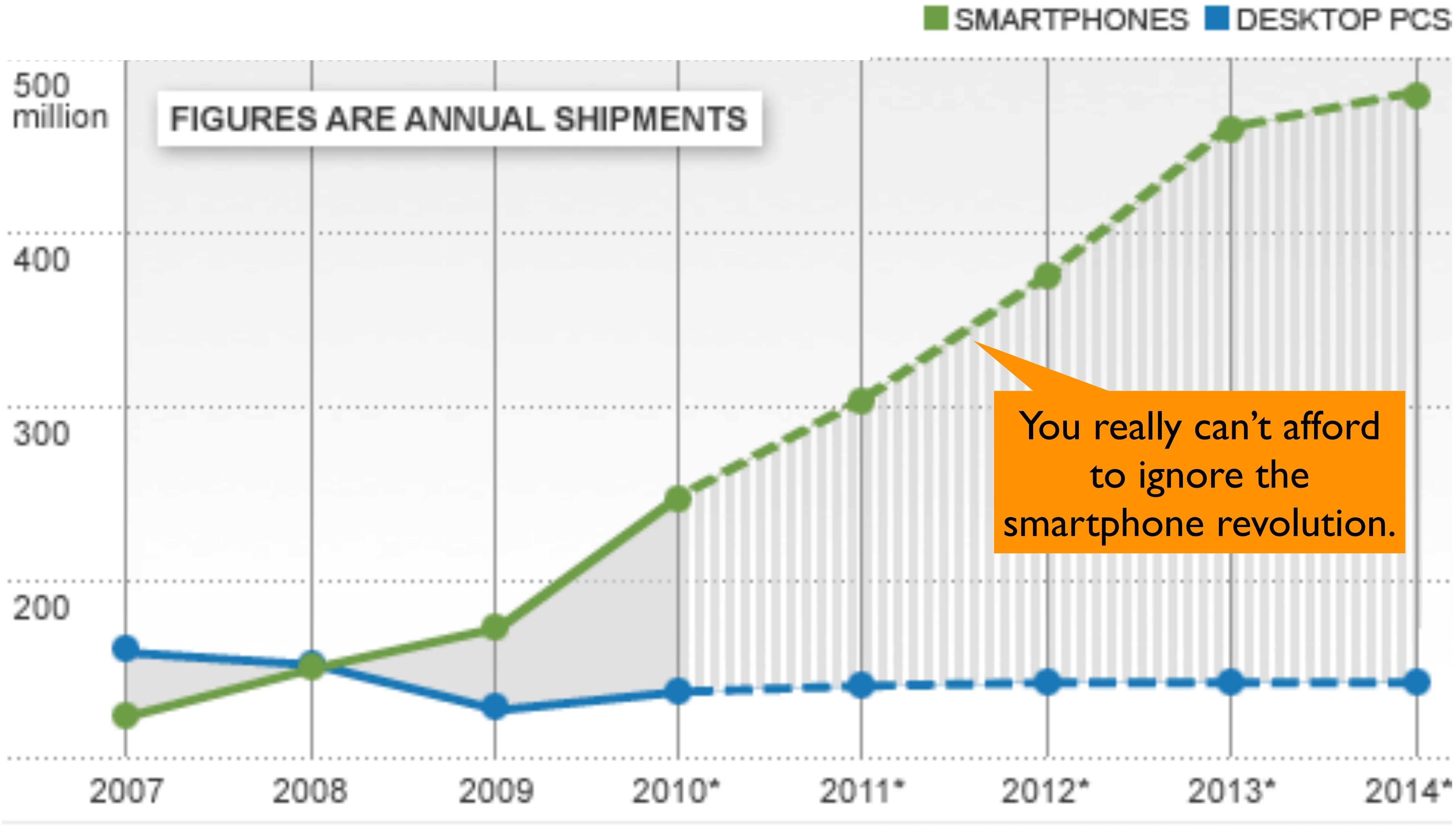
Developing for the iOS Platform

Engineer Bainomugisha

ebainomu@vub.ac.be

Software Languages Lab.
Vrije Universiteit Brussel, Belgium

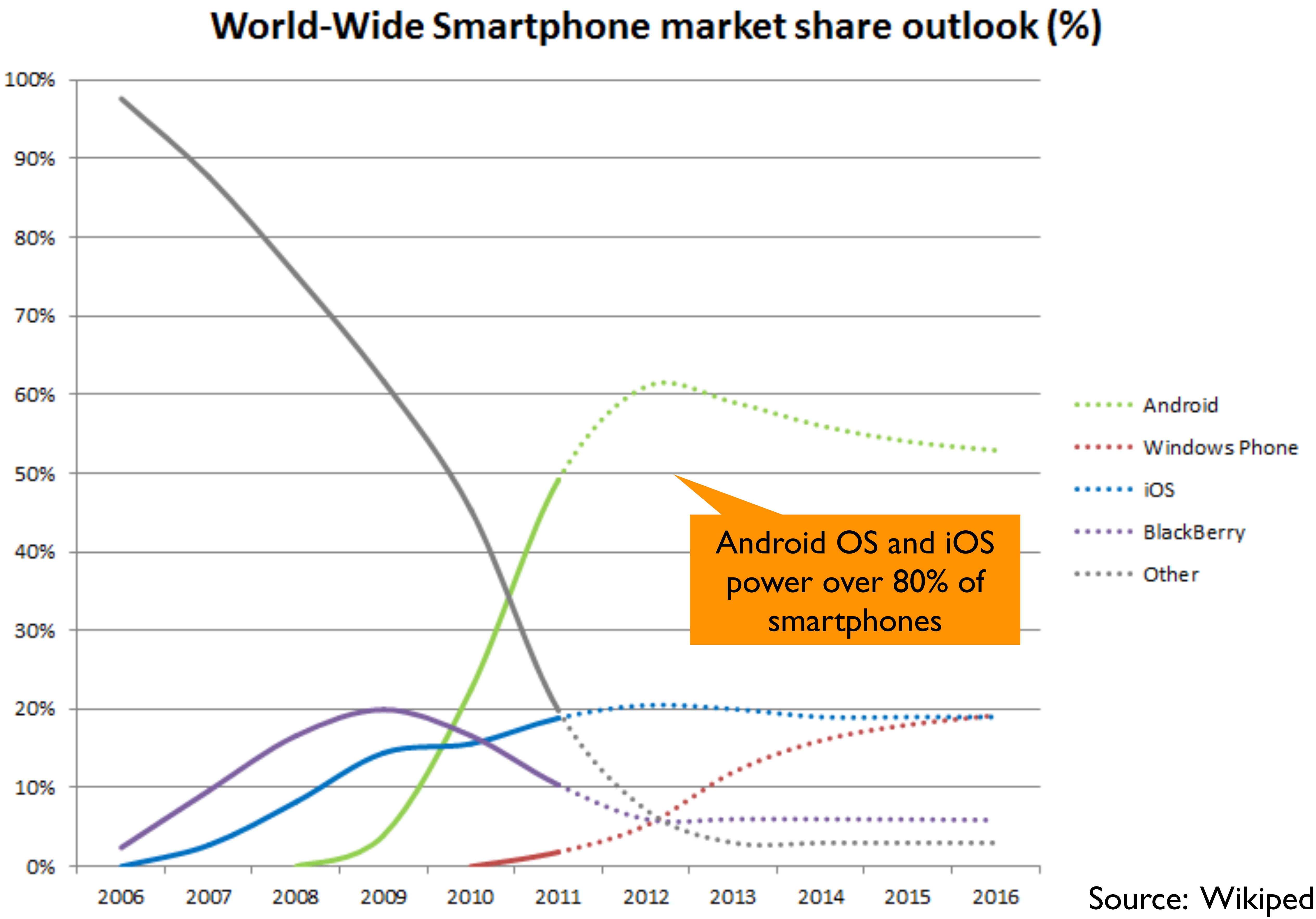
Smartphones Shipments vs PCs



*2010-2014 ARE FORECASTS; SOURCE: IDC

You really can't afford to ignore the smartphone revolution.

Source: International Data Corporation (IDC).



Android OS and iOS power over 80% of smartphones

Source: Wikipedia + International Data Corporation

What is iOS?

Apple’s mobile operating system that runs on iPhone, iPad and iPod touch devices.

Built for multi-touch interactions: responds to gestures (e.g., swiping, pinching, and tapping).

What is iOS?

Sensors enable building “intelligent” apps
(e.g., location-aware apps)

- GPS
- Compass
- Accelerometer
- Camera

Comes with a Number of Built-in Apps

Users cannot remove built-in apps.

Photos, Calendar, Mail, SMS, Music, Maps, ...

Software Development Tools for the iOS

The iOS Software Development Kit (SDK) contains the tools needed to develop native apps.

Xcode is the development environment.

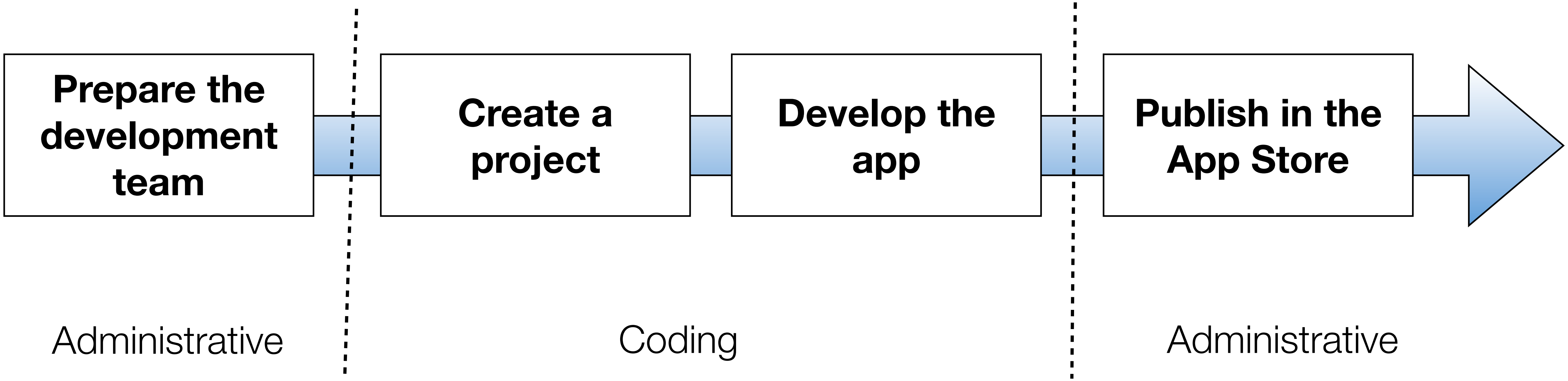
Freely available for download from Apple's developer portal

developer.apple.com

iOS Technologies are Packaged as Frameworks

- Maps
- Media Player
- Address Book
- Games
- Passbook Kit
- Core Location
- Image Kit
- Facebook
- UI Kit

Development Process Overview



Prepare the development team

Create a project

Develop the app

Publish in the App Store

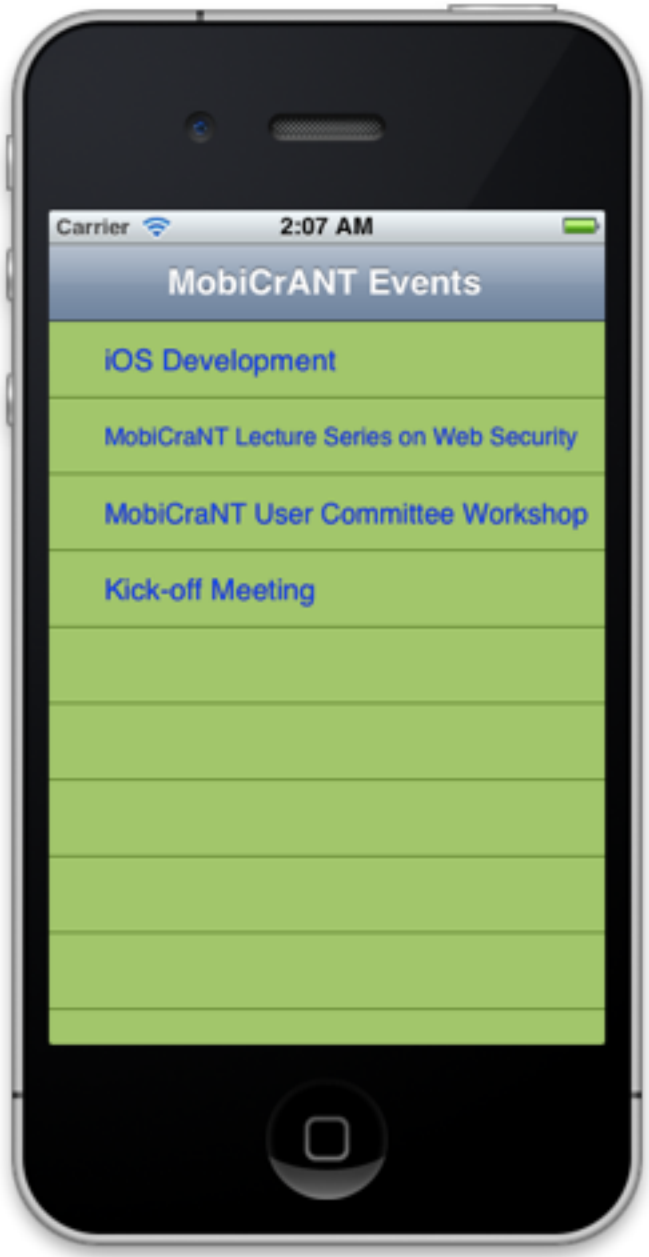
Administrative

Coding

Administrative

Two Kinds of iOS Apps

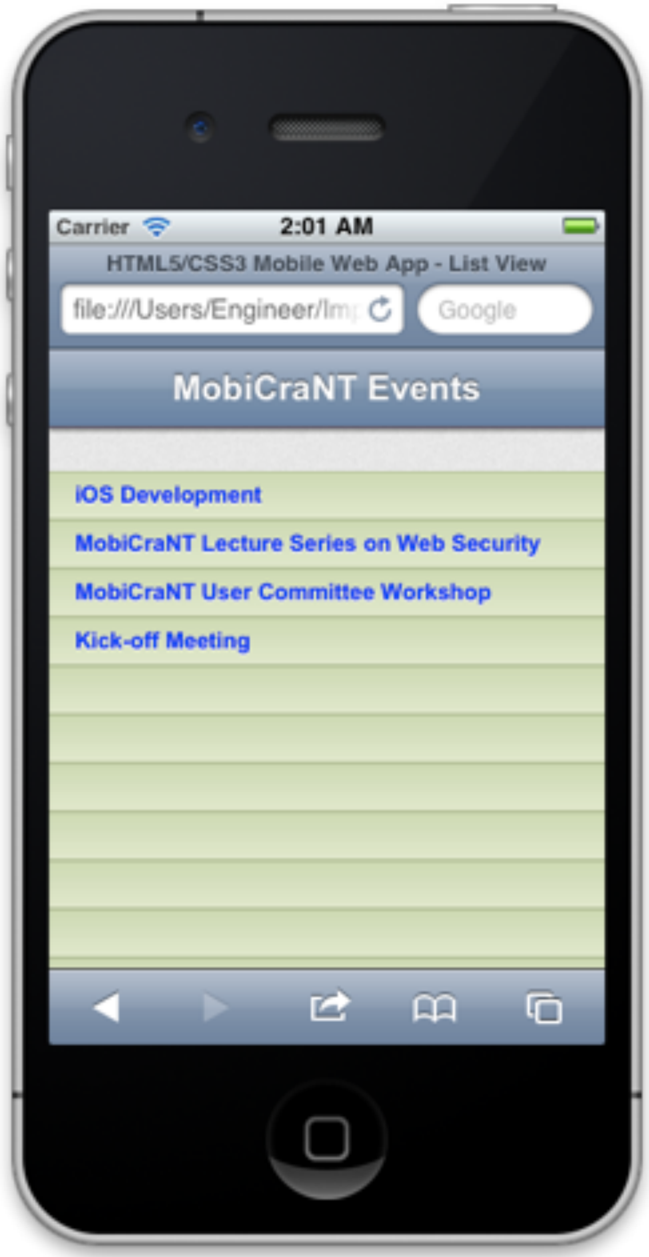
Native iOS apps



MobiCrANT Events

- iOS Development
- MobiCraNT Lecture Series on Web Security
- MobiCraNT User Committee Workshop
- Kick-off Meeting

