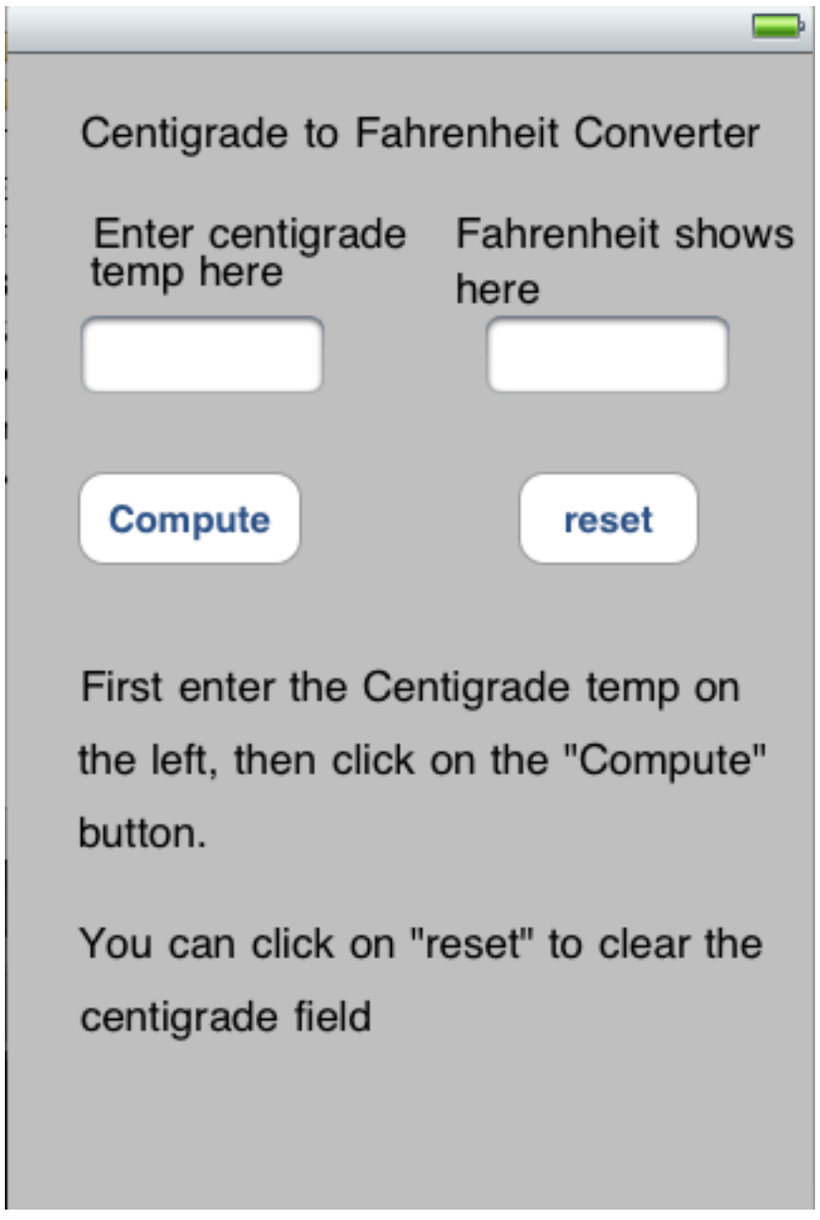
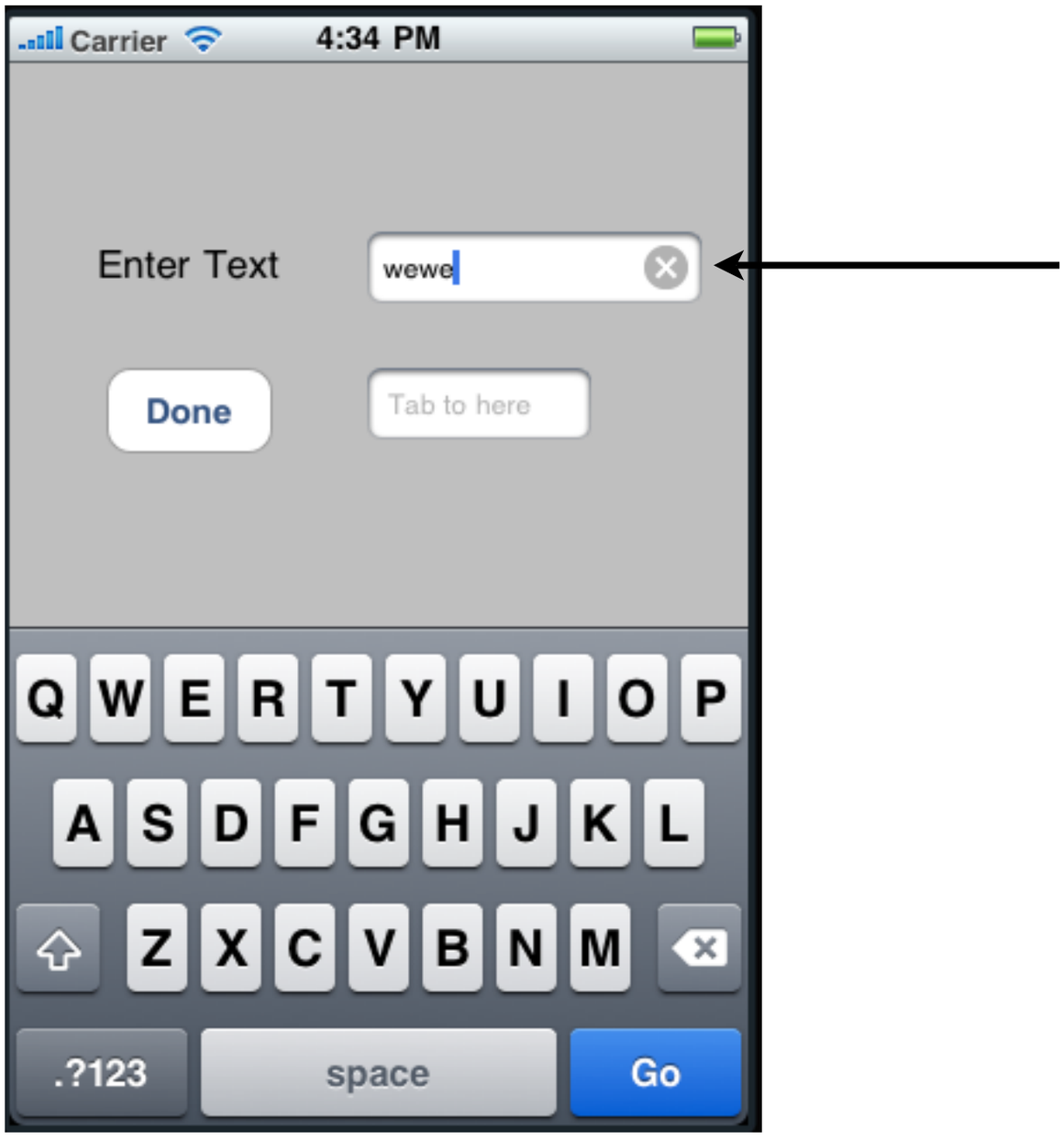
CS 696 Mobile Application Development
Fall Semester, 2010
Doc 9 Multi-View Apps
Sep 28, 2010

Assignment 2

NO

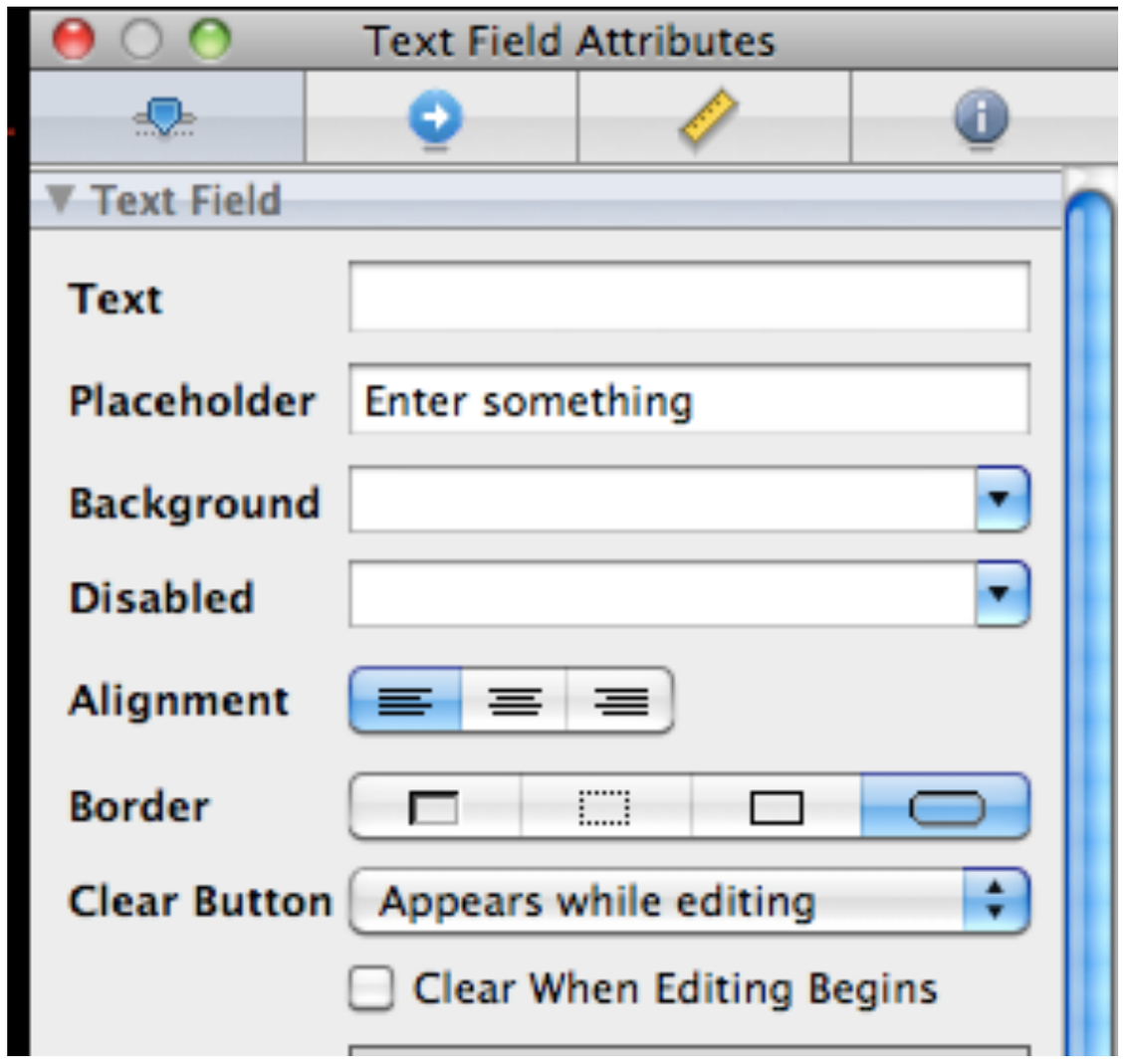

Centigrade to Fahrenheit Converter
Enter centigrade temp here
Fahrenheit shows here
Compute
reset
First enter the Centigrade temp on the left, then click on the "Compute" button.
You can click on "reset" to clear the centigrade field

Clearing a field



Clearing a field = `UITextFieldViewModeWhileEditing;`

```
sampleField.rightViewMode = UITextFieldViewModeWhileEditing;
```



TextField Delegate

In controller

```
- (void)viewDidLoad {
    [super viewDidLoad];
      sampleField.delegate = self;
}
```

TextField Delegate Methods

- (void)textFieldDidEndEditing: (UITextField *) textField
- (void)textFieldDidBeginEditing: (UITextField *) textField

- (BOOL) textFieldShouldBeginEditing: (UITextField *) textField
- (BOOL) textFieldShouldClear: (UITextField *) textField
- (BOOL)textFieldShouldEndEditing: (UITextField *) textField
- (BOOL)textFieldShouldReturn: (UITextField *) textField

- (BOOL)textField:(UITextField *) textField
  shouldChangeCharactersInRange:(NSRange)range
  replacementString:(NSString *)string

textFieldShouldClear:

Called when user clicks on the clear icon

textField: shouldChangeCharactersInRange: replacementString:

Called when user changes text in field

Return NO

Change not allowed

Return YES

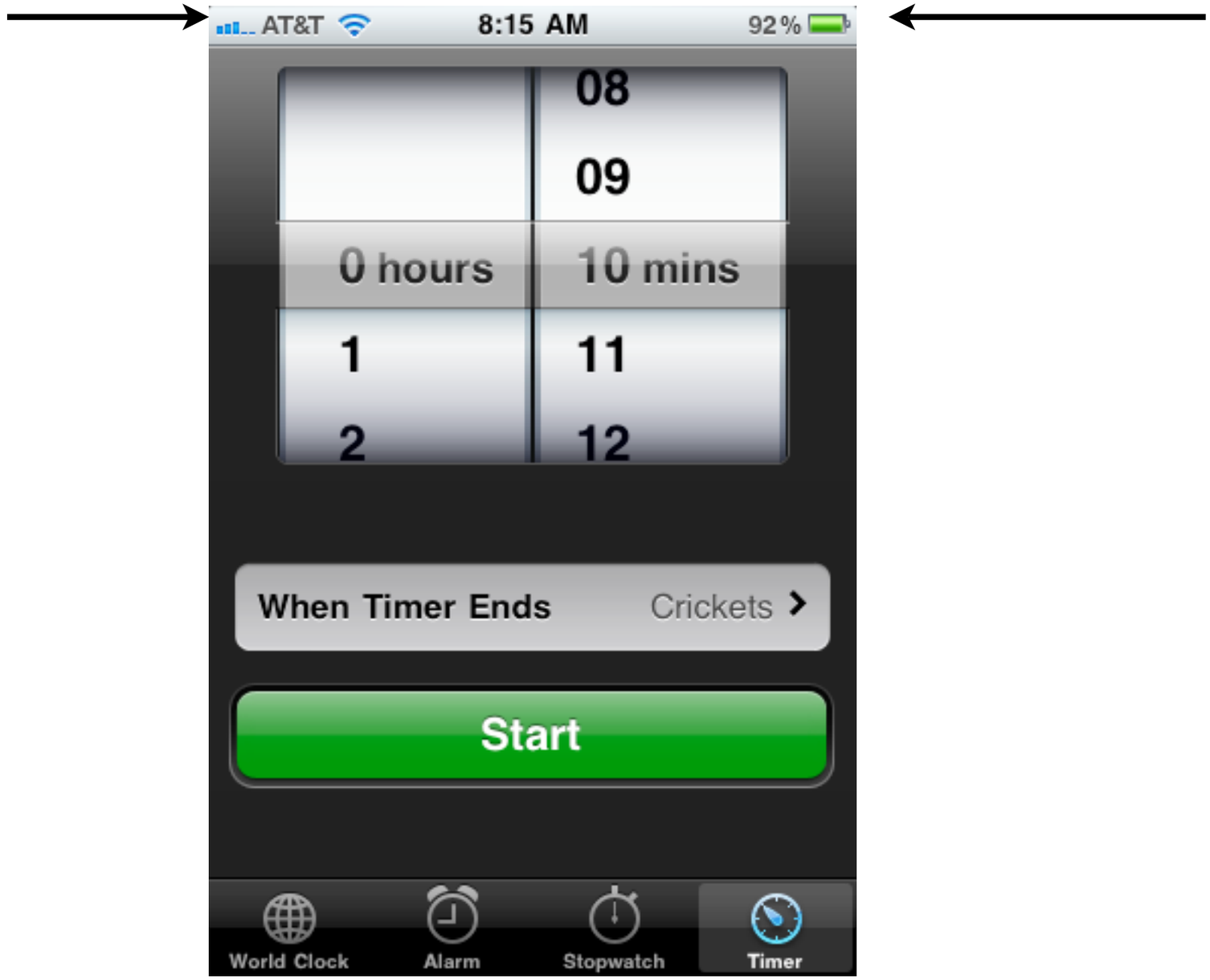
Change allowed

iPhone Bars
Navigation, Tool, Tab, Status

Status Bar

gray (default)
opaque black
translucent black

20px



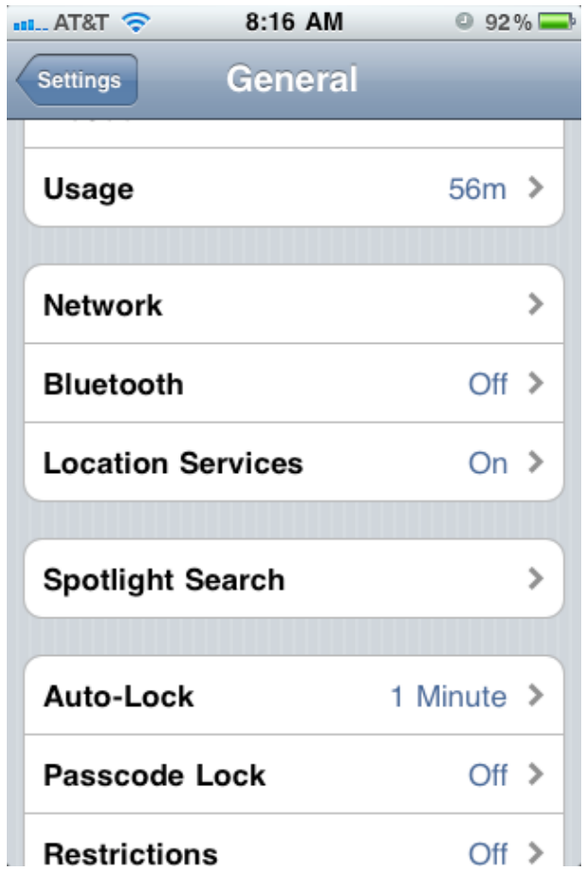
Navigation Bars

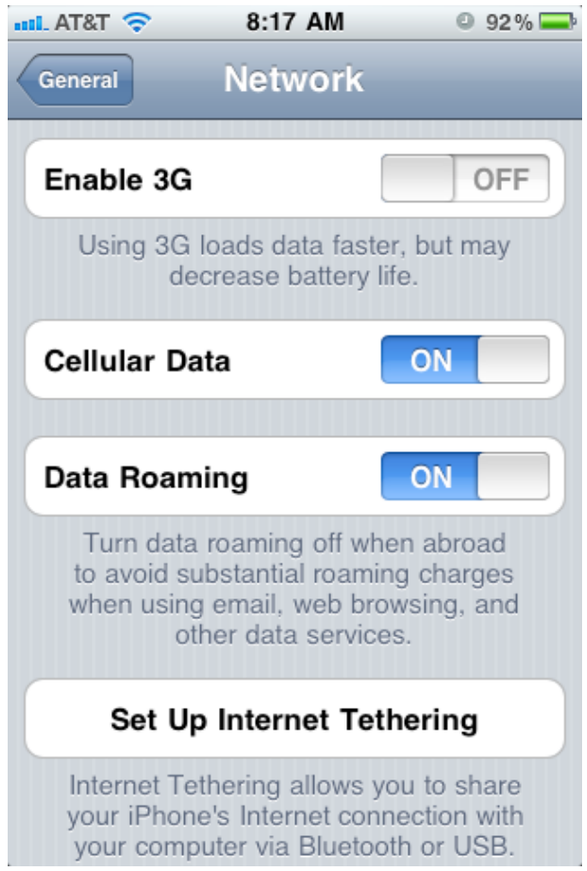
Navigation among different views

Provide controls that manage items in a view

Drill down hierarchy of views

Navigation Bars - Navigation

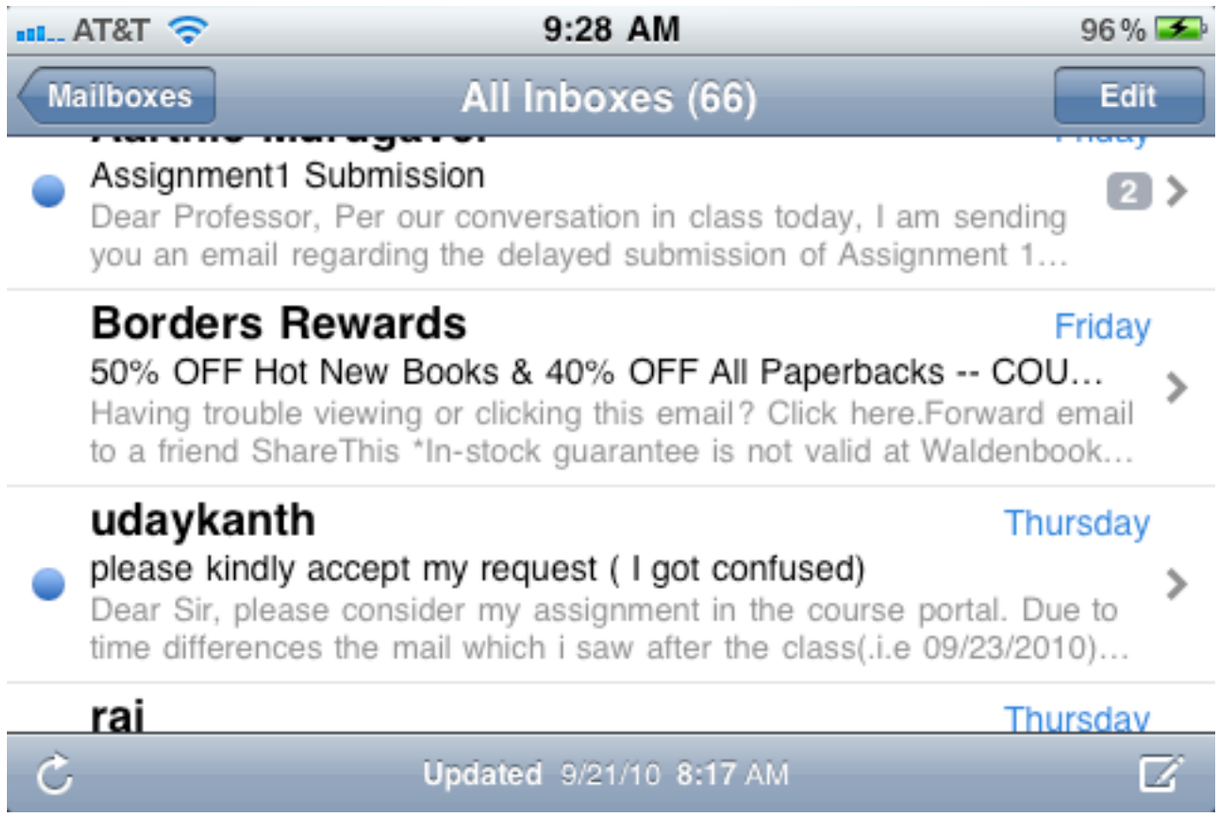


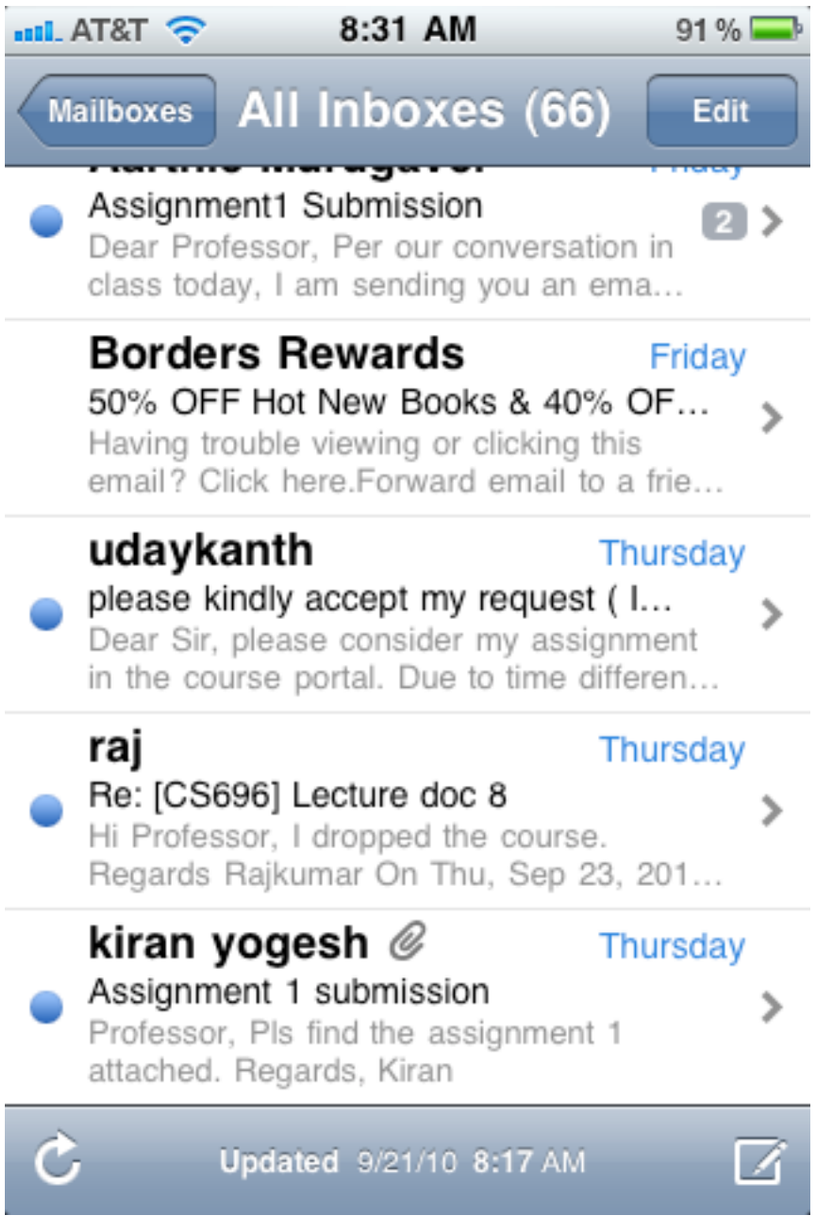


Drilling down

Notice the navigation bar has the title of each view

Navigation bar - different sizes

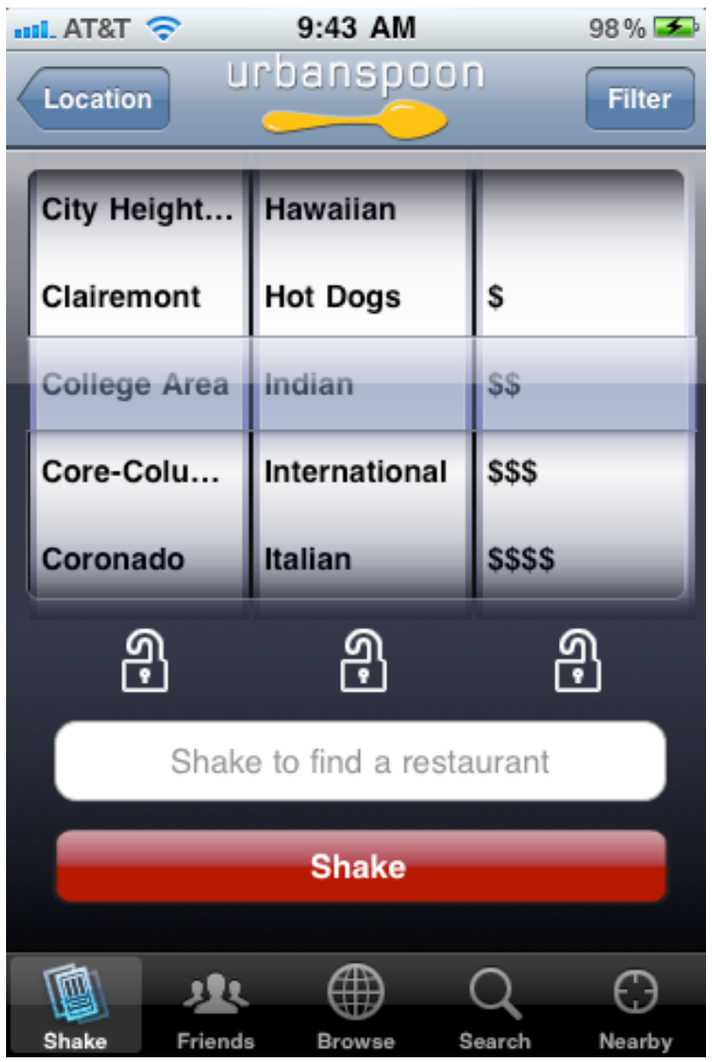


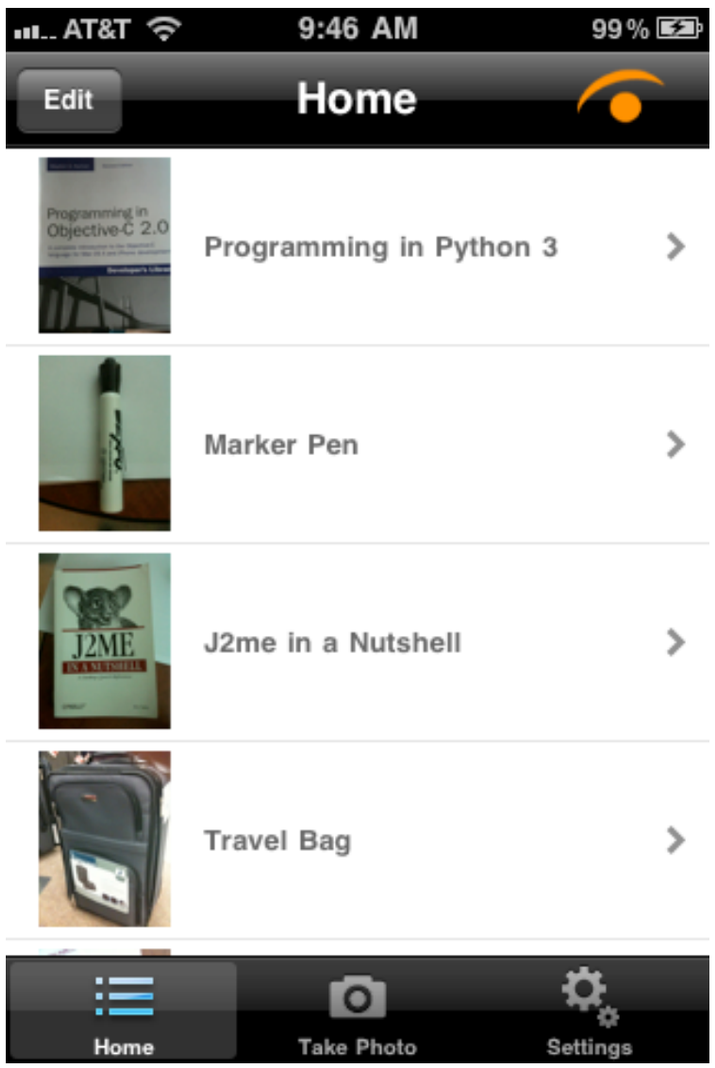


Navigation Bar Color

translucent
opaque

Blue (default)
Black
Any color



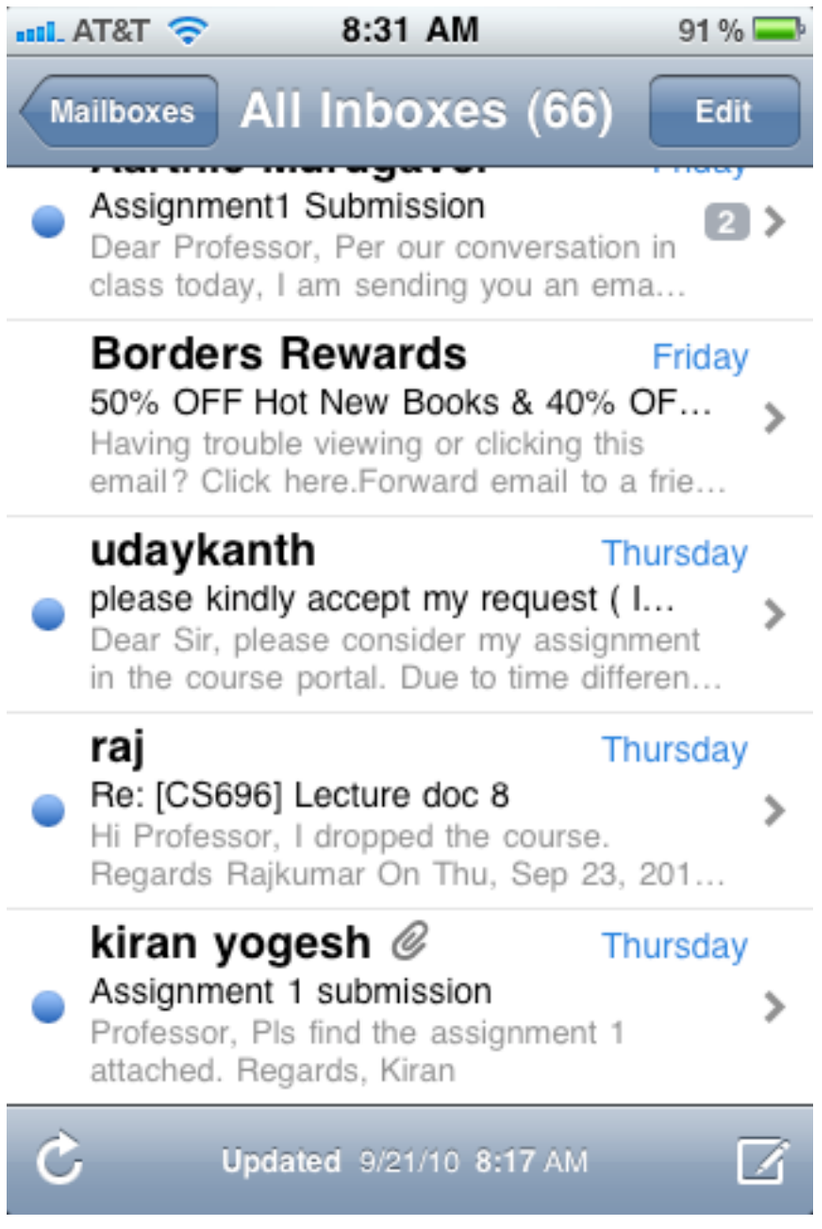


Toolbars

Contains buttons related to object in current view

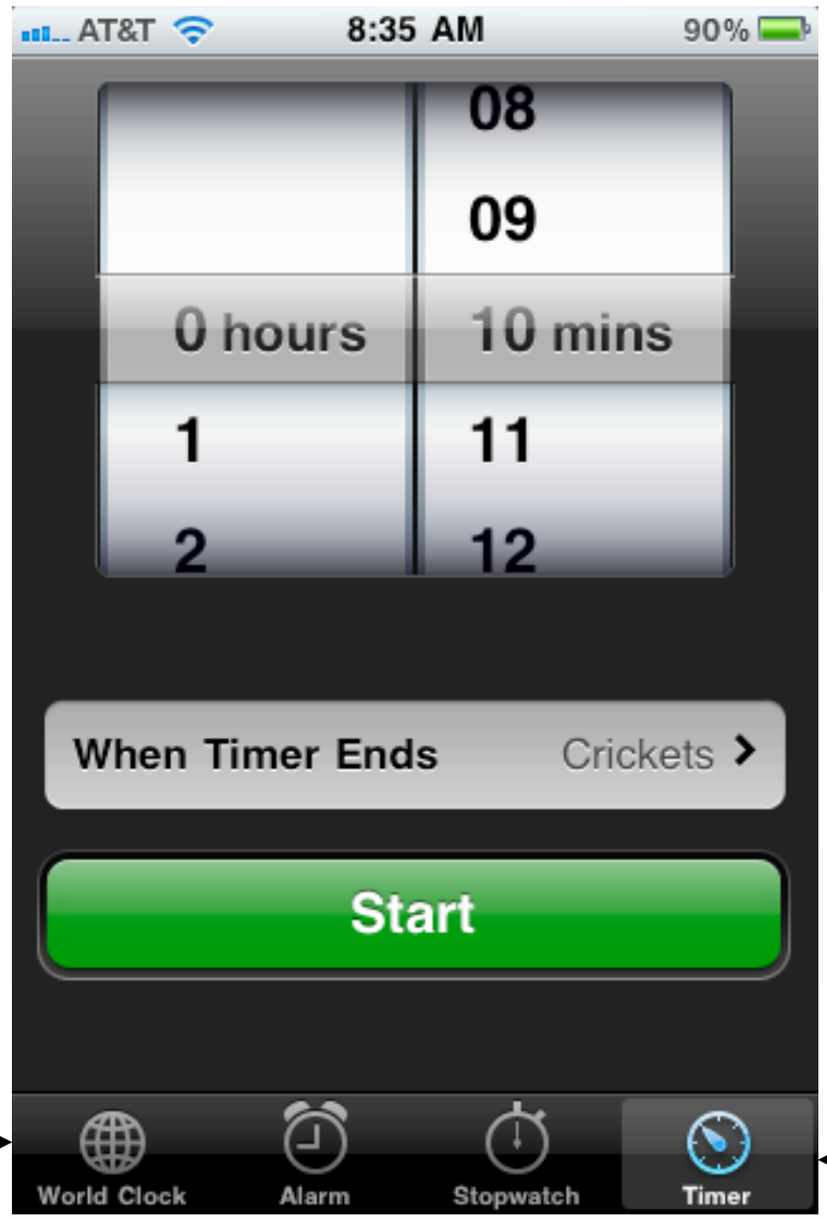
Blue (default)
