

ViewController



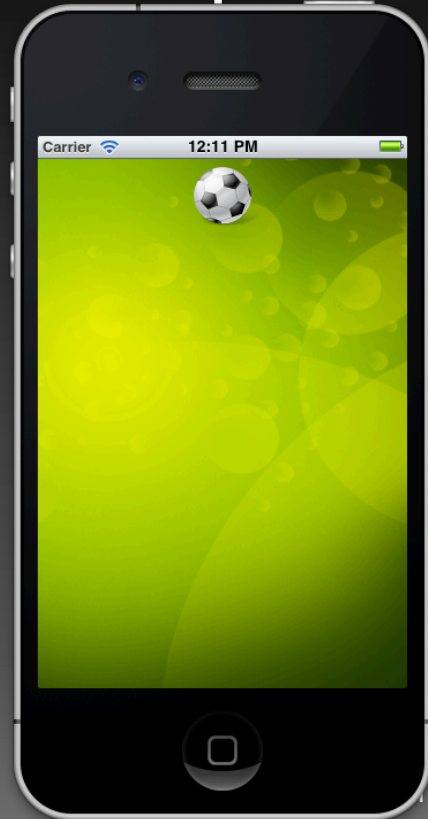
Trinh Minh Cuong

Hôm trước đã học UIView

- Hierarchy (bố, con, anh, chị, trên, dưới)
- Geometry (ở chỗ nào, to, bé ra làm sao)
- Rendering (hiển thị đậm nhạt, ẩn hiện...)
- Animation (hoạt hình cổ điển)
- Animation with Block (hoạt hình sử dụng block)

Bài tập UIView giao về nhà

- Hoạt họa di chuyển quả bóng theo hình quả trám
- Games maching kiểm tra trí nhớ



Nội dung

- UIViewController
- UINavigationController
 - Customize navigation bar
 - Share data between ViewControllers
- UITabBarController



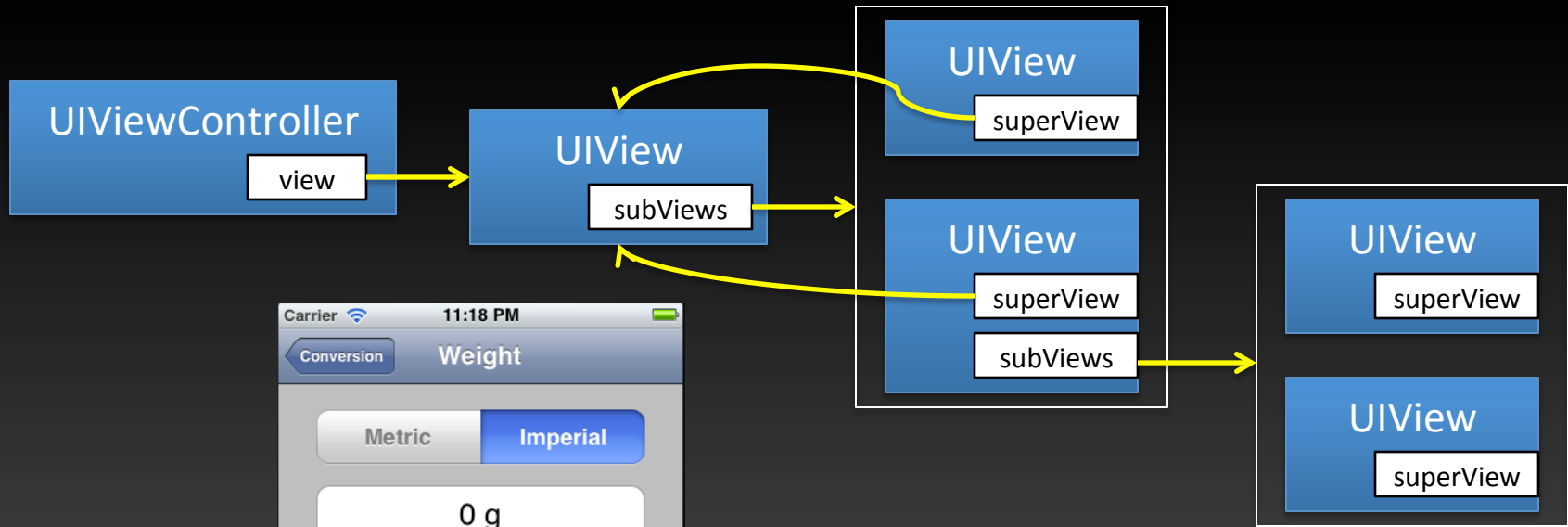




UIViewController

Quản lý tập các UIView (manage a view hierarchy)