Web apps

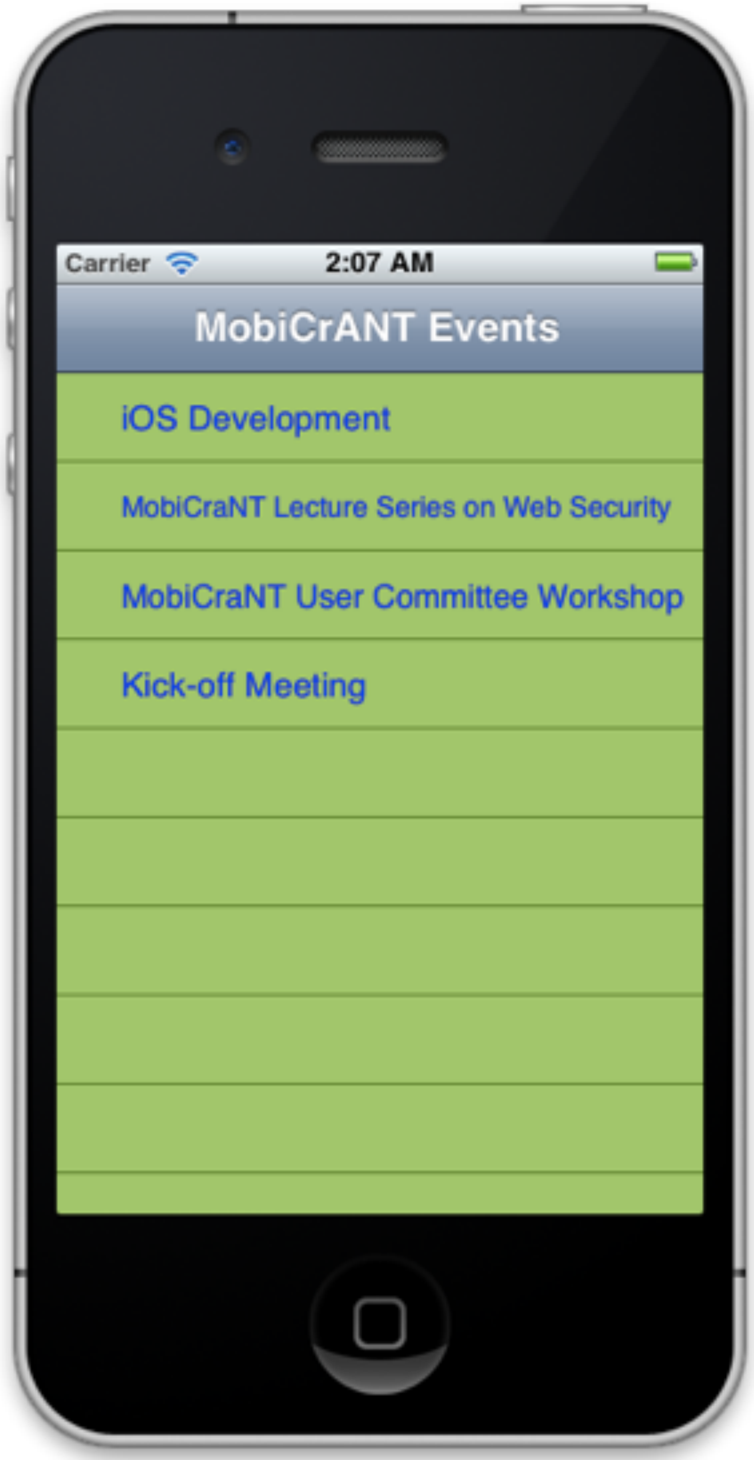

Carrier
2:01 AM
HTML5/CSS3 Mobile Web App - List View
file:///Users/Engineer/Im
Google

MobiCraNT Events

- iOS Development
- MobiCraNT Lecture Series on Web Security
- MobiCraNT User Committee Workshop
- Kick-off Meeting

Two Kinds of iOS Apps

Native iOS apps



MobiCrANT Events

iOS Development

MobiCraNT Lecture Series on Web Security

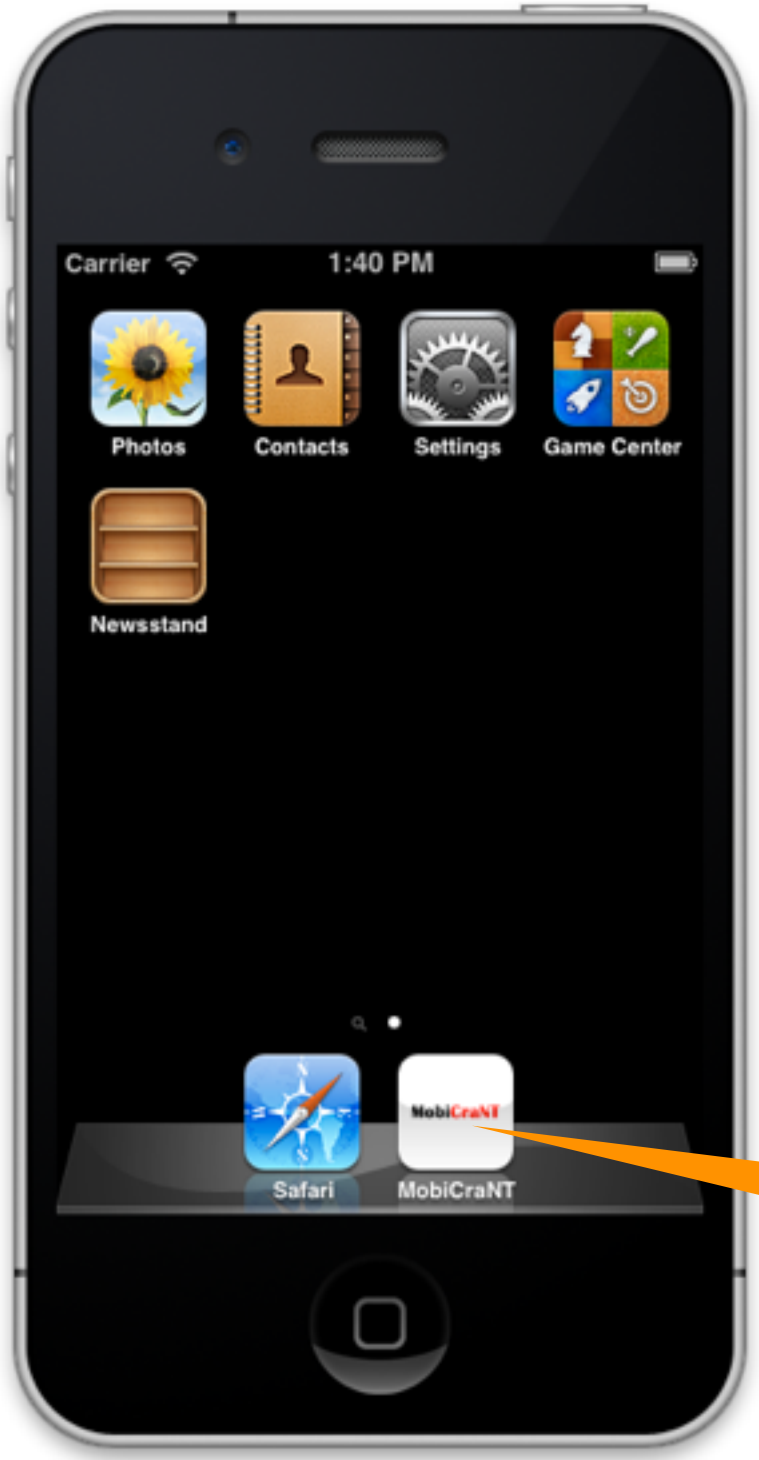
MobiCraNT User Committee Workshop

Kick-off Meeting

- Resemble the built-in apps.
- Are built using Objective-C programming language .
- Have access to the device’s hardware capabilities (e.g., GPS).
- Distributed via app store.

Two Kinds of iOS Apps

Native iOS apps

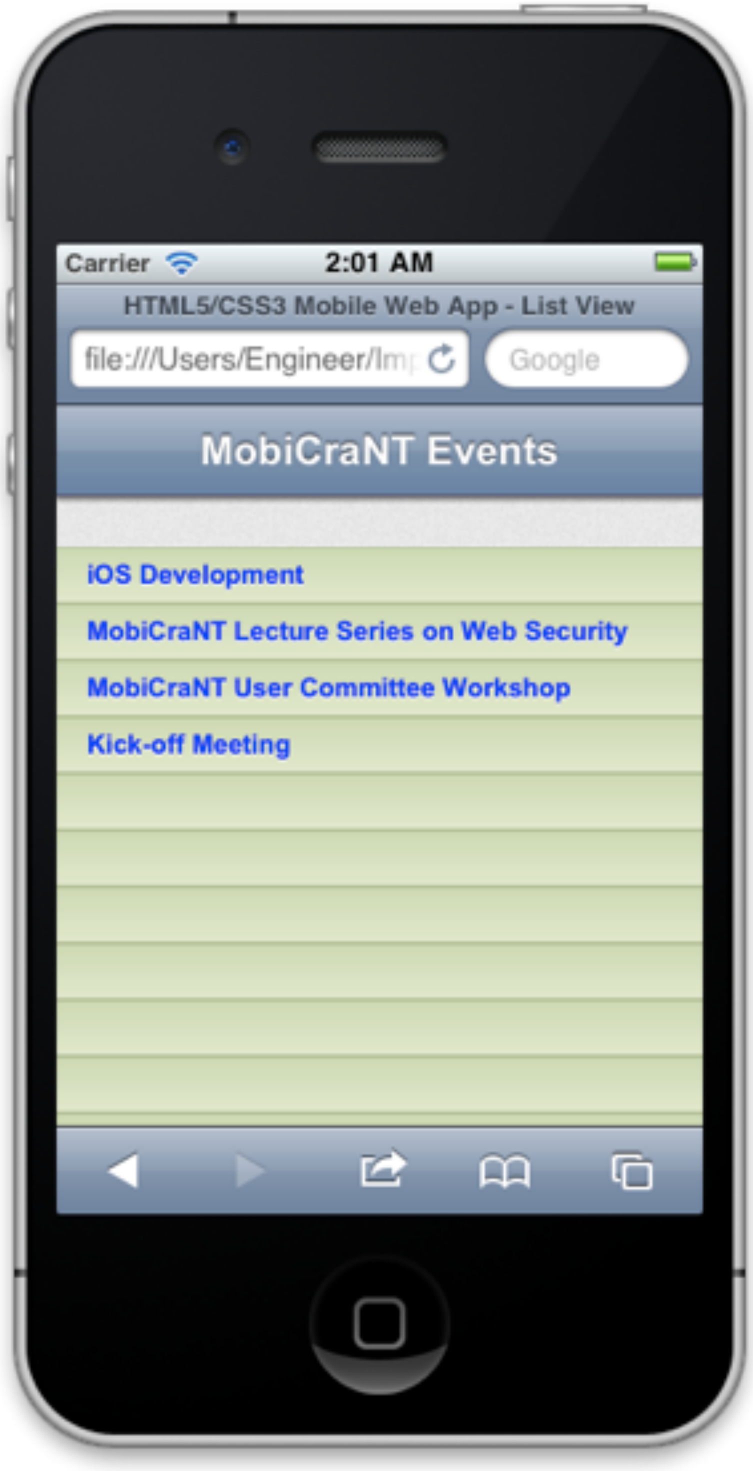


- Resemble the built-in apps.
- Are built using Objective-C programming language .
- Have access to the device’s hardware capabilities (e.g., GPS).
- Distributed via app store.

Installed on the device like a built-in app.

Two Kinds of iOS Apps

Web apps


Carrier
2:01 AM
HTML5/CSS3 Mobile Web App - List View
file:///Users/Engineer/Im
Google

MobiCraNT Events

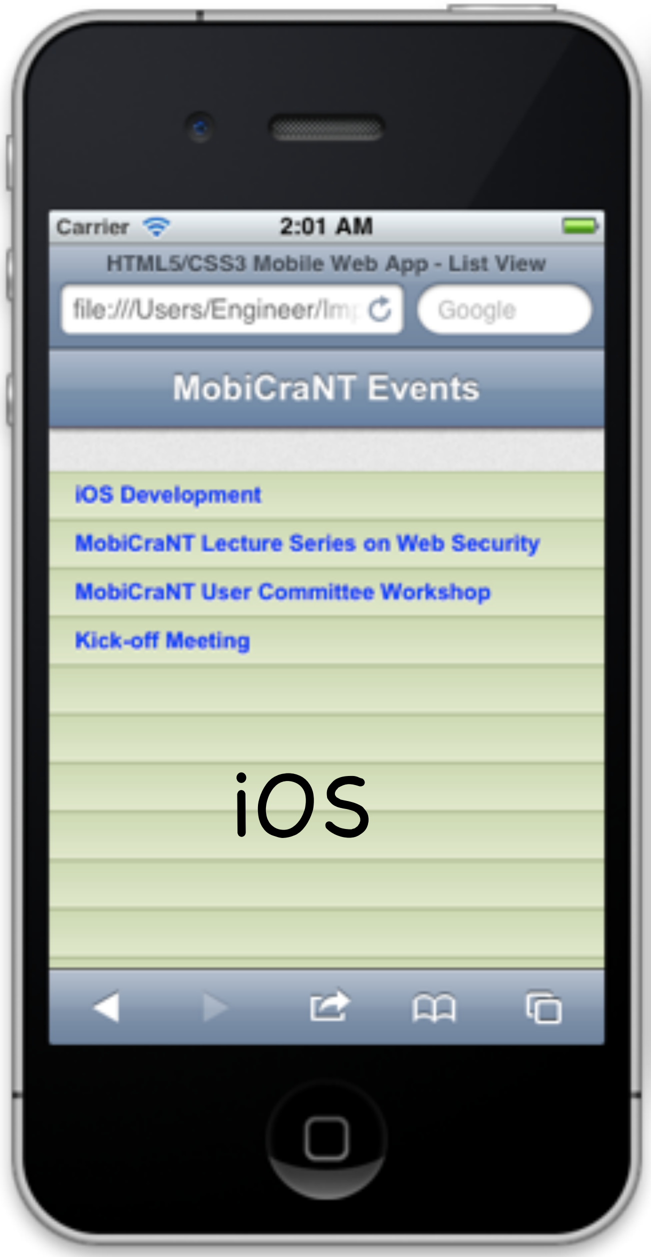
- iOS Development
- MobiCraNT Lecture Series on Web Security
- MobiCraNT User Committee Workshop
- Kick-off Meeting

- Run inside a web browser
- Are built using HTML, CSS, HTML5, and JavaScript.
- Limited access to the device's hardware capabilities (e.g., GPS).
- Run slower than native apps.