Black
Any color

translucent
opaque



Tab Bars

black background
Tabs equally sized



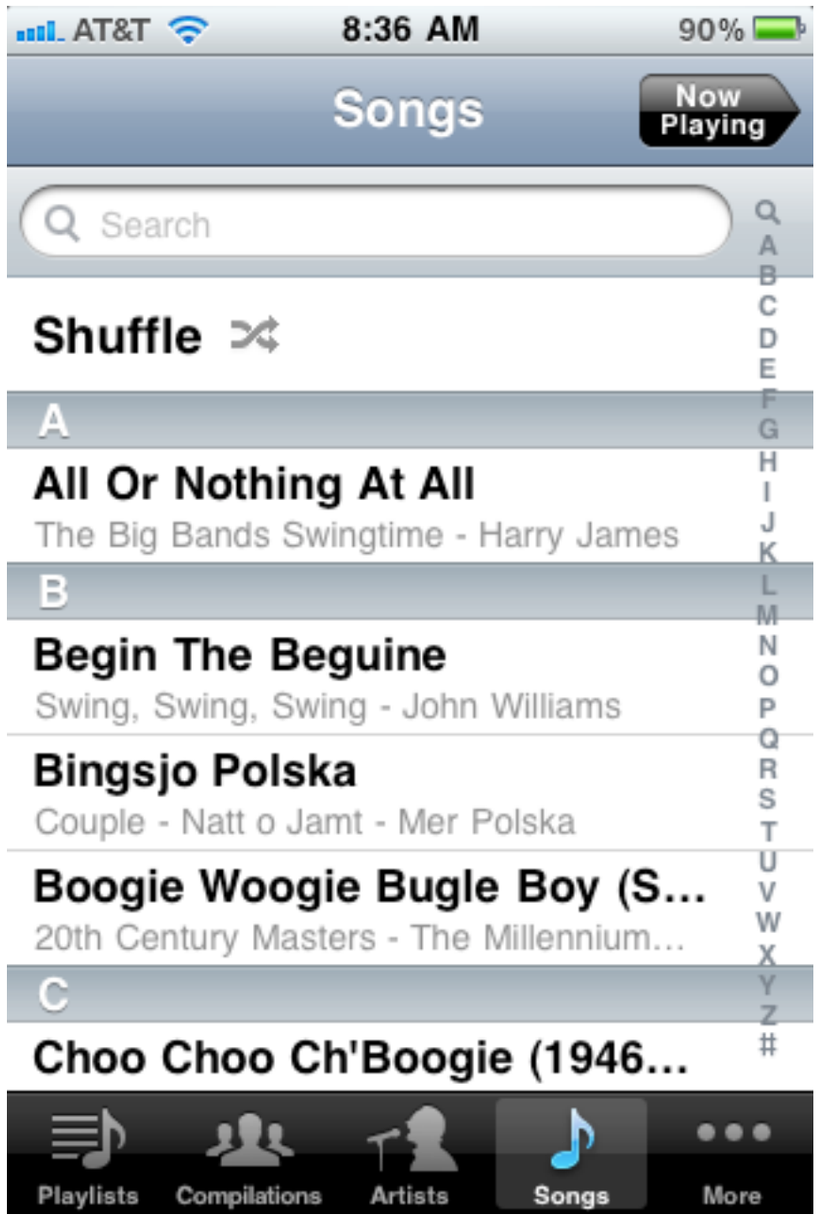
Different perspectives on same data

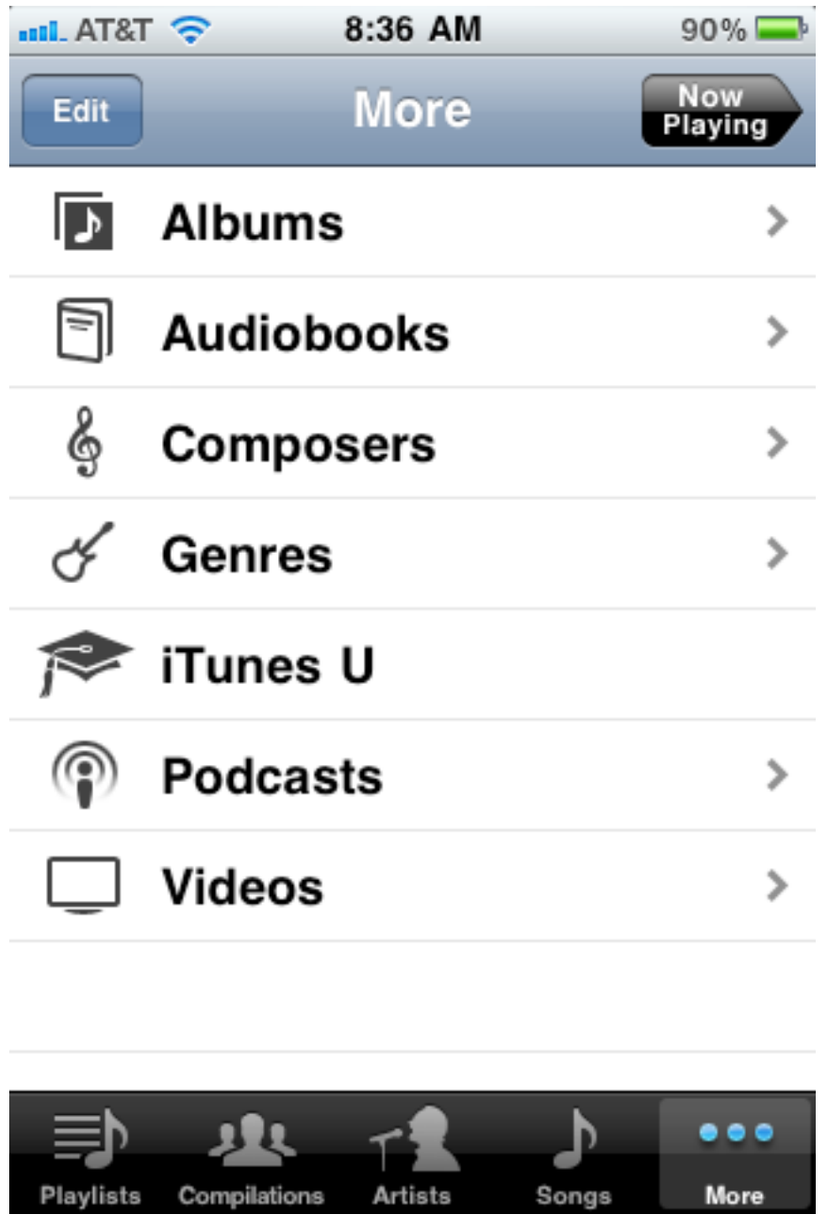
Different subtasks

Allows user to select between different mode of operation

Additional Tabs

Only shows 5 tabs



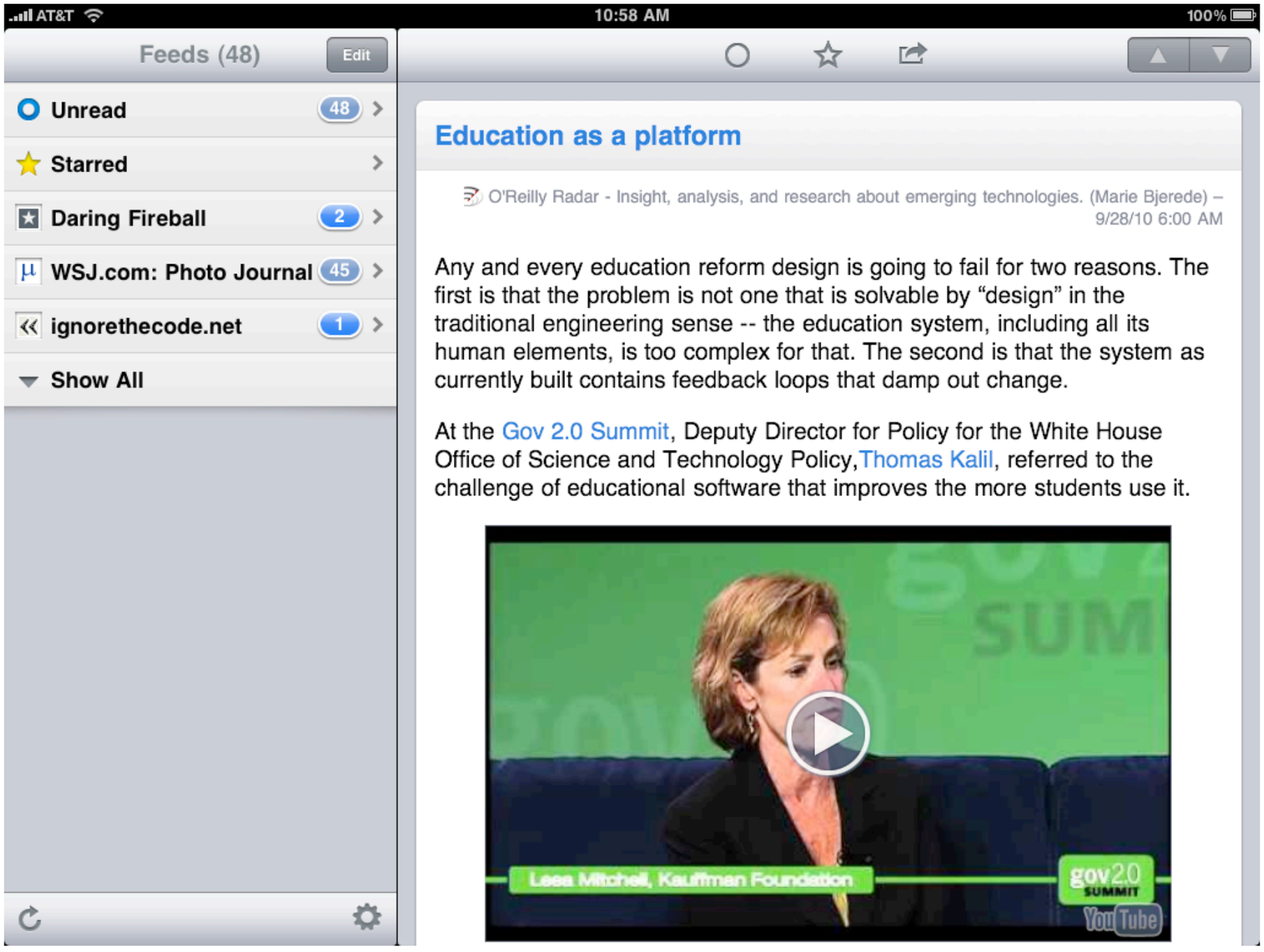


iPod app has 11 tabs

iPad
New Widgets
Bars

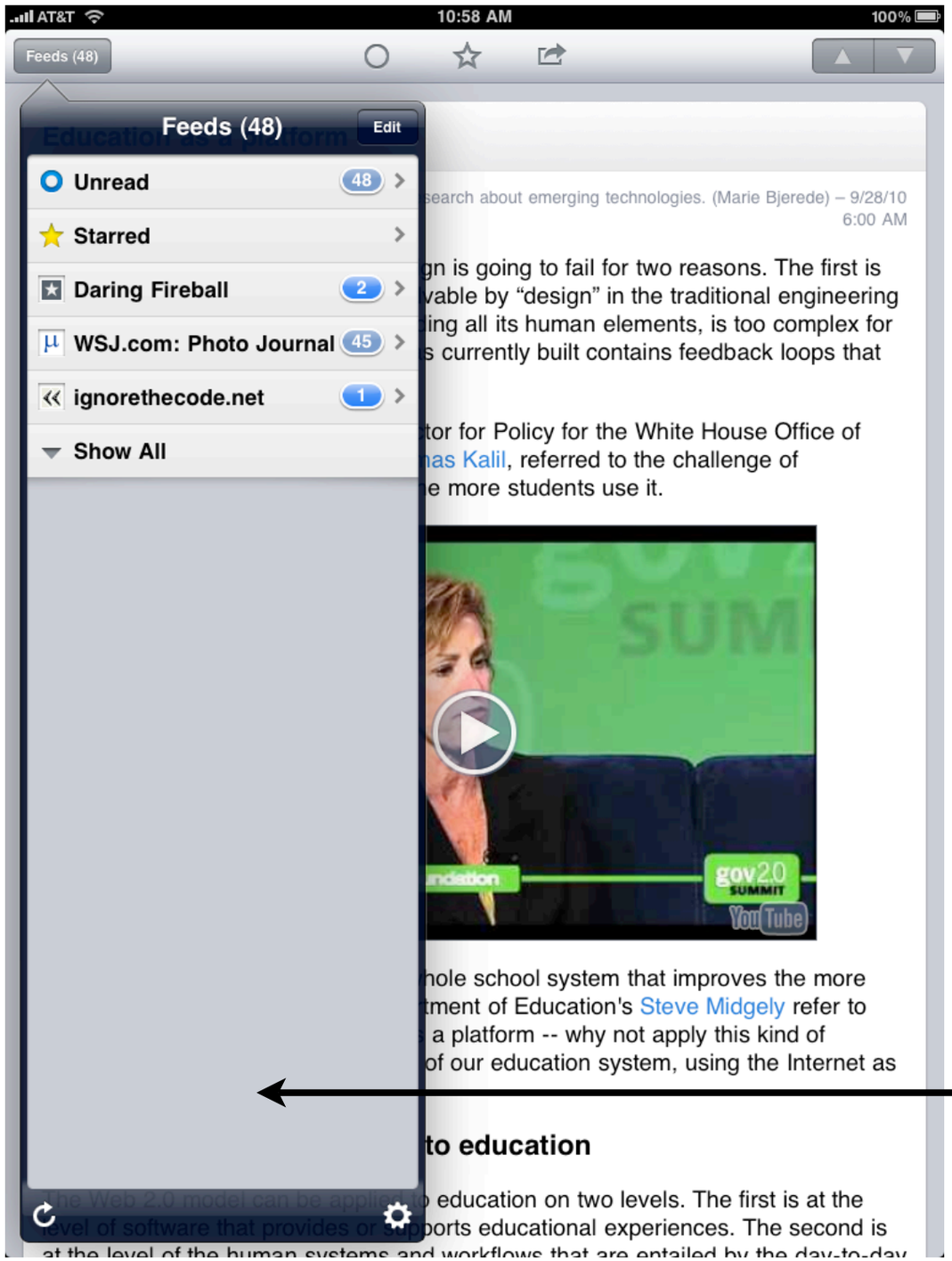
iPad - Split Screen

full screen view with two side-by-side panes

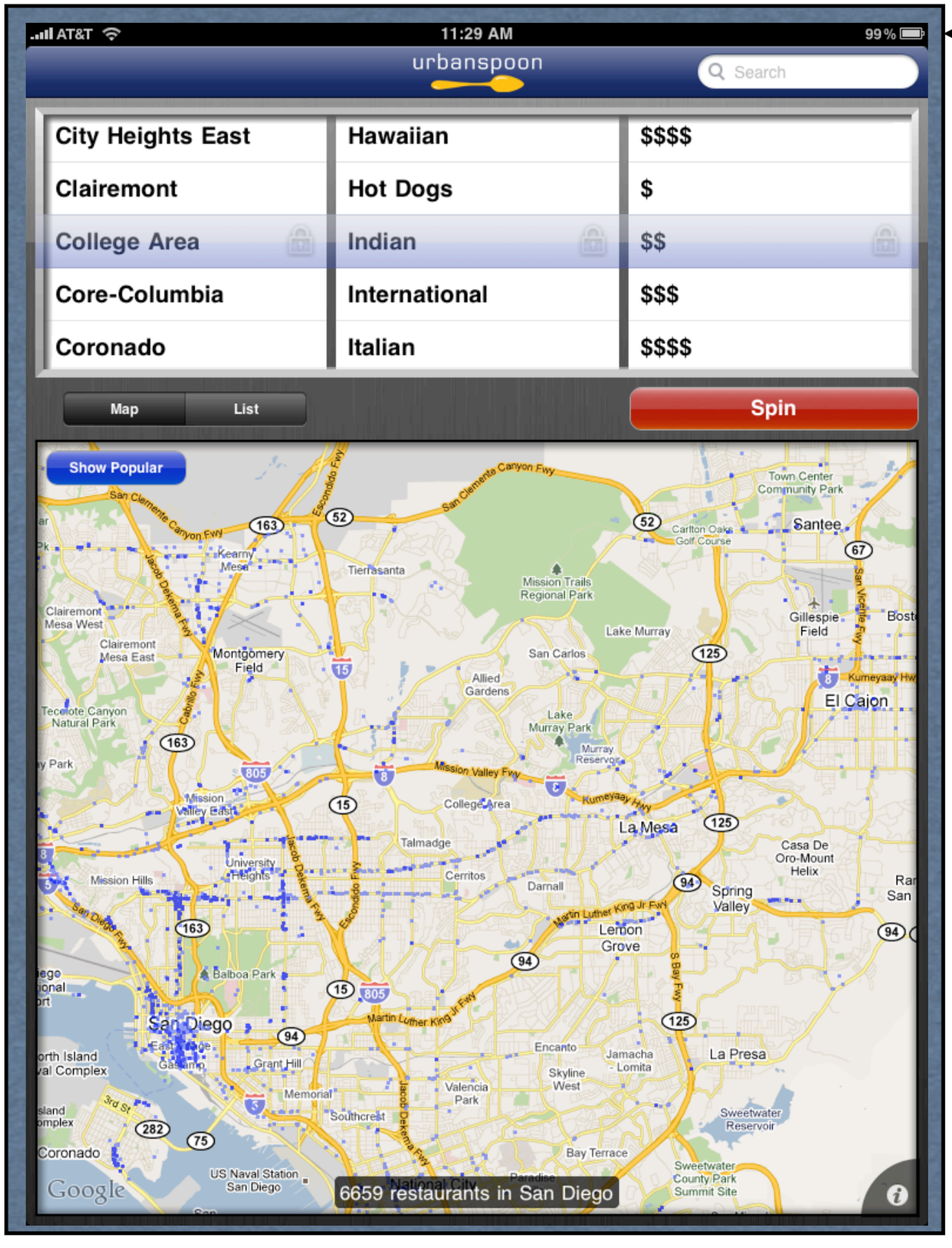


iPad - Popover

transient view displayed when people tab on control or screen



iPad - Status Bar



Unless game or full-screen media avoid hiding status bar

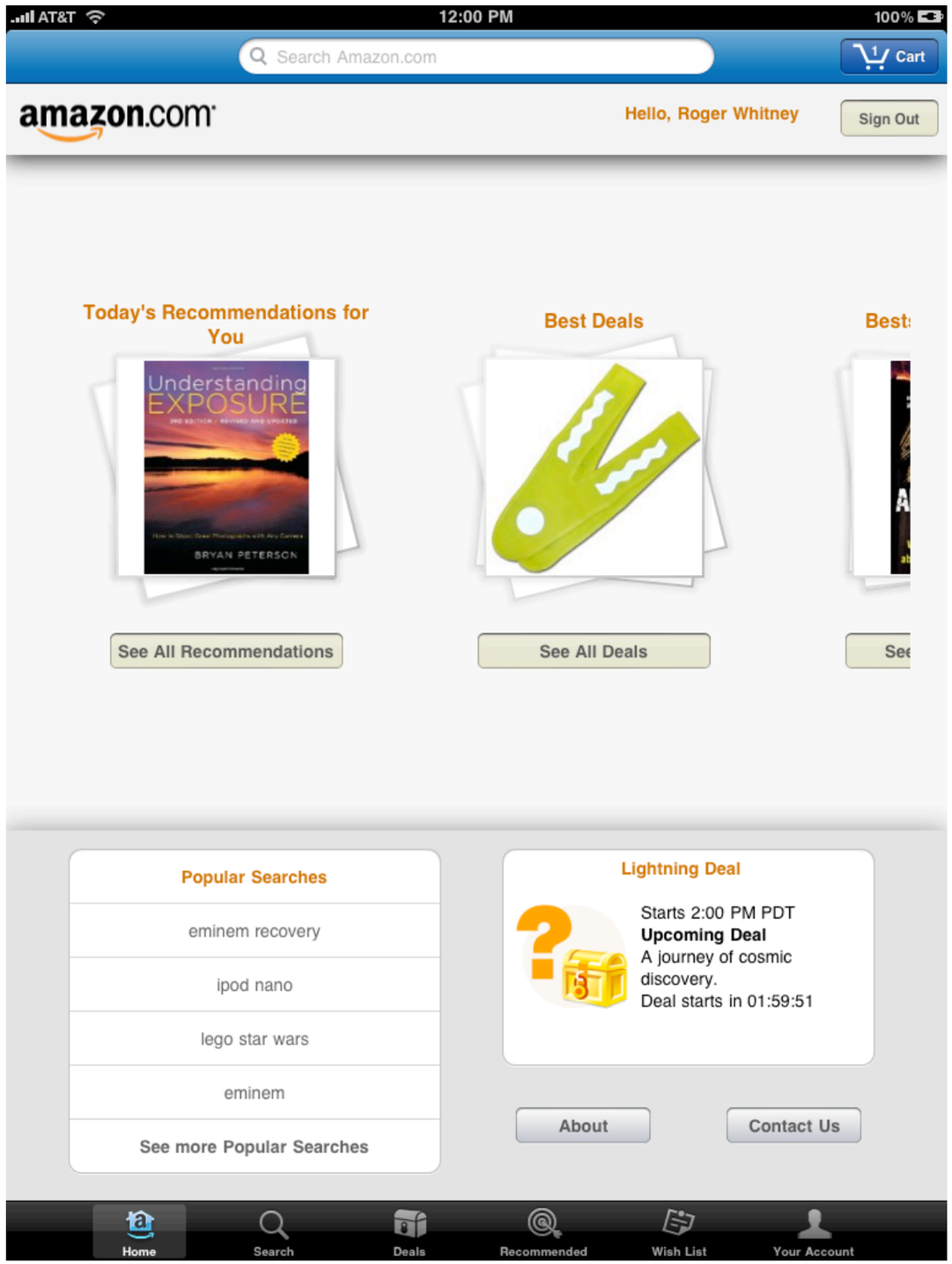
iPad - Navigation Bars

Not as common on iPad

Can be used in

spilt view
popover
full-screen view

iPad - Tab Bar



Good for top app view

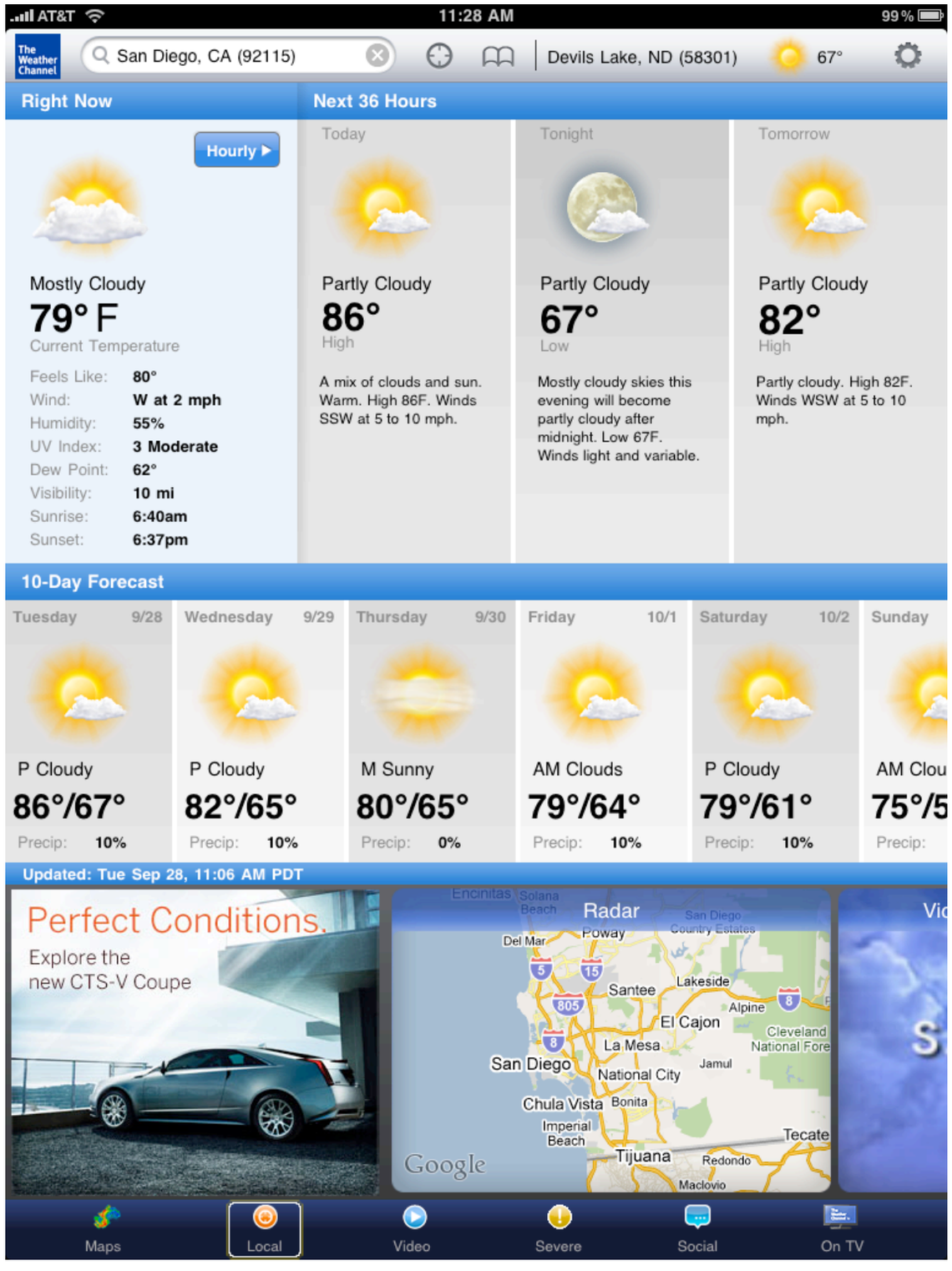
Avoid too many tabs

7 works well

Avoid "More tab"