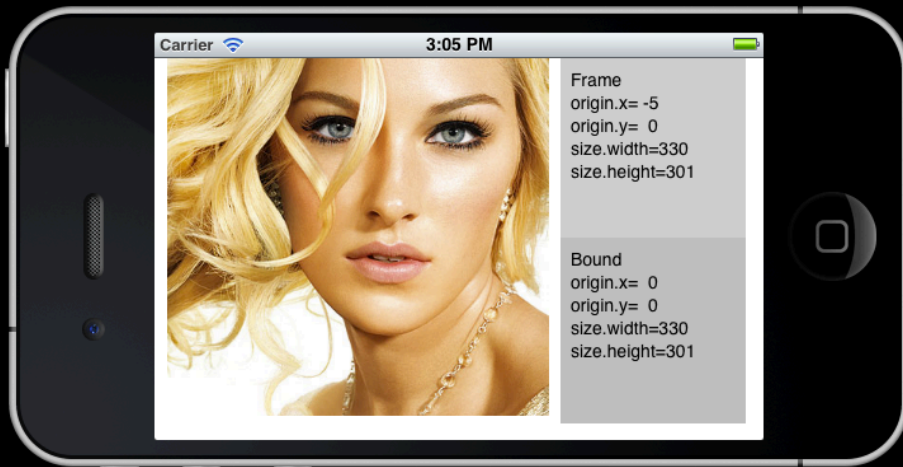
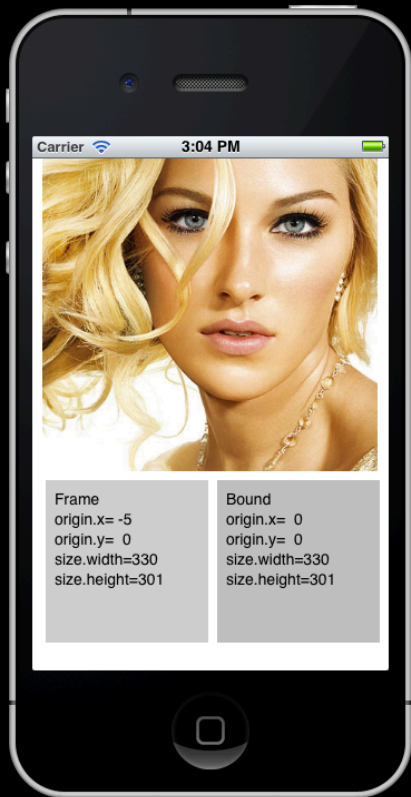


Chức năng của UIViewController #2

Hiển thị View bằng cách nạp vào file Nib.
Mở dự án “LoadMultipleViews” lần trước
ra xem lại nhé.

Chức năng của UIViewController #3

Xử lý sự kiện xoay ngang – dọc (rotation)



- (void)**willRotateToInterfaceOrientation:**
(UIInterfaceOrientation)toInterfaceOrientation duration:
(NSTimeInterval)duration;
- (void)**willAnimateRotationToInterfaceOrientation:**
(UIInterfaceOrientation)toInterfaceOrientation duration:
(NSTimeInterval)duration
- (void)**didRotateFromInterfaceOrientation:**
(UIInterfaceOrientation)fromInterfaceOrientation

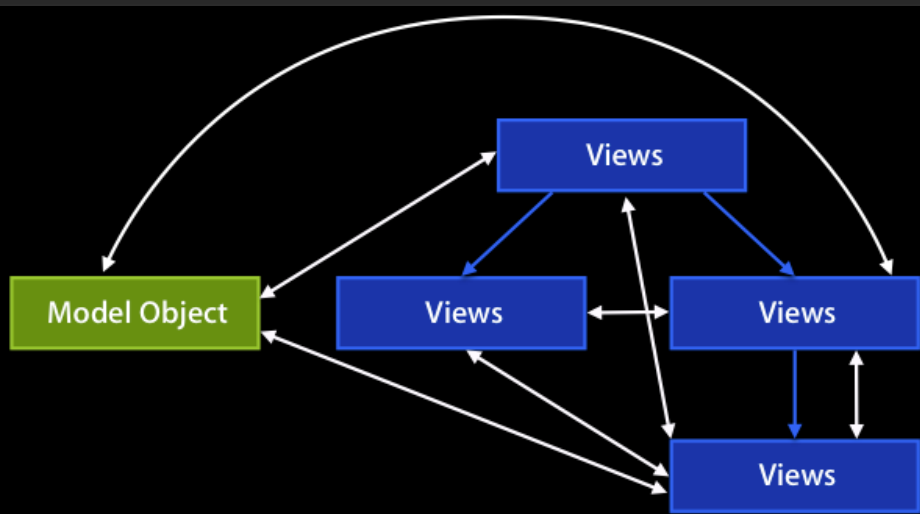
```
- (void) didRotateFromInterfaceOrientation:
(UIInterfaceOrientation)fromInterfaceOrientation
{
    [self layoutSubviews];
}

- (void)layoutSubviews {
    UIDeviceOrientation orientation = [[UIDevice currentDevice] orientation];
    if (orientation == UIDeviceOrientationLandscapeLeft || orientation ==
UIDeviceOrientationLandscapeRight)
    {
        debugText.frame = CGRectMake(320, 20, 146, 146);
        debugText2.frame = CGRectMake(320, 161, 146, 146);
    }

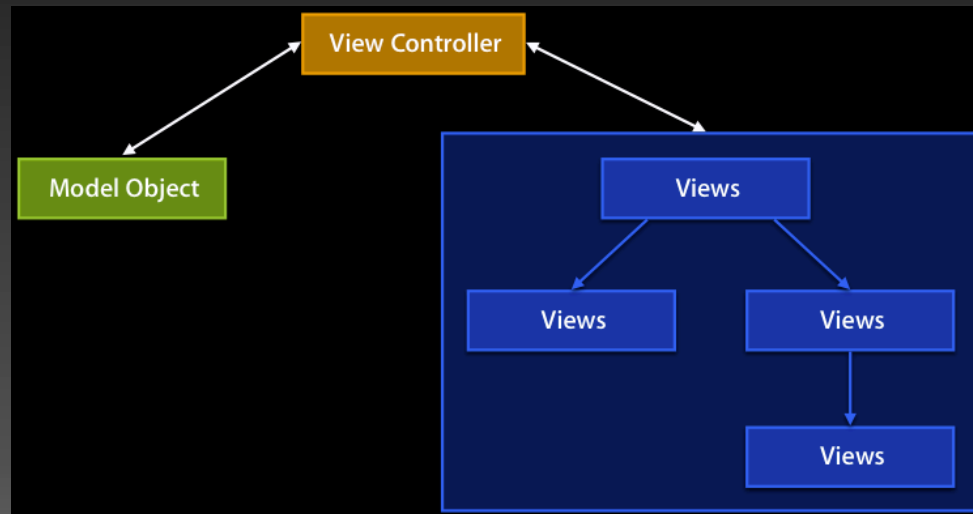
    if (orientation ==UIDeviceOrientationPortrait)
    {
        debugText.frame = CGRectMake(12, 309, 146, 146);
        debugText2.frame = CGRectMake(166, 309, 146, 146);
    }
}
```

