



# Tutorial

Christopher M. Judd  
**Judd Solutions**

# Christopher M. Judd

President/Consultant of **Judd Solutions**

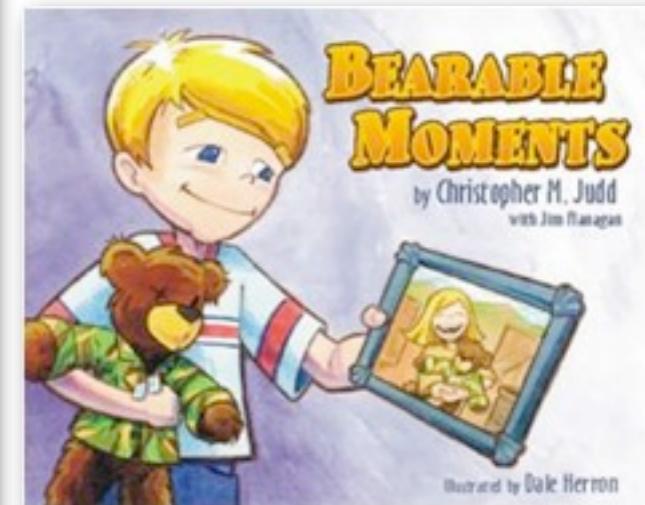
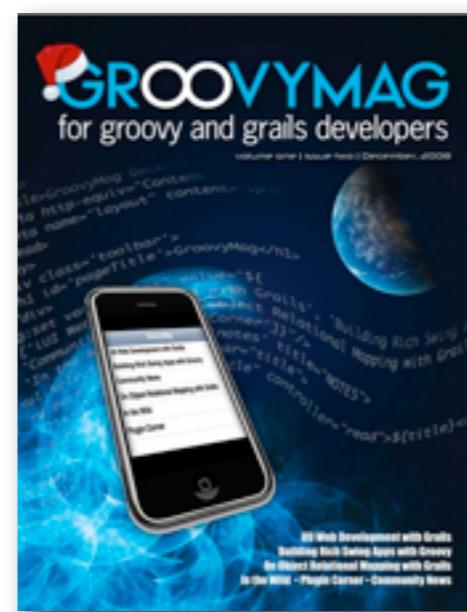
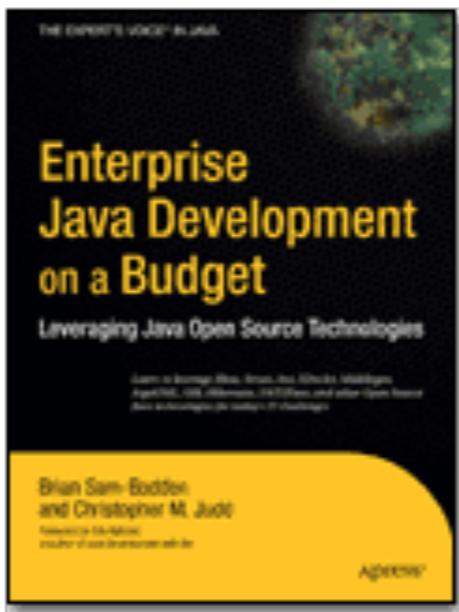
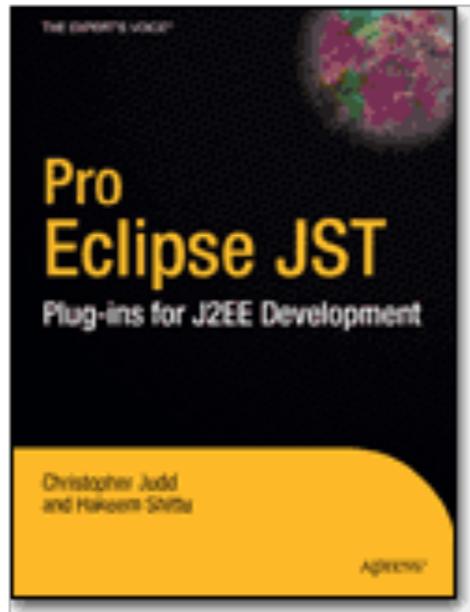
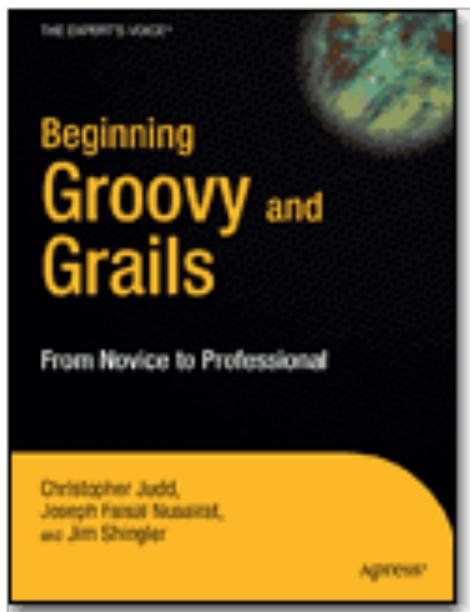


**Central Ohio Java Users Group** leader

Columbus



Developer User Group (CIDUG)





Your mobile phone has more computing power than all of NASA in 1969. NASA launched a man to the moon.





We launch a bird into pigs.

twitter

# Current tutorial stuff

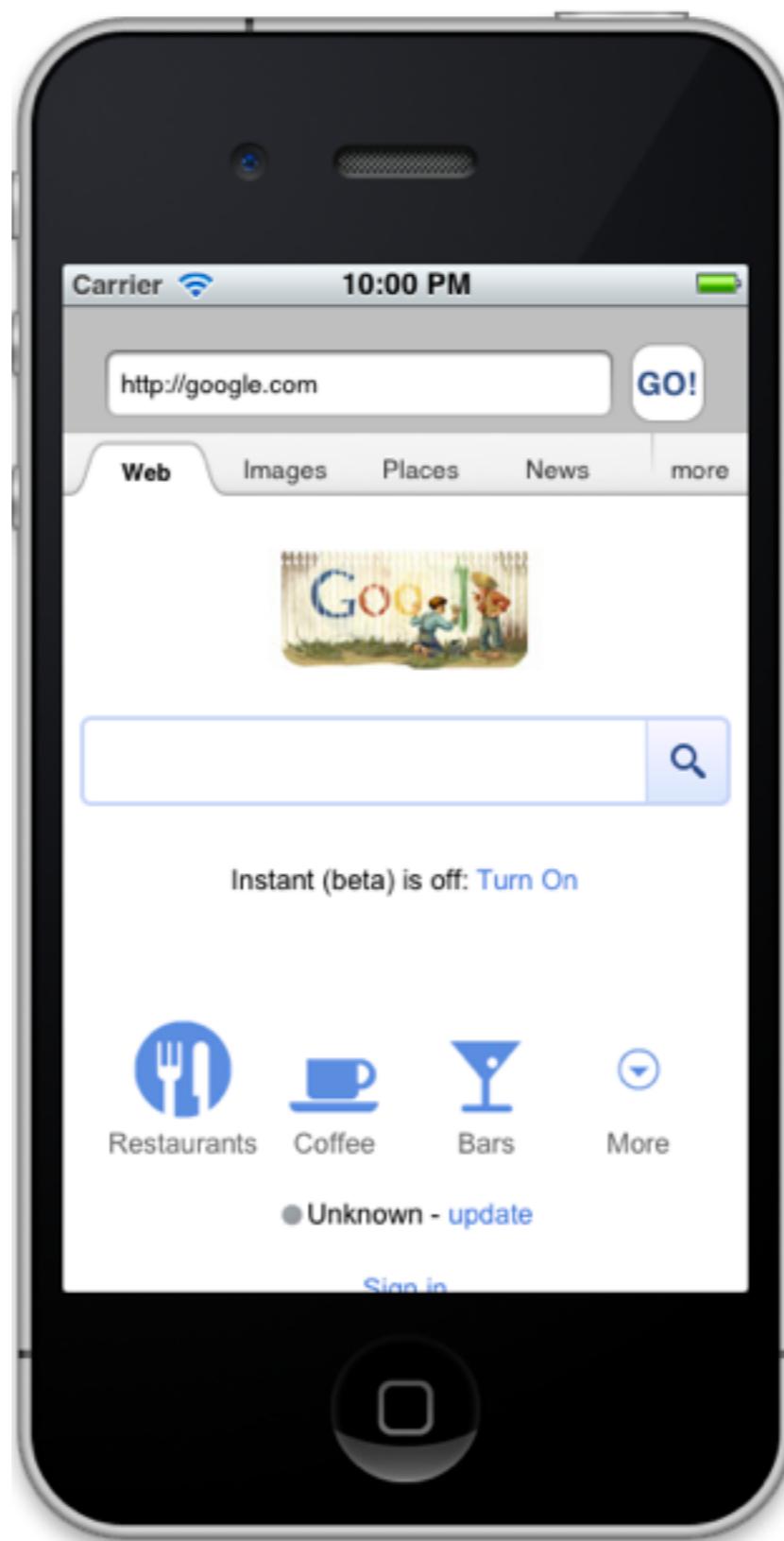


[http://juddsolutions.com/ios\\_tutorial/](http://juddsolutions.com/ios_tutorial/)



# **INTRODUCTION**

# MY BROWSER





# EVERNOTE

A screenshot of the Evernote mobile application showing a note detail view. The title is 'Basil Vinaigrette Recipe | Simply Recipes'. It was posted by Elise on Aug 28, 2010. The note includes a thumbnail image of a green liquid in a glass pitcher with basil leaves, the recipe title, a short description, and a list of ingredients. The ingredients list includes:

- 1 teaspoon Dijon mustard
- 1 shallot, chopped
- 1/2 teaspoon salt
- 1 teaspoon sugar
- 1/2 cup roughly chopped basil leaves
- 1/4 cup white wine vinegar
- 1/4 cup olive oil

At the bottom are navigation icons for 'New note', 'Search', 'Email', and 'Share'.

A screenshot of the Evernote mobile application showing a list of notes. The top bar shows 'AT&amp;T' and the time '4:05 PM'. The notes are listed in sections: 'San Francisco, California' (brazilian place, Aug 12, 2008 8:34 PM, Not Tagged), 'Santa Clara, California' (Untitled Note, Oct 16, 2008 4:01 PM, Not Tagged), 'Santa Cruz, California' (mmmw concert, Nov 19, 2008 10:55 PM, Not Tagged), and 'Stanford, California' (Untitled Note). At the bottom are navigation icons for 'New Note', 'Notes', 'Favorites', and 'Account'.

# EVERYNODE





**Design**

**Xcode 4**

**Objective-C**

**Deployment (device, ad hoc, app store)**

**Instruments**

**Images (app icon/launch image)**

**Core Data**

**Universal app dev**

**UITableViewController**

cjudd/everynote at master - GitHub, Inc. [US] https://github.com/cjudd/everynote

rampart jcr manifest Ivan Kuznetsov > Usin itteco/IttecoTracPlug Note This System Management Other Bookmarks

# github SOCIAL CODING

## cjudd / everynote

Source Commits Network Pull Requests (0) Fork Queue Issues (0) Wiki (0) Graphs Branch: master

Switch Branches (2) Switch Tags (3) Branch List

EVERNOTE clone used for iOS Hands-on Tutorial  
click here to add a homepage

SSH HTTP Git Read-Only git@github.com:cjudd/everynote.git Read+Write access Clone in Mac

Add save and fetch example.

Christopher M. Judd (author)  
15 minutes ago

commit 74c7f4c14413b9fd3  
tree c85e2fa450ea726c5d  
parent 03c6f0b8f4d2d117d

### everynote /

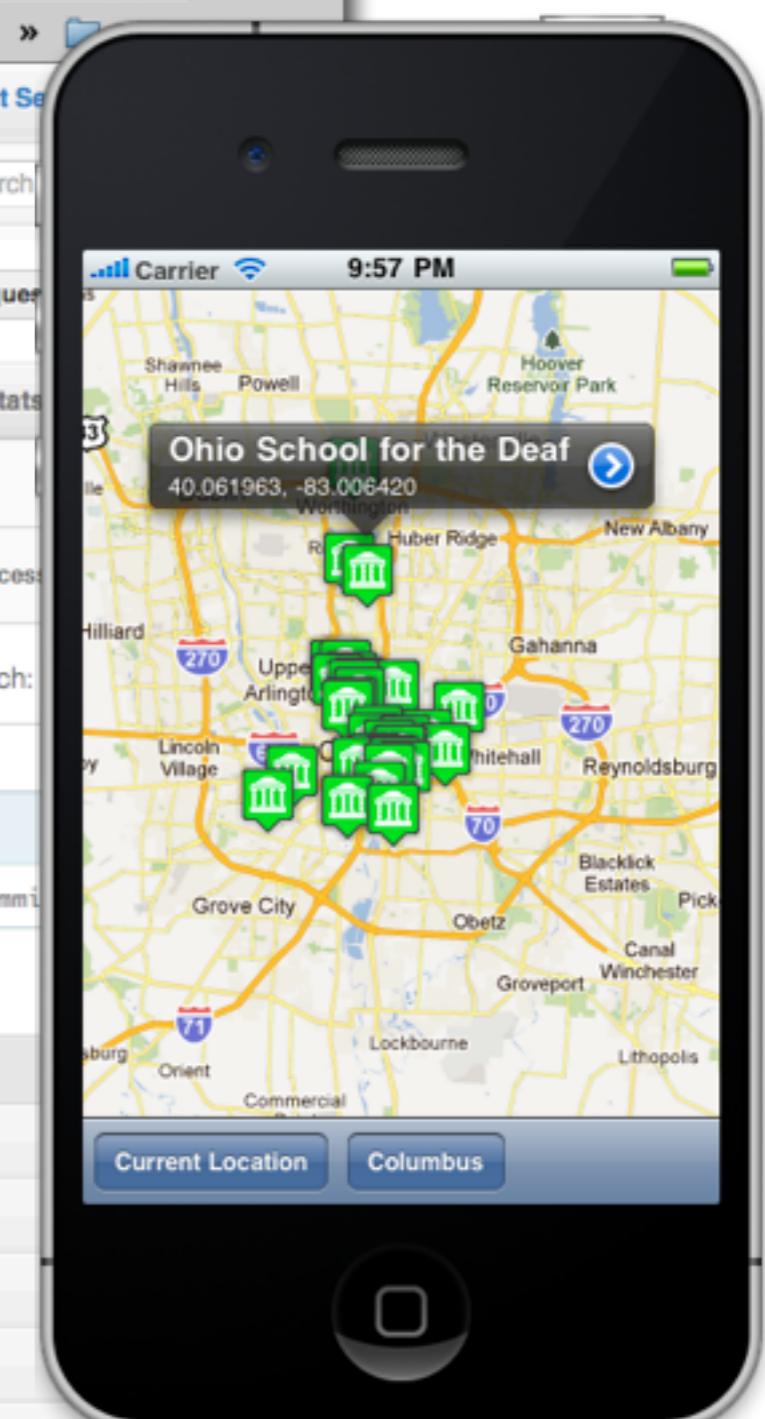
name	age	message
EveryNote.xcodeproj/	17 minutes ago	Generated Note class from CoreData schema. [Christopher M. Judd]
EveryNote/	16 minutes ago	Add save and fetch example. [Christopher M. Judd]



<https://github.com/cjudd/everynote>

Screenshot of a web browser showing the GitHub repository for 'ColumbusHistoryMap\_iOS' by cjudd. The repository page displays code, network, pull requests, issues, wiki, and stats sections. It includes links for cloning (Clone in Mac, ZIP, SSH, HTTP), a git URL, and access rights (Read+Write). The 'Files' tab is selected, showing a list of files: Classes/, ColumbusHistoryMap.xcodeproj/, ColumbusHistoryMap-Info.plist, ColumbusHistoryMapViewController.xib, ColumbusHistoryMap\_Prefix.pch, and MainWindow.xib. Each file entry includes its name, last modified date (April 11, 2011), and a message describing the commit. The commit for 'MainWindow.xib' also includes the author's GitHub link.

name	age	message
Classes/	April 11, 2011	Added orginal ColumbusHistoryMap code. [cjudd]
ColumbusHistoryMap.xcodeproj/	April 11, 2011	Removed specific user setting folder. [cjudd]
ColumbusHistoryMap-Info.plist	April 11, 2011	Added orginal ColumbusHistoryMap code. [cjudd]
ColumbusHistoryMapViewController.xib	April 11, 2011	Added orginal ColumbusHistoryMap code. [cjudd]
ColumbusHistoryMap_Prefix.pch	April 11, 2011	Added orginal ColumbusHistoryMap code. [cjudd]
MainWindow.xib	April 11, 2011	Added orginal ColumbusHistoryMap code. [cjudd]



[https://github.com/cjudd/ColumbusHistoryMap\\_iOS](https://github.com/cjudd/ColumbusHistoryMap_iOS)

# Git cheat sheet

## Clone repo

```
git clone git://github.com/cjudd/everynote.git
```

## Branch from tag

```
git checkout -b <tag name>
```

## Check status

```
git status
```

## Diff

```
git diff
```

## Adding (Staging)

```
git add .
```

```
git add <filename, folder name, or wildcard>
```

## Committing

```
git commit -m '<comment>'
```

## Switch Branch

```
git checkout <branch name>
```

## Create Branch

```
git branch <new branch name>
```

## Merging

```
git merge <branch name>
```

#94

 **DZone Refcardz**

Get More Refcardz! Visit [refcardz.com](http://refcardz.com)

[www.dzone.com](http://www.dzone.com)

**CONTENTS INCLUDE:**

- Distributed Version Control Concepts
- Cloning Existing Projects
- The Typical Local Workflow
- The Remote Workflow
- CVS & Subversion Interoperability
- and more...

**Getting Started with Git**  
By Matthew McCullough

From there you'll progress to the second epiphany of being able to share a repository and a changeset directly with a colleague without any complicated setup, without a checkin to a central server, direct network connectivity, or having to worry about firewalls getting in the way. Git has done technologically for version control what BitTorrent did for file sharing. It permanently replaced the spoke and hub structure with a peer-to-peer model, and there's no turning back. It supports transmitting binary sets of changes via USB stick, email, or in the traditional style, over a network, but amazingly, via HTTP, FTP, SCP, Samba, SSH, or WebDAV.

**WHY GET GIT?**

Git is a postmodern version control system that offers the familiar capabilities of CVS or Subversion, but doesn't stop at just matching existing tools. Git stretches the very notion of version control systems (VCS) by its ability to offer almost all of its features for use offline and without a central server. It is the brainchild of Linus Torvalds, with the first prototype written in a vitriolic two-week response to the "BitKeeper debacle" of 2005.

Today, developers everywhere are migrating in droves to this exciting platform. Users reference its blistering performance, usage flexibility, offline capabilities, and collaboration features as their motivation for switching. Let's get started with Git. You'll be using it like a master in no time at all.

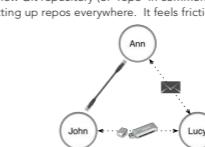
**More Than Just a Version Control System**

Though originally targeted at Linux kernel developers, Git has found a home in many other version-needing systems, such as document revision control. Git is just as capable of versioning a folder of documents or configuration files as it is of tracking decades of source code for the Perl project. This has led to Git being used by writers and network administrators — a case less likely to be encountered with the cumbersome CVS and Subversion servers and their ceremonious setup.

**DISTRIBUTED VERSION CONTROL**

If you are familiar with one or more traditional or centralized version control systems like Subversion, there will be several mental adjustments to make in your migration to Git. The first is that there is no central server. The second is that there is no central server. The full history of the repository lives on every user's machine that has cloned (checked out) a copy of the repository. This is the essence of a Distributed Version Control System (DVCS).

Once over those hurdles, it is quite liberating to be able to work entirely independently, versioning any new project that you start, even if in the incubation phase. The ease of setting up a new Git repository (or 'repo' in common parlance) leads to setting up repos everywhere. It feels frictionless.



**GETTING STARTED**

**Installing Git**

Git has a very light footprint for its installation. For most platforms, you can simple copy the binaries to a folder that is on the executable search \$PATH. Git is primarily written in C, which means there is a unique distribution for each supported platform.

**Hot Tip** The canonical reference for Git installers can be found on a subpage of the official Git site. <http://git-scm.com/download>

**Establishing User Credentials**

Once you have selected a suitable distribution of Git for your platform, you'll need to identify yourself with a username and email address to Git.

**Don't Miss An Issue!**

Get over 90 DZone Refcardz FREE from Refcardz.com!

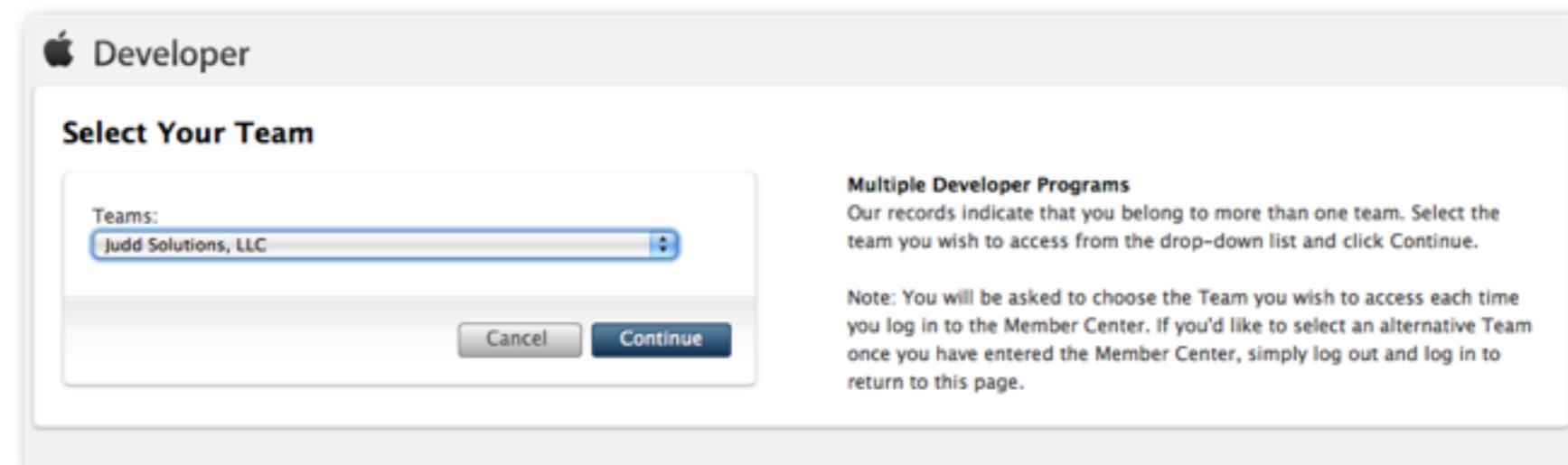


<http://help.github.com/git-cheat-sheets/>

# iOS Developer Programs

- **Developer**
- **Free**
- **Can not deploy to a device**
- **Standard Program**
- **\$99/year**
- **Deploy to up to 100 Ad Hoc devices**
- **Deploy to App Store**
- **Enterprise Program**
- **\$299/year**
- **For developing internal application and deploying to company phones**

## Register as an organization instead of individual



The screenshot shows the Apple Developer Member Center interface. The top navigation bar includes links for "People", "Programs & Add-ons", and "Your Account". The user is logged in as "Christopher Judd". The main content area displays the "All People" section, showing a table with columns for Name, Program, Status, and Actions. The table includes rows for "Christopher Judd" (Agent, Member) and another partially visible row.

Name	Program	Status	Action
Christopher Judd	iOS Developer Program	Agent	Details
[Redacted]	[Redacted]	Member	Details

# What you need

- Mac
- Mac OS X 10.6.x (Snow Leopard)
- Xcode 4 >
- iPhone, iPod Touch and/or iPad
- \$99/year
- Join Standard Program
  - <http://developer.apple.com/ios>
- To learn Objective-C



## iPhone Dev Center

Hi, Guest | [Register](#) | [Log In](#)**Log in to get the most out of the iPhone Dev Center.**[Log In](#)

The iPhone Dev Center provides access to technical resources and information to assist you in developing with the latest technologies in iPhone OS. Log in with your Registered iPhone Developer Apple ID and password, or sign-up as a free Registered iPhone Developer today.

## Developing for iPhone OS 2.2.1

Search iPhone Reference Library

**Technical Documentation**[Getting Started Documents](#)

Developers new to iPhone OS can read about the tools, frameworks, development best-practices, and design methods for creating innovative world-class iPhone applications.

[iPhone Reference Library](#)

Explore a collection of in-depth technical documentation, sample code, guides, and articles for iPhone development categorized by topic and frameworks.

**Featured Content**

- [iPhone Application Programming Guide](#)
- [iPhone Development Guide](#)
- [iPhone Human Interface Guidelines](#)
- [Your First iPhone Application](#)

To access iPhone SDK for iPhone OS 2.2.1 and additional technical resources and information, [log in](#) with your Registered iPhone developer Apple ID and password, or [sign up](#) as a free Registered iPhone Developer today.

[iPhone SDK for iPhone OS 2.2.1](#)

Registered iPhone Developers can download iPhone SDK for iPhone OS 2.2.1 which includes the Xcode IDE, iPhone simulator, and a suite of additional tools for developing applications for iPhone and iPod touch.

[Getting Started Videos](#)

Watch Apple experts discuss everything from getting started with iPhone SDK, to the tools and technologies used to create iPhone applications.

[Coding How-To's](#)

Play a movie. Play a sound. Detect motion. Learn how to incorporate features of iPhone in your application.

[Sample Code](#)

Use these samples to inspire development of your own great applications.

**iPhone Developer Program**

The iPhone Developer Program offers a complete process for developing and distributing iPhone or iPod touch applications.

When you join the iPhone Developer Program, you also receive access to iPhone SDK 3.0 and iPhone OS 3.0 beta.

[Learn More](#)

Standard Program \$99  
Enterprise Program \$299

Already applied to the iPhone Developer Program?

[Check your enrollment status now ▶](#)

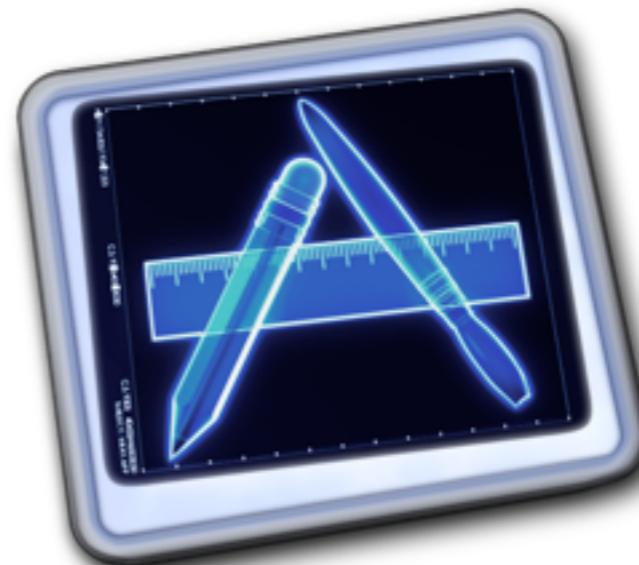
<http://developer.apple.com/ios/>



# Xcode IDE GUI Builder



# Simulator iPhone Simulator



# Instruments Profiler

**Downloads**

**Xcode 4.1 for Lion**  
  
 This is the complete Xcode developer toolset for Mac, iPhone, and iPad. It includes the Xcode IDE, iOS Simulator, and all required tools and frameworks for building Mac OS X and iOS apps.

**Xcode 4.0.2 for Snow Leopard**  
[Download >](#)  
  
[Looking for Xcode 3? >](#)

**iAd Producer 1.2**  
  
 iAd Producer, combined with the power of iOS 4 and its WebKit-based browser, makes it easy for you to create high-impact, motion-rich ads.

**Download Xcode 4**

Posted Date: Jul 20, 2011  
 Build: 4B110  
 Included iOS SDK: iOS 4.3  
 Included Mac SDK: Mac OS X 10.7

Posted Date: April 14, 2011  
 Build: 4A2002a  
 Included iOS SDK: iOS 4.3  
 Included Mac SDK: Mac OS X 10.6

Posted: June 23, 2011

**Downloads**  
[iAd Producer 1.2](#)

**Other Resources**

- [Introducing iAd Producer](#)
- [iAd Producer User Guide](#)
- [iAd Producer Tutorial](#)
- [iAd Design Guide](#)
- [iAd JS Developer Library](#)

[xcode](#)

**Xcode**

Xcode provides everything developers need to create great applications for Mac, iPhone, and iPad. Xcode 4 has been streamlined to help you write better apps. It has unified user interface design, coding, testing, and debugging all within a single window. The Xcode IDE analyzes the details of your project to identify mistakes in both syntax and logic, it...

[...More](#)

**What's New in Version 4.1.1**

This is a delta update to install Xcode.app that fixes the "Installation Alert" to quit iTunes. You do not need to re-run the installer if you already have Xcode 4.1 for Lion successfully...

[...More](#)



**Information**

Category: Developer Tools  
 Updated: Aug 04, 2011  
 Current Version: 4.1.1  
 Price: Free  
 Size: 2.96 GB  
 Language: English  
 Seller: Apple Inc.  
 © 2011 Apple Inc.

Rated 4+  
 Requirements: Mac OS X 10.7 or later

**More by Apple®**

 OS X Lion Productivity
 Final Cut Pro Video
 Pages Productivity
 Keynote Productivity

<http://developer.apple.com/ios/>

Anybody have an iOS device but does not have a developer account and doesn't want to play the \$99 today?