Same tabs in each orientation

iPad - Toolbar



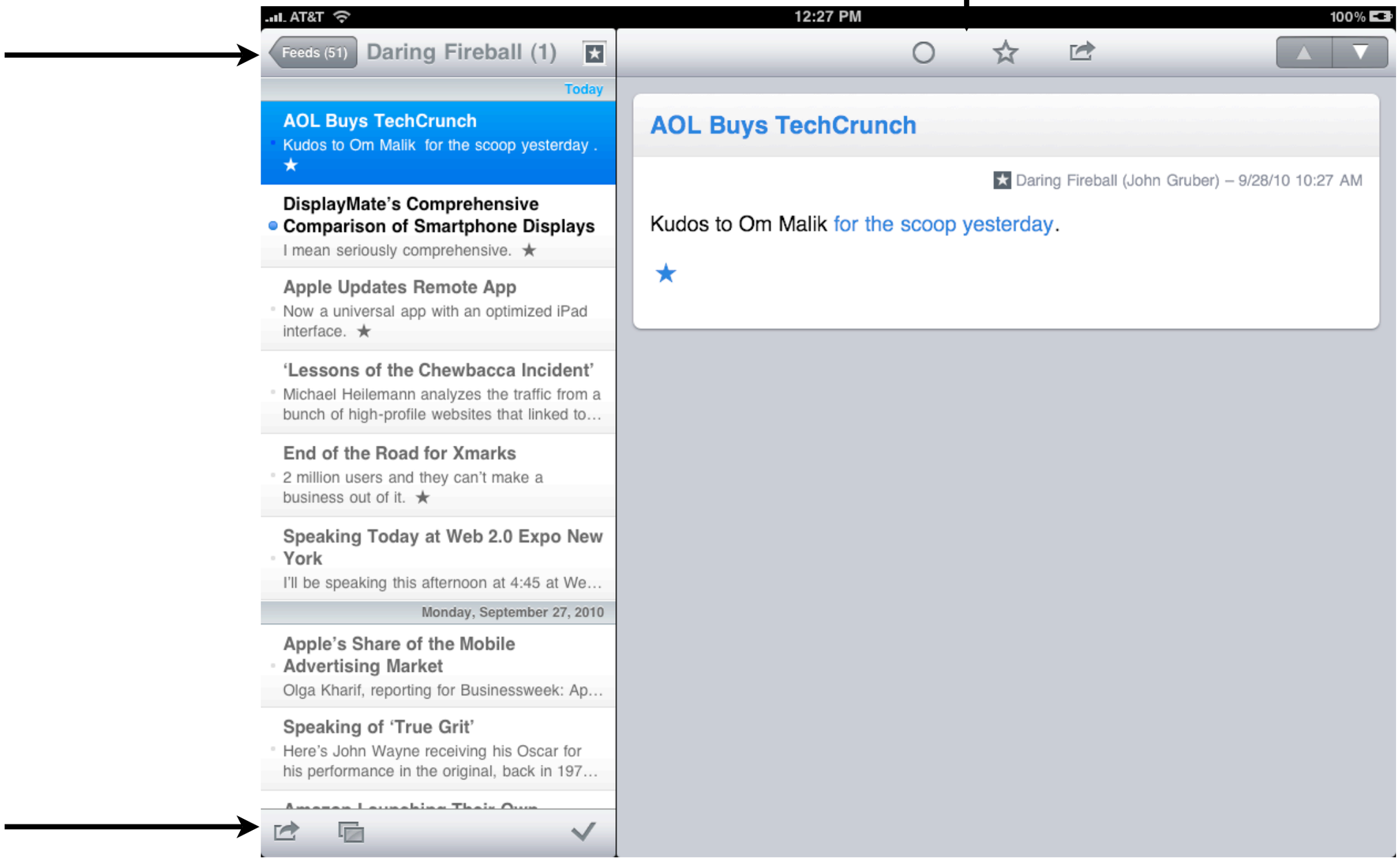
Usually on top of screen

Can contain segmented control

Each item 44 X 44 pixels or larger

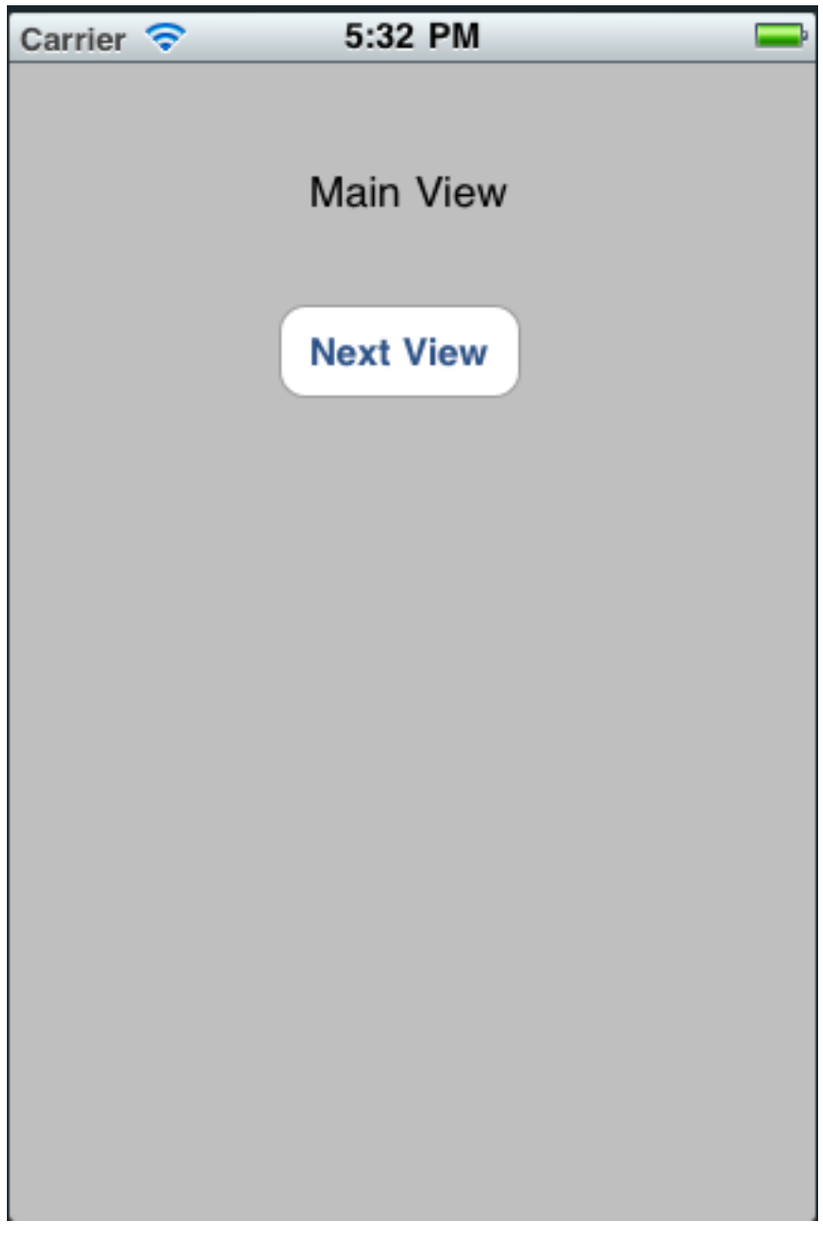
iPad - Popover & Bars

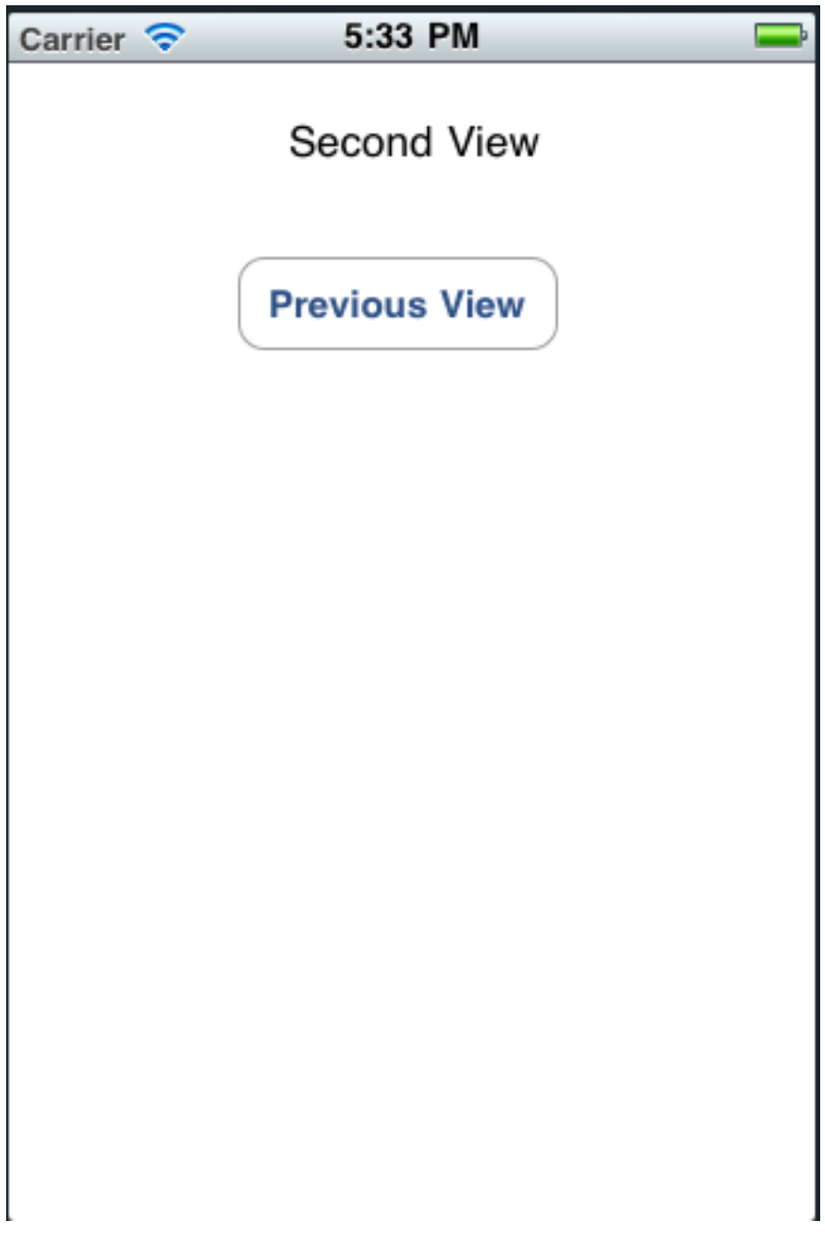
iPad - Split view & Bars



Code level
Multiple Views

Two Views, One Controller



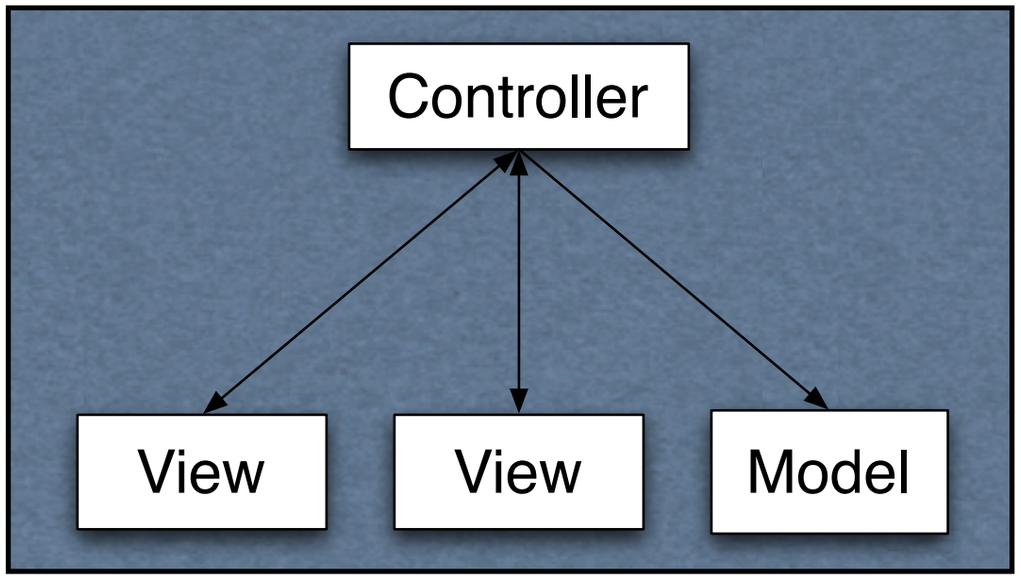


Basic Idea

UIViewController property view

Contains the view to be displayed

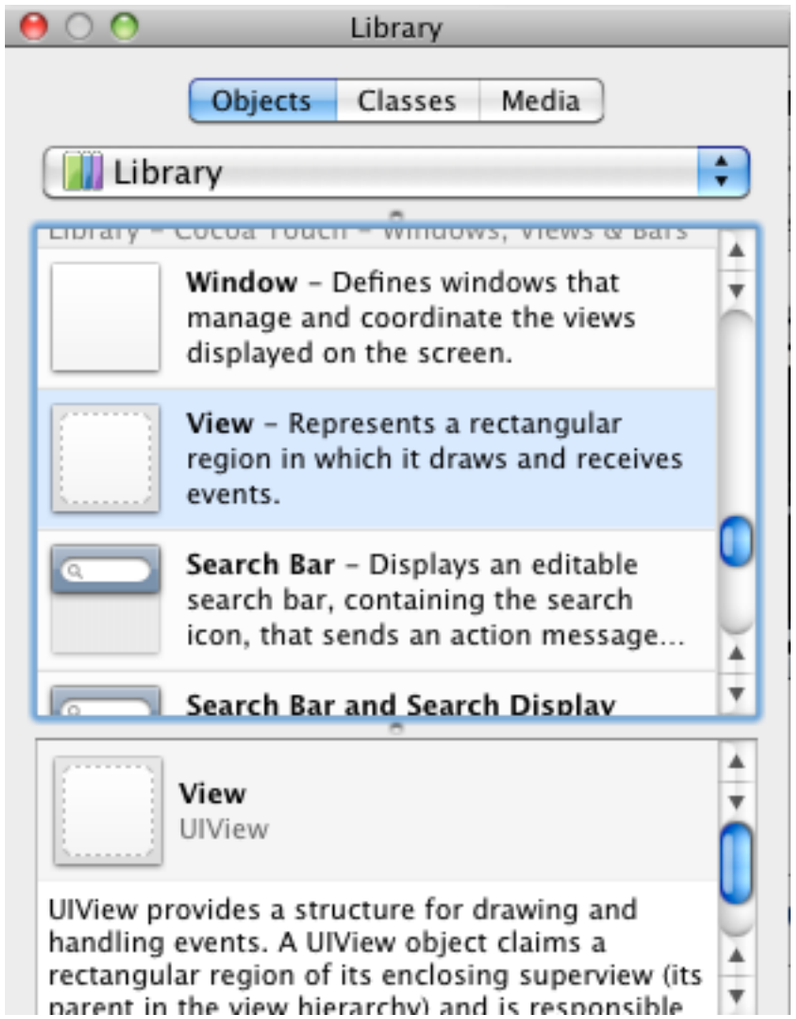
Change the view

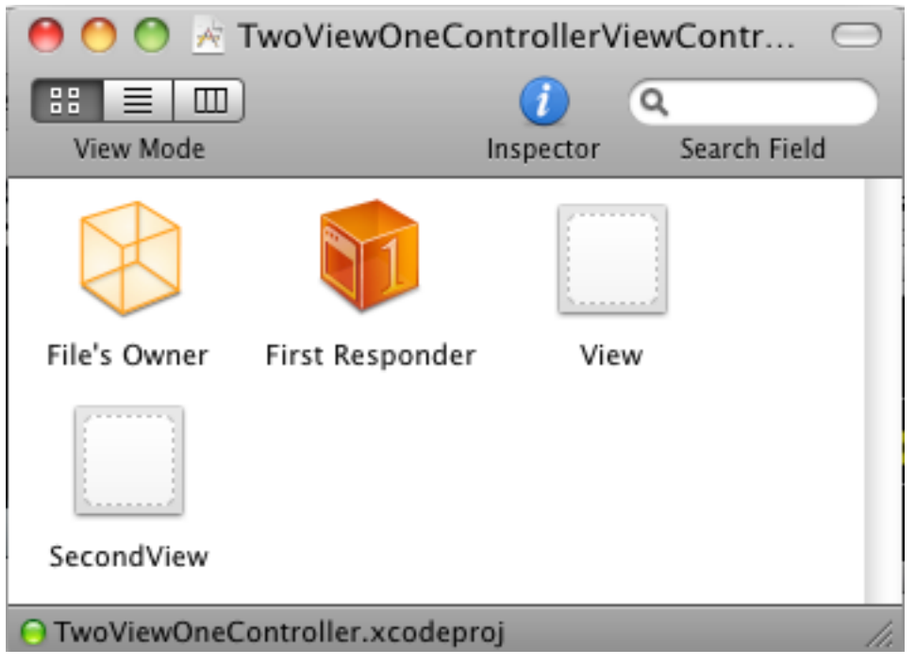


Outlets & Actions

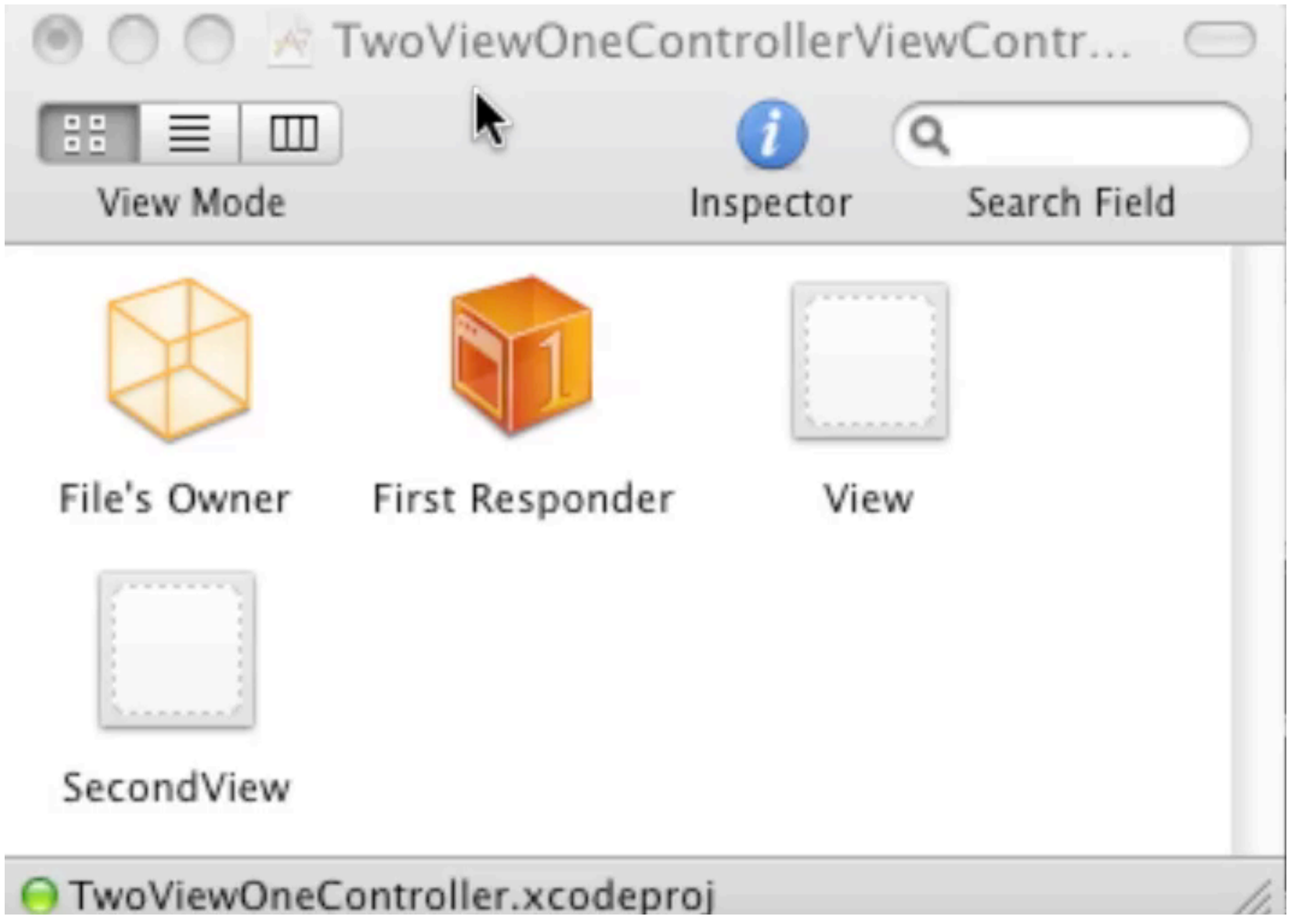
```
@interface TwoViewOneControllerViewController : UIViewController {
}

@property (nonatomic, retain) IBOutlet UIView * mainView;
@property (nonatomic, retain) IBOutlet UIView * secondView;

- (IBAction) next;
- (IBAction) previous;

@end
```

Adding a View

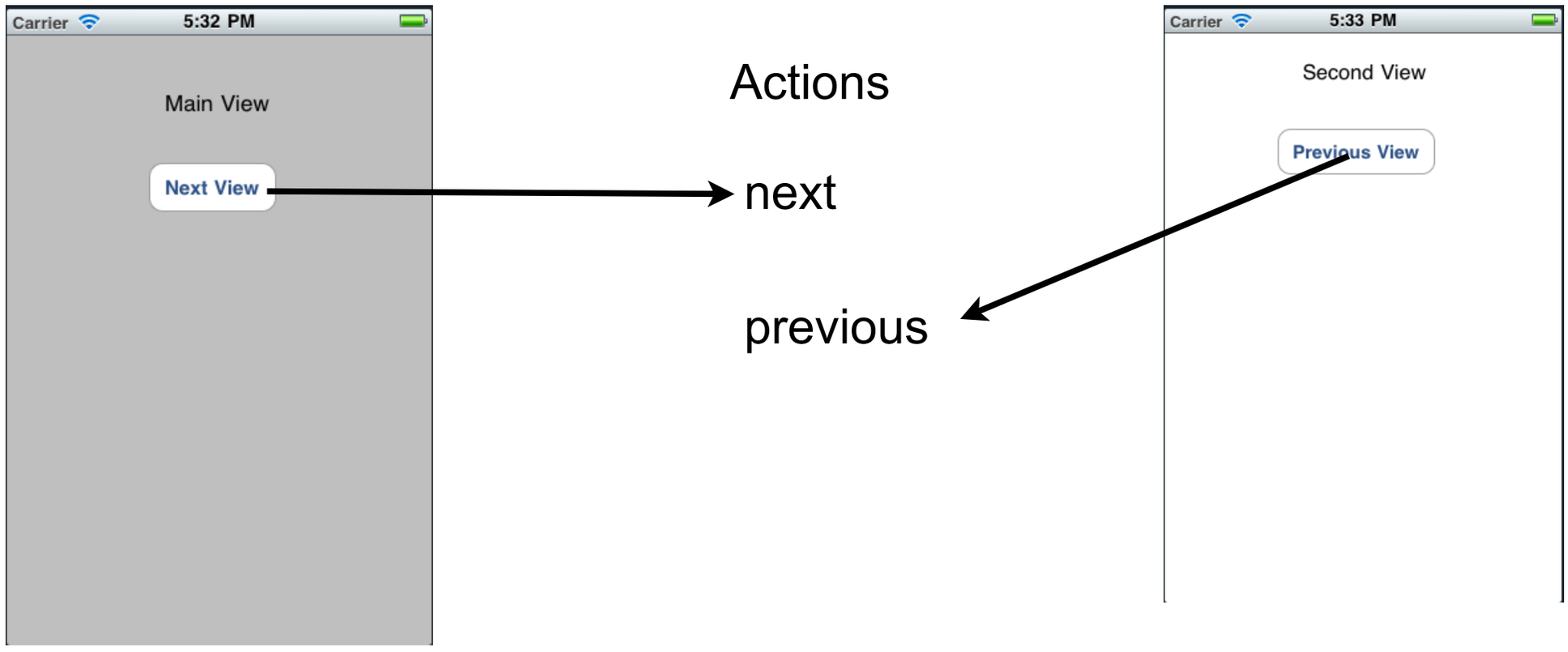




Connecting Views



Actions


Carrier
5:32 PM
Main View
Next View
Actions
next
previous
Carrier
5:33 PM
Second View
Previous View

Swapping Views

```
- (IBAction) next {
        self.view = self.secondView;
}

- (IBAction) previous {
        self.view = self.mainView;
}
```

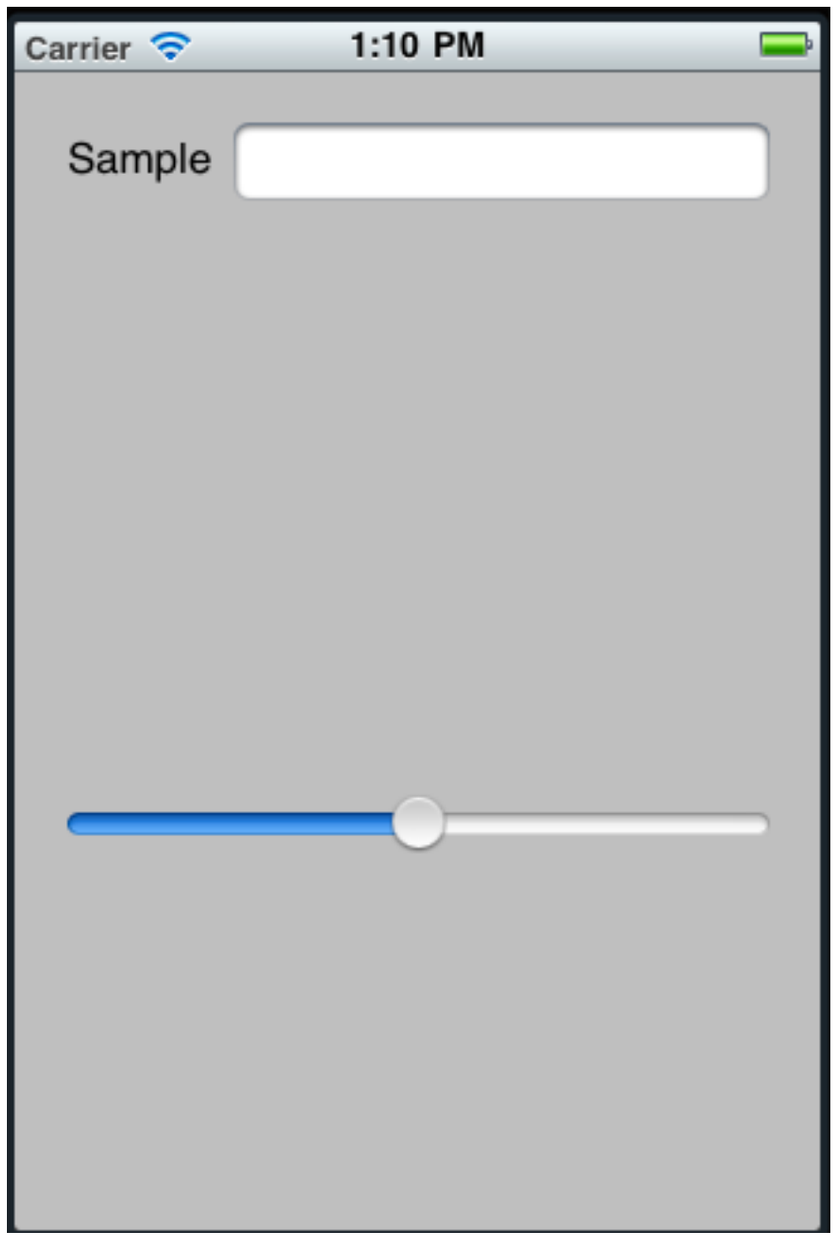
Warnings

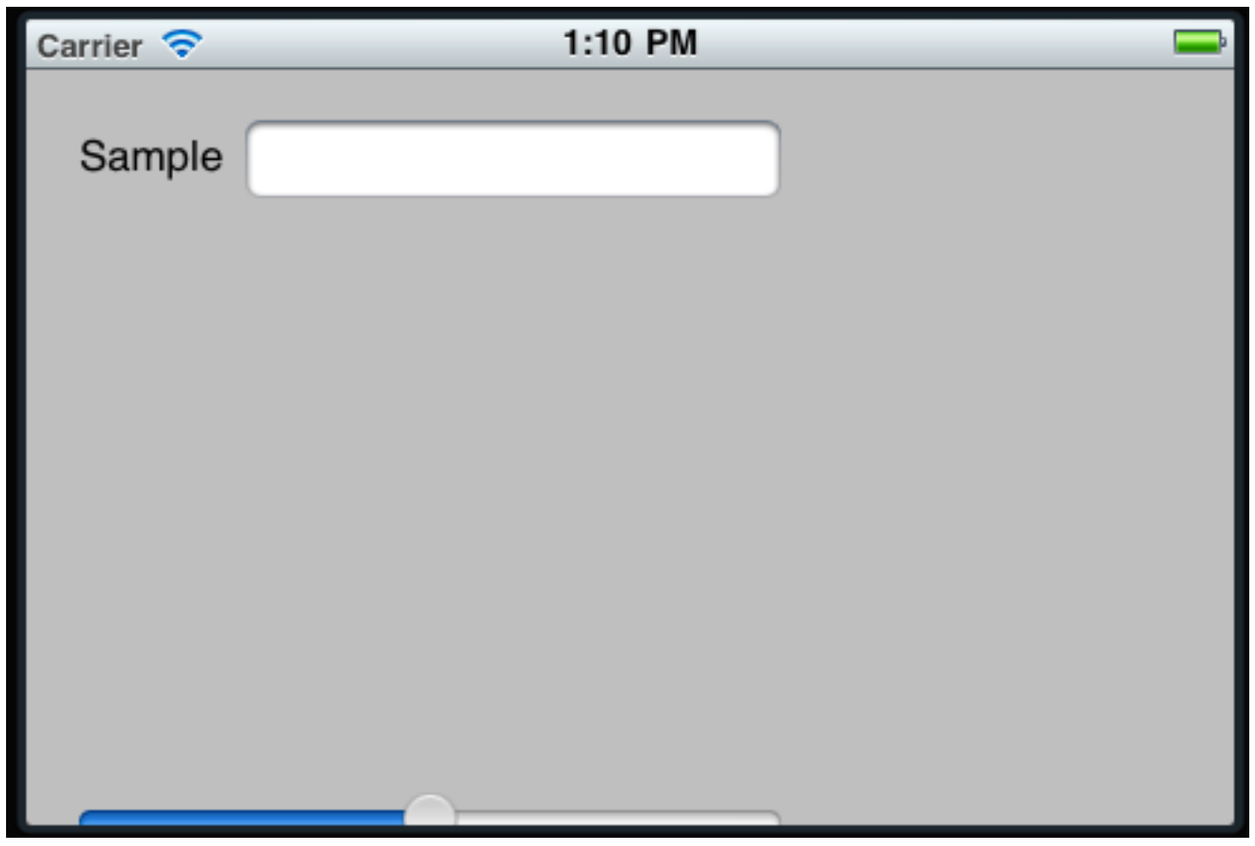
There are details left out (memory management)