Chức năng của UIViewController #4

Quản lý, xử lý tập trung các sự kiện thay vì phải phân phối xuống từng View trong View Hierarchy. Tuy nhiên trong vài trường hợp, viết code trong UIView để vẽ đồ họa, xử lý tương tác đa chạm sẽ gọn gàng hơn.



Rối !



Gọn

Chức năng của UINavigationController #5

Viết một lần để sử dụng lại nhiều lần:

- UINavigationController
- UITabBarController
- UISplitViewController
- UIPopOverController
- UIPageViewController
- CustomViewController có thể tìm trên GitHub rất nhiều !

Tổng kết lại – đừng quên !

- Quản lý tập các UIView
- Hiển thị View bằng cách nạp vào file Nib
- Xử lý thiết bị xoay ngang – dọc (rotation)
- Quản lý, xử lý tập trung các sự kiện thay vì phải phân phối xuống từng View trong View Hierarchy
- Tính đóng gói và sử dụng lại



Create a UIViewController

1. Create new project based on EmptyWindow template

Choose a template for your new project:

1

iOS

Application

Framework & Library

Other

Mac OS X

Application

Framework & Library

Application Plug-in

System Plug-in

Other

Master-Detail Application

Tabbed Application

OpenGL Game

Utility Application

Page-Based Application

Empty Application

Single View Application

Empty Application

This template provides a starting point for any application. It provides just an AppDelegate and a window.

Cancel

Previous

1

Choose options for your new project:

2

Product Name

CreateVCUsingIB

Company Identifier

techmaster.vn

Bundle Identifier

techmaster.vn.CreateVCUsingIB

Class Prefix

XYZ

Device Family

iPhone

☐ Use Core Data

☒ Use Automatic Reference Counting

☐ Include Unit Tests

Cancel

Previous

Next

Choose options for your new file:

Class
Subclass of
☐ Targeted for iPad
☒ With XIB for user interface

Cancel

Previous

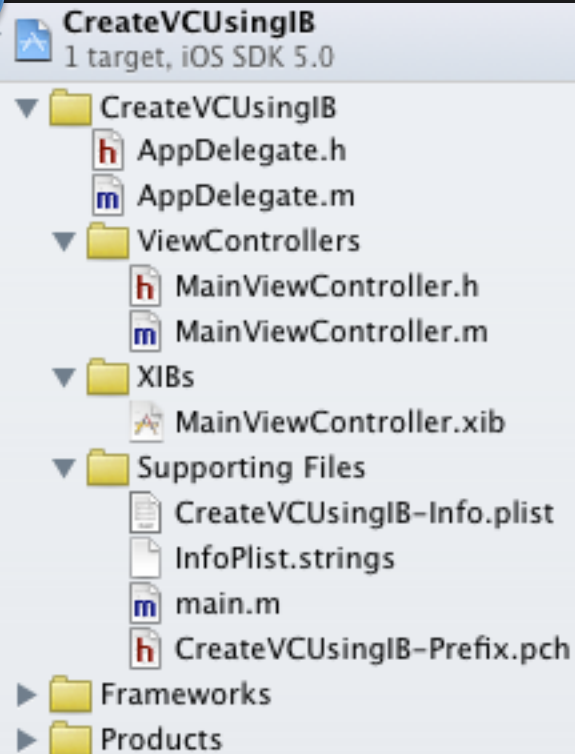
Next

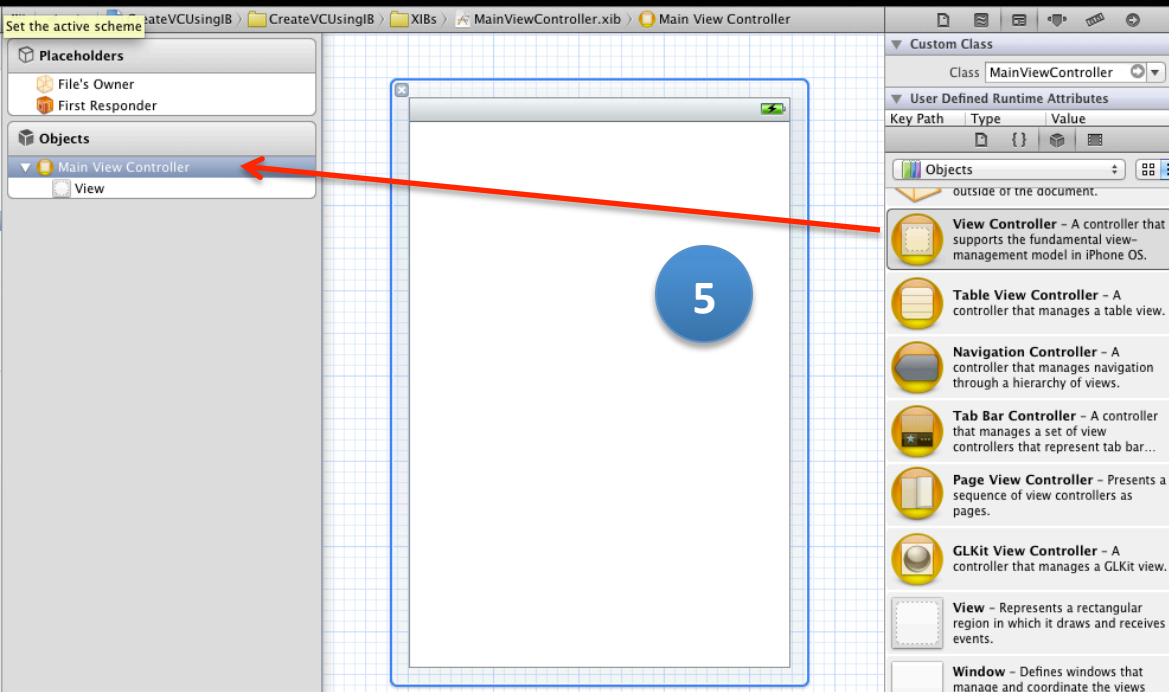
3

Create UIViewController with XIB

4

Rearrange project structure





6

Adjust custom class of MainViewController and File's Owner correctly

7

In AppDelegate.h add `#import "MainViewController.h"`

```
#import "AppDelegate.h"
#import "MainViewController.h"

@implementation AppDelegate

@synthesize window = _window;
- (BOOL)application:(UIApplication *)application
didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
{
    self.window = [[UIWindow alloc] initWithFrame:[UIScreen mainScreen]
bounds]];

    MainViewController *mainViewController = [[MainViewController alloc]
initWithNibName:@"MainViewController" bundle:nil];

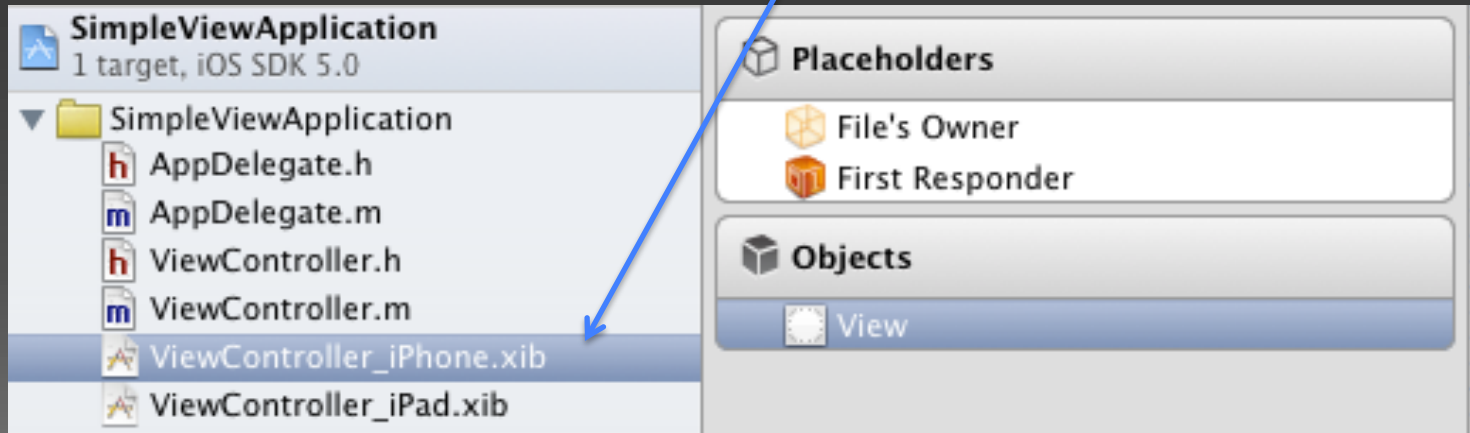
    self.window.rootViewController = mainViewController;

    [self.window makeKeyAndVisible];
    return YES;
}
```

initWithNibName

– (id) initWithNibName: (NSString *)nibName
bundle: (NSBundle *)nibBundle

```
self.viewController = [[ViewController alloc]  
initWithNibName:@"ViewController_iPhone"  
bundle:nil];
```