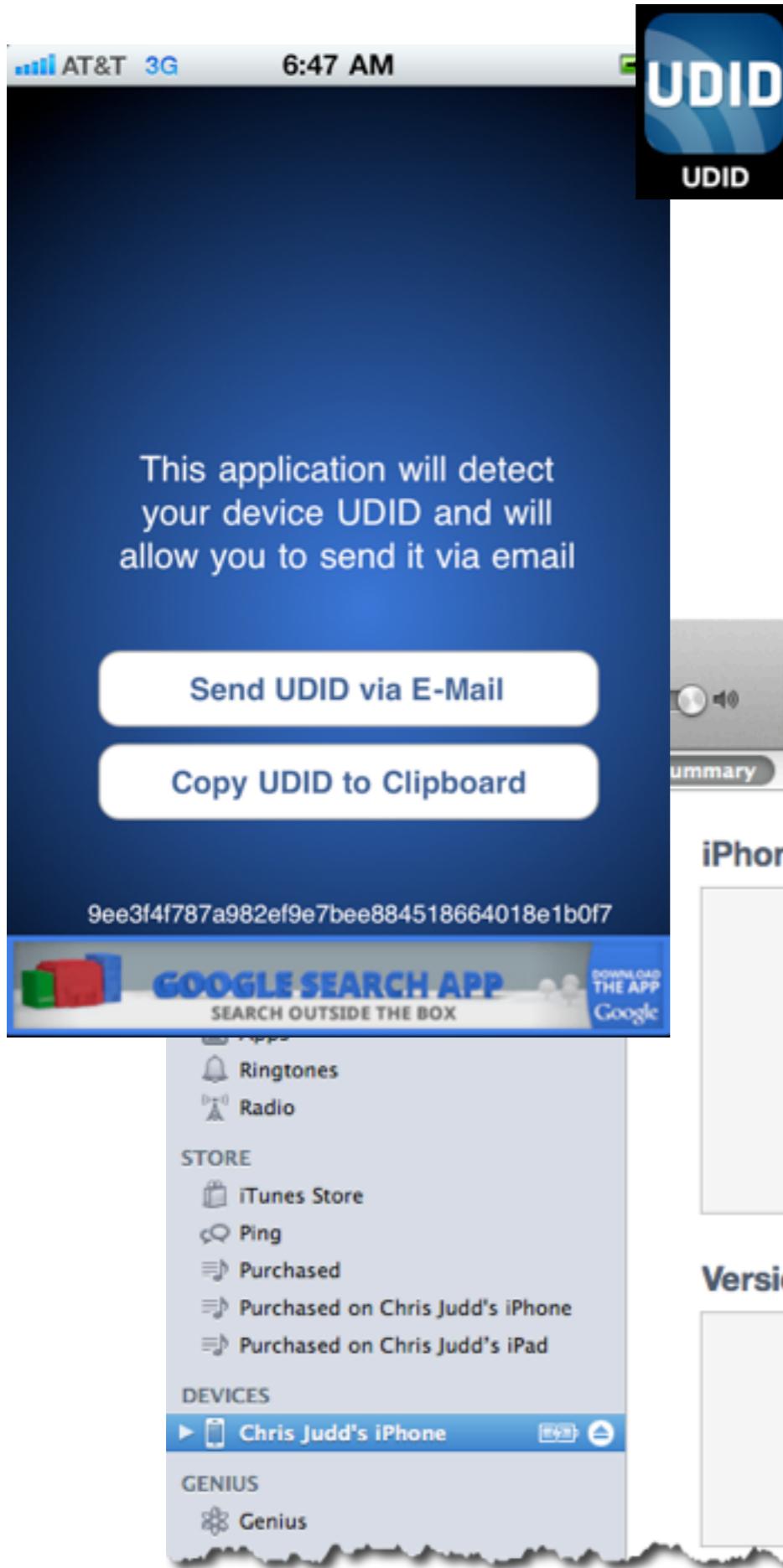


Email [javajudd@gmail.com](mailto:javajudd@gmail.com) with your



- First name
- Last name
- Email
- Device UDID

# UDID

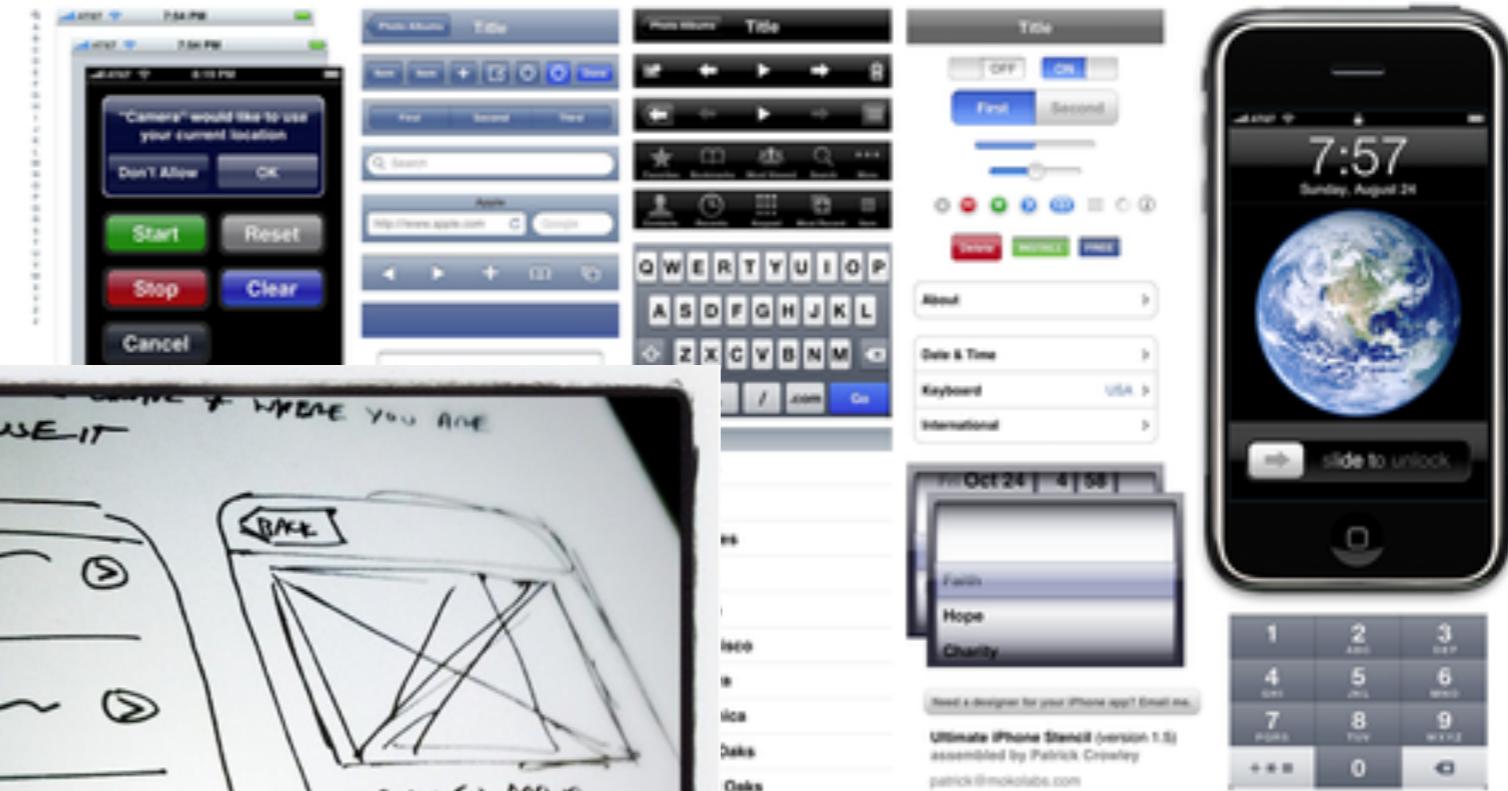
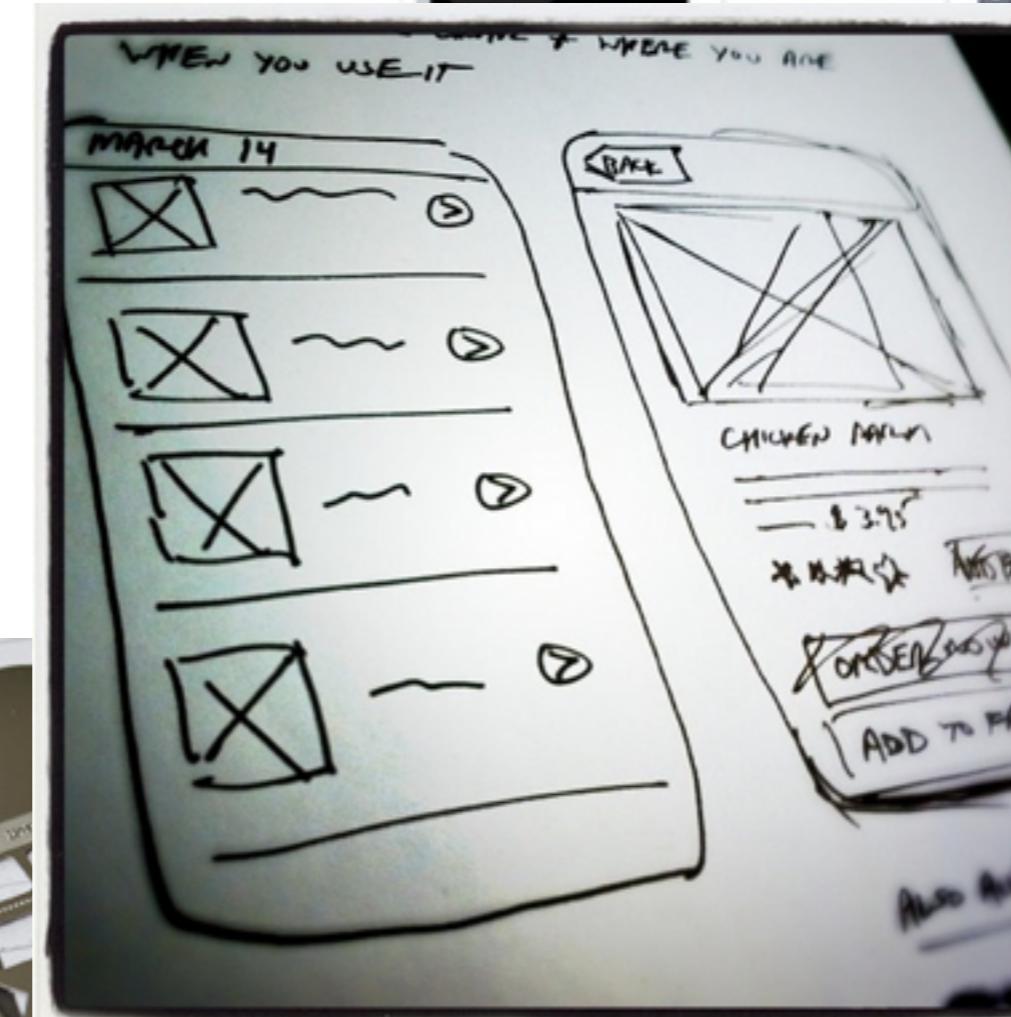
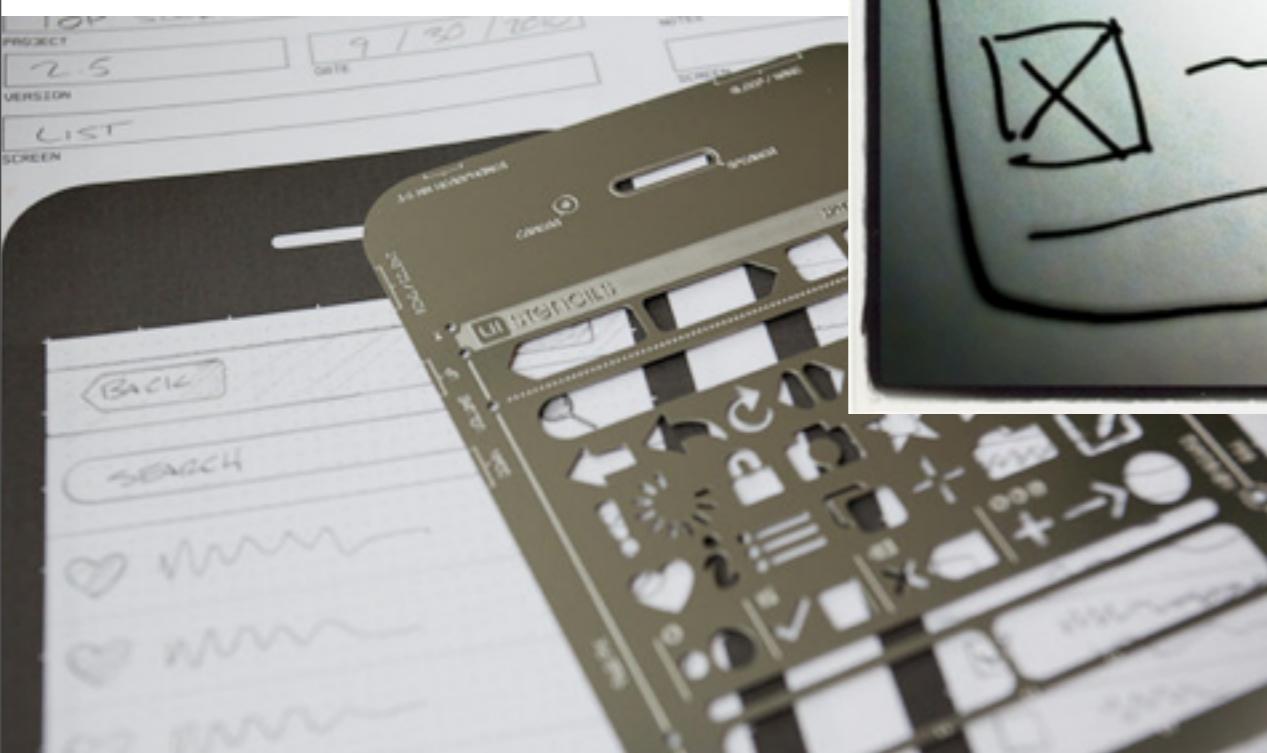


⌘ + click  
then  
Edit > Copy

The screenshot shows the iTunes application interface. At the top, it displays "iTunes" and the song "Never Alone" by BarlowGirl — WOW Hits 2005. The main content area shows an iPhone icon. To the right of the icon, device details are listed: Name: Chris Judd's iPhone, Capacity: 29.06 GB, Software Version: 4.3.5, Identifier (UDID): 9ee3f4f787a982ef9e7bee884518664018e1b0f7, and Phone Number: 1 (614) 378-4119. An arrow points from the text "Identifier (UDID): 9ee3f4f787a982ef9e7bee884518664018e1b0f7" down to the UDID value in the UDID app. At the bottom, there's a "Version" section with "Check for Update" and "Restore" buttons, and a note about the software being up to date.

# **DESIGN**

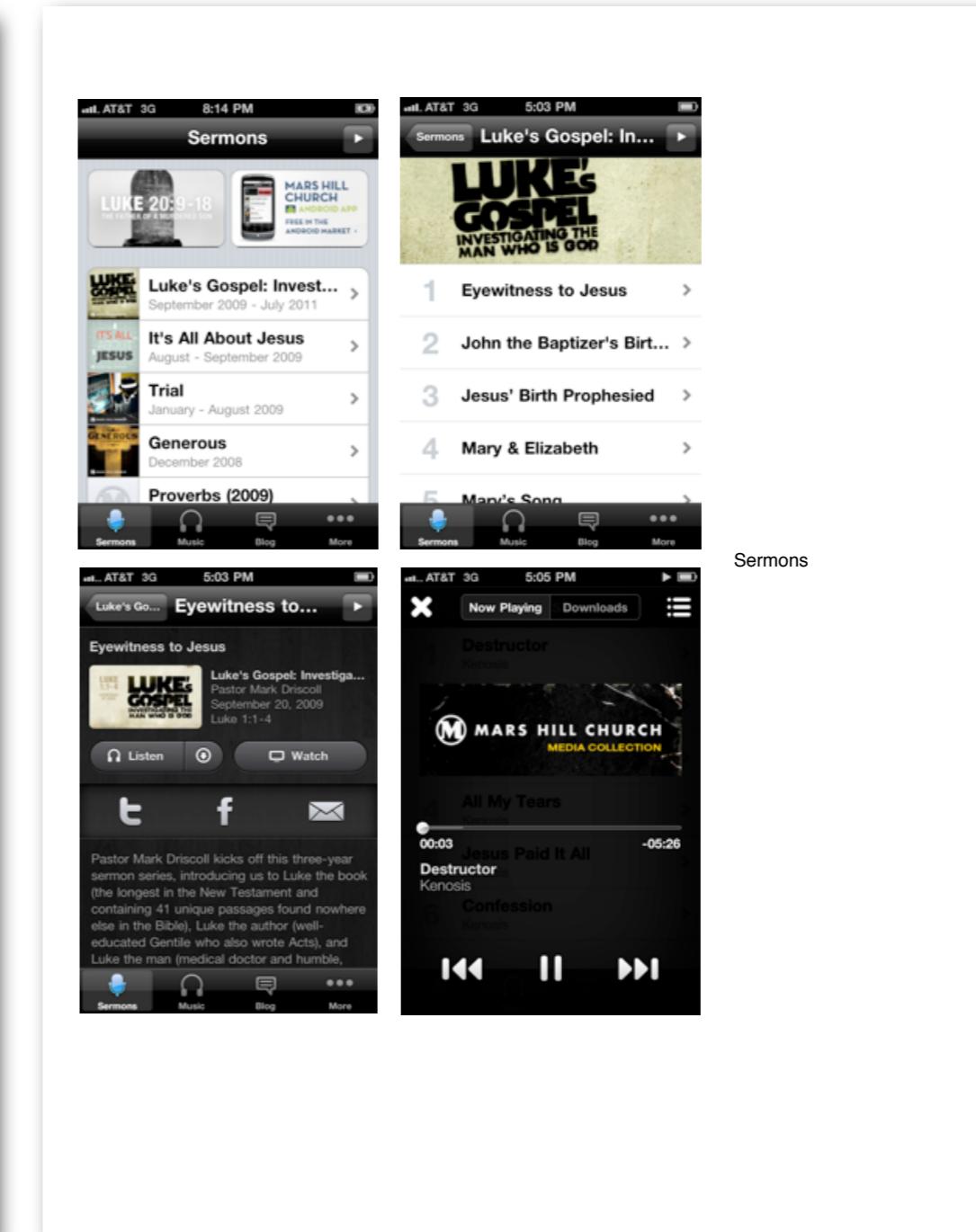
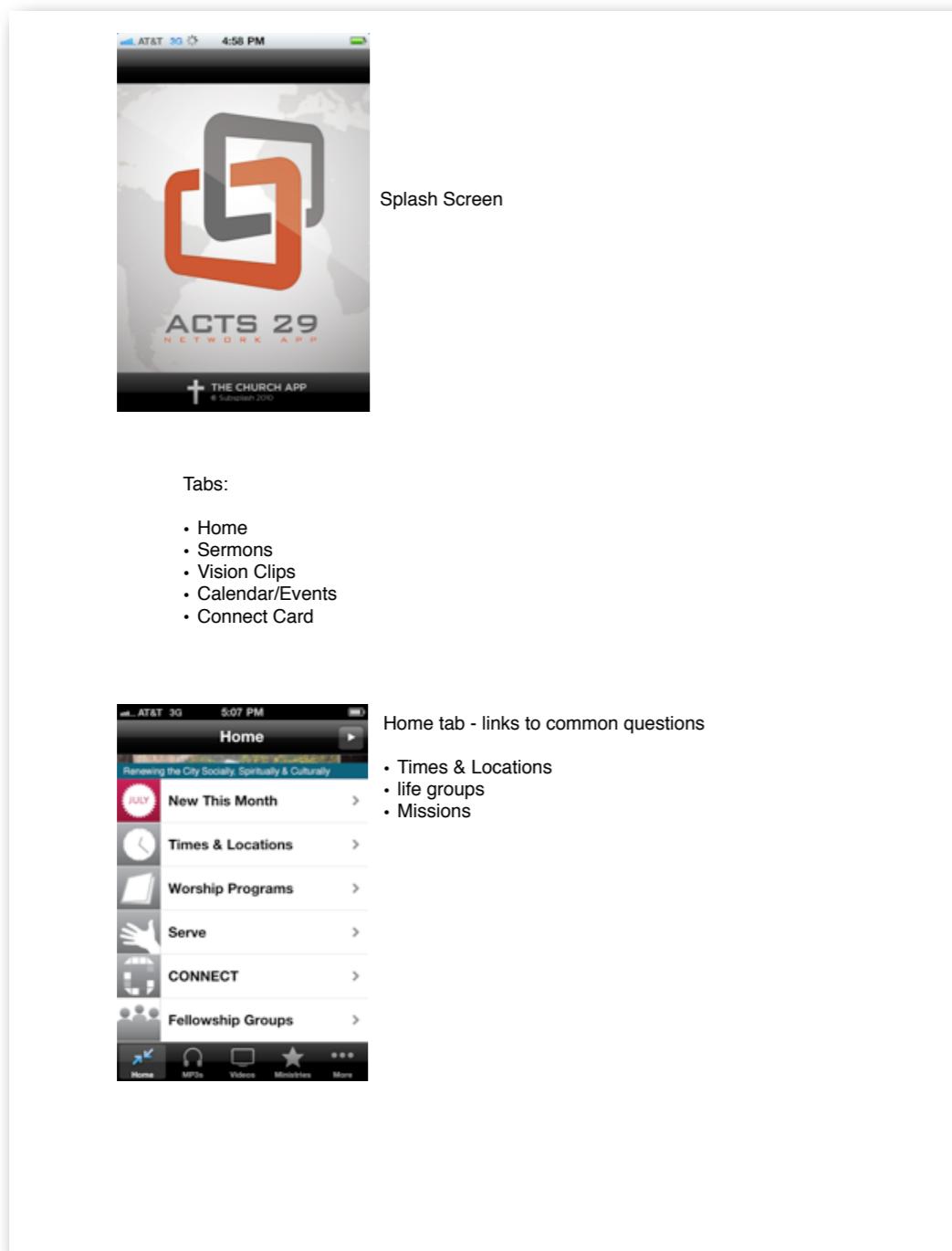
# Layout the design



<http://www.graffletopia.com/stencils/570>

<http://www.uistencils.com/products/iphone-stencil-kit>

# Storyboards



# iOS Human Interface Guidelines

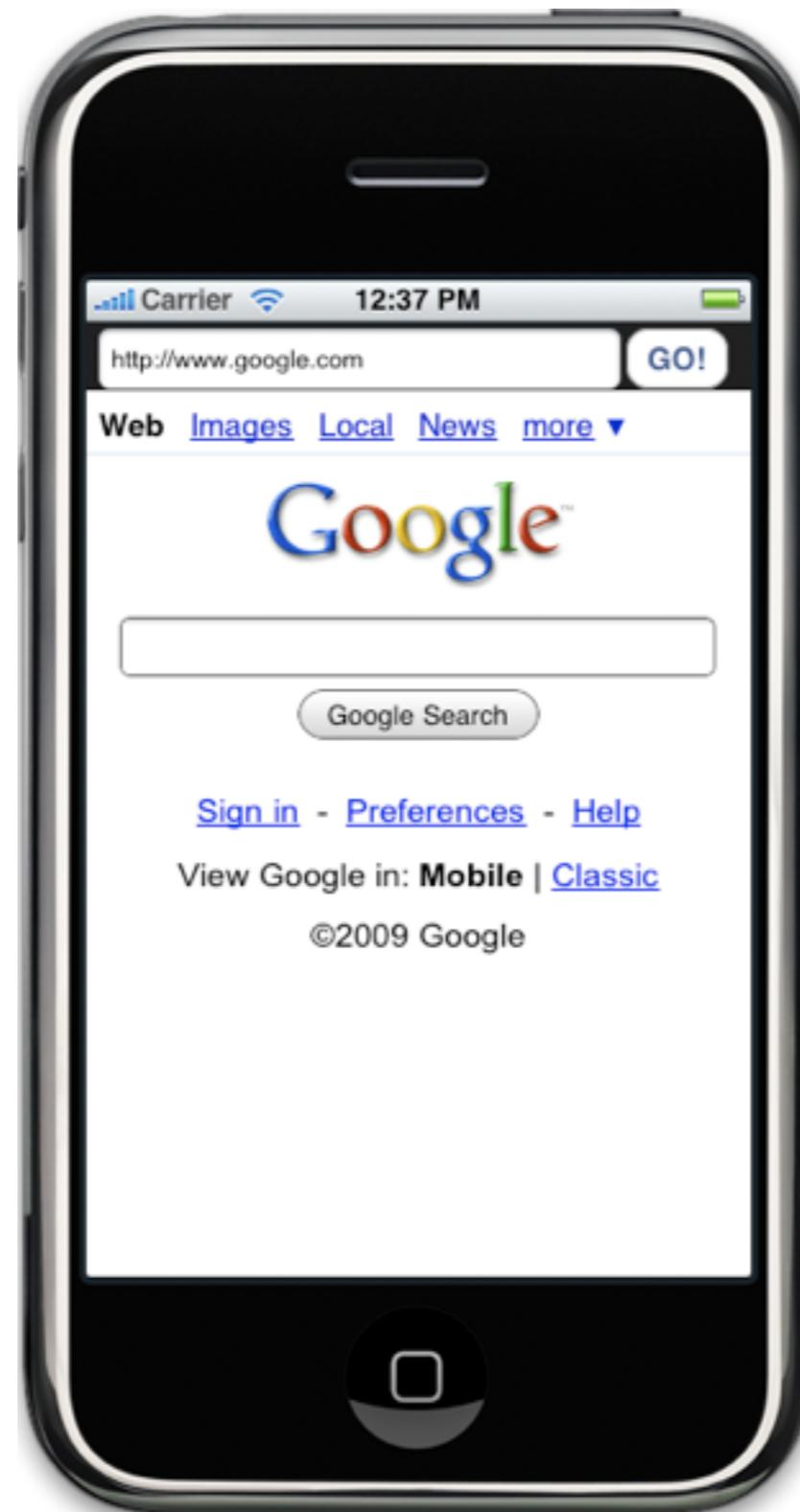
The screenshot shows a web browser window displaying the 'iOS Human Interface Guidelines' page from developer.apple.com. The browser's address bar shows the URL. The page itself has a dark header with the title 'iOS Human Interface Guidelines'. On the left is a sidebar with a 'Table of Contents' section containing links to various chapters like 'Introduction', 'Platform Characteristics', and 'User Experience Guidelines'. The main content area features a large heading 'Introduction' and a paragraph explaining the purpose of the guidelines. Below this is a composite image showing a physical iPhone next to a Mac screen. The Mac screen displays a wireframe diagram of an iPhone interface with various UI elements labeled ('counter', 'speaker', 'send indicators', 'Record', 'Play', 'Stop', 'Dial', 'Speaker', 'Volume', 'Home'). To the right of the wireframe is a list of design principles with checkboxes, some of which are checked (e.g., 'Focus: Primary'). Further to the right is a screenshot of an 'Audio Levels' application on the iPhone, and at the bottom is a screenshot of a recording application showing a waveform and playback controls. A vertical blue scroll bar is visible on the right side of the page.

<http://developer.apple.com/library/ios/#documentation/userexperience/conceptual/mobilehig/Introduction/Introduction.html>