There are more structured ways of doing this

Does not scale well

Screen Rotation





Issues

Allowing the view to rotate with device

Handling the rotation

- Springs & Struts in interface builder
- Change positions in code
- Use two views for landscape & portrait

Allowing view to rotate

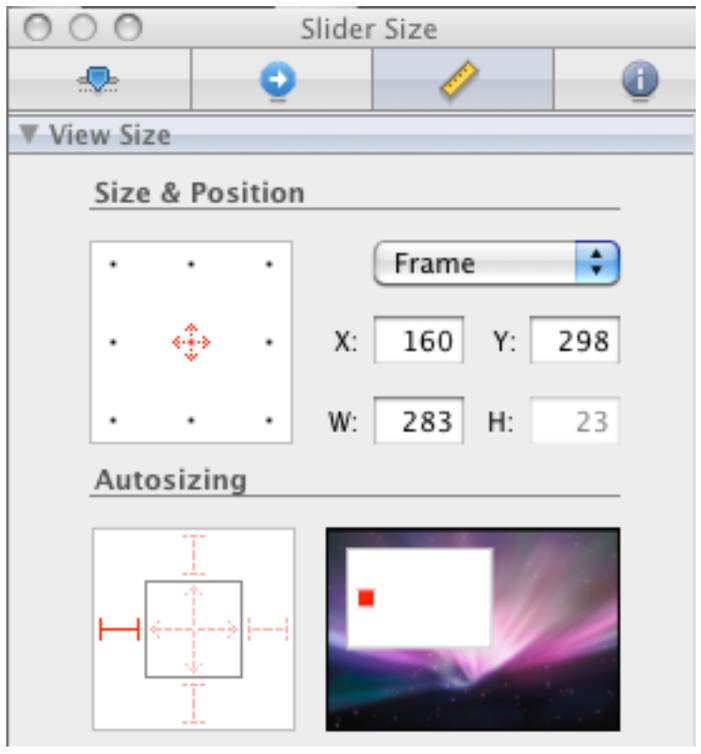
In controller class

```
- (BOOL)shouldAutorotateToInterfaceOrientation:
        (UIInterfaceOrientation)interfaceOrientation {
    // Return YES for supported orientations
    return (interfaceOrientation == UIInterfaceOrientationPortrait);
}
```

Possible orientation

UIInterfaceOrientationPortrait
UIInterfaceOrientationPortraitUpsideDown
UIInterfaceOrientationLandscapeLeft
UIInterfaceOrientationLandscapeRight

Structs & Springs

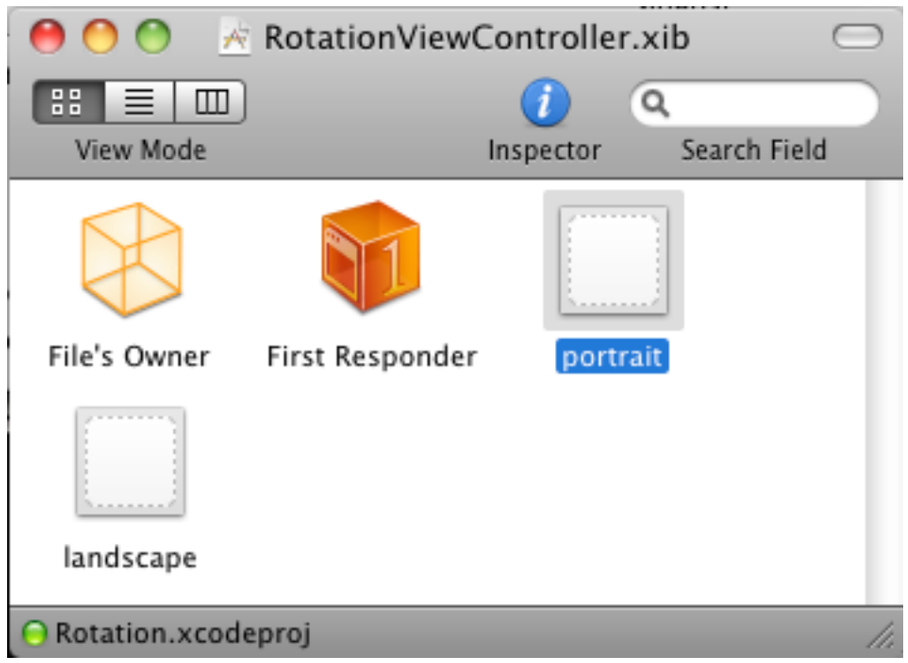


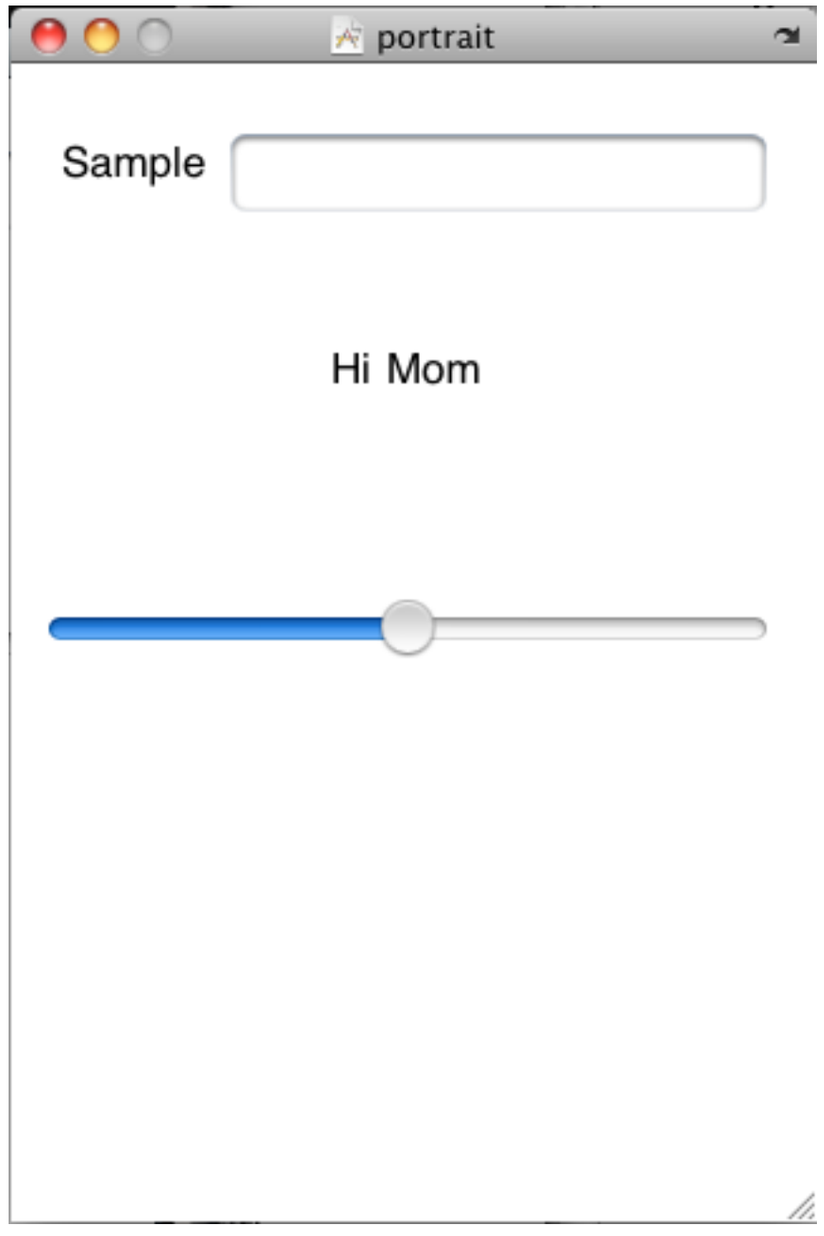
Changing location in Code

In controller

```
-(void)willAnimateRotationToInterfaceOrientation:(UIInterfaceOrientation) interfaceOrientation
        duration:(NSTimeInterval)duration {

    if (interfaceOrientation == UIInterfaceOrientationPortrait || interfaceOrientation ==
        UIInterfaceOrientationPortraitUpsideDown) {
        label.frame = CGRectMake(50, 30, 58, 21);          //CGRectMake(x, y, width, height)
        text.frame = CGRectMake(193, 35, 213, 31);
        slider.frame = CGRectMake(160, 298, 283, 23);
    } else {
        label.frame = CGRectMake(50, 33, 58, 21);
        text.frame = CGRectMake(229, 35, 289, 31);
        slider.frame = CGRectMake(229, 350, 289, 23);
    }
}
```

Using 2 views





Double outlets

```
#import <UIKit/UIKit.h>

@interface RotationViewController : UIViewController {

}
@property (nonatomic, retain) IBOutlet UIView * portrait;
@property (nonatomic, retain) IBOutlet UIView * landscape;

@property (nonatomic, retain) IBOutlet UILabel * portraitLabel;
@property (nonatomic, retain) IBOutlet UITextField * portraitText;
@property (nonatomic, retain) IBOutlet UISlider * portraitSlider;

@property (nonatomic, retain) IBOutlet UILabel * landscapeLabel;
@property (nonatomic, retain) IBOutlet UITextField * landscapeText;
@property (nonatomic, retain) IBOutlet UISlider * landscapeSlider;

@end
```

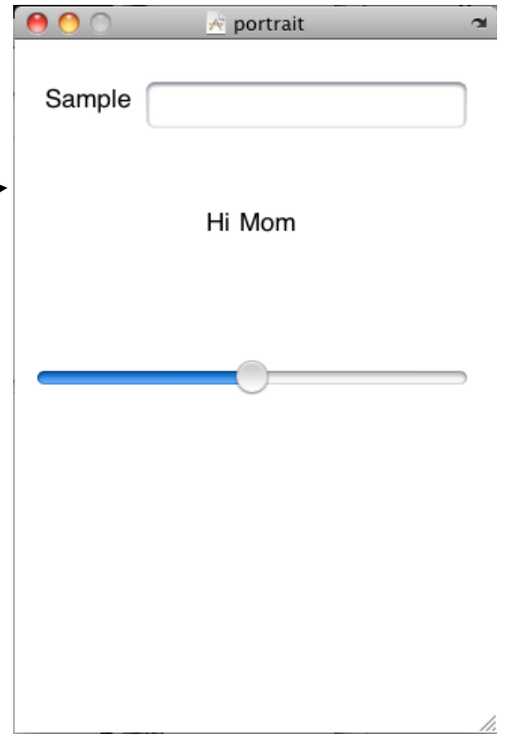
Some Outlets

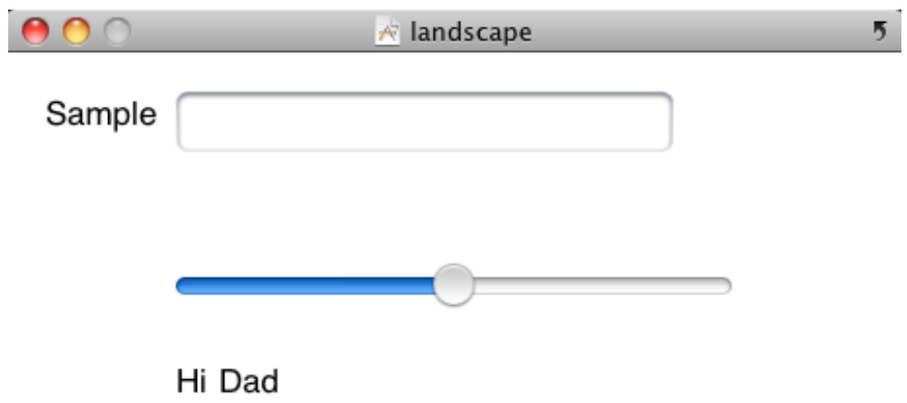
IBOutlets

view

portrait

landscape





Switching Views

```
#define degreesToRadians(x) (M_PI * (x) / 180.0)


-(void)willAnimateRotationToInterfaceOrientation:
    (UIInterfaceOrientation) interfaceOrientation duration:(NSTimeInterval)duration {

     if (interfaceOrientation == UIInterfaceOrientationPortrait) {
          self.view = self.portrait;
          self.view.transform = CGAffineTransformIdentity;
          self.view.transform = CGAffineTransformMakeRotation(degreesToRadians(0));
          self.view.bounds = CGRectMake(0.0, 0.0, 300.0, 480.0);
     } else if (interfaceOrientation == UIInterfaceOrientationLandscapeLeft) {
          self.view = self.landscape;
          self.view.transform = CGAffineTransformIdentity;
          self.view.transform = CGAffineTransformMakeRotation(degreesToRadians(-90));
          self.view.bounds = CGRectMake(0.0, 0.0, 460.0, 320.0);
     }
}
```