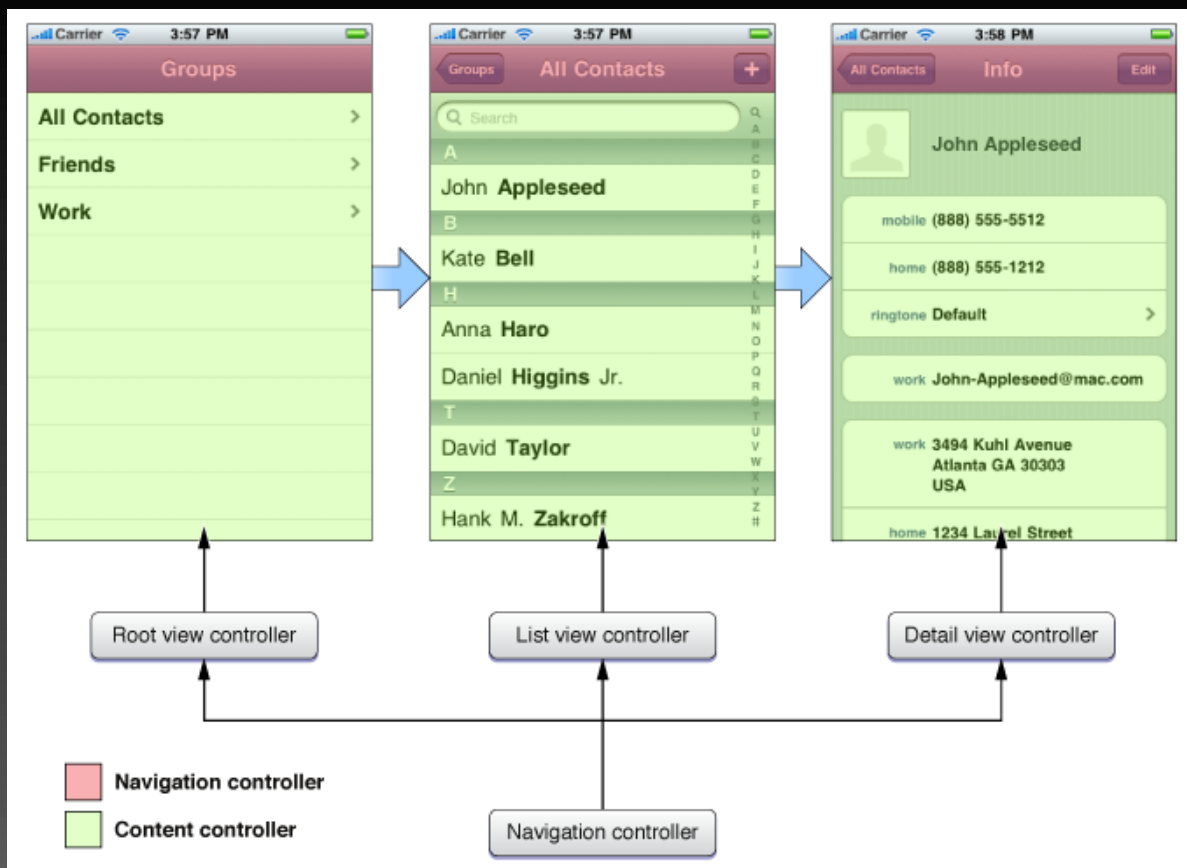
- **viewDidLoad** called after the view controller has loaded its view hierarchy into memory. Should **overwrite it** !
- **viewWillUnload** Called just before releasing the controller's view from memory
- **viewDidUnload** called after the view controller's view has been released. Overwrite to perform final cleanup

