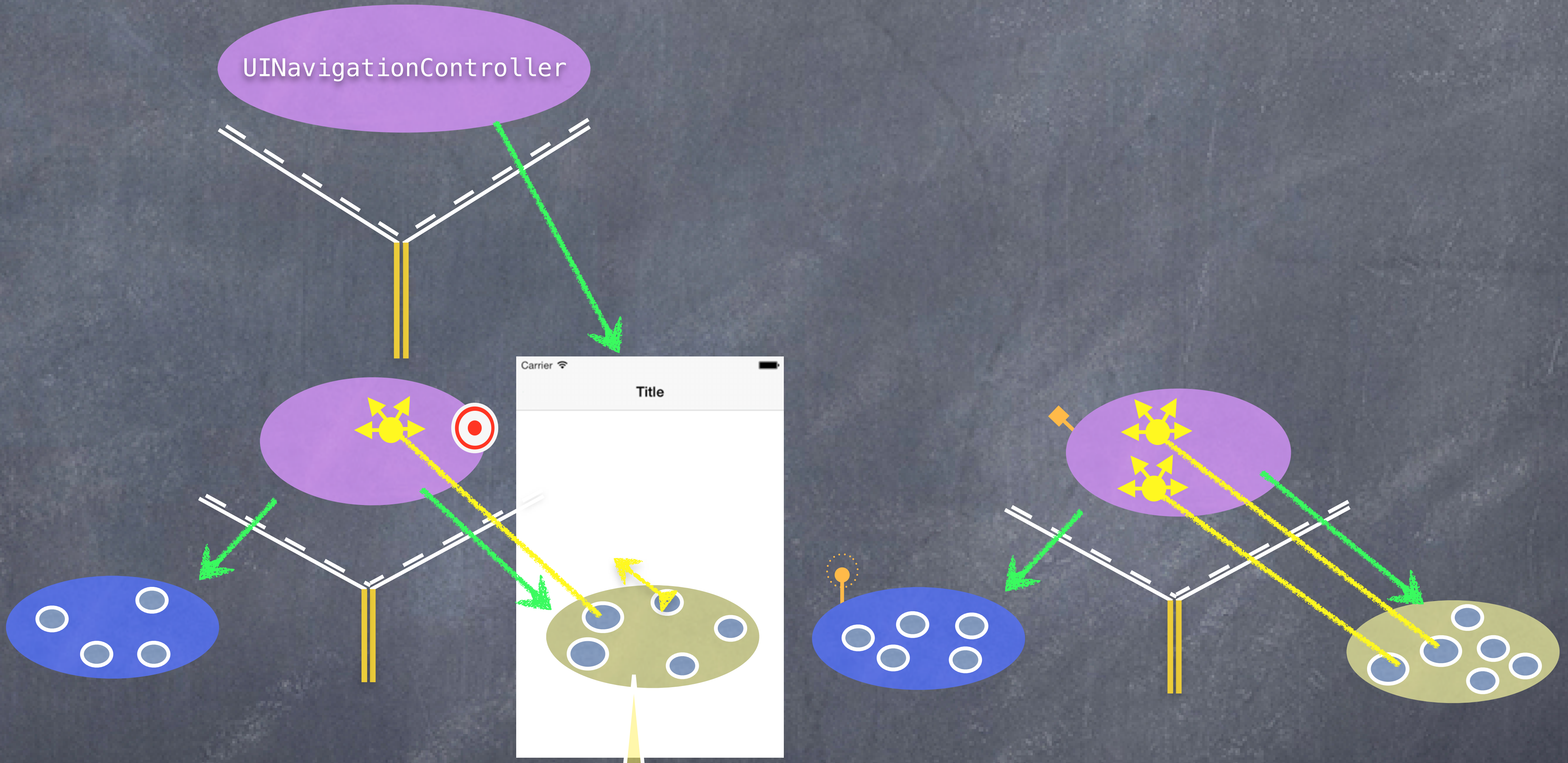
But it's special because we can set its `rootViewController` outlet to another MVC ...

MVCs working together



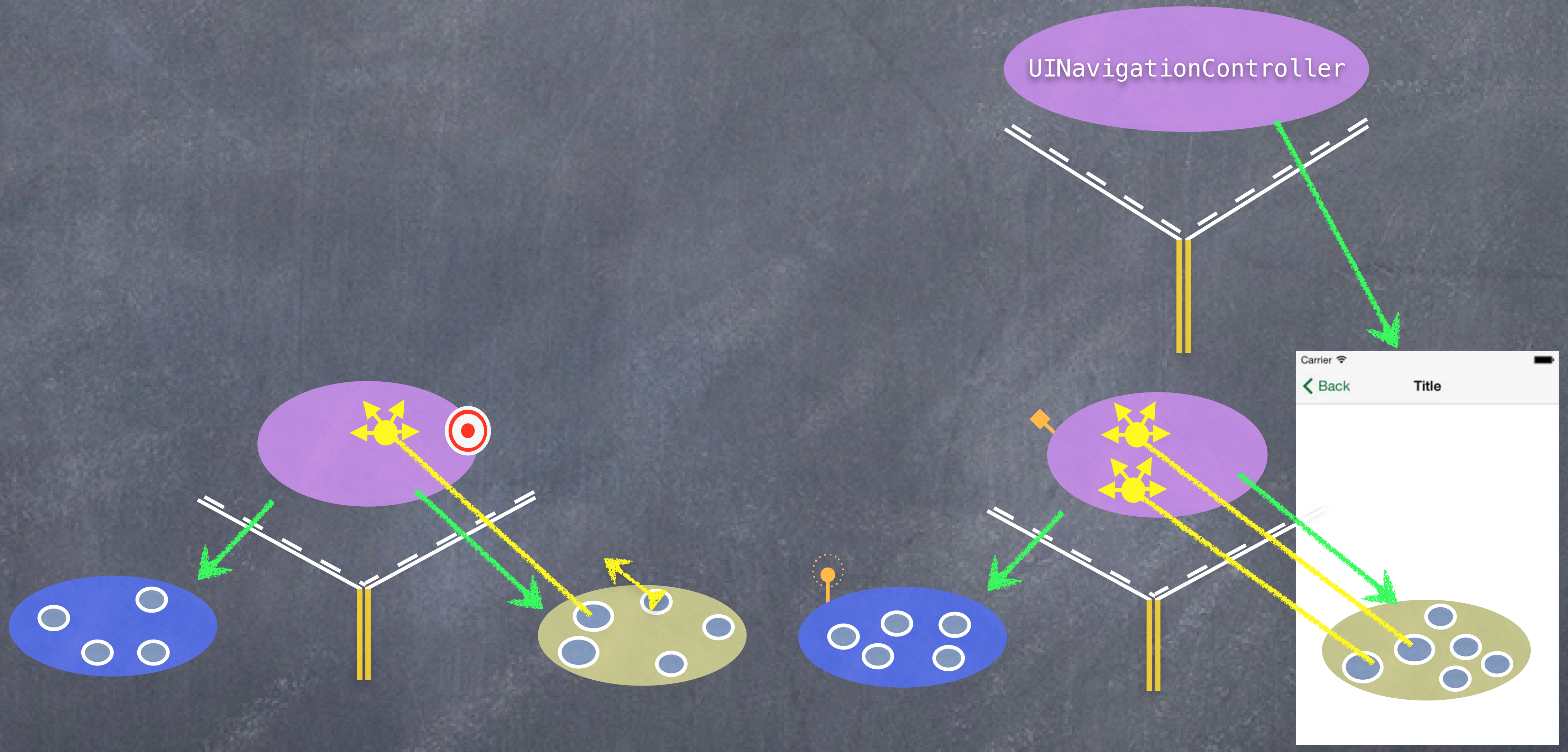
... and it will embed that MVC's View inside its own View.

MVCs working together



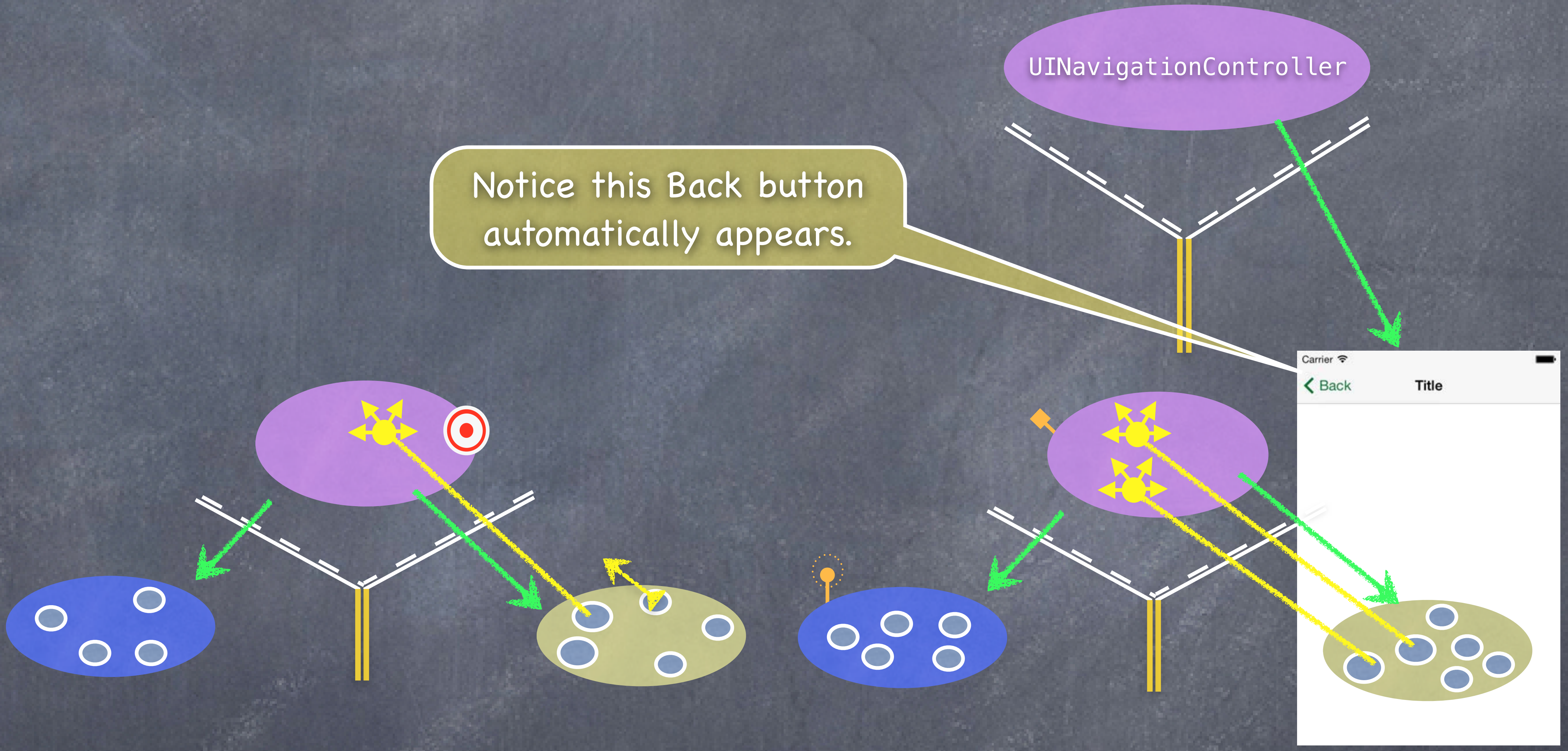
Then a UI element in this View (e.g. a UIButton) can segue to the other MVC and its View will now appear in the UINavigationController instead.

MVCs working together



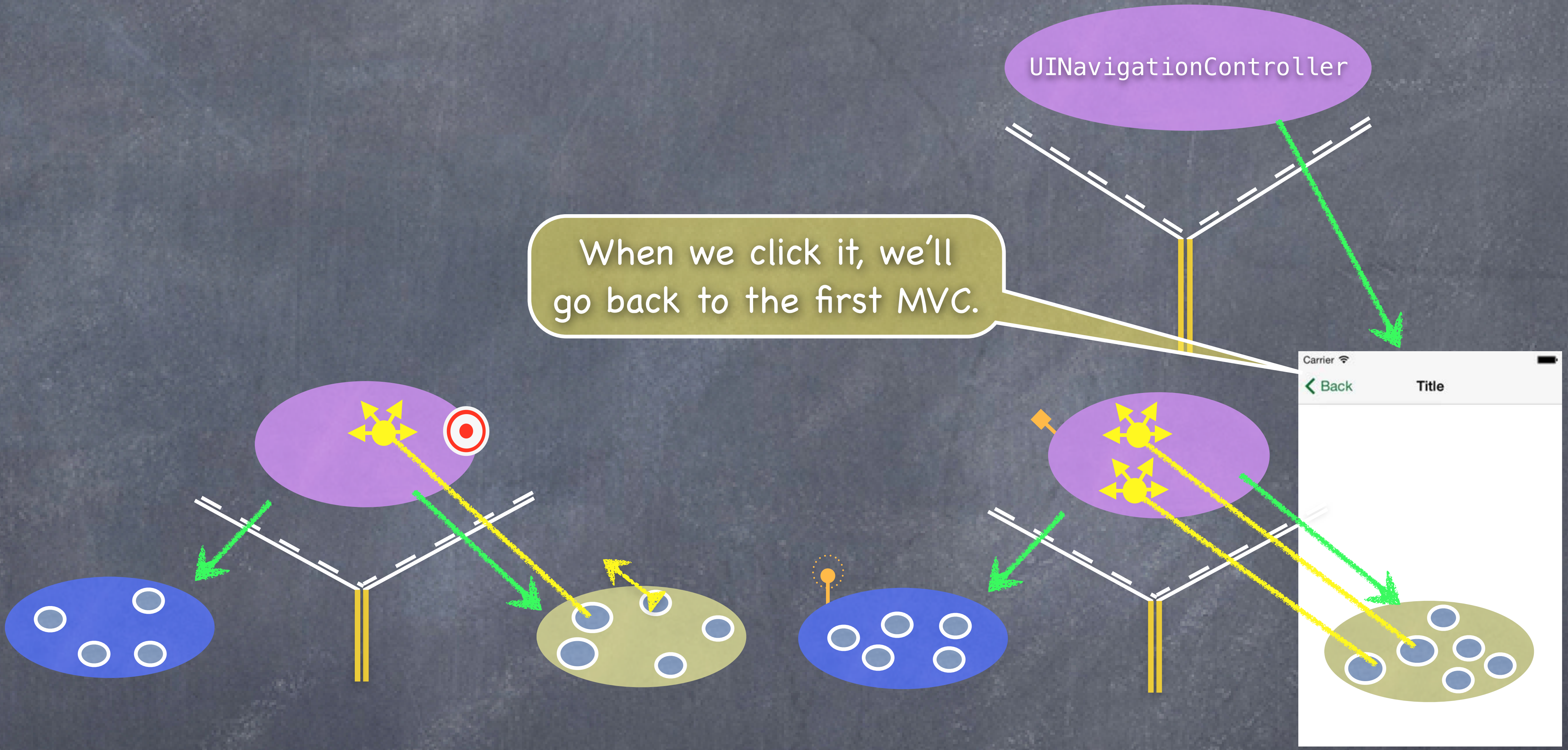
We call this kind of segue a "push segue".

MVCs working together

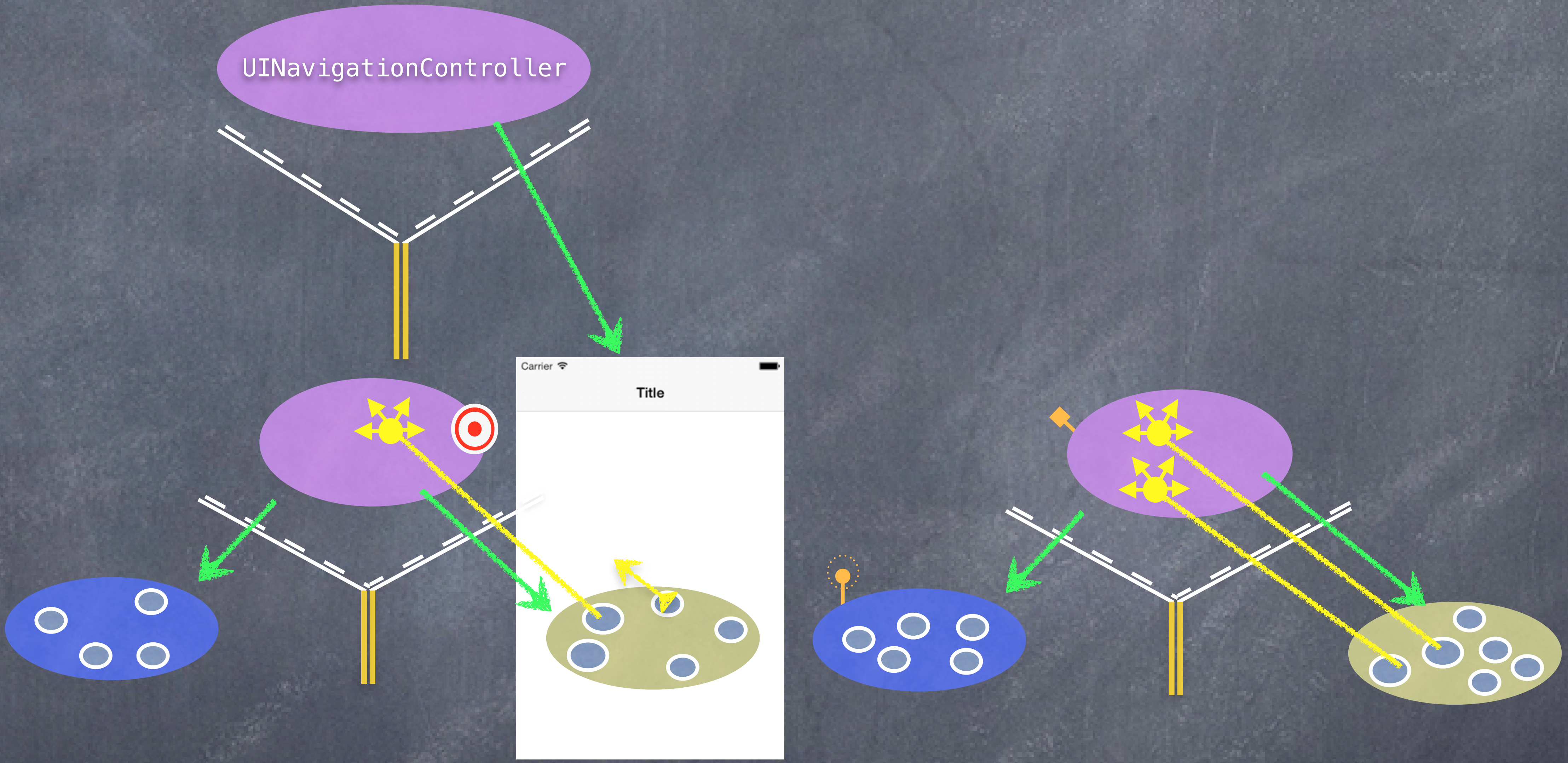


Notice this Back button automatically appears.

MVCs working together



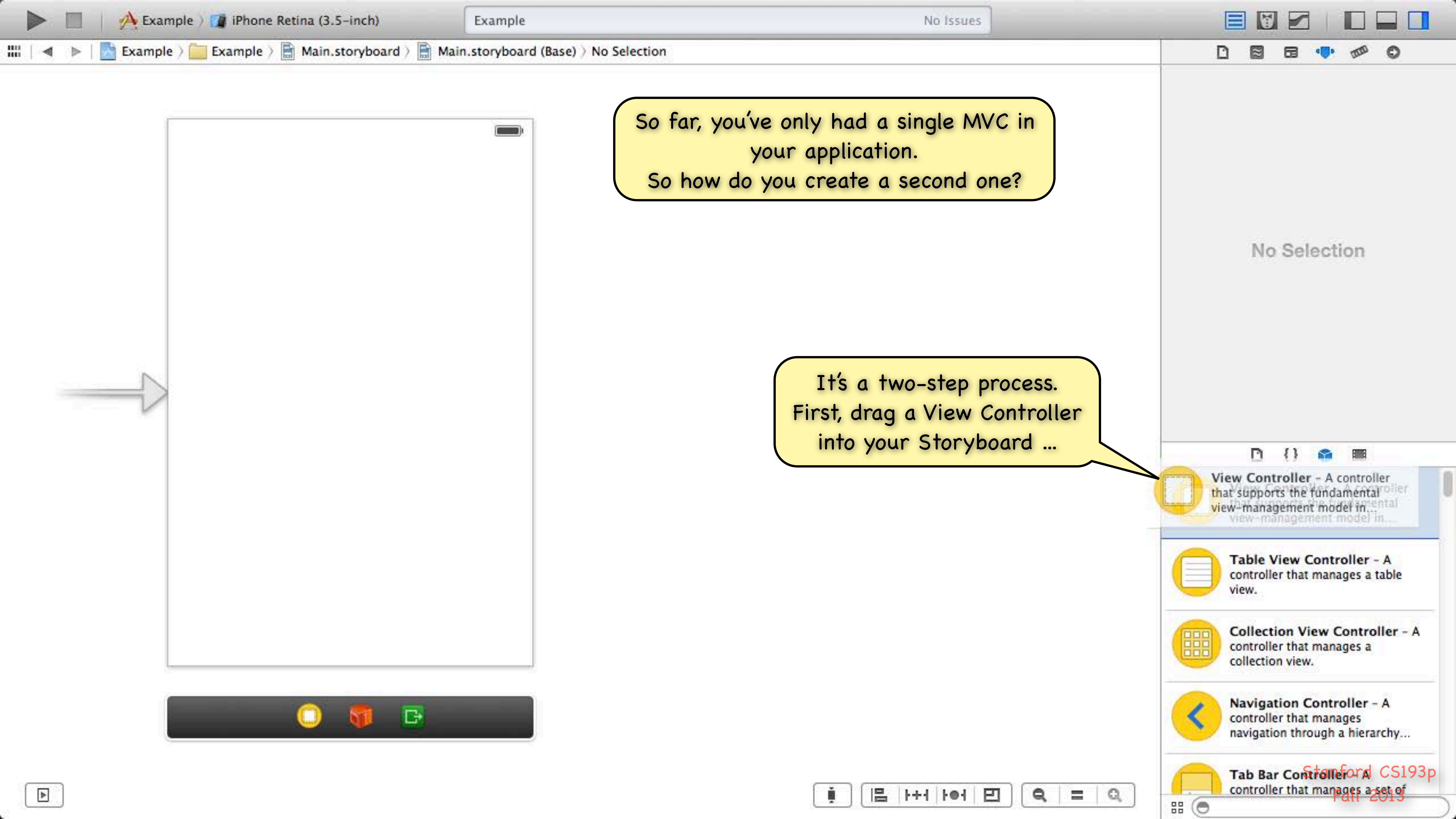
MVCs working together



Segues

- Let's talk about how the segue gets set up first

  Then we'll look at how we create a `UINavigationController` in our storyboard.



So far, you've only had a single MVC in your application.
So how do you create a second one?

It's a two-step process.
First, drag a View Controller into your Storyboard …

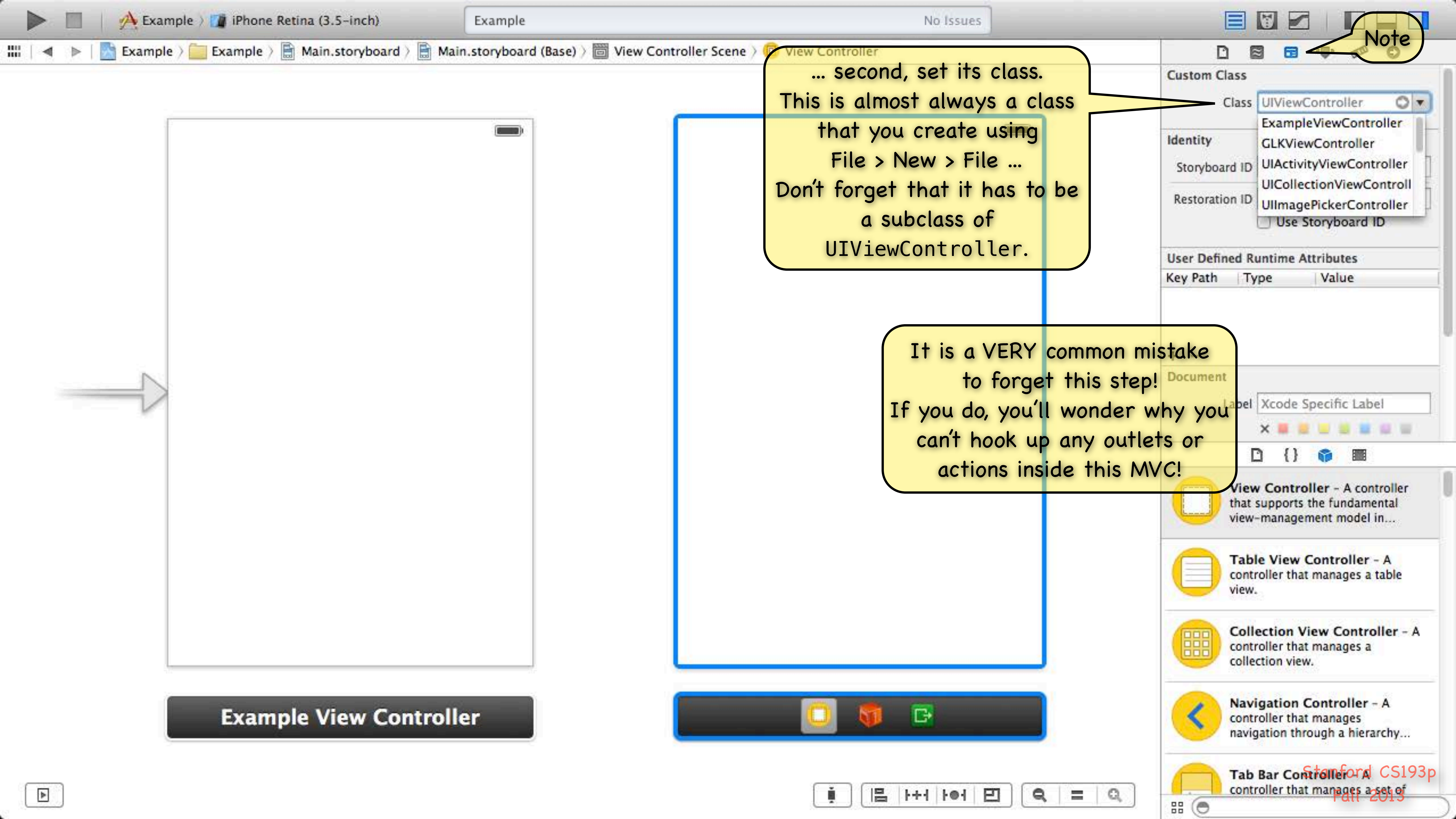
View Controller – A controller that supports the fundamental view-management model in…

Table View Controller – A controller that manages a table view.

Collection View Controller – A controller that manages a collection view.

Navigation Controller – A controller that manages navigation through a hierarchy…

Tab Bar Controller – A controller that manages a set of



Class: UIViewController

Note

… second, set its class.
This is almost always a class that you create using File > New > File …
Don't forget that it has to be a subclass of `UIViewController`.

It is a VERY common mistake to forget this step!
If you do, you'll wonder why you can't hook up any outlets or actions inside this MVC!

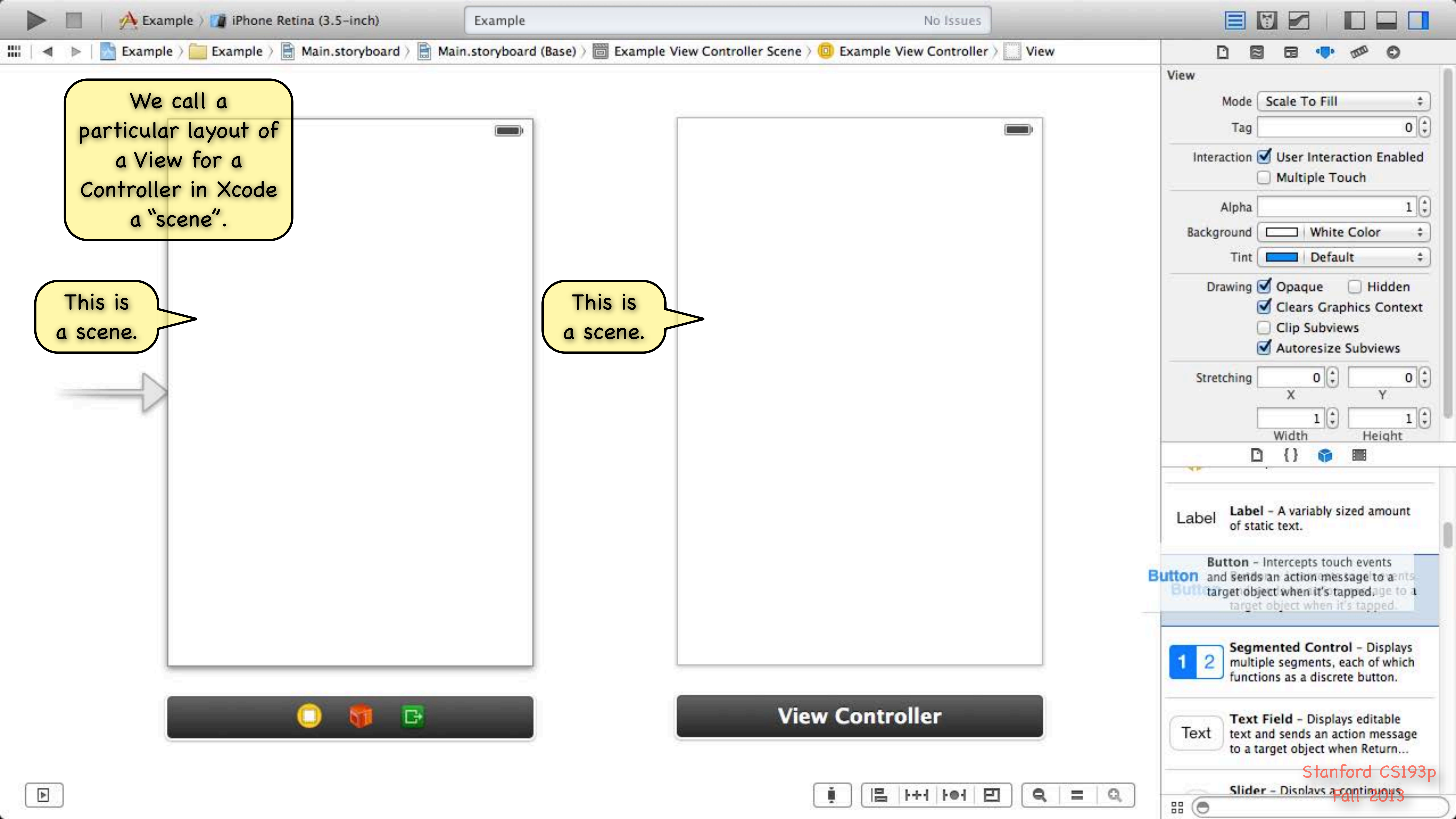
View Controller - A controller that supports the fundamental view-management model in…

Table View Controller - A controller that manages a table view.