CGAffineTransformIdentity

preserves parallel lines

does not necessarily preserve lengths or angles

CGAffineTransformTranslate

CGAffineTransformScale

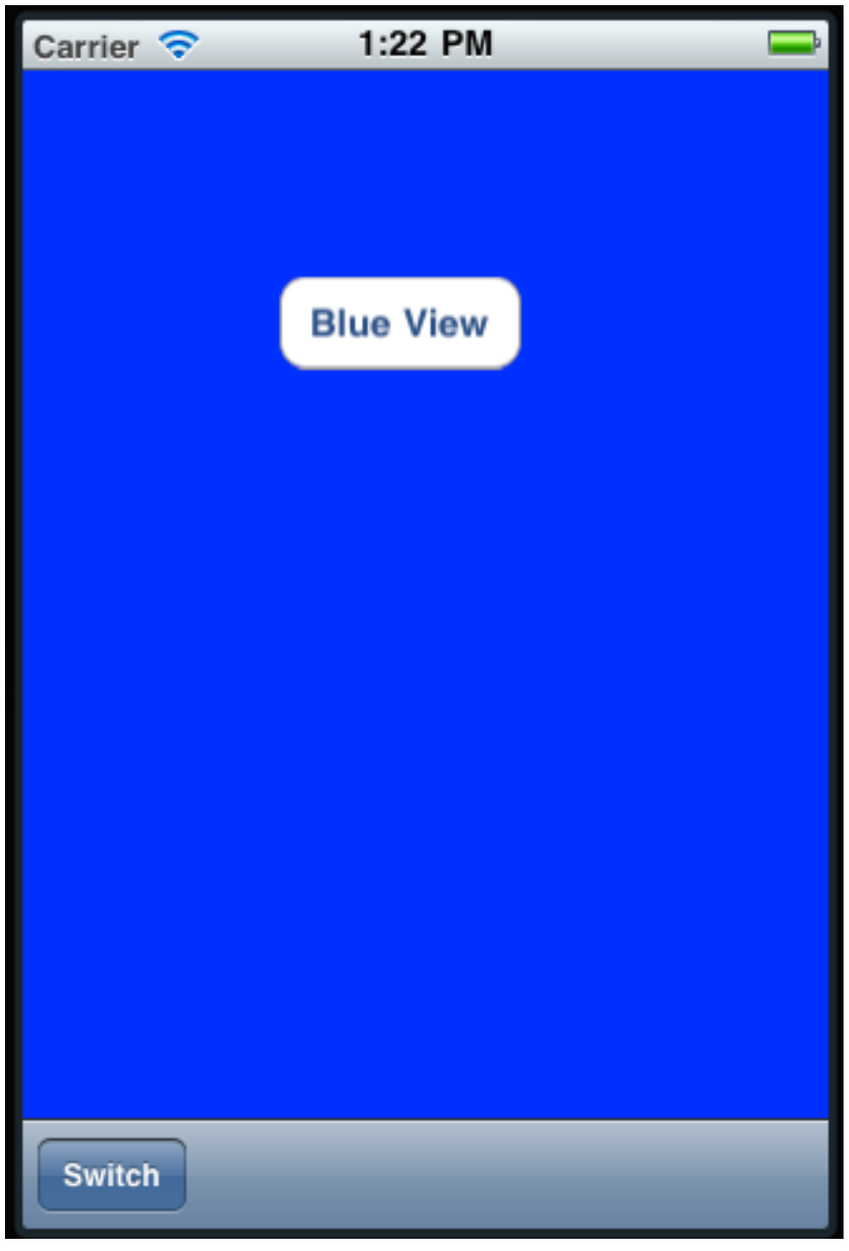
CGAffineTransformRotate

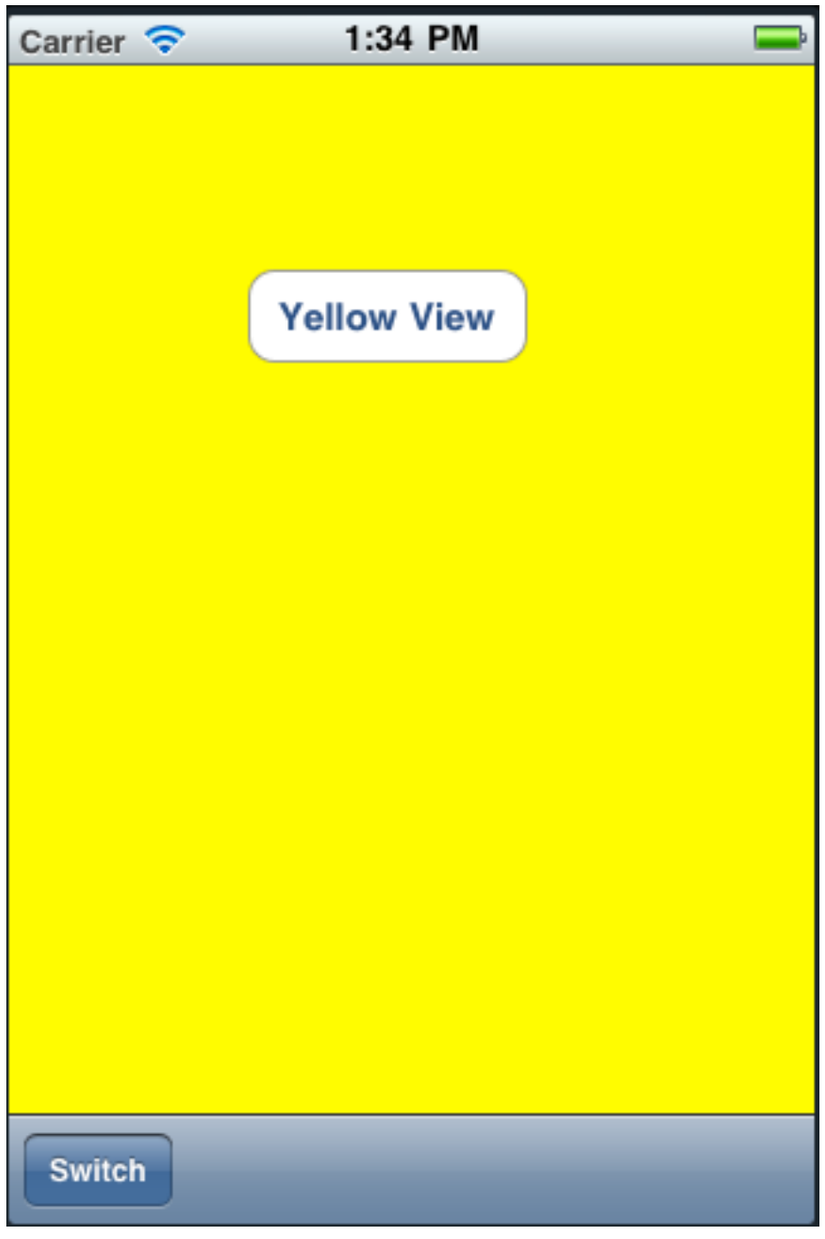
CGAffineTransformInvert

CGAffineTransformConcat

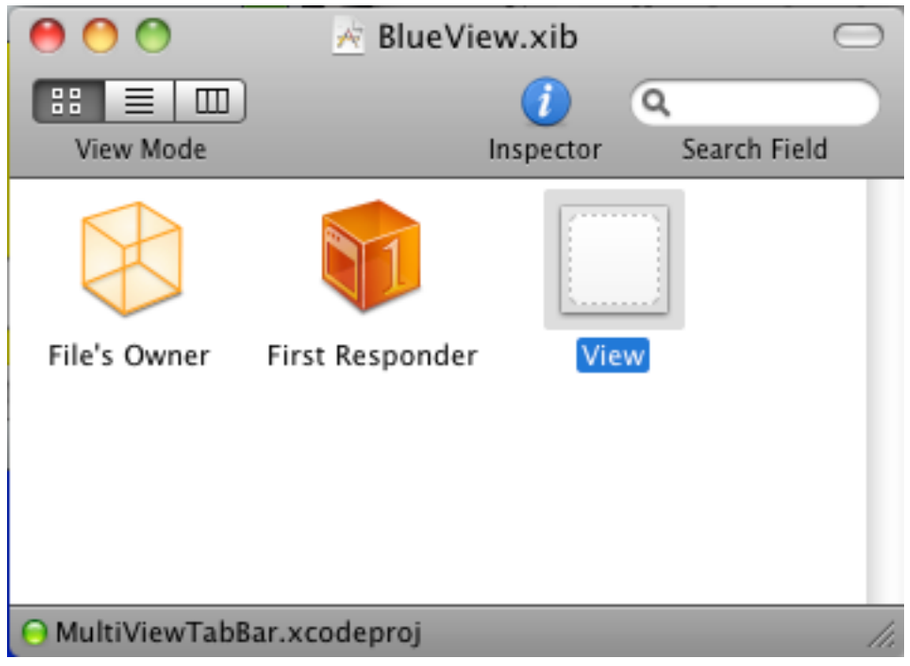
Toolbar

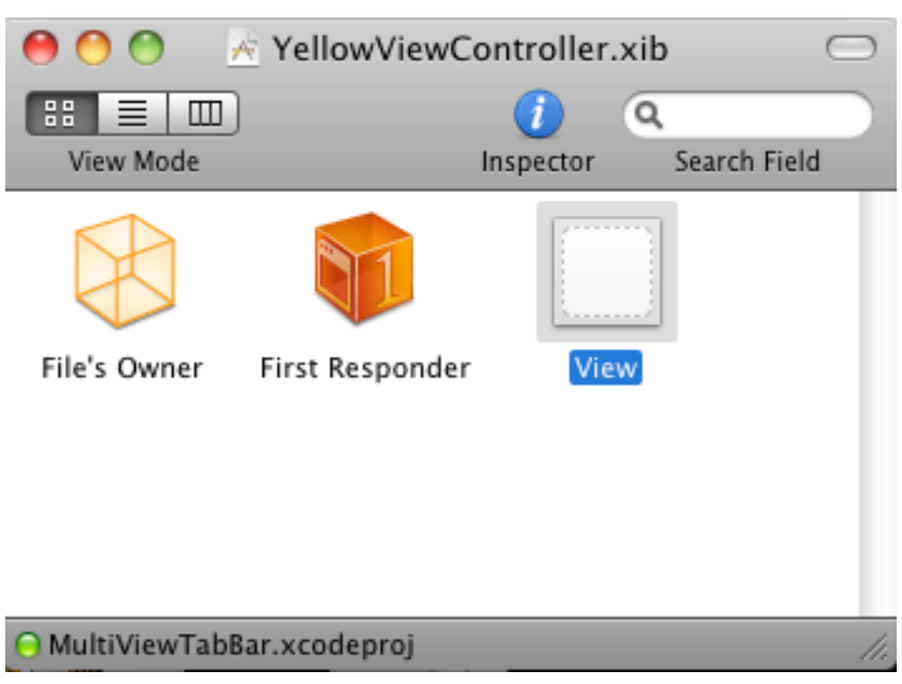
Tool bar with two views



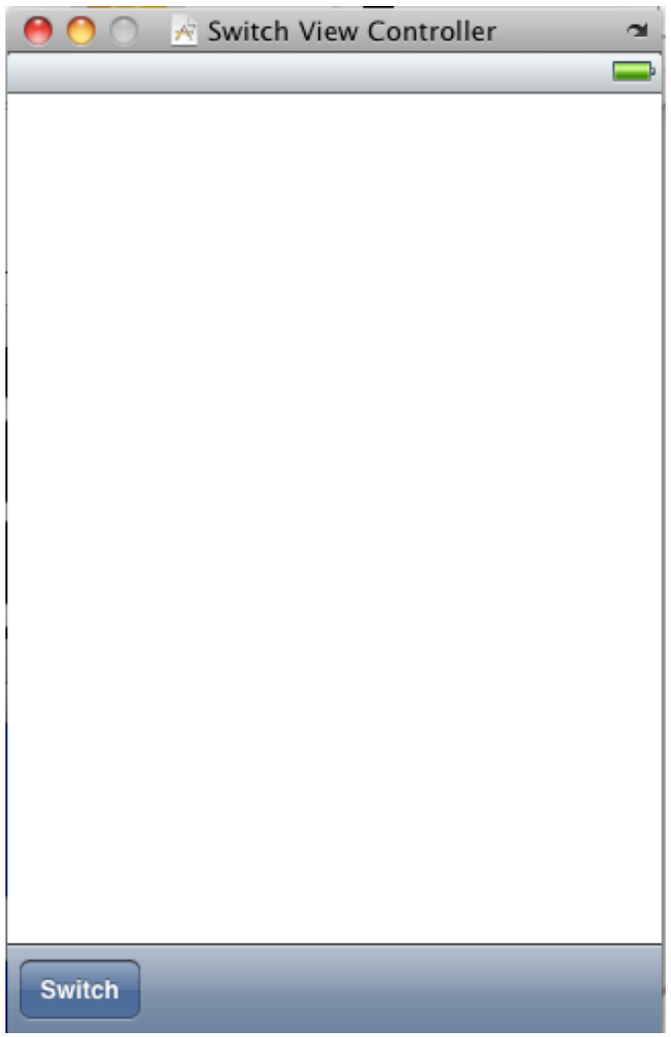


3 Views, 3 controllers, 3 xib files

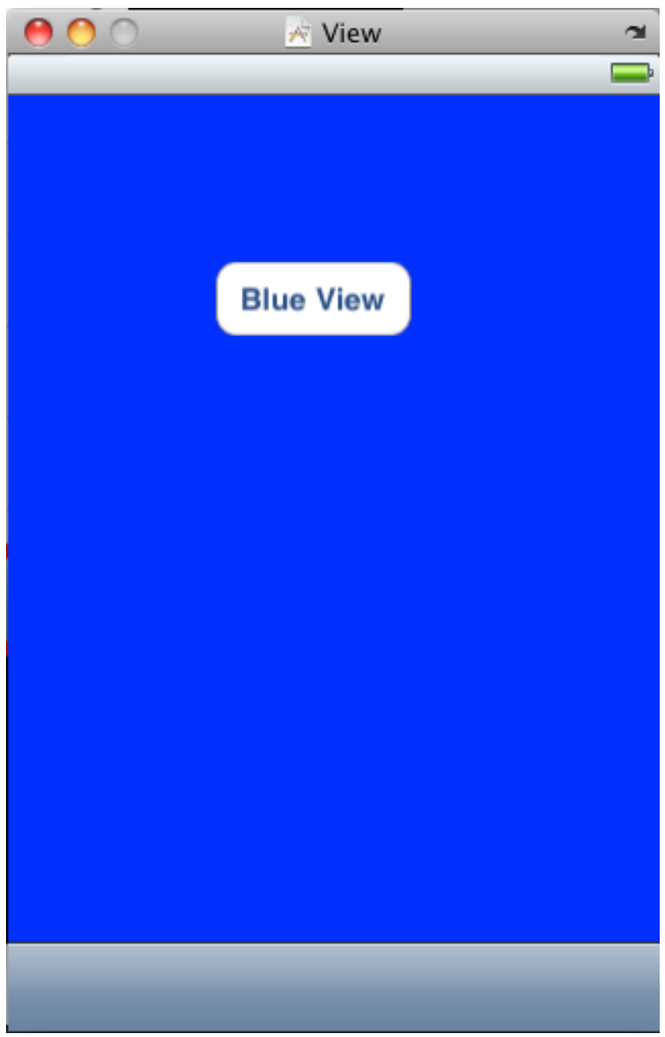




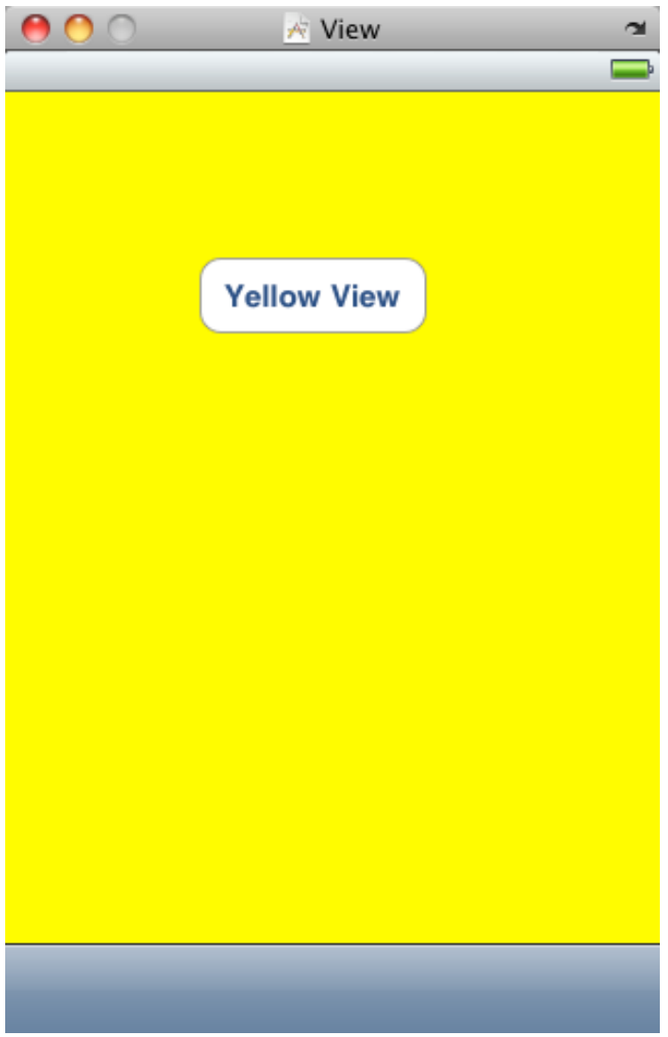
3 Views, 3 controllers, 3 xib files



SwitchViewController
view

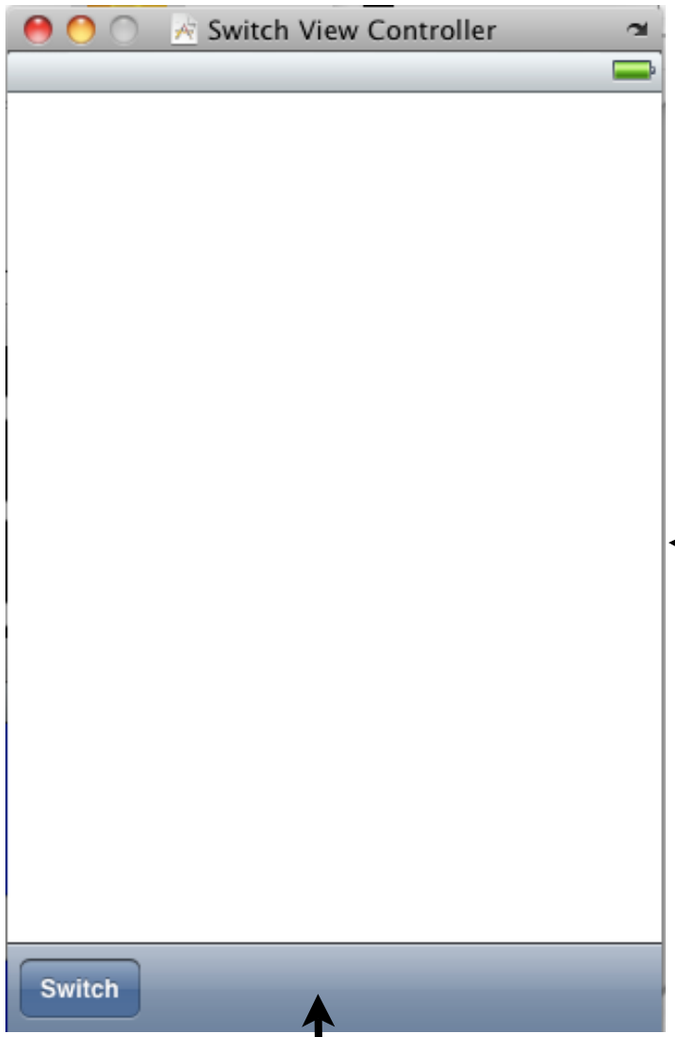


BlueViewController
view



YellowViewController
view

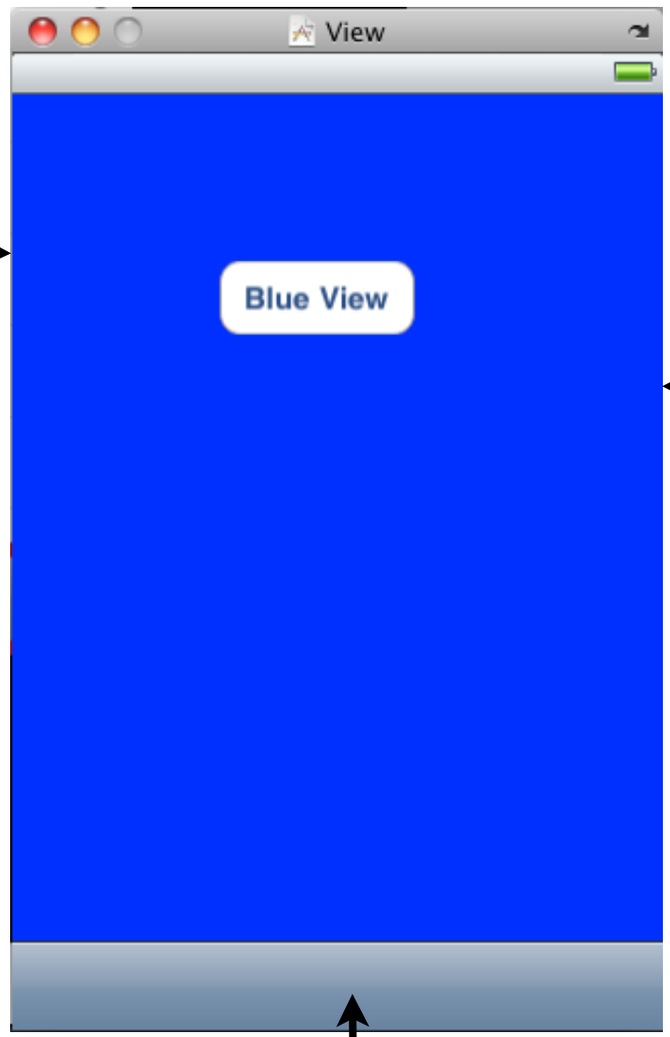
3 Views, 3 controllers, 3 xib files



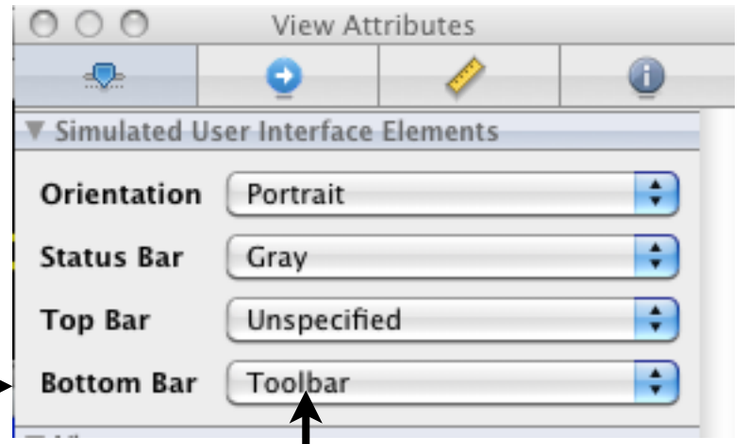
tool bar

view

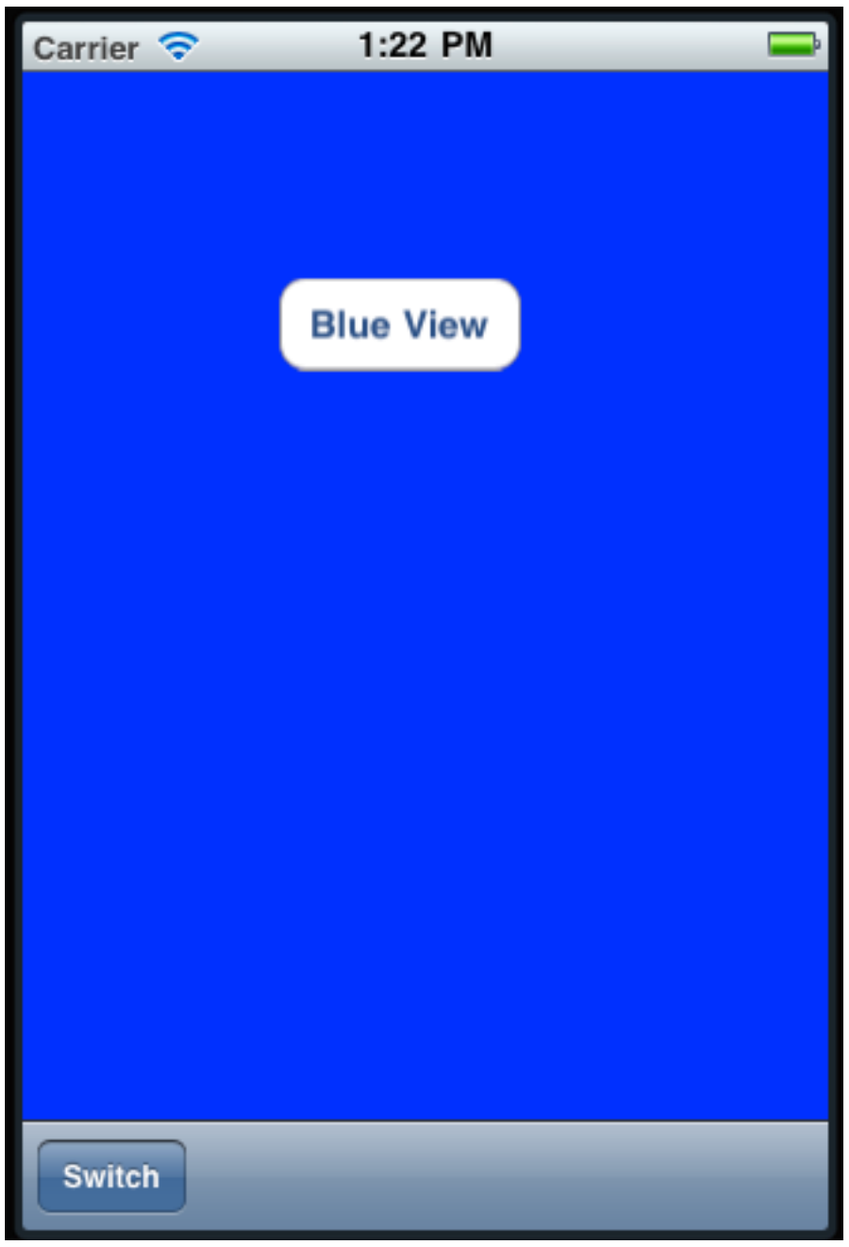
empty

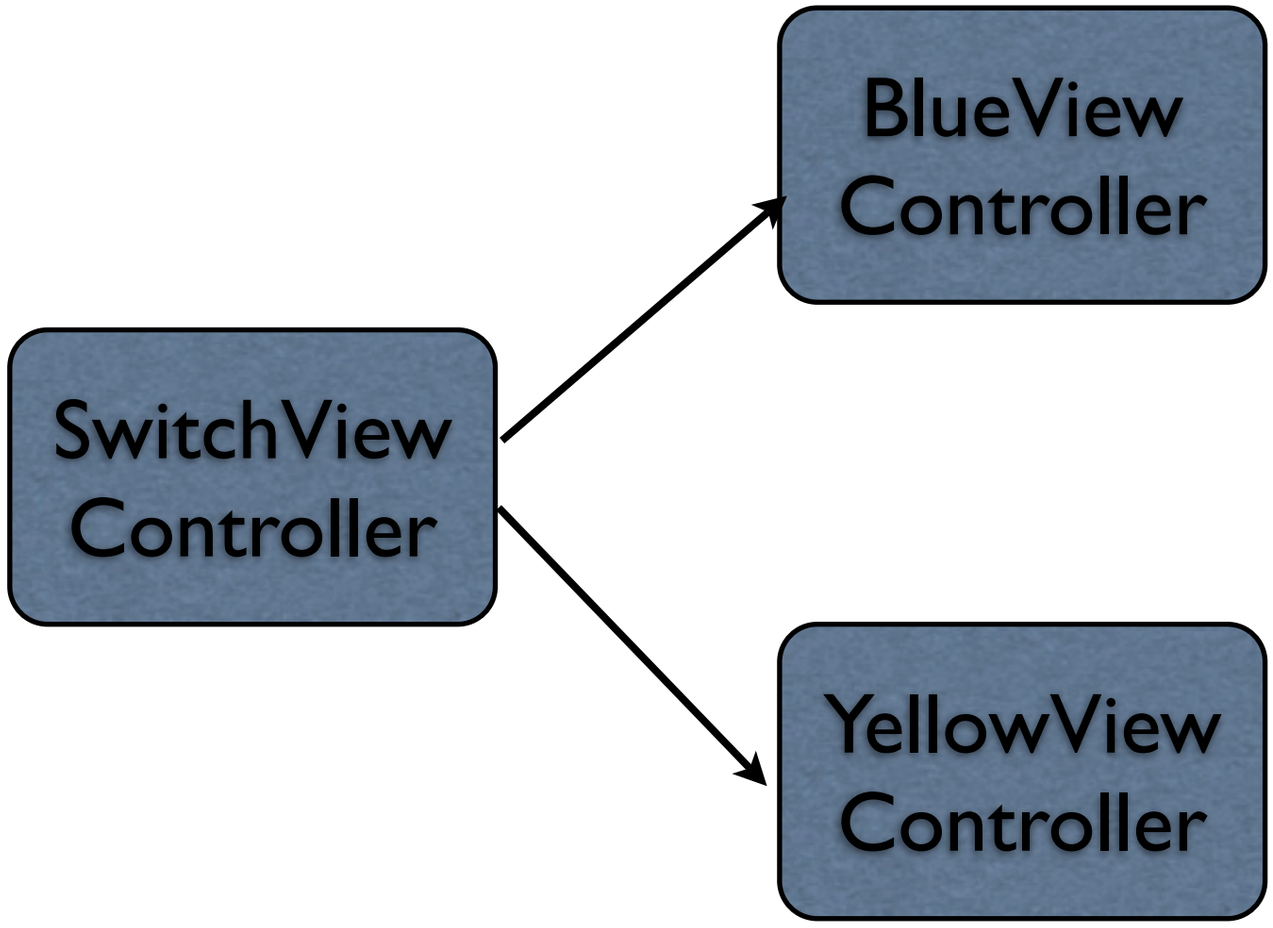


empty

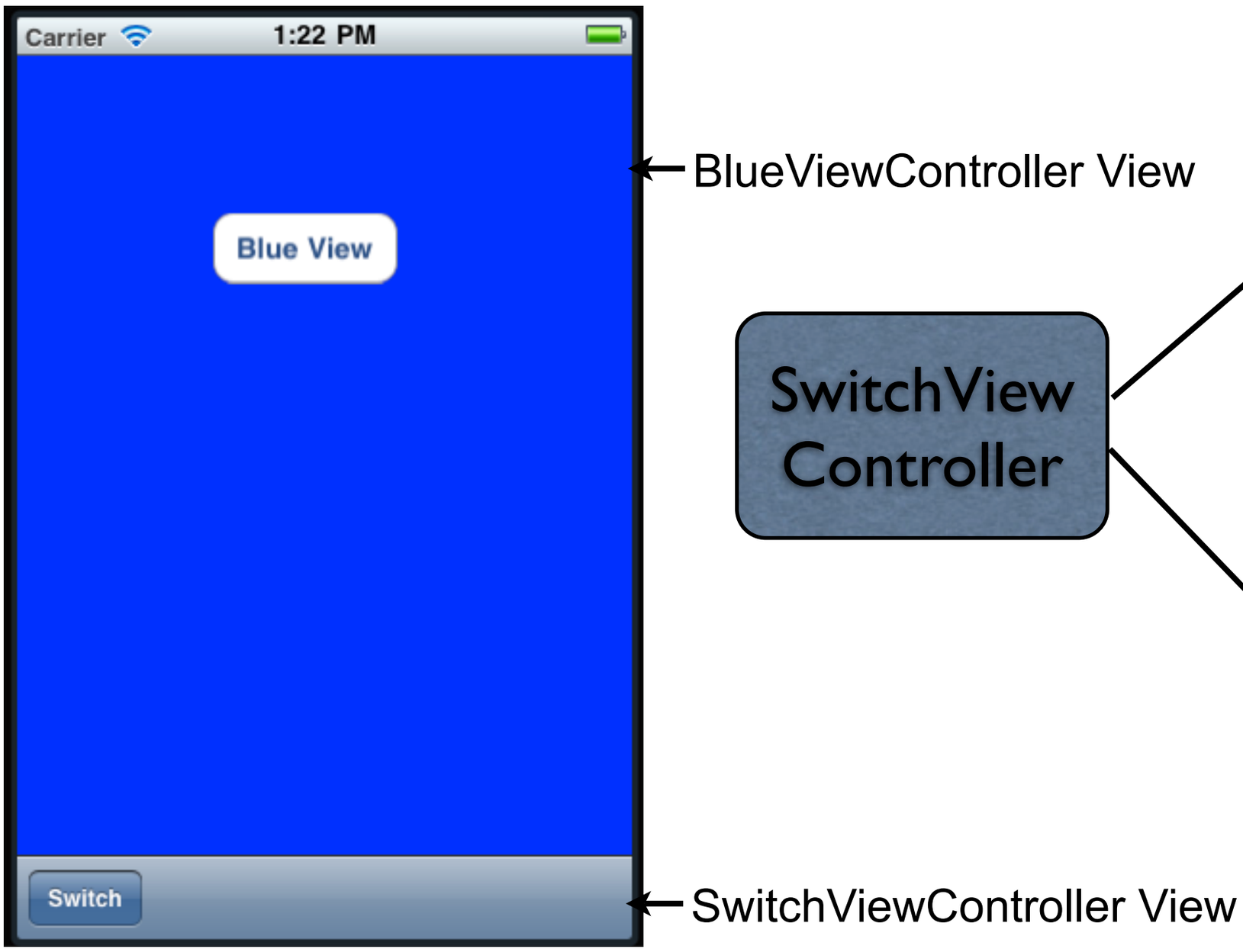


Controllers





Controllers & Views

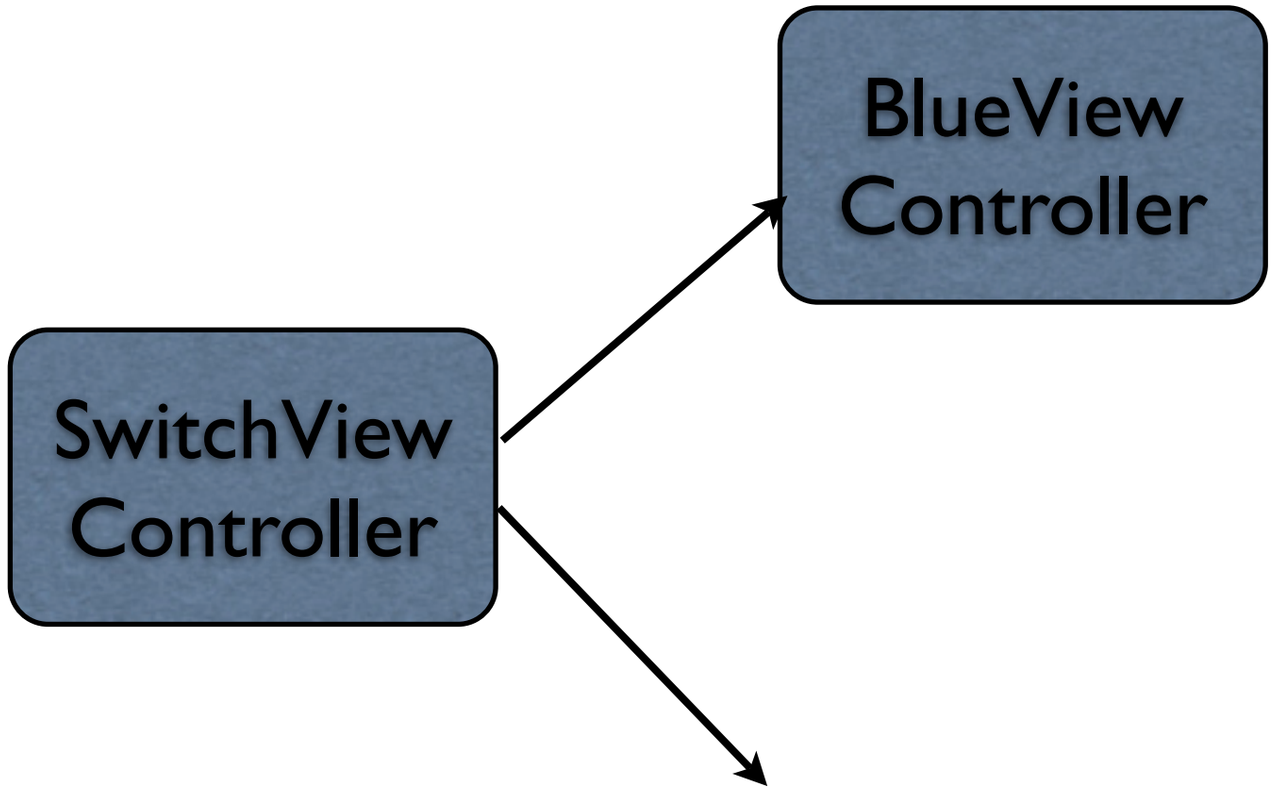


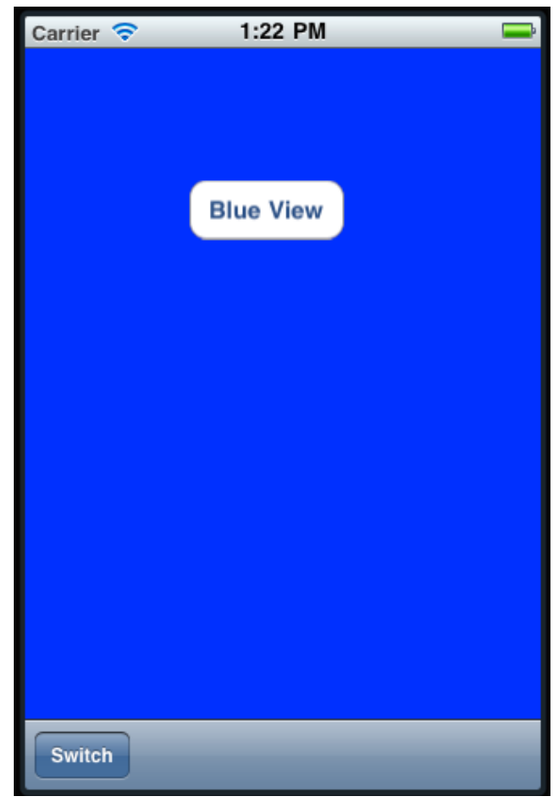
SwitchViewController

```
@class BlueViewController;
@class YellowViewController;

@interface SwitchViewController : UIViewController {

}
@property (retain, nonatomic) YellowViewController *yellowViewController;
@property (retain, nonatomic) BlueViewController *blueViewController;
-(IBAction)switchViews:(id)sender;
@end
```

SwitchViewController setup

```
- (void)viewDidLoad {
    BlueViewController *blueController =