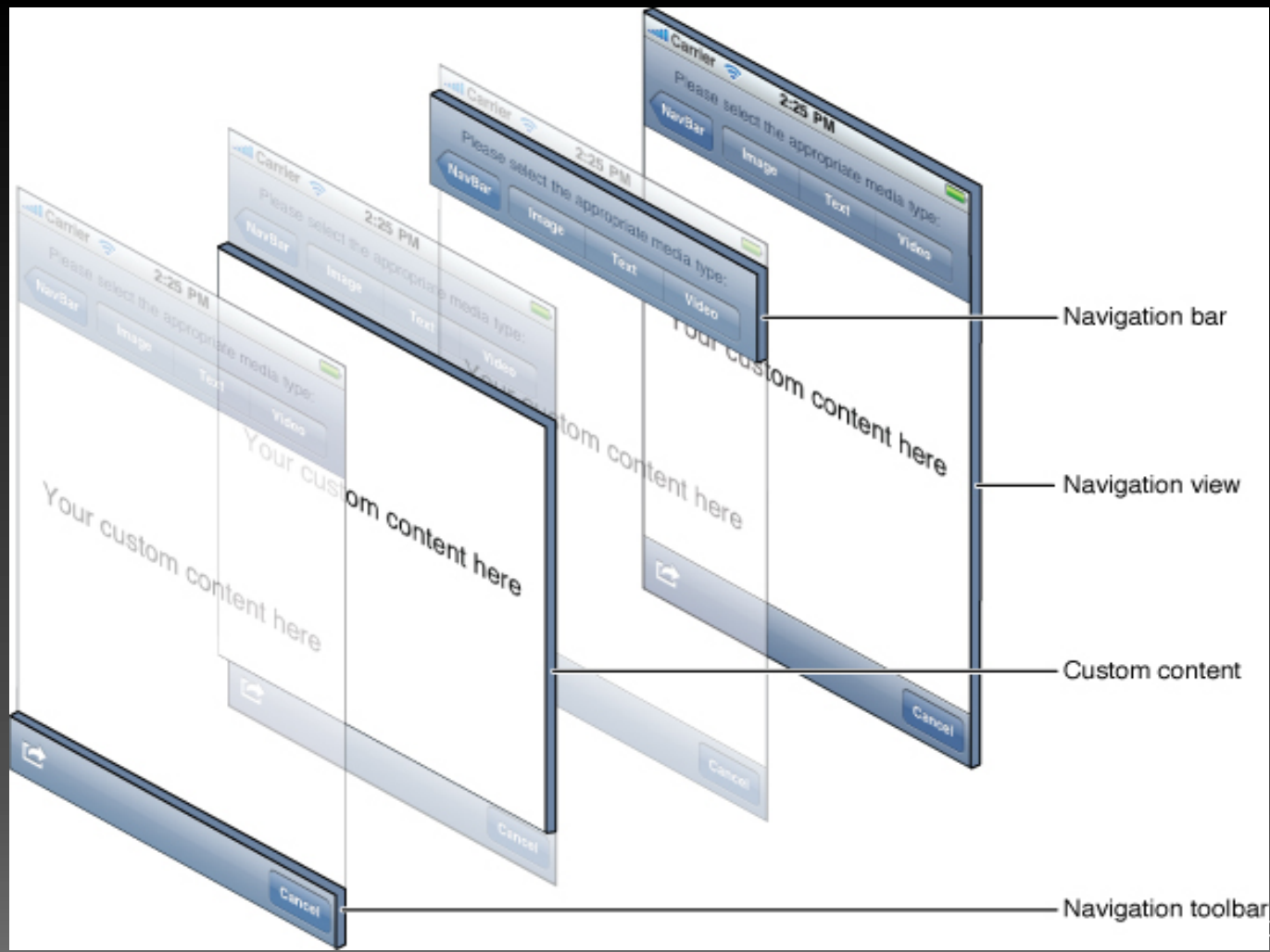
Responding to View Events

- (void) `viewWillAppear:` (BOOL) animated
- (void) `viewWillDisappear:`
(BOOL) animated `viewWillDisappear:`
- (void) `viewDidDisappear:` (BOOL) animated
- (void) `viewWillLayoutSubviews`
- (void) `viewDidLayoutSubviews`



UINavigationController



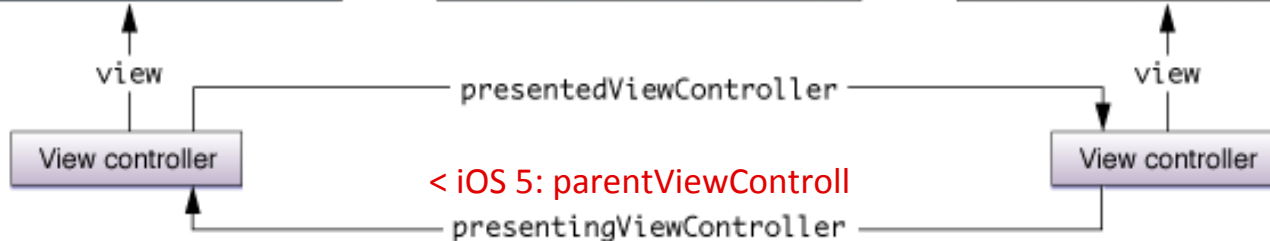
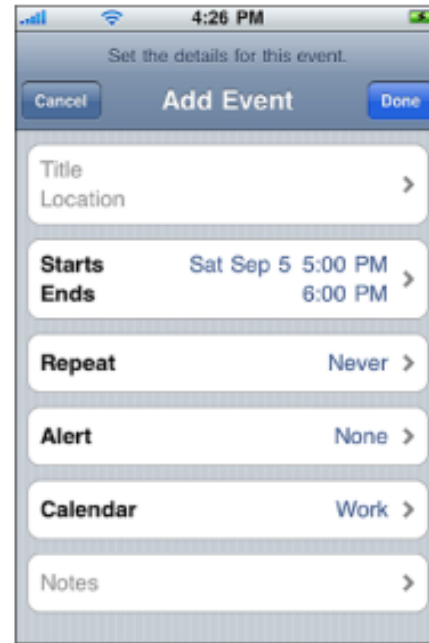
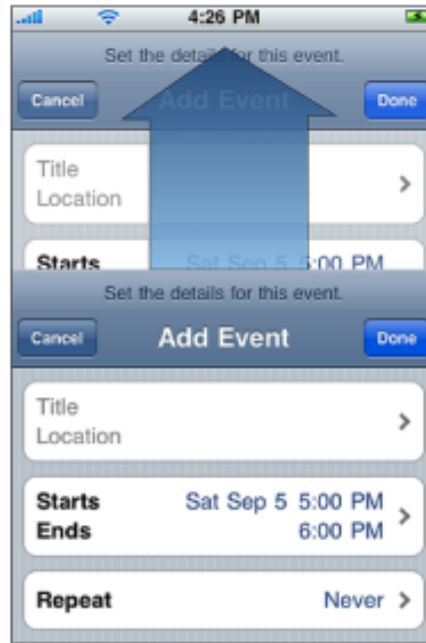