Collection View Controller - A controller that manages a collection view.

Navigation Controller - A controller that manages navigation through a hierarchy…

Tab Bar Controller - A controller that manages a set of



We call a particular layout of a View for a Controller in Xcode a "scene".

This is a scene.

This is a scene.

View

Mode: Scale To Fill

Tag: 0

Interaction: ☑ User Interaction Enabled

□ Multiple Touch

Alpha: 1

Background: White Color

Tint: Default

Drawing: ☑ Opaque □ Hidden

☑ Clears Graphics Context

□ Clip Subviews

☑ Autoresize Subviews

Stretching: X 0, Y 0, Width 1, Height 1

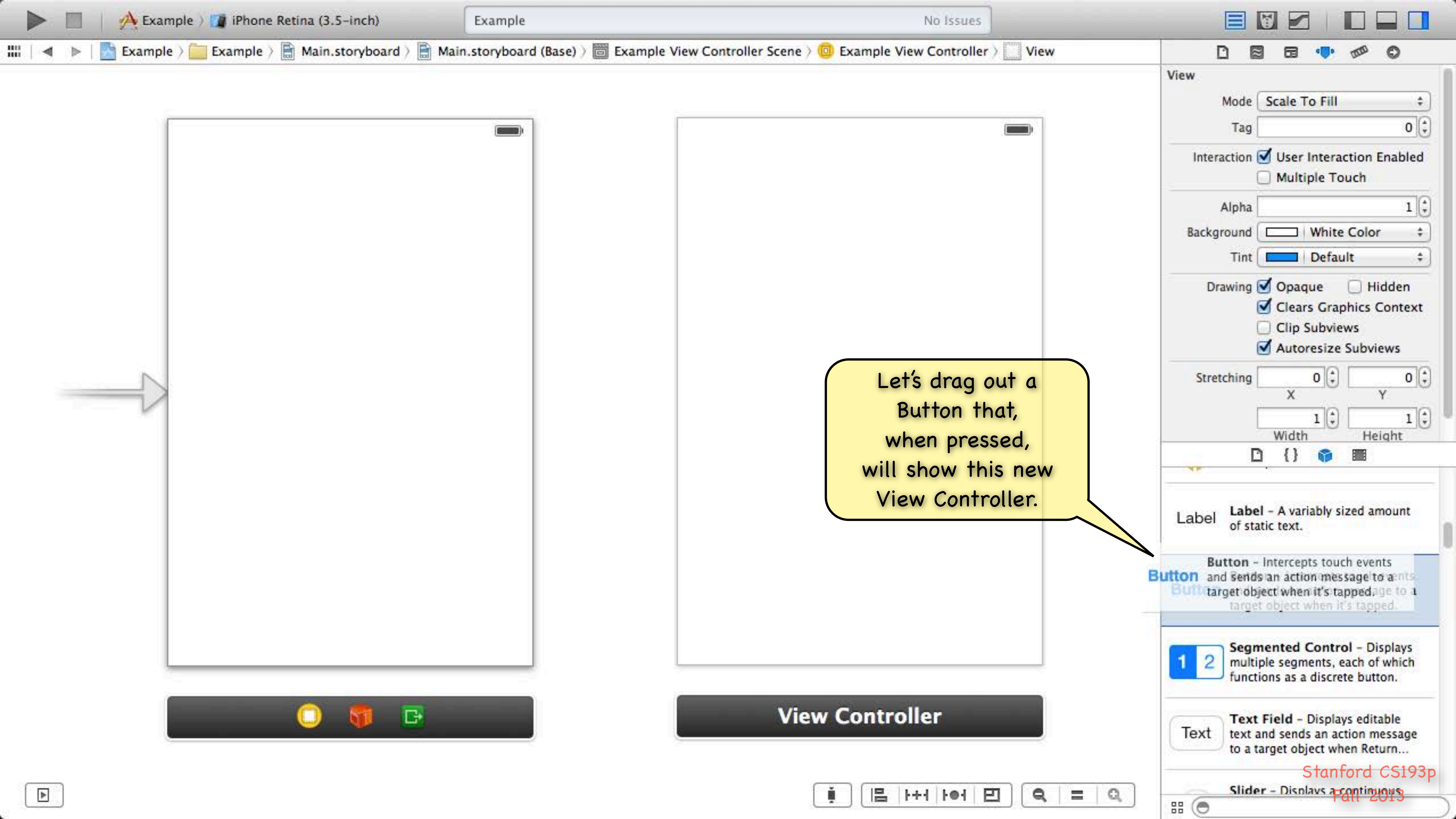
Label – A variably sized amount of static text.

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

Segmented Control – Displays multiple segments, each of which functions as a discrete button.

Text Field – Displays editable text and sends an action message to a target object when Return...

Slider – Displays a continuous


Example
iPhone Retina (3.5-inch)
Example
No Issues
Example 〉 Example 〉 Main.storyboard 〉 Main.storyboard (Base) 〉 Example View Controller Scene 〉 Example View Controller 〉 View
View Controller
Label
Button
1 2
Text

View

Mode: Scale To Fill

Tag: 0

Interaction: ☑ User Interaction Enabled

☐ Multiple Touch

Alpha: 1

Background: White Color

Tint: Default

Drawing: ☑ Opaque ☐ Hidden

☑ Clears Graphics Context

☐ Clip Subviews

☑ Autoresize Subviews

Stretching: 0 (X) 0 (Y)

1 (Width) 1 (Height)

Label – A variably sized amount of static text.

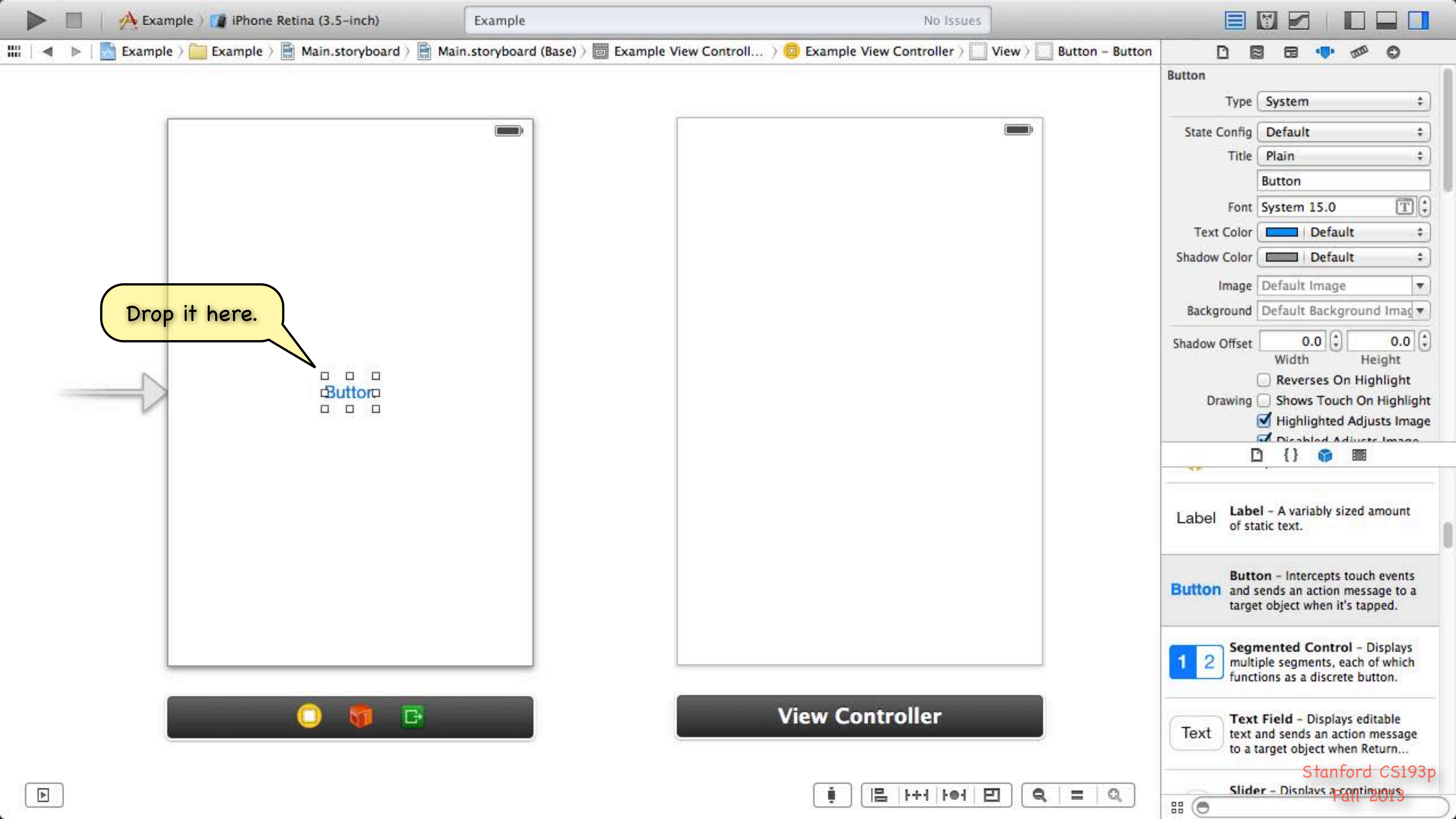
Button – Intercepts touch events and sends an action message to a target object when it's tapped.

Segmented Control – Displays multiple segments, each of which functions as a discrete button.

Text Field – Displays editable text and sends an action message to a target object when Return...

Slider – Displays a continuous...

Let's drag out a Button that, when pressed, will show this new View Controller.



Drop it here.

Button

Type: System

State Config: Default

Title: Plain

Button

Font: System 15.0

Text Color: Default

Shadow Color: Default

Image: Default Image

Background: Default Background Imag

Shadow Offset: 0.0 (Width), 0.0 (Height)

Drawing:
- ☐ Reverses On Highlight
- ☐ Shows Touch On Highlight
- ☑ Highlighted Adjusts Image
- ☑ Disabled Adjusts Image

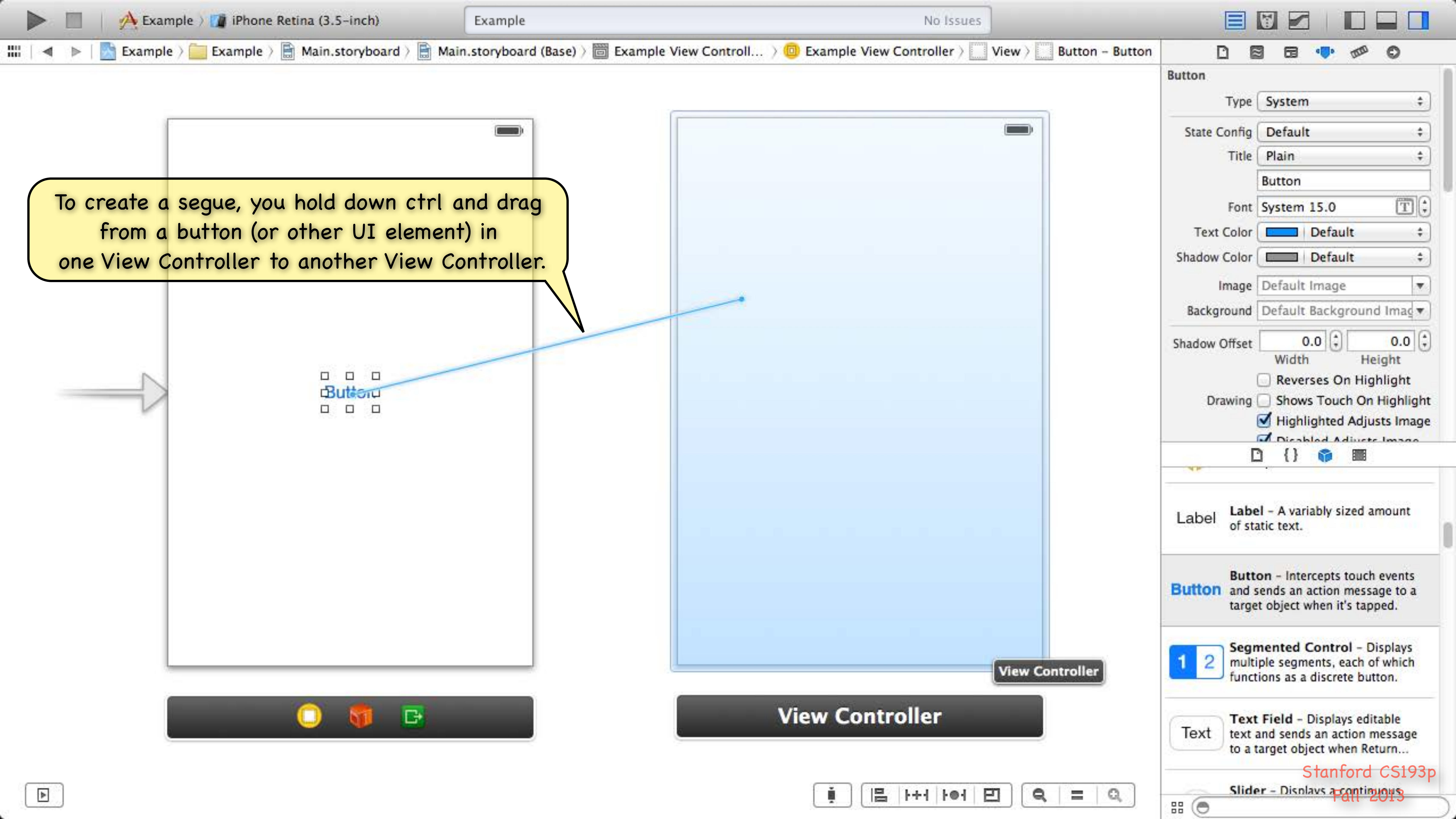
Label – A variably sized amount of static text.

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

Segmented Control – Displays multiple segments, each of which functions as a discrete button.

Text Field – Displays editable text and sends an action message to a target object when Return...

Slider – Displays a continuous



To create a segue, you hold down ctrl and drag from a button (or other UI element) in one View Controller to another View Controller.

Button

Type: System
State Config: Default
Title: Plain
Button
Font: System 15.0
Text Color: Default
Shadow Color: Default
Image: Default Image
Background: Default Background Imag
Shadow Offset: 0.0 Width, 0.0 Height
Drawing: □ Reverses On Highlight
□ Shows Touch On Highlight
☑ Highlighted Adjusts Image

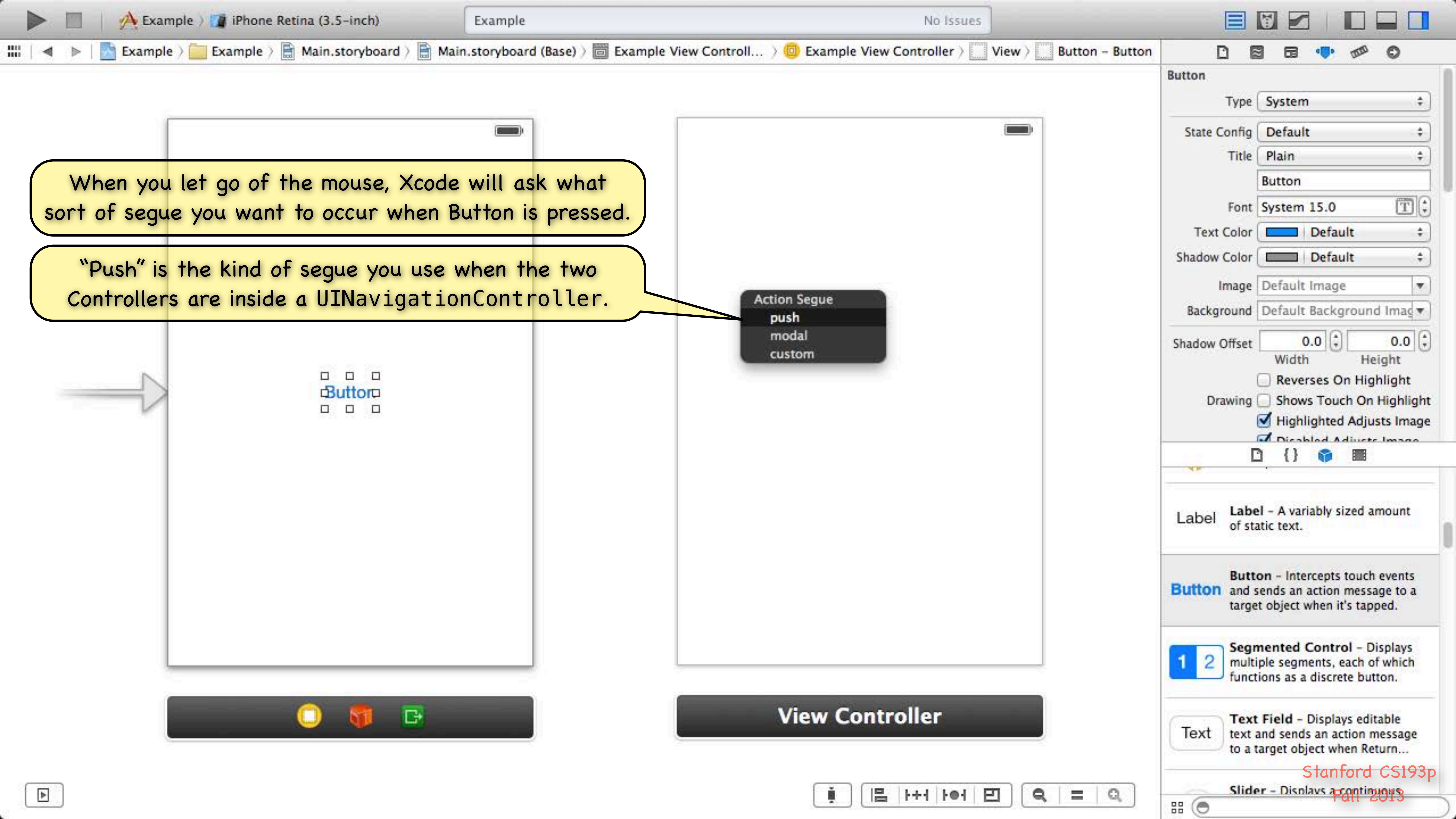
Label – A variably sized amount of static text.

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

Segmented Control – Displays multiple segments, each of which functions as a discrete button.

Text Field – Displays editable text and sends an action message to a target object when Return...

Slider – Displays a continuous



When you let go of the mouse, Xcode will ask what sort of segue you want to occur when Button is pressed.

“Push” is the kind of segue you use when the two Controllers are inside a `UINavigationController`.

Action Segue
- push
- modal
- custom

Button

Type: System
State Config: Default
Title: Plain
Button
Font: System 15.0
Text Color: Default
Shadow Color: Default
Image: Default Image
Background: Default Background Imag
Shadow Offset: 0.0 Width, 0.0 Height
Drawing:
☐ Reverses On Highlight
☐ Shows Touch On Highlight
☑ Highlighted Adjusts Image
☑ Disabled Adjusts Image

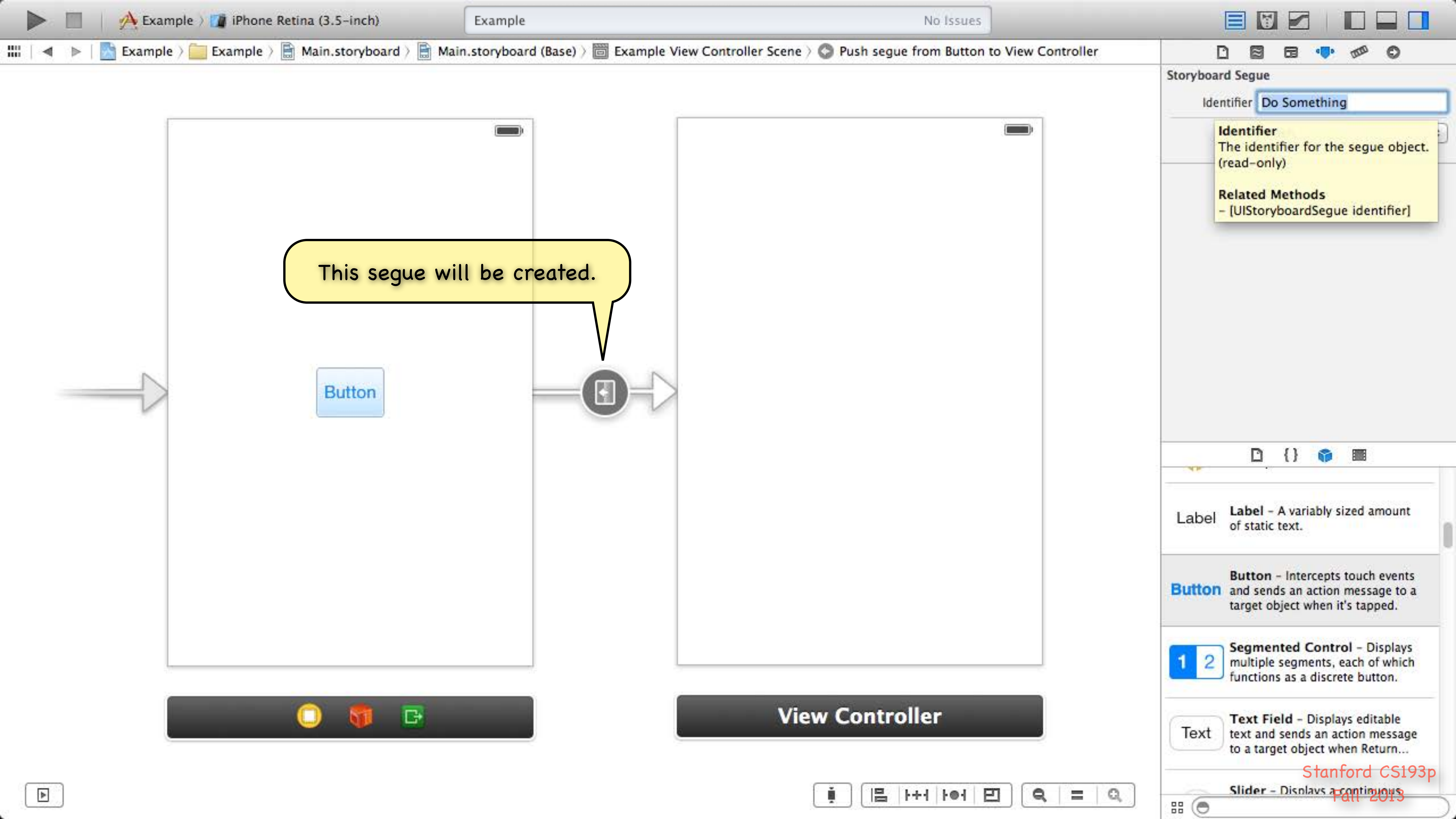
Label – A variably sized amount of static text.

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

Segmented Control – Displays multiple segments, each of which functions as a discrete button.

Text Field – Displays editable text and sends an action message to a target object when Return...

Slider – Displays a continuous



This segue will be created.

Storyboard Segue

Identifier: Do Something

Identifier
The identifier for the segue object. (read-only)

Related Methods
- [UIStoryboardSegue identifier]

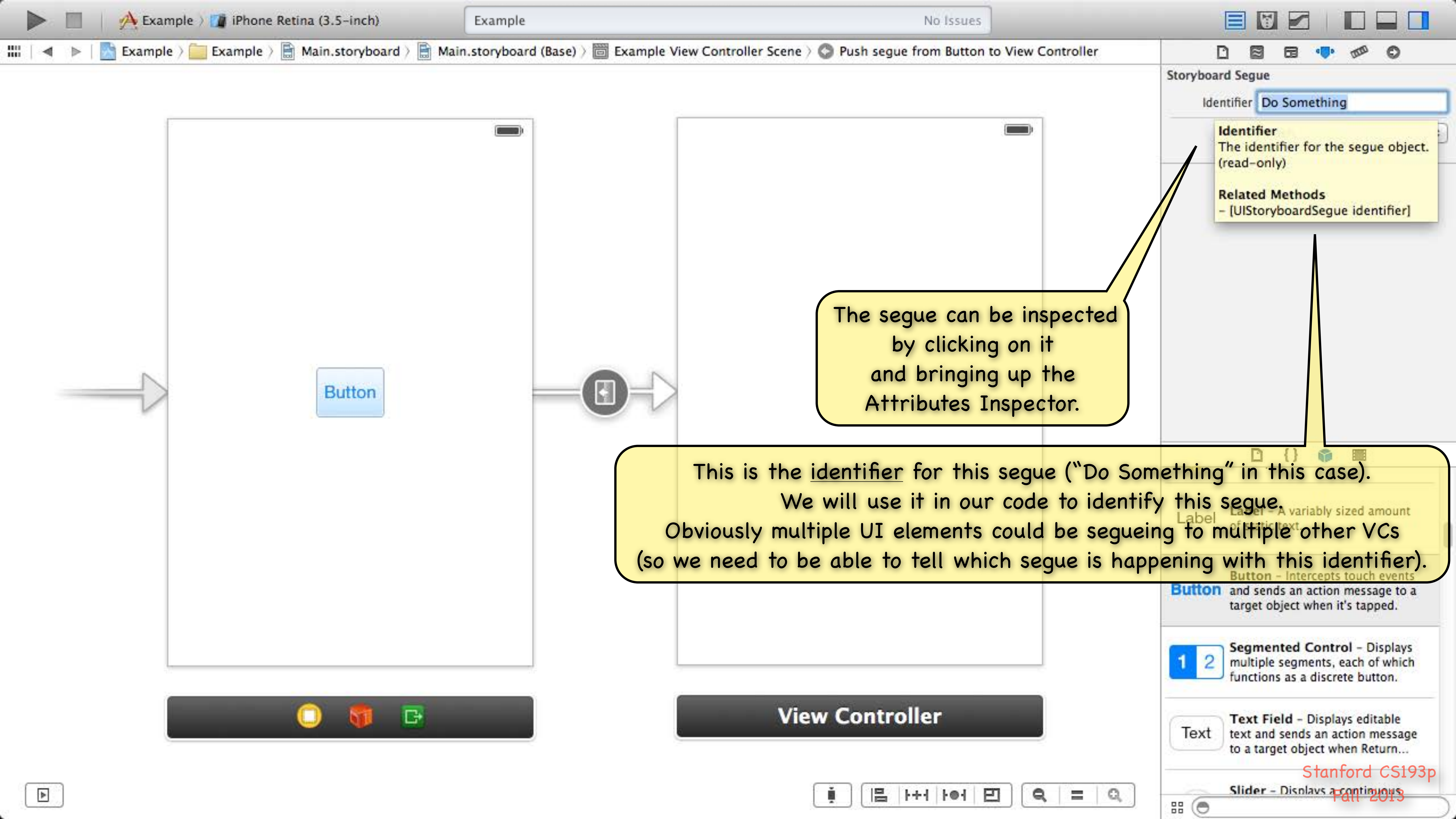
Label - A variably sized amount of static text.

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

Segmented Control - Displays multiple segments, each of which functions as a discrete button.

Text Field - Displays editable text and sends an action message to a target object when Return...

Slider - Displays a continuous

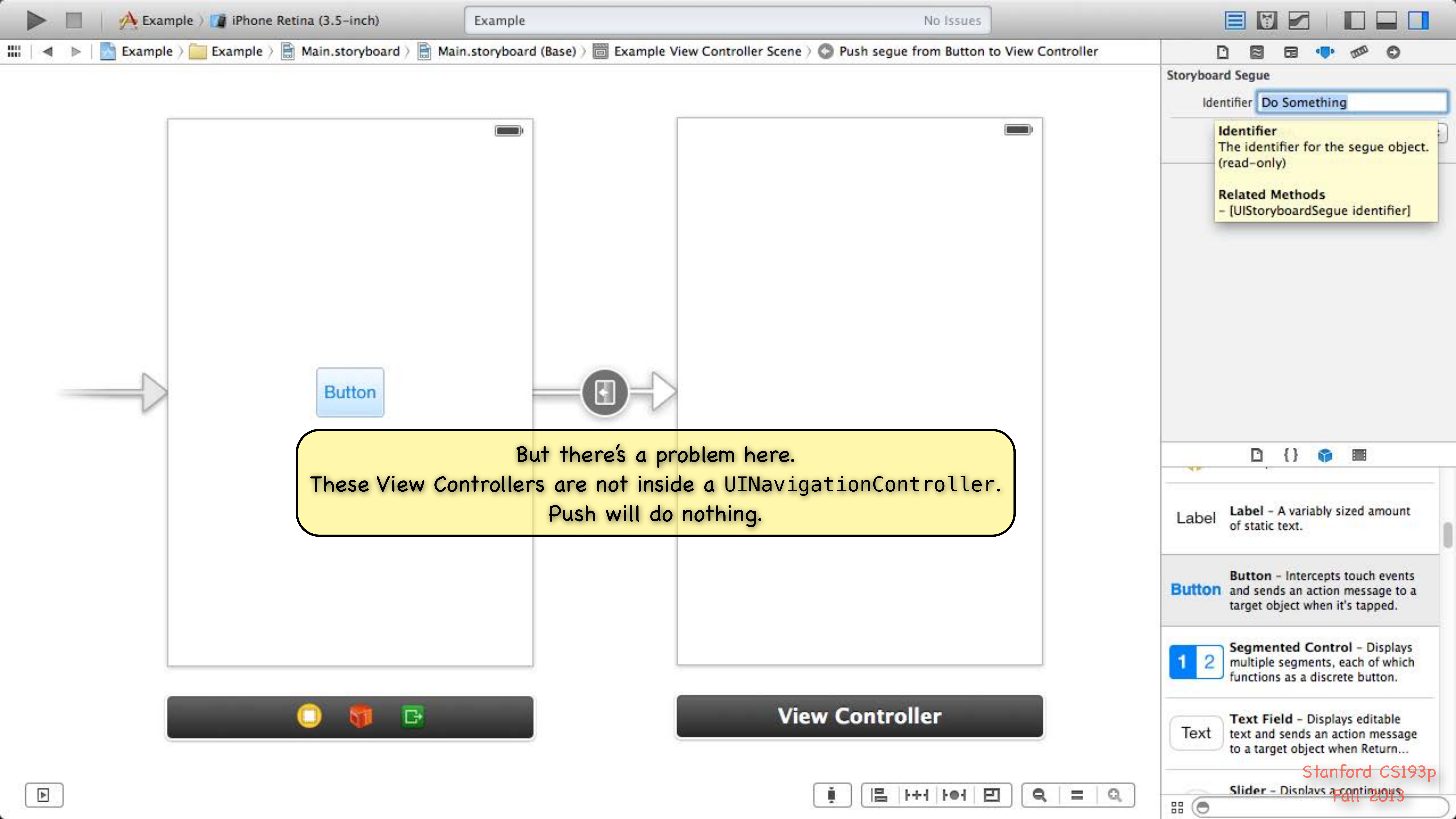

Example
iPhone Retina (3.5-inch)
Example
No Issues
Example
Example
Main.storyboard
Main.storyboard (Base)
Example View Controller Scene
Push segue from Button to View Controller
Button
View Controller
Storyboard Segue
Identifier
Do Something
Label – A variably sized amount of static text.
Button – Intercepts touch events and sends an action message to a target object when it's tapped.
Segmented Control – Displays multiple segments, each of which functions as a discrete button.
Text Field – Displays editable text and sends an action message to a target object when Return...
Slider – Displays a continuous

Identifier
The identifier for the segue object. (read-only)

Related Methods
– [UIStoryboardSegue identifier]

The segue can be inspected by clicking on it and bringing up the Attributes Inspector.