        [[BlueViewController alloc] initWithNibName:@"BlueView" bundle:nil];
    self.blueViewController = blueController;
    [self.view insertSubview:blueController.view atIndex:0];
    [blueController release];
    [super viewDidLoad];
}
```



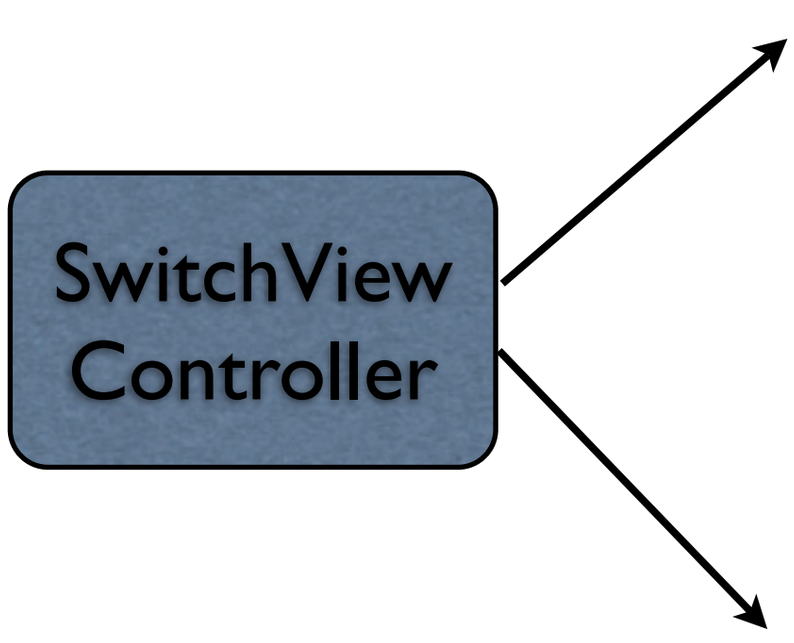


SwitchViewController setup

Load the nib file

Return controller

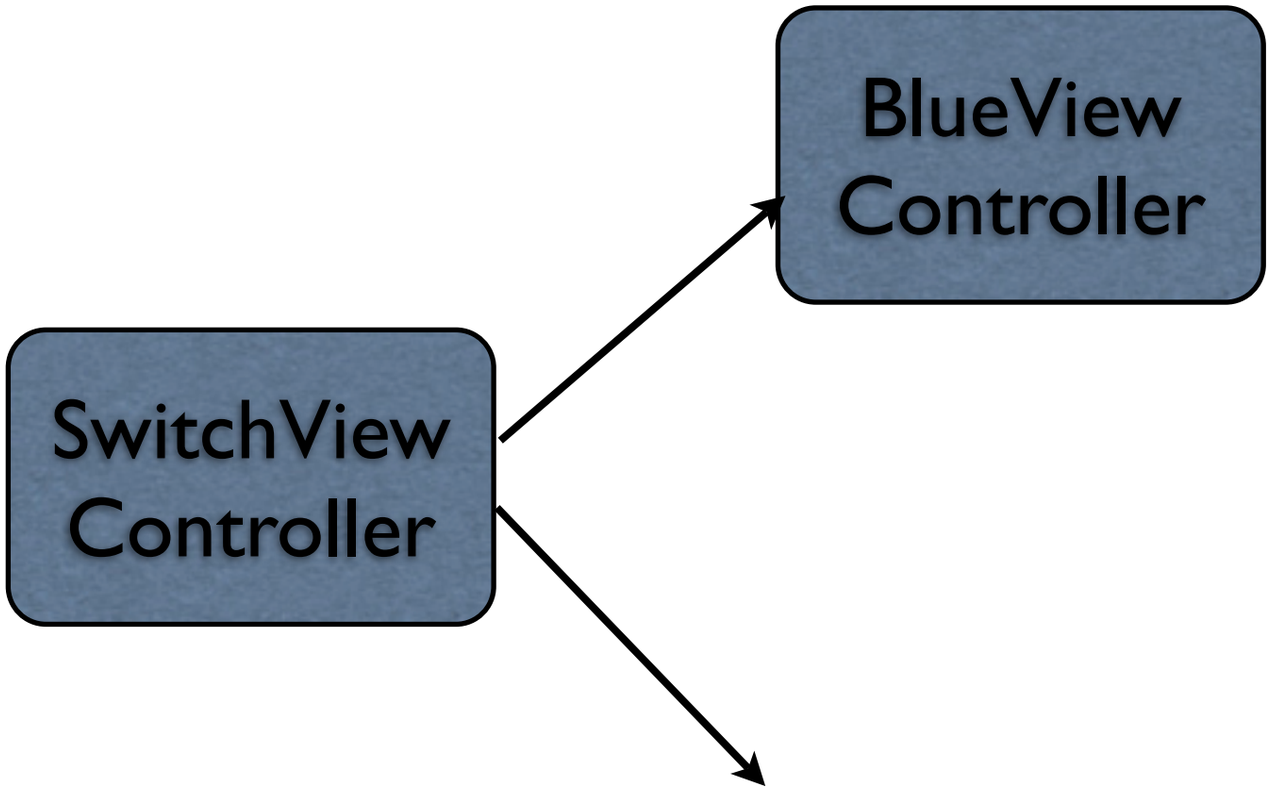
```
- (void)viewDidLoad {
    BlueViewController *blueController =
        [[BlueViewController alloc] initWithNibName:@"BlueView" bundle:nil];
    self.blueViewController = blueController;
    [self.view insertSubview:blueController.view atIndex:0];
    [blueController release];
    [super viewDidLoad];
}
```



SwitchViewController setup

Set the controller

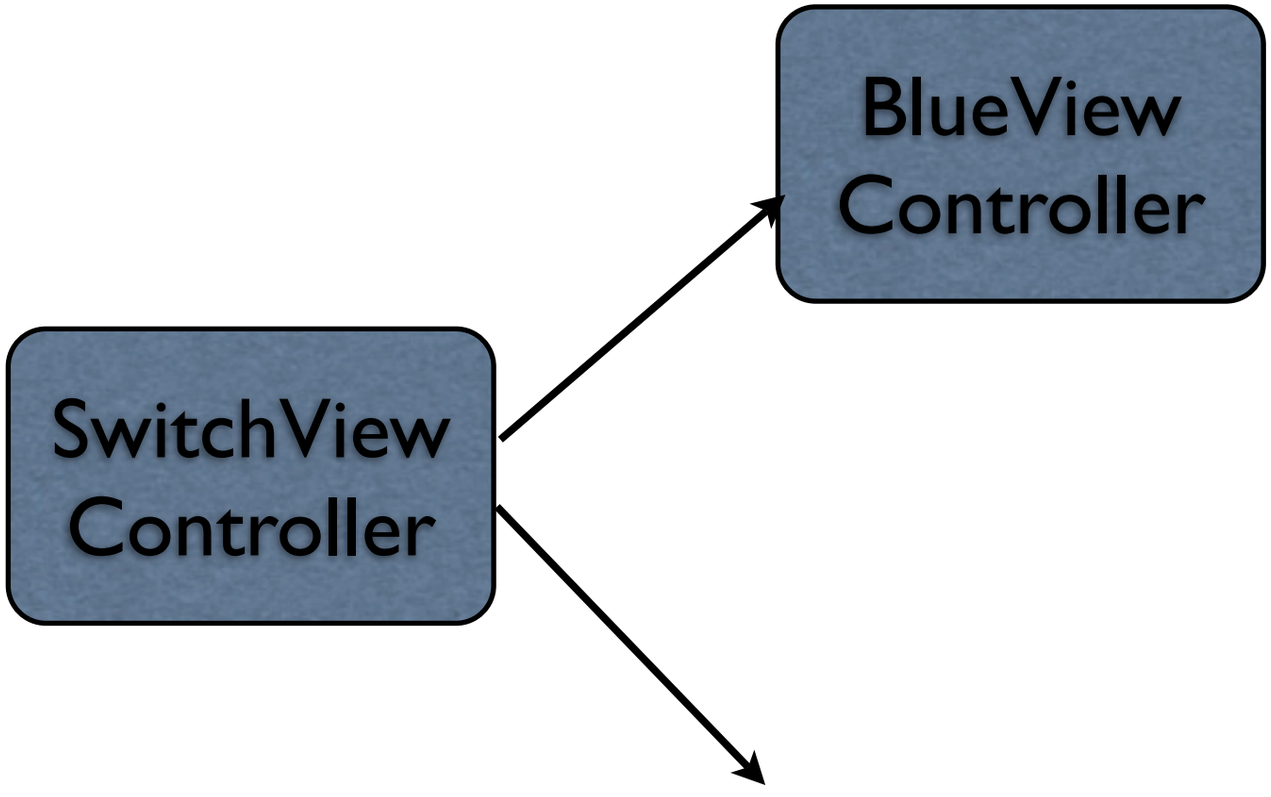
```
- (void)viewDidLoad {
    BlueViewController *blueController =
        [[BlueViewController alloc] initWithNibName:@"BlueView" bundle:nil];
    self.blueViewController = blueController;
    [self.view insertSubview:blueController.view atIndex:0];
    [blueController release];
    [super viewDidLoad];
}
```

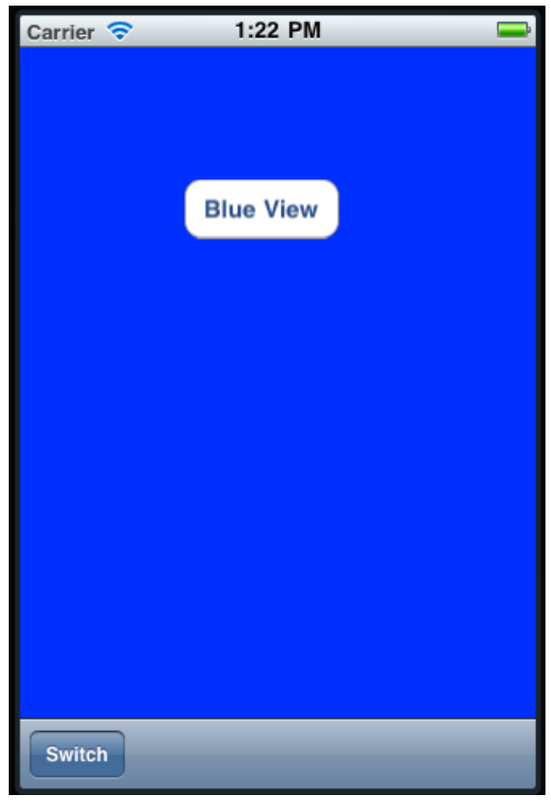


SwitchViewController setup

Set the view

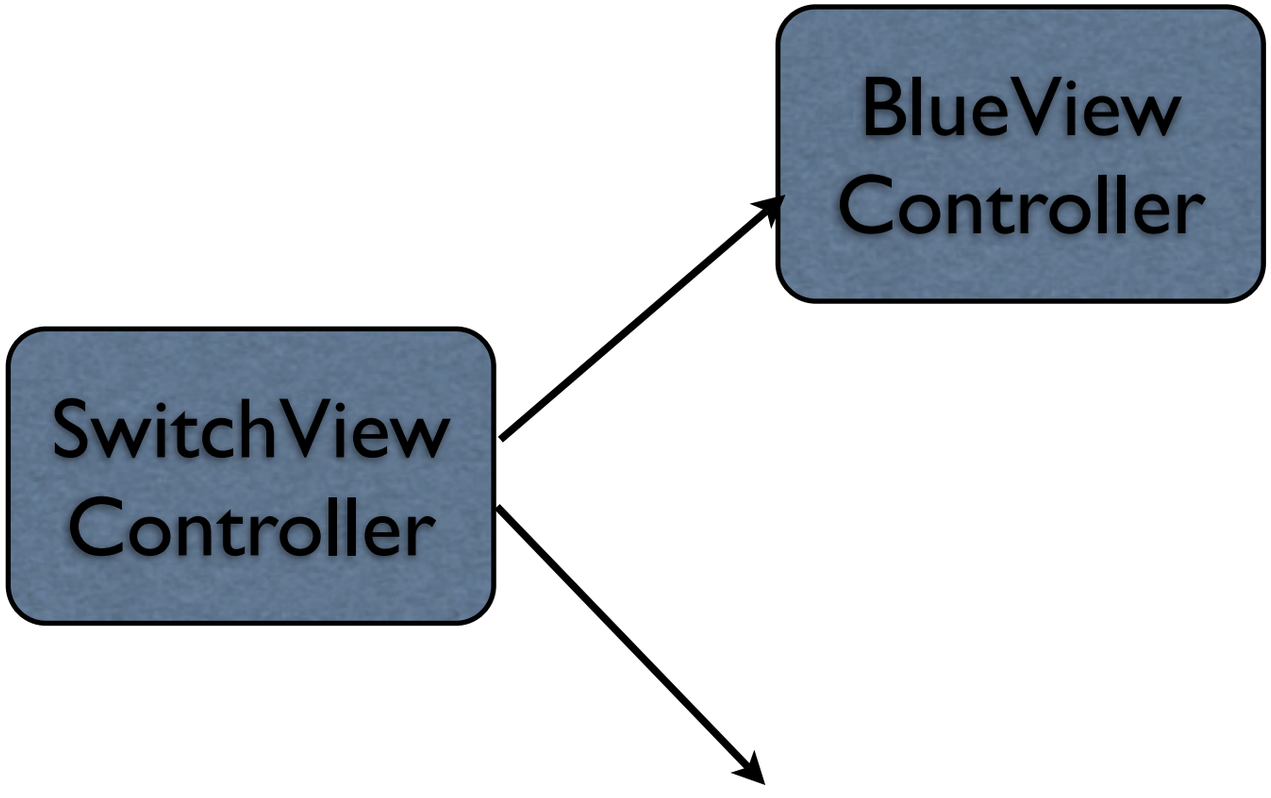
```
- (void)viewDidLoad {
    BlueViewController *blueController =
      [[BlueViewController alloc] initWithNibName:@"BlueView" bundle:nil];
    self.blueViewController = blueController;
    [self.view insertSubview:blueController.view atIndex:0];
    [blueController release];
    [super viewDidLoad];
}
```