Demo CreateNavigationUsingIB

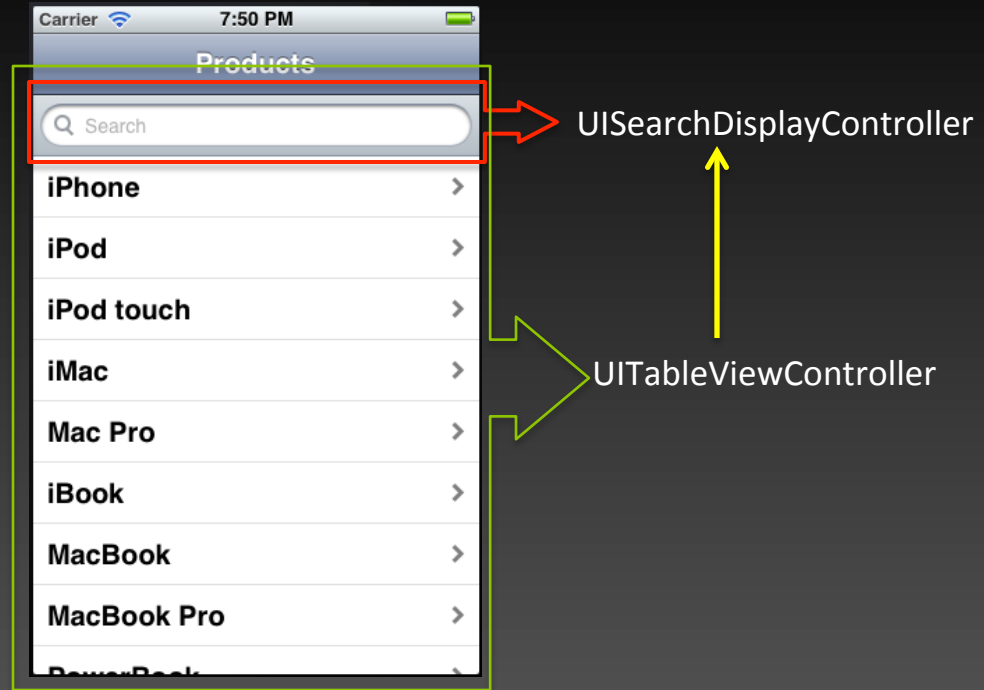
1. Create Empty project

Relationship between UIViewControllers

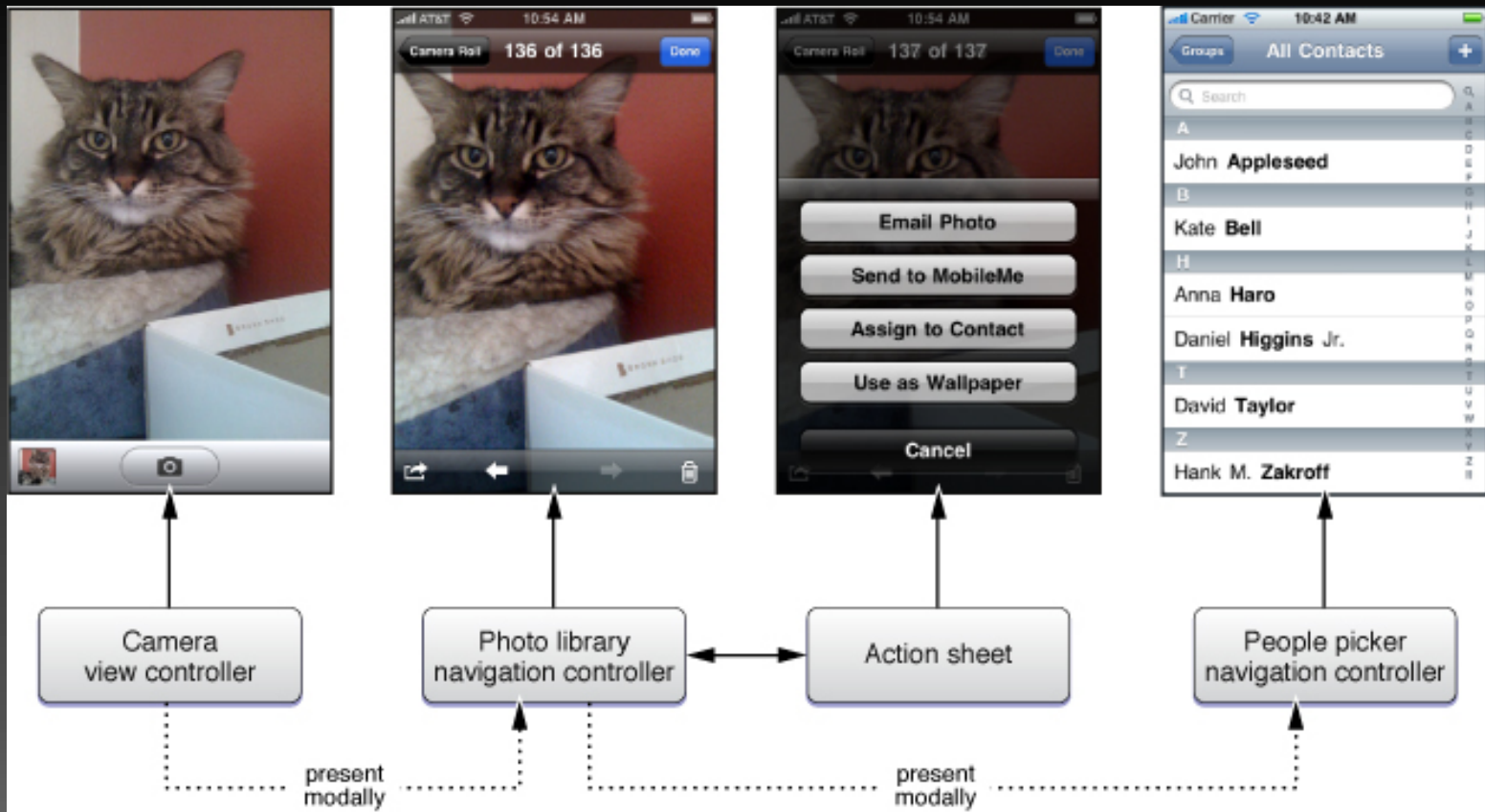


Property that points to related ViewController

- navigationController
- splitViewController
- tabBarController
- searchDisplayController
- modalViewController