This is the identifier for this segue ("Do Something" in this case).
We will use it in our code to identify this segue.
Obviously multiple UI elements could be segueing to multiple other VCs
(so we need to be able to tell which segue is happening with this identifier).



Identifier
The identifier for the segue object.
(read-only)

Related Methods
- [UIStoryboardSegue identifier]

Label - A variably sized amount of static text.

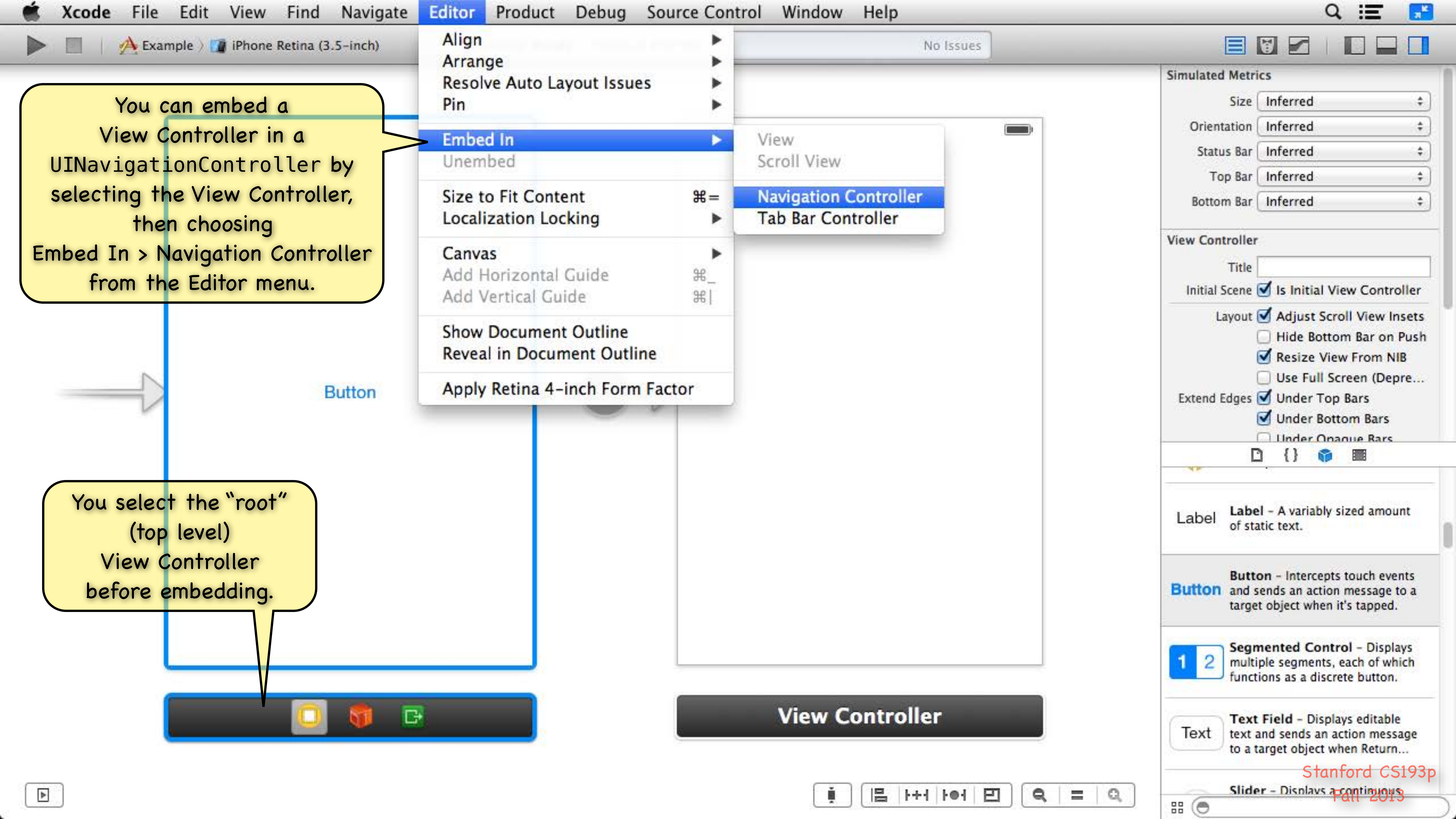
Button - Intercepts touch events and sends an action message to a target object when it's tapped.

Segmented Control - Displays multiple segments, each of which functions as a discrete button.

Text Field - Displays editable text and sends an action message to a target object when Return...

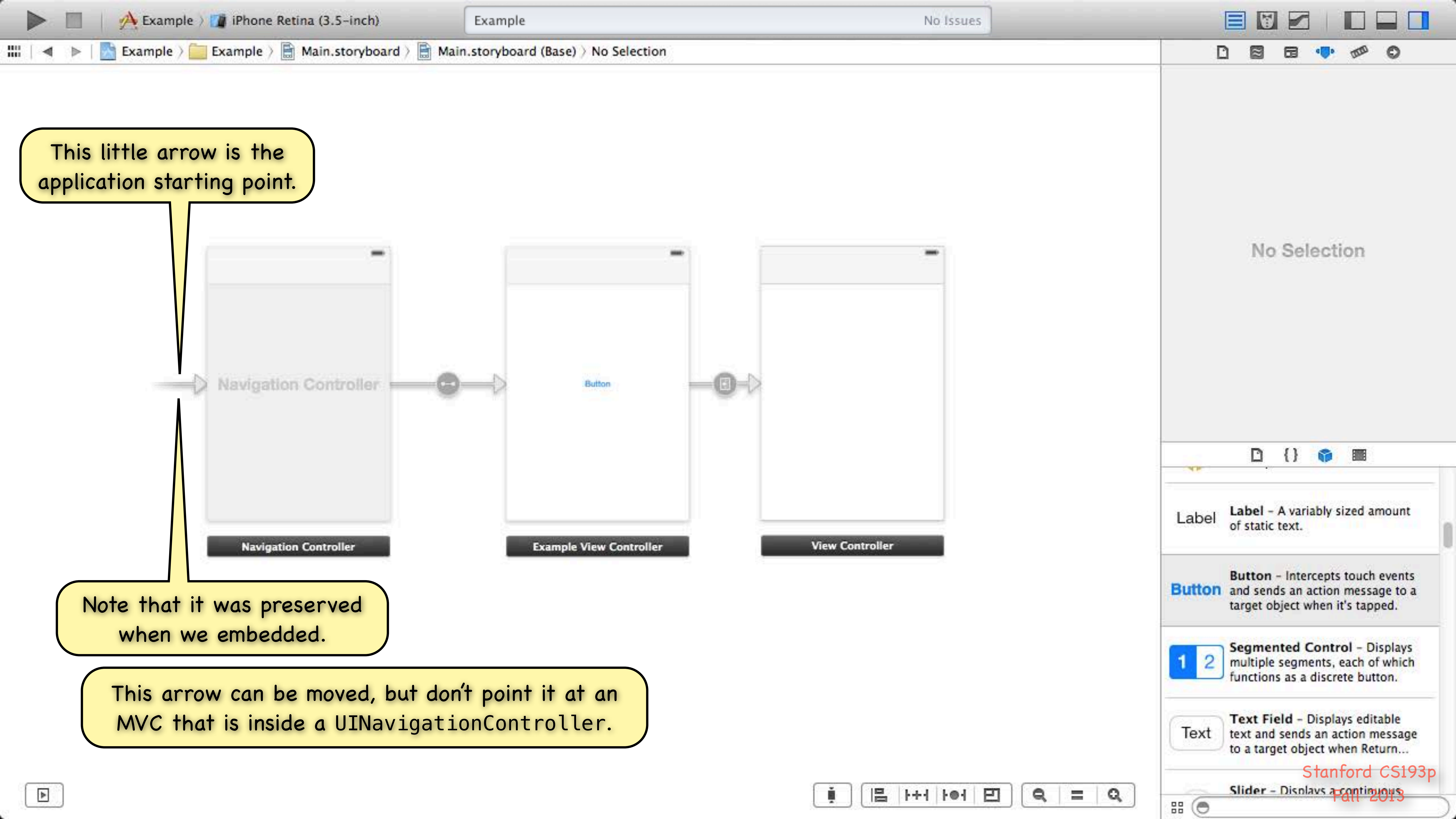
Slider - Displays a continuous

But there's a problem here.
These View Controllers are not inside a `UINavigationController`.
Push will do nothing.


Xcode File Edit View Find Navigate Editor Product Debug Source Control Window Help
Example iPhone Retina (3.5-inch)
No Issues
Align
Arrange
Resolve Auto Layout Issues
Pin
Embed In
Unembed
Size to Fit Content ⌘=
Localization Locking
Canvas
Add Horizontal Guide ⌘_
Add Vertical Guide ⌘|
Show Document Outline
Reveal in Document Outline
Apply Retina 4-inch Form Factor
View
Scroll View
Navigation Controller
Tab Bar Controller
Button
View Controller
Simulated Metrics
Size Inferred
Orientation Inferred
Status Bar Inferred
Top Bar Inferred
Bottom Bar Inferred
View Controller
Title
Initial Scene ☑ Is Initial View Controller
Layout ☑ Adjust Scroll View Insets
□ Hide Bottom Bar on Push
☑ Resize View From NIB
□ Use Full Screen (Depre...
Extend Edges ☑ Under Top Bars
☑ Under Bottom Bars
□ Under Opaque Bars
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 functions as a discrete button.
Text Text Field - Displays editable text and sends an action message to a target object when Return...
Slider - Displays a continuous...

You can embed a View Controller in a `UINavigationController` by selecting the View Controller, then choosing Embed In > Navigation Controller from the Editor menu.

You select the "root" (top level) View Controller before embedding.



This little arrow is the application starting point.

Note that it was preserved when we embedded.

This arrow can be moved, but don't point it at an MVC that is inside a UINavigationController.

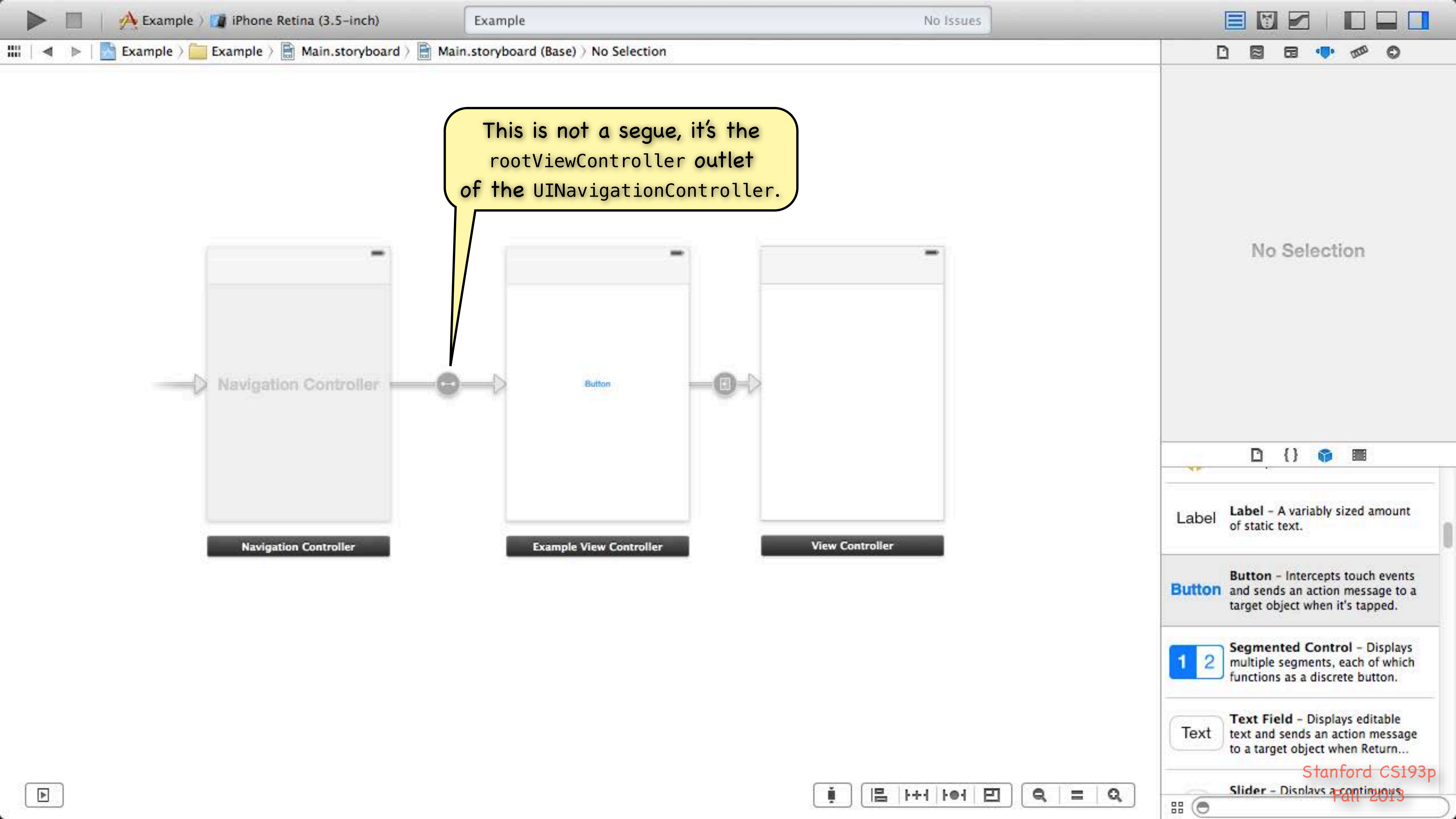
Label – A variably sized amount of static text.

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

Segmented Control – Displays multiple segments, each of which functions as a discrete button.

Text Field – Displays editable text and sends an action message to a target object when Return...

Slider – Displays a continuous



This is not a segue, it's the `rootViewController` outlet of the `UINavigationController`.

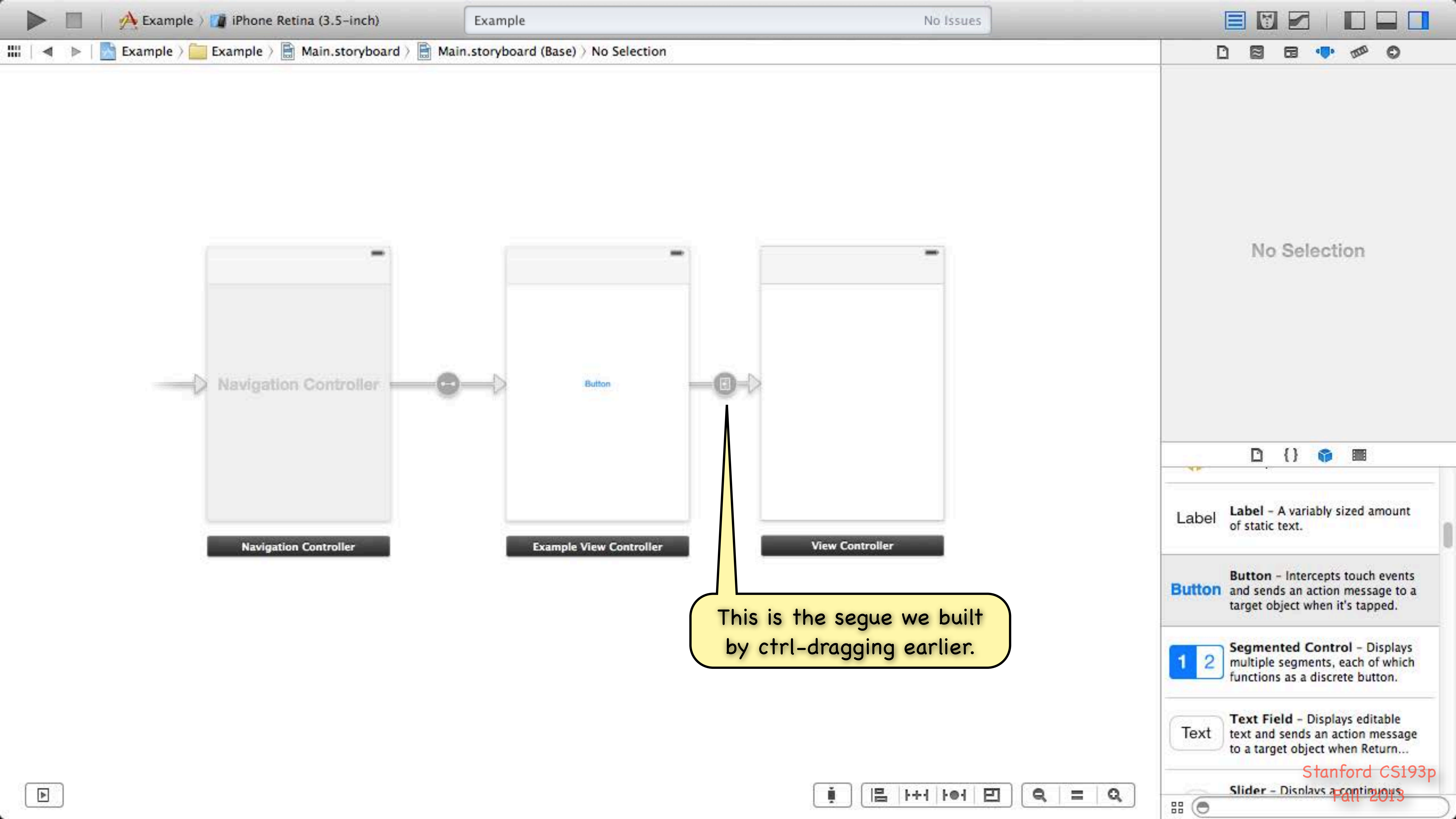
Label - A variably sized amount of static text.

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

Segmented Control - Displays multiple segments, each of which functions as a discrete button.

Text Field - Displays editable text and sends an action message to a target object when Return...

Slider - Displays a continuous



This is the segue we built by ctrl-dragging earlier.

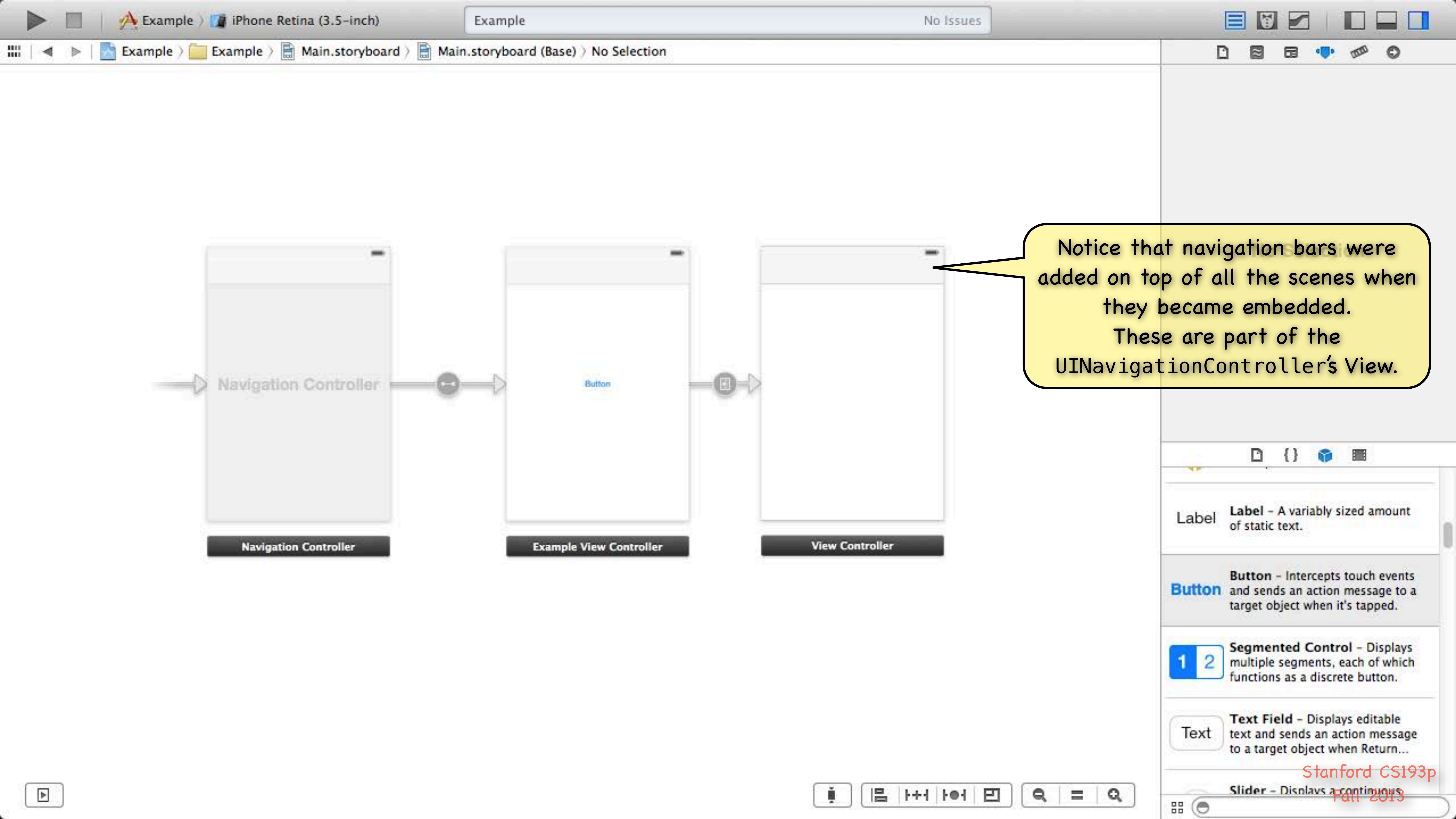
Label — Label - A variably sized amount of static text.

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 functions as a discrete button.

Text — Text Field - Displays editable text and sends an action message to a target object when Return...

Slider - Displays a continuous



Notice that navigation bars were added on top of all the scenes when they became embedded.
These are part of the `UINavigationController`'s View.

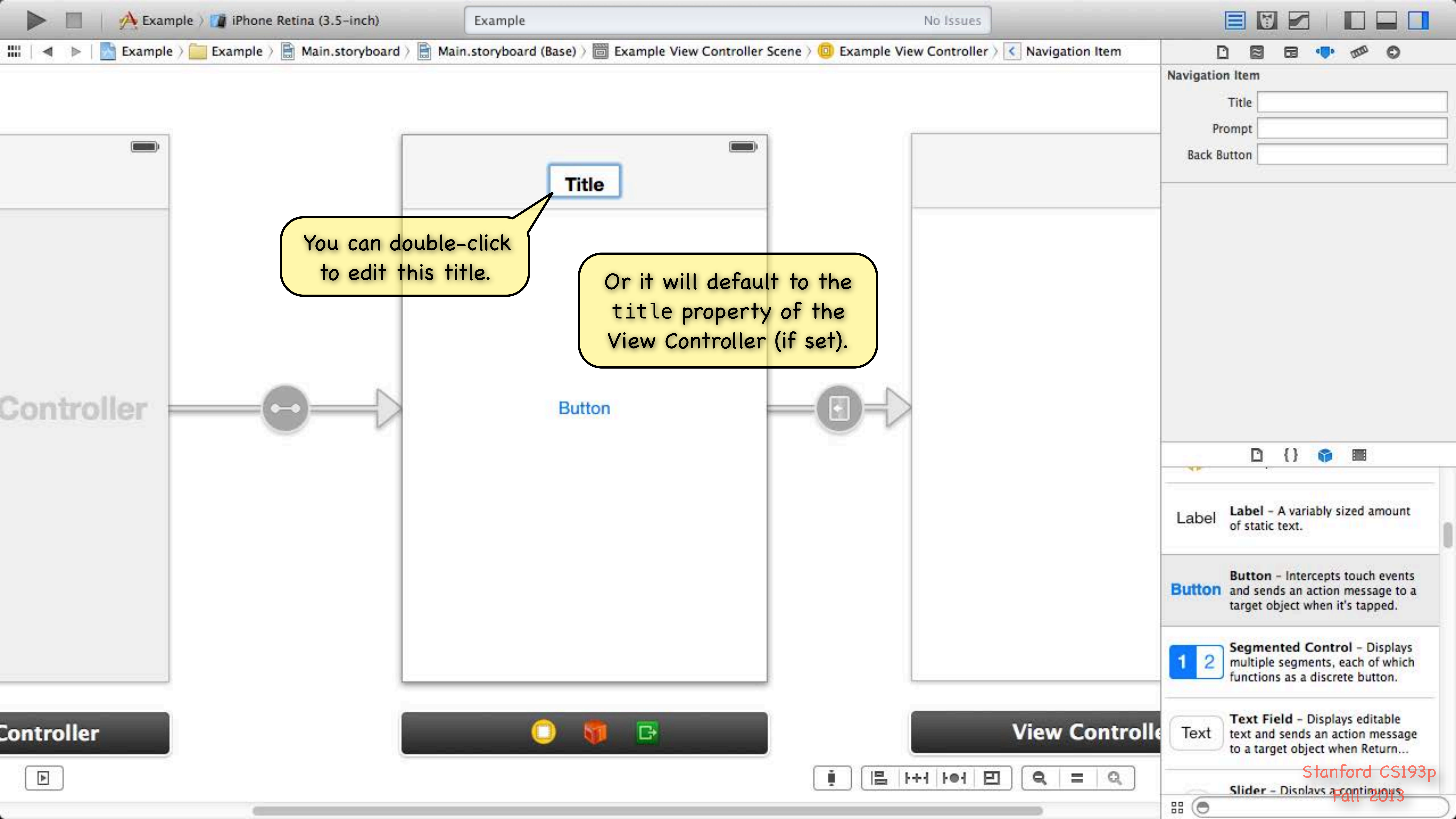
Label – A variably sized amount of static text.

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

Segmented Control – Displays multiple segments, each of which functions as a discrete button.

Text Field – Displays editable text and sends an action message to a target object when Return...

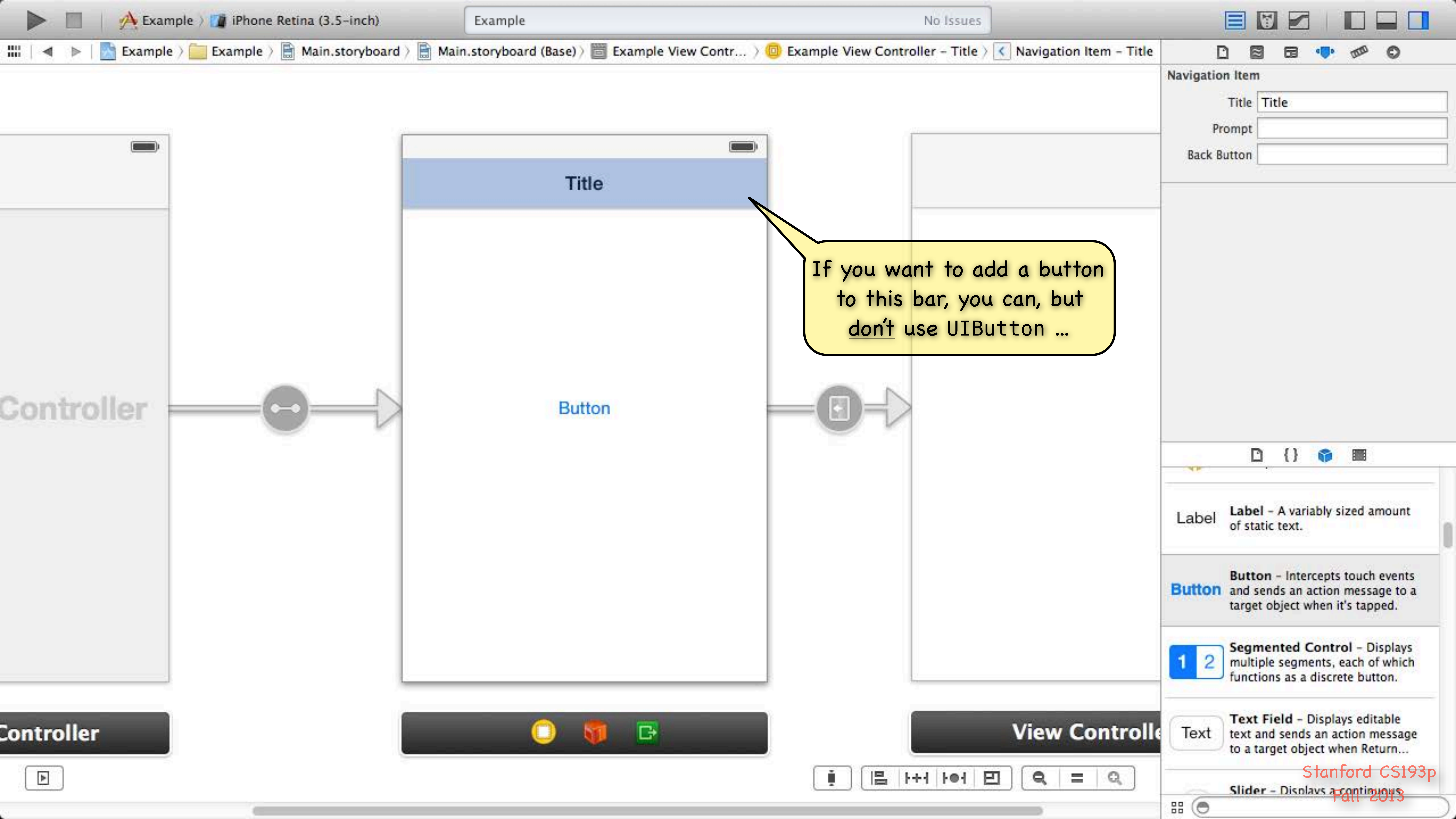
Slider – Displays a continuous



Example
iPhone Retina (3.5-inch)
Example
No Issues
Example 〉 Example 〉 Main.storyboard 〉 Main.storyboard (Base) 〉 Example View Controller Scene 〉 Example View Controller 〉 Navigation Item
Navigation Item
Title
Prompt
Back Button
Controller
Title
Button
Controller
View Controlle
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 functions as a discrete button.
Text
Text Field – Displays editable text and sends an action message to a target object when Return...
Slider – Displays a continuous

You can double-click to edit this title.