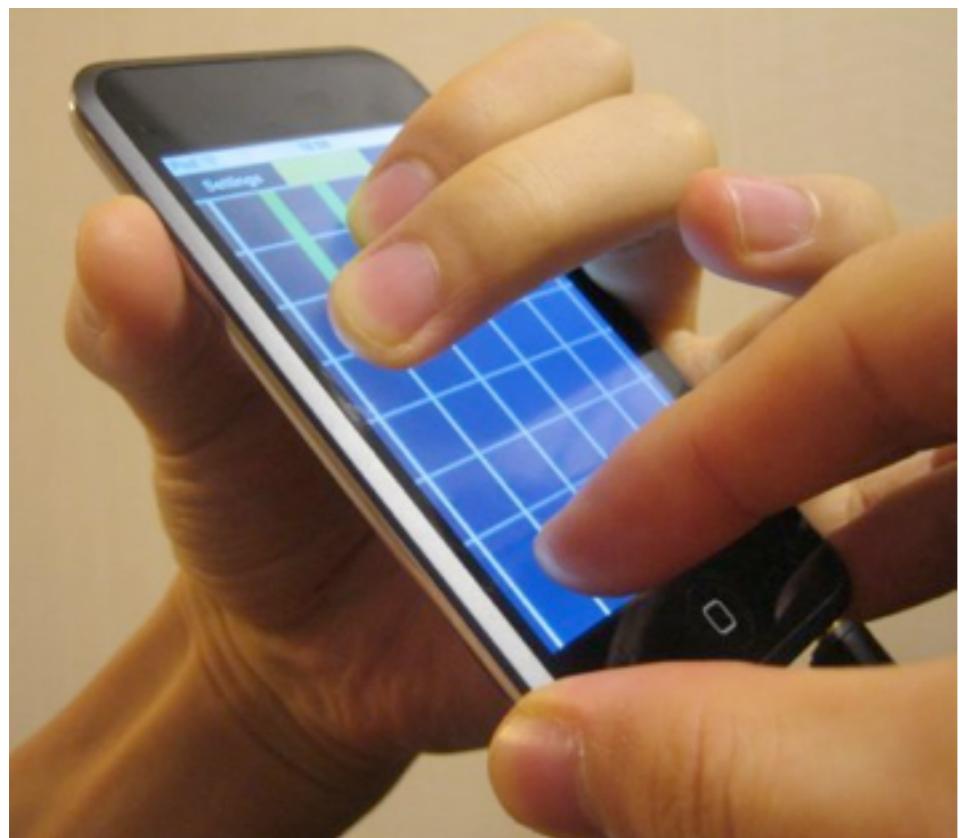
Be Compelling

**BE MOBILE**

# Mobile Yet Always Connected



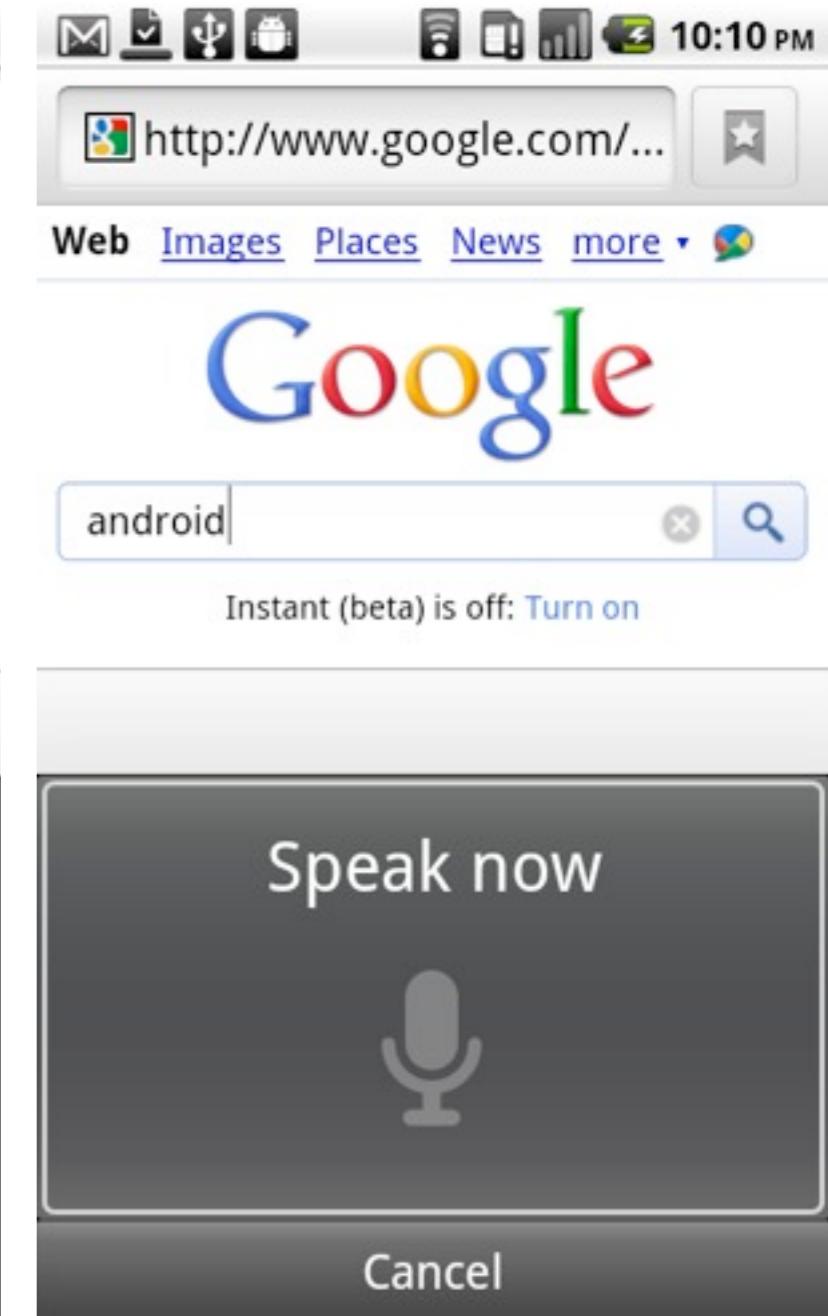
# Input



Multi-touch



Virtual Keyboard



Speech

# Location Aware



# Accelerometer/Gyroscope

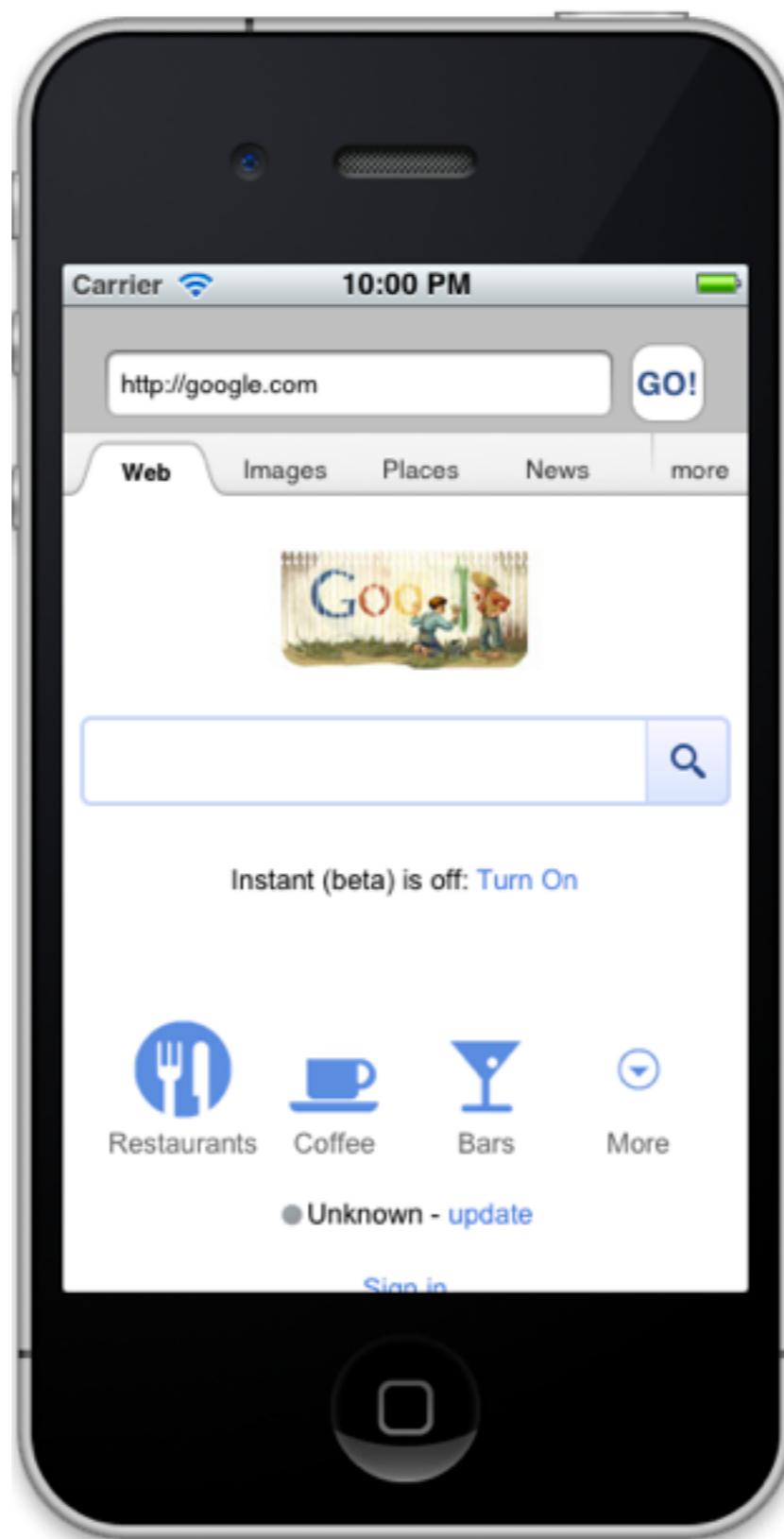


# Camera/Video



# **PROJECT 1**

# MY BROWSER



# **Learn the Basics**

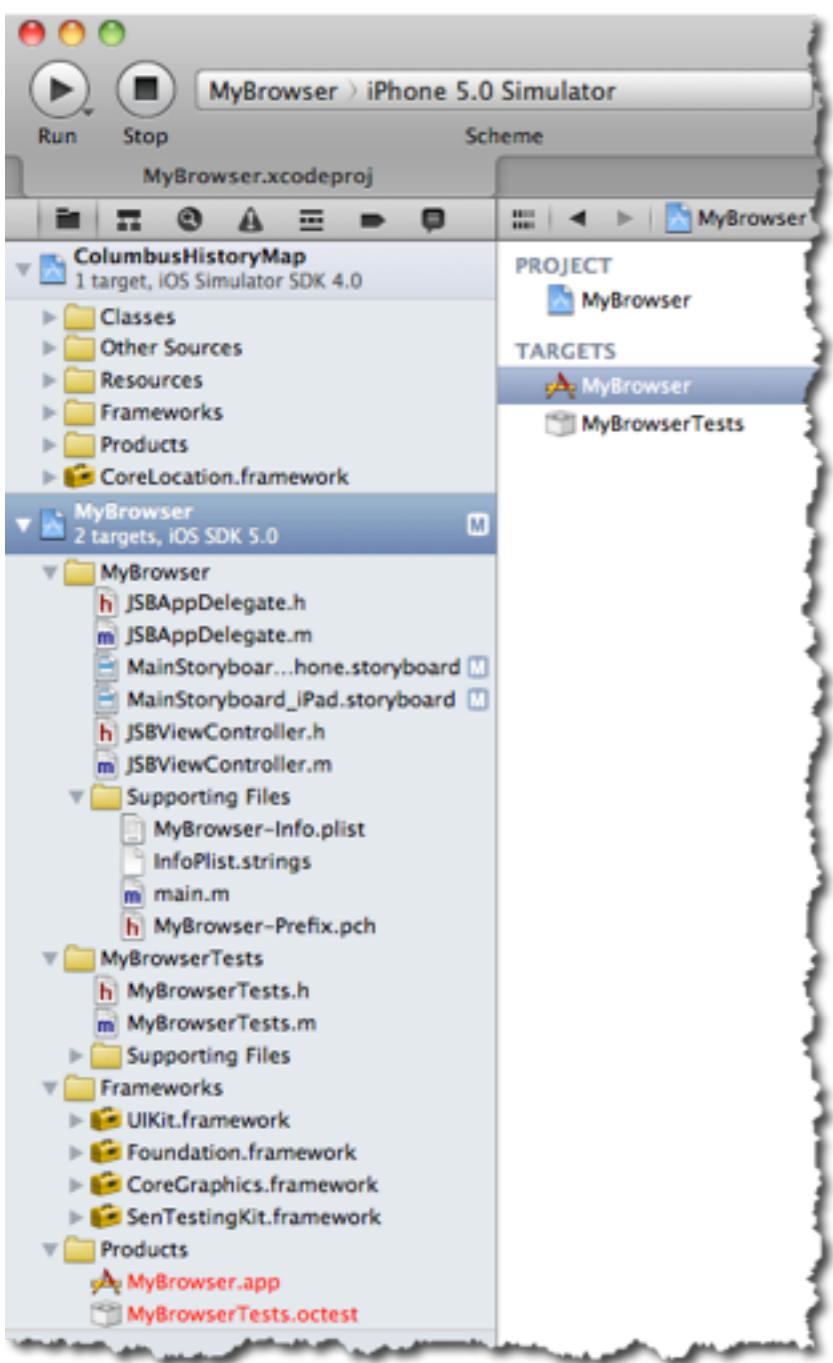
1. Create project 
2. Design interface 
3. Wire components to code  
4. Implement event handlers 
5. Run application 
6. Profile application 
7. Deploy the application

# **STARTING PROJECT**

# Create new workspace



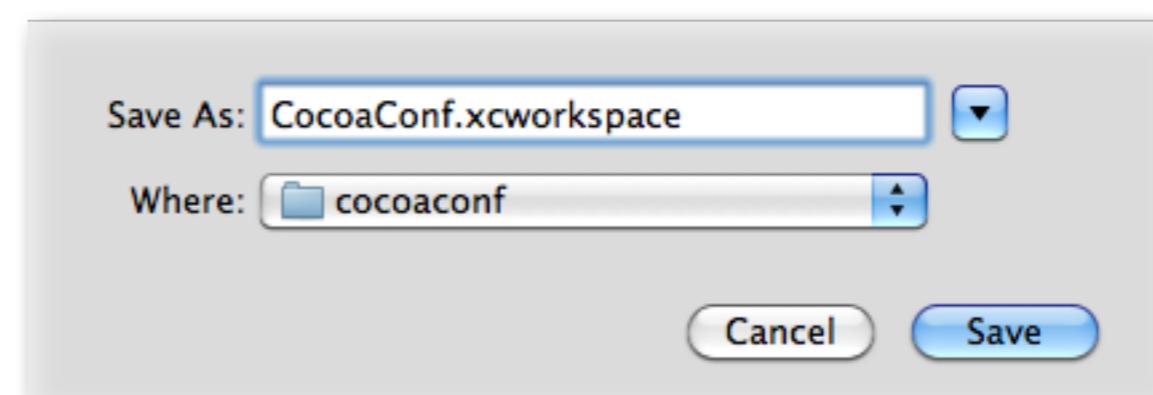
A workspace is an XCode document that groups projects and other documents so you can work on them together.



File > New Workspace...

or

⌘ ⌘ N



# Opening workspace



**Welcome to Xcode**  
Version 4.2 (4C199)

**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](http://developer.apple.com)

**Recents**

	<b>SendForHelp</b> ~/devl/workspaces/msw-mobile/iphone
	<b>CocoaConf</b> ~/devl/workspaces/cocoa/cocoaconf
	ColumbusHistoryMap ~/devl/workspaces/cocoa
	MyBrowser3 ~/devl/workspaces/cocoa
	uso ~/devl/workspaces/etech
	marker ~/devl/workspaces/etech
	Tabernacle ~/devl/workspaces/animman/tabernacle
	SendForHelp ...safe_old/sadrobot_msw/trunk/mobile/iphone

Last opened Today 10:32 PM

Show this window when Xcode launches

# Create new project



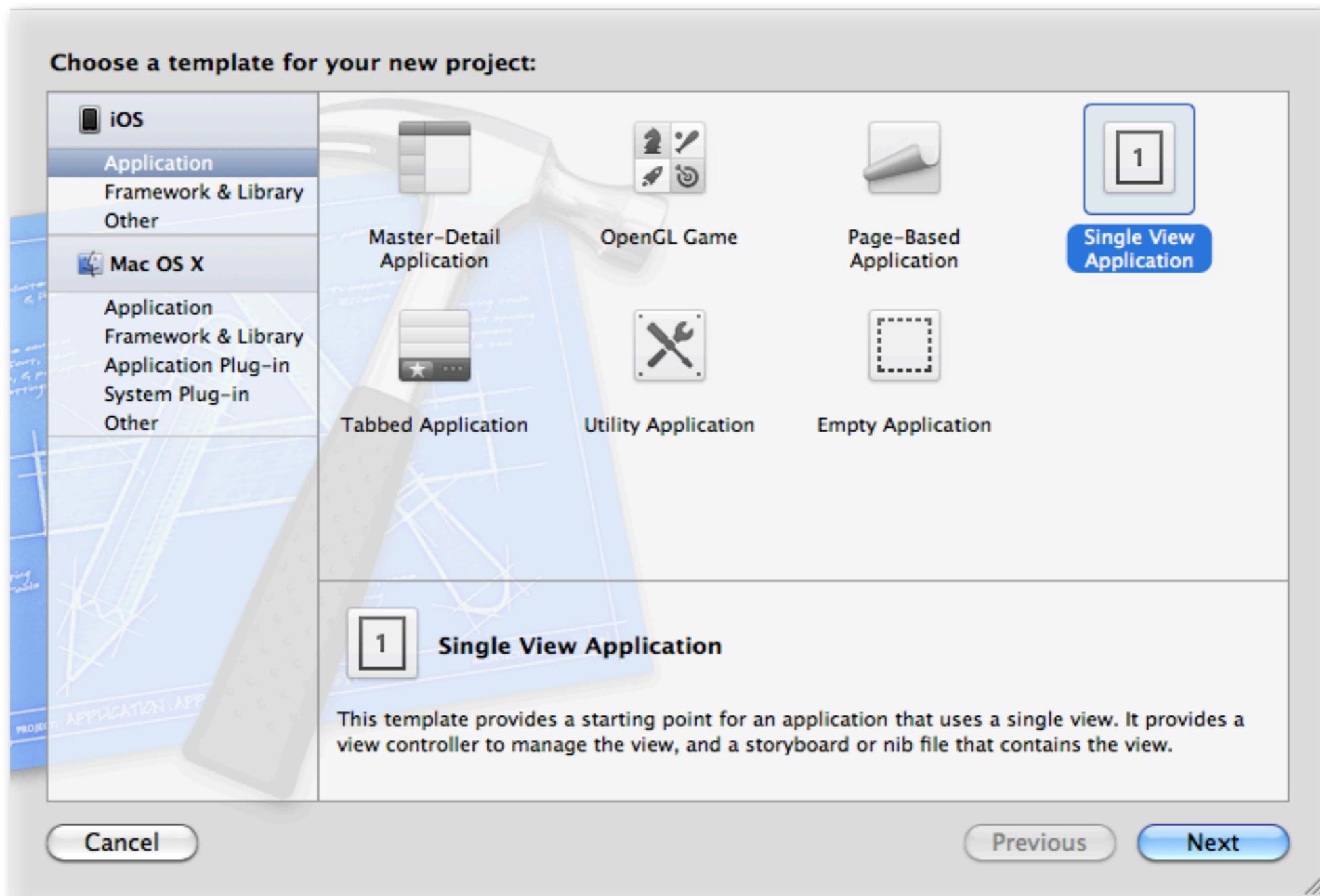
or

File > New Project...

or

⬆⌘N

# Choose project template



# Application Templates



Navigation-Based

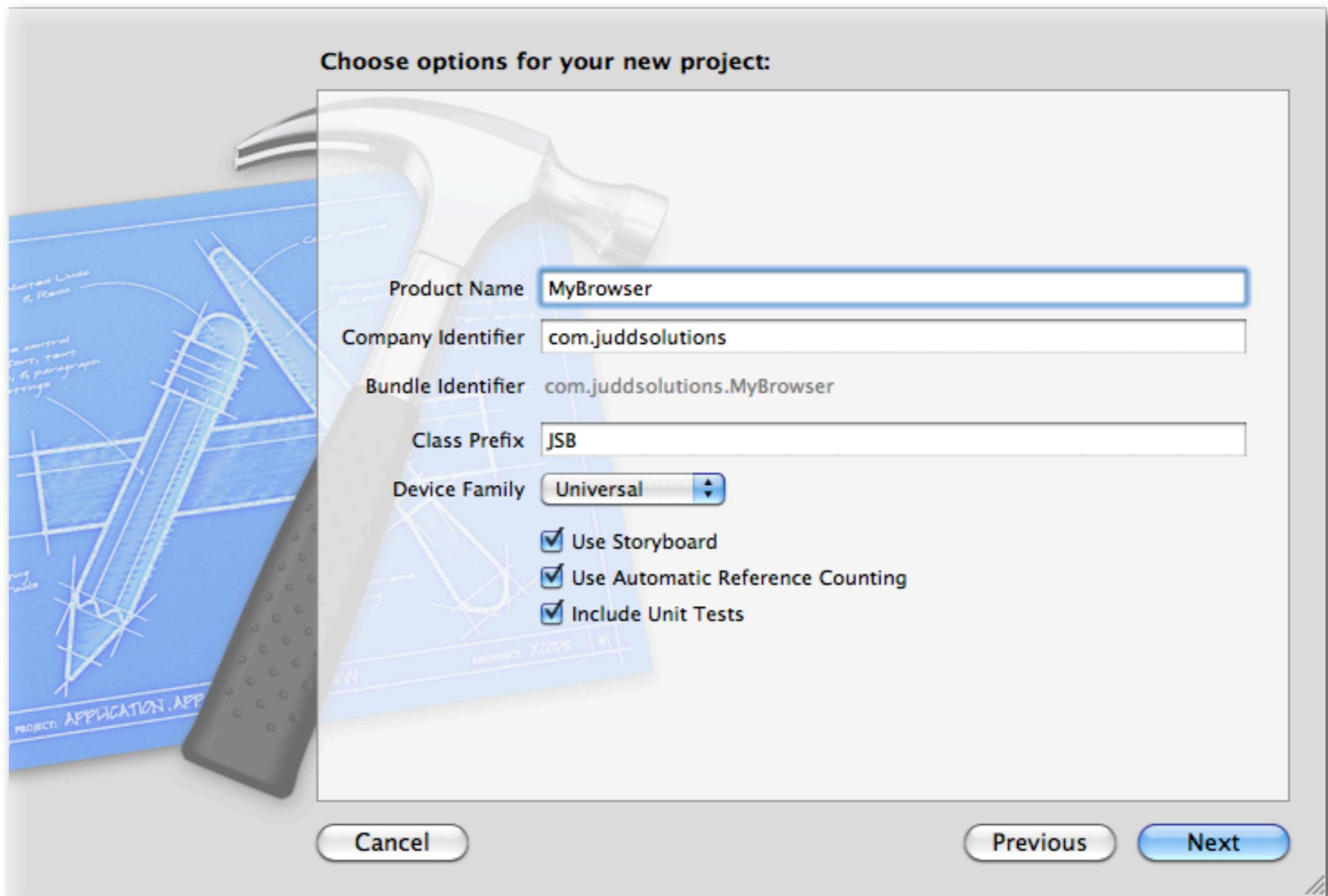
OpenGL ES

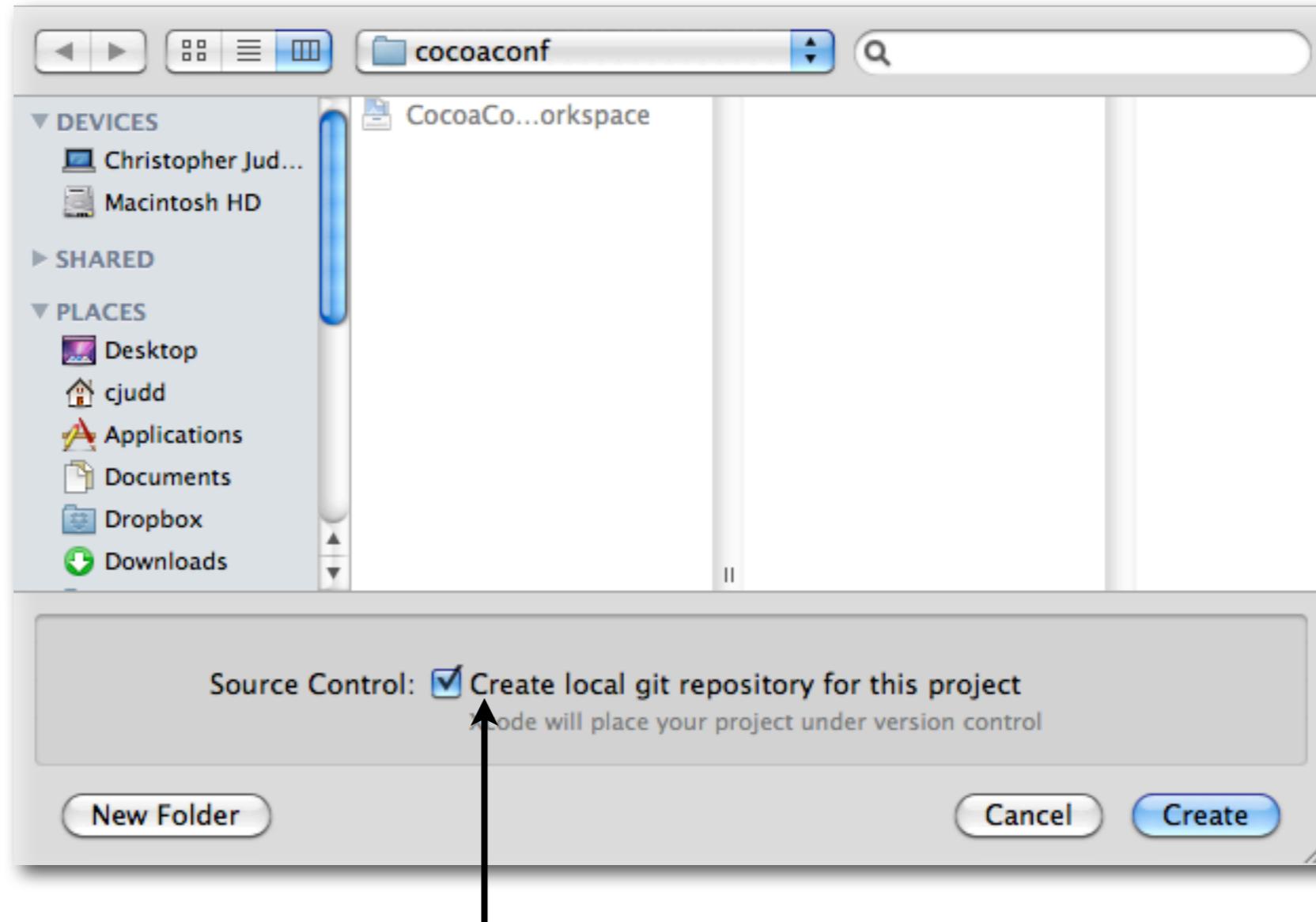
Tab Bar

Utility

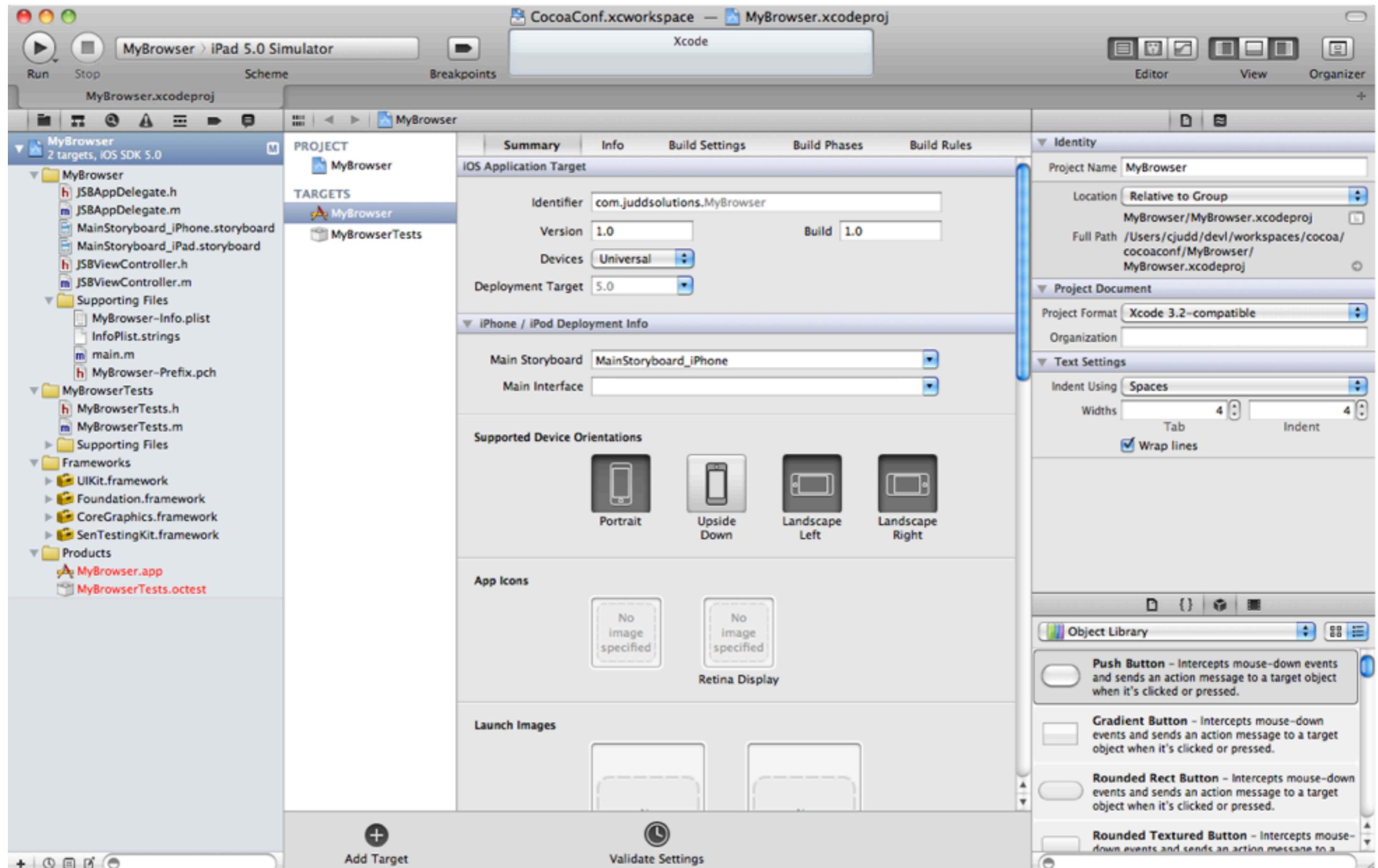
View-Based

Window-Based is total generic



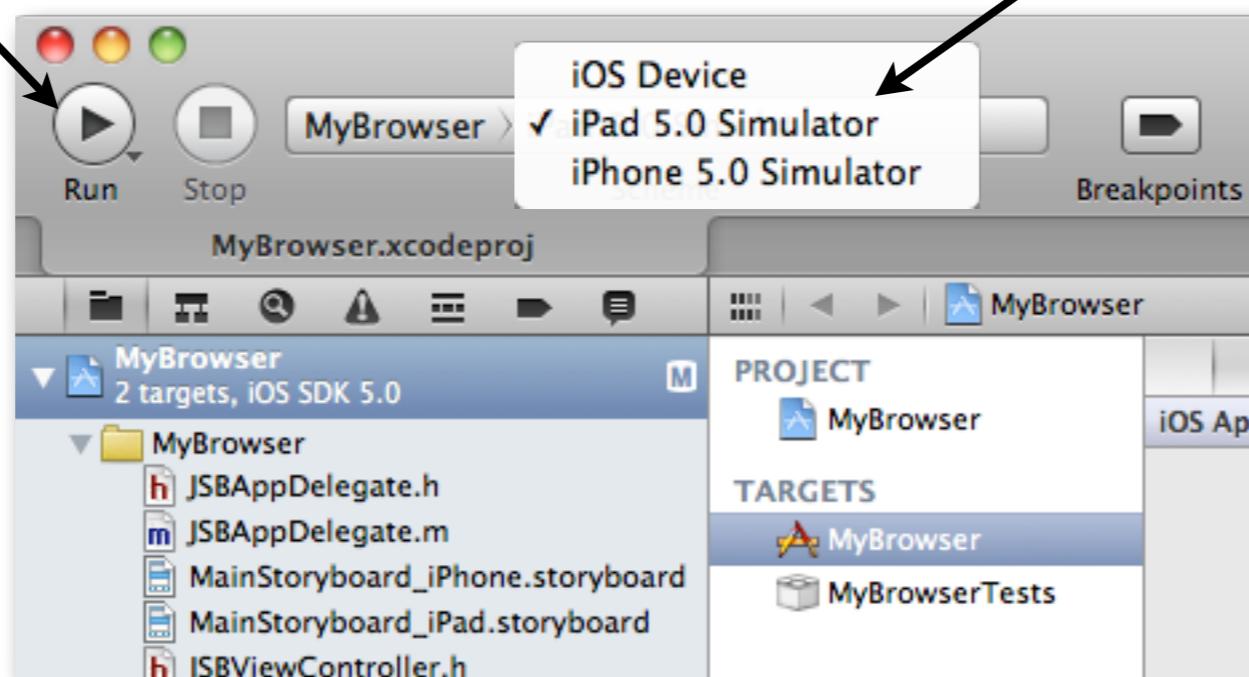


version control



# Running

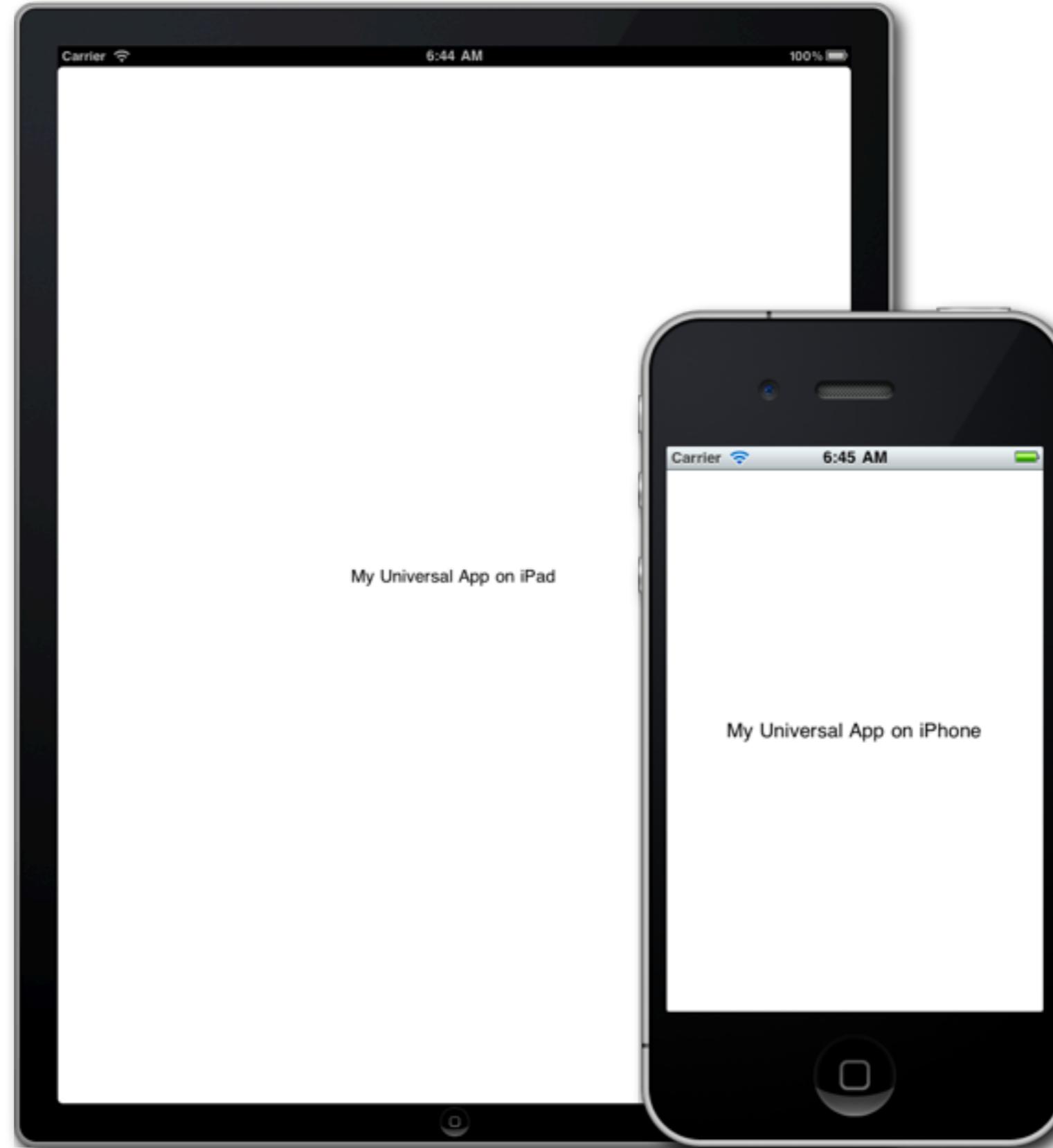
Run



or  
⌘ R

Select simulator or device

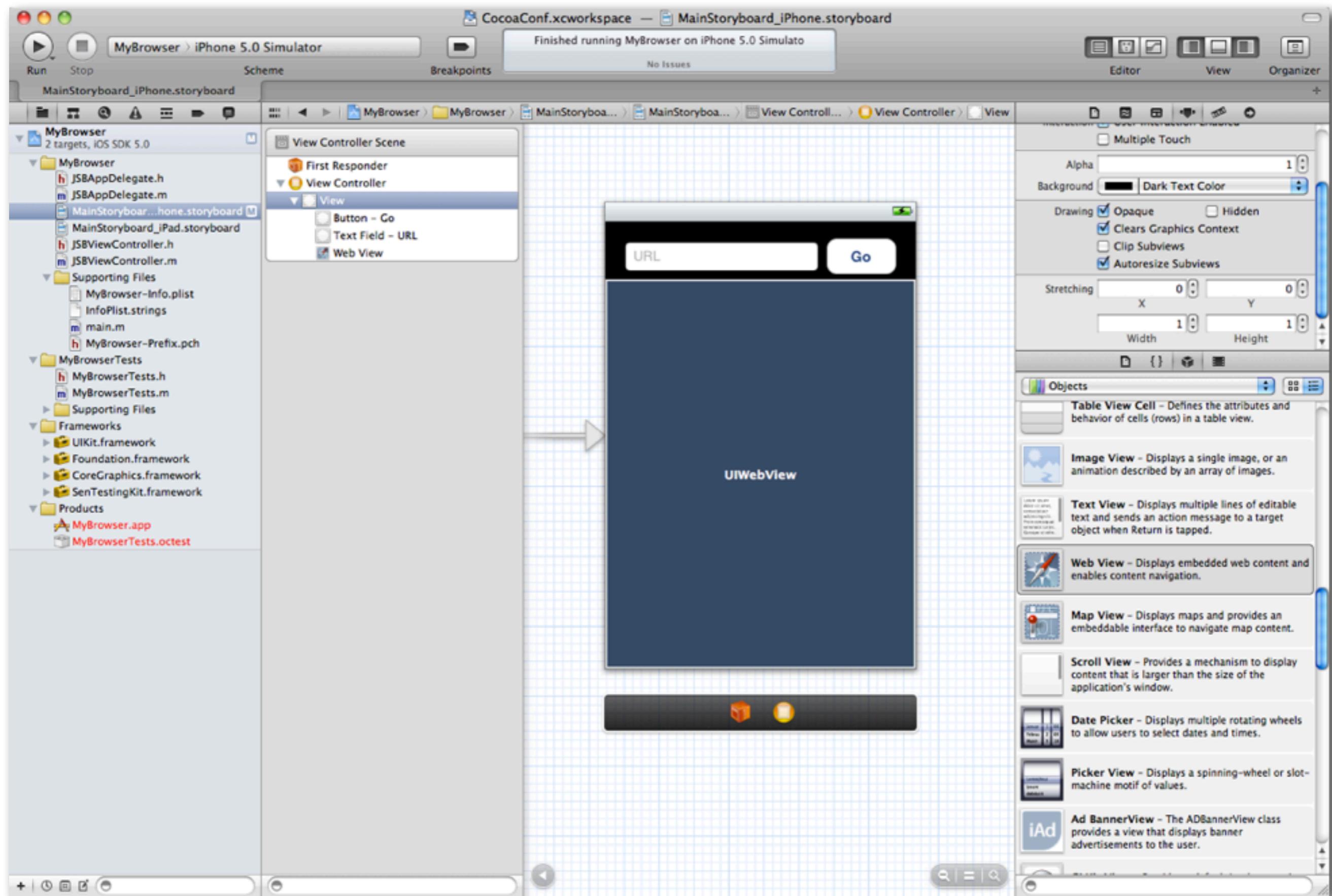
# iPhone & iPad Simulator



# *Lab 1*

- 1. Create a CocoaConf Workspace**
- 2. Create a MyBrowser Project with Git support for both iPhone and iPad**
- 3. Run the application in the simulator**