Two Kinds of iOS Apps

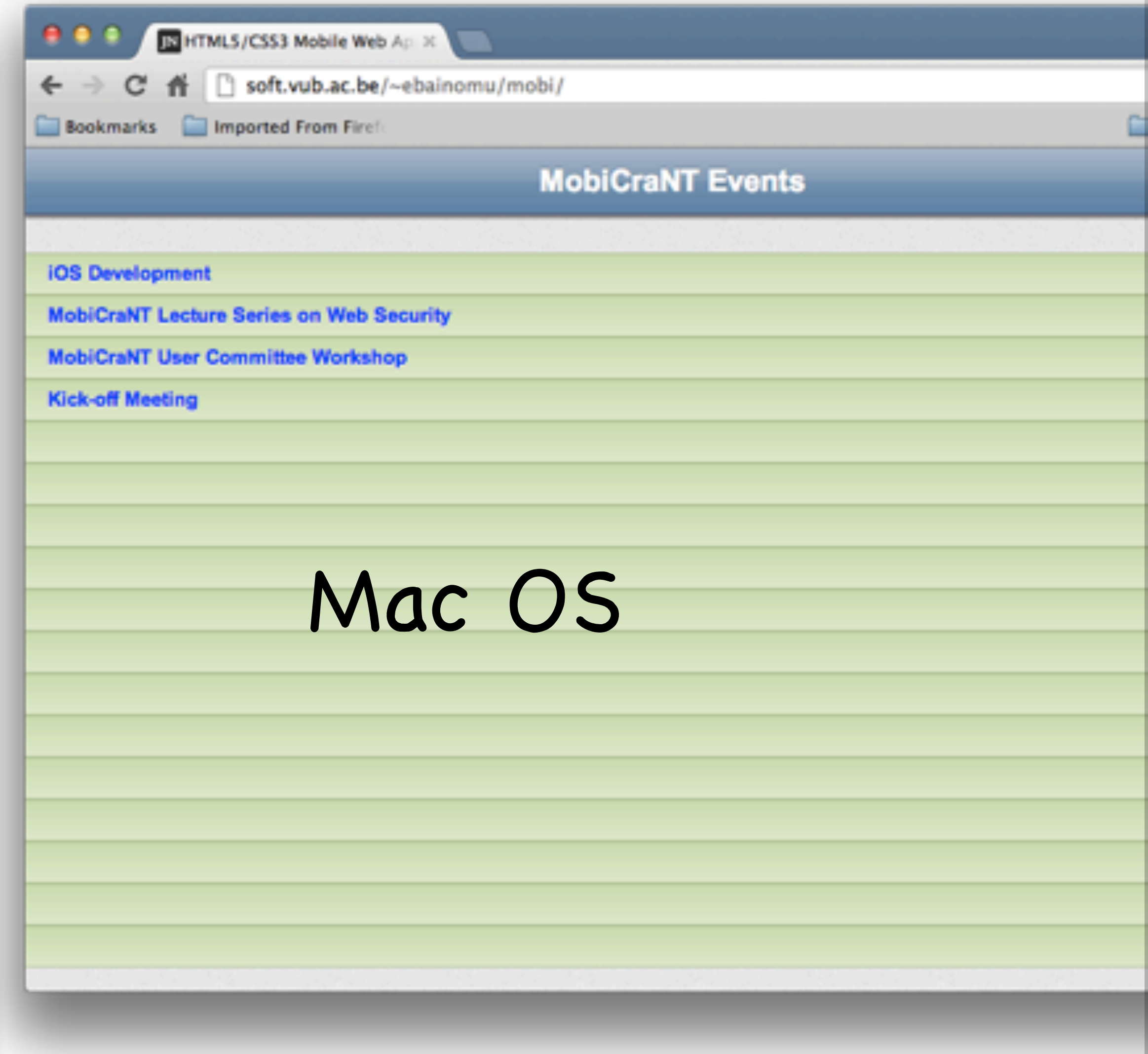
Web apps

“Write once, run anywhere (WORA)”


Carrier
2:01 AM
HTML5/CSS3 Mobile Web App - List View
file:///Users/Engineer/Im
Google
iOS

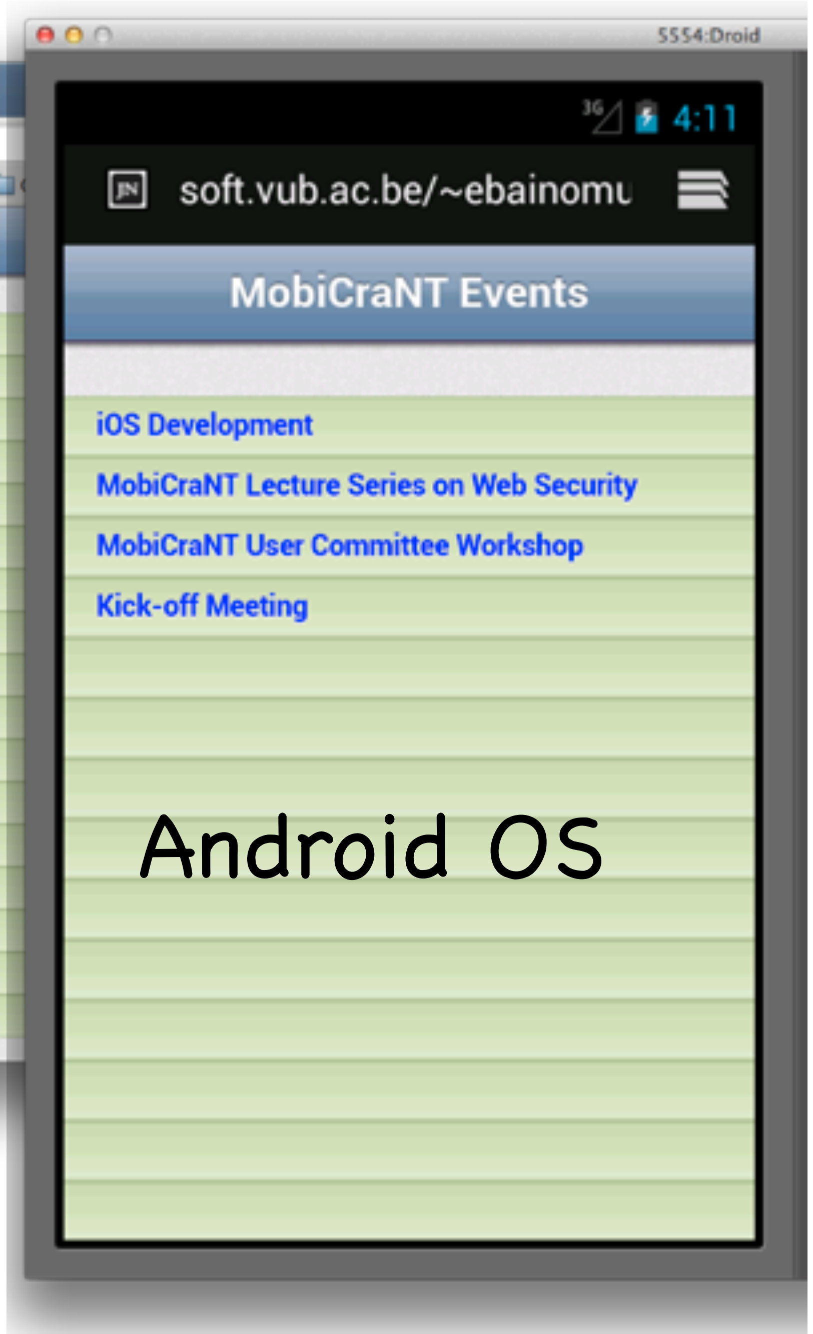
MobiCraNT Events

- iOS Development
- MobiCraNT Lecture Series on Web Security
- MobiCraNT User Committee Workshop
- Kick-off Meeting



MobiCraNT Events

- iOS Development
- MobiCraNT Lecture Series on Web Security
- MobiCraNT User Committee Workshop
- Kick-off Meeting

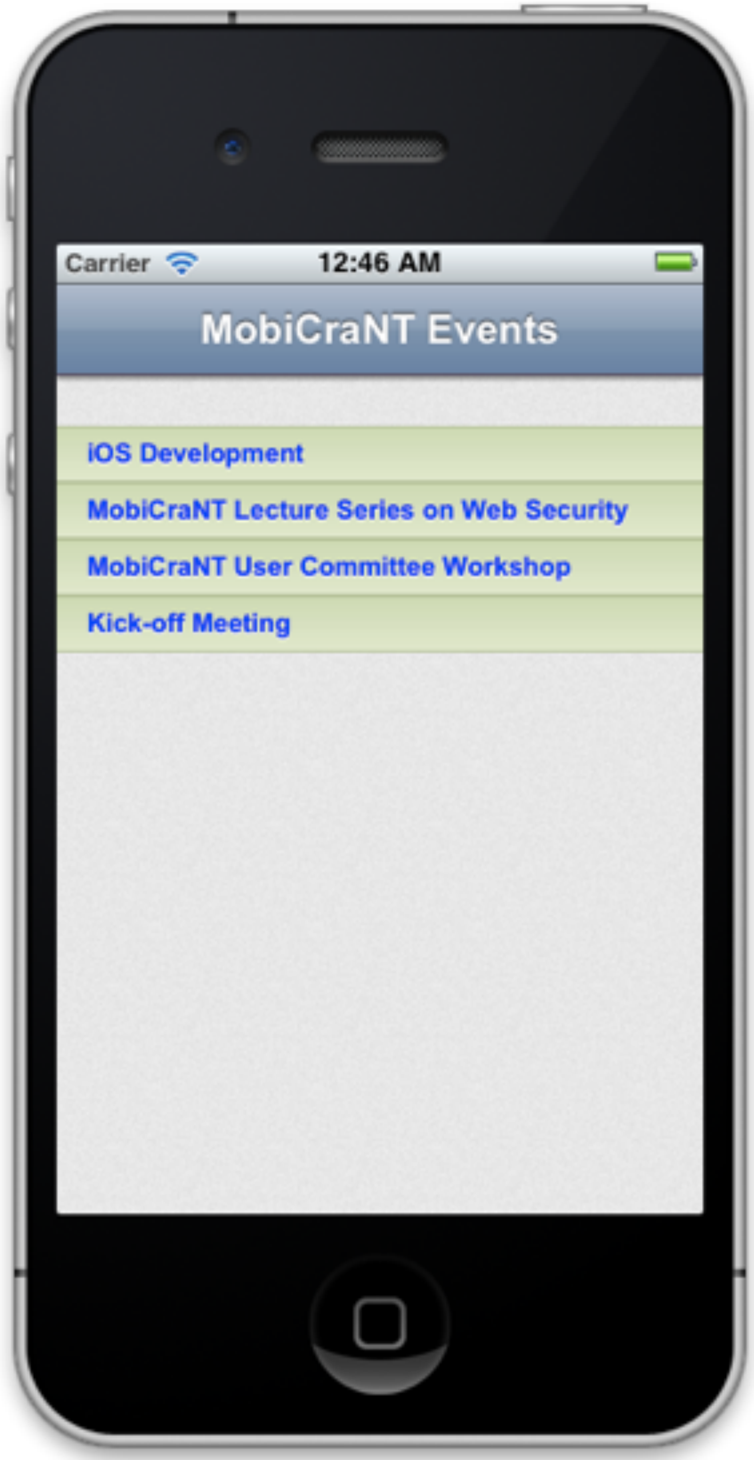


MobiCraNT Events

- iOS Development
- MobiCraNT Lecture Series on Web Security
- MobiCraNT User Committee Workshop
- Kick-off Meeting

Two Kinds of iOS Apps

Native apps + Web apps



Carrier 12:46 AM

MobiCraNT Events

iOS Development

MobiCraNT Lecture Series on Web Security

MobiCraNT User Committee Workshop

Kick-off Meeting

- Possible to embed web content in a native app.

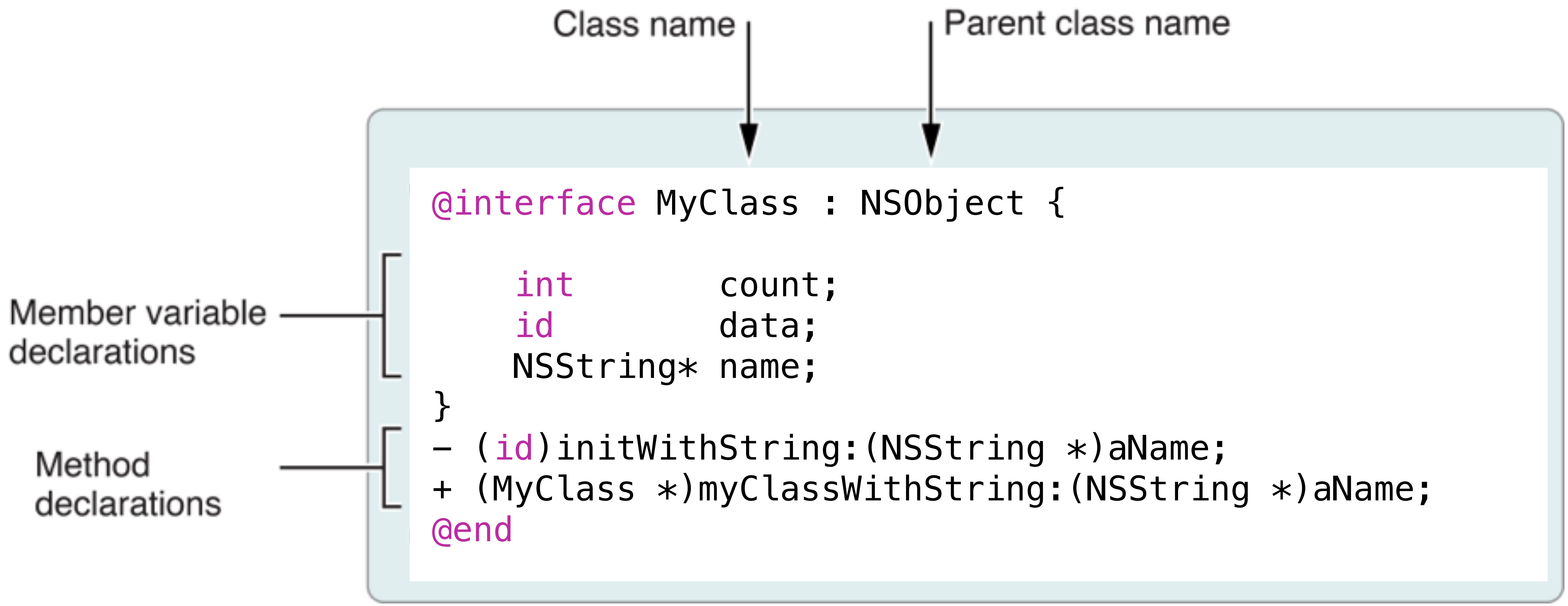
Objective-C Programming Language

```
 Implement viewDidLoad to do additional
(void)viewDidLoad {
CGRect toolbarFrame;
NSString *path;
[super viewDidLoad];

CGRect bounds = [[UIScreen mainScr
UIImageView *imageView = (UIImage
imageView.contentMode = UIViewCon
imageView.backgroundColor = [UIC

path = [[NSBundle mainBundle] p
imageView.image = [UIImage imag

if(imageView.image)
  self.imageURL = [NSURL file

if (UI_USER_INTERFACE_IDIOM(
  toolbarFrame = CGRectMake(
}else{
  toolbarFrame = CGRectMake
}

UIToolbar *aToolbar = [[[U
aToolbar.barStyle = UIBar
```

- Object-oriented programming language.
- Easy to learn— if you have experience with other O-O languages such as Java or C++.
- Objective-C is a superset of C.
- Simple, small, powerful.