Or it will default to the title property of the View Controller (if set).



Navigation Item

Title: Title

Prompt:

Back Button:

Label – A variably sized amount of static text.

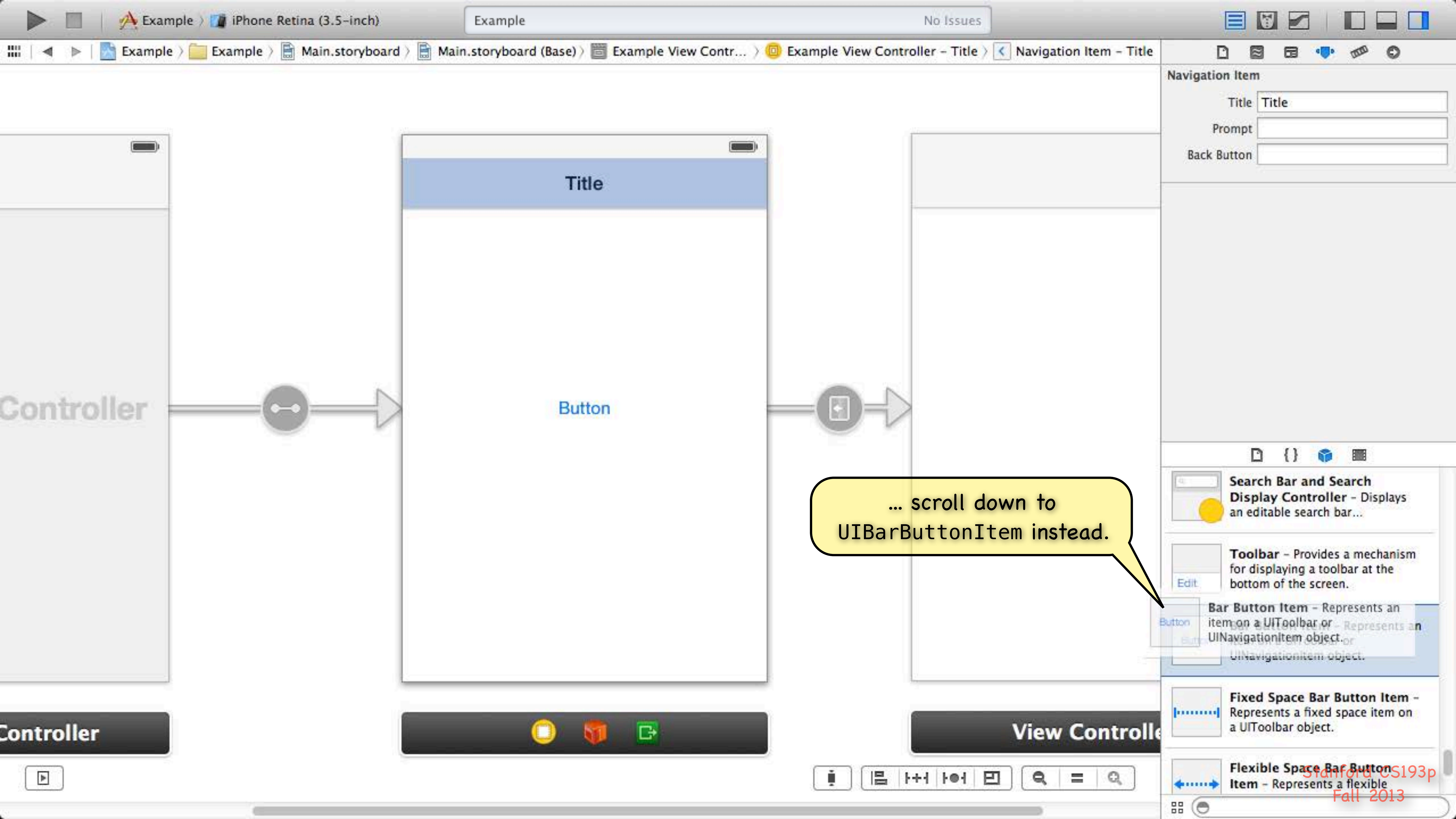
Button – Intercepts touch events and sends an action message to a target object when it's tapped.

Segmented Control – Displays multiple segments, each of which functions as a discrete button.

Text Field – Displays editable text and sends an action message to a target object when Return...

Slider – Displays a continuous

If you want to add a button to this bar, you can, but <u>don't</u> use `UIButton` ...



Navigation Item

Title: Title

Prompt:

Back Button:

Search Bar and Search Display Controller – Displays an editable search bar...

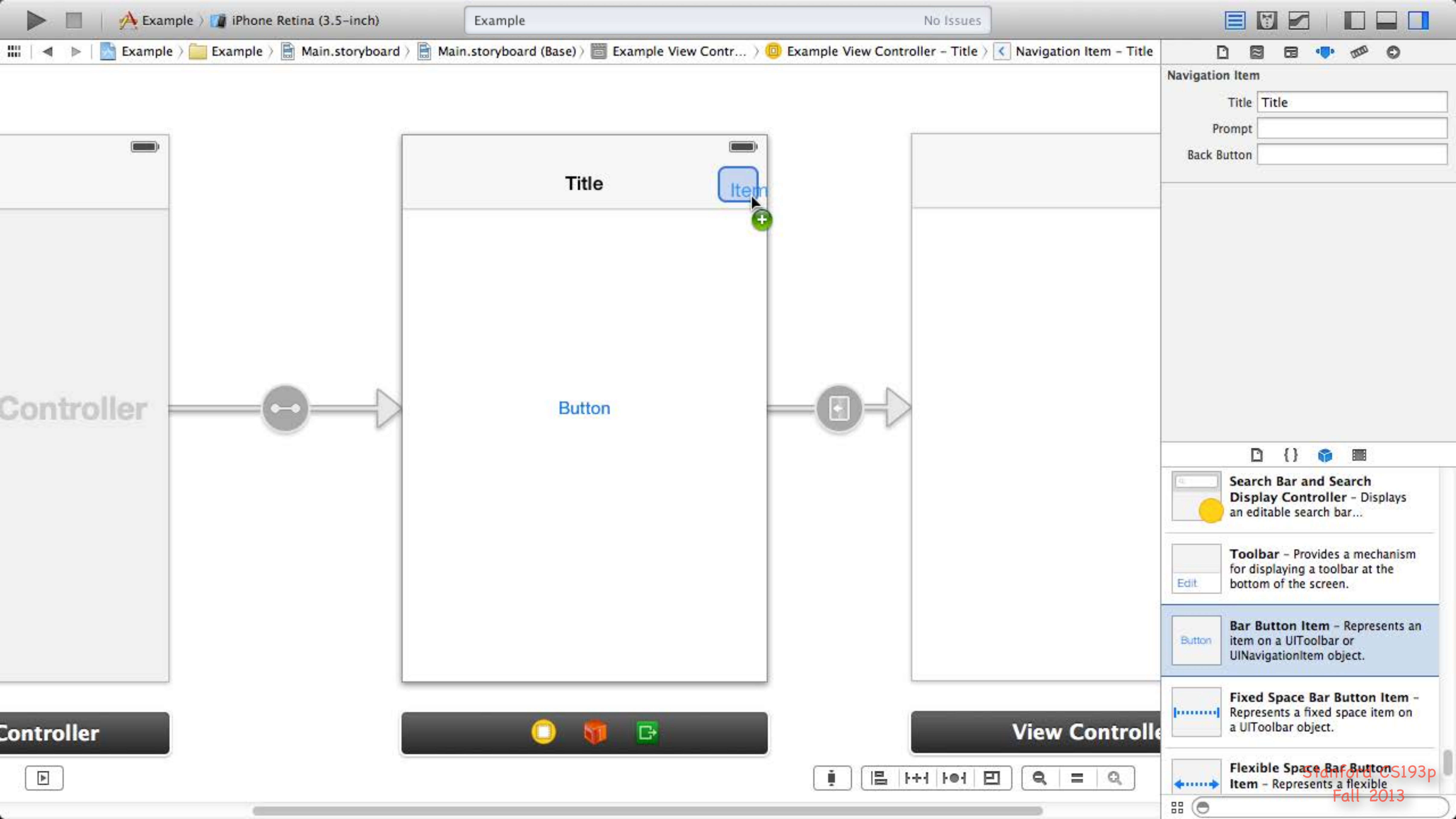
Toolbar – Provides a mechanism for displaying a toolbar at the bottom of the screen.

Bar Button Item – Represents an item on a UIToolbar or UINavigationItem object.

Fixed Space Bar Button Item – Represents a fixed space item on a UIToolbar object.

Flexible Space Bar Button Item – Represents a flexible

... scroll down to `UIBarButtonItem` instead.



Navigation Item

Title: Title

Prompt:

Back Button:

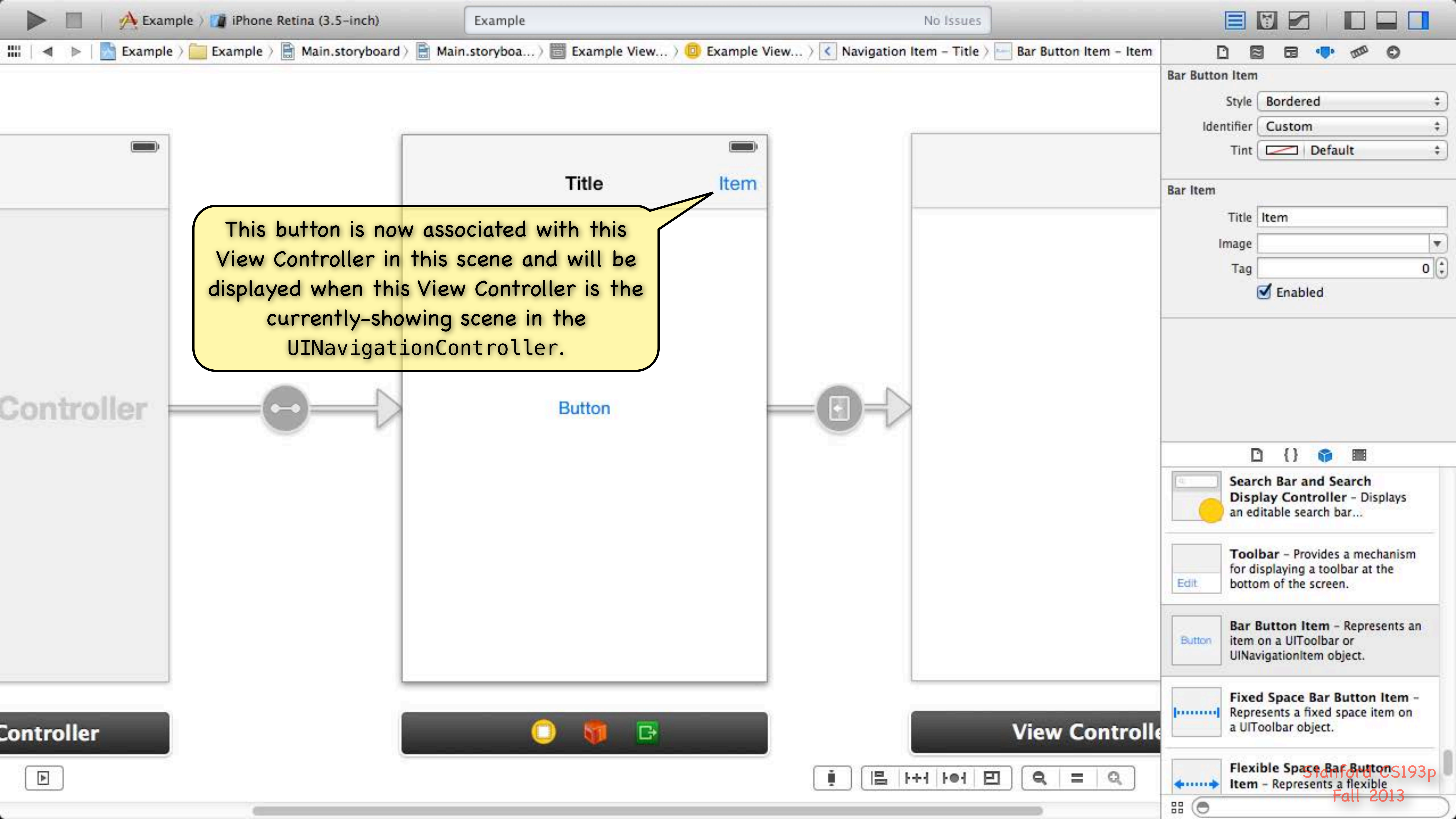
Search Bar and Search Display Controller – Displays an editable search bar...

Toolbar – Provides a mechanism for displaying a toolbar at the bottom of the screen.

Bar Button Item – Represents an item on a UIToolbar or UINavigationItem object.

Fixed Space Bar Button Item – Represents a fixed space item on a UIToolbar object.

Flexible Space Bar Button Item – Represents a flexible



This button is now associated with this View Controller in this scene and will be displayed when this View Controller is the currently-showing scene in the UINavigationController.

Bar Button Item

Style: Bordered

Identifier: Custom

Tint: Default

Bar Item

Title: Item

Image:

Tag: 0

☑ Enabled

Search Bar and Search Display Controller - Displays an editable search bar...

Toolbar - Provides a mechanism for displaying a toolbar at the bottom of the screen.

Bar Button Item - Represents an item on a UIToolbar or UINavigationItem object.

Fixed Space Bar Button Item - Represents a fixed space item on a UIToolbar object.

Flexible Space Bar Button Item - Represents a flexible

UINavigationController

- When does a pushed MVC pop off?

Usually because the user presses the "back" button (shown on the previous slide).
But it can happen programmatically as well with this UINavigationController instance method

```
- (void)popViewControllerAnimated:(BOOL)animated;
```

This does the same thing as clicking the back button.
Somewhat rare to call this method. Usually we want the user in control of navigating the stack.
But you might do it if some action the user takes in a view makes it irrelevant to be on screen.

- Example

Let's say we push an MVC which displays a database record and has a delete button w/this action:

```
- (IBAction)deleteCurrentRecord:(UIButton *)sender
{
    // delete the record we are displaying
    // we just deleted the record we are displaying!
    // so it does not make sense to be on screen anymore, so pop
    [self.navigationController popViewControllerAnimated:YES];
}
```

Notice that all UIViewControllers know the UINavigationController they are in.
This is nil if they are not in one.

View Controller

- Other kinds of segues besides Push

  Replace - Replaces the right-hand side of a `UISplitViewController` (iPad only)

  Popover - Puts the view controller on the screen in a popover (iPad only)

  Modal - Puts the view controller up in a way that blocks the app until it is dismissed

  Custom - You can create your own subclasses of `UIStoryboardSegue`

- We'll talk about iPad-related segues in future lectures

  Replace & Popover

- We'll talk about Modal segues later in the quarter too

  People often use Modal UIs as a crutch, so we don't want to go to that too early.

View Controller

- Firing off a segue from code

Sometimes it makes sense to segue directly when a button is touched, but not always.
For example, what if you want to conditionally segue?
You can programmatically invoke segues using this method in UIViewController:

```
- (void)performSegueWithIdentifier:(NSString *)segueId sender:(id)sender;
```

The segueId is set in the attributes inspector in Xcode (seen on previous slide).
The sender is the initiator of the segue (a UIButton or yourself (UIViewController) usually).

```
- (IBAction)rentEquipment
{
    if (self.snowTraversingTalent == Skiing) {
        [self performSegueWithIdentifier:@“AskAboutSkis” sender:self];
    } else {
        [self performSegueWithIdentifier:@“AskAboutSnowboard” sender:self];
    }
}
```

Segues

- When a segue happens, what goes on in my code?

The segue offers the source VC the opportunity to “prepare” the new VC to come on screen.

This method is sent to the VC that contains the button that initiated the segue:

```
- (void)prepareForSegue:(UIStoryboardSegue *)segue sender:(id)sender
{
    if ([segue.identifier isEqualToString:@“DoSomething”]) {
        if ([segue.destinationViewController isKindOfClass:[DoSomethingVC class]]) {
            DoSomethingVC *doVC = (DoSomethingVC *)segue.destinationViewController;
            doVC.neededInfo = …;
        }
    }
}
```

You should pass data the new VC needs here and “let it run.”

Think of the new VC as part of the View of the Controller that initiates the segue.

It must play by the same rules as a View.

For example, it should not talk back to you (except through blind communication like delegation).

Segues

- You can prevent a segue from happening

Your Controller usually just always segues.

But if you respond NO to this method, it would prevent the identified segue from happening.

```
- (BOOL)shouldPerformSegueWithIdentifier:(NSString *)identifier sender:(id)sender
{
    if ([segue.identifier isEqualToString:@“DoAParticularThing”]) {
        return [self canDoAParticularThing] ? YES : NO;
    }
}
```

Do not create “dead UI” with this (e.g. buttons that do nothing).

This is a very rare method to ever implement.

Unwinding

- There are also ways to unwind from a series of segues

  Some people think of this as "reverse segueing".

  Used if you want to dismiss the VC you are in and go back to a previous VC that segued to you.

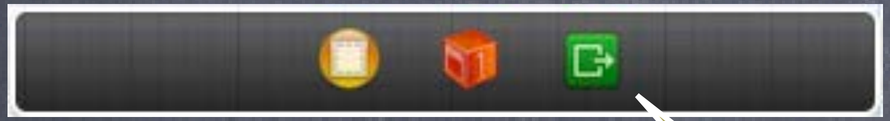
  For example, what if you wanted to pop back multiple levels in a navigation controller?

  (if you were only going back one level, you could just use `popViewControllerAnimated:`).

  The little green button in the black bar at the bottom of a scene can be used to wire that up.

  We will probably cover this when we talk about the Modal segue type (i.e. later).

  You need to master segueing forward before you start thinking about going backward!



This is the "little green button."

View Controller

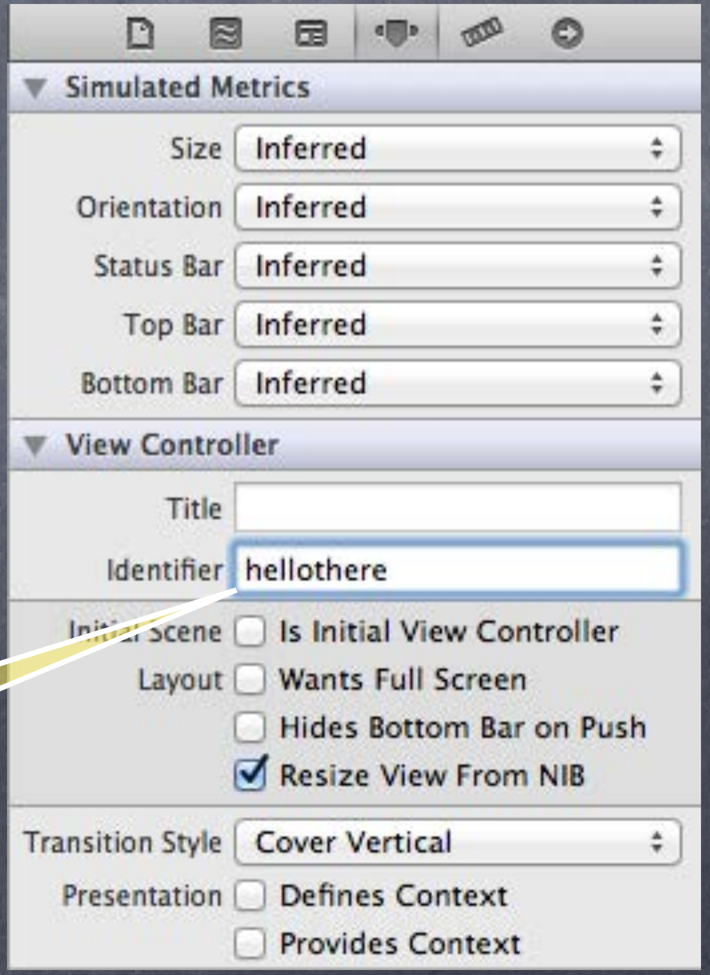
- Instantiating a UIViewController by name from a storyboard

  Sometimes (very rarely) you might want to put a VC on screen yourself (i.e., not use a segue).

```
NSString *vcid = @“something”;
UIViewController *controller = [storyboard instantiateViewControllerWithIdentifier:vcid];
```

  Usually you get the storyboard above from self.storyboard in an existing UIViewController.

  The identifier vcid must match a string you set in Xcode to identify a UIViewController there.



▼ Simulated Metrics

Size: Inferred
Orientation: Inferred
Status Bar: Inferred
Top Bar: Inferred
Bottom Bar: Inferred

▼ View Controller

Title:
Identifier: hellothere
Initial Scene: □ Is Initial View Controller
Layout: □ Wants Full Screen
□ Hides Bottom Bar on Push
☑ Resize View From NIB
Transition Style: Cover Vertical
Presentation: □ Defines Context
□ Provides Context

This UIViewController in the storyboard can be instantiated using the identifier “hellothere”.

View Controller

- Instantiating a UIViewController by name from a storyboard

  Sometimes (very rarely) you might want to put a VC on screen yourself (i.e., not use a segue).

  ```
  NSString *vcid = @"something";
  UIViewController *controller = [storyboard instantiateViewControllerWithIdentifier:vcid];
  ```

  Usually you get the storyboard above from self.storyboard in an existing UIViewController.

  The identifier vcid must match a string you set in Xcode to identify a UIViewController there.

- Example: creating a UIViewController in a target/action method

  Lay out the View for a DoitViewController in your storyboard and name it "doit1".

  ```
  - (IBAction)doit
  {
      DoitViewController *doit =
          [self.storyboard instantiateViewControllerWithIdentifier:@"doit1"];
      doit.infoDoitNeeds = self.info;
      [self.navigationController pushViewController:doit animated:YES];
  }
  ```

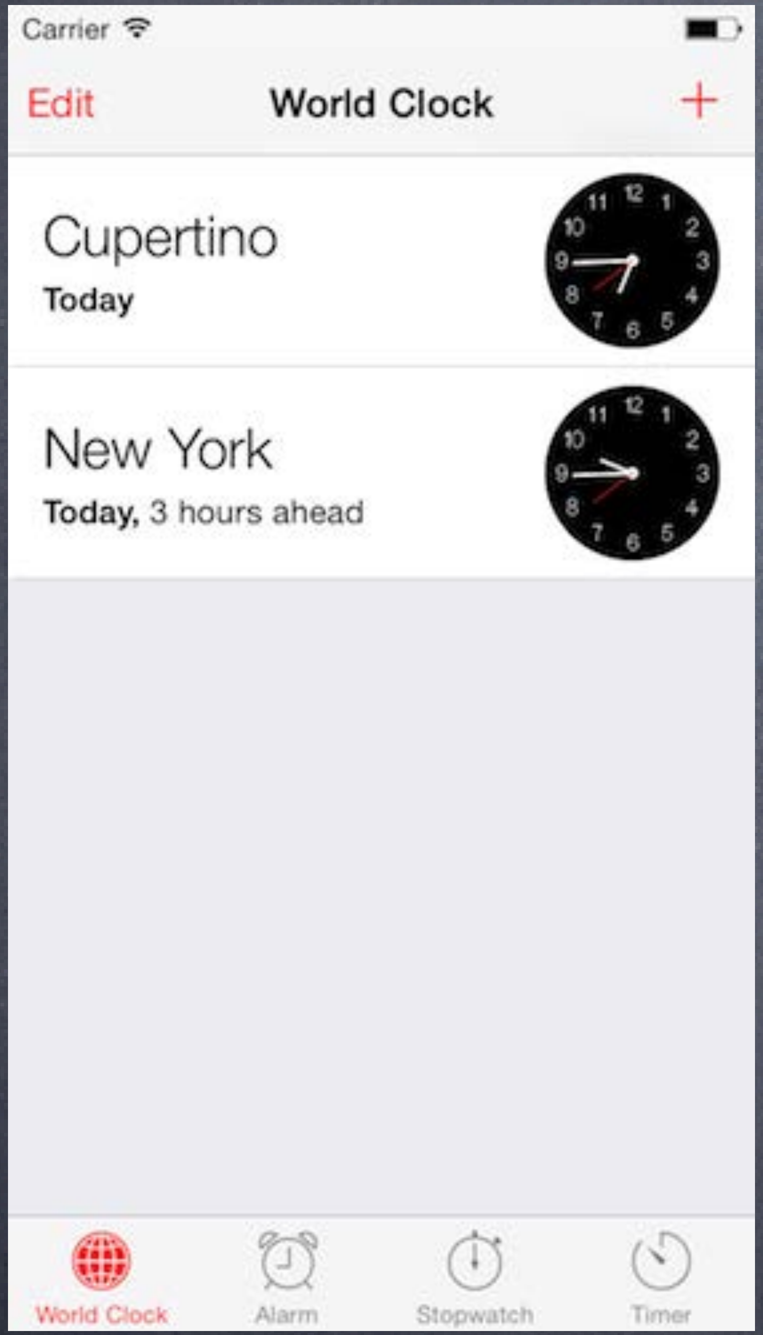
  Note use of self.navigationController again.