# **LAYING IT ALL OUT**



**Text Field**  
Placeholder: URL  
Keyboard: URL



**Go**

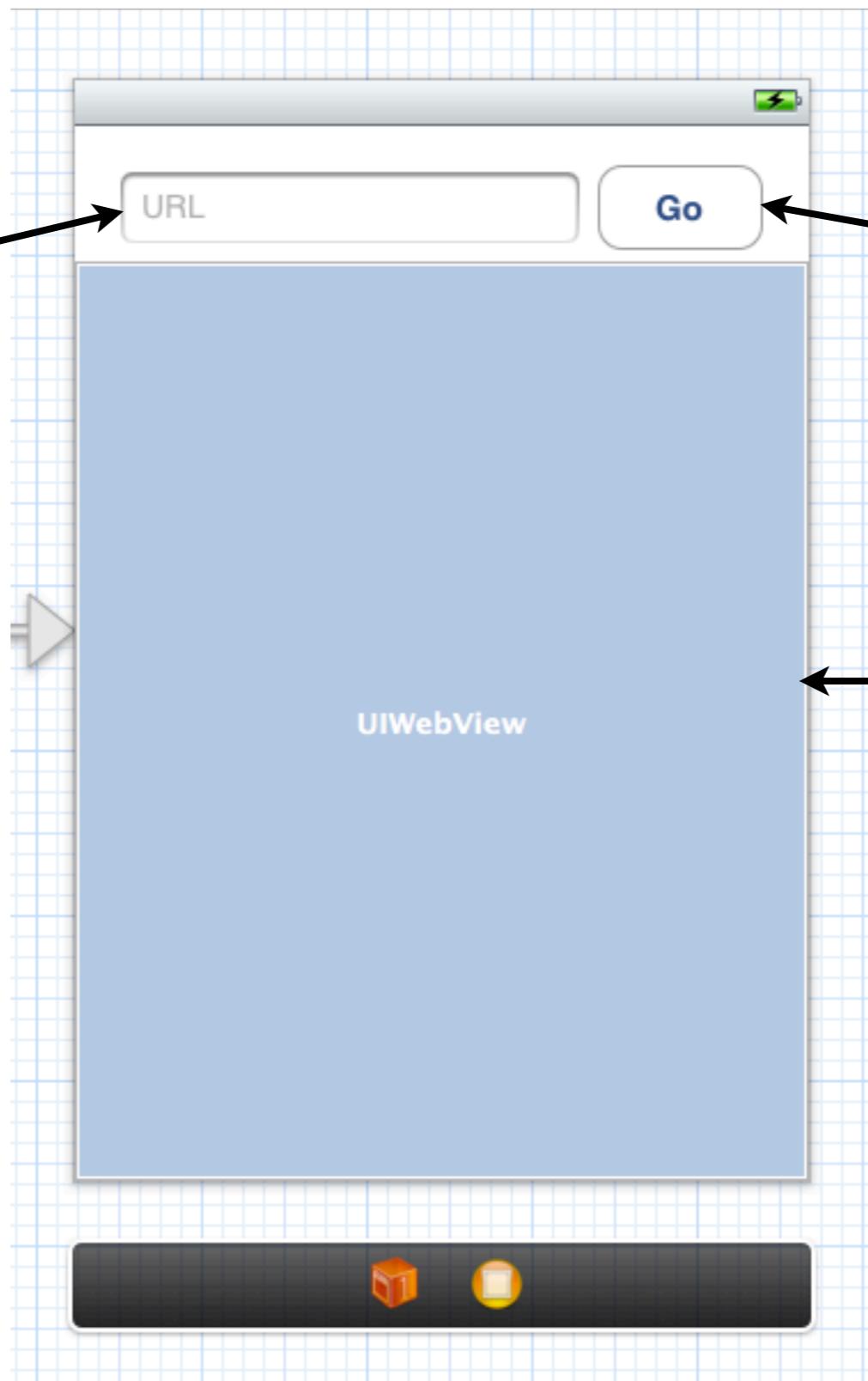


**Rounded Rect Button**  
**Title: GO!**



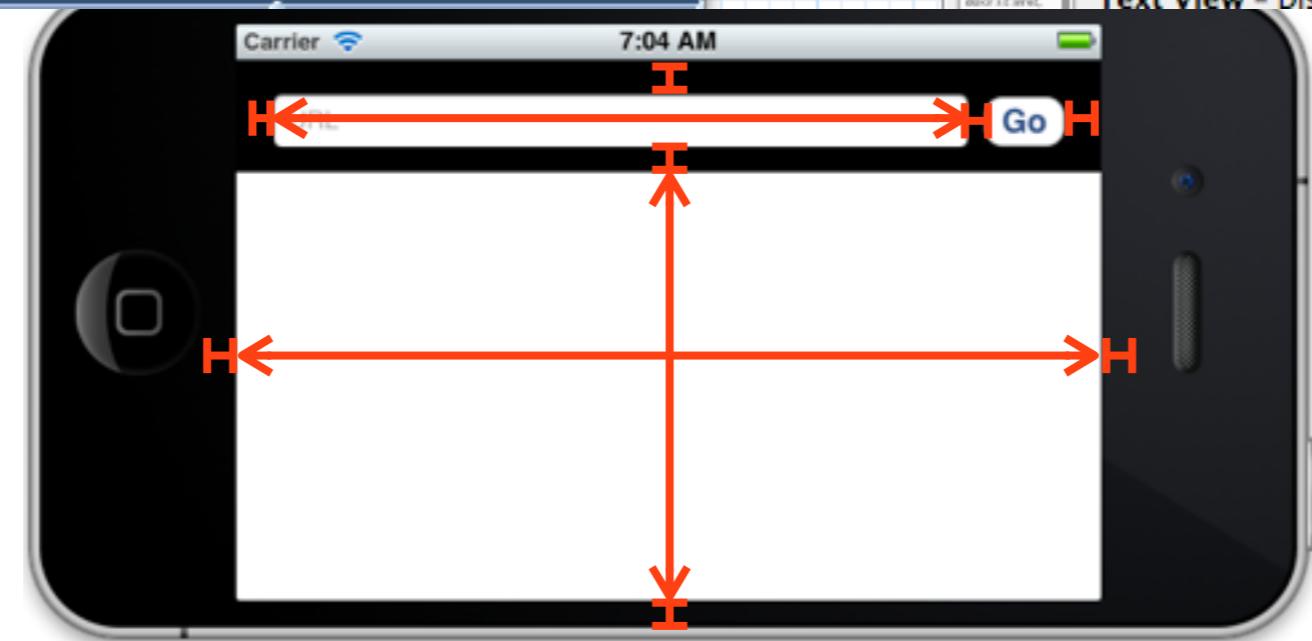
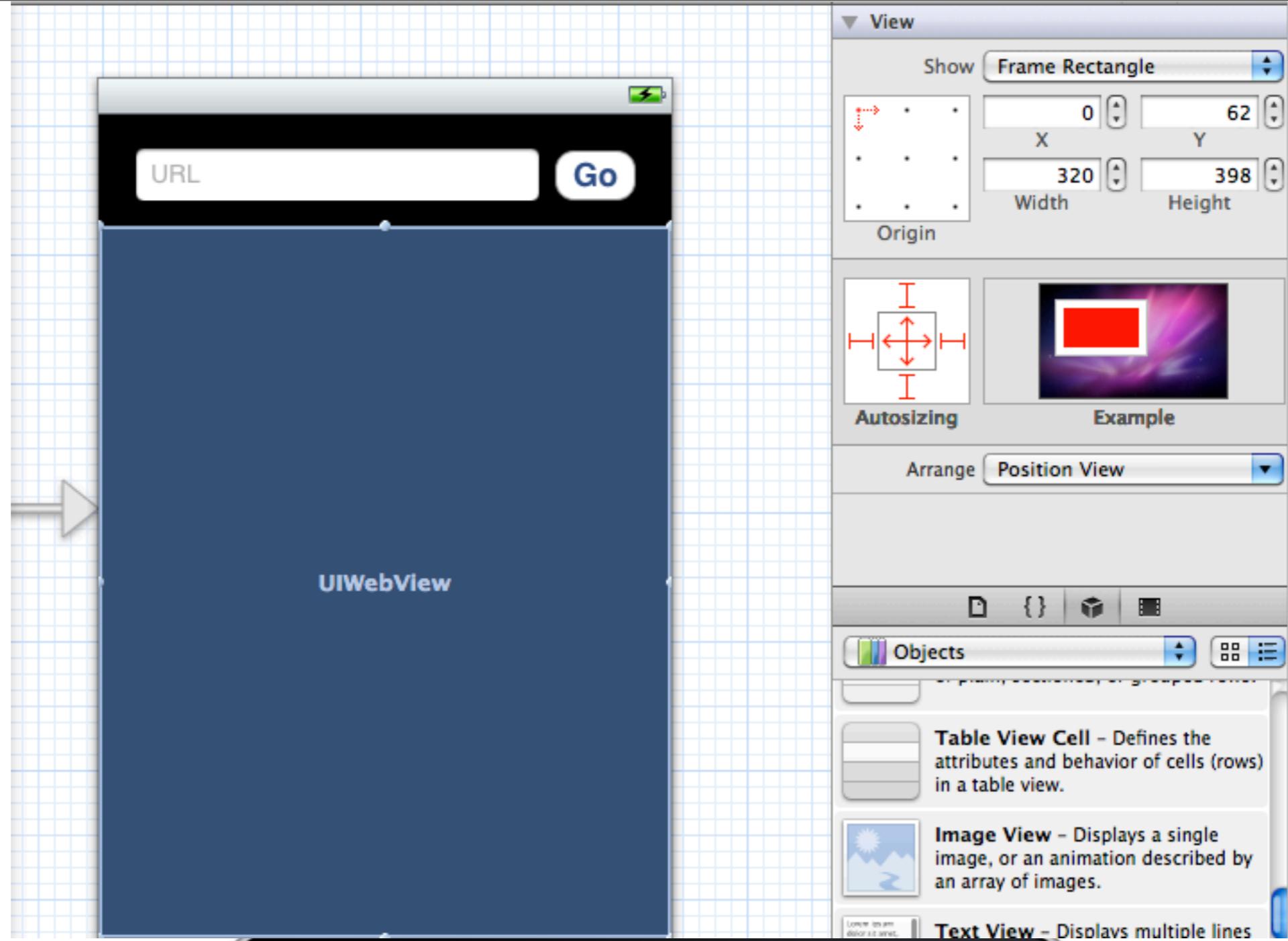
**Web View**

**UIWebView**









# *Lab 2*

1. Layout iPhone version of browser by adding Text Field, Rounded Rect Button and Web View
2. Layout iPad version of browser by adding Text Field, Rounded Rect Button and Web View
3. Run the application in the simulator
4. Experiment with some of the properties
5. Layout landscape

# OBJECTIVE-C PRIMER

# Classes

MyClass.h

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

@interface MyClass : NSObject {
    int count;
    id data; // Weak typing
    NSString* name; // String typing
}

- (id)initWithString:(NSString*)aName;
+ (MyClass*)createMyClassWithString:(NSString*) aName;

@end
```

Class name      Parent class name

Member variable declaration (ivars)

Instance method (constructor)

Class method (factory method)

MyClass.m

```
#import "MyClass.h"

@implementation MyClass

- (id)initWithString:(NSString *)aName
{
    self = [super init];
    if (self) {
        name = [aName copy];
    }
    return self;
}

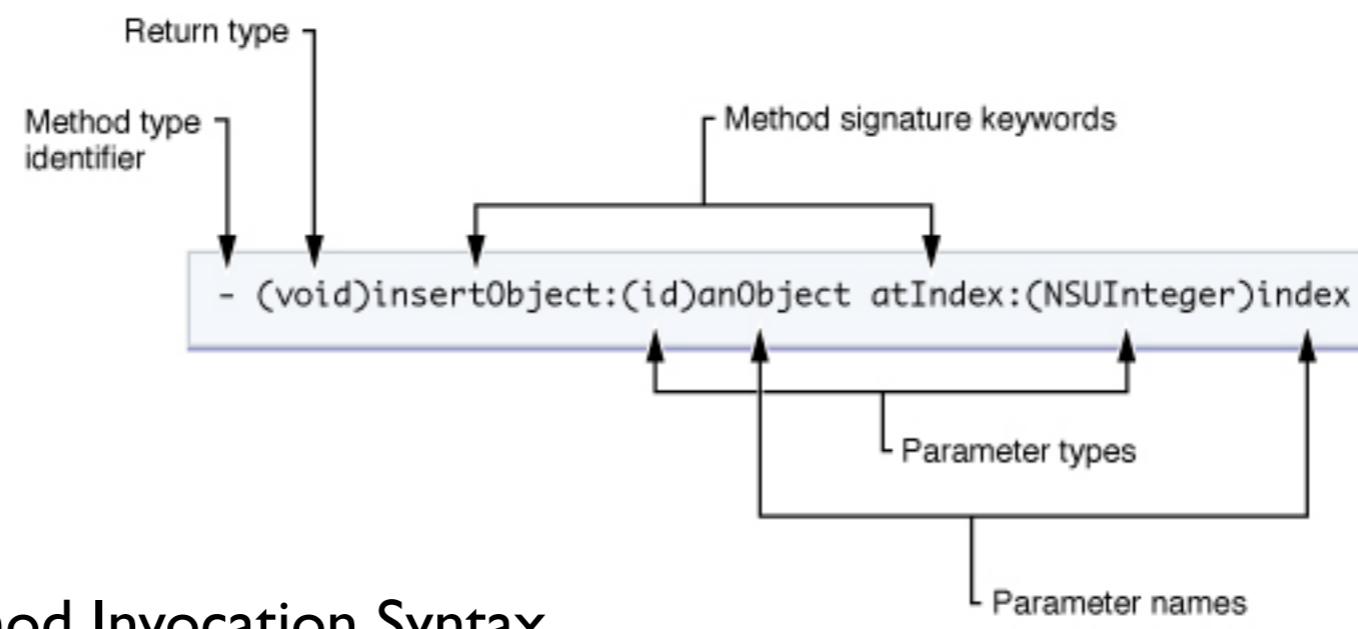
+ (MyClass *)createMyClassWithString: (NSString *)aName
{
    return [[[self alloc] initWithString:aName] autorelease];
}

@end
```

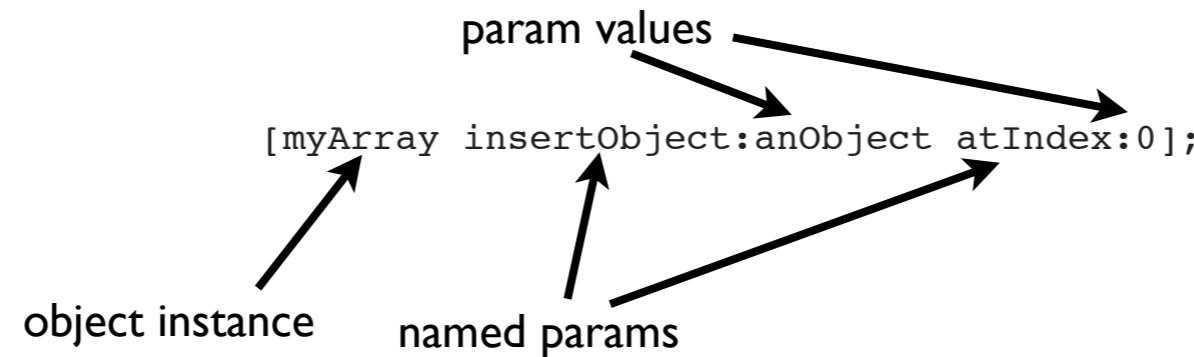
allocate memory    constructor call    memory management

# Methods and Messaging

## Method Declaration Syntax



## Method Invocation Syntax



# Properties and Synthesize

MyClass.h

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

@interface MyClass : NSObject {
    NSString* title;
}

// use of @property declaration for the title variable is equivalent to
// declaring a "setTitle" mutator and "title" accessor method.
@property (retain, nonatomic) NSString* title;

@end
```

MyClass.m

```
#import "MyClass.h"

@implementation MyClass

// use the @synthesize declaration in the .m implementation file
// to automatically implement setter and getter methods.
@synthesize title;

@end
```

# Property Usage

```
myClass.title = @"title"  
id title = myClass.title
```

=

```
[myClass setTitle: @"title"]  
id title = [myClass title]
```

# Protocols/Delegates

JSBViewController.h

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

@interface JSBViewController : UIViewController <UITextFieldDelegate>

@property (weak, nonatomic) IBOutlet UITextField *urlField;
@property (weak, nonatomic) IBOutlet UIWebView *webView;

- (IBAction)browseTo:(id)sender;

@end
```

implements protocol



JSBViewController.m

```
#import "JSBViewController.h"

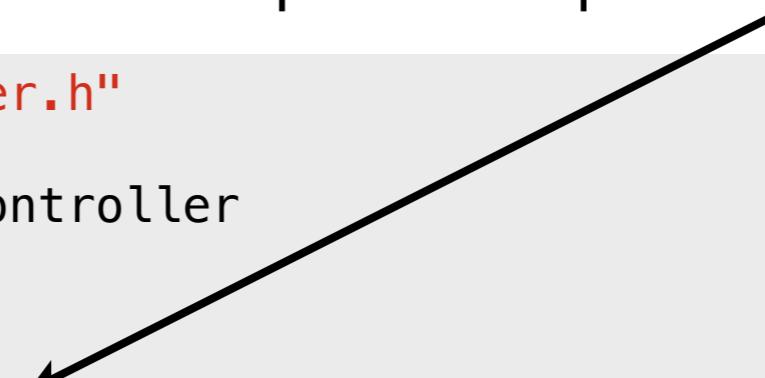
@implementation JSBViewController
@synthesize urlField;
@synthesize webView;

- (BOOL)textFieldShouldReturn:(UITextField *)textField {
    [self browseTo:textField];
    return YES;
}

// removed code for brevity

@end
```

implement required and/or optional methods



**CONTENTS INCLUDE:**

- An Introduction to the Language and Tools
- The Syntax
- Memory Management
- Tools
- Debugging
- XCode Keyboard Shortcuts and more...

**AN INTRODUCTION TO THE LANGUAGE AND TOOLS**

Objective-C is the primary language used to create applications for Apple's Mac OS X and iOS (iPhone and iPad) platforms. Objective-C was created by Brad Cox and Tom Love in the early 1980s. In 1988, Objective-C was licensed by NeXT, a company founded and helmed by Steve Jobs during his absence from Apple. Apple acquired NeXT in 1996, bringing Objective-C to the Macintosh platform.

Objective-C is an object oriented superset of ANSI C. Its object syntax is derived from Smalltalk. It supports single inheritance, implementation of multiple interfaces via the `@protocol` syntax, and the redefinition and augmentation of (open) classes via categories. Apple's iPhone SDK for the iOS mobile operating system offers developers a rich set of Objective-C APIs. This free SDK, which includes the Xcode IDE, is used to create applications for the iPhone, iPad, and iPod Touch.

**THE SYNTAX****Class Declaration**

Objective-C classes typically include an interface .h and an implementation .m pair of files. The .h file contains property and method declarations. The .m file contains method implementations.

**Example .h interface file**

```
#import <Foundation/Foundation.h>
@interface Speaker : NSObject {
    NSInteger *ID;
    NSString *name;
}
@property NSInteger *ID;
@property(nonatomic,retain) NSString *name;
- (void)doSomething: (NSString *) value anotherValue: (int) value2;
@end
```

**Example .m implementation file**

```
#import "Speaker.h"
@implementation Speaker
@synthesize ID,name;
- (void)doSomething: (NSString *) value anotherValue: (int) value2 {
    // do something
}
@end
```

**Inheritance**

Class inheritance is specified in the .h interface file with the syntax: `@interface <MyClass> : <ParentClass>`. The following example tells the compiler that the Employee class inherits from (extends) the Person class.

```
// Employee.h file
@interface Employee : Person {
}
@end
```

# Objective-C

## for the iPhone and iPad

By Ben Ellingson and Matthew McCullough



The keyword `@interface` can distract developers coming from some languages such as Java, suggesting this is a mere contract. However, `@interface` is indeed the keyword for defining the properties and method signatures of a concrete class in Obj-C.

**Interfaces**

Objective-C interfaces are created using the `@protocol` declaration. Any class can implement multiple interfaces. Interfaces are typically declared in a .h header file and can be included via `#import` statements in the .h header file of other classes.

```
// Mappable.h: Declare the Mappable @protocol
@protocol Mappable
- (double) latitude;
- (double) longitude;
@end

// Location.h: Specify that Location class implements the Mappable
// @protocol
#import "Mappable.h"
@interface Location : NSObject <Mappable> {
}
@end

// Location.m: Provide implementations for the Mappable methods
@implementation Location
- (double) latitude {
    return 46.553666;
}
- (double) longitude {
    return -87.40551;
}
@end
```

**Primitive Data Types**

As a superset of ANSI C, Objective-C supports its same primitive data types.

int	Integral numbers without decimal points
float	Numbers with decimal points

**Don't Miss An Issue!**

Get over 90 DZone Refcardz  
FREE from [Refcardz.com](http://Refcardz.com)!



Visit [Refcardz.com](http://Refcardz.com) to get them all Free!

---

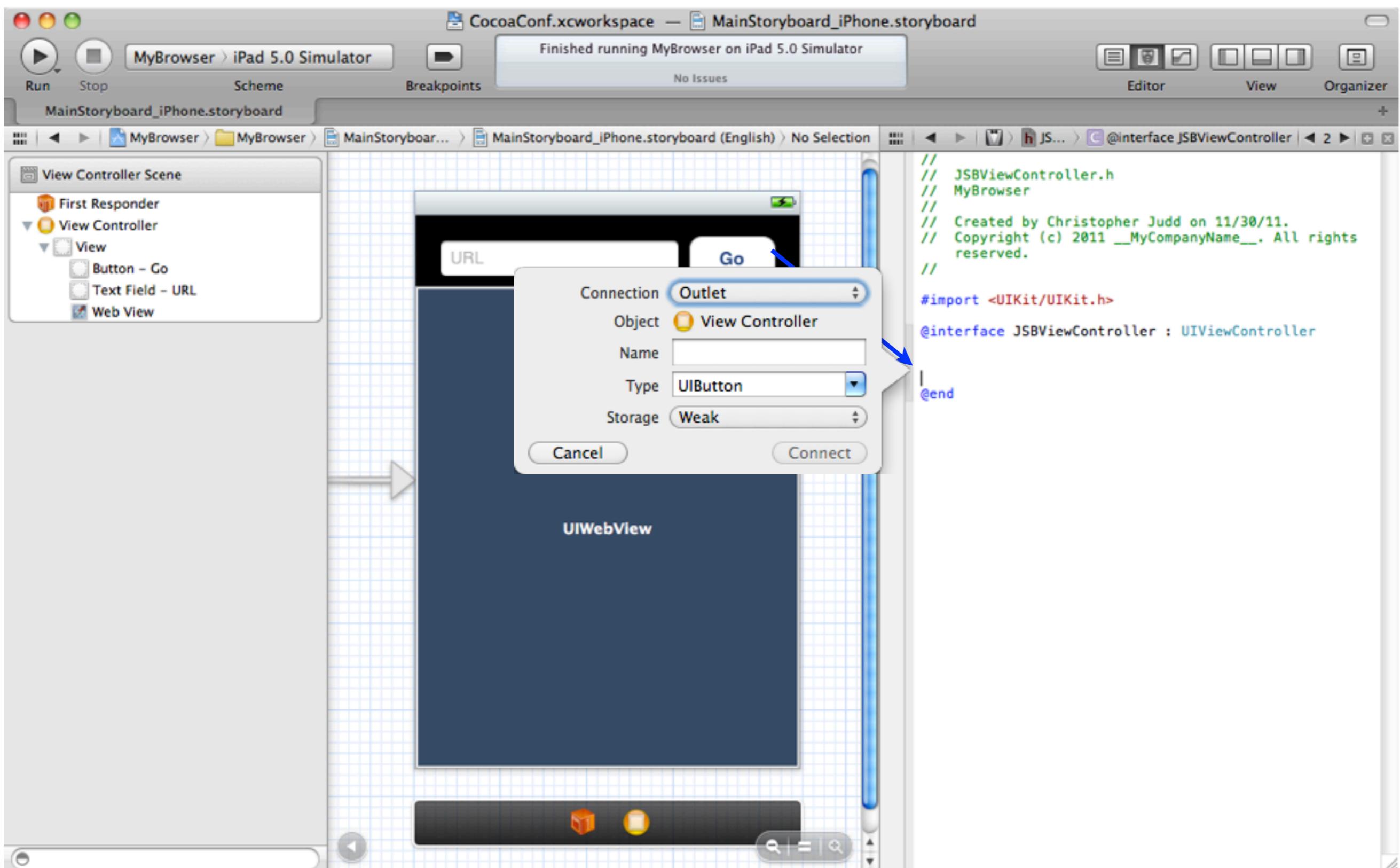
## The Objective-C 2.0 Programming Language



2008-02-08

<http://developer.apple.com/library/mac/documentation/cocoa/conceptual/objectivec/objc.pdf>

# CODING IT UP



# \*ViewController.h

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

@interface JSBViewController : UIViewController

@property (weak, nonatomic) IBOutlet UITextField *urlField;
@property (weak, nonatomic) IBOutlet UIWebView *webView;

- (IBAction)browseTo:(id)sender;

@end
```

# \*ViewController.m

```
#import "JSBViewController.h"

@implementation JSBViewController

@synthesize urlField;
@synthesize webView;

#pragma mark - View lifecycle
// Removed other stubbed methods for brevity

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

- (IBAction)browseTo:(id)sender {
    // hide keyboard
    [urlField resignFirstResponder];
    NSURL *url = [[NSURL alloc] initWithString:urlField.text];
    NSURLRequest *request = [[NSURLRequest alloc] initWithURL: url];
    [webView loadRequest:request];
}

@end
```

# *Lab 3*

- 1. Add Text Field and WebView outlets**
- 2. Add browseTo event handler**
- 3. Run the application in the simulator**



\$

# **DEPLOYING APP**



- Development (device)
- Ad Hoc
- App Store

# **DEPLOYING TO DEVICE**

# Team Management

The screenshot shows the Apple Developer Member Center interface. At the top, there is a navigation bar with links for Technologies, Resources, Programs, Support, Member Center, and a search bar. Below the navigation bar is a secondary header with tabs for iOS Dev Center, Mac Dev Center, and Safari Dev Center. The main content area displays sections for iOS SDK 4.3, iOS SDK beta, Resources for iOS 4.3, Featured Content, and the iOS Developer Program. A prominent arrow points from the 'People' link in the top navigation to the 'All People' section in the main content area. The 'All People' section lists 'Christopher Judd' and another user whose name is partially obscured. To the right of the 'All People' list, there is a 'Member Center' sidebar with links for the iOS Provisioning Portal and a 'Developer Program Overview' section.

Developer

Technologies Resources Programs Support Member Center Search Developer

iOS Dev Center Mac Dev Center Safari Dev Center

Hi, Christopher Judd | My Profile | Log out

iOS SDK 4.3 iOS SDK beta

Resources for iOS 4.3

Featured Content

iOS Developer Program

iOS Provisioning Portal

Member Center

Downloads Download the latest build of iOS SDK

Getting Started Videos Watch Apple experts discuss a range of topics

Organization: Judd Solutions, LLC

Hi, Christopher Judd | Log out

Developer Program Resources

Technical Resources and Tools

Dev Centers Quickly access a range of technical resources.

iOS Mac Safari

iOS Provisioning Portal Provision & test your apps on your iPhone, iPad & iPod touch.

Developer Program Overview

Read about accessing the resources and benefits of your iOS Developer Program

Apple Developer Member Center

People Programs & Add-ons Your Account

Organization: Judd Solutions, LLC

All People

Christopher Judd

iOS Developers

Invitations

iOS Developer Program

Agent

Member

# Roles

The screenshot shows a user profile with a blurred name. Below it is a 'Contact Information' section with an email field containing a blurred address. Under 'Access and Roles', there's a section for 'iOS Developer Program' with three radio button options: 'No Access' (unselected), 'Member' (selected and highlighted with a red border), and 'Admin' (unselected). A link 'iOS Developer Program Roles Overview' is visible. At the bottom are 'Cancel' and 'Save' buttons, with 'Save' being the active one.

iOS Developer Program	Agent	Admin	Member
Download Development Provisioning Profiles	✓	✓	✓
Submit Certificate Signing Request	✓	✓	✓
Approve Certificate Signing Request	✓	✓	
Create Development Provisioning Profiles	✓	✓	
Assign iOS Device IDs and Register App IDs	✓	✓	
Create an iOS Distribution Certificate and Distribution Provisioning Profiles	✓	✓	
Enable App IDs and create SSL Certificates for Apple Push Notification service	✓	✓	
Submit Apps and App Updates to iTunes Connect	✓		

Agent = Super Admin

# Provisioning

The image shows two screenshots of the Apple Developer website. The top screenshot is the 'iOS Dev Center' page, featuring a navigation bar with links for Technologies, Resources, Programs, Support, Member Center, and a search bar. Below the navigation is a main content area with sections for 'Resources for iOS 4.3' (Downloads, Getting Started Videos), 'Featured Content' (New Subscription Service for iOS Apps, Getting Ready for iOS 4.3), and a sidebar for the 'iOS Developer Program' (links to the iOS Provisioning Portal, iTunes Connect, and Apple Developer Forums). An arrow points from the 'iOS Provisioning Portal' link in the sidebar down to the second screenshot. The bottom screenshot is the 'iOS Provisioning Portal' itself, showing a 'Welcome to the iOS Provisioning Portal' message and a sidebar with links for Home, Certificates (which is highlighted with a red box), Devices, App IDs, Provisioning, Distribution, and Portal Resources (Program User Guide, Program User Guide (JP)). The main content area includes a 'Visit the Member Center' callout and a section about the Development Provisioning Assistant.

**iOS Dev Center**

Hi, Christopher Judd | My Profile | Log out

**iOS Developer Program**

- iOS Provisioning Portal
- iTunes Connect
- Apple Developer Forums

**iOS Provisioning Portal**

Welcome, Christopher Judd | Edit Profile | Log out

**Provisioning Portal : Judd Solutions, LLC**

Go to iOS Dev Center

**Home**

Certificates (highlighted with a red box)

Devices

App IDs

Provisioning

Distribution

**Welcome to the iOS Provisioning Portal**

The iOS Provisioning Portal is designed to take you through the necessary steps to test your applications on iOS devices and prepare them for distribution.

**Visit the Member Center for Team, Account, and Program info**

The new Member Center is now your destination for:

- Sending invitations to join your development team and editing existing development team members.
- Requesting or purchasing Technical Support.
- Viewing account information, such as your Team ID, profile, and Program details.

[Visit the Member Center now](#)

**Get your application on an iOS with the Development Provisioning Assistant**

As a Program Admin, you can use the Development Provisioning Assistant to create and install a Provisioning Profile and iOS Development Certificate needed to build and install applications you're developing for iOS devices.

[Launch Assistant](#)

**Portal Resources**

- Program User Guide
- Program User Guide (JP)

**How-To's**

- Obtaining your Certificate
- Assigning Devices
- Creating your App IDs
- Creating Provisioning Profiles

**Support Resources**

**iTunes Connect Support**

Find answers to questions about the iTunes Connect system, including uploading your application or finding a Finance Report.

**Technical Support**

Receive code-level, technical assistance for your specific development issue.

**Developer Support**

Contact us for general inquiries, including Program questions, account issues and change of contact information.

# Development Certificates

Provisioning Portal : Judd Solutions, LLC [Go to iOS Dev Center](#)

Home Certificates **Development** Distribution History How To

### Current Development Certificates

 Your Certificate

Name	Provisioning Profiles	Expiration Date	Status	Action
Christopher Judd	Team Provisioning Profile: *	Jan 03, 2012	Issued	<a href="#">Download</a> <a href="#">Revoke</a>

\*If you do not have the WWDR intermediate certificate installed, [click here to download now.](#)

 Team Signing Requests (0)

Signing Requests	Active Certificates		
Name	Email	Status	Date Submitted

must create developer certificate

# Devices

Provisioning Portal : Judd Solutions, LLC [Go to iOS Dev Center](#)

Home Certificates **Devices** App IDs Provisioning Distribution

Manage History How To

**Current Registered Devices** [Upload Devices](#) [Add Devices](#)

 **Important:** Your iOS Developer Program membership can be terminated if you provide pre-release Apple Software to anyone other than employees, contractors, and members of your organization who are registered as Apple Developers and have a demonstrable need to know or use Apple Software in order to develop and test applications on your behalf. Unauthorized distribution of Apple Confidential Information (including pre-release Apple Software) is prohibited and may subject you to both civil and criminal liability.