Objective-C Programming Language (1 of 4)

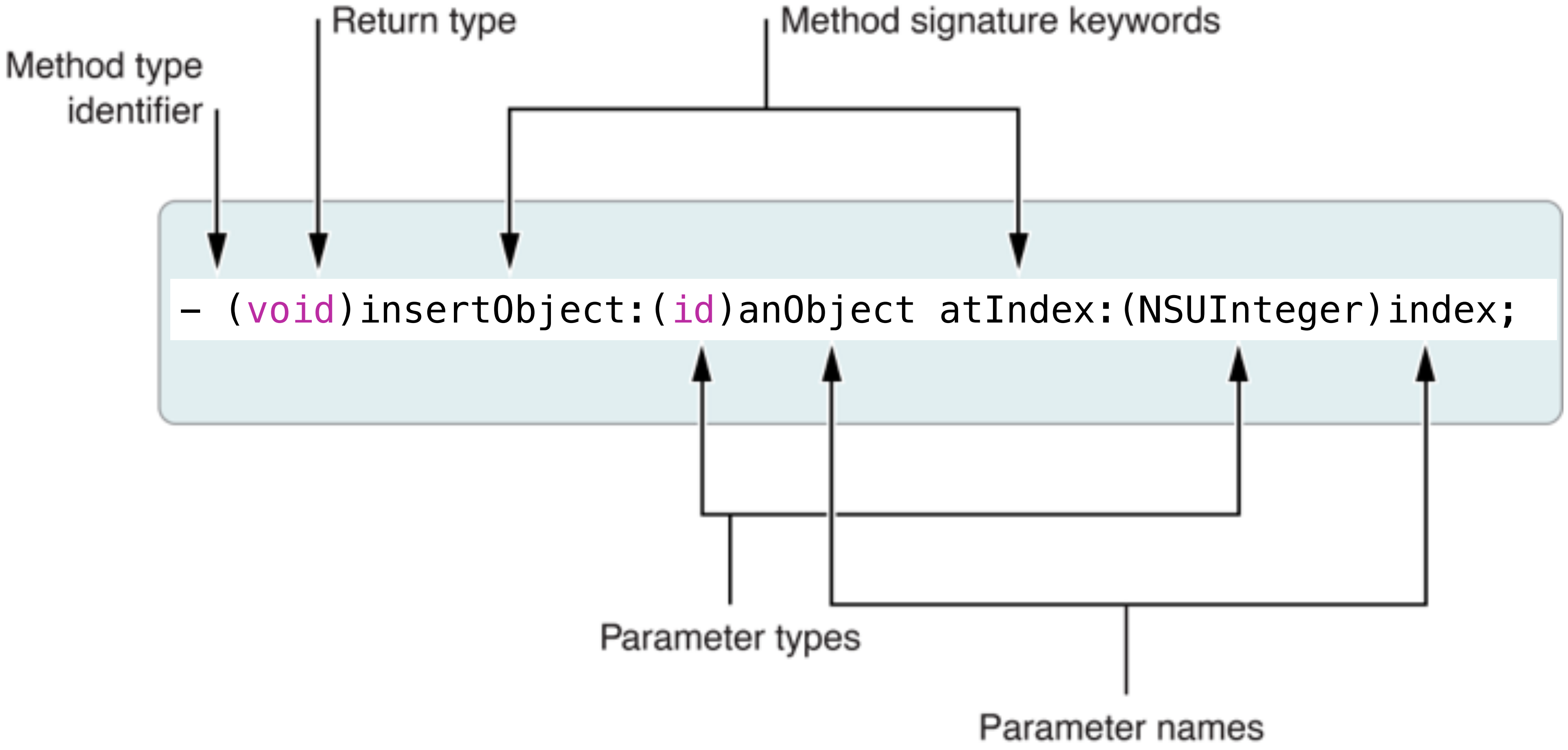
Class specification: **the interface** + the implementation



```
@interface MyClass : NSObject {

    int        count;
    id         data;
    NSString*  name;
}
- (id)initWithString:(NSString *)aName;
+ (MyClass *)myClassWithString:(NSString *)aName;
@end
```

Objective-C Programming Language (2 of 4)

Method Declaration



```
- (void)insertObject:(id)anObject atIndex:(NSUInteger)index;
```

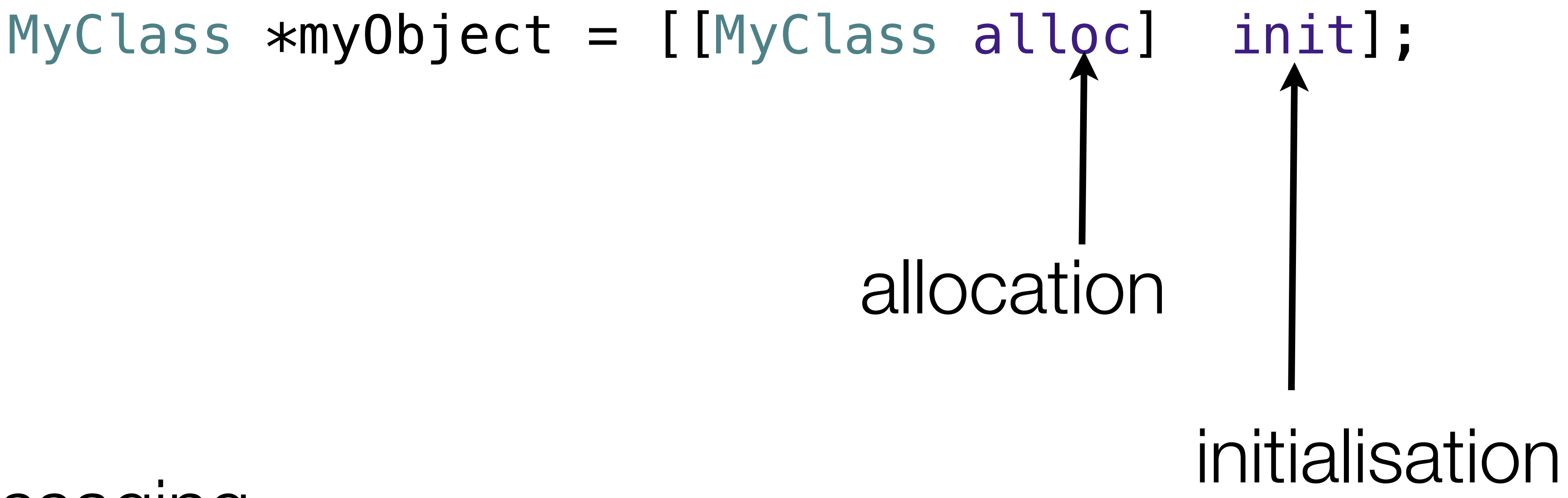
Objective-C Programming Language (3 of 4)

Class specification: the interface + **the implementation**

```
#import "MyClass.h"

@implementation MyClass
- (id)initWithString:(NSString *)aName
{
    // method implementation code goes here.
}
+ (MyClass *)myClassWithString:(NSString *)aName
{
    // method implementation code goes here.
}
@end
```

Objective-C Programming Language (4 of 4)

Object creation: allocation and initialisation.

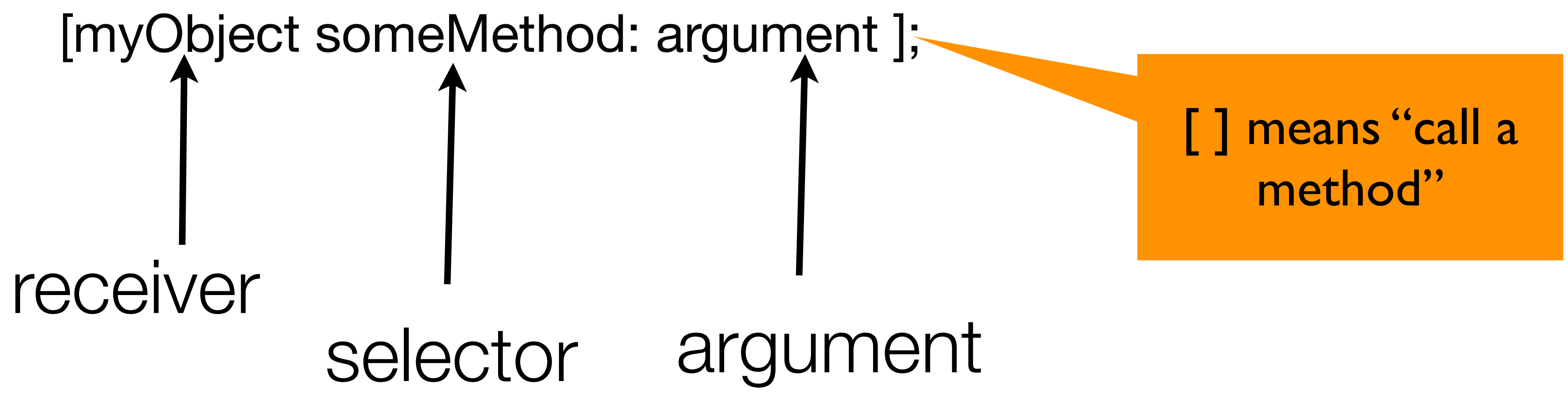


```
MyClass *myObject = [[MyClass alloc]  init];
```

allocation

initialisation

Messaging



```
[myObject someMethod: argument ];
```

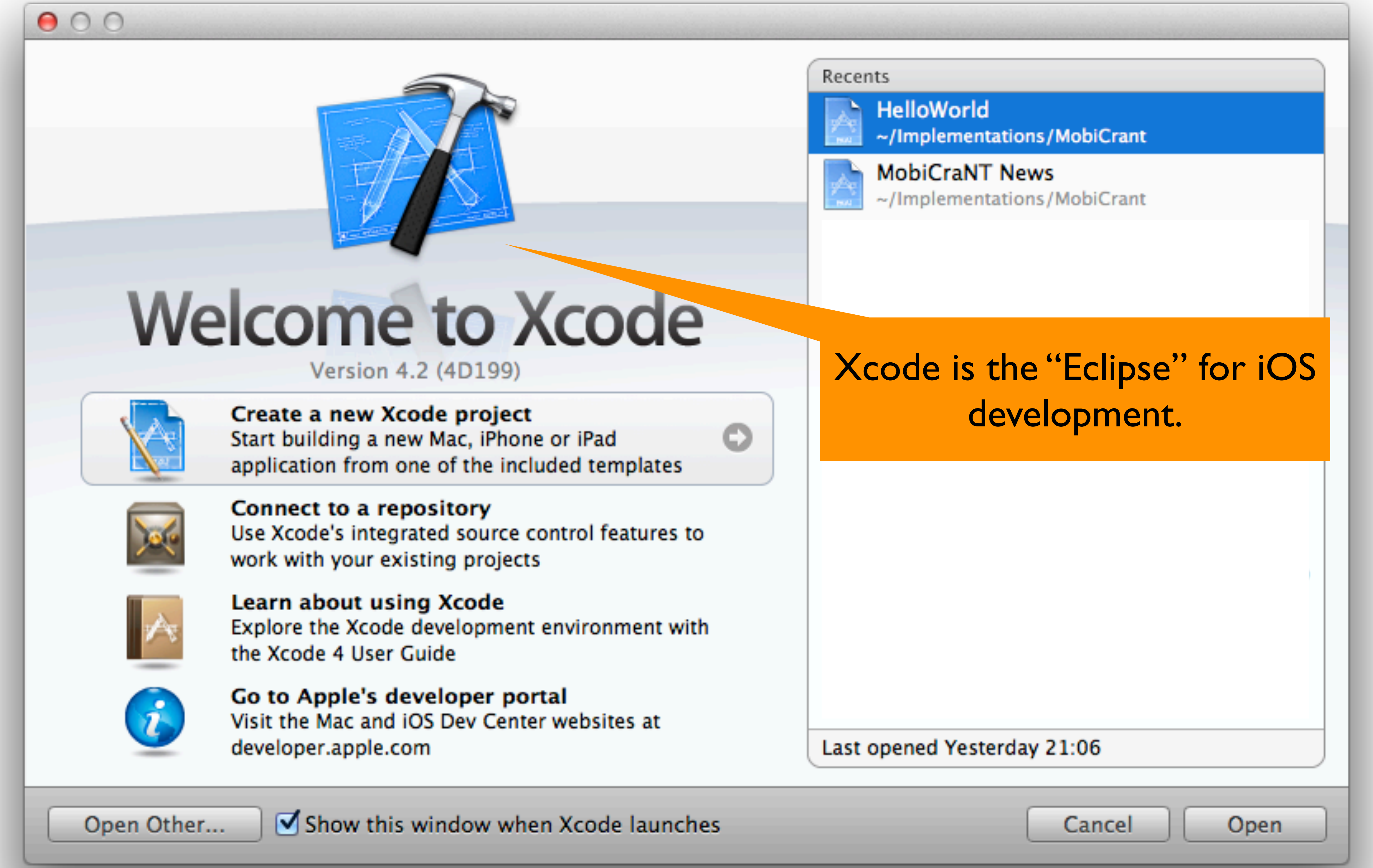
receiver

selector

argument

[ ] means "call a method"

Getting Started with iOS Development



Welcome to Xcode

Version 4.2 (4D199)

Create a new Xcode project
Start building a new Mac, iPhone or iPad application from one of the included templates

Connect to a repository
Use Xcode's integrated source control features to work with your existing projects

Learn about using Xcode
Explore the Xcode development environment with the Xcode 4 User Guide

Go to Apple's developer portal
Visit the Mac and iOS Dev Center websites at developer.apple.com

Recents

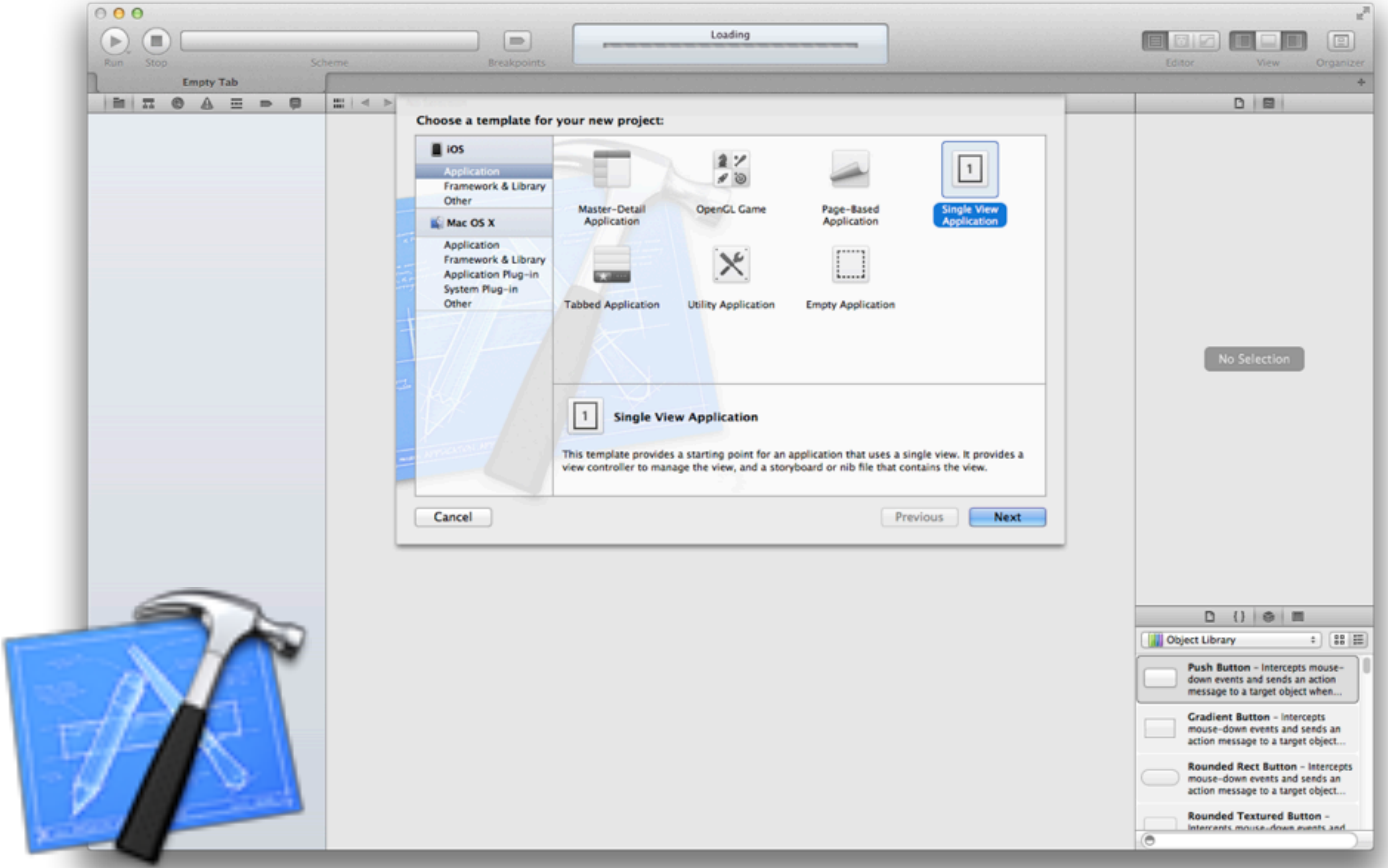
HelloWorld
~/Implementations/MobiCrant

MobiCraNT News
~/Implementations/MobiCrant

Last opened Yesterday 21:06

Xcode is the “Eclipse” for iOS development.