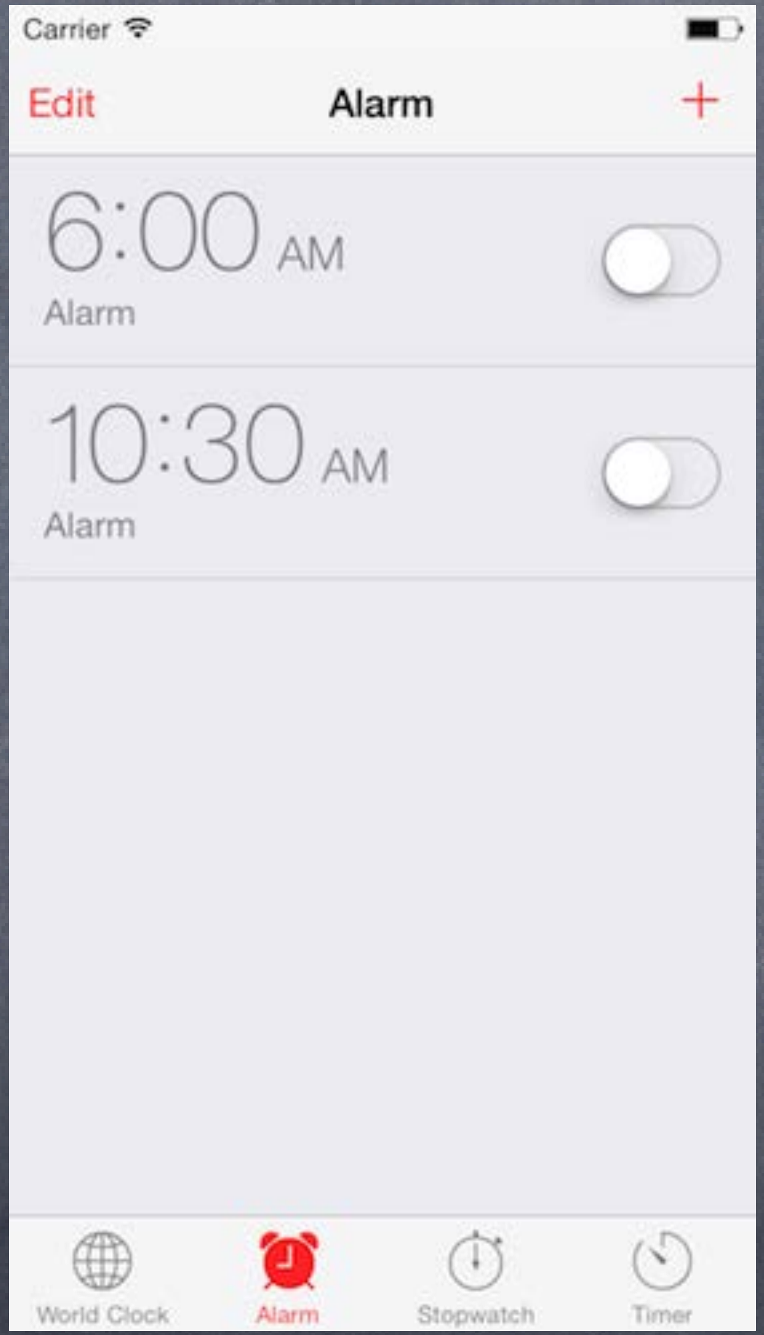
Demo

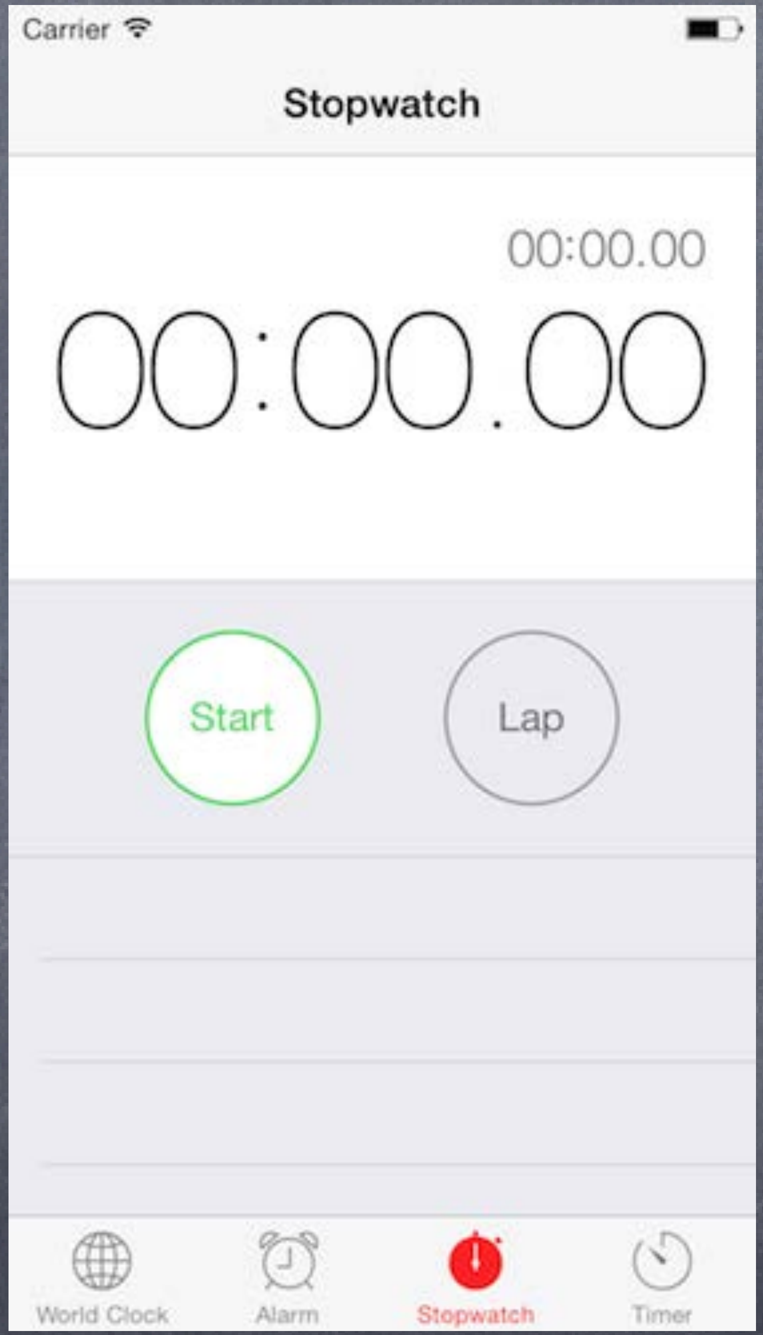
- Attributor Stats

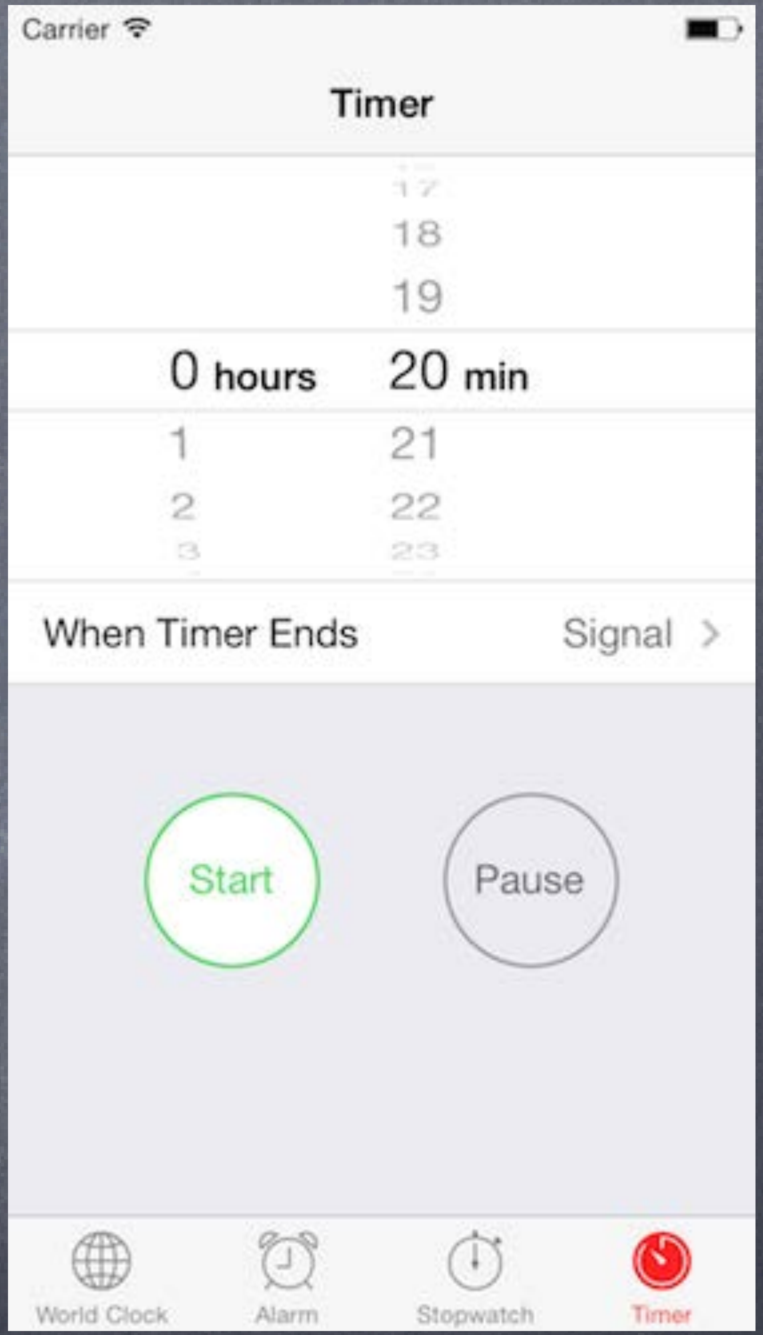
  Use a UINavigationController to show "statistics" on colors and outlining in Attributor.

UITabBarController

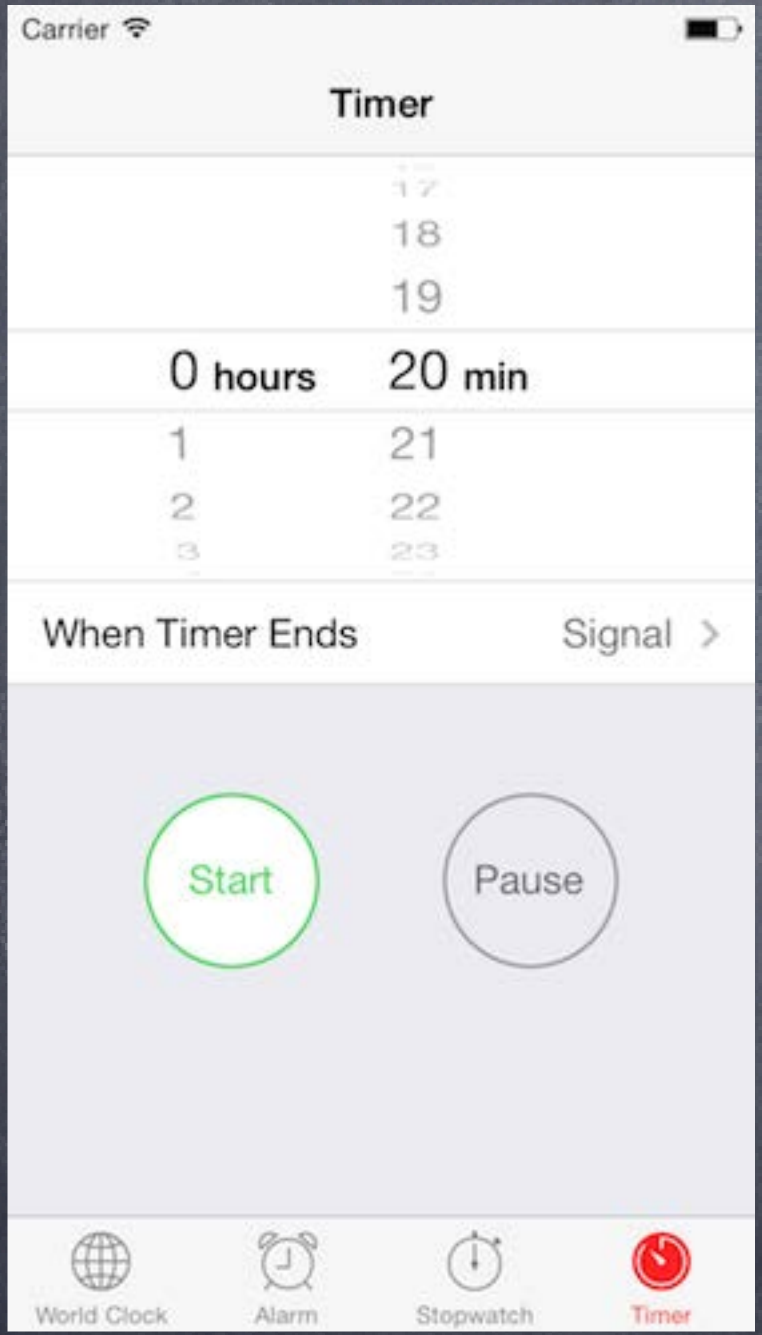


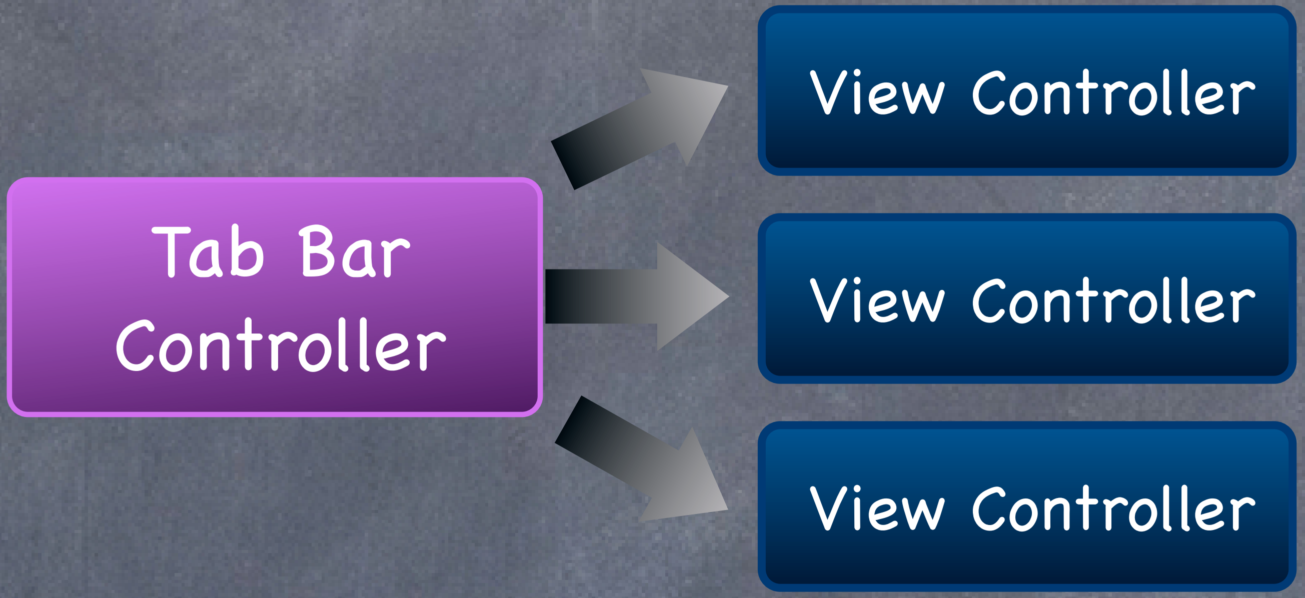






UITabBarController

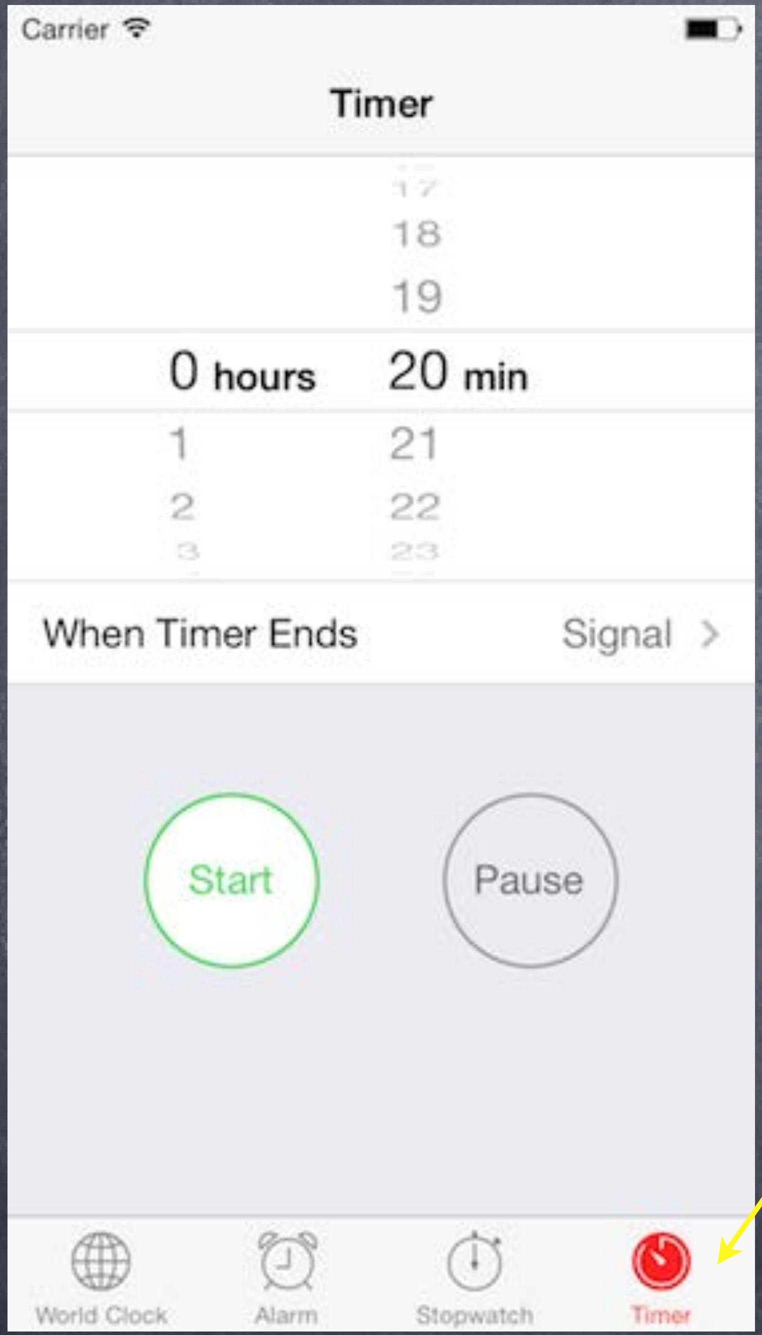


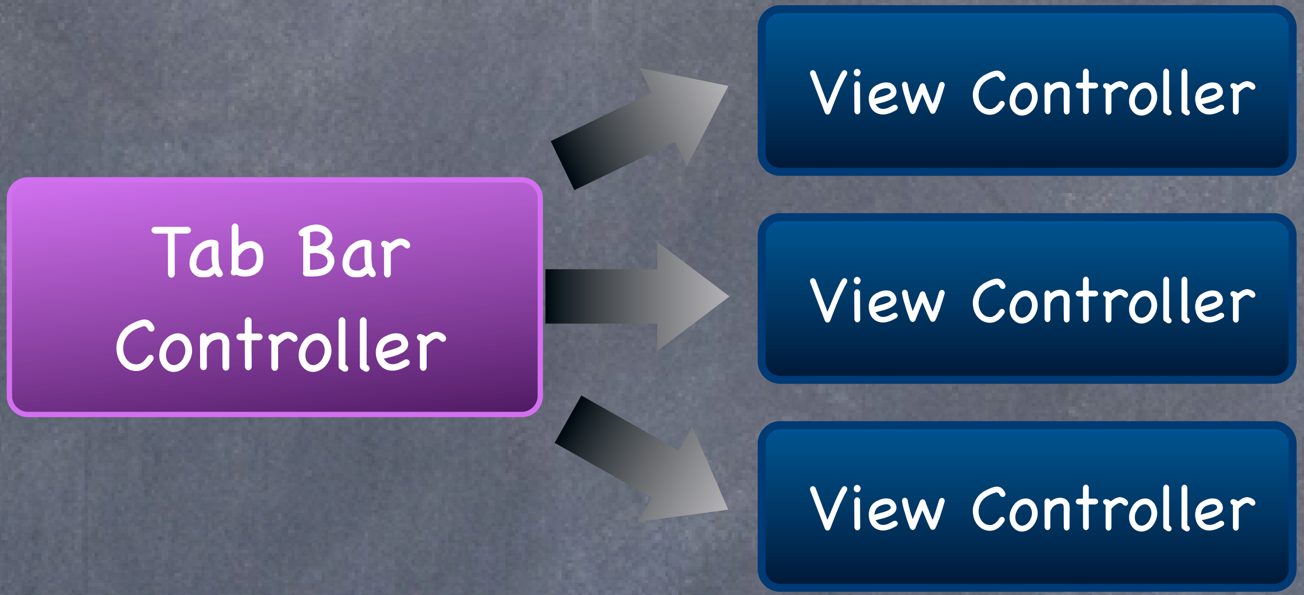


You control drag to create these connections in Xcode.

Doing so is setting
`@property (nonatomic, strong) NSArray *viewControllers;`
inside your UITabBarController.

UITabBarController

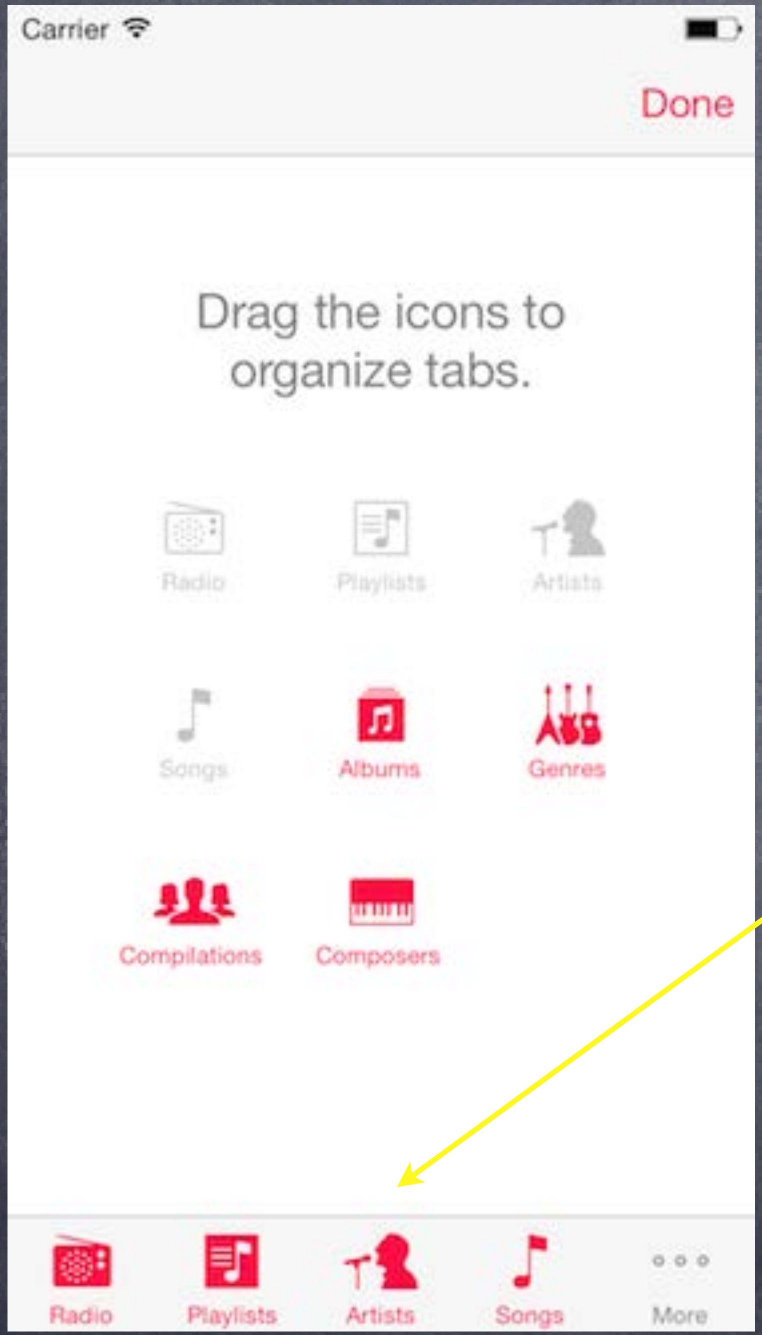




By default this is the UIViewController's title property (and no image)

But usually you set both of these in your storyboard in Xcode.

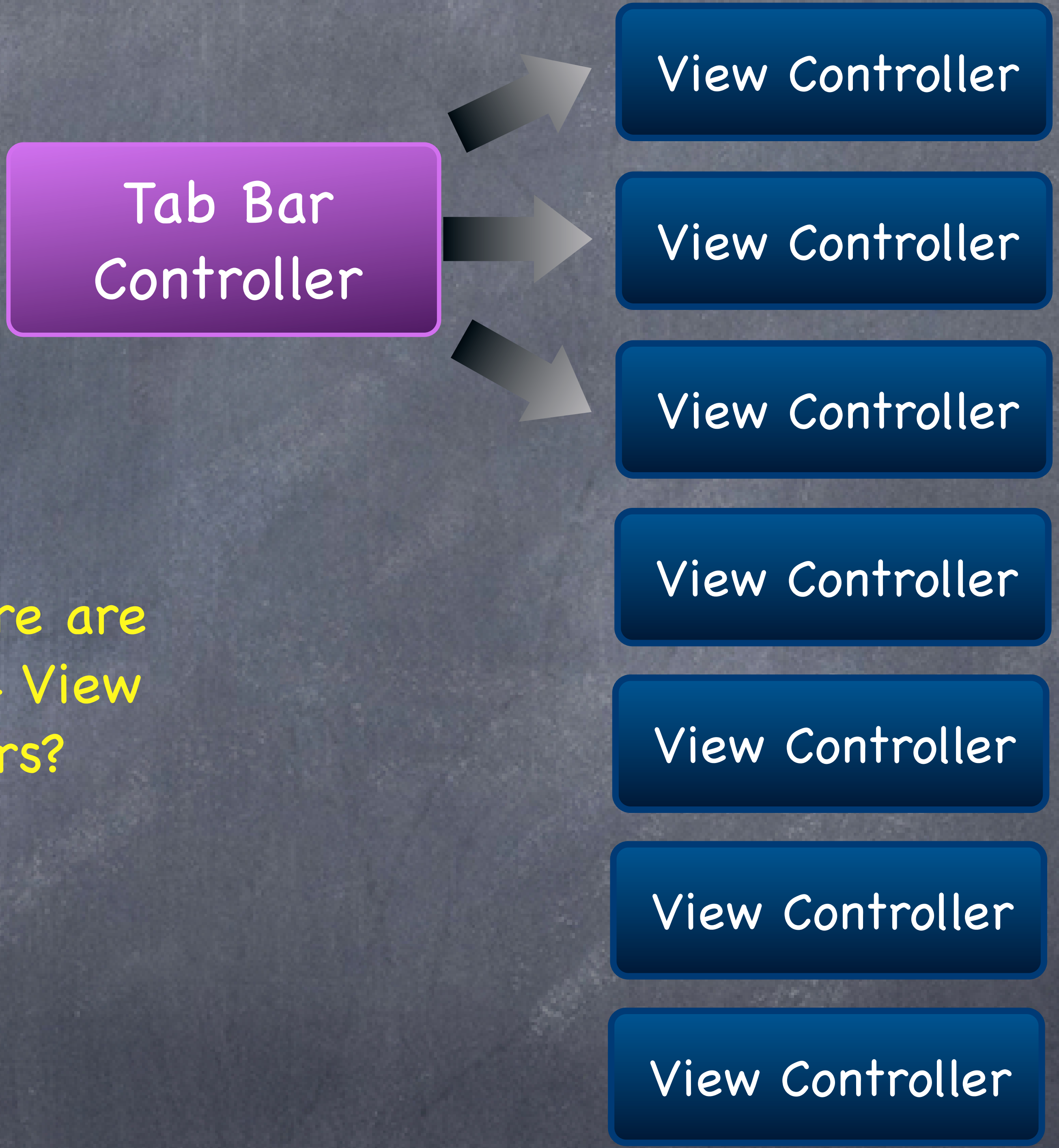
UITabBarController



Carrier

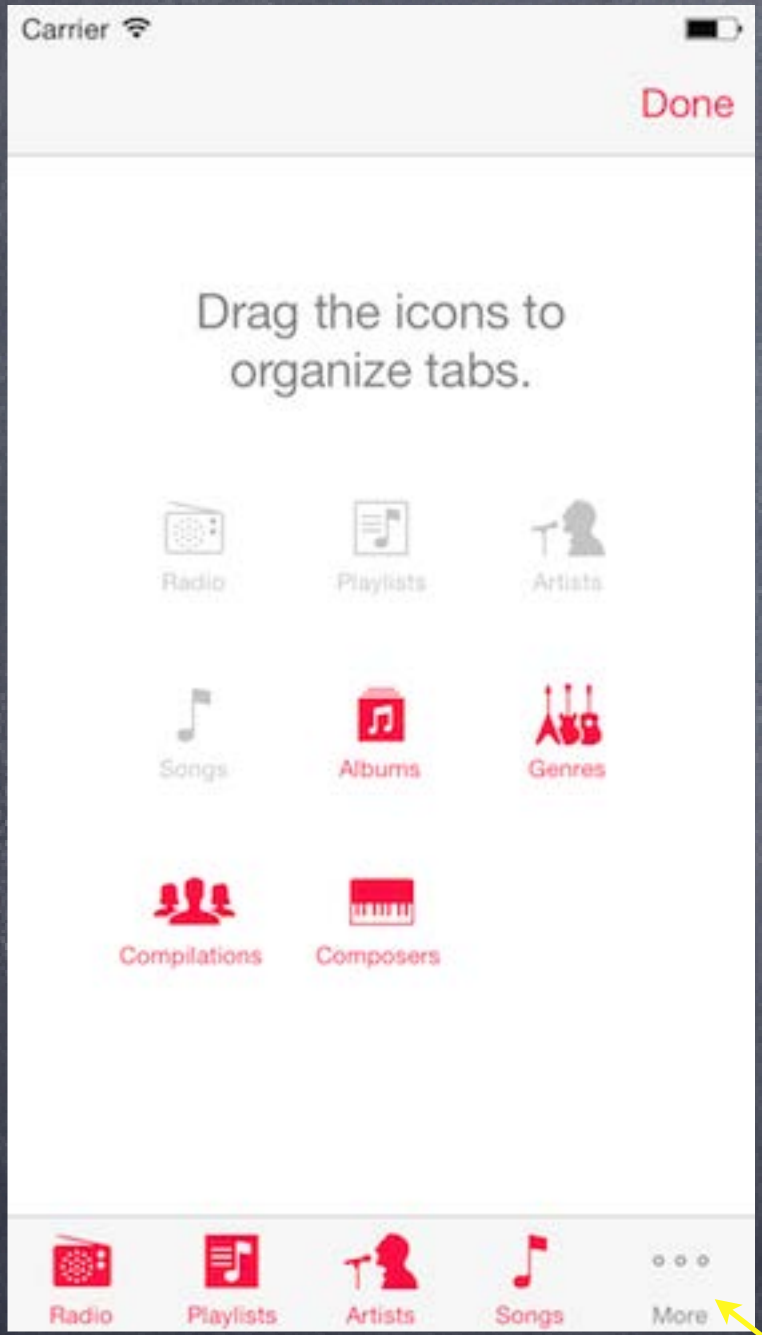
Done

Drag the icons to organize tabs.



What if there are more than 4 View Controllers?

UITabBarController



Carrier

Done

Drag the icons to organize tabs.