You can register 91 additional devices. You can register up to 100 devices per year for development purposes. Any devices added, then later removed, still count towards your maximum number of registered devices per year. [Learn more](#)

<input type="checkbox"/>	Device Name	Device ID	Profiles	Actions
<input type="checkbox"/>	Alexander Moore beta t...	4d97cf1acf5bb055bf7ce0d5f7...	2 ( <a href="#">Details</a> )	<a href="#">Edit</a>
<input type="checkbox"/>	Chris Judd's iPhone	9ee3f4f787a982ef9e7bee8845...	1 ( <a href="#">Details</a> )	<a href="#">Edit</a>
<input type="checkbox"/>	Christopher Judd iPad	e89e29471548cb58ea0214d396...	2 ( <a href="#">Details</a> )	<a href="#">Edit</a>
<input type="checkbox"/>	Christopher Judd iPhone	dfc95646fb25972cdfeb78e184...	2 ( <a href="#">Details</a> )	<a href="#">Edit</a>
<input type="checkbox"/>	Christopher Judds iPod	872afaccc3ac76ac40ab3b4d07...	2 ( <a href="#">Details</a> )	<a href="#">Edit</a>
<input type="checkbox"/>	Jim Shingler iPod Touc...	ce51689cd0e6b779db1ec9c083...	1 ( <a href="#">Details</a> )	<a href="#">Edit</a>
<input type="checkbox"/>	Rick Cherry beta teste...	4a6a9872be9dc17bc701157344...	2 ( <a href="#">Details</a> )	<a href="#">Edit</a>
<input type="checkbox"/>	Robert Stevenson beta ...	b9a73667b580fae2416ee92d51...	2 ( <a href="#">Details</a> )	<a href="#">Edit</a>
<input type="checkbox"/>	Ron McCamish beta test...	f75a817506ec118c52cf5d6a1d...	2 ( <a href="#">Details</a> )	<a href="#">Edit</a>

[Remove Selected](#)

# Development Provisioning Profiles

Provisioning Portal : Judd Solutions, LLC [Go to iOS Dev Center](#)

Home Certificates Devices App IDs **Provisioning** Distribution

**Development** Distribution History How To

## Development Provisioning Profiles

New Profile

<input type="checkbox"/> Provisioning Profile	App ID	Status	Actions
<input type="checkbox"/> Chmod	CKWT37BFD5.com.juddsolutions.c...	Expired	<a href="#">Renew</a>
<input type="checkbox"/> Generic Judd Solutions Develop...	5FB78XGVZL.com.juddsolutions.*	Expired	<a href="#">Renew</a>
<input type="checkbox"/> Team Provisioning Profile: *	NR99UM579H.*	Invalid • Managed by Xcode	<a href="#">Renew</a>

[Remove Selected](#)

Organizer - Devices

Devices Repositories Projects Archives Documentation

LIBRARY

- Developer Profile
- Provisioning Profiles**
- Software Images
- Device Logs
- Screenshots

DEVICES

- My Mac 10.6.8 (10K549)
- Chris Judd's iPod 3.1.3 (7E18)
  - Device Logs
  - Screenshots
- Chris Judd's iPhone 5.0.1 (9A405)
  - Device Logs
  - Screenshots
- Chris Judd's iPad 4.3.5 (8L1)
  - Device Logs
  - Screenshots

**Team Provisioning Profile: \***

Creation Date Tuesday, November 22, 2011 2:35:20 PM ET  
Expiration Date Wednesday, November 21, 2012 2:35:20 PM ET  
Profile Identifier A9B12454-E115-4C2E-8903-85DA4D7097EF  
App Identifier NR99UM579H.\*  
Devices Chris Judd's iPhone, Chris Judd's iPad, Chris Judd's iPod, and 9 other devices

Profile Name

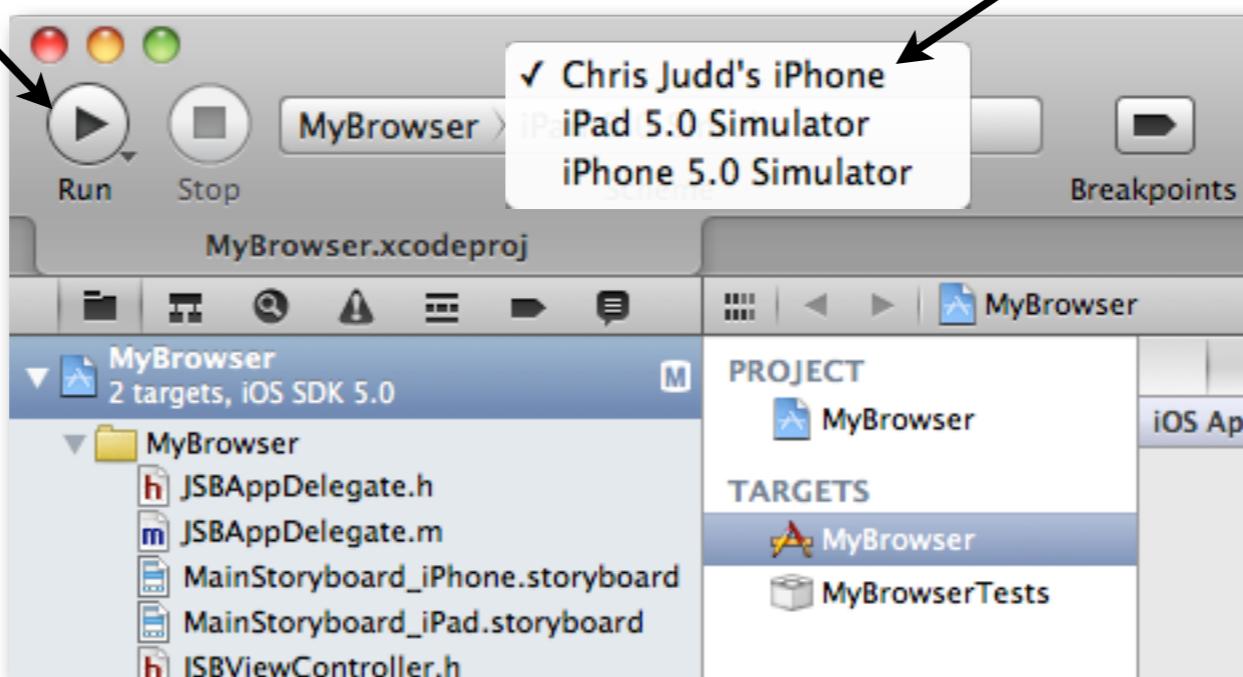
Name	Expiration Date	Portal Team	Status
Tabernacle Ad Hoc	December 20, 2011 7:...	Unknown	
Tabernacle App Store	September 23, 2012 4...	Unknown	
Team Provisioning Profile: *	December 20, 2011 7:...	Unknown	
Team Provisioning Profile: *	November 21, 2012 2:...	Judd Solutions, LLC	
TheTabernacle	December 20, 2011 7:...	Unknown	
iOS Team Provisioning Profile: *	November 22, 2012 4:...	Shoutback Concepts, LLC	
iOS Team Provisioning Profile: *	November 22, 2012 4:...	eTech Ohio	
iOS Team Provisioning Profile: *	November 22, 2012 4:...	Mobile Security Worldwide Ltd	

New Import Export  Automatic Device Provisioning Refresh

download profiles

# Running

Run



device

or  
⌘ R

# *Lab 4*

- I. If you have an iOS device with you, deploy and run the application on your device.

# **INSTRUMENTS**

# **AD HOC DEPLOYMENT**

# Provisioning

The image shows two screenshots of the Apple Developer website. The top screenshot is the 'iOS Dev Center' page, featuring a navigation bar with links for Technologies, Resources, Programs, Support, Member Center, and a search bar. Below the navigation is a main content area with sections for 'Resources for iOS 4.3' (Downloads, Getting Started Videos), 'Featured Content' (New Subscription Service for iOS Apps, Getting Ready for iOS 4.3), and the 'iOS Developer Program' (links to the iOS Provisioning Portal, iTunes Connect, and Apple Developer Forums). An arrow points from the 'iOS Provisioning Portal' link in the top navigation to the second screenshot. The bottom screenshot is the 'iOS Provisioning Portal' itself, titled 'iOS Provisioning Portal'. It has a sidebar with links for Home, Certificates (which is highlighted with a red box), Devices, App IDs, Provisioning, Distribution, and a 'Go to iOS Dev Center' button. The main content area features a 'Welcome to the iOS Provisioning Portal' message, a 'Visit the Member Center for Team, Account, and Program info' section with a list of steps, and a 'Get your application on an iOS with the Development Provisioning Assistant' section with a 'Launch Assistant' button. To the right of the main content is a 'Portal Resources' sidebar with links to Program User Guide and Program User Guide (JP), and a 'How-To's' sidebar with links to Obtaining your Certificate, Assigning Devices, Creating your App IDs, and Creating Provisioning Profiles, each with a 'View video' option. The bottom right corner of the portal screenshot contains 'Support Resources' links for iTunes Connect Support, Technical Support, and Developer Support.

Developer

iOS Dev Center

Hi, Christopher Judd | My Profile | Log out

iOS SDK 4.3 iOS SDK beta

Resources for iOS 4.3

Downloads

Getting Started Videos

Featured Content

New Subscription Service for iOS Apps

Getting Ready for iOS 4.3

iOS Developer Program

iOS Provisioning Portal

iTunes Connect

Apple Developer Forums

Developer

iOS Provisioning Portal

Welcome, Christopher Judd | Edit Profile | Log out

Provisioning Portal : Judd Solutions, LLC

Home

Certificates

Devices

App IDs

Provisioning

Distribution

Go to iOS Dev Center

Welcome to the iOS Provisioning Portal

The iOS Provisioning Portal is designed to take you through the necessary steps to test your applications on iOS devices and prepare them for distribution.

Visit the Member Center for Team, Account, and Program info

The new Member Center is now your destination for:

- Sending invitations to join your development team and editing existing development team members.
- Requesting or purchasing Technical Support.
- Viewing account information, such as your Team ID, profile, and Program details.

Visit the Member Center now

Get your application on an iOS with the Development Provisioning Assistant

As a Program Admin, you can use the Development Provisioning Assistant to create and install a Provisioning Profile and iOS Development Certificate needed to build and install applications you're developing for iOS devices.

Launch Assistant

Portal Resources

Program User Guide

Program User Guide (JP)

How-To's

Obtaining your Certificate

Assigning Devices

Creating your App IDs

Creating Provisioning Profiles

Support Resources

iTunes Connect Support

Technical Support

Developer Support

# App IDs

Developer Technologies Resources Programs Support Member Center Search Developer

iOS Provisioning Portal Welcome, Christopher Judd | Edit Profile | Log out

Provisioning Portal : JUDD SOLUTIONS, LLC Go to iOS Dev Center

Home Certificates Devices App IDs Provisioning Distribution

Manage How To ID App IDs New App ID

App IDs are an integral part of the iOS Development and Provisioning Process that allows an application to communicate with the Apple Push Notification service and/or an external hardware accessory. In addition, an App ID can also be used to share keychain data (such as passwords) between a suite of applications and share document and configuration data between your applications using iCloud.

Description	Apple Push Notification service	In App Purchase	Game Center	iCloud	Action
5FB78XGVZL.com.juddsoluti... Generic Test	Unavailable	Unavailable	Unavailable	Enabled	<a href="#">Details</a>
BD5B4V25LM.com.juddsoluti... Checkbox	Unavailable	Unavailable	Unavailable	Enabled	<a href="#">Details</a>
C2C5H32X45.com.juddsoluti... Test Deployment	Configurable for Development Configurable for Production	Enabled	Enabled	Enabled	<a href="#">Configure</a>
CKWT37BFDS.com.juddsoluti... Chmod	Unavailable	Unavailable	Unavailable	Enabled	<a href="#">Details</a>
ENH7TL5476.com.juddsoluti... Hello World	Configurable for Development Configurable for Production	Enabled	Enabled	Enabled	<a href="#">Configure</a>
K66KCKN979.com.juddsoluti... UICatalog	Unavailable	Unavailable	Unavailable	Enabled	<a href="#">Details</a>
MW4B7WE5L6.com.juddsoluti... Chmon	Unavailable	Unavailable	Unavailable	Enabled	<a href="#">Details</a>
NR99UM579H.* vibman	Unavailable	Unavailable	Unavailable	Enabled	<a href="#">Details</a>

# Ad Hoc Store Provisioning Profile

include Ad Hoc in name

The image shows two screenshots of the Apple Provisioning Portal. The top screenshot displays the 'Distribution Provisioning Profiles' page, which lists three profiles: 'Provisioning Profile' (status: Not Installed), 'Chmod Ad Hoc Provisioning Prof...' (App ID: CKWT37BFDS.com.juddsolutions.c..., Status: Expired), and 'Chmod App Store Provisioning P...' (App ID: CKWT37BFDS.com.juddsolutions.c..., Status: Expired). An arrow points from the text 'include Ad Hoc in name' to the second profile. The bottom screenshot shows the 'Create iOS Distribution Provisioning Profile' form. It includes fields for 'Profile Name' (set to 'EveryNote Ad Hoc'), 'Distribution Certificate' (selected certificate: 'Judd Solutions, LLC (expiring on Jan 3, 2012)'), 'App ID' (button: 'Select an App ID'), and 'Devices (optional)' (checkboxes for selecting devices: 'Select All' followed by a list of devices including 'Alexander Moore beta tester', 'Christopher Judd iPad', 'Christopher Judds iPod', 'Rick Cherry beta tester', 'Ron McCamish beta tester', 'Chris Judd's iPhone', 'Christopher Judd iPhone', 'Jim Shingler iPod Touch', and 'Robert Stevenson beta tester').

Provisioning Portal : Judd Solutions, LLC

Home Certificates Devices App IDs Provisioning Distribution

Development Distribution History How To

Distribution Provisioning Profiles

New Profile

Provisioning Profile CKWT37BFDS.com.juddsolutions.c... Status Actions

Chmod Ad Hoc Provisioning Prof... CKWT37BFDS.com.juddsolutions.c... Expired Modify

Chmod App Store Provisioning P... CKWT37BFDS.com.juddsolutions.c... Expired Modify

Provisioning Portal : Judd Solutions, LLC

Home Certificates Devices App IDs Provisioning Distribution History How To

Create iOS Distribution Provisioning Profile

Generate provisioning profiles here. All fields are required unless otherwise noted. To learn more, visit the [How To](#) section.

**Distribution Method**  App Store  Ad Hoc

**Profile Name** EveryNote Ad Hoc

**Distribution Certificate** Judd Solutions, LLC (expiring on Jan 3, 2012)

**App ID** Select an App ID

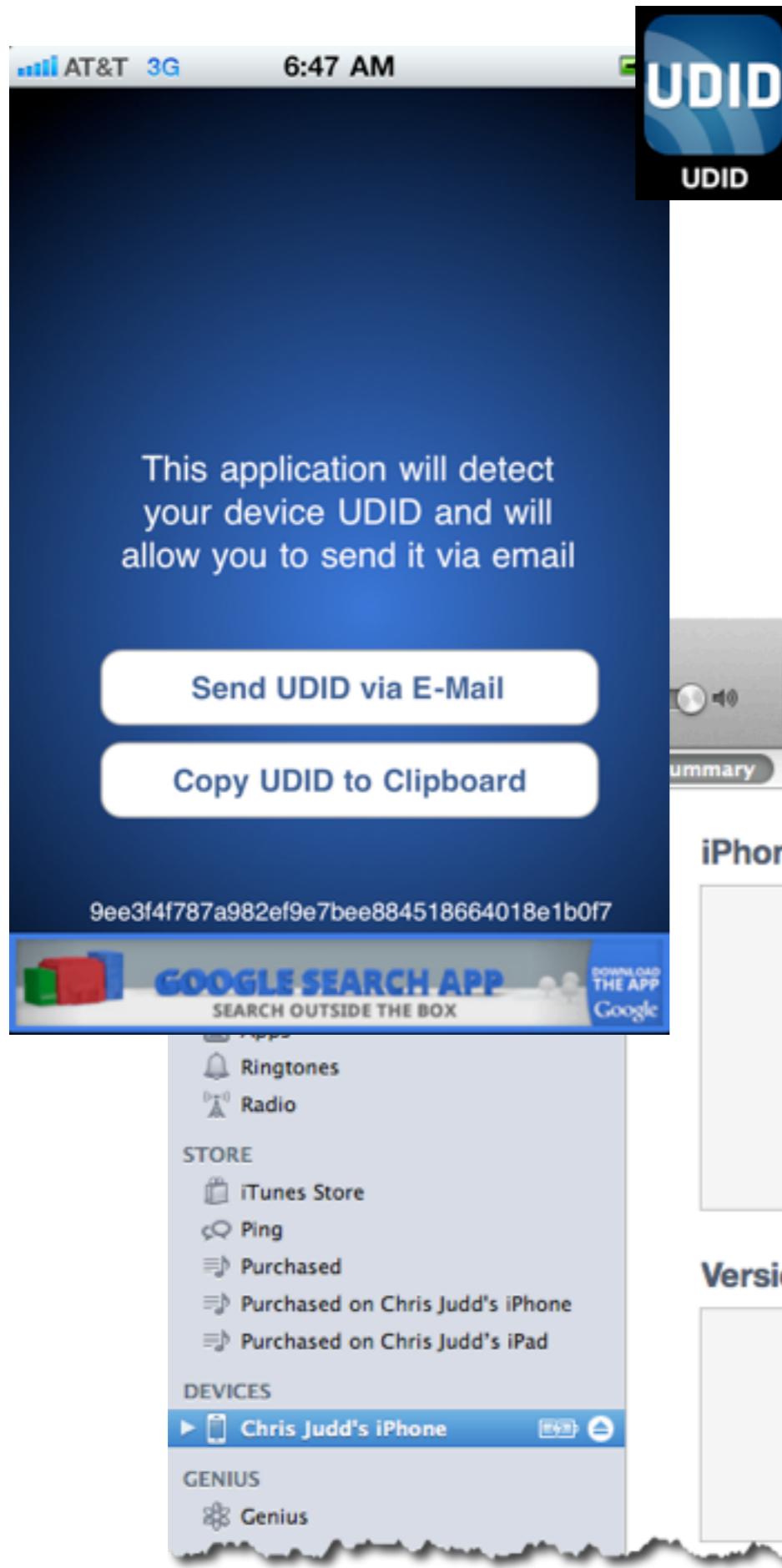
**Devices (optional)** Select up to 100 devices for distributing the final application; the final application will run only on these selected devices.

Select All

Alexander Moore beta tester  Chris Judd's iPhone  
 Christopher Judd iPad  Christopher Judd iPhone  
 Christopher Judds iPod  Jim Shingler iPod Touch  
 Rick Cherry beta tester  Robert Stevenson beta tester  
 Ron McCamish beta tester

Cancel Submit

# Devices



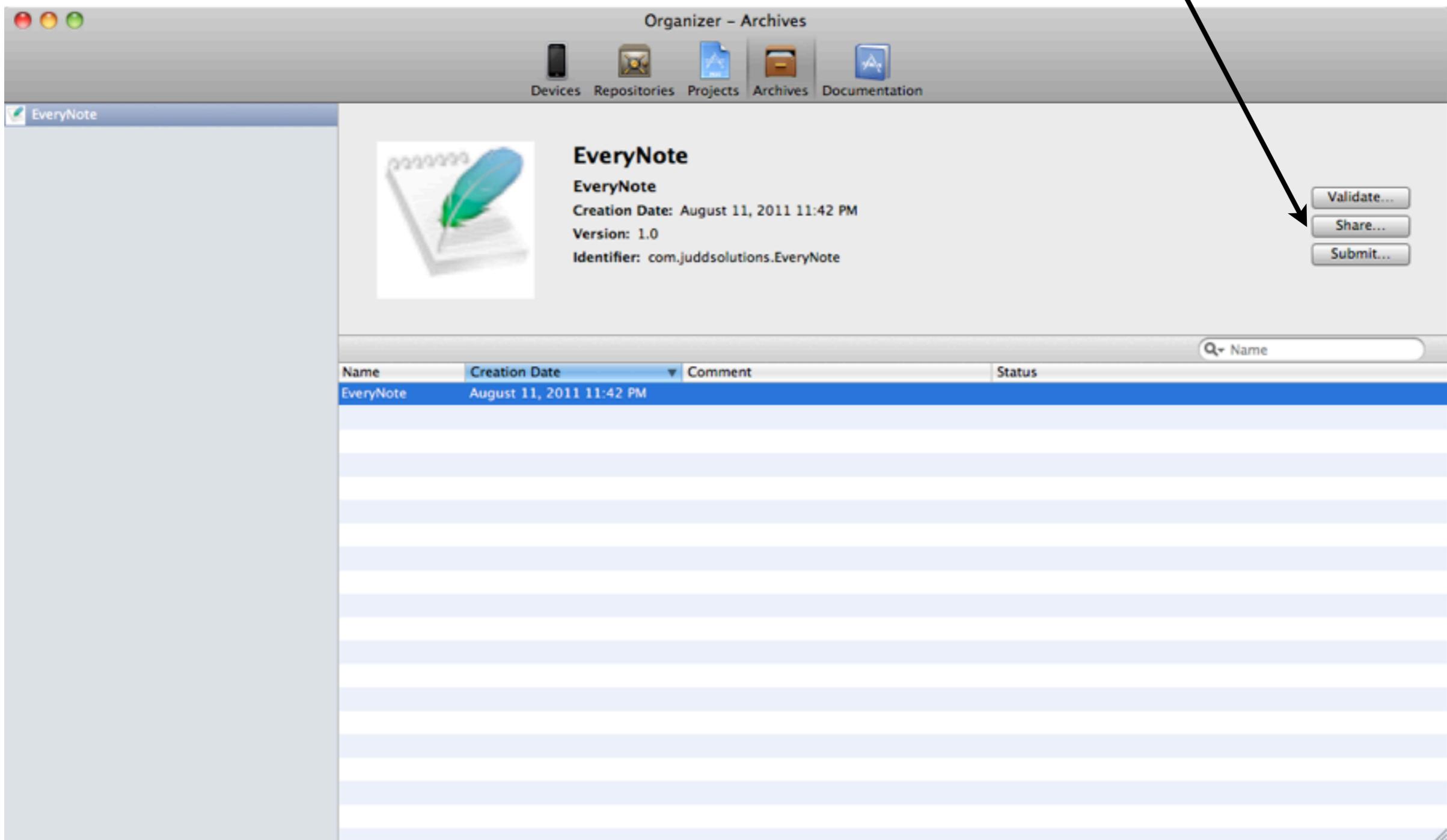
⌘ + click  
then  
Edit > Copy

A screenshot of the iTunes interface. The top menu bar shows "iTunes". The main window displays an iPhone. The device summary on the right side of the screen includes the following information:  
**Name:** Chris Judd's iPhone  
**Capacity:** 29.06 GB  
**Software Version:** 4.3.5  
**Identifier (UDID):** 9ee3f4f787a982ef9e7bee884518664018e1b0f7  
**Phone Number:** 1 (614) 378-4119

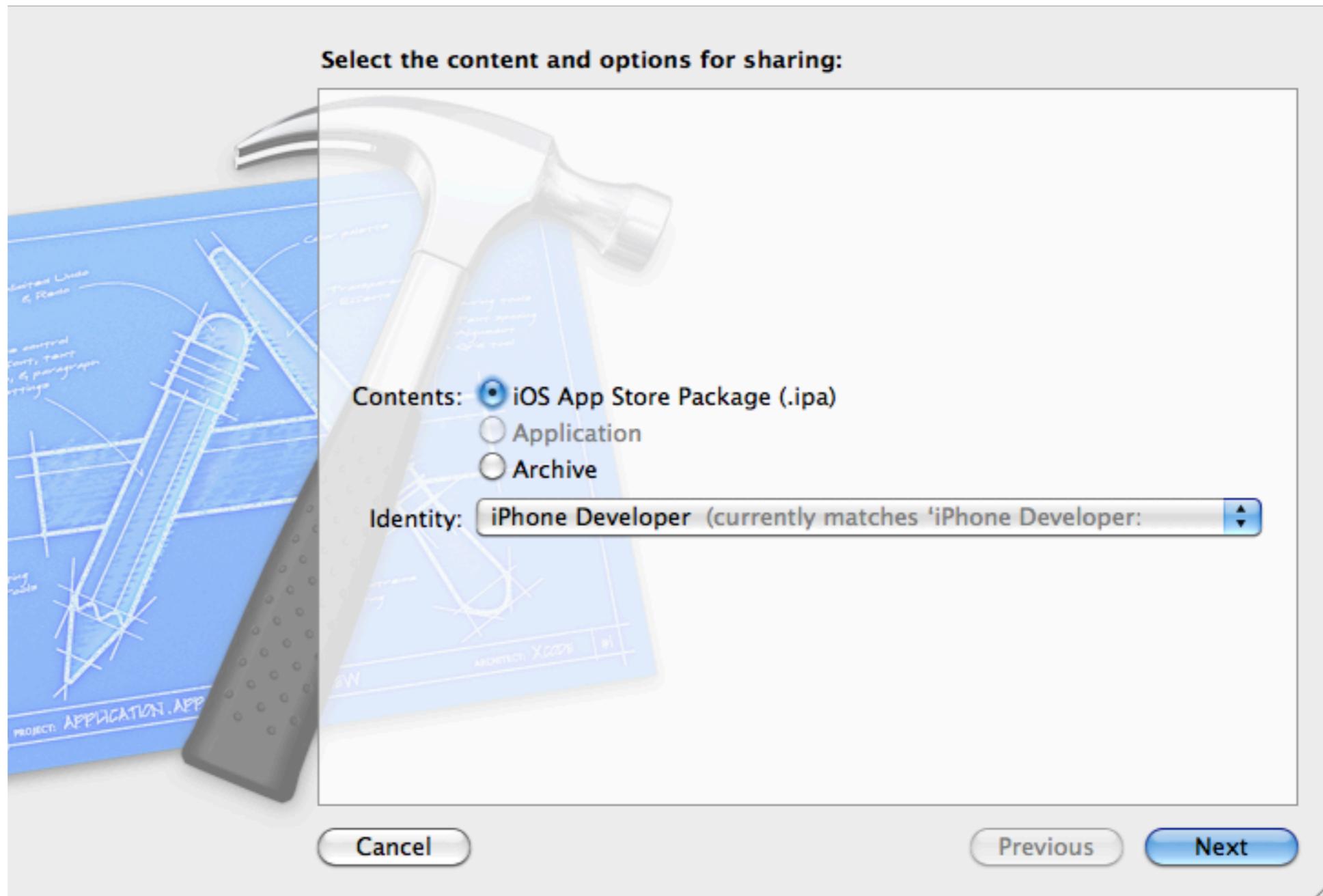
Below the device summary, under the "Version" section, there is a note: "Your iPhone software is up to date. iTunes will automatically check for an update again on 8/14/11." There are also "Check for Update" and "Restore" buttons.

# Archive

## Product > Archive



# Email to Beta Testers



**TestFlight \* Manage The Builds**

<https://testflightapp.com/dashboard/builds/>

Christopher Judd

Install Apps Builds Team Support

**Manage The Builds**

**Tabernacle 1.0.4**  
Released: October 6, 2011 Built For: iPad iOS 3.2+ Size: 108.4 MB

1 INSTALLED 1 NOT INSTALLED

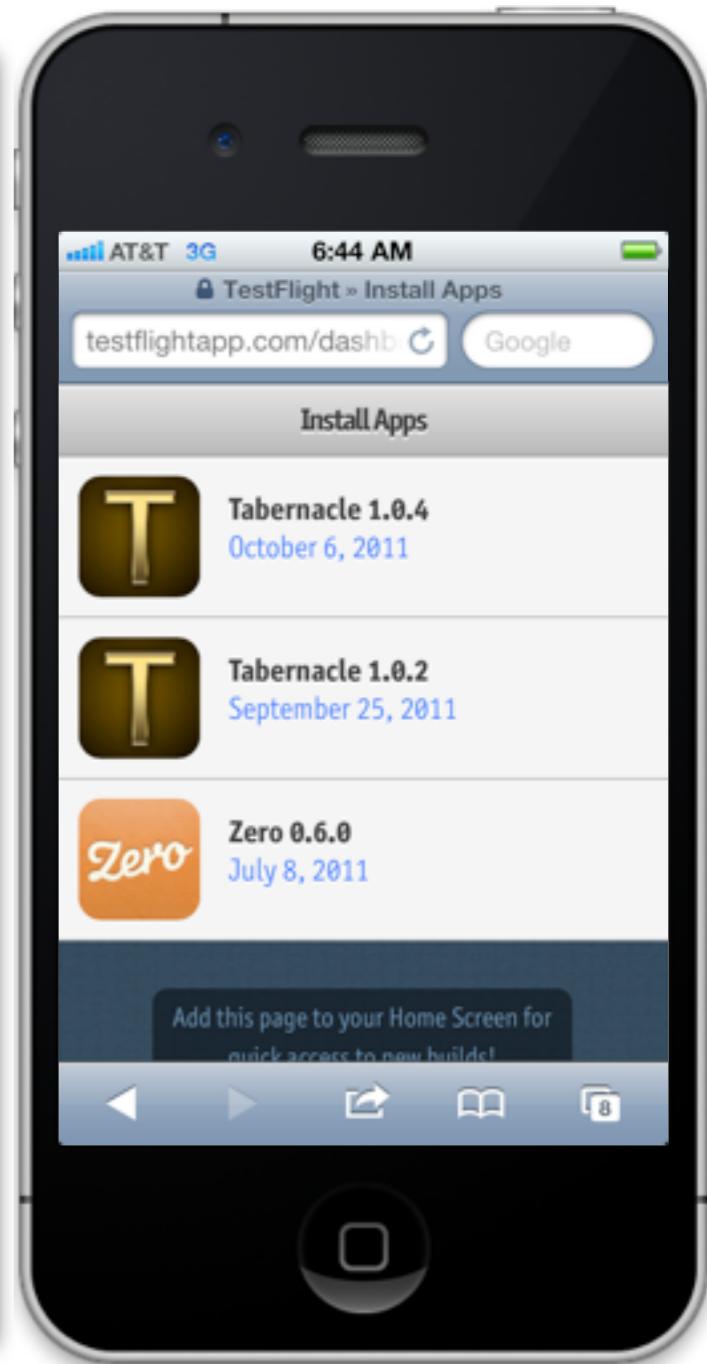
Testers Edit or Remove Reports Feedback History Sessions Crashes Checkpoints Questions

**Tabernacle 1.0.2**  
Released: September 25, 2011 Built For: iPad iOS 3.2+ Size: 108.4 MB

1 INSTALLED 1 NOT INSTALLED

Testers Edit or Remove Reports Feedback History Sessions Crashes Checkpoints Questions

Download the SDK Upload API Terms of Use Privacy Policy Blog About Contact Help & Support



<https://testflightapp.com>

# **APP STORE DEPLOYMENT**

# Provisioning

The image shows two screenshots of the Apple Developer website. The top screenshot is the 'iOS Dev Center' page, featuring a navigation bar with links for Technologies, Resources, Programs, Support, Member Center, and a search bar. Below the navigation is a main content area with sections for 'Resources for iOS 4.3' (Downloads, Getting Started Videos), 'Featured Content' (New Subscription Service for iOS Apps, Getting Ready for iOS 4.3), and a sidebar for the 'iOS Developer Program' (links to the iOS Provisioning Portal, iTunes Connect, and Apple Developer Forums). An arrow points from the 'iOS Provisioning Portal' link in the sidebar down to the second screenshot. The bottom screenshot is the 'iOS Provisioning Portal' itself, showing a 'Welcome to the iOS Provisioning Portal' message and a sidebar with links for Home, Certificates (which is highlighted with a red box), Devices, App IDs, Provisioning, Distribution, and Portal Resources (Program User Guide, Program User Guide (JP)). The main content area includes a 'Visit the Member Center' callout, a 'Development Provisioning Assistant' section with a 'Launch Assistant' button, and support resources for iTunes Connect, Technical, and Developer Support.