Getting Started with iOS Development



Choose a template for your new project:

Single View Application

This template provides a starting point for an application that uses a single view. It provides a view controller to manage the view, and a storyboard or nib file that contains the view.

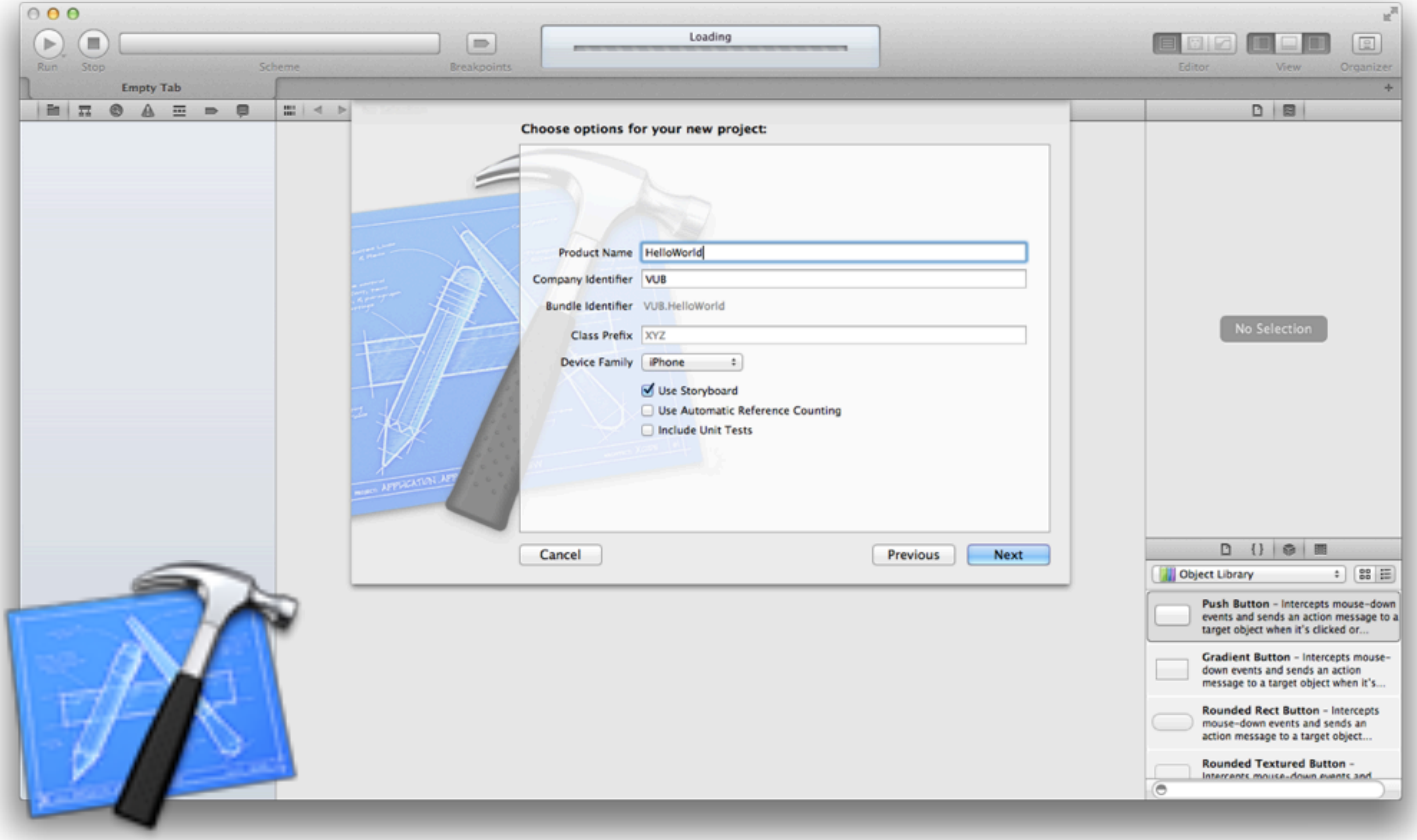
Push Button - Intercepts mouse-down events and sends an action message to a target object when...

Gradient Button - Intercepts mouse-down events and sends an action message to a target object...

Rounded Rect Button - Intercepts mouse-down events and sends an action message to a target object...

Rounded Textured Button - Intercepts mouse-down events and

Getting Started with iOS Development



Choose options for your new project:

Product Name: HelloWorld

Company Identifier: VUB

Bundle Identifier: VUB.HelloWorld

Class Prefix: XYZ

Device Family: iPhone

☑ Use Storyboard

☐ Use Automatic Reference Counting

☐ Include Unit Tests

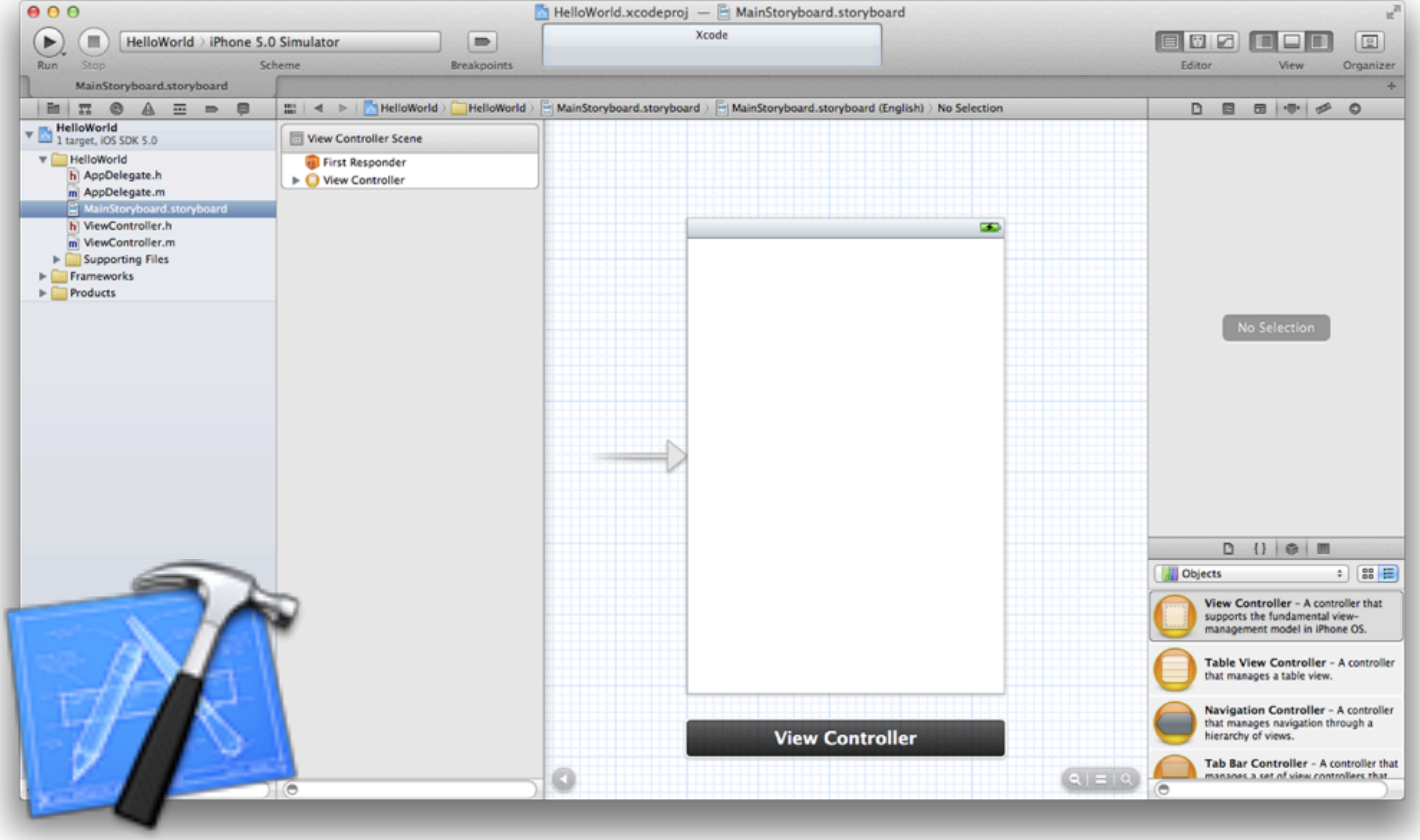
Push Button – Intercepts mouse-down events and sends an action message to a target object when it's clicked or...

Gradient Button – Intercepts mouse-down events and sends an action message to a target object when it's...

Rounded Rect Button – Intercepts mouse-down events and sends an action message to a target object...

Rounded Textured Button – Intercepts mouse-down events and

Getting Started with iOS Development



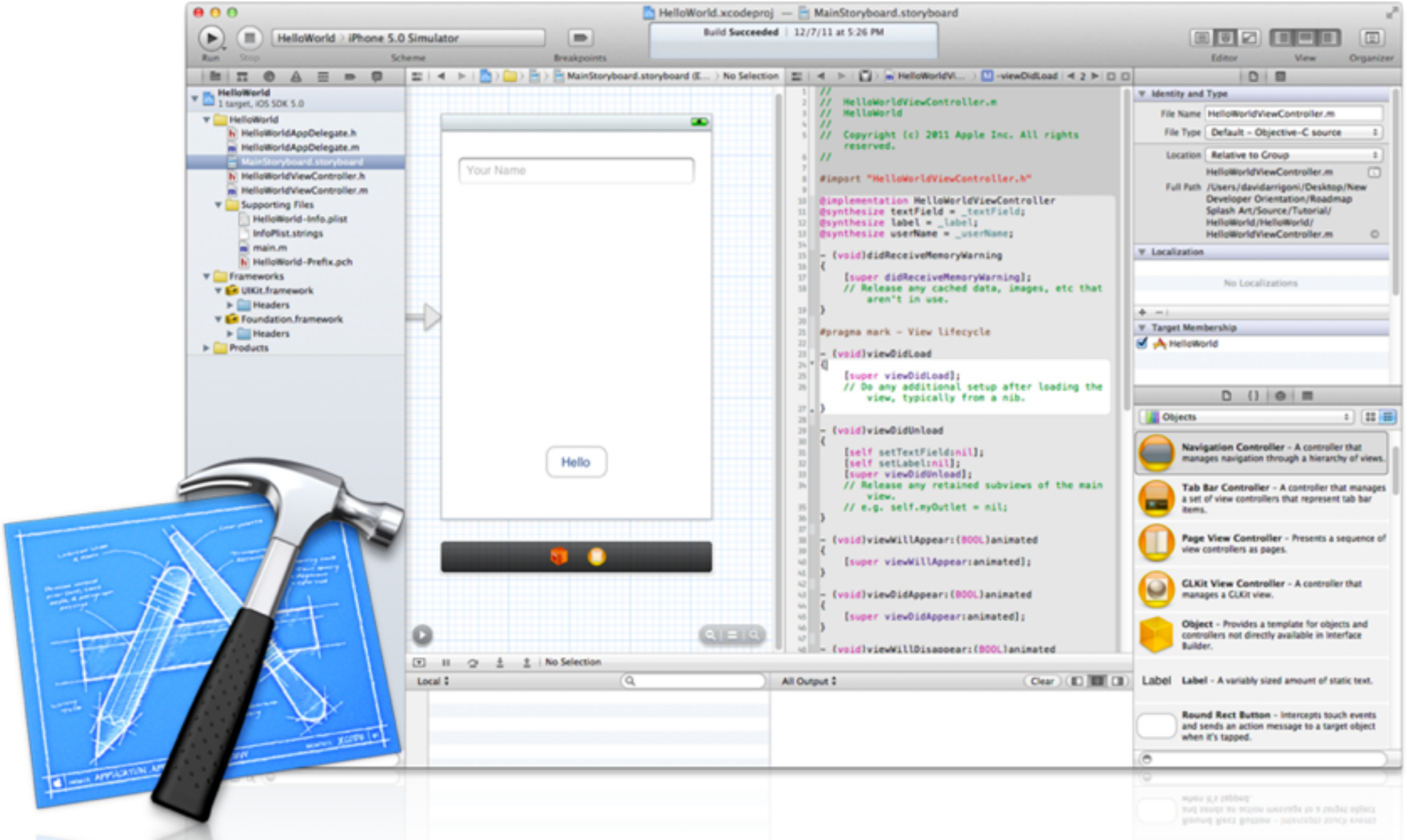
View Controller - A controller that supports the fundamental view-management model in iPhone OS.