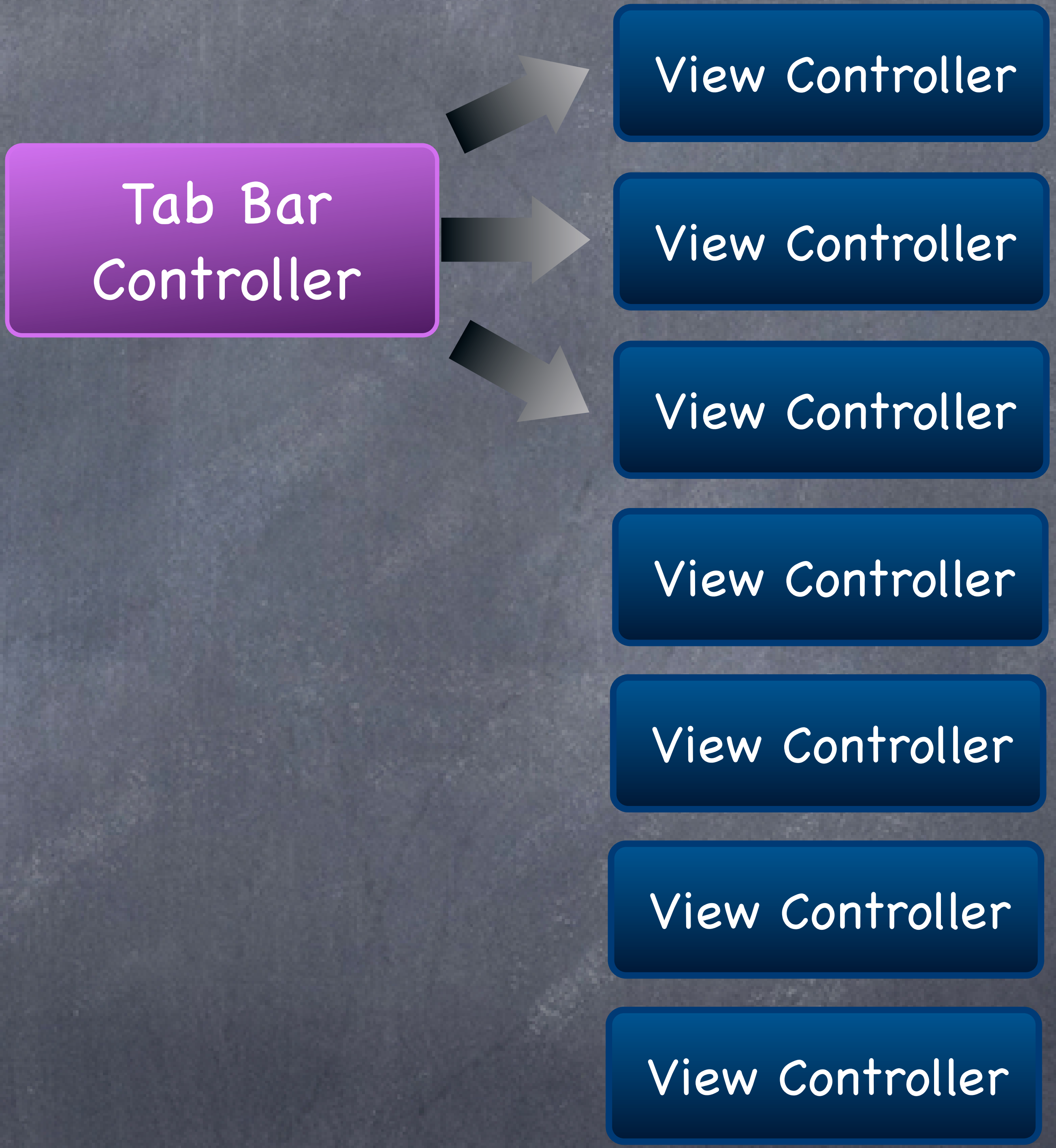
Radio Playlists Artists

Songs Albums Genres

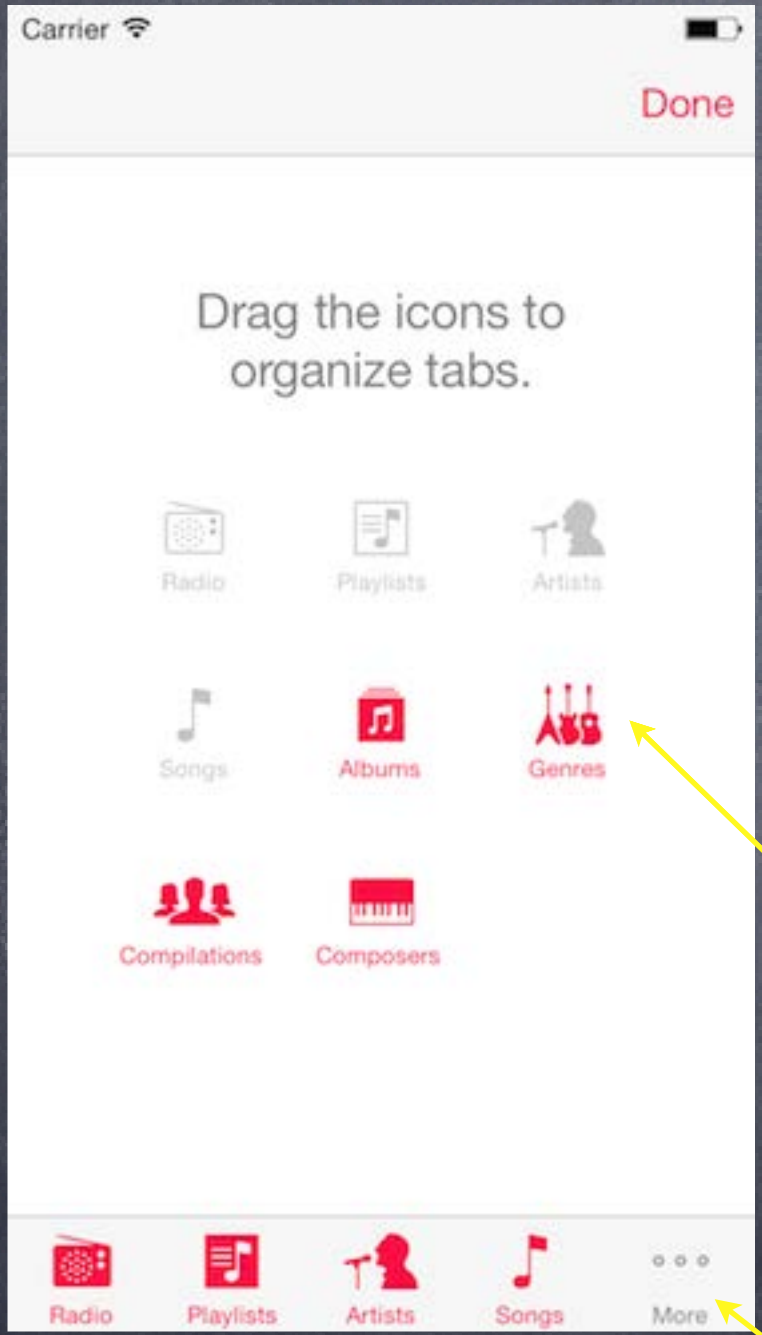
Compilations Composers

Radio Playlists Artists Songs More

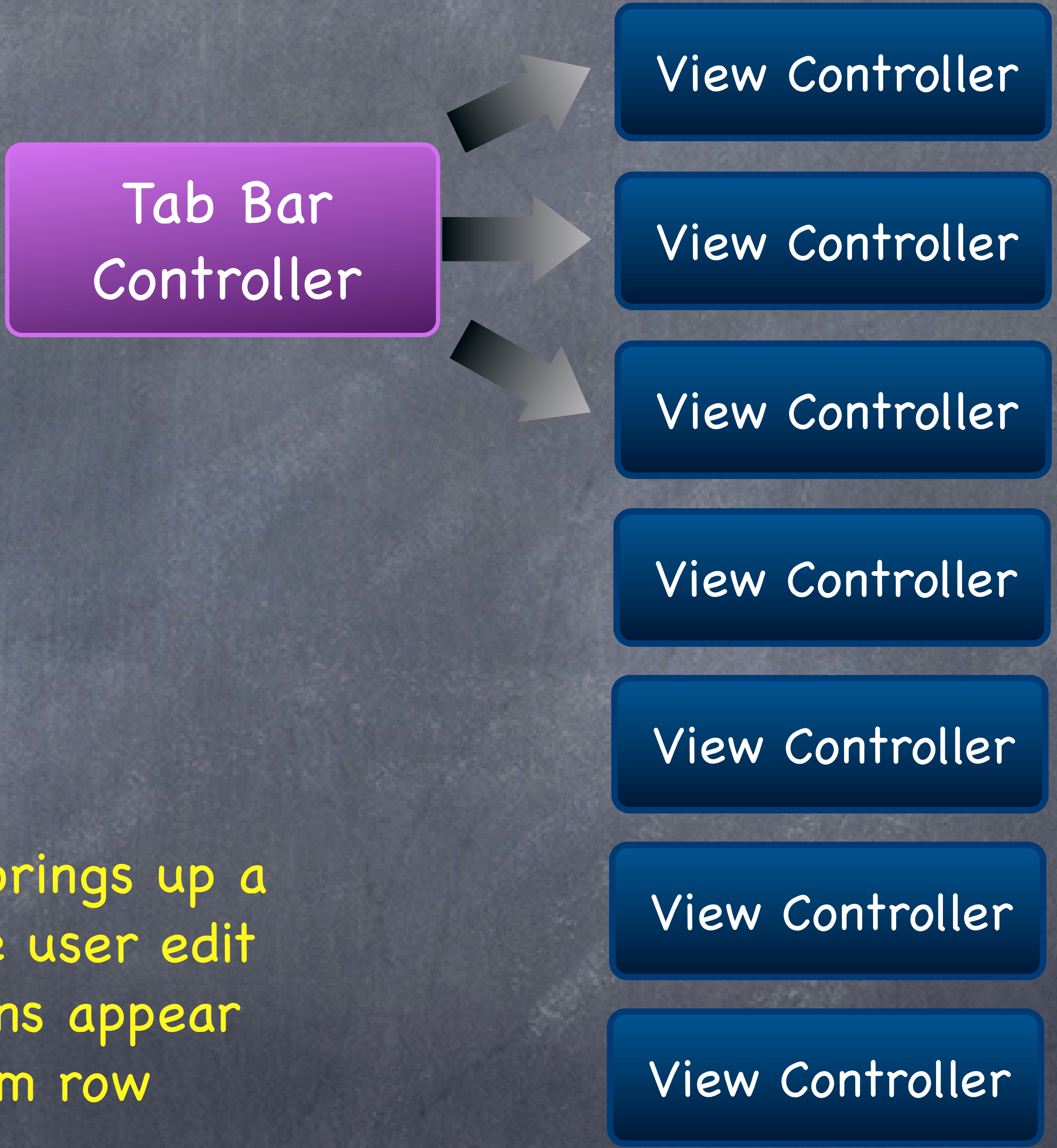
A More button appears.



UITabBarController



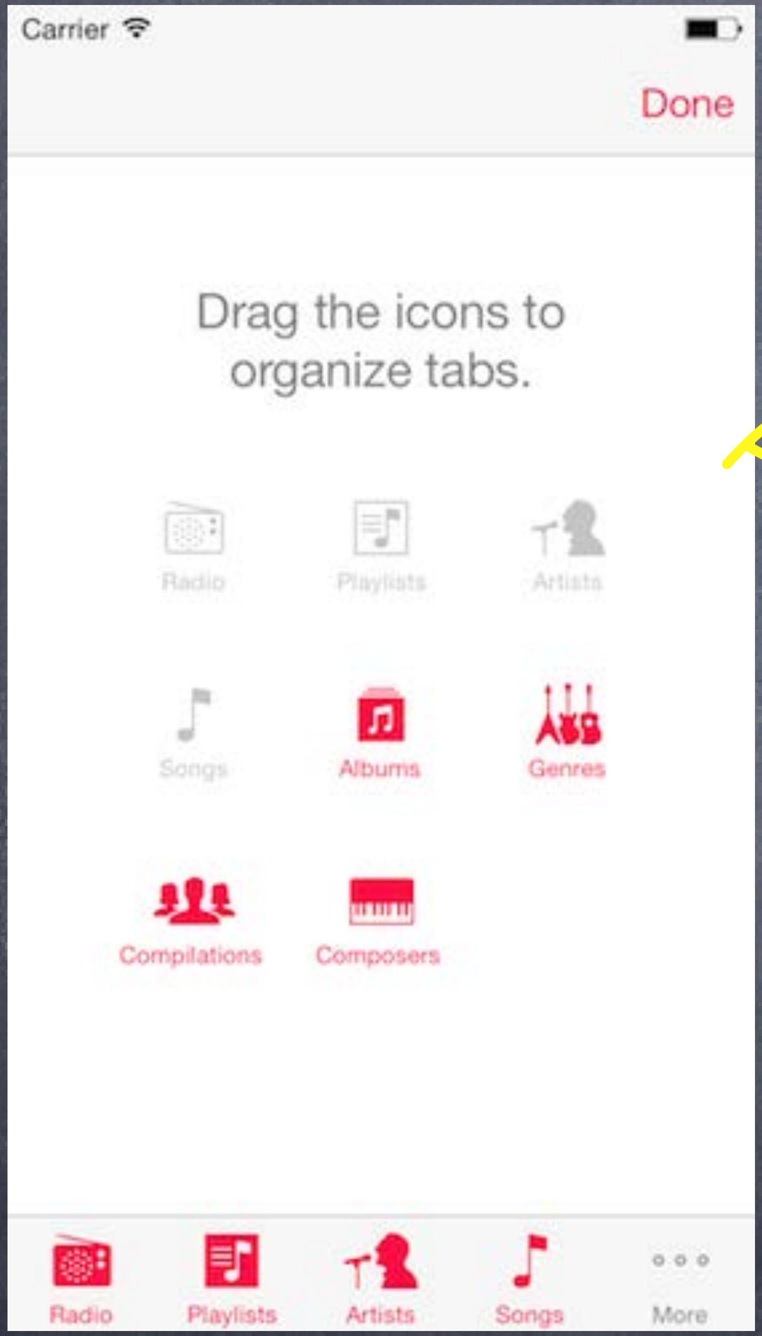
Drag the icons to organize tabs.



`More` button brings up a UI to let the user edit which buttons appear on bottom row

A `More` button appears.

UITabBarController

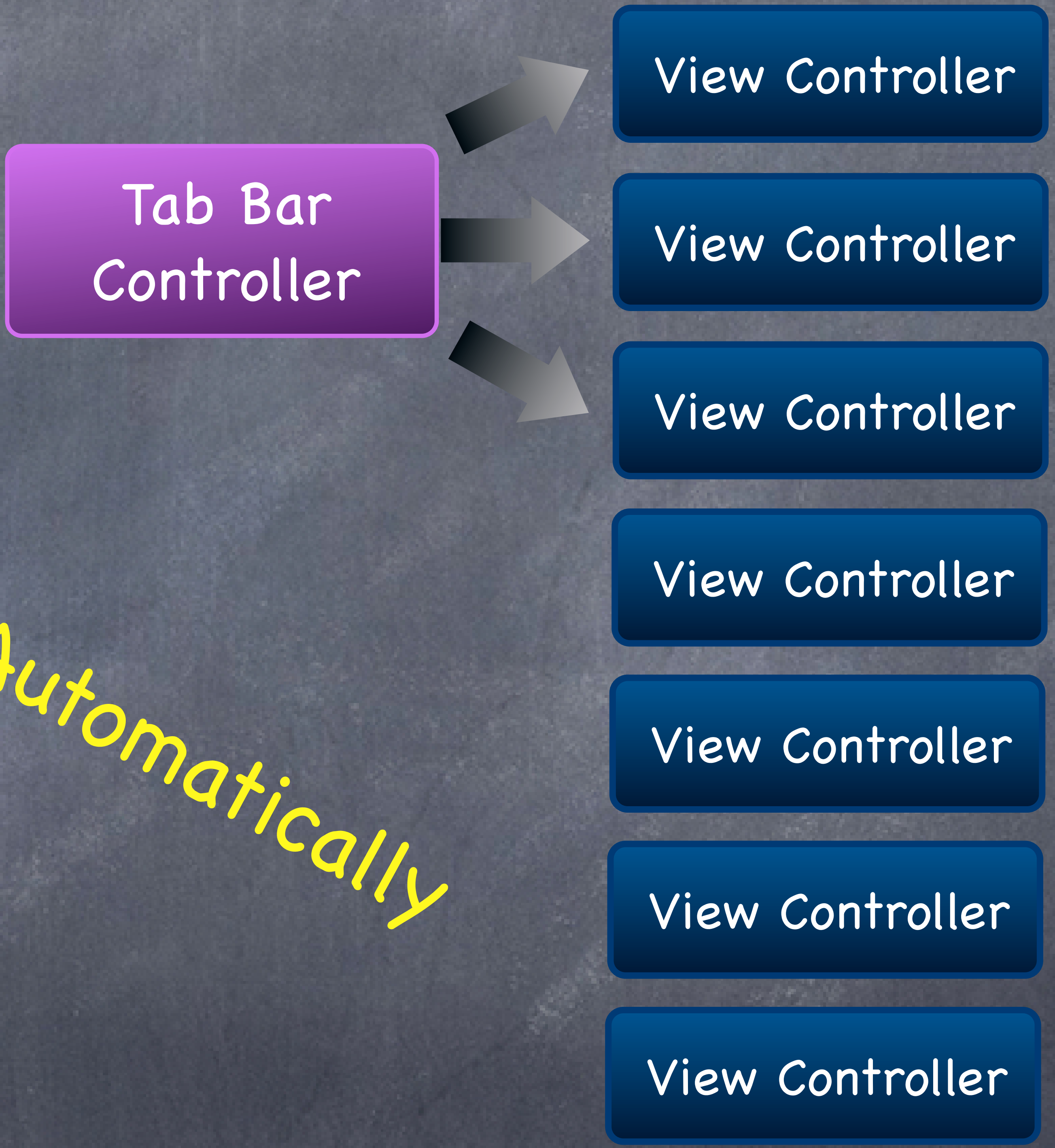


Carrier

Done

Drag the icons to organize tabs.

All Happens Automatically



Tab Bar Controller

View Controller

View Controller

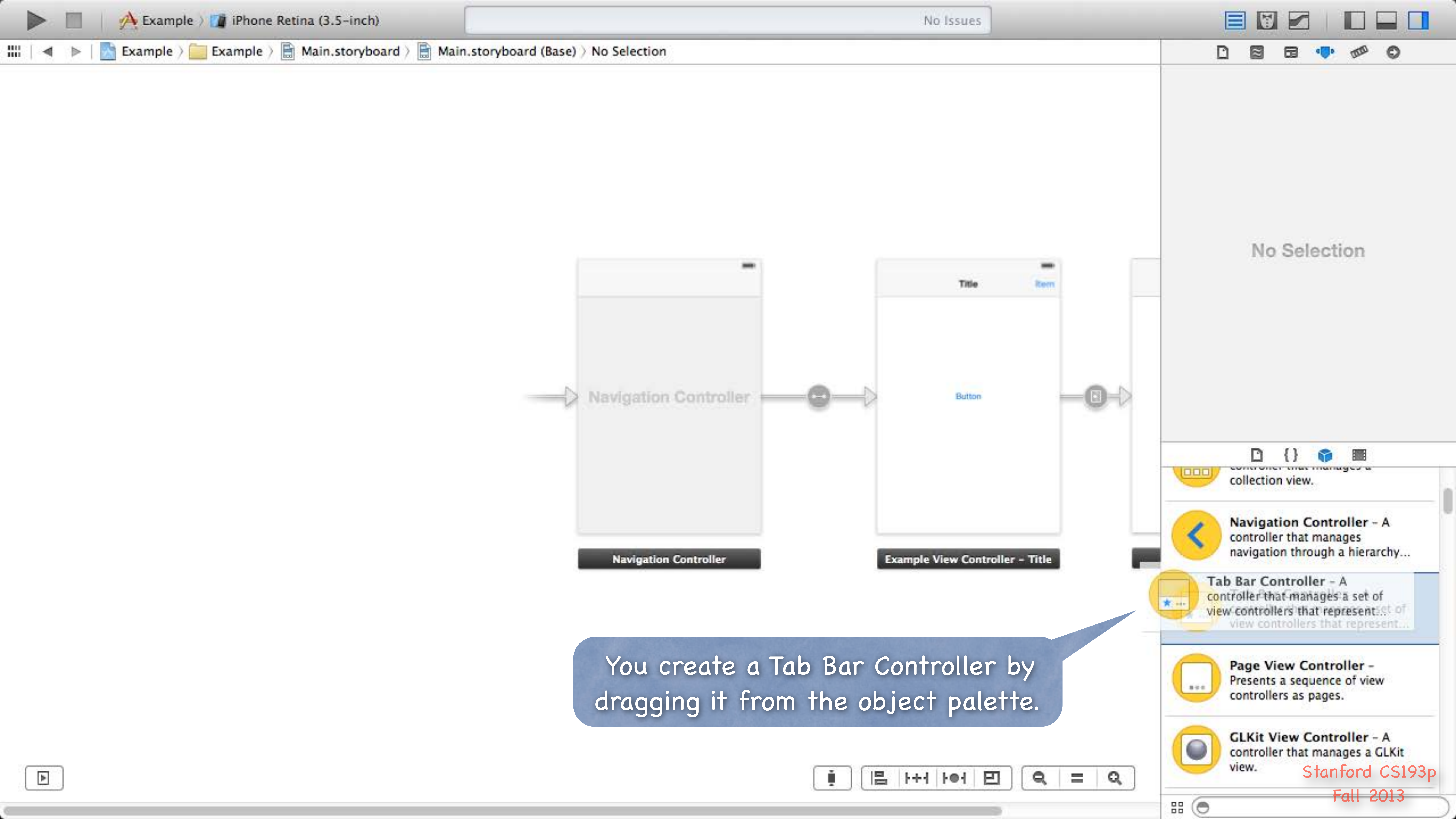
View Controller

View Controller

View Controller

View Controller

View Controller



collection view.

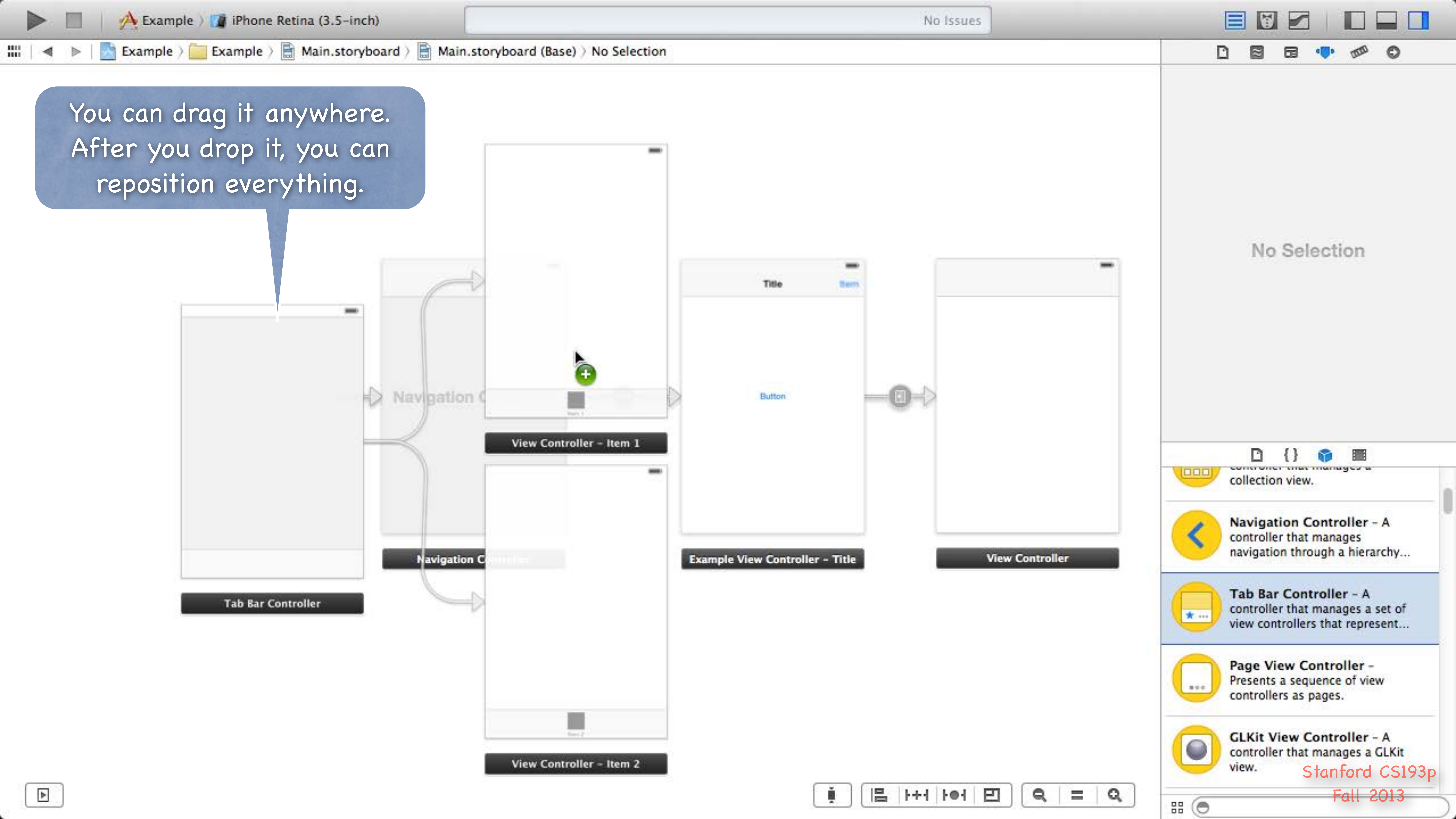
Navigation Controller - A controller that manages navigation through a hierarchy...

Tab Bar Controller - A controller that manages a set of view controllers that represent...

Page View Controller - Presents a sequence of view controllers as pages.

GLKit View Controller - A controller that manages a GLKit view.

You create a Tab Bar Controller by dragging it from the object palette.



You can drag it anywhere. After you drop it, you can reposition everything.

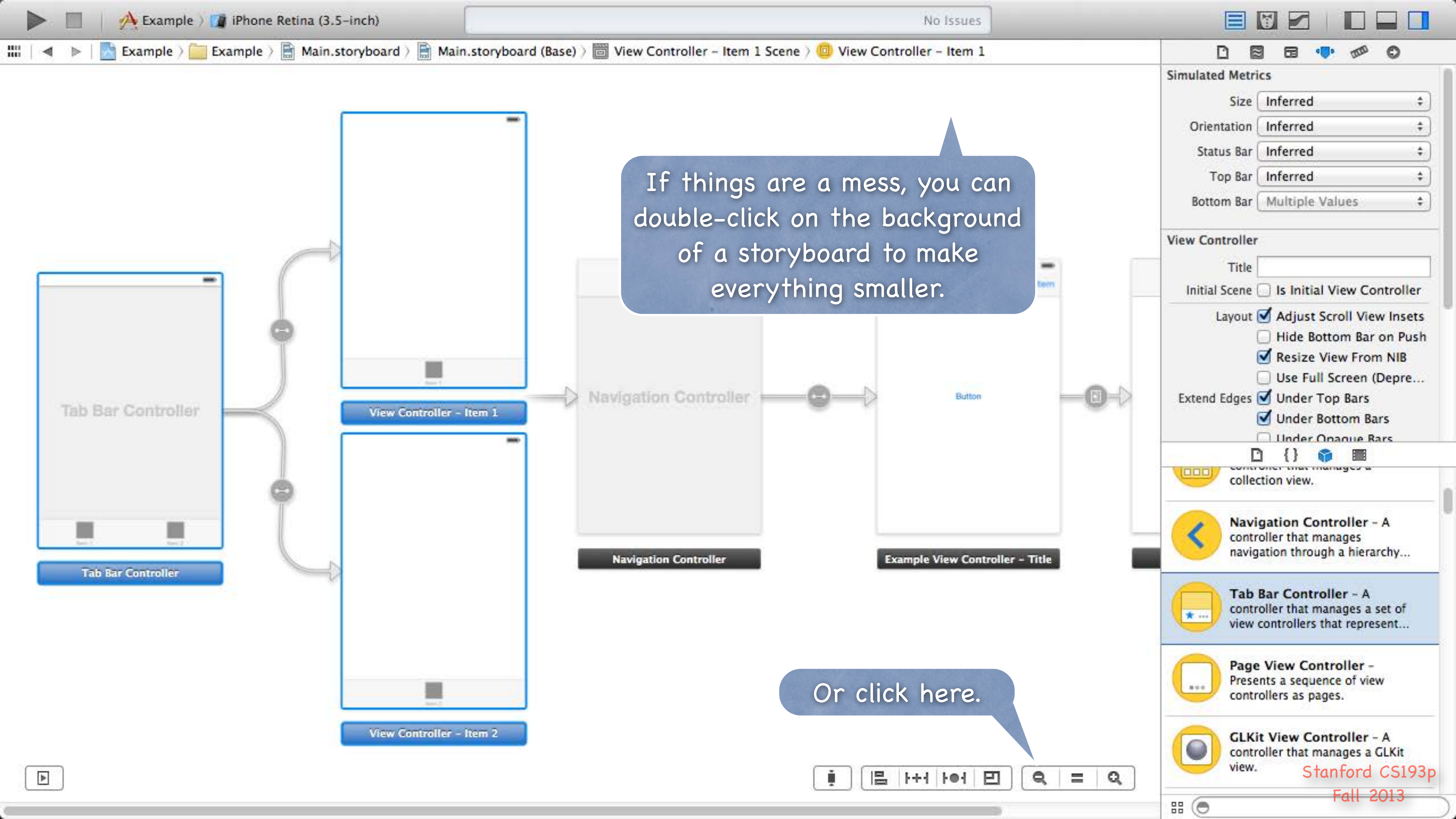
collection view.

Navigation Controller – A controller that manages navigation through a hierarchy...

Tab Bar Controller – A controller that manages a set of view controllers that represent...

Page View Controller – Presents a sequence of view controllers as pages.

GLKit View Controller – A controller that manages a GLKit view.



If things are a mess, you can double-click on the background of a storyboard to make everything smaller.

Or click here.

Example 〉 iPhone Retina (3.5-inch)

No Issues

Example 〉 Example 〉 Main.storyboard 〉 Main.storyboard (Base) 〉 View Controller – Item 1 Scene 〉 View Controller – Item 1

Simulated Metrics

Size: Inferred

Orientation: Inferred

Status Bar: Inferred

Top Bar: Inferred

Bottom Bar: Multiple Values

View Controller

Title:

Initial Scene: □ Is Initial View Controller

Layout: ☑ Adjust Scroll View Insets

□ Hide Bottom Bar on Push

☑ Resize View From NIB

□ Use Full Screen (Depre...

Extend Edges: ☑ Under Top Bars

☑ Under Bottom Bars

□ Under Opaque Bars

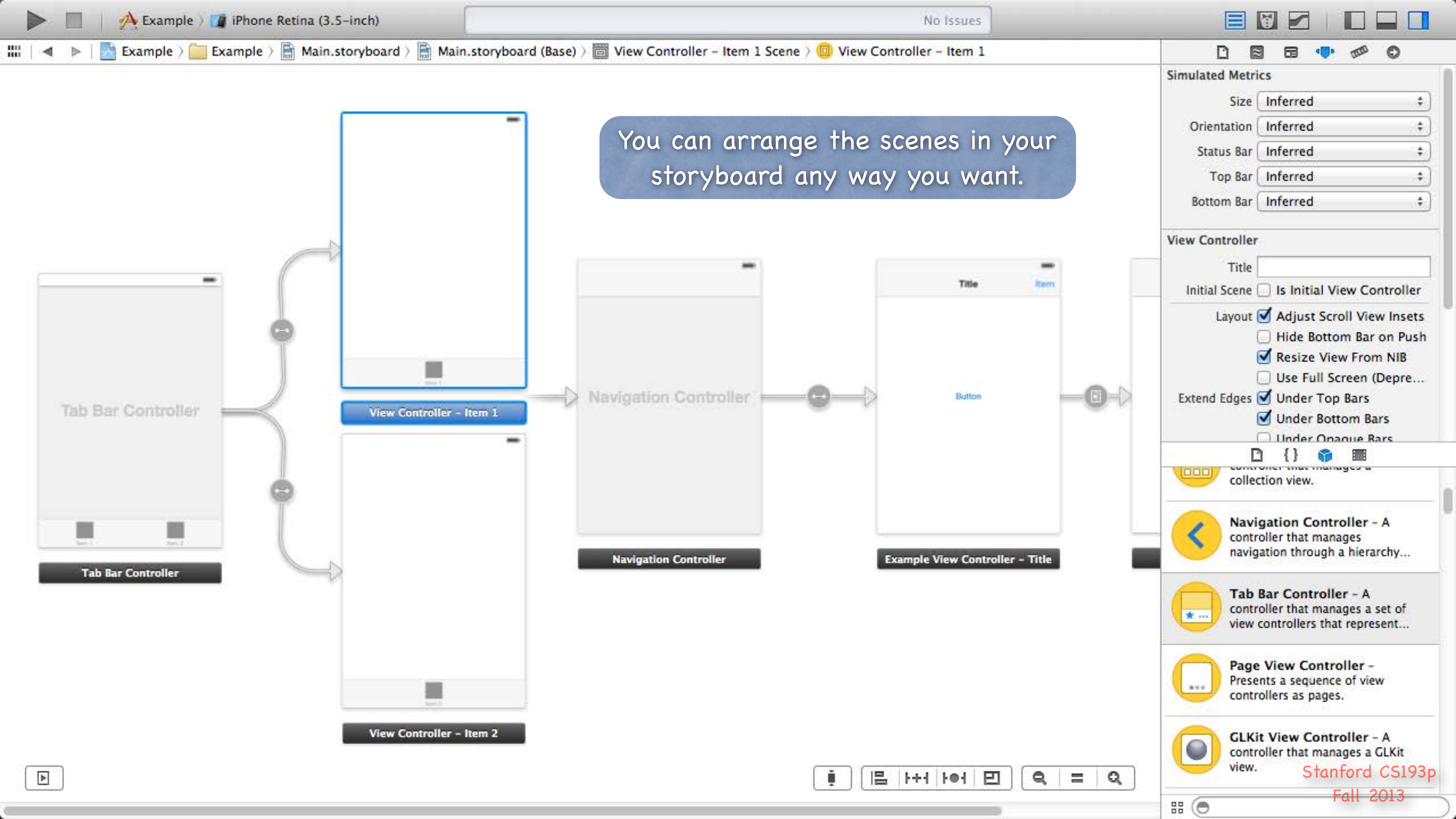
controller that manages a collection view.