Developer

iOS Dev Center

Technologies Resources Programs Support Member Center Search Developer

Hi, Christopher Judd | My Profile | Log out

iOS SDK 4.3 iOS SDK beta

Resources for iOS 4.3

Downloads

Getting Started Videos

Featured Content

New Subscription Service for iOS Apps

Getting Ready for iOS 4.3

iOS Developer Program

iOS Provisioning Portal

iTunes Connect

Apple Developer Forums

Developer

iOS Provisioning Portal

Welcome, Christopher Judd | Edit Profile | Log out

Go to iOS Dev Center

Provisioning Portal : Judd Solutions, LLC

Home

Certificates

Devices

App IDs

Provisioning

Distribution

Welcome to the iOS Provisioning Portal

The iOS Provisioning Portal is designed to take you through the necessary steps to test your applications on iOS devices and prepare them for distribution.

Visit the Member Center for Team, Account, and Program info

The new Member Center is now your destination for:

- Sending invitations to join your development team and editing existing development team members.
- Requesting or purchasing Technical Support.
- Viewing account information, such as your Team ID, profile, and Program details.

Visit the Member Center now

Get your application on an iOS with the Development Provisioning Assistant

As a Program Admin, you can use the Development Provisioning Assistant to create and install a Provisioning Profile and iOS Development Certificate needed to build and install applications you're developing for iOS devices.

Launch Assistant

Portal Resources

Program User Guide

Program User Guide (JP)

How-To's

Obtaining your Certificate

View video

Assigning Devices

View video

Creating your App IDs

View video

Creating Provisioning Profiles

View video

Support Resources

iTunes Connect Support

Find answers to questions about the iTunes Connect system, including uploading your application or finding a Finance Report.

Technical Support

Receive code-level, technical assistance for your specific development issue.

Developer Support

Contact us for general inquiries, including Program questions, account issues and change of contact information.

# Distribution Certificates

Provisioning Portal : Judd Solutions, LLC [Go to iOS Dev Center](#)

Home **Certificates** Development Distribution History How To

Devices  
App IDs  
Provisioning  
Distribution

**Current Distribution Certificate**

Name	Expiration Date	Provisioning Profiles	Status	Actions
<a href="#">Judd Solutions, LLC</a>	Jan 3, 2012		Issued	<a href="#">Download</a> <a href="#">Revoke</a>

\*If you do not have the WWDR intermediate certificate installed, [click here to download now.](#)

# App Store Provisioning Profile

include App Store in name

The screenshot shows two views of the Apple Provisioning Portal for Judd Solutions, LLC.

**Distribution Provisioning Profiles View:** This view lists existing provisioning profiles. An arrow points from the "Chmod App Store Provisioning Prof..." entry in the list to the "Profile Name" field in the "Create iOS Distribution Provisioning Profile" view below.

Provisioning Profile	App ID	Status	Actions
Chmod Ad Hoc Provisioning Prof...	CKWT378FD5.com.juddsolutions.c...	Expired	Modify
Chmod App Store Provisioning P...	CKWT378FD5.com.juddsolutions.c...	Expired	Modify

**Create iOS Distribution Provisioning Profile View:** This view allows generating a new provisioning profile. The "Distribution Method" is set to "App Store". The "Profile Name" is "EveryNote App Store". The "Distribution Certificate" is "Judd Solutions, LLC (expiring on Jan 3, 2012)". The "App ID" dropdown is set to "Select an App ID". Under "Devices (optional)", the "Select All" checkbox is checked, and several device checkboxes are listed: Alexander Moore beta tester, Christopher Judd iPad, Christopher Judds iPod, Rick Cherry beta tester, Ron McCamish beta tester, Chris Judd's iPhone, Christopher Judd iPhone, Jim Shingler iPod Touch, and Robert Stevenson beta tester.

# Submit to App Store

The screenshot shows the Apple Developer website's iOS Dev Center. At the top right, there is a dropdown menu titled "iOS Developer Program" with options for "iOS Provisioning Portal", "iTunes Connect", "Apple Developer Forums", and "Developer Support Center". An arrow points from the text "Click here to log in" to the "iTunes Connect" option. Below this, the main content area displays the "iTunes Connect" landing page, which includes sections for "Sales and Trends", "Contracts, Tax, and Banking", "Payments and Financial Reports", and "Manage Users". There is also a section for "Manage Your Applications" with a link to "Add, view, and manage your applications in the iTunes Store".

**iOS Developer Program**

- iOS Provisioning Portal
- iTunes Connect**
- Apple Developer Forums
- Developer Support Center

Welcome, Judd Solutions, LLC

iTunes Connect provides tools to help manage your content in the App Store.

Are you looking for opportunities to generate additional revenue and market your apps to millions of iPhone and iPod touch users around the world? Apple now offers two choices to help promote your apps and earn extra revenue through advertising. iAd Network allows you to create media-rich ads and earn 60 percent of advertising revenue generated by iAds served to your app. iAd for Developers gives you a unique, cost-effective way to promote your iOS 4 apps to millions of iPhone and iPod touch users. Learn more about the [iAd Network](#) and [iAd for Developers](#) and take advantage of these programs today.

Version 1.2 of the [iTunes Connect Mobile app](#) is now available for download on the App Store. iTunes Connect Mobile 1.2 allows you to receive push notifications related to the status of your apps.

iAd is a new mobile advertising platform that combines the emotion of TV ads with the interactivity of the web. For an opportunity to earn advertising revenue, join the iAd Network and enable ads in your applications. [Learn more](#).

To use iAd in your applications, your Team Agent must agree to the [iAd Network Contract](#). Note that you must agree to the latest version of the [iPhone Developer Program Agreement](#) before you can access the iAd Network Contract.

**Sales and Trends**  
Preview or download your daily and weekly sales information here.

**Contracts, Tax, and Banking**  
Manage your contracts, tax, and banking information.

**Payments and Financial Reports**  
View and download your monthly financial reports and payments.

**Manage Users**  
Set access permissions and email settings for users of your iTunes Connect account.

**Manage Your Applications**  
Add, view, and manage your applications in the iTunes Store.

**Contact Us**  
Having a problem uploading your application? Can't find a Finance Report? Use our Contact Us system to find an answer to your question or to generate a question to an iTunes Rep.

[Download the Developer Guide.](#) [FAQs](#) Review our answers to common inquiries.

**iTunes Connect Mobile**  
Access your sales and trend information anywhere. Get it free from the App Store. [Download](#)



Christopher Judd, Judd Solutions, LLC [Sign Out](#)

[Add New App](#)

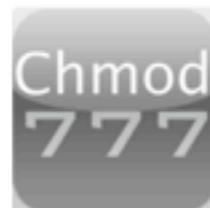
## Manage Your Apps

### Recent Activity

iOS App Recent Activity

[See All](#)

1 Total



Chmod 1.0

### Search

Name :  starts with

Apple ID :

Status :

[Search](#)

### Developer Guides

[English](#)

[Japanese](#)

[Chinese \(Simplified\)](#)

### App Loader Guide

[English](#)

[Go Back](#)

[Home](#) | [FAQs](#) | [Contact Us](#) | [Sign Out](#)  
Copyright © 2011 Apple Inc. All rights reserved. [Terms of Service](#) | [Privacy Policy](#)

## App Information

Enter the following in English.

App Name



SKU Number



Bundle ID



You can register a new Bundle ID [here](#).



**Make sure this is the correct Bundle ID for your app. The Bundle ID cannot be changed once it is saved.**

Does your app have specific device requirements? [Learn more](#)

[Cancel](#)[Continue](#)

[Home](#) | [FAQs](#) | [Contact Us](#) | [Sign Out](#)

Copyright © 2011 Apple Inc. All rights reserved. [Terms of Service](#) | [Privacy Policy](#)

## EveryNote

Select the availability date and price tier for your app.

Availability Date

08/Aug 11 2011



Price Tier

Free



[View Pricing Matrix ▶](#)

Discount for Educational Institutions



Unless you select **specific stores**, your app will be for sale in all App Stores worldwide.

[Go Back](#)

[Continue](#)

[Home](#) | [FAQs](#) | [Contact Us](#) | [Sign Out](#)

Copyright © 2011 Apple Inc. All rights reserved. [Terms of Service](#) | [Privacy Policy](#)

## EveryNote

Enter the following information in English.

### Metadata

Version Number



Description



Primary Category



Secondary Category (optional)



Keywords



Copyright



Contact Email Address



Support URL



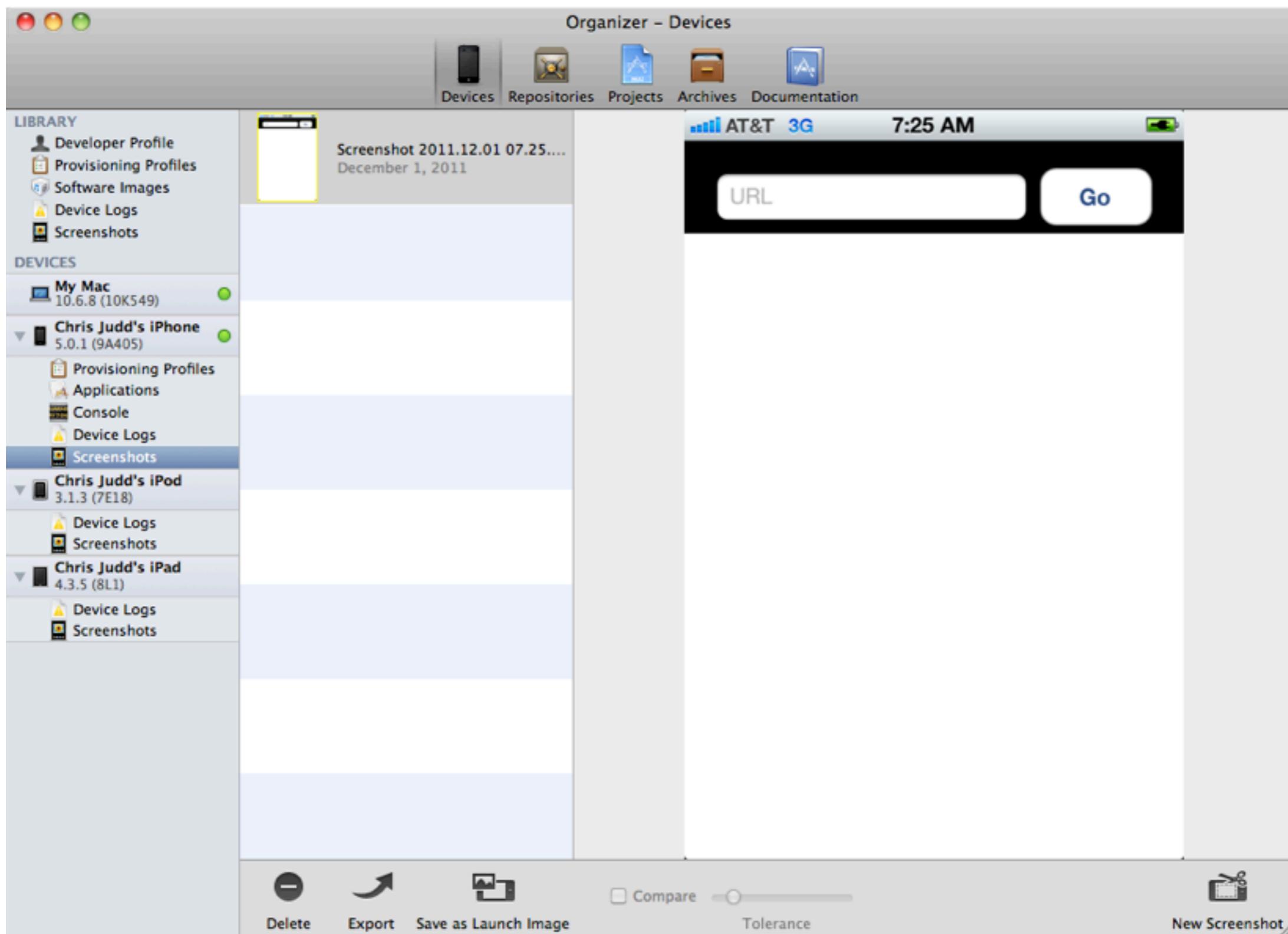
App URL (optional)



Review Notes (optional)



# Screenshots



# Screenshots



**Press and Hold**

# Screenshots

## iOS-Simulator Cropper

Easy way to take screenshots from the iOS-Simulator

I „hacked“ this tool to take screenshots of my iPhone and iPad Apps running in the iOS-Simulator. Before that, i took a screenshot (cmd+shift+4) and cropped the application screen with the help of Pixelmator. Especially when you need screenshots in different languages this takes a lot of time (version by version of your App).

Another point to mention, you should upload screenshots in iTunes Connect (App Store) without the status bar on top! None of your potential customers is interested in the provider you are using, what time you have taken the screenshots, or how your battery level was.

**Usage**

It's very easy to use. Just start your App in the iOS-Simulator and press the „Take Simulator Screenshot“-button. Hint: The iOS-Simulator and the iOS-Simulator Cropper must be running on the same Space. Additionally you can drag&drop screenshots to the small drop zone.

**iOS-Simulator Cropper**

Purpose of result image: iTunes Connect / Apple Store (Status Bar removed)

Save result image to:

- Directory: /Users/of/Desktop
- Select
- Filename: gamecenter\_en\_%COUNTER%
- (without Ext.)
- Counter: 1
- Reset
- (Filename keyword %COUNTER%)

Additionally:

Copy to Clipboard

Drop iOS Screenshots  or **Take Simulator Screenshot**

**iPhone Screenshots**

iOS-Simulator Screenshot: A screenshot of an iPhone displaying the Game Center welcome screen.

Device from teehan+lax: A screenshot of an iPhone displaying the Game Center welcome screen.

iTunes Connect / App Store: A screenshot of the iTunes Connect App Store interface.

**home**

**contact & imprint**

**kontakt & impressum**

 **Download**

**Version 2.7.4**  
Nov. 24, 2011  
- black iPhone 4S  
- white iPhone 4S

**Version 2.7.3**  
Sept. 7, 2011  
- Copy to Clipboard added

**Version 2.7.2**  
Sept. 7, 2011  
- Check for Updates added  
- Missing menu items added

Do you find this free application useful? Support it's development by flattr!

15 

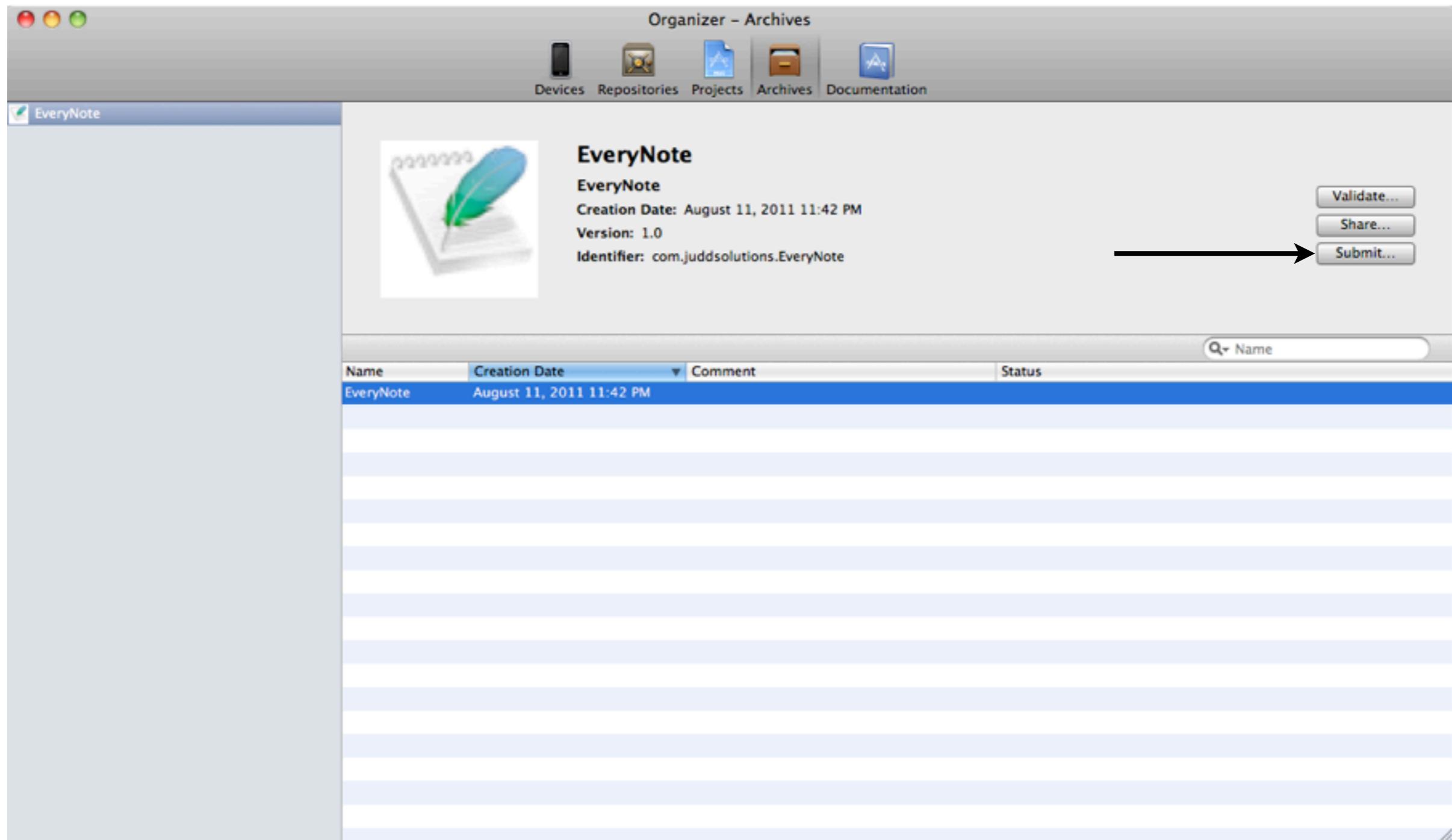
 27usethis

**::: REVIEW'S :::**  
Marin Todorov published a fantastic review of the Cropper on his **Touch Code Magazin**. (08/03/2010)

John Muchow from the great **[iPhone developer:tips]** blog wrote a nice review of the iOS-Simulator Cropper. (07/28/2010)

<http://www.curioustimes.de/iphonesimulatorkropper/>

**status is now waiting to submit**





wait for approval

iTunes

App Store > Education > AnimMan Studios

The Tabernacle

Description

AnimMan Studios

...  
...More

AnimMan Studios Web Site > The Tabernacle Support >

What's New in Version 1.0.4

Fixed issue with images stretching.

Category: Education  
Updated: Oct 26, 2011  
Version: 1.0.4  
Size: 108 MB  
Language: English  
Seller: Eric Bouchoc  
© 2010 AnimMan Studios

Update

Rated 4+

Requirements: Compatible with iPad.  
Requires iOS 3.2 or later

iPad Screenshots

Customer Ratings

We have not received enough ratings to display an average for the current version of this application.

Rate this application: ★★★★★

Customer Reviews

Current Version (0) All Versions (2)

There are no reviews for the current version of this application. See all 2 reviews for previous versions.

Be the first to write a review >

LIBRARY

- Music
- Movies
- TV Shows
- Podcasts
- iTunes U
- Books
- Apps
- Tones
- Radio

STORE

- iTunes Store
- iTunes Match
- Ping
- Purchased
- Purchased on Chris Judd's iPhone
- Purchased on Chris Judd's iPad
- Downloads

DEVICES

- Chris Judd's iPhone

SHARED

- Chey B's milk crate

GENIUS

- Genius

PLAYLISTS

- iTunes DJ
- Audio Recently Added
- Classical Music
- Recently Added
- Recently Played
- Religious
- Top 25 Most Played
- 80s
- Beyond The Goal
- Blake music
- Burn CD
- Christmas Music
- Christopher Judd's Playlist
- COJUG Interviews
- COJUG Podcast
- E-Myth Revisited
- Emotional Intelligence
- Favorite Christian
- feel good
- Heavy Christian
- Lost Highway
- must listen
- On-The-Go
- Rio
- Stanford iPhone Class
- Voice Memos
- WOW 2011

+

Friday, December 2, 2011



\$

# Analytics

## Apple

Store Mac iPod + iTunes iPhone Downloads Support

iTunes Connect apple@juddsolutions.com, Judd Solutions, LLC Sign Out

Reporting > Monthly Free App Reports > Preview

Monthly Top 50 For: 11/30/2009 To 12/27/2009 Previous Download

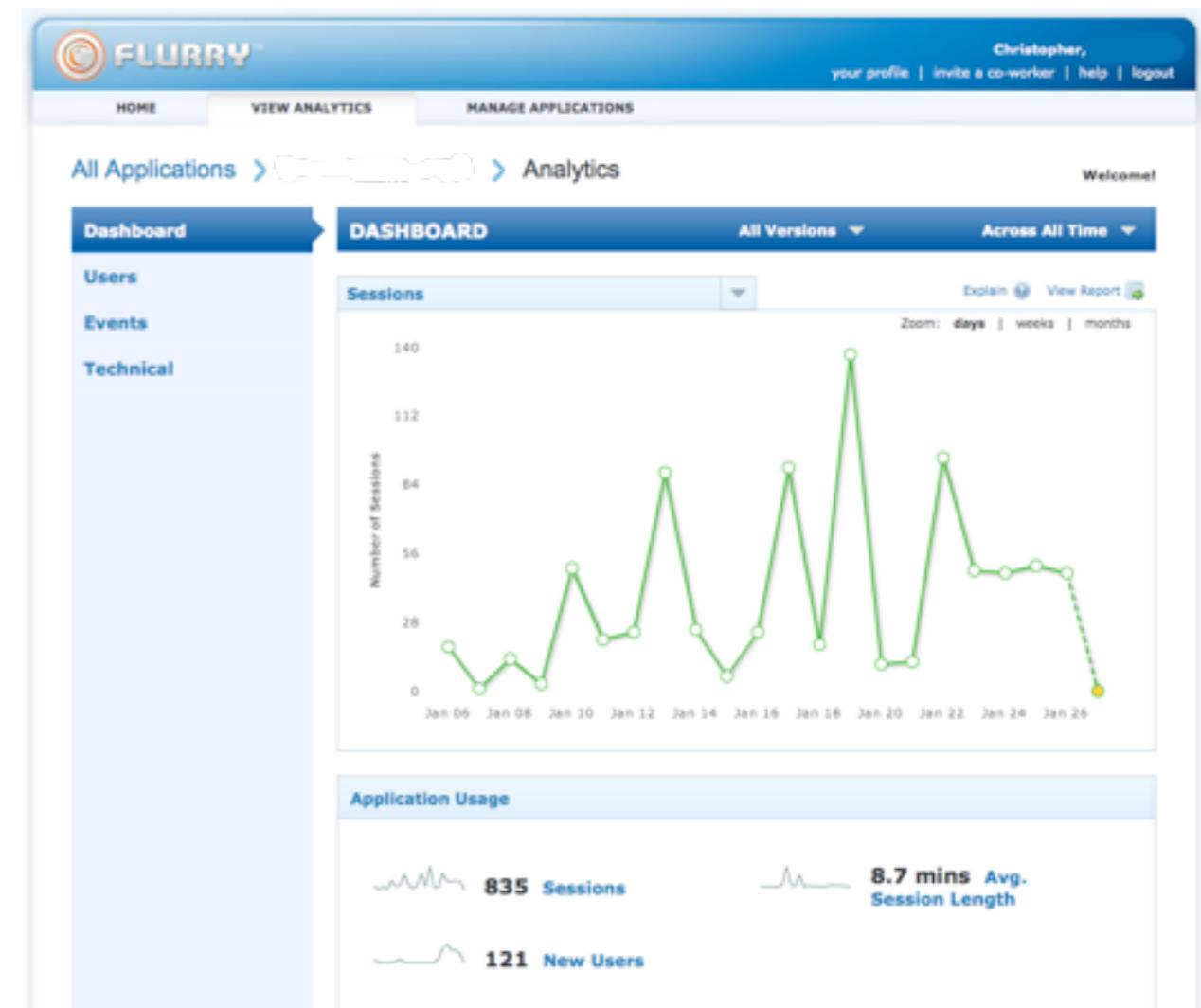
User Guide

Developer	Title/Application	Product Type Identifier	Units	Partner Share	Customer Currency	Customer Price	Customer Currency	Country Code	Apple Identifier	Vendor Identifier
Judd Solutions	Chmod	1	47	0	USD	0	USD	US	306208057	com.juddsolutions.chmod.*
Judd Solutions	Chmod	1	15	0	EUR	0	EUR	DE	306208057	com.juddsolutions.chmod.*
Judd Solutions	Chmod	1	14	0	EUR	0	EUR	FR	306208057	com.juddsolutions.chmod.*
Judd Solutions	Chmod	1	7	0	GBP	0	GBP	GB	306208057	com.juddsolutions.chmod.*
Judd Solutions	Chmod	1	5	0	CAD	0	CAD	CA	306208057	com.juddsolutions.chmod.*
Judd Solutions	Chmod	1	4	0	EUR	0	SEK	SE	306208057	com.juddsolutions.chmod.*
Judd Solutions	Chmod	1	3	0	EUR	0	EUR	BE	306208057	com.juddsolutions.chmod.*
Judd Solutions	Chmod	1	3	0	USD	0	USD	SG	306208057	com.juddsolutions.chmod.*
Judd Solutions	Chmod	1	2	0	EUR	0	EUR	AT	306208057	com.juddsolutions.chmod.*
Judd Solutions	Chmod	1	2	0	EUR	0	EUR	ES	306208057	com.juddsolutions.chmod.*
Judd Solutions	Chmod	1	2	0	EUR	0	EUR	GR	306208057	com.juddsolutions.chmod.*
Judd Solutions	Chmod	1	2	0	EUR	0	EUR	IT	306208057	com.juddsolutions.chmod.*
Judd Solutions	Chmod	1	2	0	USD	0	MXN	MX	306208057	com.juddsolutions.chmod.*
Judd Solutions	Chmod	1	2	0	USD	0	USD	PK	306208057	com.juddsolutions.chmod.*
Judd Solutions	Chmod	1	1	0	EUR	0	CHF	CH	306208057	com.juddsolutions.chmod.*
Judd Solutions	Chmod	1	1	0	EUR	0	EUR	FI	306208057	com.juddsolutions.chmod.*
Judd Solutions	Chmod	1	1	0	EUR	0	EUR	HU	306208057	com.juddsolutions.chmod.*
Judd Solutions	Chmod	1	1	0	EUR	0	EUR	IE	306208057	com.juddsolutions.chmod.*
Judd Solutions	Chmod	1	1	0	EUR	0	EUR	NL	306208057	com.juddsolutions.chmod.*
Judd Solutions	Chmod	1	1	0	JPY	0	JPY	JP	306208057	com.juddsolutions.chmod.*
Judd Solutions	Chmod	1	1	0	USD	0	USD	AE	306208057	com.juddsolutions.chmod.*
Judd Solutions	Chmod	1	1	0	USD	0	USD	CN	306208057	com.juddsolutions.chmod.*
Judd Solutions	Chmod	1	1	0	USD	0	USD	CO	306208057	com.juddsolutions.chmod.*
Judd Solutions	Chmod	1	1	0	USD	0	USD	HK	306208057	com.juddsolutions.chmod.*
Judd Solutions	Chmod	1	1	0	USD	0	USD	KR	306208057	com.juddsolutions.chmod.*
Judd Solutions	Chmod	1	1	0	USD	0	USD	RU	306208057	com.juddsolutions.chmod.*

If you have feedback or questions please let us know.

Previous Download

## Downloads



- Uses
- New Users
- Device Types
- Locations
- Events

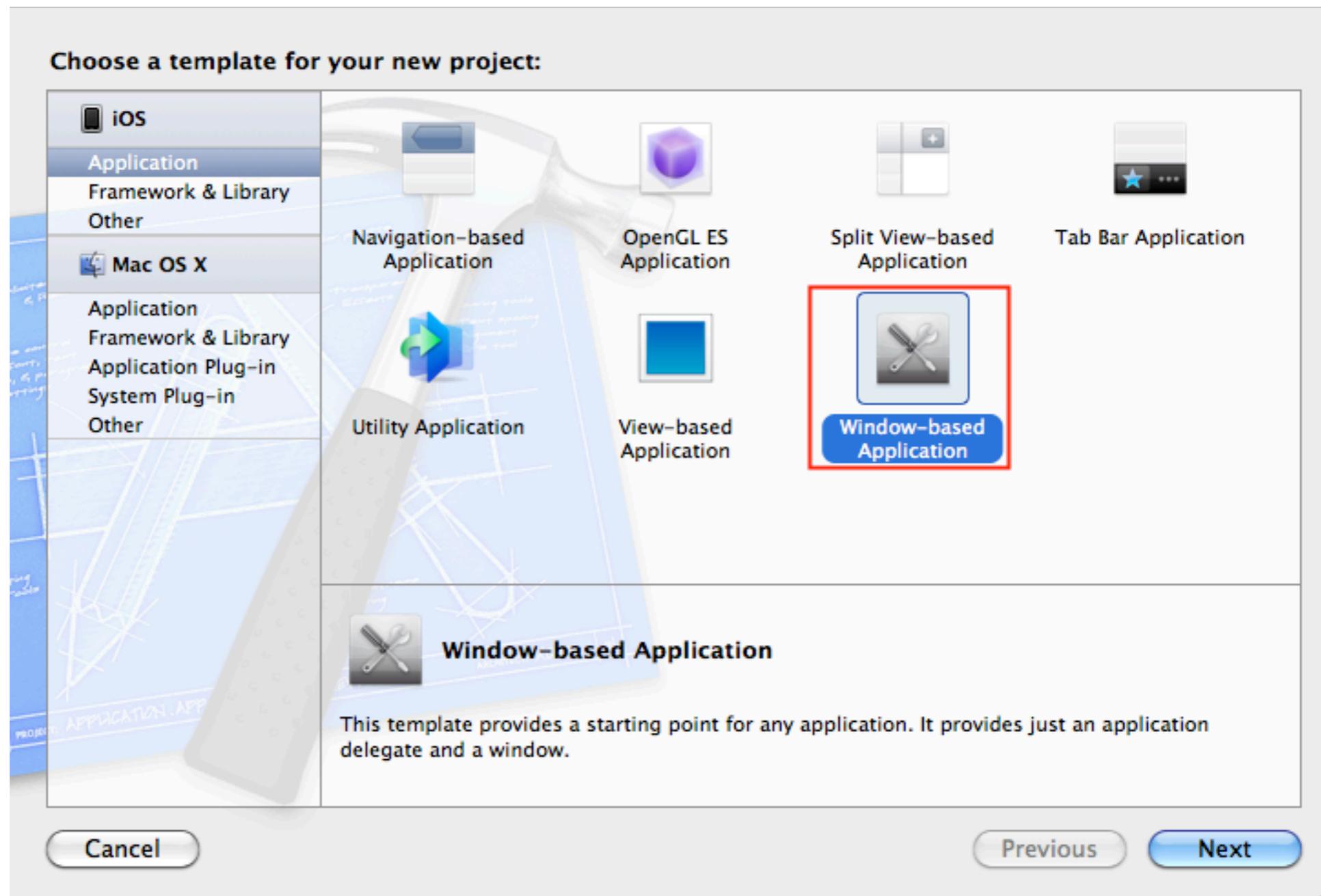
# **PROJECT 2**

# EVERYNODE

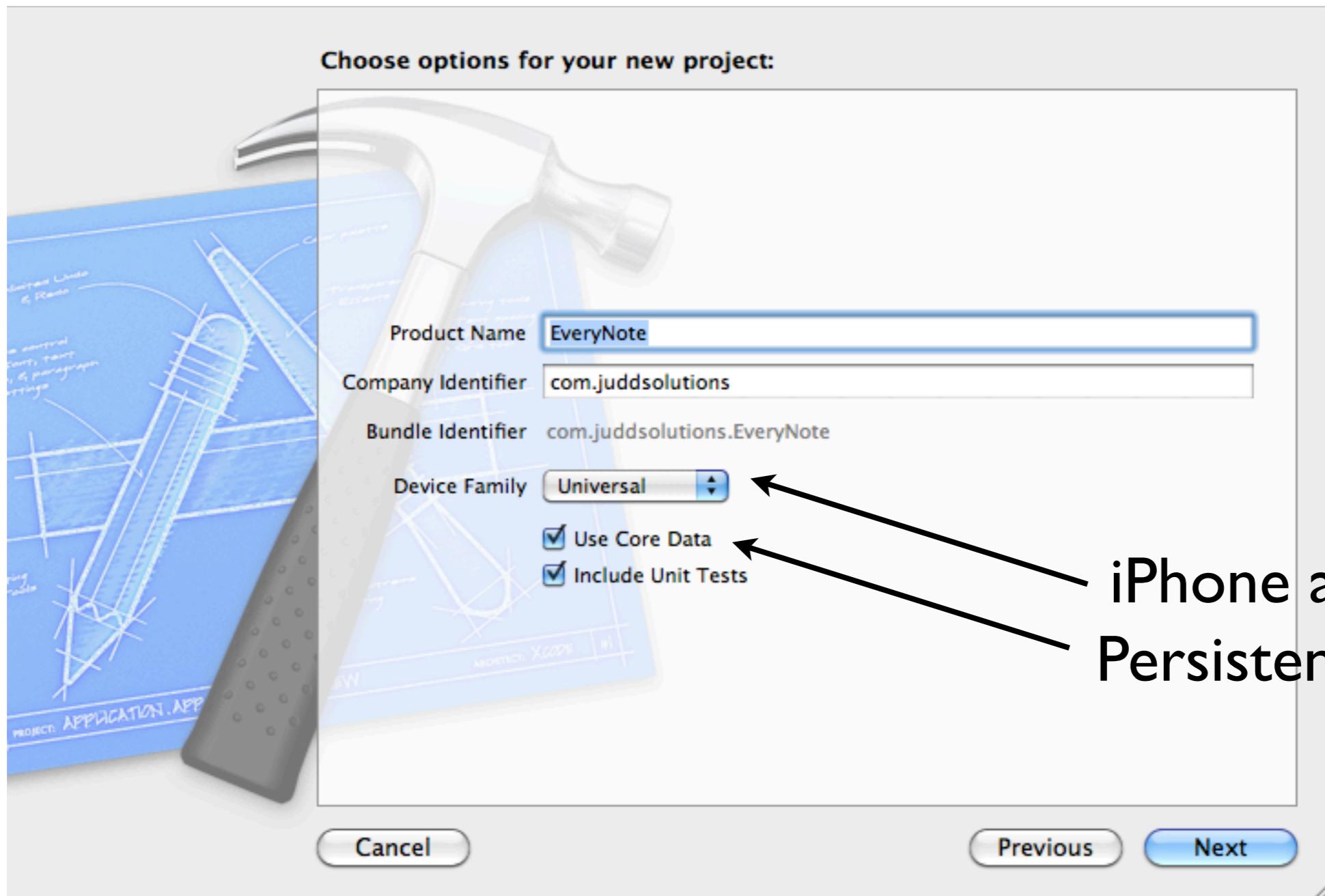


# **MAKING IT FANCY**

# Choose project template



*Window-based is the only universal option*



iPhone and iPad  
Persistence

# App Icons

Screenshot of the Xcode Project Navigator showing the 'EveryNote' target settings.