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

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

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

Getting Started with iOS Development



HelloWorld.xcodeproj — MainStoryboard.storyboard

Build Succeeded | 12/7/11 at 5:26 PM

```
//
//  HelloWorldViewController.m
//  HelloWorld
//
//  Copyright (c) 2011 Apple Inc. All rights
    reserved.
//

#import "HelloWorldViewController.h"

@implementation HelloWorldViewController
@synthesize textField = _textField;
@synthesize label = _label;
@synthesize userName = _userName;

- (void)didReceiveMemoryWarning
{
    [super didReceiveMemoryWarning];
    // Release any cached data, images, etc that
        aren't in use.
}

#pragma mark - View lifecycle

- (void)viewDidLoad
{
    [super viewDidLoad];
    // Do any additional setup after loading the
        view, typically from a nib.
}

- (void)viewDidUnload
{
    [self setTextField:nil];
    [self setLabel:nil];
    [super viewDidUnload];
    // Release any retained subviews of the main
        view.
    // e.g. self.myOutlet = nil;
}

- (void)viewWillAppear:(BOOL)animated
{
    [super viewWillAppear:animated];
}

- (void)viewDidAppear:(BOOL)animated
{
    [super viewDidAppear:animated];
}

- (void)viewWillDisappear:(BOOL)animated
```

Identity and Type

File Name: HelloWorldViewController.m

File Type: Default - Objective-C source

Location: Relative to Group

HelloWorldViewController.m

Full Path: /Users/davidarrigoni/Desktop/New Developer Orientation/Roadmap Splash Art/Source/Tutorial/HelloWorld/HelloWorld/HelloWorldViewController.m

Localization

No Localizations

Target Membership

☑ HelloWorld

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

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

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

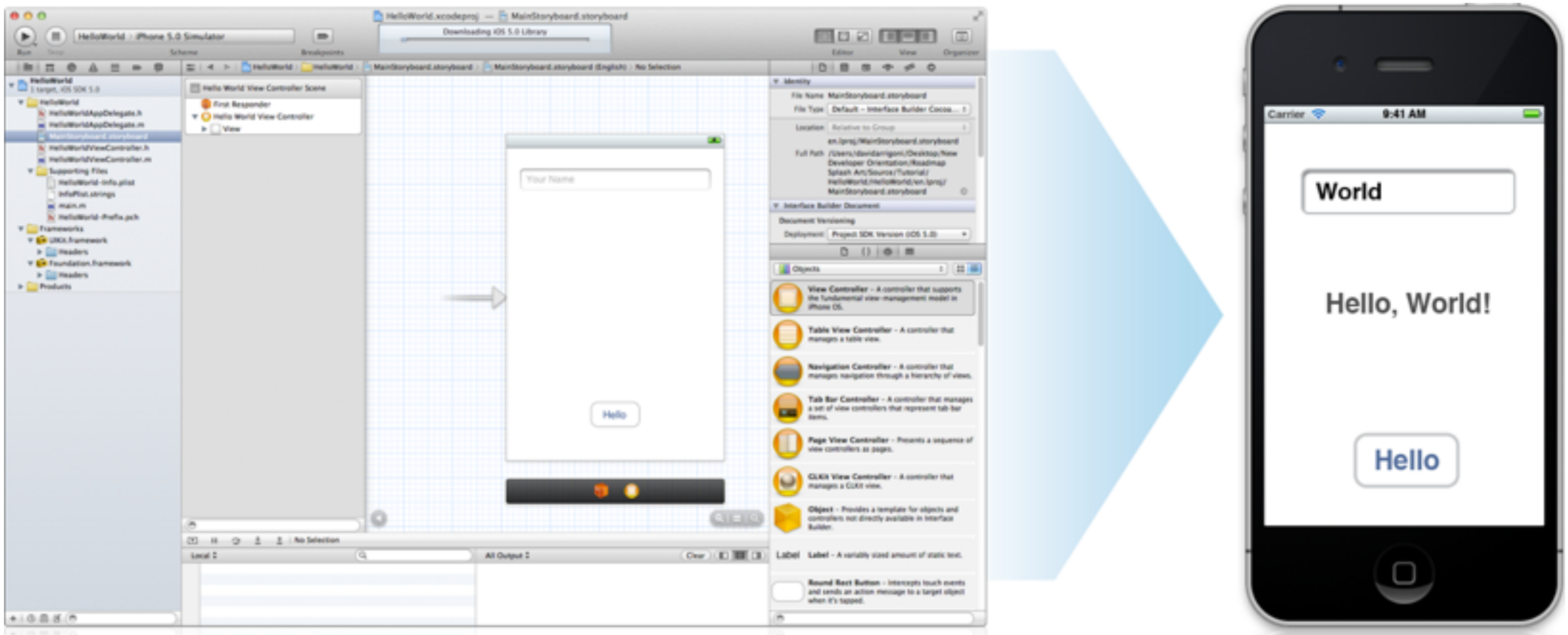
GLKit View Controller - A controller that manages a GLKit view.

Object - Provides a template for objects and controllers not directly available in Interface Builder.

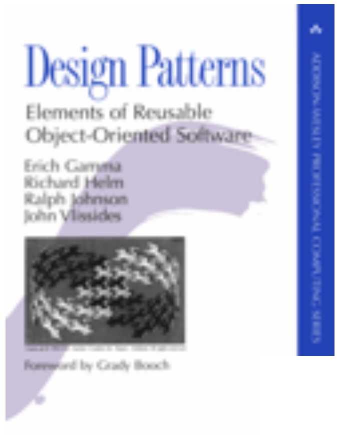
Label - A variably sized amount of static text.

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

Getting Started with iOS Development



iOS Design Patterns and Techniques



Design Patterns

Elements of Reusable Object-Oriented Software

Erich Gamma
Richard Helm
Ralph Johnson
John Vlissides

Foreword by Grady Booch

A design pattern is a template for a design that solves a general, recurring problem.

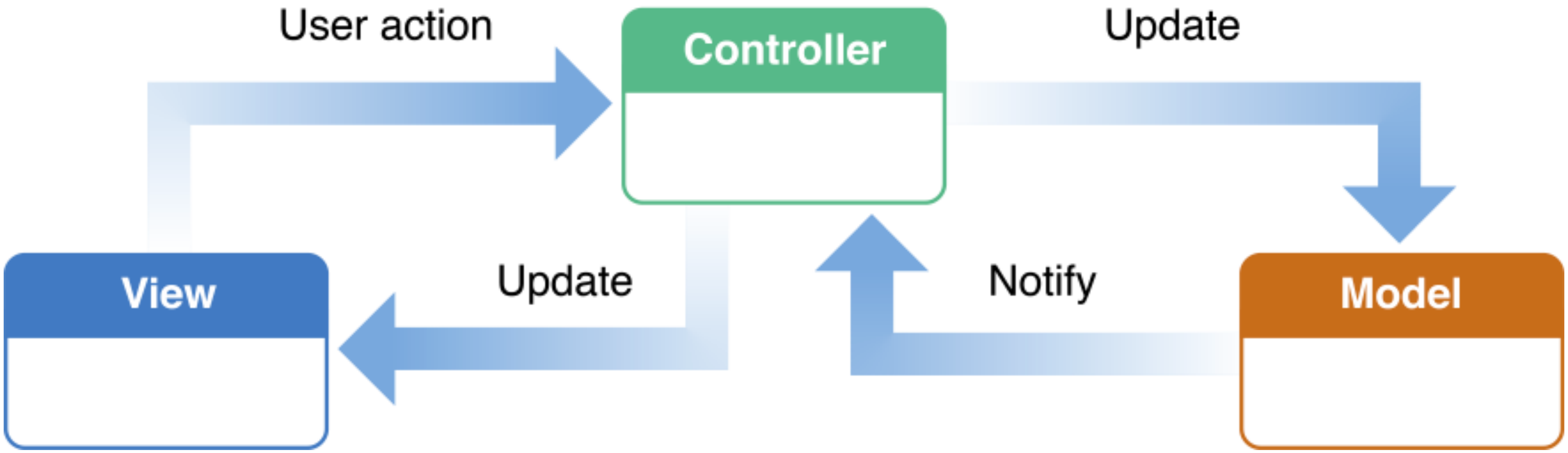
Common iOS design patterns:

Model-View-Controller (MVC), Delegation, and Target-action

iOS Design Patterns and Techniques

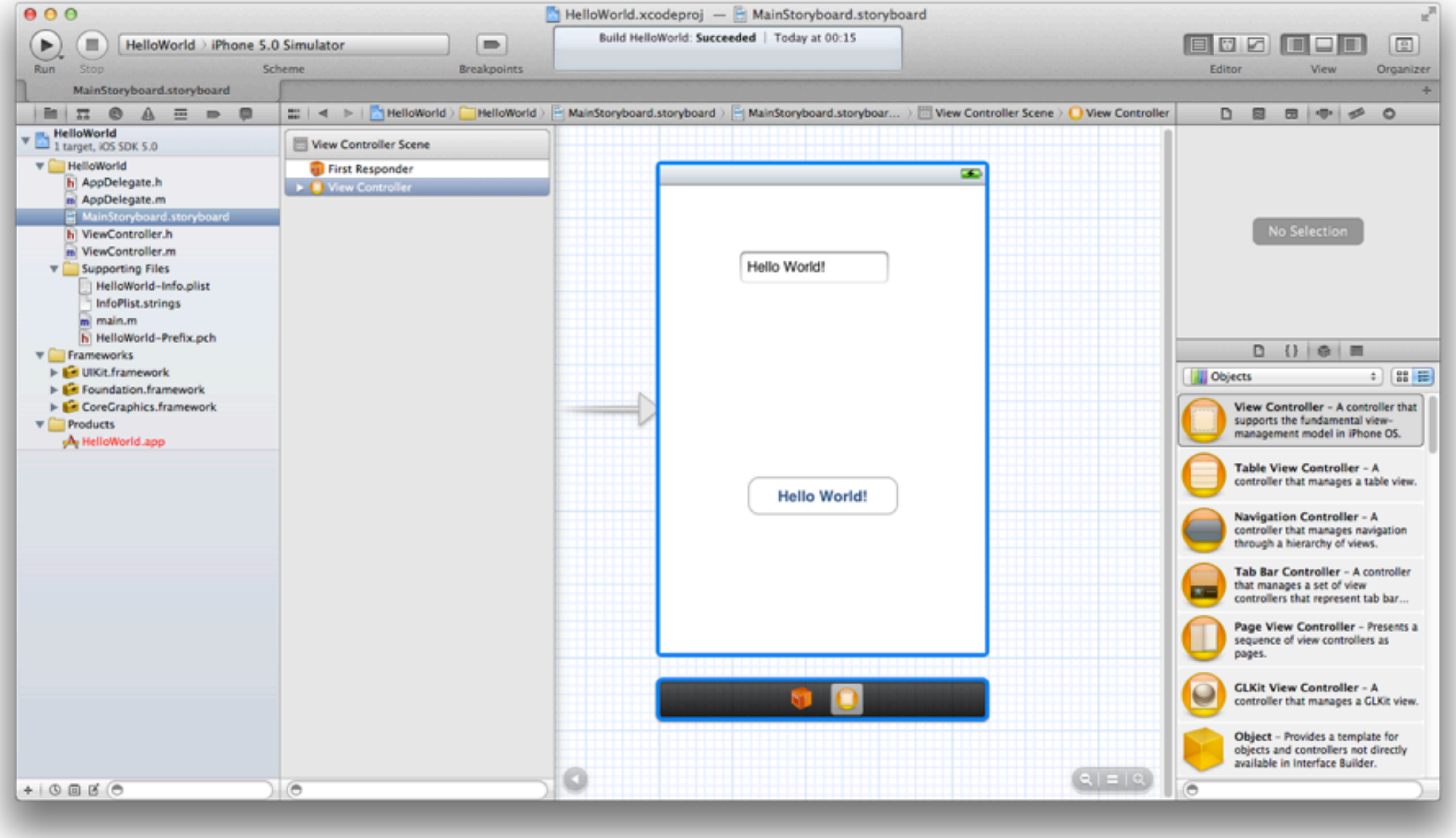
MVC pattern

The overall structure of iOS apps is based on the MVC pattern.



iOS Design Patterns and Techniques

Creating the V in MVC



Build HelloWorld: **Succeeded** | Today at 00:15

View Controller – A controller that supports the fundamental view-management model in iPhone OS.

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

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

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

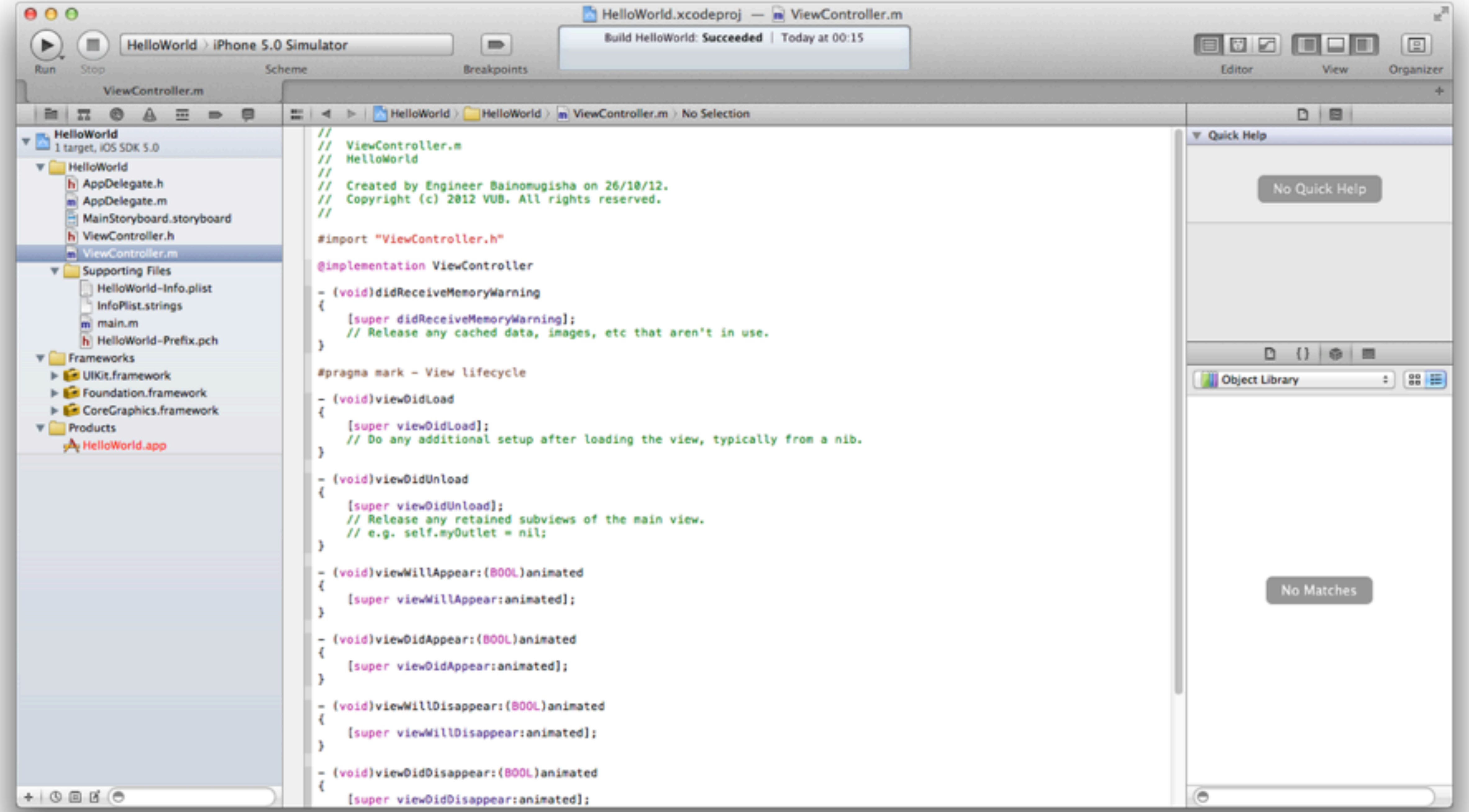
Page View Controller – Presents a sequence of view controllers as pages.