Navigation Controller – A controller that manages navigation through a hierarchy...

Tab Bar Controller – A controller that manages a set of view controllers that represent...

Page View Controller – Presents a sequence of view controllers as pages.

GLKit View Controller – A controller that manages a GLKit view.



You can arrange the scenes in your storyboard any way you want.

Simulated Metrics

Size: Inferred

Orientation: Inferred

Status Bar: Inferred

Top Bar: Inferred

Bottom Bar: Inferred

View Controller

Title:

Initial Scene: ☐ Is Initial View Controller

Layout: ☑ Adjust Scroll View Insets
☐ Hide Bottom Bar on Push
☑ Resize View From NIB
☐ Use Full Screen (Depre...

Extend Edges: ☑ Under Top Bars
☑ Under Bottom Bars
☐ Under Opaque Bars

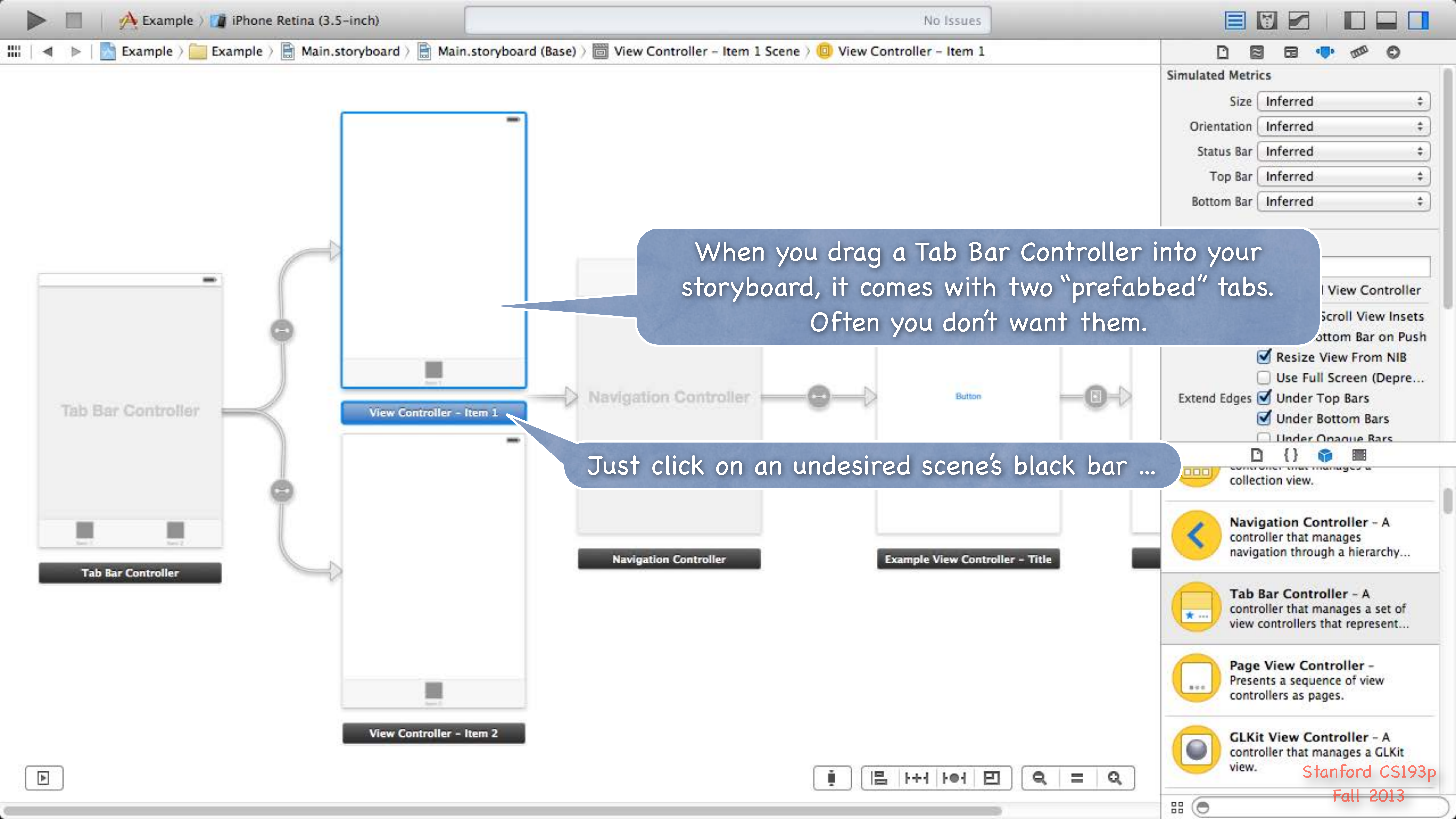
controller that manages a collection view.

Navigation Controller – A controller that manages navigation through a hierarchy...

Tab Bar Controller – A controller that manages a set of view controllers that represent...

Page View Controller – Presents a sequence of view controllers as pages.

GLKit View Controller – A controller that manages a GLKit view.



When you drag a Tab Bar Controller into your storyboard, it comes with two "prefabbed" tabs. Often you don't want them.

Just click on an undesired scene's black bar …

Simulated Metrics

Size: Inferred

Orientation: Inferred

Status Bar: Inferred

Top Bar: Inferred

Bottom Bar: Inferred

☑ Resize View From NIB

□ Use Full Screen (Depre…

Extend Edges ☑ Under Top Bars

☑ Under Bottom Bars

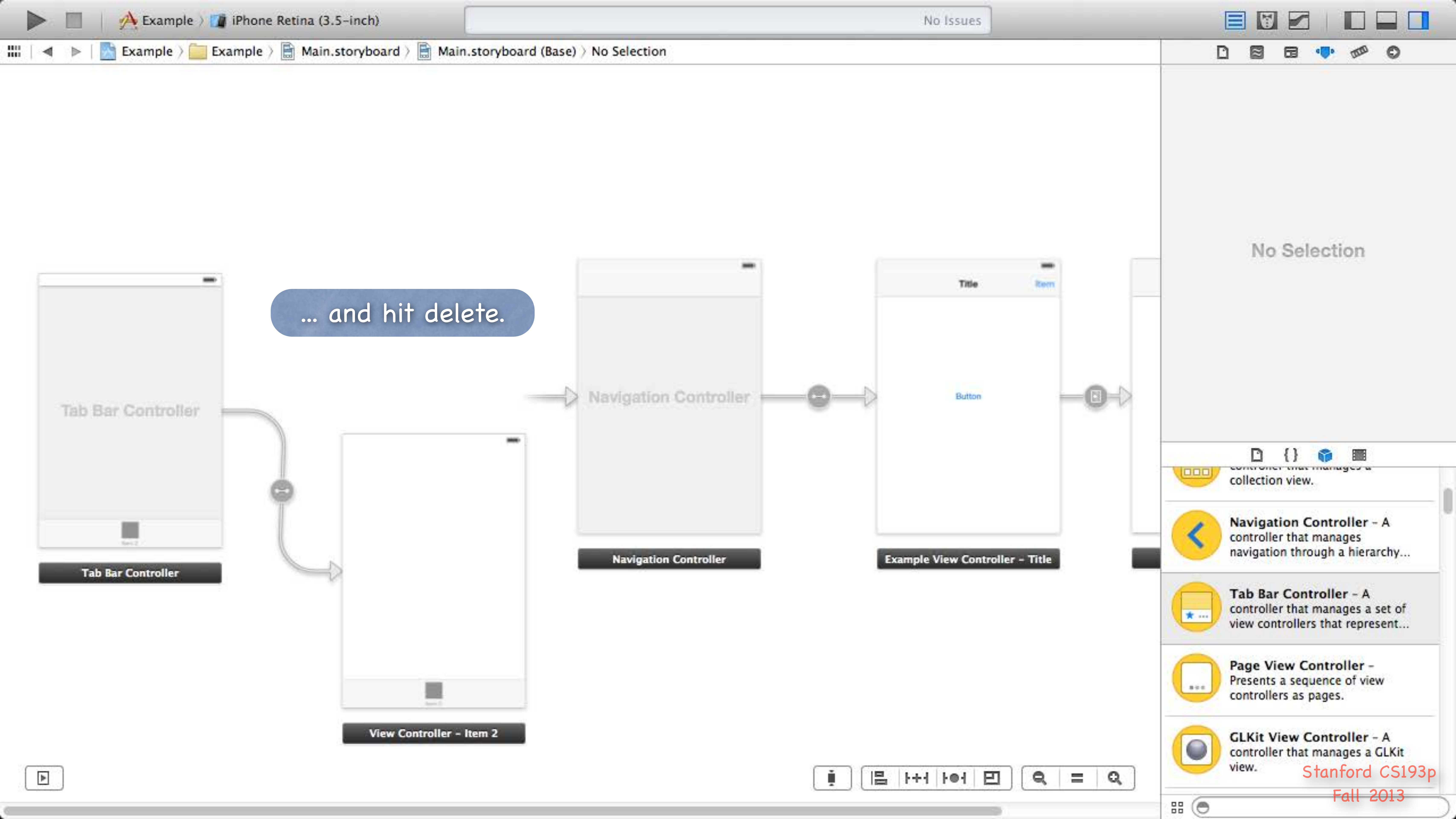
□ Under Opaque Bars

Navigation Controller - A controller that manages navigation through a hierarchy…

Tab Bar Controller - A controller that manages a set of view controllers that represent…

Page View Controller - Presents a sequence of view controllers as pages.

GLKit View Controller - A controller that manages a GLKit view.



… and hit delete.

controller that manages a collection view.

Navigation Controller – A controller that manages navigation through a hierarchy…

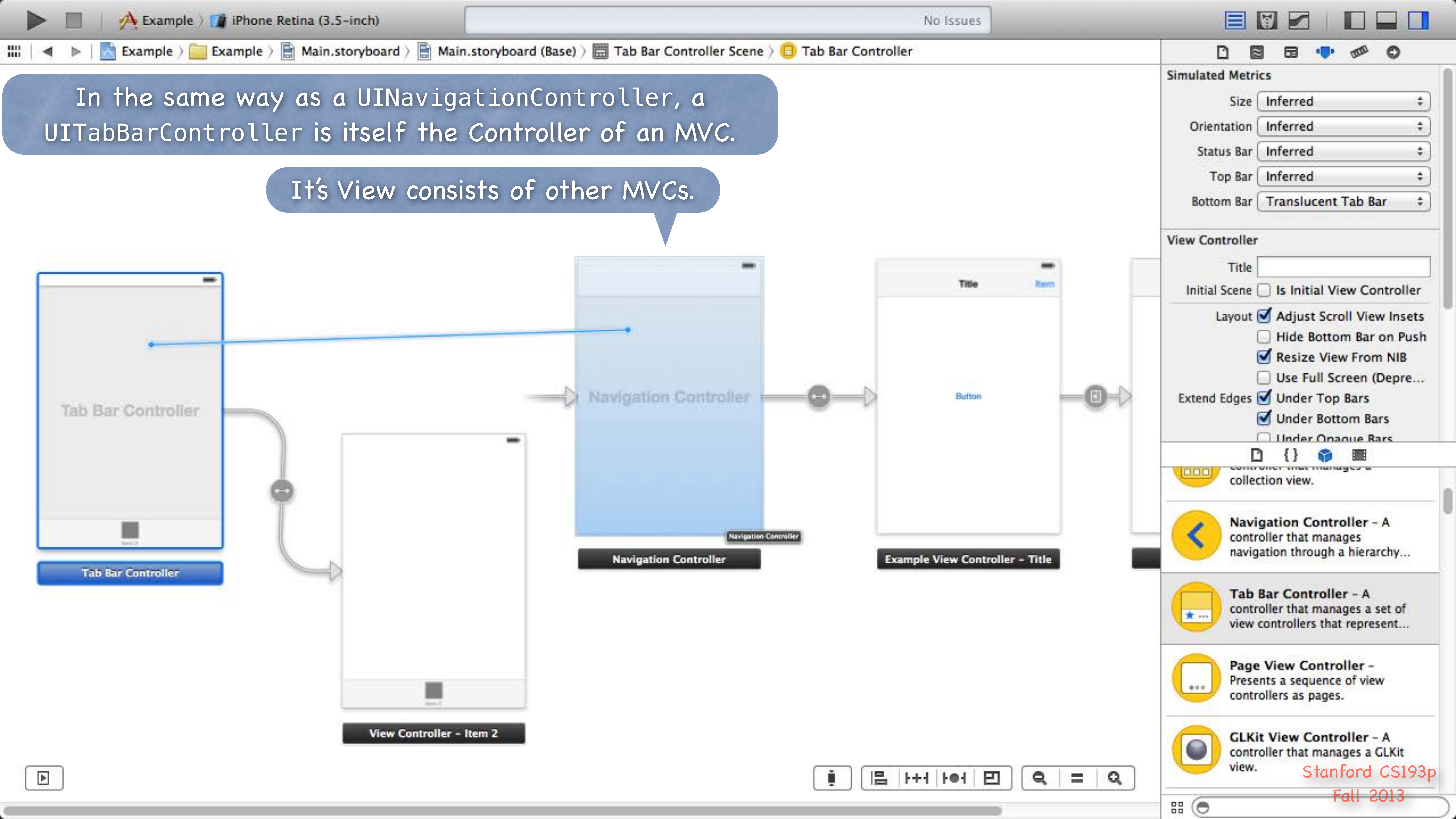
Tab Bar Controller – A controller that manages a set of view controllers that represent…

Page View Controller – Presents a sequence of view controllers as pages.

GLKit View Controller – A controller that manages a GLKit view.

In the same way as a UINavigationController, a UITabBarController is itself the Controller of an MVC.

It's View consists of other MVCs.



Simulated Metrics

Size: Inferred

Orientation: Inferred

Status Bar: Inferred

Top Bar: Inferred

Bottom Bar: Translucent Tab Bar

View Controller

Title:

Initial Scene: ☐ Is Initial View Controller

Layout: ☑ Adjust Scroll View Insets

☐ Hide Bottom Bar on Push

☑ Resize View From NIB

☐ Use Full Screen (Depre...

Extend Edges: ☑ Under Top Bars

☑ Under Bottom Bars

☐ Under Opaque Bars

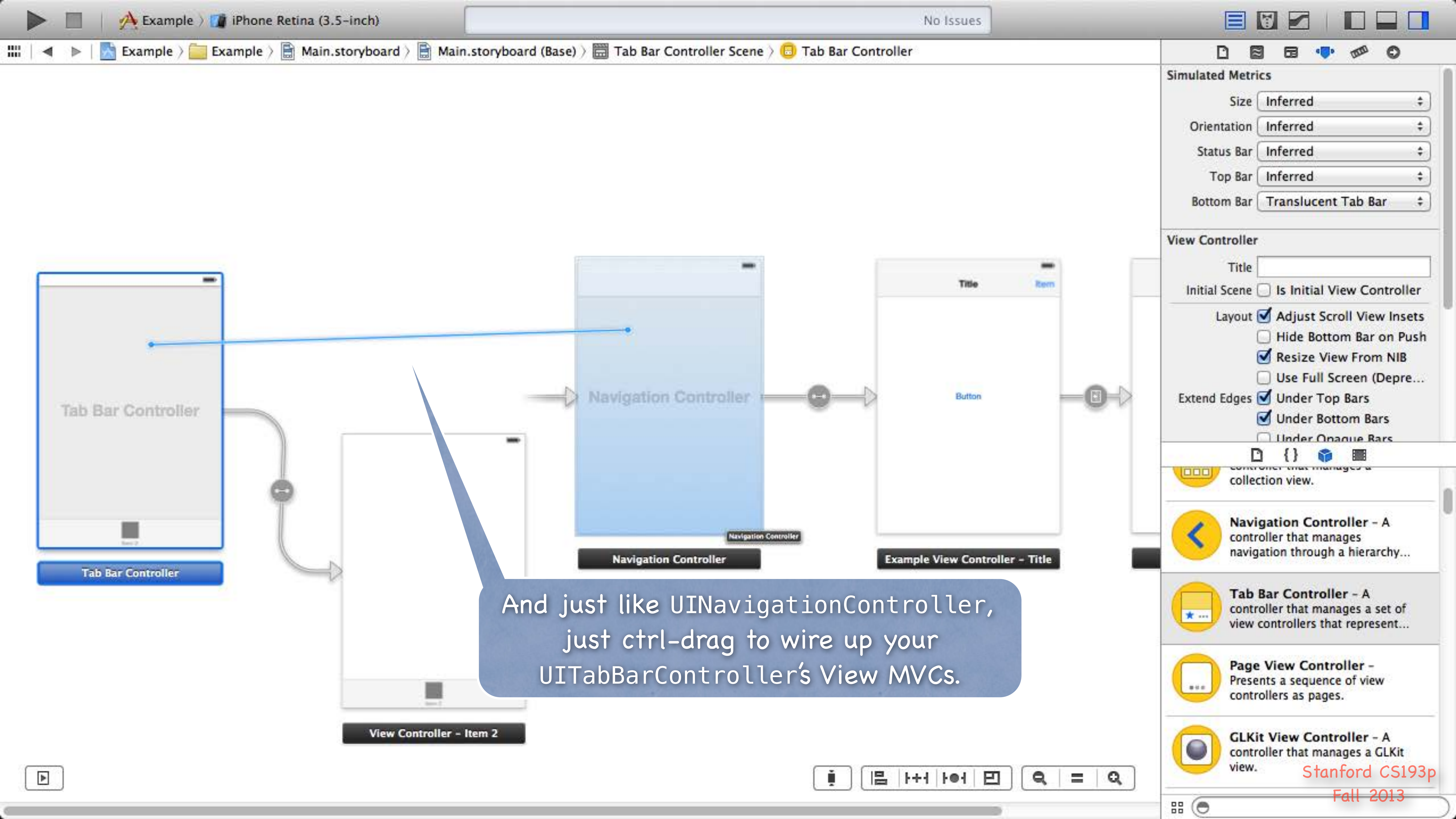
controller that manages a collection view.

Navigation Controller - A controller that manages navigation through a hierarchy...

Tab Bar Controller - A controller that manages a set of view controllers that represent...

Page View Controller - Presents a sequence of view controllers as pages.

GLKit View Controller - A controller that manages a GLKit view.



Simulated Metrics

Size: Inferred

Orientation: Inferred

Status Bar: Inferred

Top Bar: Inferred

Bottom Bar: Translucent Tab Bar

View Controller

Title:

Initial Scene: □ Is Initial View Controller

Layout: ☑ Adjust Scroll View Insets
□ Hide Bottom Bar on Push
☑ Resize View From NIB
□ Use Full Screen (Depre...

Extend Edges: ☑ Under Top Bars
☑ Under Bottom Bars
□ Under Opaque Bars

collection view.

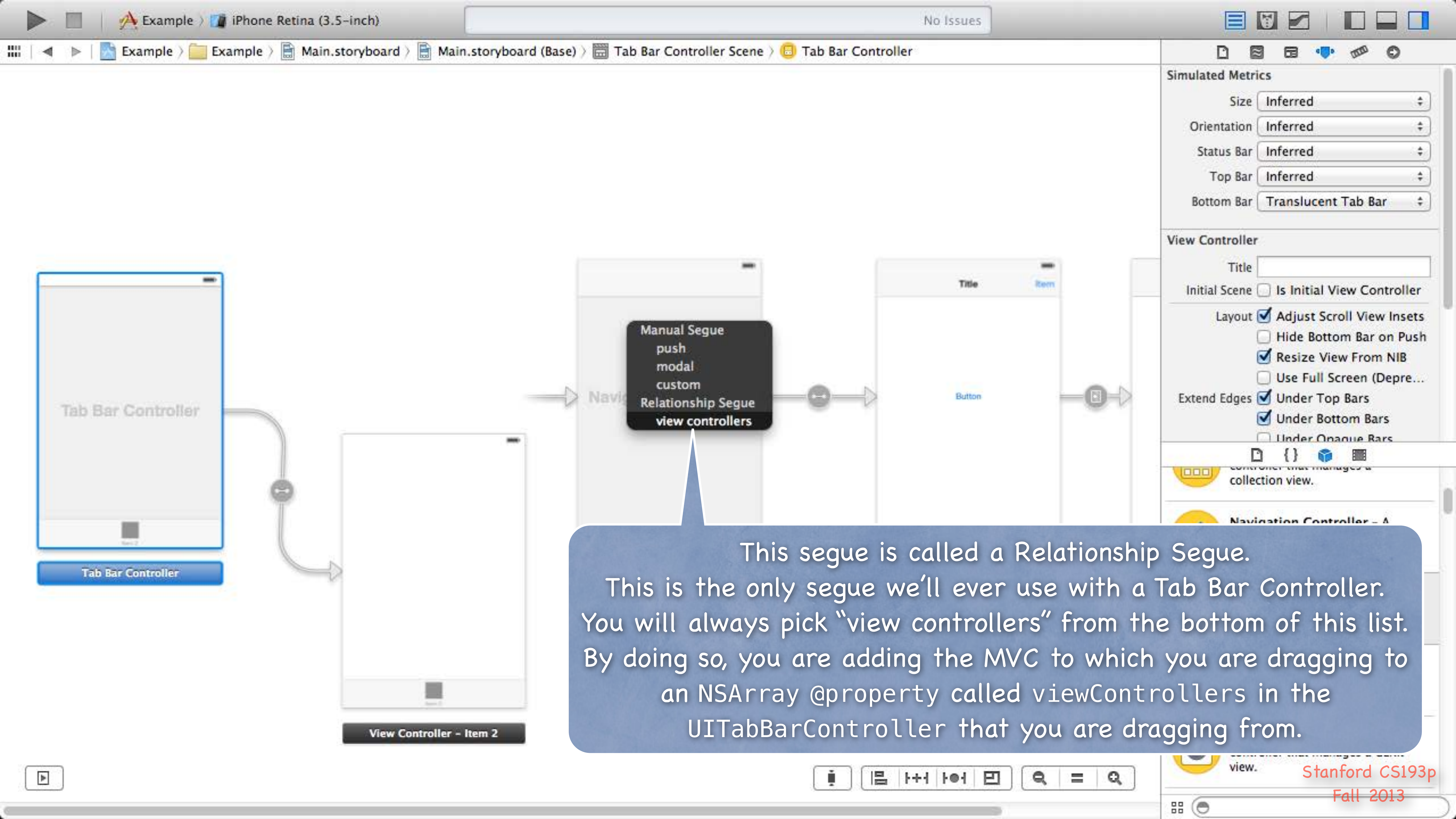
Navigation Controller - A controller that manages navigation through a hierarchy...

Tab Bar Controller - A controller that manages a set of view controllers that represent...

Page View Controller - Presents a sequence of view controllers as pages.

GLKit View Controller - A controller that manages a GLKit view.

And just like UINavigationController, just ctrl-drag to wire up your UITabBarController's View MVCs.



Simulated Metrics

Size: Inferred

Orientation: Inferred

Status Bar: Inferred

Top Bar: Inferred

Bottom Bar: Translucent Tab Bar

View Controller

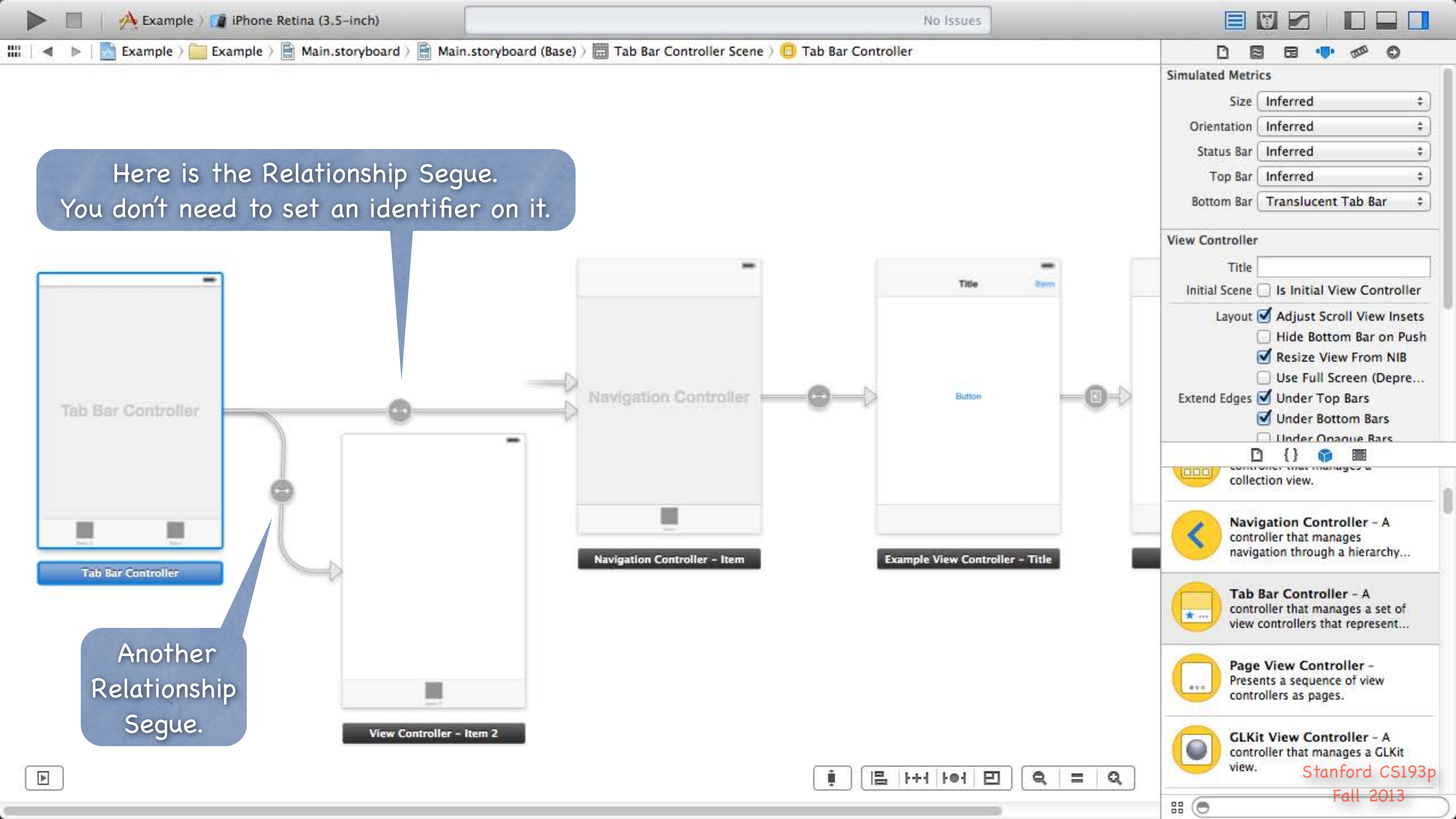
Title:

Initial Scene: □ Is Initial View Controller

Layout: ☑ Adjust Scroll View Insets
□ Hide Bottom Bar on Push
☑ Resize View From NIB
□ Use Full Screen (Depre...

Extend Edges: ☑ Under Top Bars
☑ Under Bottom Bars
□ Under Opaque Bars

This segue is called a Relationship Segue.
This is the only segue we'll ever use with a Tab Bar Controller.
You will always pick "view controllers" from the bottom of this list.
By doing so, you are adding the MVC to which you are dragging to an NSArray `@property` called `viewControllers` in the `UITabBarController` that you are dragging from.


Example 〉 iPhone Retina (3.5-inch)
No Issues
Example 〉 Example 〉 Main.storyboard 〉 Main.storyboard (Base) 〉 Tab Bar Controller Scene 〉 Tab Bar Controller
Tab Bar Controller
Navigation Controller
Title
Item
Button
Tab Bar Controller
View Controller – Item 2
Navigation Controller – Item
Example View Controller – Title

Here is the Relationship Segue.
You don't need to set an identifier on it.

Another Relationship Segue.

Simulated Metrics

Size: Inferred

Orientation: Inferred

Status Bar: Inferred

Top Bar: Inferred

Bottom Bar: Translucent Tab Bar

View Controller

Title:

Initial Scene: □ Is Initial View Controller

Layout: ☑ Adjust Scroll View Insets
□ Hide Bottom Bar on Push
☑ Resize View From NIB
□ Use Full Screen (Depre...

Extend Edges: ☑ Under Top Bars
☑ Under Bottom Bars
□ Under Opaque Bars

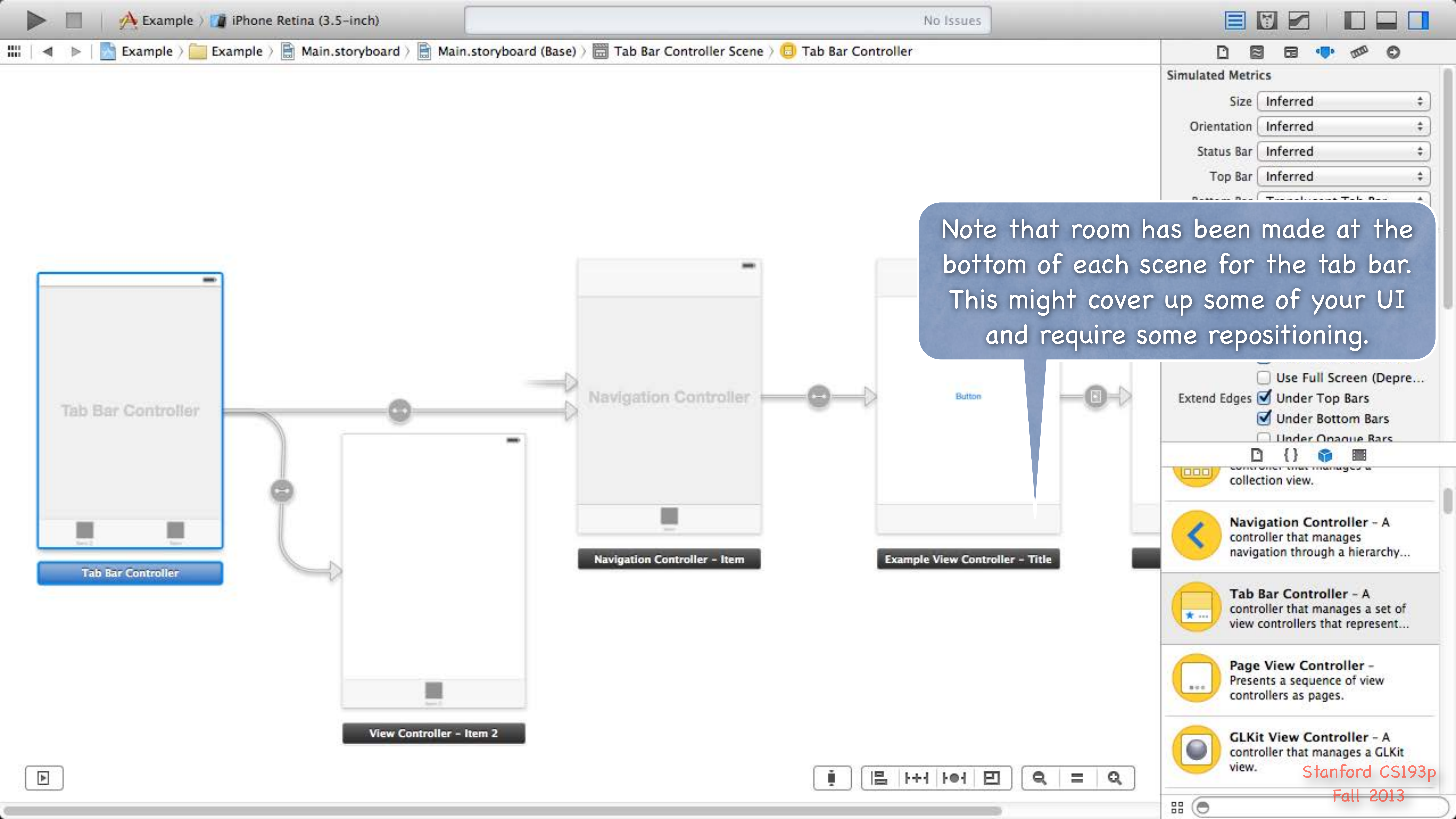
...controller that manages a collection view.