**PROJECT**: EveryNote

**TARGETS**: EveryNote (selected)

**iOS Application Target**

- Identifier: com.juddsolutions.EveryNote
- Version: 1.0
- Devices: Universal
- Deployment Target: 4.3

**iPhone / iPod Deployment Info**

- Main Interface: MainWindow\_iPhone

**Supported Device Orientations**

- Portrait
- Upside Down
- Landscape Left
- Landscape Right

**App Icons**

- Icon.png (iPhone, 57x57)
- Icon-72.png (iPad, 72x72)
- Icon@2x.png (iPhone Retina, 114x114)



Icon.png  
iPhone  
57x57

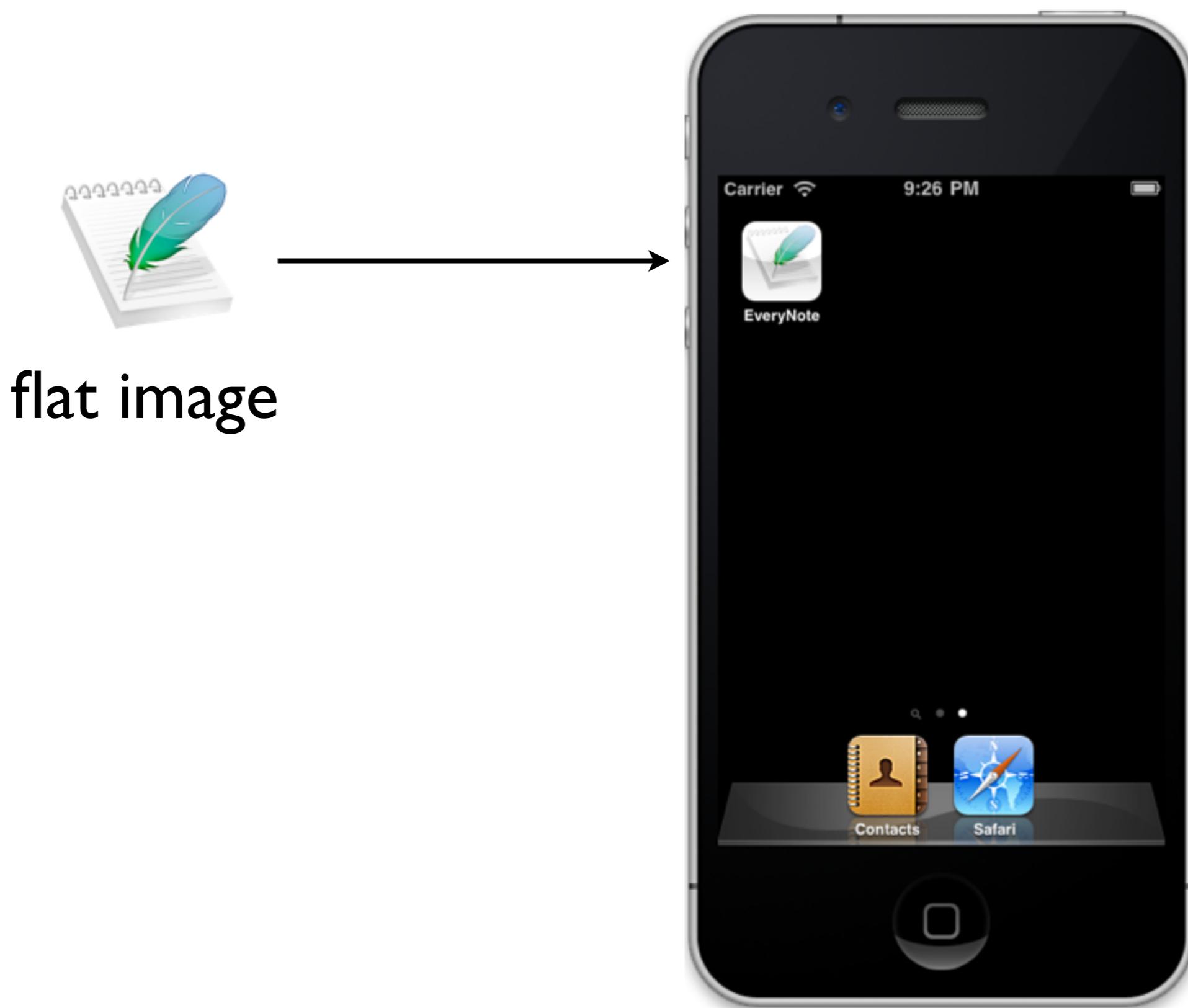


Icon-72.png  
iPad  
72x72



Icon@2x.png  
iPhone Retina  
114x114

# Default App Icon

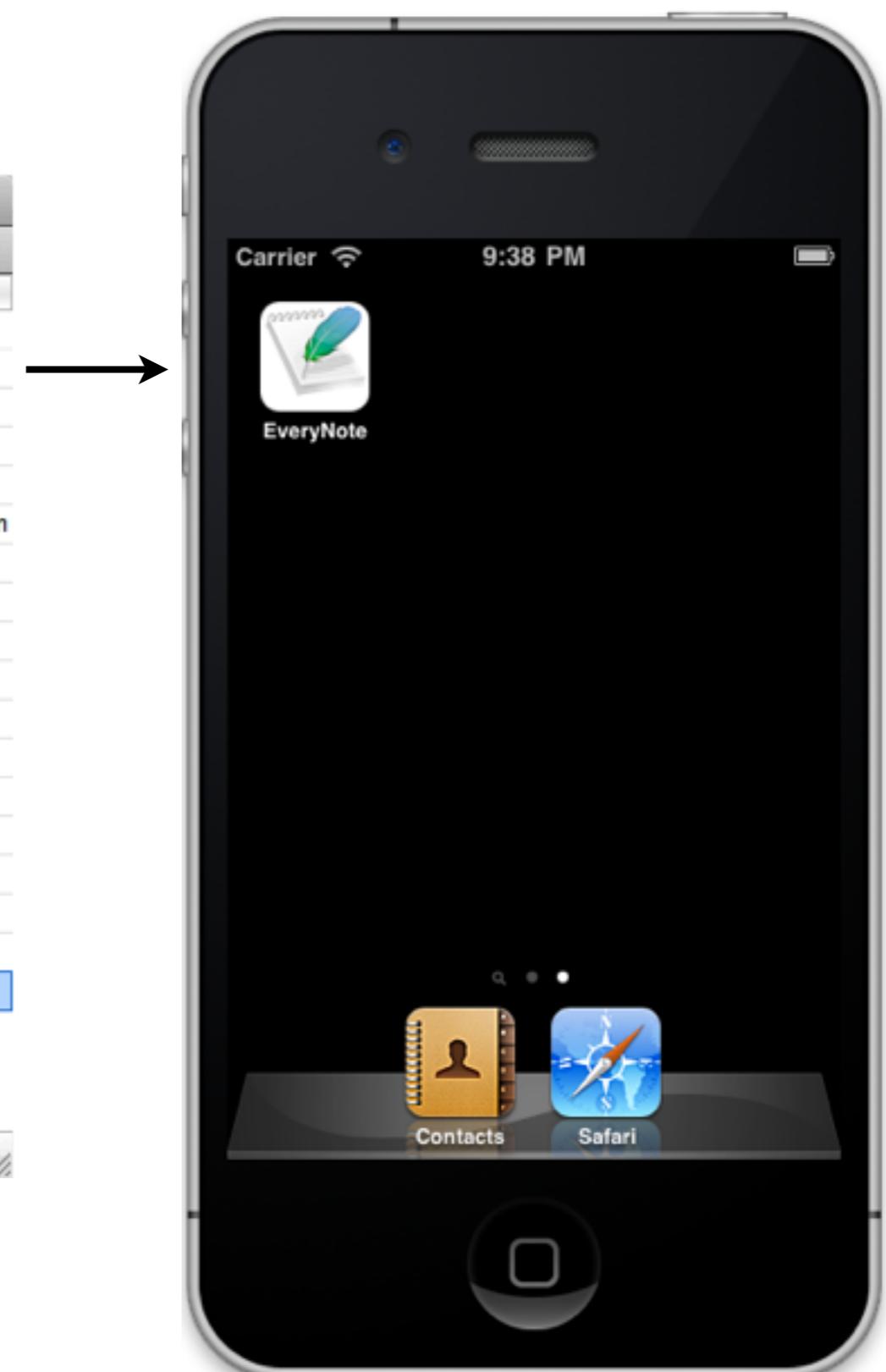


rounded & shine

# No Shine

## Info.plist

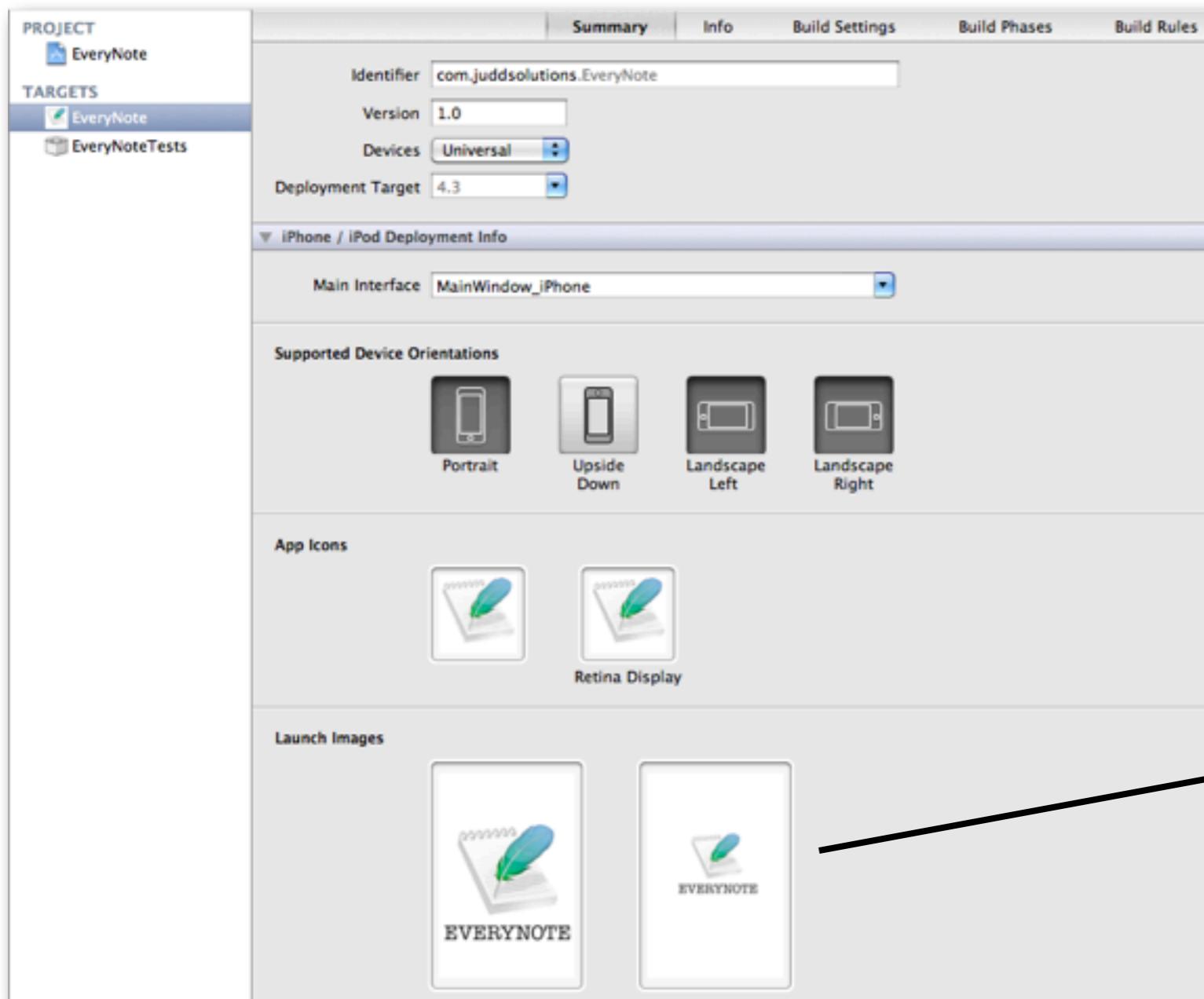
Key	Type	Value
Localization native development region	String	en
Bundle display name	String	\${PRODUCT_NAME}
Executable file	String	\${EXECUTABLE_NAME}
Icon file	String	
▶ Icon files	Array	(3 items)
Bundle identifier	String	com.juddsolutions.\${PRODUCT_NAME}:rfc1034iden
InfoDictionary version	String	6.0
Bundle name	String	\${PRODUCT_NAME}
Bundle OS Type code	String	APPL
Bundle versions string, short	String	1.0
Bundle creator OS Type code	String	????
Bundle version	String	1.0
Application requires iPhone environment	Boolean	YES
Main nib file base name	String	MainWindow_iPhone
Main nib file base name (iPad)	String	MainWindow_iPad
▶ Supported interface orientations	Array	(3 items)
▶ Supported interface orientations (iPad)	Array	(4 items)
Icon already includes gloss effects	Boolean	YES



# Required and optional icons

Image Size (px)	File Name	Used For	Required Status
512x512	iTunesArtwork	Ad Hoc iTunes	Optional but recommended
57x57	Icon.png	App Store and Home screen on iPhone/iPod touch	Required
114x114	Icon@2x.png	Home screen for iPhone 4 High Resolution	Optional but recommended
72x72	Icon-72.png	Home screen for iPad compatibility	Optional but recommended
29x29	Icon-Small.png	Spotlight and Settings	Optional but recommended
50x50	Icon-Small-50.png	Spotlight for iPad compatibility	Recommended if you have a Settings bundle, otherwise optional but recommended
58x58	Icon-Small@2x.png	Spotlight and Settings for iPhone 4 High Resolution	Recommended if you have a Settings bundle, otherwise optional but recommended

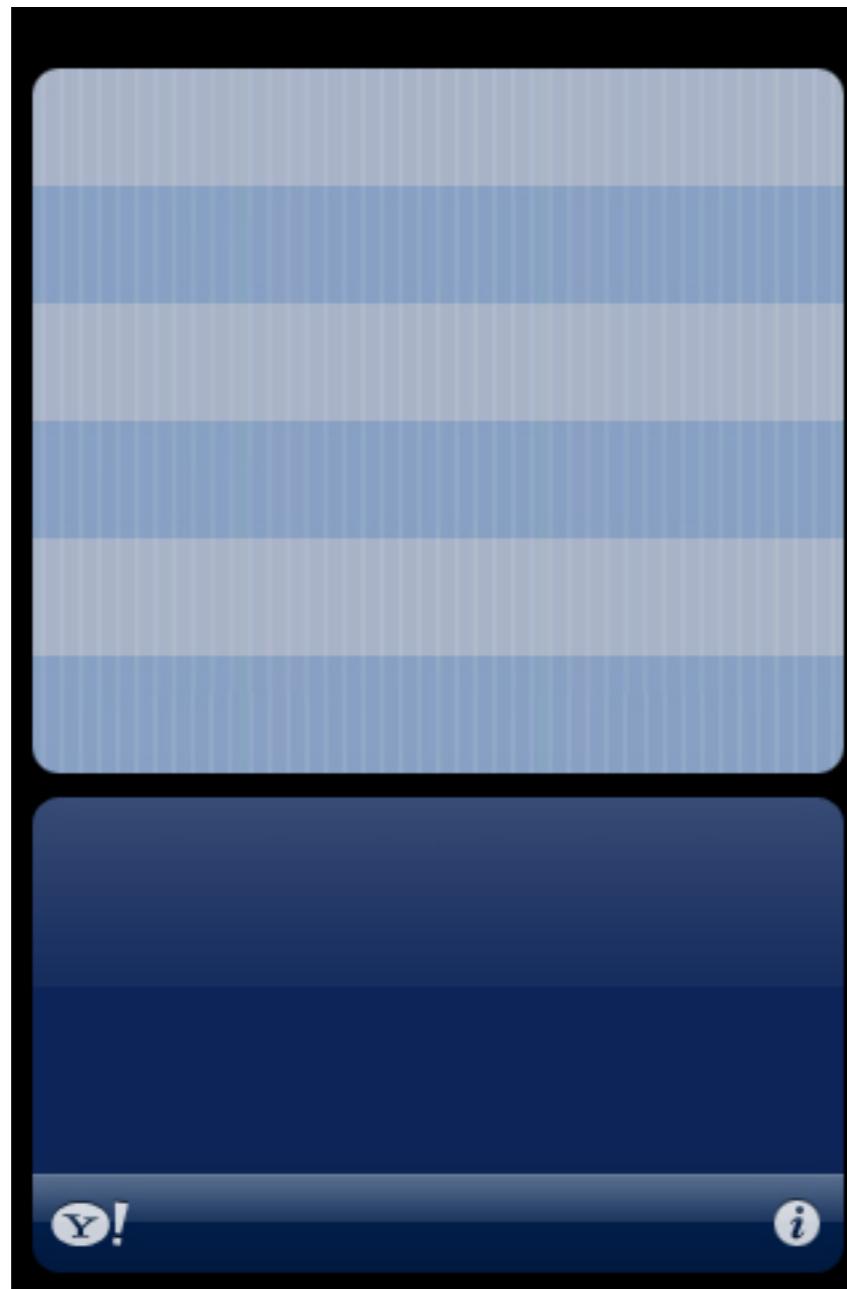
# Launch Image/Splash screen



Device	Portrait	Landscape
iPhone and iPod touch	320 x 480 pixels 640 x 960 pixels (high resolution)	Not supported
iPad	768 x 1004 pixels	1024 x 748 pixels

[http://developer.apple.com/library/ios/documentation/userexperience/conceptual/mobilehig/IconsImages.html#/apple\\_ref/doc/uid/TP40006556-CH14-SW5](http://developer.apple.com/library/ios/documentation/userexperience/conceptual/mobilehig/IconsImages.html#/apple_ref/doc/uid/TP40006556-CH14-SW5)

# Launch Image



Apple's recommendation is the first screen to make it look like it loads faster

# *Lab 5*

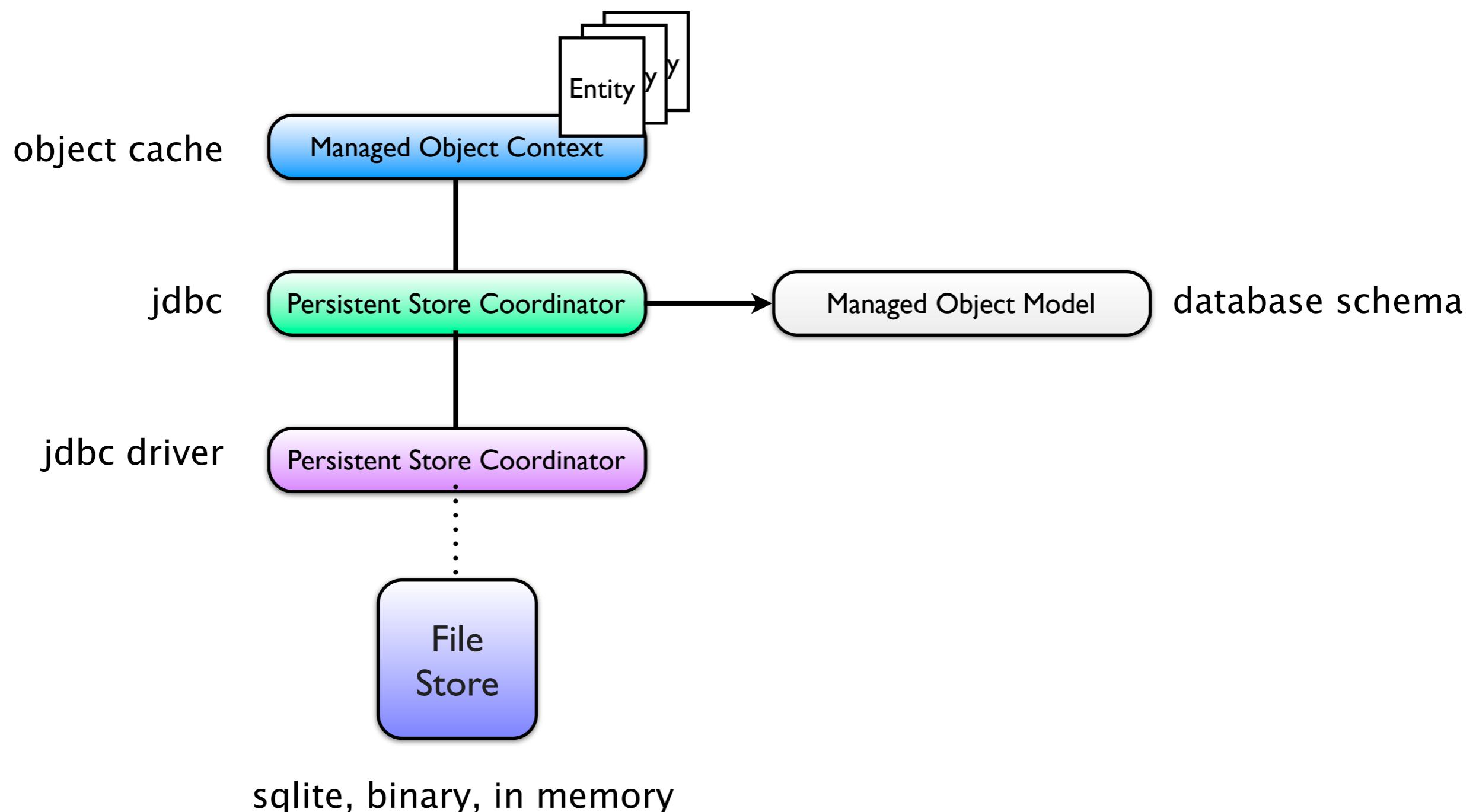
- 1. Create new EveryNote Universal project with CoreData**
- 2. Add iPhone and iPad app icons**
- 3. Add iPhone and iPad launch image**
- 4. Run the application in the simulator**

**note: use images in the resource.zip file**

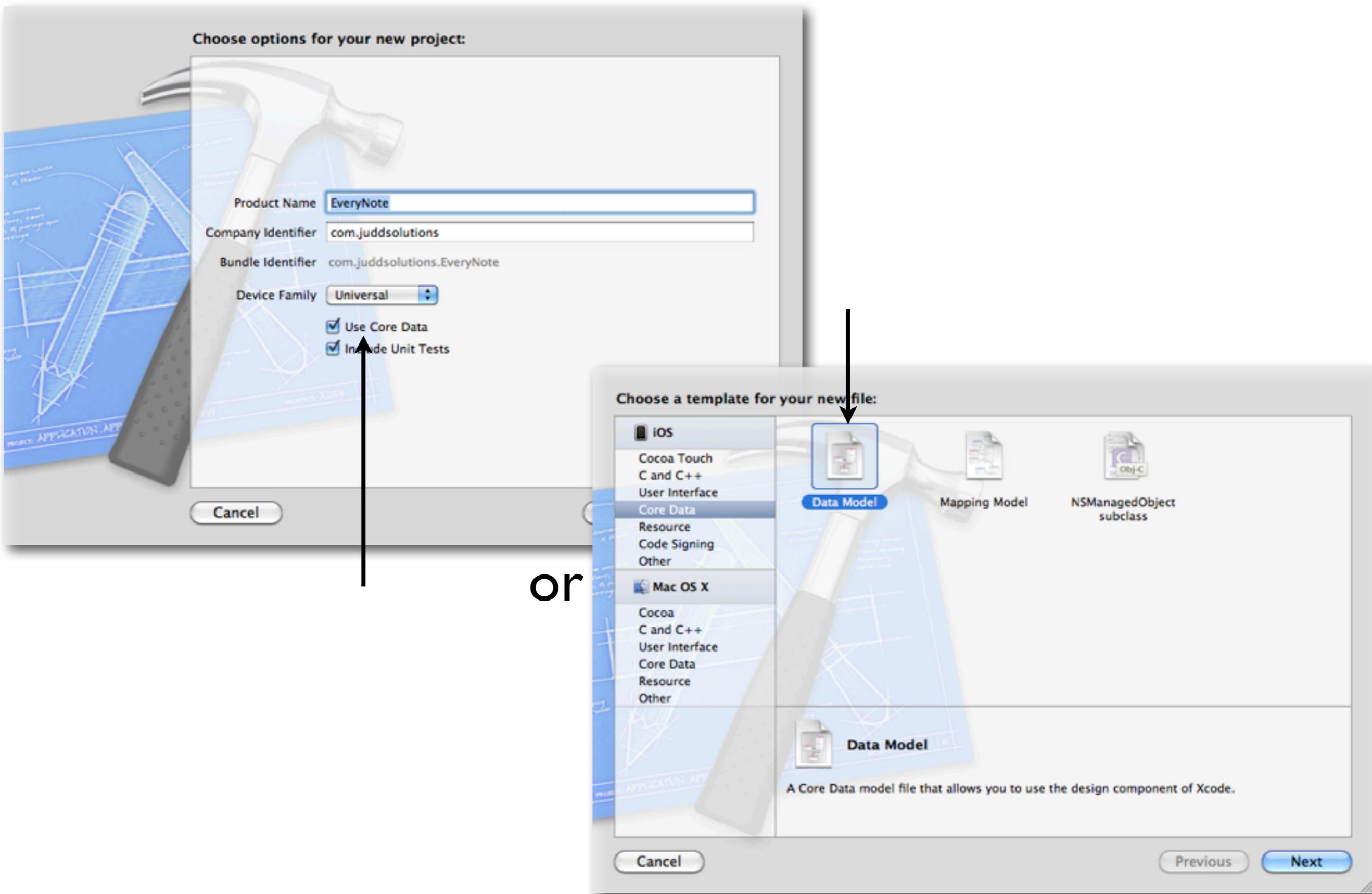
# CORE DATA

**Core Data is a schema-driven object graph management and persistence framework**

# Core Data Architecture

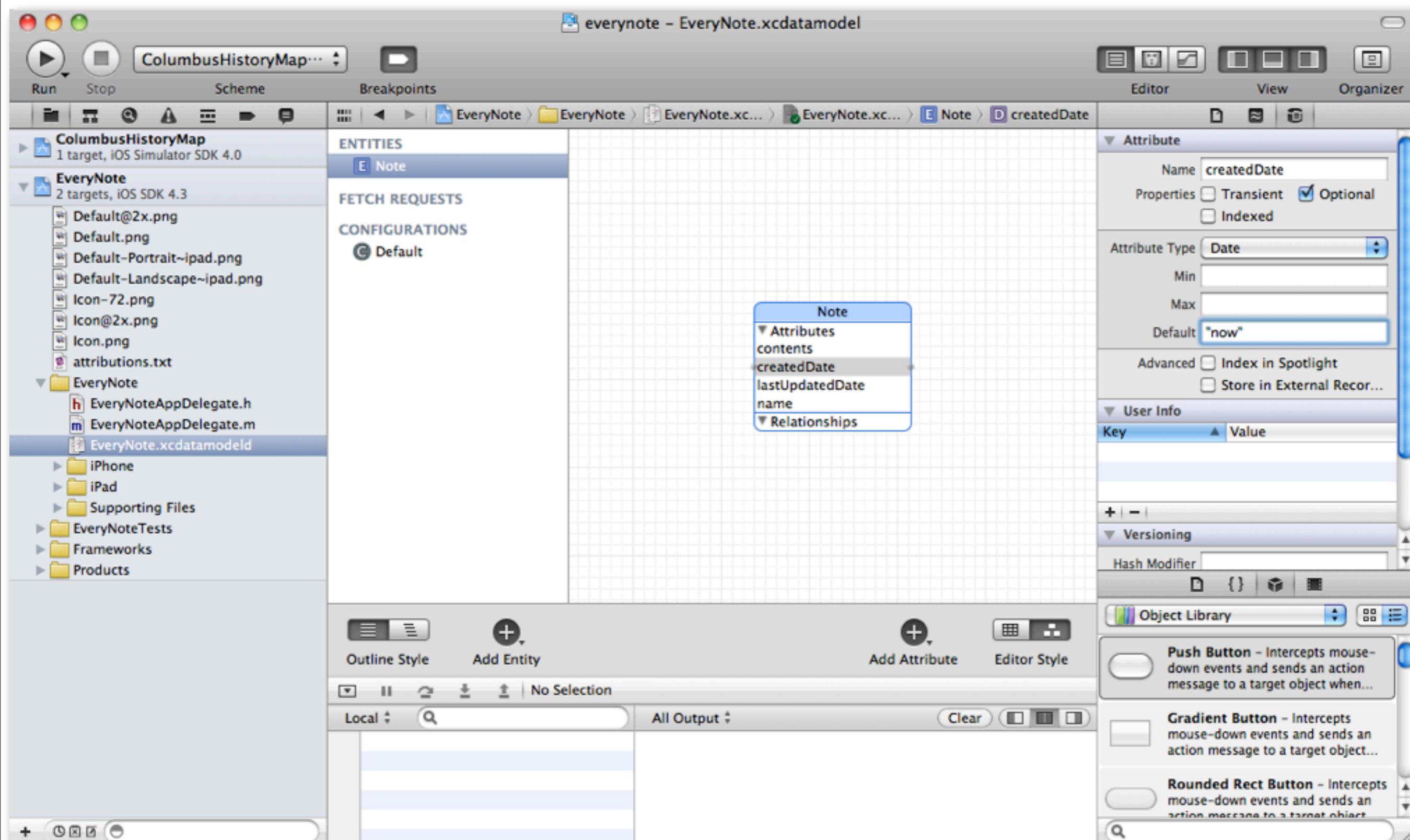


# Adding Core Data

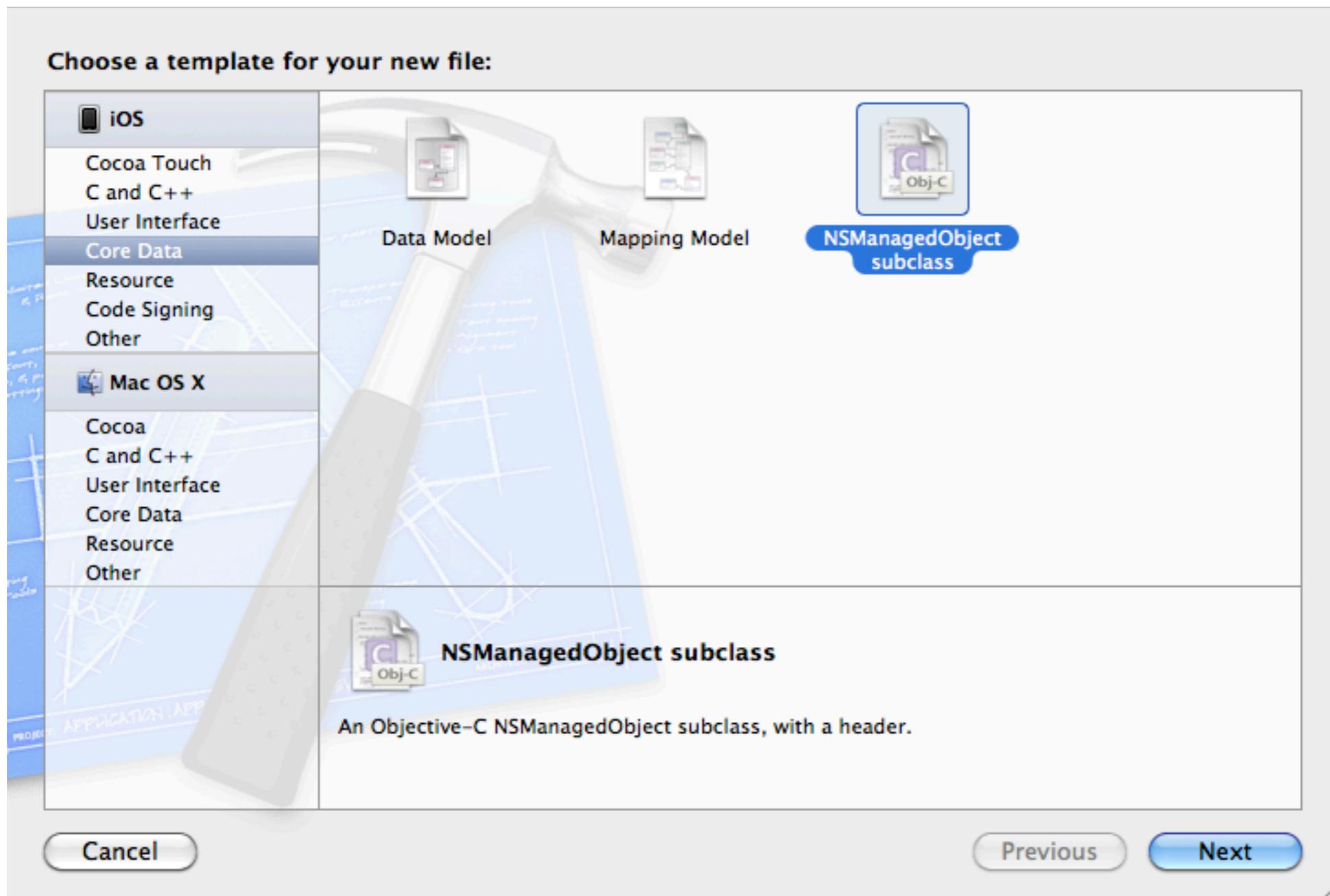


or

# Model Schema



# Generate Classes



or

Editor > Create NSManagedObject subclass

# Create Entity

```
NSManagedObjectContext *context = [self managedObjectContext];
Note *note = [NSEntityDescription
              insertNewObjectForEntityForName:@"Note"
              inManagedObjectContext:context];

note.name = @"Test Note";
note.contents = @"CocoaConf is awesome!!!";
note.createdDate = [NSDate date];
note.lastUpdatedDate = [NSDate date];
```

table name



or

```
NSManagedObjectContext *context = [self managedObjectContext];
NSManagedObject *note = [NSEntityDescription
                           insertNewObjectForEntityForName:@"Note"
                           inManagedObjectContext:context];

[note setValue:@"Test Note" forKey:@"name"];
[note setValue:@"CocoaConf is awesome!!!" forKey:@"contents"];
[note setValue:[NSDate date] forKey:@"createdDate"];
[note setValue:[NSDate date] forKey:@"lastUpdatedDate"];
```