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

Object – Provides a template for objects and controllers not directly available in Interface Builder.

iOS Design Patterns and Techniques

Creating the C & M in MVC

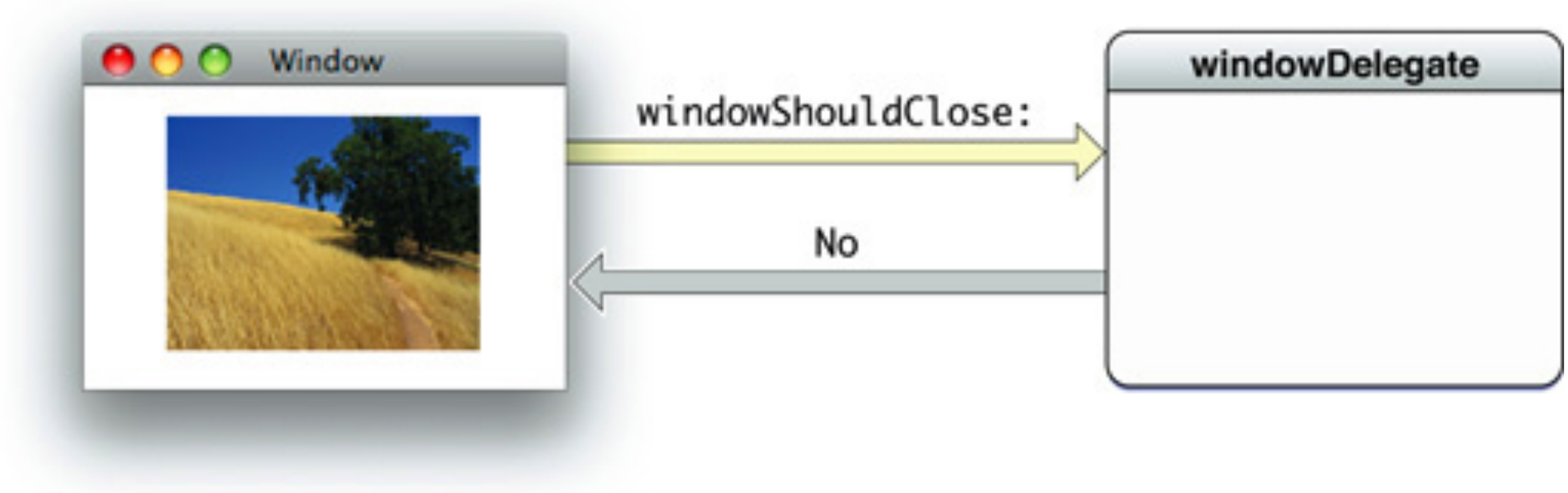


```
//
//  ViewController.m
//  HelloWorld
//
//  Created by Engineer Bainomugisha on 26/10/12.
//  Copyright (c) 2012 VUB. All rights reserved.
//

#import "ViewController.h"

@implementation ViewController

- (void)didReceiveMemoryWarning
{
    [super didReceiveMemoryWarning];
    // Release any cached data, images, etc that aren't in use.
}

#pragma mark - View lifecycle

- (void)viewDidLoad
{
    [super viewDidLoad];
    // Do any additional setup after loading the view, typically from a nib.
}

- (void)viewDidUnload
{
    [super viewDidUnload];
    // Release any retained subviews of the main view.
    // e.g. self.myOutlet = nil;
}

- (void)viewWillAppear:(BOOL)animated
{
    [super viewWillAppear:animated];
}

- (void)viewDidAppear:(BOOL)animated
{
    [super viewDidAppear:animated];
}

- (void)viewWillDisappear:(BOOL)animated
{
    [super viewWillDisappear:animated];
}

- (void)viewDidDisappear:(BOOL)animated
{
    [super viewDidDisappear:animated];
```

iOS Design Patterns and Techniques

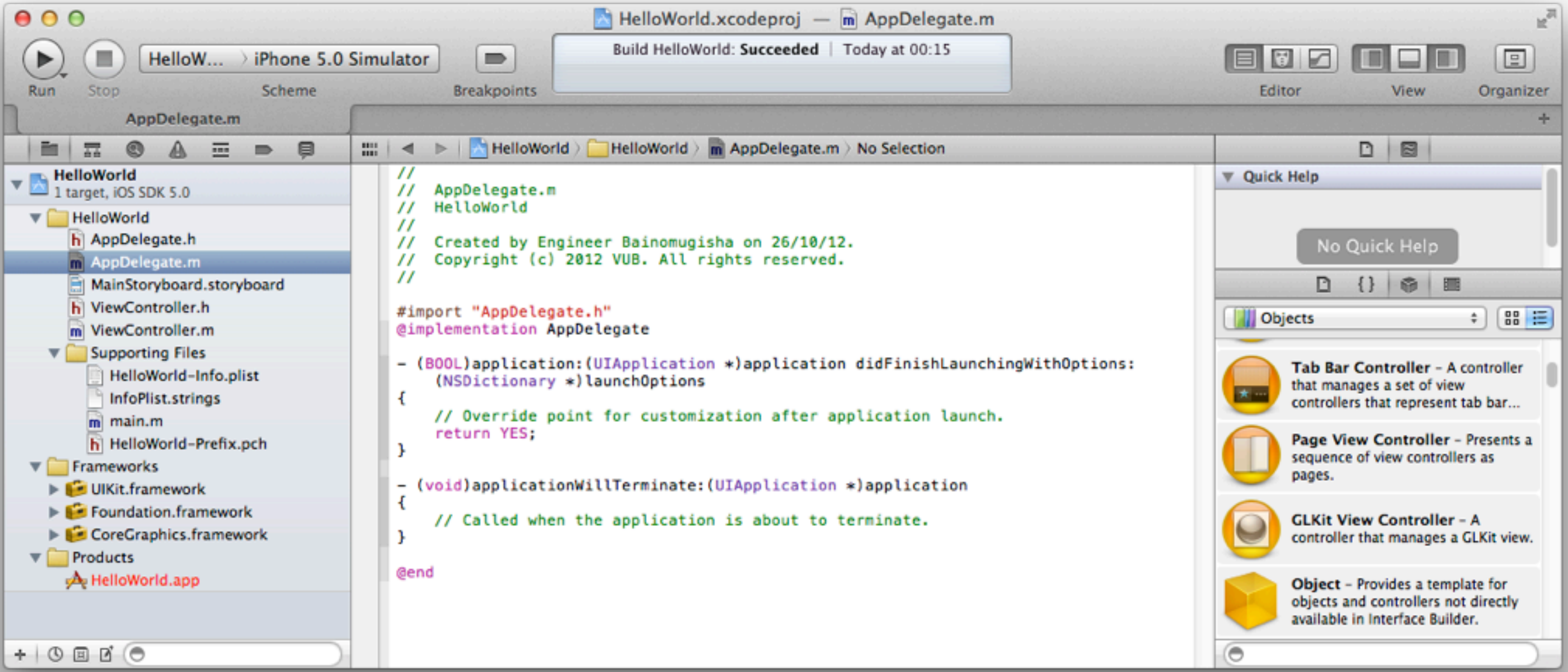
Delegation pattern

Used for interaction with iOS frameworks (instead of subclassing).



iOS Design Patterns and Techniques

Delegation pattern



Build HelloWorld: **Succeeded** | Today at 00:15

```
//
//  AppDelegate.m
//  HelloWorld
//
//  Created by Engineer Bainomugisha on 26/10/12.
//  Copyright (c) 2012 VUB. All rights reserved.
//

#import "AppDelegate.h"
@implementation AppDelegate

- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:
    (NSDictionary *)launchOptions
{
    // Override point for customization after application launch.
    return YES;
}

- (void)applicationWillTerminate:(UIApplication *)application
{
    // Called when the application is about to terminate.
}

@end
```

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

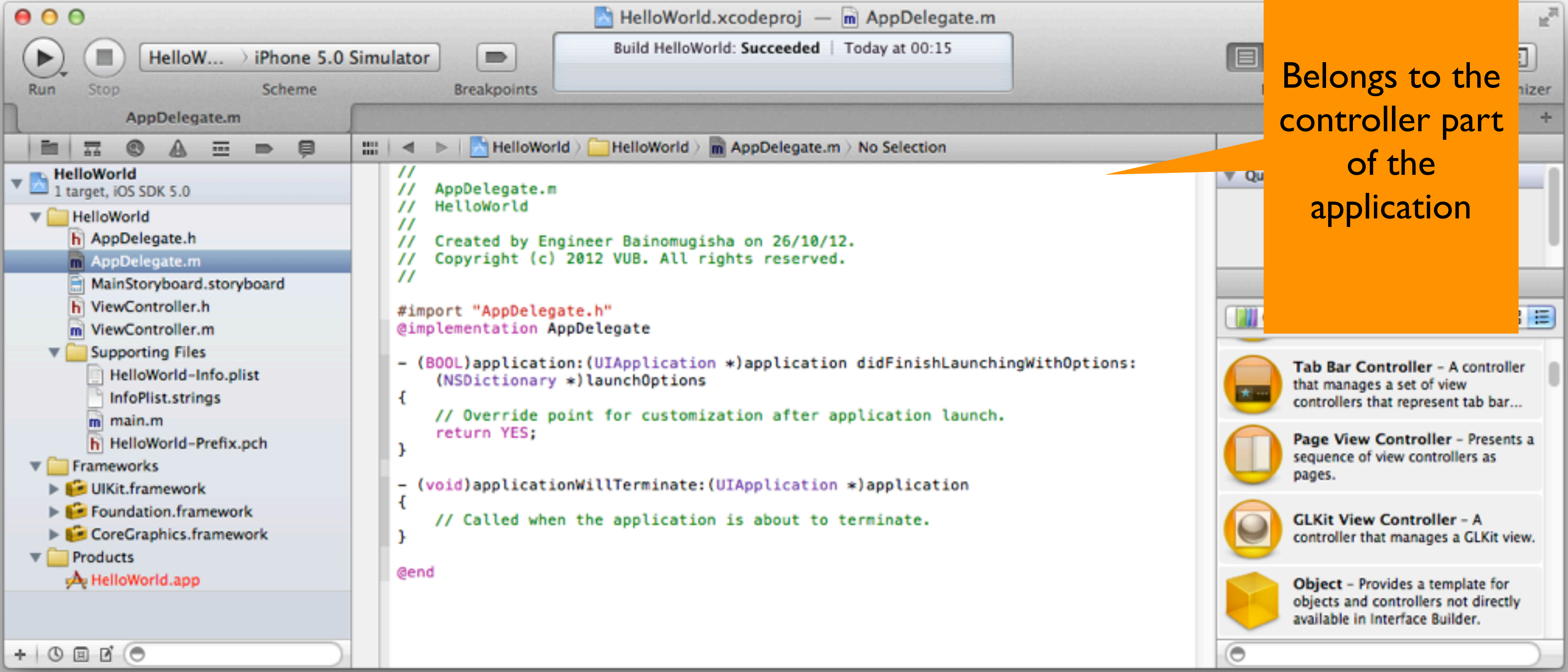
Page View Controller – Presents a sequence of view controllers as pages.

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

Object – Provides a template for objects and controllers not directly available in Interface Builder.

iOS Design Patterns and Techniques

Delegation pattern



```
//
//  AppDelegate.m
//  HelloWorld
//
//  Created by Engineer Bainomugisha on 26/10/12.
//  Copyright (c) 2012 VUB. All rights reserved.
//

#import "AppDelegate.h"
@implementation AppDelegate

- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:
    (NSDictionary *)launchOptions
{
    // Override point for customization after application launch.
    return YES;
}

- (void)applicationWillTerminate:(UIApplication *)application
{
    // Called when the application is about to terminate.
}

@end
```

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

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

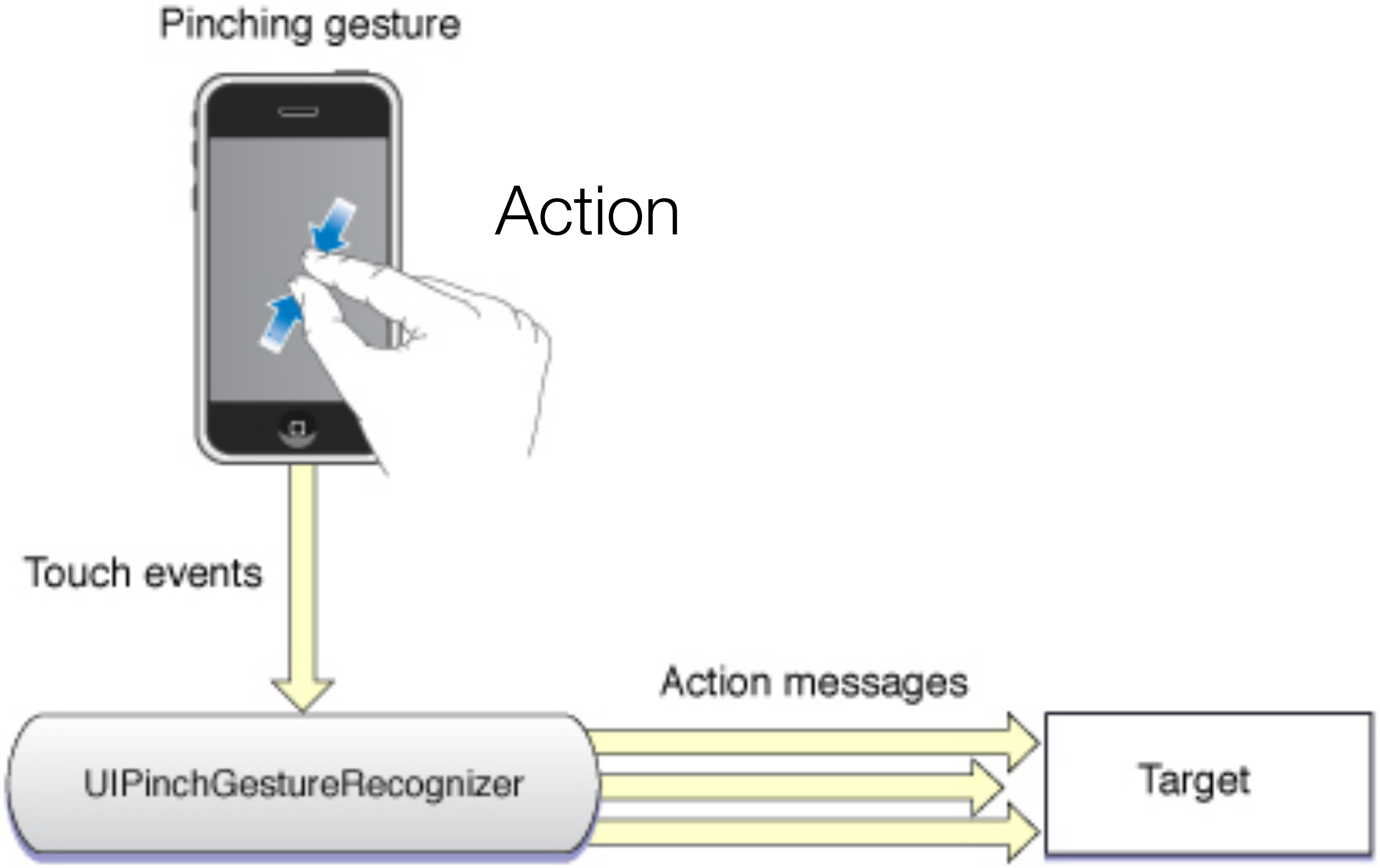
GLKit View Controller – A controller that manages a GLKit view.

Object – Provides a template for objects and controllers not directly available in Interface Builder.