Navigation Controller – A controller that manages navigation through a hierarchy...

Tab Bar Controller – A controller that manages a set of view controllers that represent...

Page View Controller – Presents a sequence of view controllers as pages.

GLKit View Controller – A controller that manages a GLKit view.


Example 〉 iPhone Retina (3.5-inch)
No Issues
Example 〉 Example 〉 Main.storyboard 〉 Main.storyboard (Base) 〉 Tab Bar Controller Scene 〉 Tab Bar Controller
Tab Bar Controller
Navigation Controller
Button
Tab Bar Controller
View Controller – Item 2
Navigation Controller – Item
Example View Controller – Title

Simulated Metrics

Size: Inferred

Orientation: Inferred

Status Bar: Inferred

Top Bar: Inferred

Bottom Bar: Translucent Tab Bar

□ Use Full Screen (Depre...

Extend Edges ☑ Under Top Bars

☑ Under Bottom Bars

□ Under Opaque Bars

collection view.

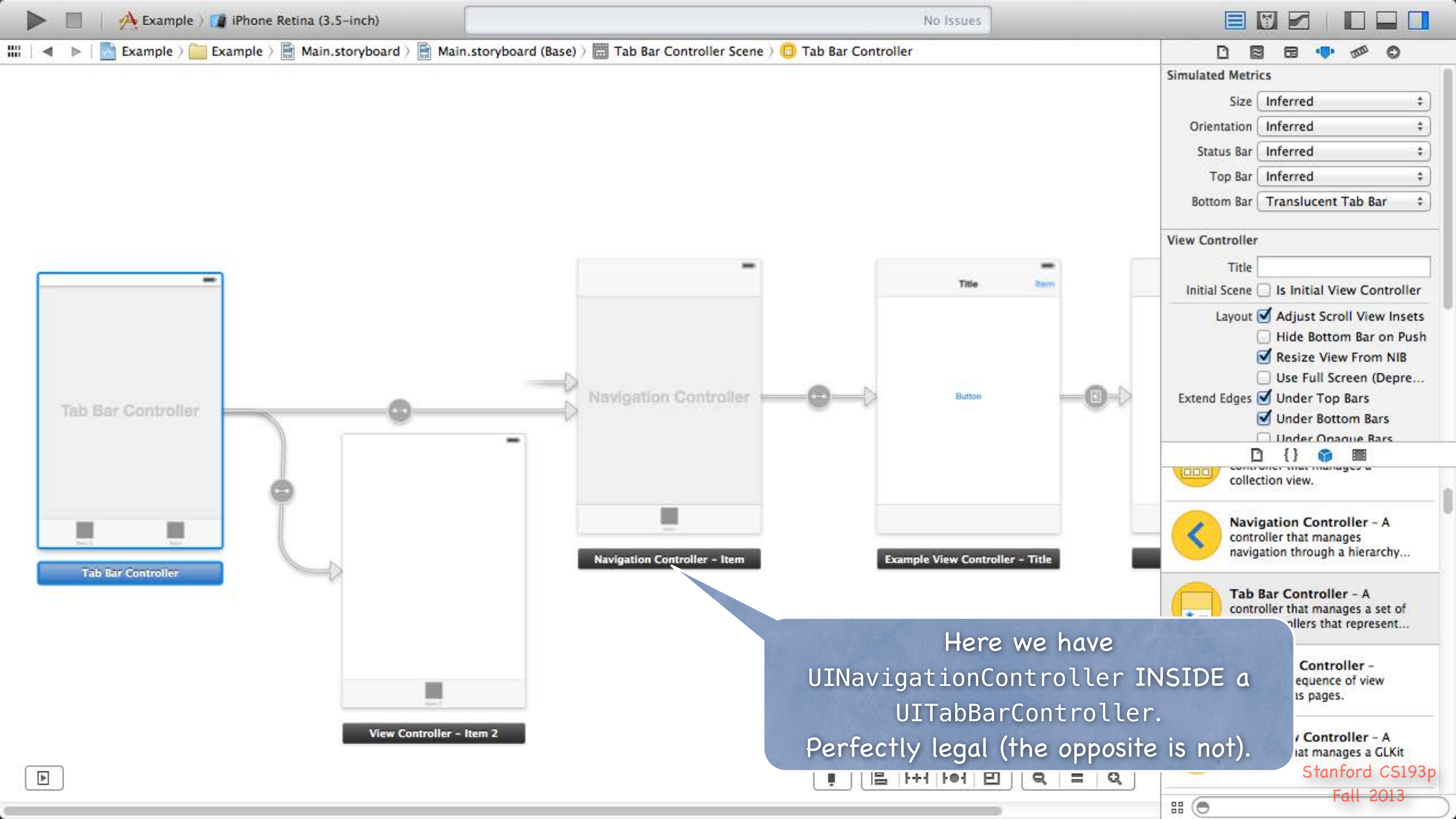
Navigation Controller – A controller that manages navigation through a hierarchy...

Tab Bar Controller – A controller that manages a set of view controllers that represent...

Page View Controller – Presents a sequence of view controllers as pages.

GLKit View Controller – A controller that manages a GLKit view.

Note that room has been made at the bottom of each scene for the tab bar. This might cover up some of your UI and require some repositioning.


Example 〉 iPhone Retina (3.5-inch)
No Issues
Example 〉 Example 〉 Main.storyboard 〉 Main.storyboard (Base) 〉 Tab Bar Controller Scene 〉 Tab Bar Controller
Tab Bar Controller
Navigation Controller
Title
Item
Button
Tab Bar Controller
View Controller – Item 2
Navigation Controller – Item
Example View Controller – Title

Simulated Metrics

Size: Inferred

Orientation: Inferred

Status Bar: Inferred

Top Bar: Inferred

Bottom Bar: Translucent Tab Bar

View Controller

Title:

Initial Scene: □ Is Initial View Controller

Layout: ☑ Adjust Scroll View Insets

□ Hide Bottom Bar on Push

☑ Resize View From NIB

□ Use Full Screen (Depre…

Extend Edges: ☑ Under Top Bars

☑ Under Bottom Bars

□ Under Opaque Bars

controller that manages a collection view.

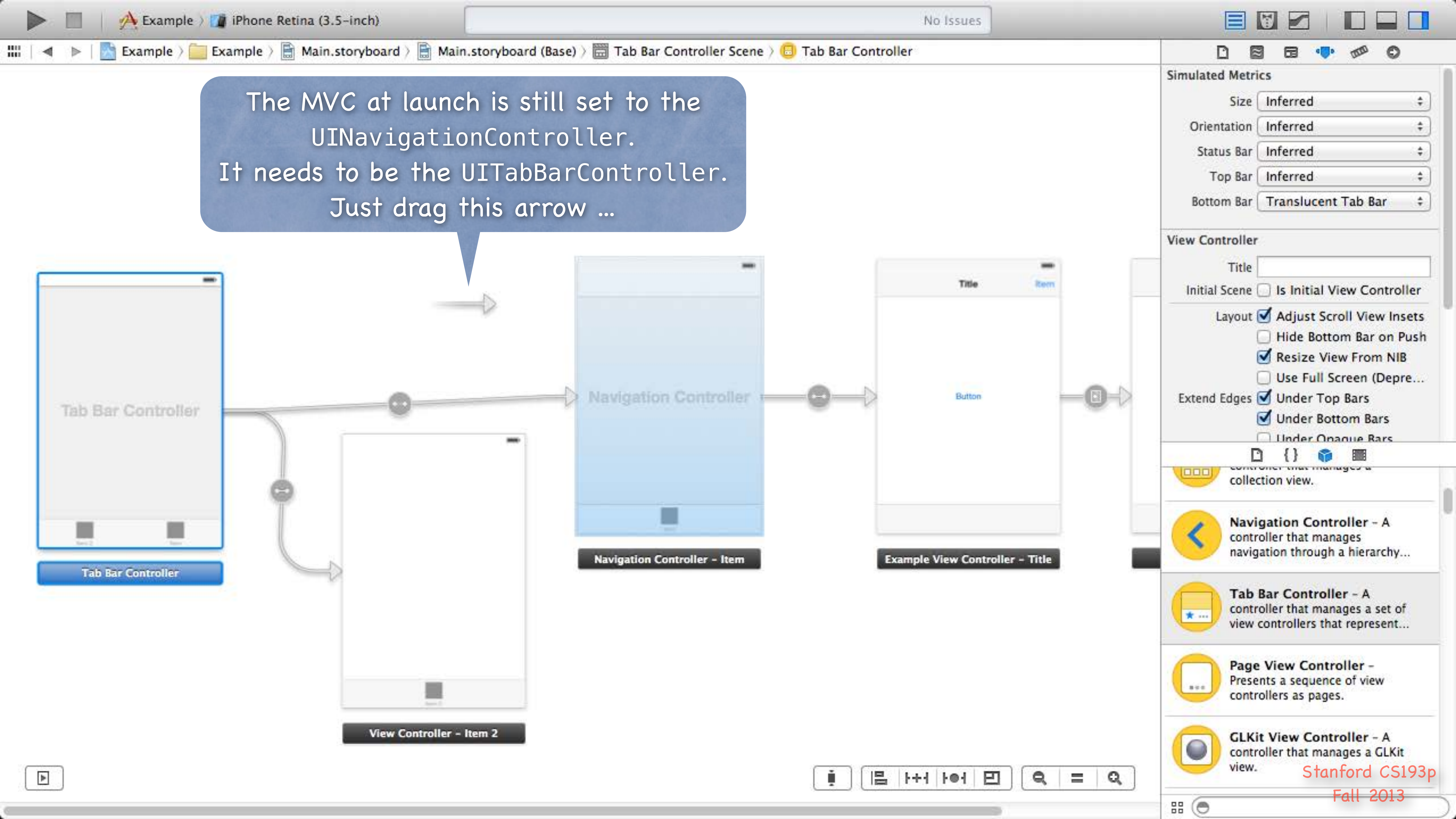
Navigation Controller – A controller that manages navigation through a hierarchy…

Tab Bar Controller – A controller that manages a set of …ollers that represent…

…Controller – …equence of view …s pages.

…Controller – A …hat manages a GLKit

Here we have
UINavigationController INSIDE a
UITabBarController.
Perfectly legal (the opposite is not).



The MVC at launch is still set to the `UINavigationController`.
It needs to be the `UITabBarController`.
Just drag this arrow …

Simulated Metrics

Size: Inferred
Orientation: Inferred
Status Bar: Inferred
Top Bar: Inferred
Bottom Bar: Translucent Tab Bar

View Controller

Title:
Initial Scene: □ Is Initial View Controller
Layout: ☑ Adjust Scroll View Insets
□ Hide Bottom Bar on Push
☑ Resize View From NIB
□ Use Full Screen (Depre…
Extend Edges: ☑ Under Top Bars
☑ Under Bottom Bars
□ Under Opaque Bars

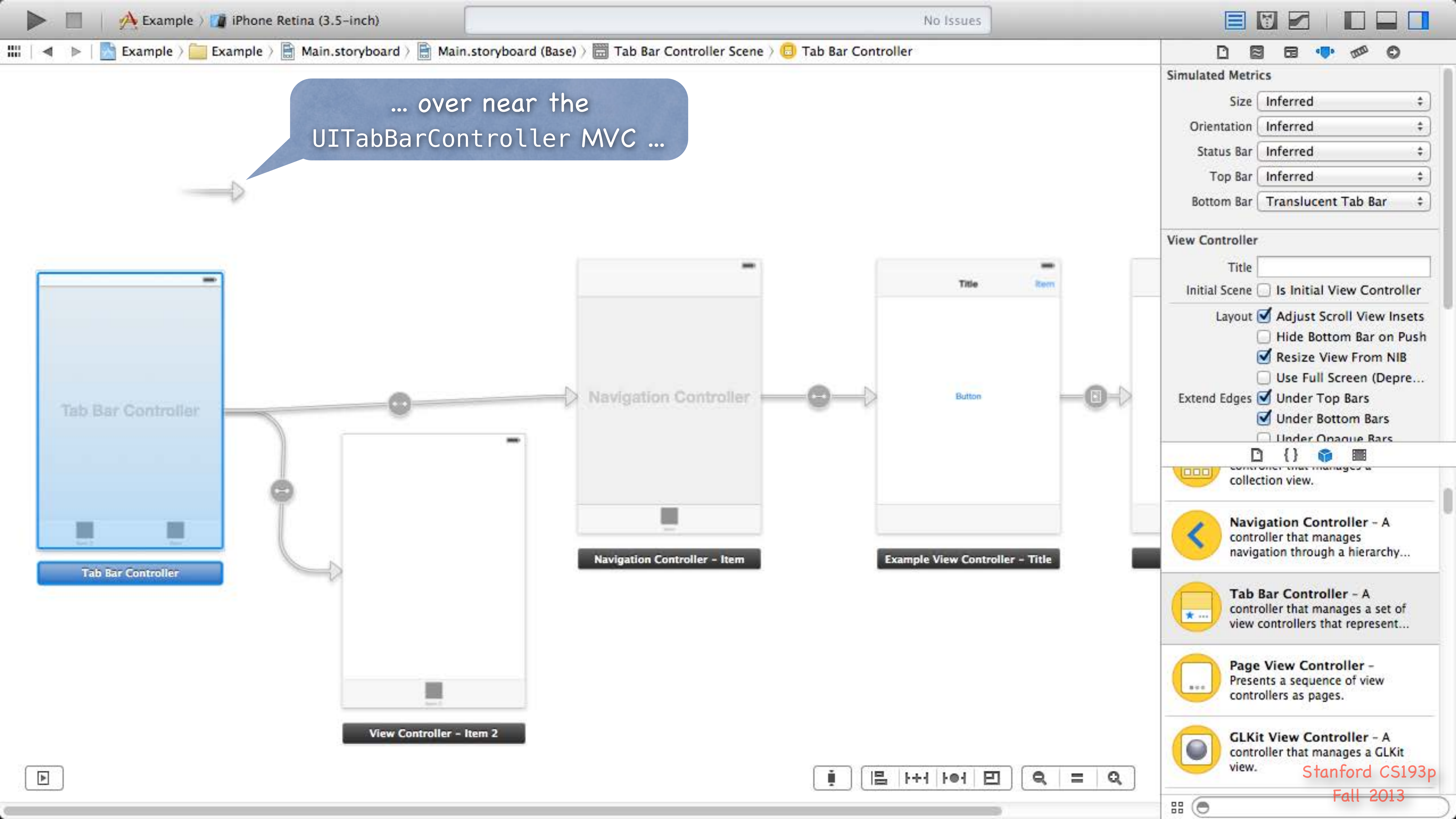
collection view.

Navigation Controller - A controller that manages navigation through a hierarchy…

Tab Bar Controller - A controller that manages a set of view controllers that represent…

Page View Controller - Presents a sequence of view controllers as pages.

GLKit View Controller - A controller that manages a GLKit view.



… over near the UITabBarController MVC …

Simulated Metrics

- Size: Inferred
- Orientation: Inferred
- Status Bar: Inferred
- Top Bar: Inferred
- Bottom Bar: Translucent Tab Bar

View Controller

- Title:
- Initial Scene: □ Is Initial View Controller
- Layout: ☑ Adjust Scroll View Insets
  - □ Hide Bottom Bar on Push
  - ☑ Resize View From NIB
  - □ Use Full Screen (Depre…
- Extend Edges: ☑ Under Top Bars
  - ☑ Under Bottom Bars
  - □ Under Opaque Bars

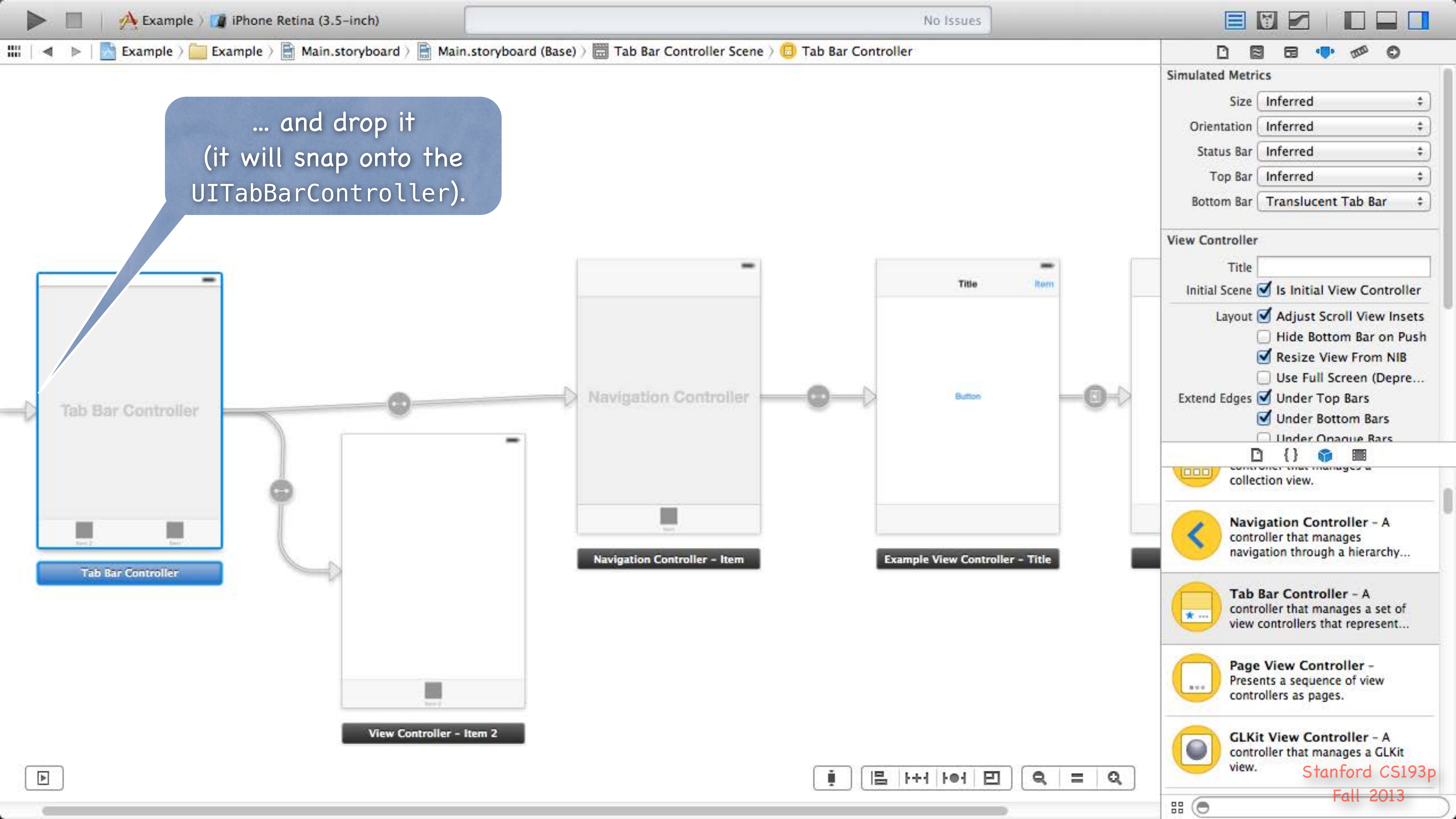
collection view.

Navigation Controller – A controller that manages navigation through a hierarchy…

Tab Bar Controller – A controller that manages a set of view controllers that represent…

Page View Controller – Presents a sequence of view controllers as pages.

GLKit View Controller – A controller that manages a GLKit view.



… and drop it
(it will snap onto the
UITabBarController).

Simulated Metrics

Size Inferred

Orientation Inferred

Status Bar Inferred

Top Bar Inferred

Bottom Bar Translucent Tab Bar

View Controller

Title

Initial Scene ☑ Is Initial View Controller

Layout ☑ Adjust Scroll View Insets
□ Hide Bottom Bar on Push
☑ Resize View From NIB
□ Use Full Screen (Depre…

Extend Edges ☑ Under Top Bars
☑ Under Bottom Bars
□ Under Opaque Bars

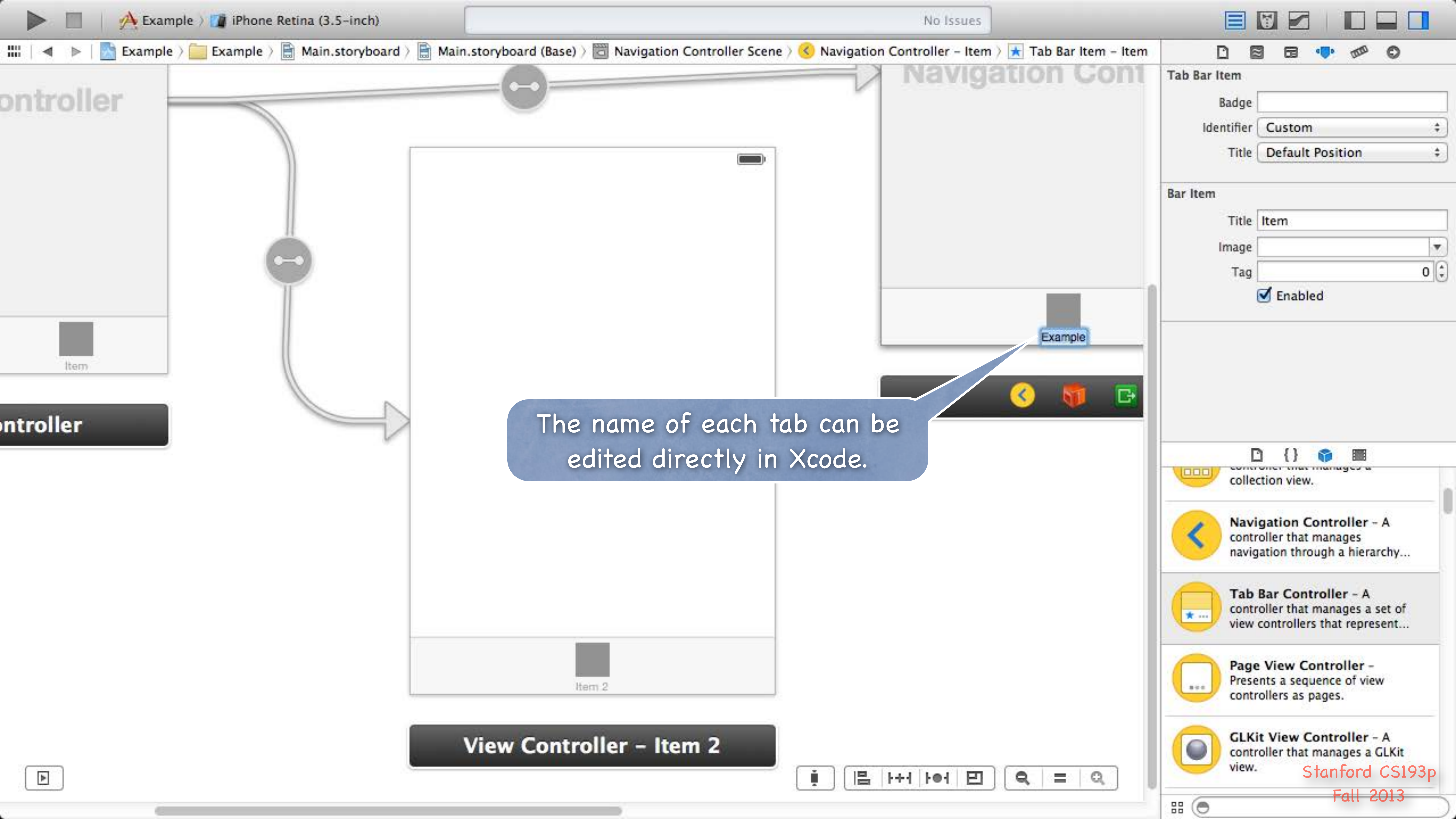
collection view.

Navigation Controller – A controller that manages navigation through a hierarchy…

Tab Bar Controller – A controller that manages a set of view controllers that represent…

Page View Controller – Presents a sequence of view controllers as pages.

GLKit View Controller – A controller that manages a GLKit view.


Example
iPhone Retina (3.5-inch)
No Issues
Example 〉 Example 〉 Main.storyboard 〉 Main.storyboard (Base) 〉 Navigation Controller Scene 〉 Navigation Controller – Item 〉 Tab Bar Item – Item
ontroller
Item
ntroller
Navigation Cont
Example
Item 2
View Controller – Item 2

The name of each tab can be edited directly in Xcode.

Tab Bar Item

Badge

Identifier: Custom

Title: Default Position

Bar Item

Title: Item

Image

Tag: 0

☑ Enabled

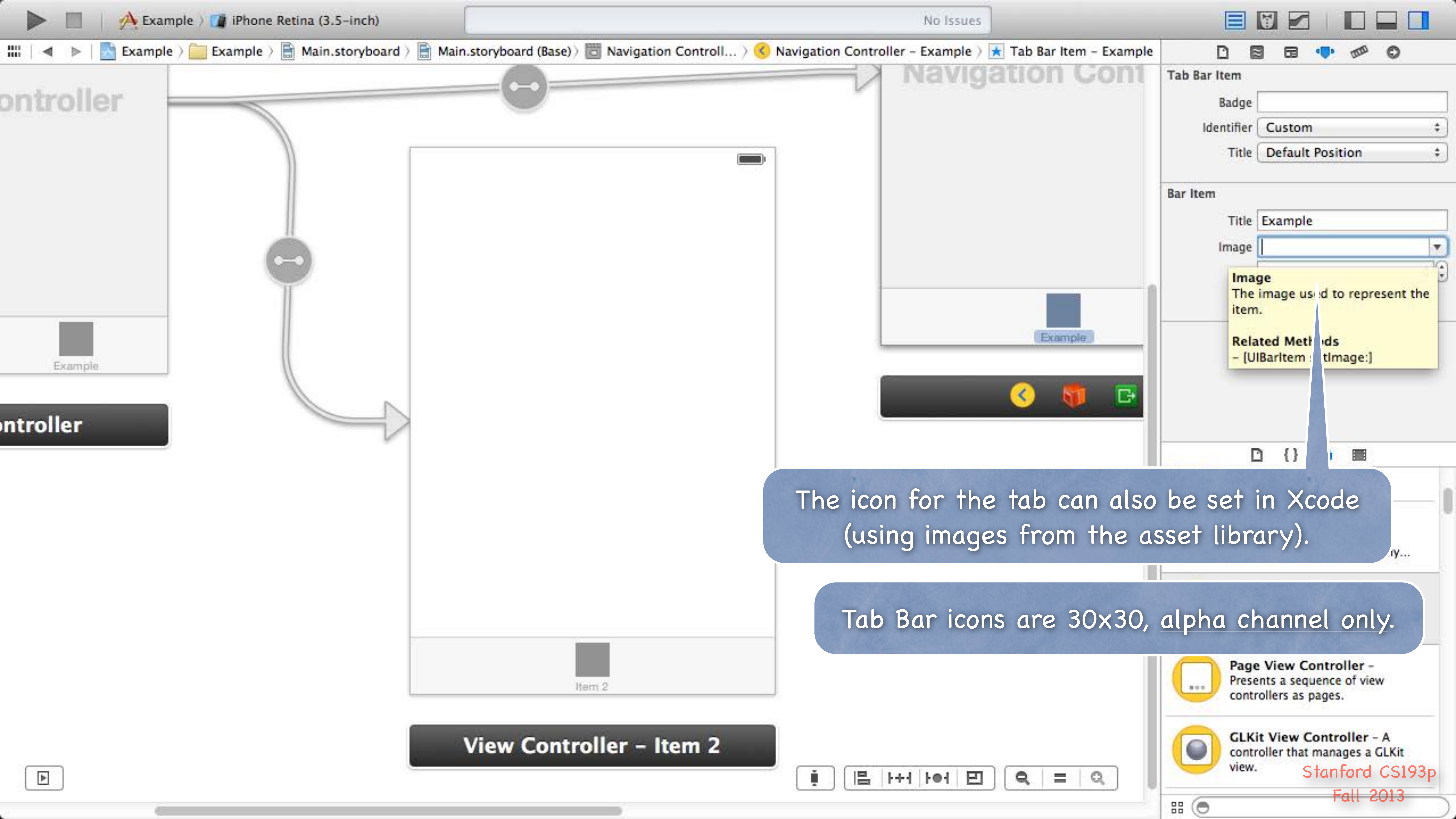
collection view.

Navigation Controller – A controller that manages navigation through a hierarchy...

Tab Bar Controller – A controller that manages a set of view controllers that represent...

Page View Controller – Presents a sequence of view controllers as pages.

GLKit View Controller – A controller that manages a GLKit view.



Tab Bar Item

Badge:

Identifier: Custom

Title: Default Position

Bar Item

Title: Example

Image:

The icon for the tab can also be set in Xcode (using images from the asset library).

Tab Bar icons are 30x30, alpha channel only.

Coming Up

- Friday

  No Section

- Next couple of weeks ...

  Drawing in your own custom View class

  Gestures

  Autolayout

  Animation

No Lecture next Monday!