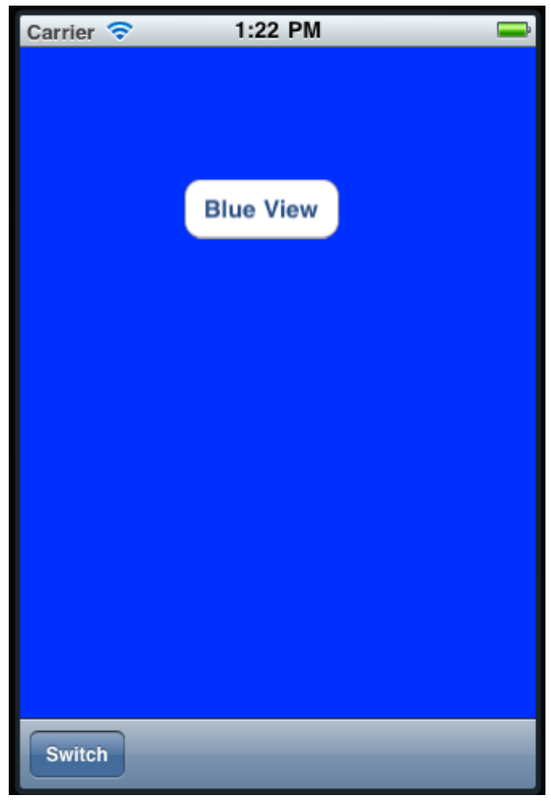




SwitchViewController setup


```
- (void)viewDidLoad {
    BlueViewController *blueController =
        [[BlueViewController alloc] initWithNibName:@"BlueView" bundle:nil];
    self.blueViewController = blueController;
    [self.view insertSubview:blueController.view atIndex:0];
    [blueController release];
    [super viewDidLoad];
}
```





Swapping Views

```
- (IBAction) switchViews:(id)sender {
    if (self.yellowViewController.view.superview == nil) {
        if (self.yellowViewController == nil) {
            [self readYellowViewNib];
        }
        [blueViewController.view removeFromSuperview];
        [self.view insertSubview:yellowViewController.view atIndex:0];
    } else {
        if (self.blueViewController == nil) {
            [self readBlueViewNib];
        }
        [yellowViewController.view removeFromSuperview];
        [self.view insertSubview:blueViewController.view atIndex:0];
    }
}
```

Swapping Views

Current view has superview

```
- (IBAction) switchViews:(id)sender {
    if (self.yellowViewController.view.superview == nil) {
        if (self.yellowViewController == nil) {
            [self readYellowViewNib];
        }
        [blueViewController.view removeFromSuperview];
        [self.view insertSubview:yellowViewController.view atIndex:0];
    } else {
        if (self.blueViewController == nil) {
            [self readBlueViewNib];
        }
        [yellowViewController.view removeFromSuperview];
        [self.view insertSubview:blueViewController.view atIndex:0];
    }
}
```

Swapping Views

Lazy evaluation

```
- (IBAction) switchViews:(id)sender {
    if (self.yellowViewController.view.superview == nil) {
        if (self.yellowViewController == nil) {
            [self readYellowViewNib];
        }
        [blueViewController.view removeFromSuperview];
        [self.view insertSubview:yellowViewController.view atIndex:0];
    } else {
        if (self.blueViewController == nil) {
            [self readBlueViewNib];
        }
        [yellowViewController.view removeFromSuperview];
        [self.view insertSubview:blueViewController.view atIndex:0];
    }
}
```

Swapping Views

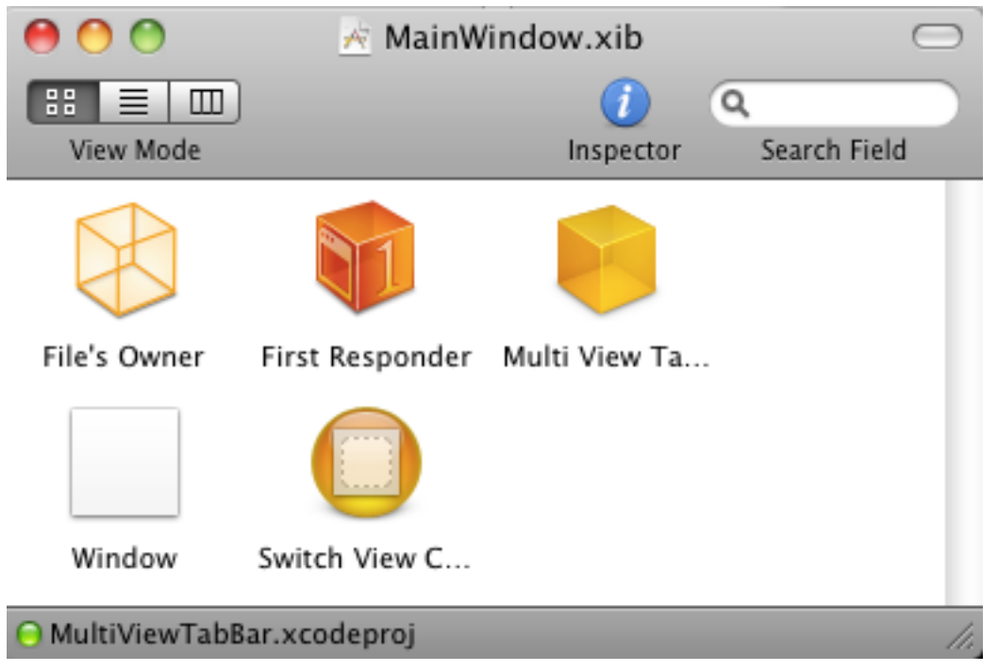
The swap

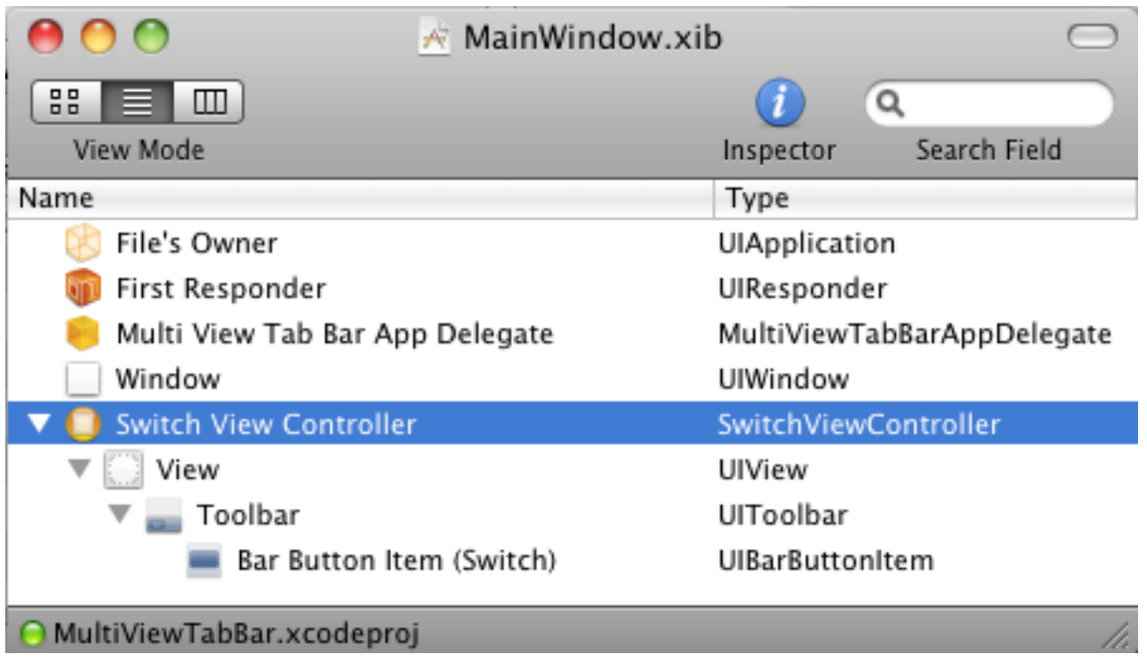
```
- (IBAction) switchViews:(id)sender {
    if (self.yellowViewController.view.superview == nil) {
        if (self.yellowViewController == nil) {
            [self readYellowViewNib];
        }
        [blueViewController.view removeFromSuperview];
        [self.view insertSubview:yellowViewController.view atIndex:0];
    } else {
        if (self.blueViewController == nil) {
            [self readBlueViewNib];
        }
        [yellowViewController.view removeFromSuperview];
        [self.view insertSubview:blueViewController.view atIndex:0];
    }
}
```

readYellowViewNib

```
- (void) readYellowViewNib {
    YellowViewController *yellowController =
            [[YellowViewController alloc]
                initWithNibName:@"YellowViewController" bundle:nil];
    self.yellowViewController = yellowController;
    [yellowController release];
}
```

MainWindow.xib

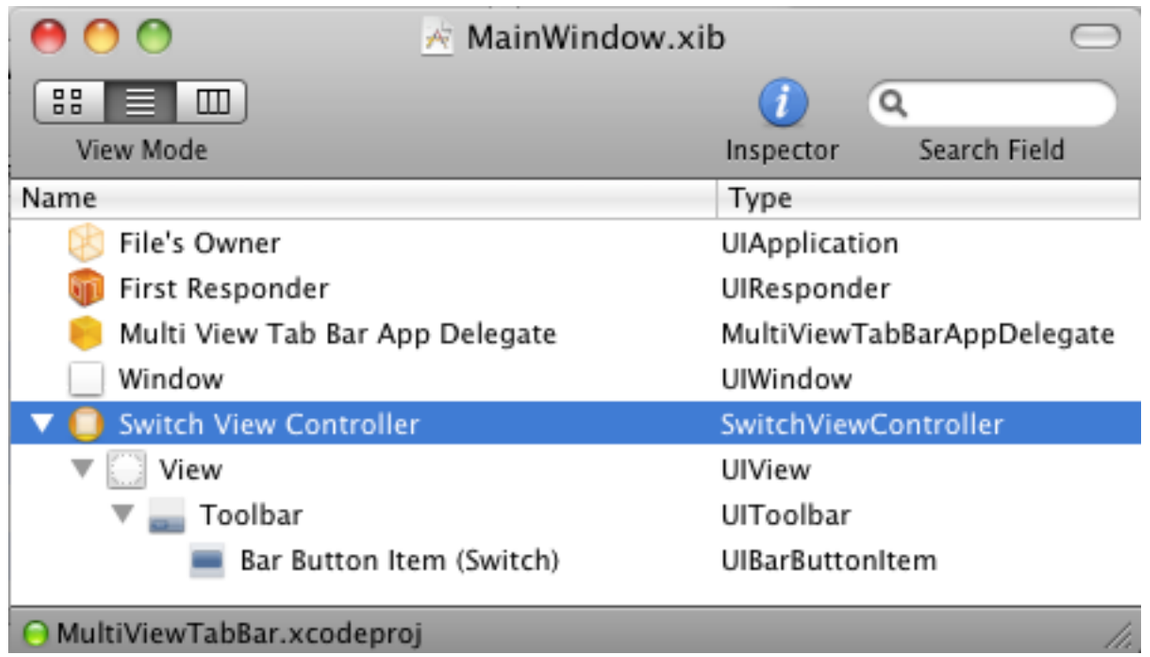




File Owner

UIApplication

IBOutlet - App Delegate

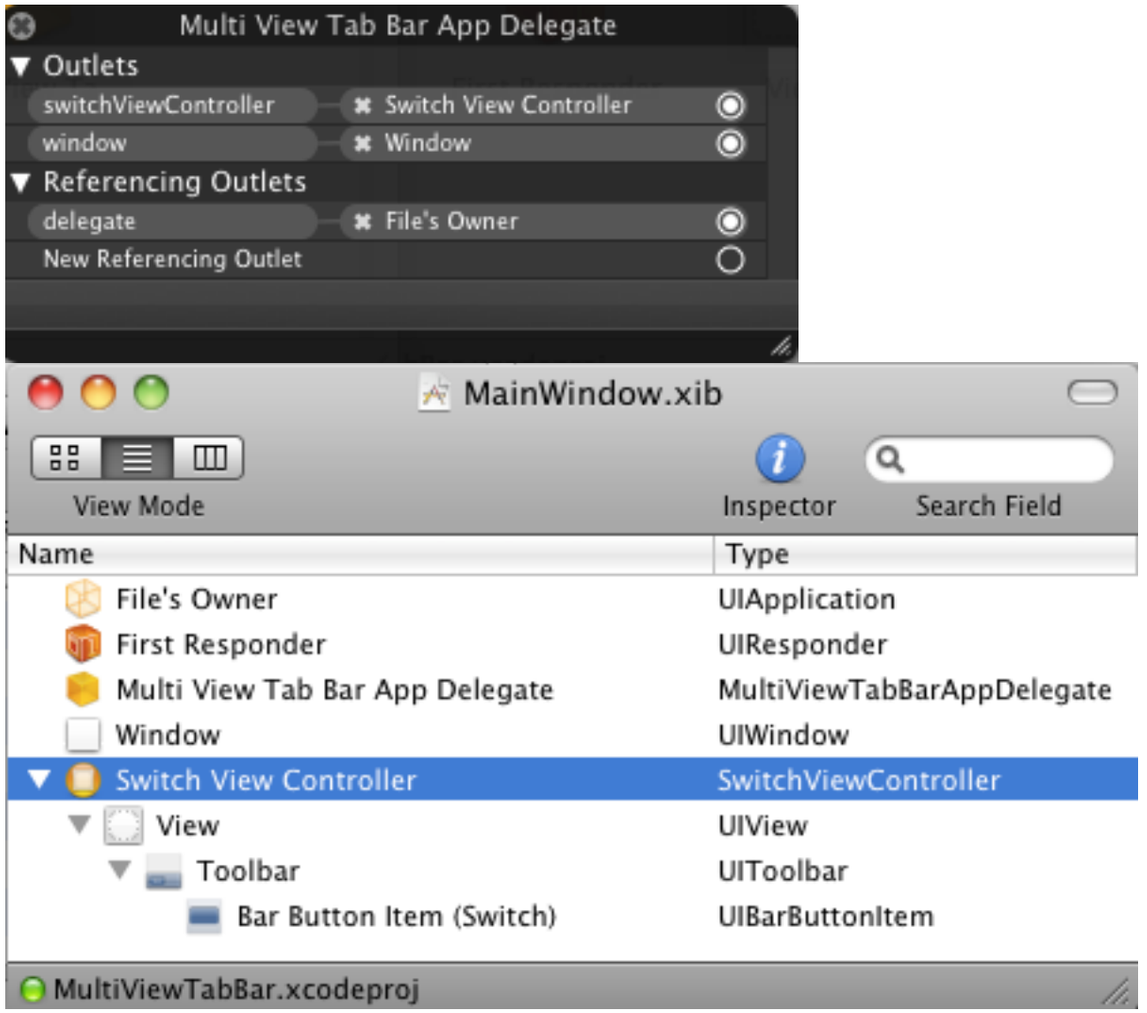


App Deleate

MultiViewTabBarAppDelegate

IBOutlets

- Window
- Controller



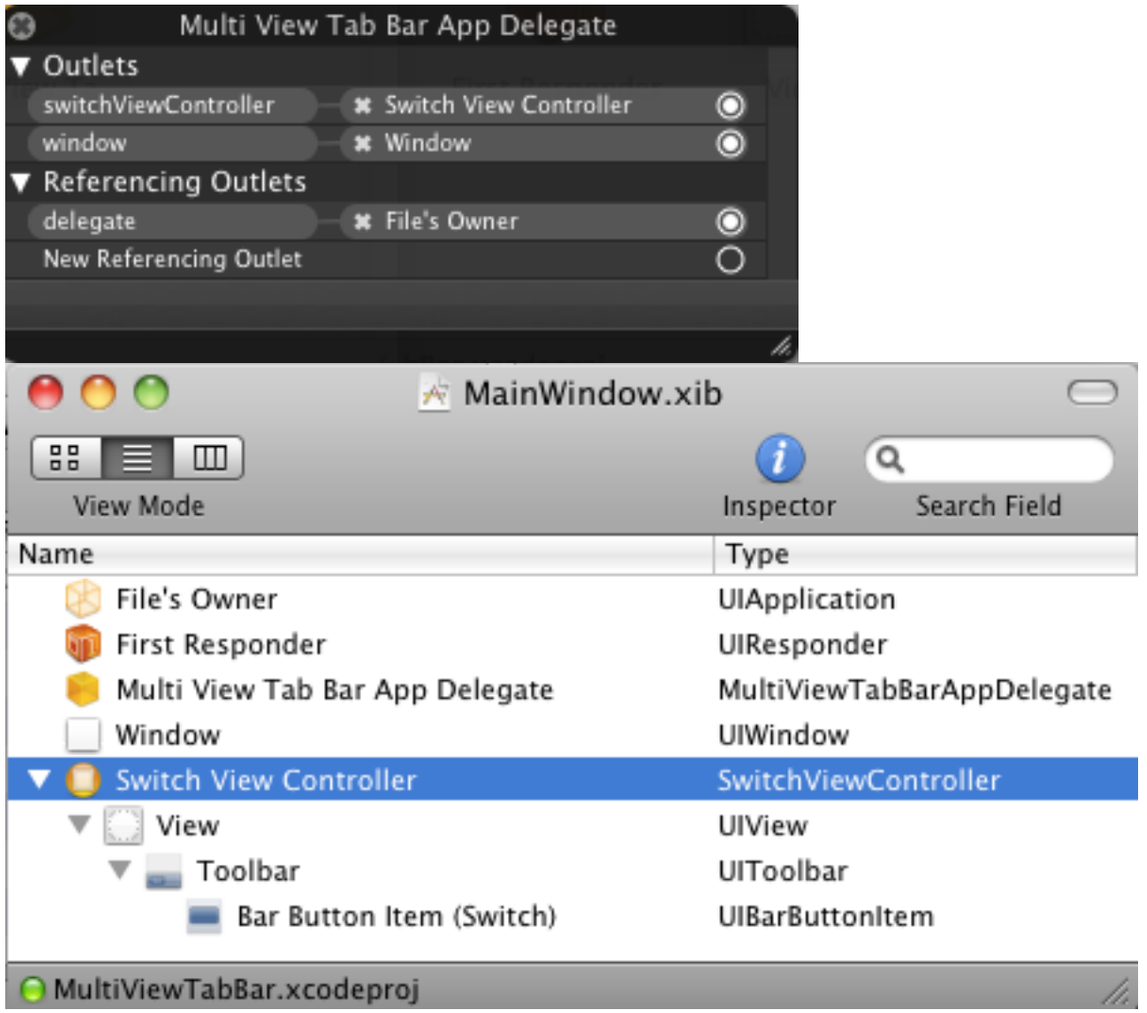

```
@class SwitchViewController;
@interface MultiViewTabBarAppDelegate : NSObject <UIApplicationDelegate> {
}

@property (nonatomic, retain) IBOutlet UIWindow *window;
@property (nonatomic, retain) IBOutlet SwitchViewController *switchViewController;
@end
```

App Delegate

MultiViewTabBarAppDelegate

IBOutlets

- Window
- Controller




```
@class SwitchViewController;
@interface MultiViewTabBarAppDelegate : NSObject <UIApplicationDelegate> {
}

@property (nonatomic, retain) IBOutlet UIWindow *window;
@property (nonatomic, retain) IBOutlet SwitchViewController *switchViewController;
@end
```

Controller

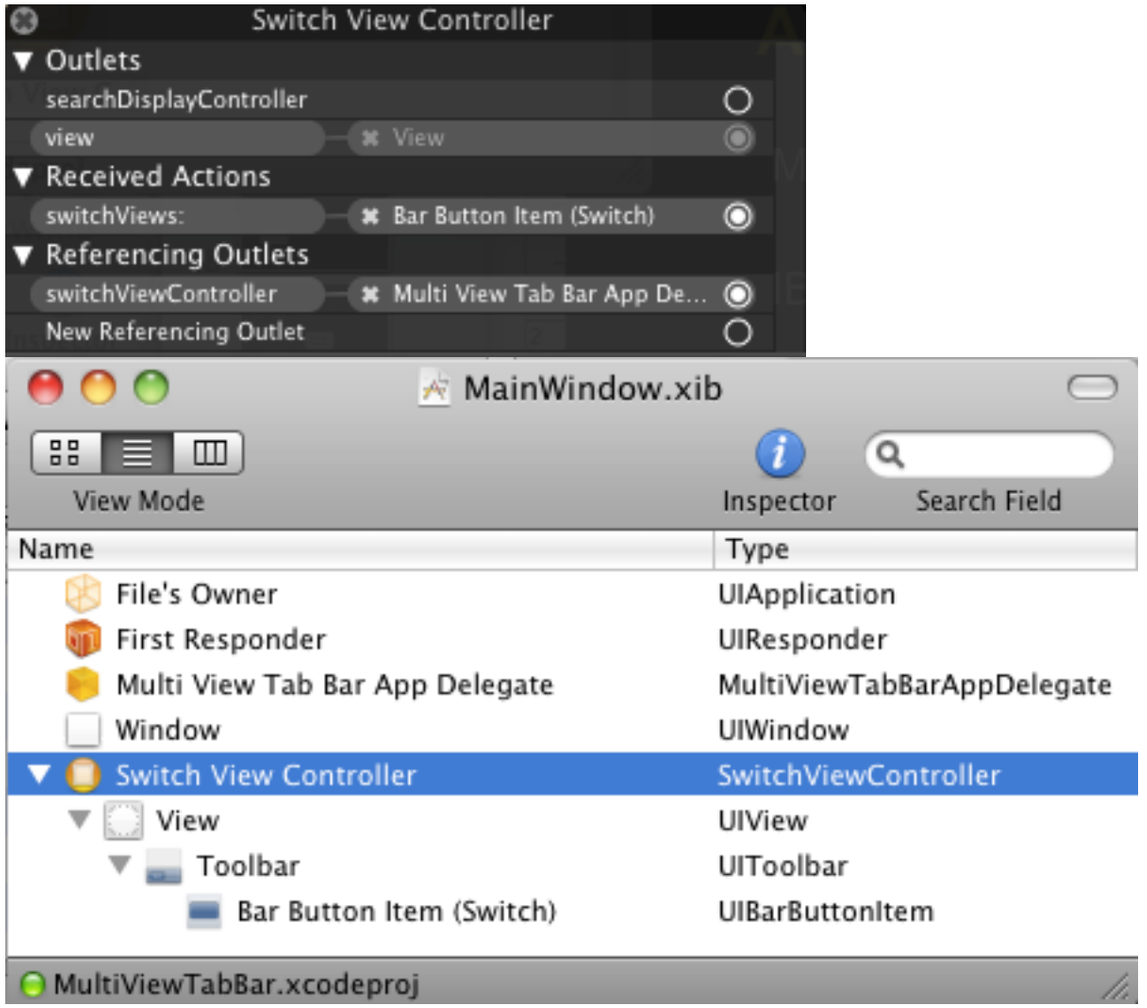
MultiViewTabBarAppDelegate

IBOutlets

view

IBActions

switchViews:




```
@class BlueViewController;
@class YellowViewController;

@interface SwitchViewController : UIViewController {

}
@property (retain, nonatomic) YellowViewController *yellowViewController;
@property (retain, nonatomic) BlueViewController *blueViewController;
-(IBAction)switchViews:(id)sender;
@end
```