Chained of modal view controllers

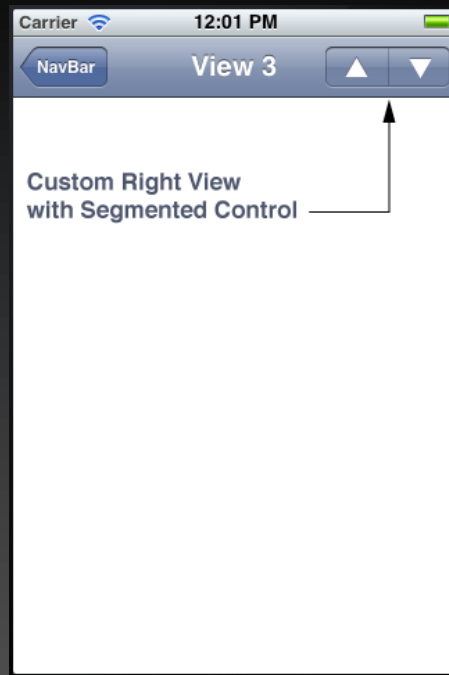
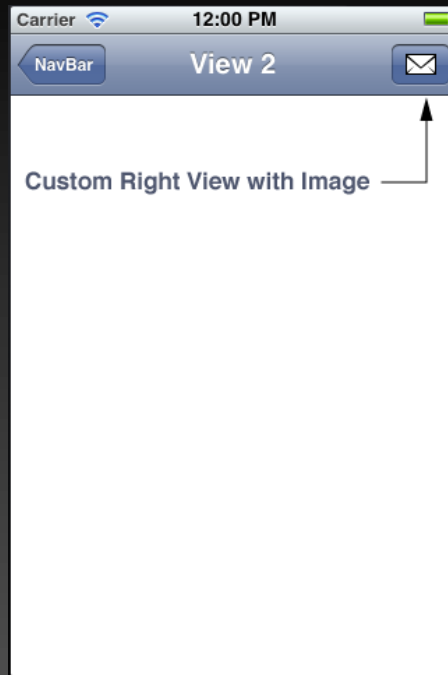
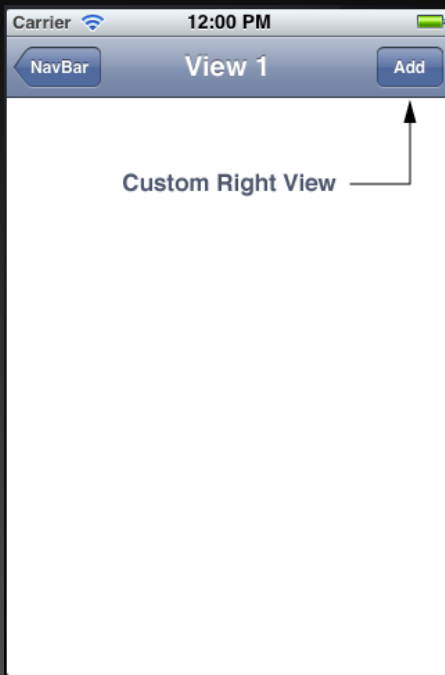
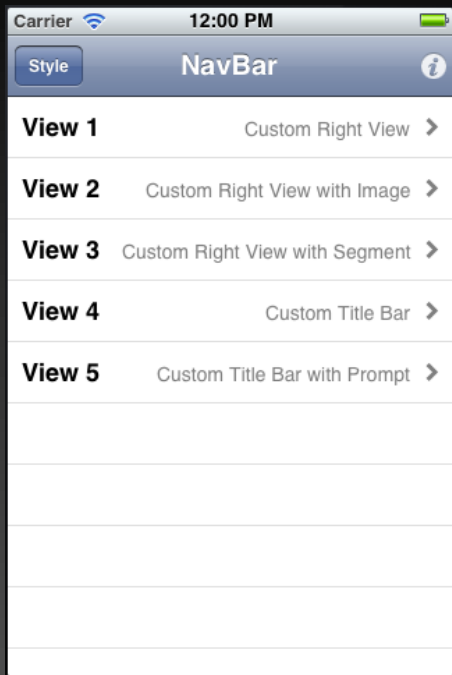


Many ways to create GUI

- Using project template (traditional IB or StoryBoard)



Demo



What we learnt from this example

- UINavigationController with UITableView