Belongs to the controller part of the application

iOS Design Patterns and Techniques

Target-Action pattern



- Used for handling user interactions with the UI.
- The view sends an action when things happen in the UI.

iOS Design Patterns and Techniques

Target-Action pattern (1 of 2)

The kind of gesture we need to handle

```
UIPinchGestureRecognizer *pinchGesture = [[UIPinchGestureRecognizer alloc] init];

[pinchGesture addTarget: self action:@selector(handleMyPinchGesture:)];

[self.view addGestureRecognizer:pinchGesture];
```

Our gesture handler

Adding a gesture recogniser to a view

“self” is the target

iOS Design Patterns and Techniques

Target-Action pattern (2 of 2)

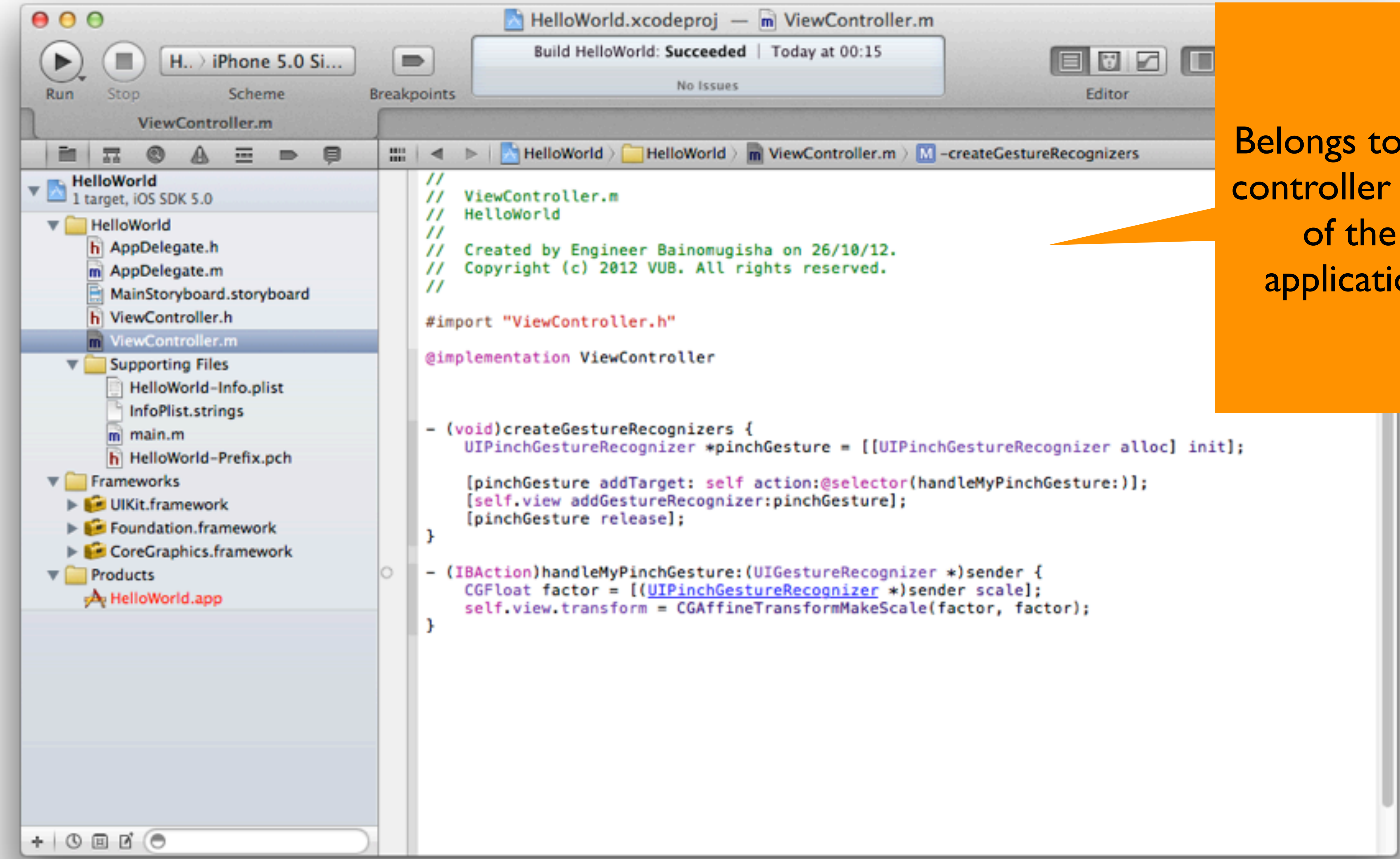
Implementing our gesture handler

```
- (IBAction)handleMyPinchGesture:(UIGestureRecognizer *)sender {

    CGFloat factor = [(UIPinchGestureRecognizer *)sender scale];
    self.view.transform = CGAffineTransformMakeScale(factor, factor);

}
```

We transform the view according to the pinching scale

iOS Design Patterns and Techniques

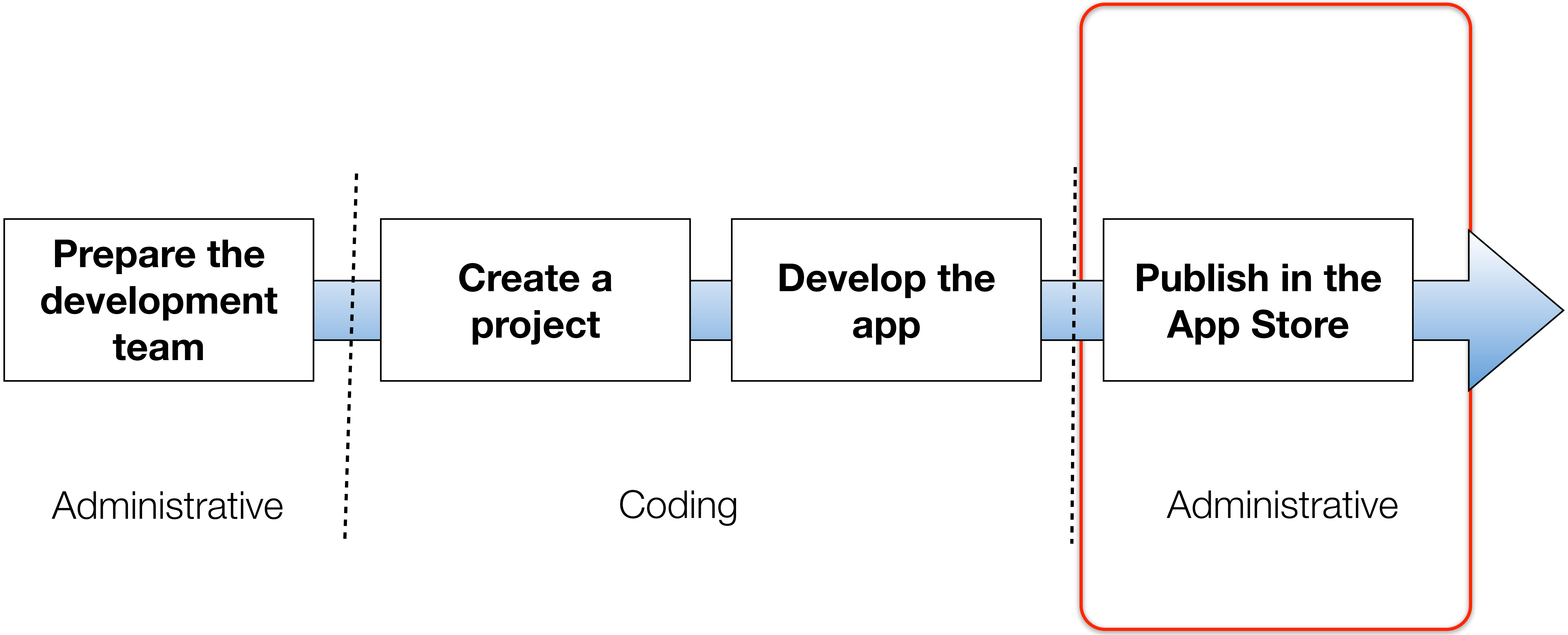
Target-Action pattern



```
//
//  ViewController.m
//  HelloWorld
//
//  Created by Engineer Bainomugisha on 26/10/12.
//  Copyright (c) 2012 VUB. All rights reserved.
//

#import "ViewController.h"

@implementation ViewController


- (void)createGestureRecognizers {
    UIPinchGestureRecognizer *pinchGesture = [[UIPinchGestureRecognizer alloc] init];

    [pinchGesture addTarget: self action:@selector(handleMyPinchGesture:)];
    [self.view addGestureRecognizer:pinchGesture];
    [pinchGesture release];
}

- (IBAction)handleMyPinchGesture:(UIGestureRecognizer *)sender {
    CGFloat factor = [(UIPinchGestureRecognizer *)sender scale];
    self.view.transform = CGAffineTransformMakeScale(factor, factor);
}
```

Belongs to the controller part of the application

Development Process Overview



Distributing Apps on the App Store

Develop & Test

Xcode comes with an iOS simulator for testing applications.

Distribute

Testing on real devices and distribution on the App store requires subscription to iOS Developer Program.

iOS Developer Program

iOS Developer Program ($99/year)
App store distribution (an individual, or company).

iOS Developer Enterprise Program ($299/year)
Internal distribution within your company.

iOS Developer University Program (Free)
Higher education institutions that need to introduce iOS development into their curriculum.

Signing Your App

- Each app must be code-signed to run on a device.
- Requires a **developer certificate** and a **provisioning profile**

    1. Create a certificate request.

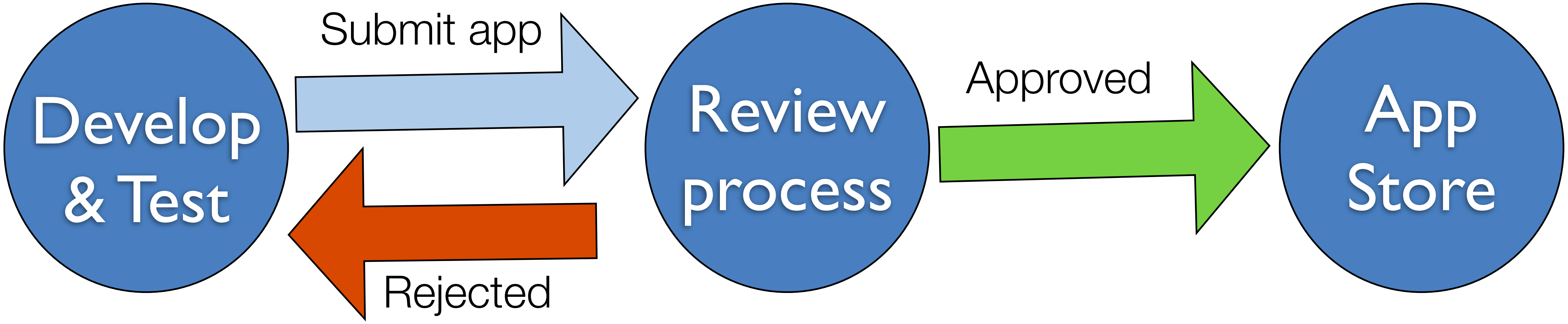
    2. Submit it to Apple via *developer.apple.com*

    3. Download/install the certificate.

    4. Obtain/install a provisioning profile.

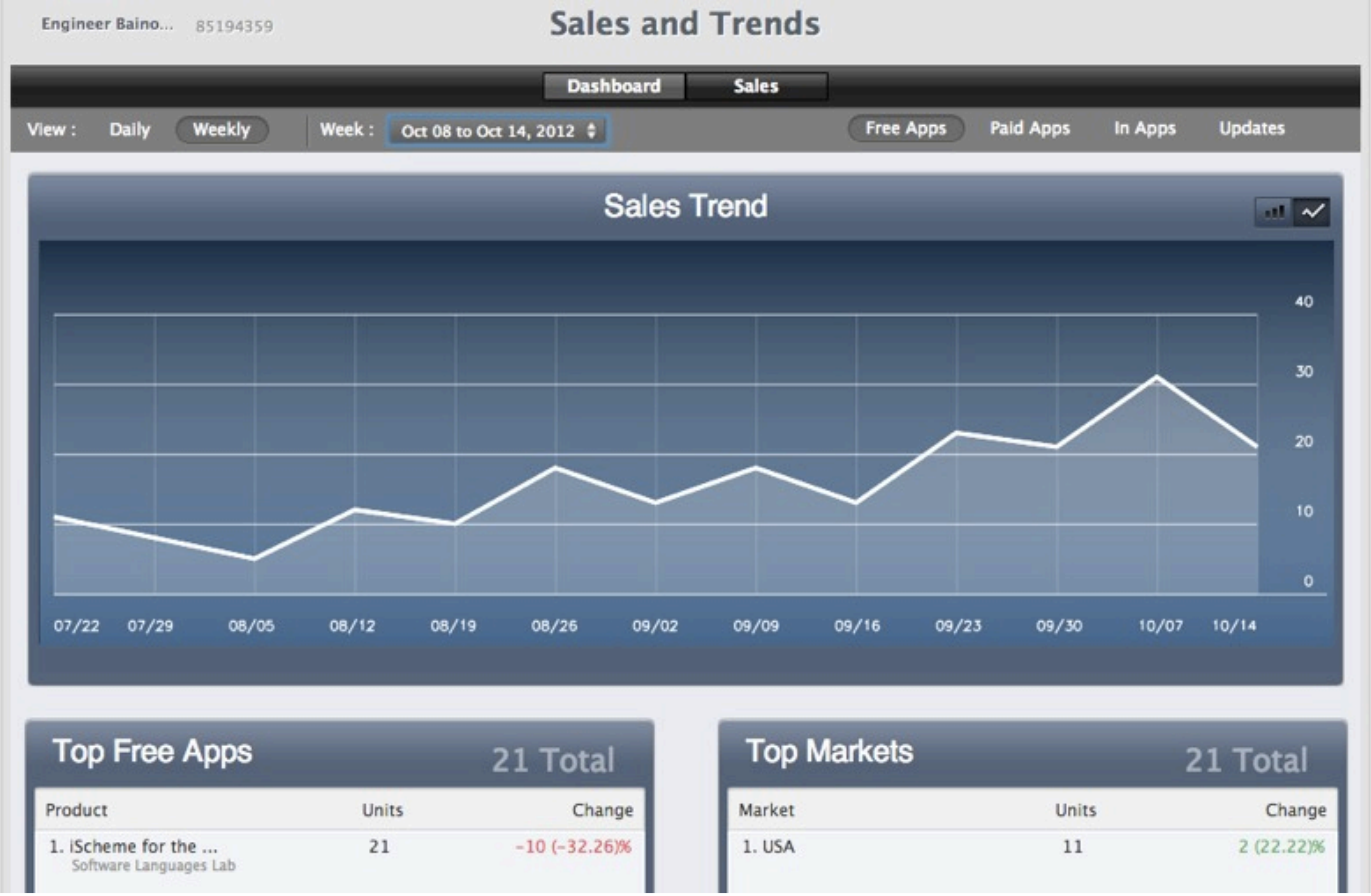
    5. Generate a binary for distribution.

Distributing Apps on the App Store



- App review usually takes at least 3 weeks.
- You set the price.
- In-app purchases also possible.
- You get 70% of sales revenue.

Sales Reports (Daily, Weekly, Monthly)



Sales and Trends

Week : Oct 08 to Oct 14, 2012

Sales Trend

Top Free Apps

21 Total

Product	Units	Change
1. iScheme for the ... Software Languages Lab	21	-10 (-32.26)%

Top Markets

21 Total

Market	Units	Change
1. USA	11	2 (22.22)%

Developing for the iOS Platform

Development Tools

Xcode

Interface Builder

Programming Language

Objective-C

Design Patterns

MVC, Delegate, Target-Action

Frameworks

Passbook

Maps

Facebook

Games

App Distribution Model

Available on the App Store

Where to Start

Learn Objective-C

http://developer.apple.com/library/mac/navigation/

iOS Developer Library

+

Sample Code

http://developer.apple.com/library/ios/

Note: Xcode requires a Mac.