# Save

notice the context is saved on the entity

```
NSError *error;  
if (![context save:&error]) {  
    NSLog(@"Couldn't save: %@", [error localizedDescription]);  
}
```



# Retrieve Entities

fetch is like select

```
NSFetchRequest *fetchRequest = [[NSFetchRequest alloc] init];
NSEntityDescription *entity = [NSEntityDescription entityForName:@"Note"
                             inManagedObjectContext:context];

[fetchRequest setEntity:entity];

NSArray *notes = [context executeFetchRequest:fetchRequest error:&error];

for (Note *note in notes) {
    NSLog(@"Note: name=%@, content=%@, createdDate=%@",
          note.name, note.contents, note.createdDate);
}

[fetchRequest release];
```

table name

can add sorting, predicates, limits

# Deleting Entities

```
[context deleteObject:note];
```

note: don't forget to save the context

*Look Mom, No SQL!*

# View Database

location is value of storeURL

EveryNote | iPhone 4.3 Simul... Breakpoints Editor View Organizer

EveryNoteAppDelegate.m

By Thread By Queue

Thread 1 com.apple.main-thread

- 0 -[EveryNoteAppDelegate persistentS...
- 1 -[EveryNoteAppDelegate managedO...
- 2 -[EveryNoteAppDelegate application...
- 3 -[UIApplication \_callInitializationDel...
- 15 UIApplicationMain
- 16 main

Thread 2

Thread 3

Thread 4 WebThread

everynote - EveryNoteAppDelegate.m

```
/*
- (NSPersistentStoreCoordinator *)persistentStoreCoordinator
{
    if (_persistentStoreCoordinator != nil)
    {
        return _persistentStoreCoordinator;
    }

    NSURL *storeURL = [[self applicationDocumentsDirectory] URLByAppendingPathComponent:@"EveryNote.sqlite"];

    NSError *error = nil;
    _persistentStoreCoordinator = [[NSPersistentStoreCoordinator alloc] initWithManagedObjectModel:[self
        managedObjectModel]];
    if (![_persistentStoreCoordinator addPersistentStoreWithType:NSSQLiteStoreType configuration:nil URL:
        storeURL options:nil error:&error])
    {
        /*
         Replace this implementation with code to handle the error appropriately.

         abort() causes the application to generate a crash log and terminate. You should not use this function
         in a shipping application, although it may be useful during development. If it is not possible to
         recover from the error, display an alert panel that instructs the user to quit the application by
         pressing the Home button.

         Typical reasons for an error here include:
         * The persistent store is not accessible;
         * The schema for the persistent store is incompatible with current managed object model.
         Check the error message to determine what the actual problem was.
        */
    }
}
```

No Selection

Local All Output

```
GNU gdb 6.3.50-20050815 (Apple version gdb-1518) (Sat Feb 12 02:52:12 UTC 2011)
Copyright 2004 Free Software Foundation, Inc.
GDB is free software, covered by the GNU General Public License, and you are
welcome to change it and/or distribute copies of it under certain conditions.
Type "show copying" to see the conditions.
There is absolutely no warranty for GDB. Type "show warranty" for details.
This GDB was configured as "x86_64-apple-darwin".sharedlibrary apply-load-rule
s all
Attaching to process 23158.
Pending breakpoint 2 - ""EveryNoteAppDelegate.m":182" resolved
Current language: auto; currently objective-c
Printing description of storeURL:
<CFURL 0x4d35120 [0x1005400]>{type = 15, string = file://localhost/Users/cjudd
/Library/Application%20Support/iPhone%20Simulator/4.3/Applications/1C403EFB-73
9A-41EF-BD74-286A8AFF1E82/Documents/EveryNote.sqlite, encoding = 134217984, ba
se = (null)}
(gdb)
```

Object Library

- 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...

location

# View Database

```
$ sqlite3 EveryNote.sqlite
SQLite version 3.6.12
Enter ".help" for instructions
Enter SQL statements terminated with a ";"

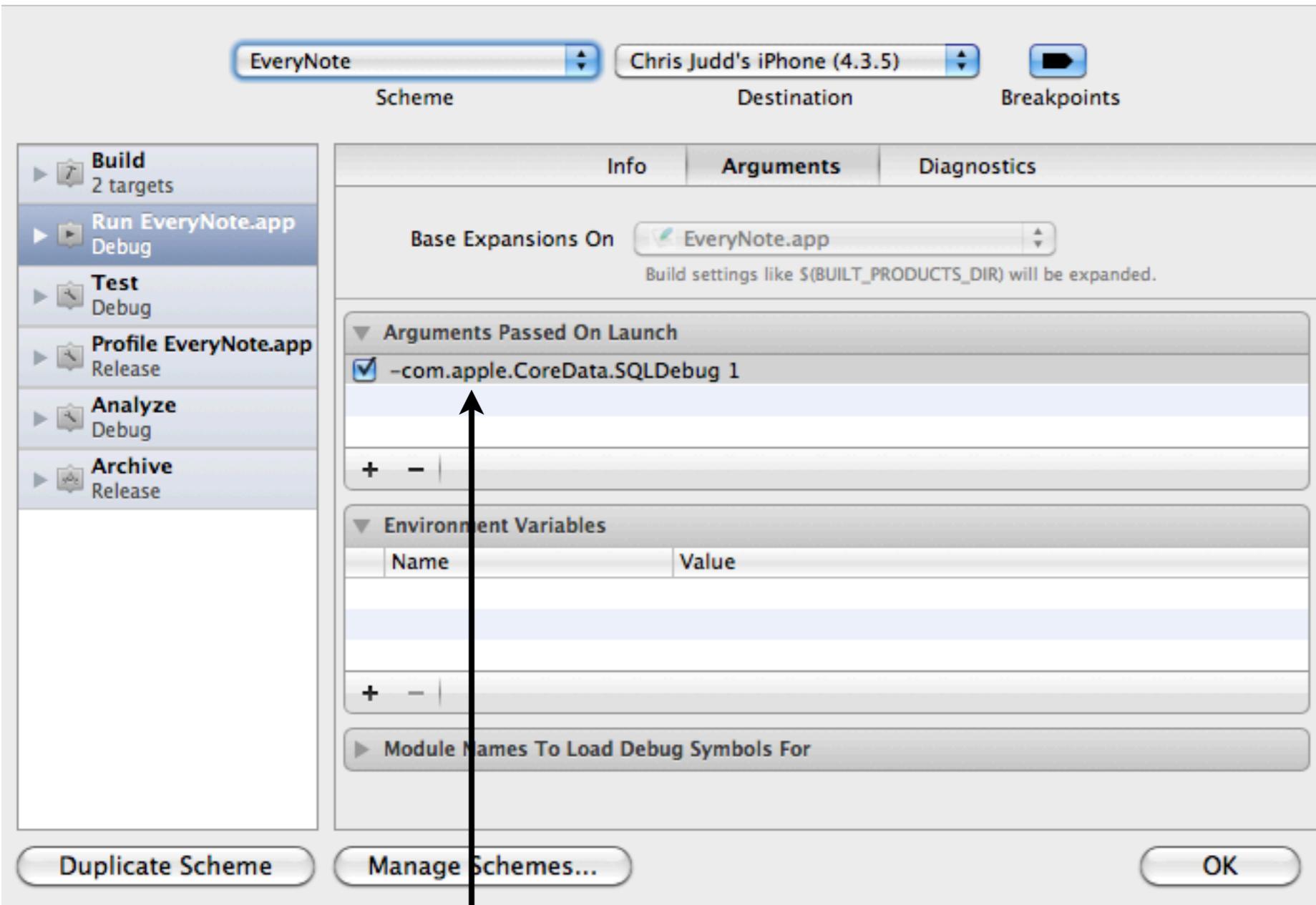
sqlite> .tables
ZNOTE      Z_METADATA   Z_PRIMARYKEY

sqlite> .schema ZNOTE
CREATE TABLE ZNOTE ( Z_PK INTEGER PRIMARY KEY, Z_ENT INTEGER, Z_OPT
INTEGER, ZLASTUPDATED DATE TIMESTAMP, ZCREATED DATE TIMESTAMP, ZNAME
VARCHAR, ZCONTENTS VARCHAR );

sqlite> select * from ZNOTE;
9||1|334667144.391775|334667144.391766[Test Note|CocoaConf is awesome!!!

sqlite>
```

# View Generated SQL



show SQL

produces

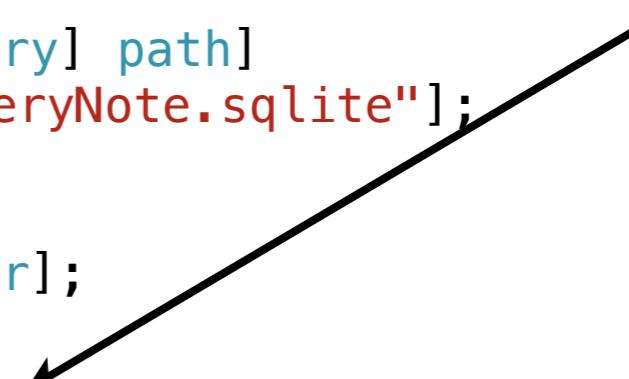
```
2011-08-10 07:15:17.950 EveryNote[8288:707] CoreData: annotation: Connecting to sqlite database file at "/var/mobile/Applications/C7A691D8-5EBF-456D-AFD3-0F6B80DEC739/Documents/EveryNote.sqlite"
2011-08-10 07:15:17.963 EveryNote[8288:707] CoreData: sql: pragma cache_size=200
2011-08-10 07:15:17.968 EveryNote[8288:707] CoreData: sql: SELECT Z_VERSION, Z_UUID, Z_PLIST FROM Z_METADATA
2011-08-10 07:15:17.983 EveryNote[8288:707] CoreData: sql: BEGIN EXCLUSIVE
2011-08-10 07:15:17.988 EveryNote[8288:707] CoreData: sql: SELECT Z_MAX FROM Z_PRIMARYKEY WHERE Z_ENT = ?
2011-08-10 07:15:17.993 EveryNote[8288:707] CoreData: sql: UPDATE Z_PRIMARYKEY SET Z_MAX = ? WHERE Z_ENT = ? AND Z_MAX = ?
2011-08-10 07:15:18.001 EveryNote[8288:707] CoreData: sql: COMMIT
2011-08-10 07:15:18.104 EveryNote[8288:707] CoreData: sql: BEGIN EXCLUSIVE
2011-08-10 07:15:18.110 EveryNote[8288:707] CoreData: sql: INSERT INTO ZNOTE(Z_PK, Z_ENT, Z_OPT, ZNAME, ZLASTUPDATEDDATE, ZCREATEDDATE, ZCONTENTS) VALUES(?, ?, ?, ?, ?, ?, ?)
2011-08-10 07:15:18.118 EveryNote[8288:707] CoreData: sql: COMMIT
2011-08-10 07:15:18.297 EveryNote[8288:707] CoreData: sql: pragma page_count
2011-08-10 07:15:18.313 EveryNote[8288:707] CoreData: annotation: sql execution time: 0.0160s
2011-08-10 07:15:18.317 EveryNote[8288:707] CoreData: sql: pragma freelist_count
2011-08-10 07:15:18.321 EveryNote[8288:707] CoreData: annotation: sql execution time: 0.0039s
2011-08-10 07:15:18.327 EveryNote[8288:707] CoreData: sql: SELECT t0.Z_PK, t0.Z_OPT, t0.ZNAME, t0.ZLASTUPDATEDDATE, t0.ZCREATEDDATE, t0.ZCONTENTS FROM ZNOTE t0
2011-08-10 07:15:18.332 EveryNote[8288:707] CoreData: annotation: sql connection fetch time: 0.0047s
2011-08-10 07:15:18.335 EveryNote[8288:707] CoreData: annotation: total fetch execution time: 0.0086s for 2 rows.
2011-08-10 07:15:18.342 EveryNote[8288:707] Note: name=Test Note, content=CocosConf is awesome!!!, createdDate=2011-08-10 11:12:56 +0000
2011-08-10 07:15:18.348 EveryNote[8288:707] Note: name=Test Note, content=CocosConf is awesome!!!, createdDate=2011-08-10 11:15:17 +0000
```

# Use pre-populated database

```
// Stage existing database
NSString *storePath = [[[self applicationDocumentsDirectory] path]
                      stringByAppendingString: @"/EveryNote.sqlite"];
NSURL *storeURL = [NSURL fileURLWithPath:storePath];

NSFileManager *fileManager = [NSFileManager defaultManager];
if (![fileManager fileExistsAtPath:storePath]) {
    NSString *defaultStorePath = [[NSBundle mainBundle]
                                  pathForResource:@"EveryNote" ofType:@"sqlite"];
    if (defaultStorePath) {
        [fileManager copyItemAtPath:defaultStorePath toPath:storePath error:NULL];
    }
}
```

Include database in app



\* replace storeURL in persistentStoreCoordinator

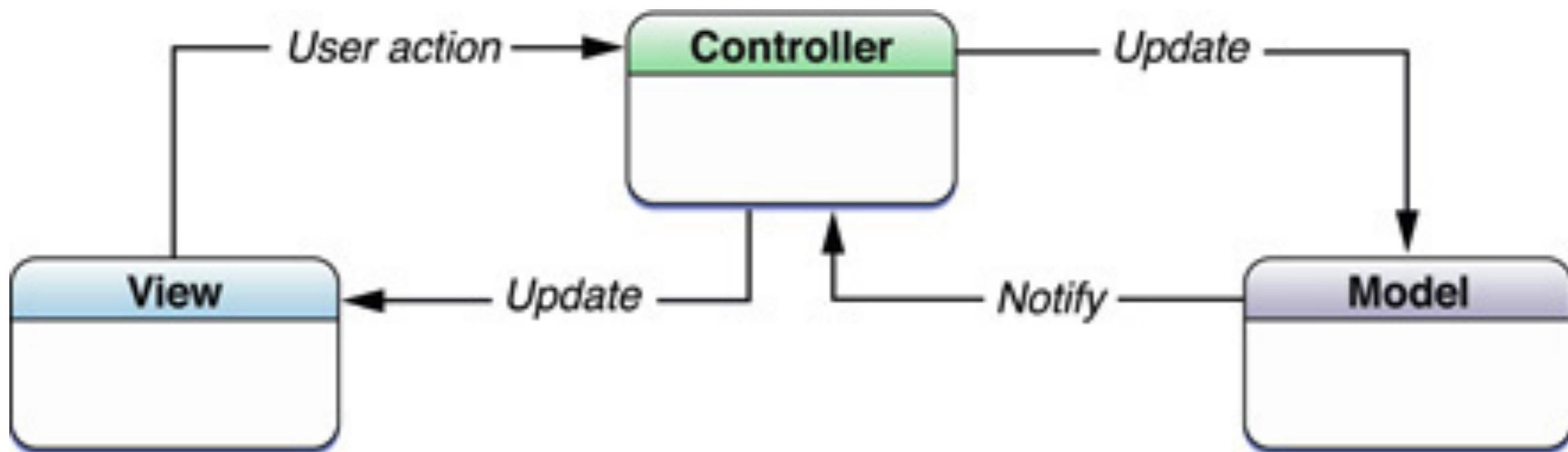
# *Lab 6*

1. On startup (appdelegate's `didFinishLaunchingWithOptions`) create, populate, fetch and print out note data using Core Data.
2. Change application to use a pre-populated database included in the application bundle.

note: use database in the resource.zip file

# **BUILDING THE GUI**

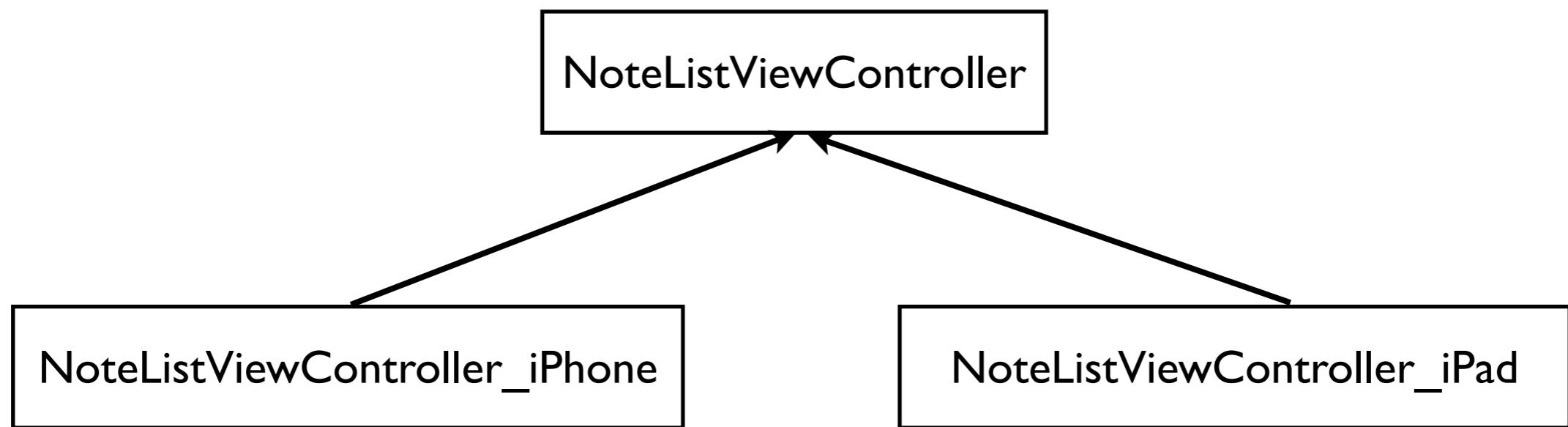
# Model-View-Controller (MVC)



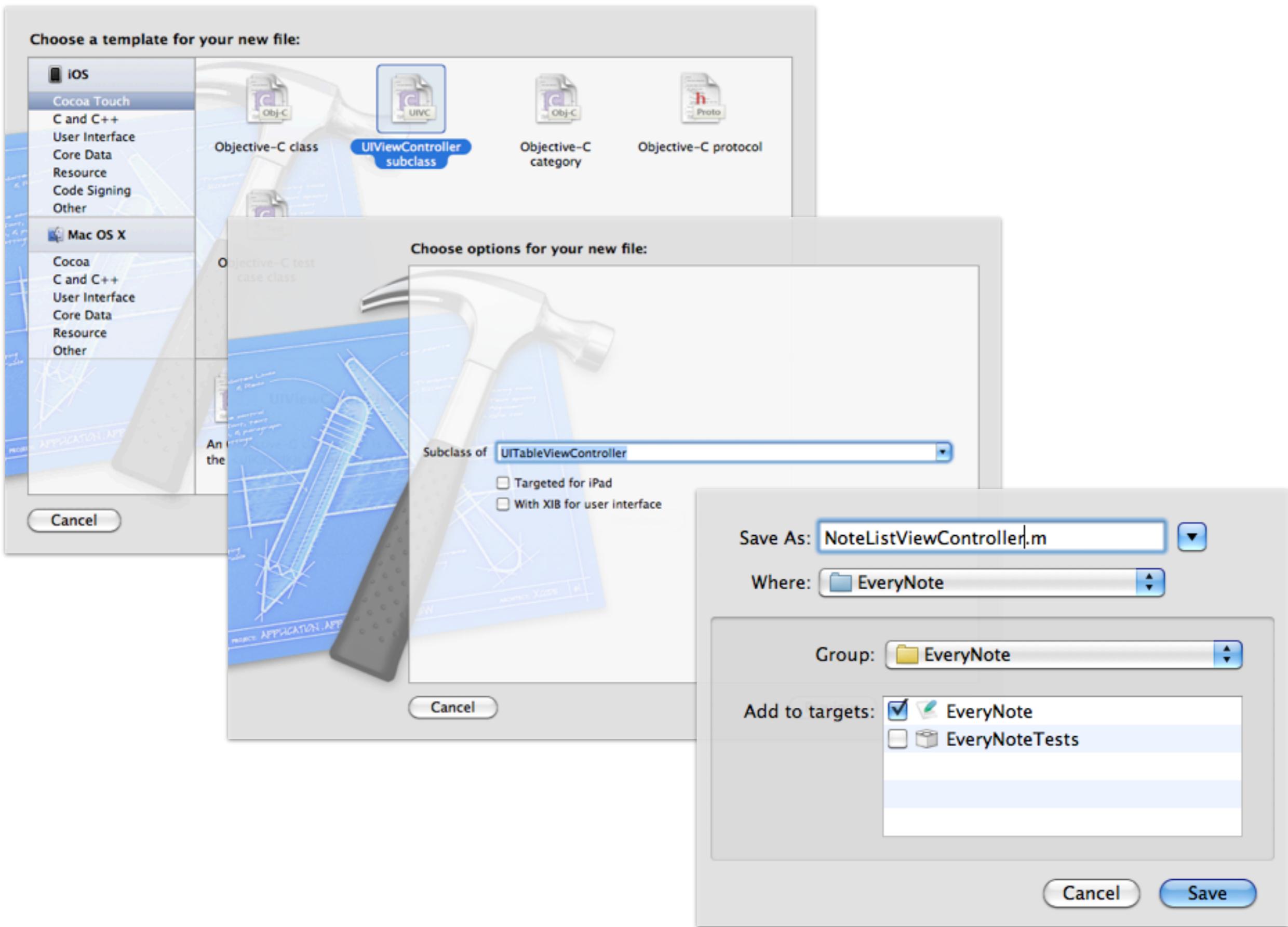
# Determine Device Type

```
if (UI_USER_INTERFACE_IDIOM() == UIUserInterfaceIdiomPad) {  
    NSLog(@"iPad");  
} else {  
    NSLog(@"iPhone");  
}
```

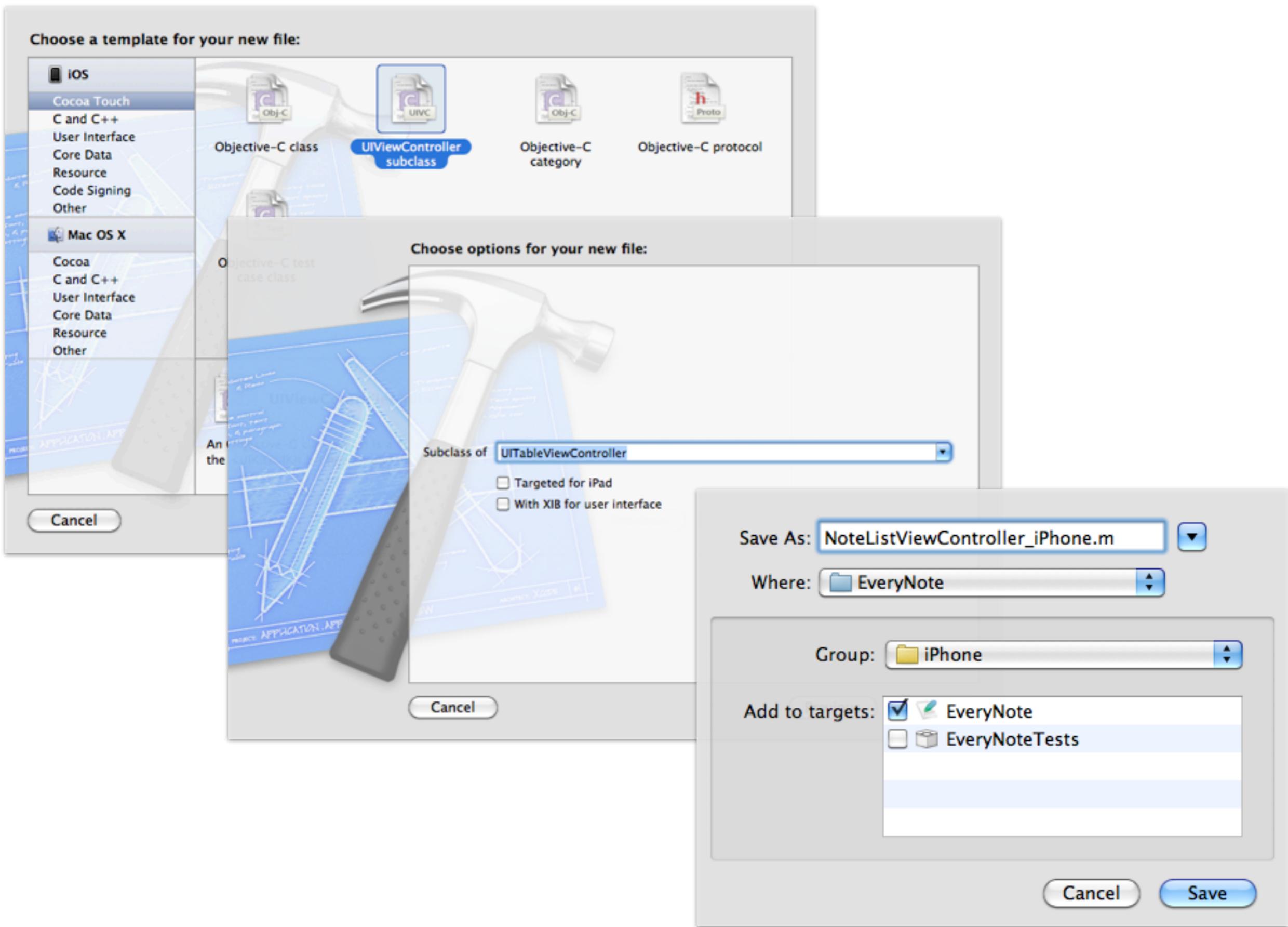
# Universal App Strategy



# Generate Base UITableViewController



# Generate iPhone UITableViewController



# Add Navigation Controller Property

EveryNoteAppDelegate.h

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

@interface EveryNoteAppDelegate : NSObject <UIApplicationDelegate> {

}

@property (nonatomic, retain) IBOutlet UIWindow *window;
@property (nonatomic, retain) IBOutlet UINavigationController *navController;

@property (nonatomic, retain, readonly) NSManagedObjectContext
*managedObjectContext;
@property (nonatomic, retain, readonly) NSManagedObjectModel *managedObjectModel;
@property (nonatomic, retain, readonly) NSPersistentStoreCoordinator
*persistentStoreCoordinator;

- (void)saveContext;
- (NSURL *)applicationDocumentsDirectory;

@end
```

EveryNoteAppDelegate.m

```
#import "EveryNoteAppDelegate.h"
#import "Note.h"

@implementation EveryNoteAppDelegate

@synthesize window=_window;
@synthesize navController=_navController;

@synthesize managedObjectContext=__managedObjectContext;
@synthesize managedObjectModel=__managedObjectModel;
@synthesize persistentStoreCoordinator=__persistentStoreCoordinator;

// Removed for brevity

@end
```

# Create Navigation Controller

EveryNoteAppDelegate\_iPhone.m

```
#import "EveryNoteAppDelegate_iPhone.h"
#import "NoteListViewController_iPhone.h"

@implementation EveryNoteAppDelegate_iPhone

- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions {
    [super application:application didFinishLaunchingWithOptions:launchOptions];
    NoteListViewController *viewController = [[NoteListViewController_iPhone alloc]
                                             initWithStyle:UITableViewStylePlain];
    NSManagedObjectContext *context = [self managedObjectContext];
    // Pass the managed object context to the view controller.
    viewController.managedObjectContext = context;
    UINavigationController *aNavController = [[UINavigationController alloc]
                                             initWithRootViewController:viewController];
    self.navController = aNavController;
    [self.window addSubview:[self.navController view]];
    [self.window makeKeyAndVisible];
    [viewController release];
    [aNavController release];
    return YES;
}

- (void)dealloc
{
    [super dealloc];
}

@end
```

# Removed Generated Template Code

NoteListViewController\_iPhone.m

```
#import "NoteListViewController_iPhone.h"

@implementation NoteListViewController_iPhone

@end
```

# Add Displaying Data

NoteListViewController.h

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

@interface NoteListViewController : UITableViewController {
}

@property (nonatomic, retain) NSMutableArray *notes;
@property (nonatomic, retain) NSManagedObjectContext *managedObjectContext;

@end
```

```
#import "NoteListViewController.h"
#import "Note.h"

@implementation NoteListViewController

@synthesize notes=__notes;
@synthesize managedObjectContext=__managedObjectContext;

- (void)dealloc
{
    [__notes release];
    [__managedObjectContext release];
    [super dealloc];
}

- (void)viewDidLoad
{
    [super viewDidLoad];

    self.title = @"Notes";

    NSFetchedResultsController *fetchedResultsController = [[NSFetchedResultsController alloc] init];
    NSEntityDescription *entity = [NSEntityDescription
                                  entityForName:@"Note" inManagedObjectContext:__managedObjectContext];
    [fetchedResultsController setEntity:entity];
    NSError *error;
    __notes = [[__managedObjectContext executeFetchRequest:fetchedResultsController error:&error] mutableCopy];
    [fetchedResultsController release];
}

- (NSInteger)numberOfSectionsInTableView:(UITableView *)tableView
{
    return 1;
}

- (NSInteger)tableView:(UITableView *)tableView numberOfRowsInSection:(NSInteger)section
{
    return [__notes count];
}

// tableView:(UITableView *)tableView cellForRowAtIndexPath is on next slide
// Removed for brevity

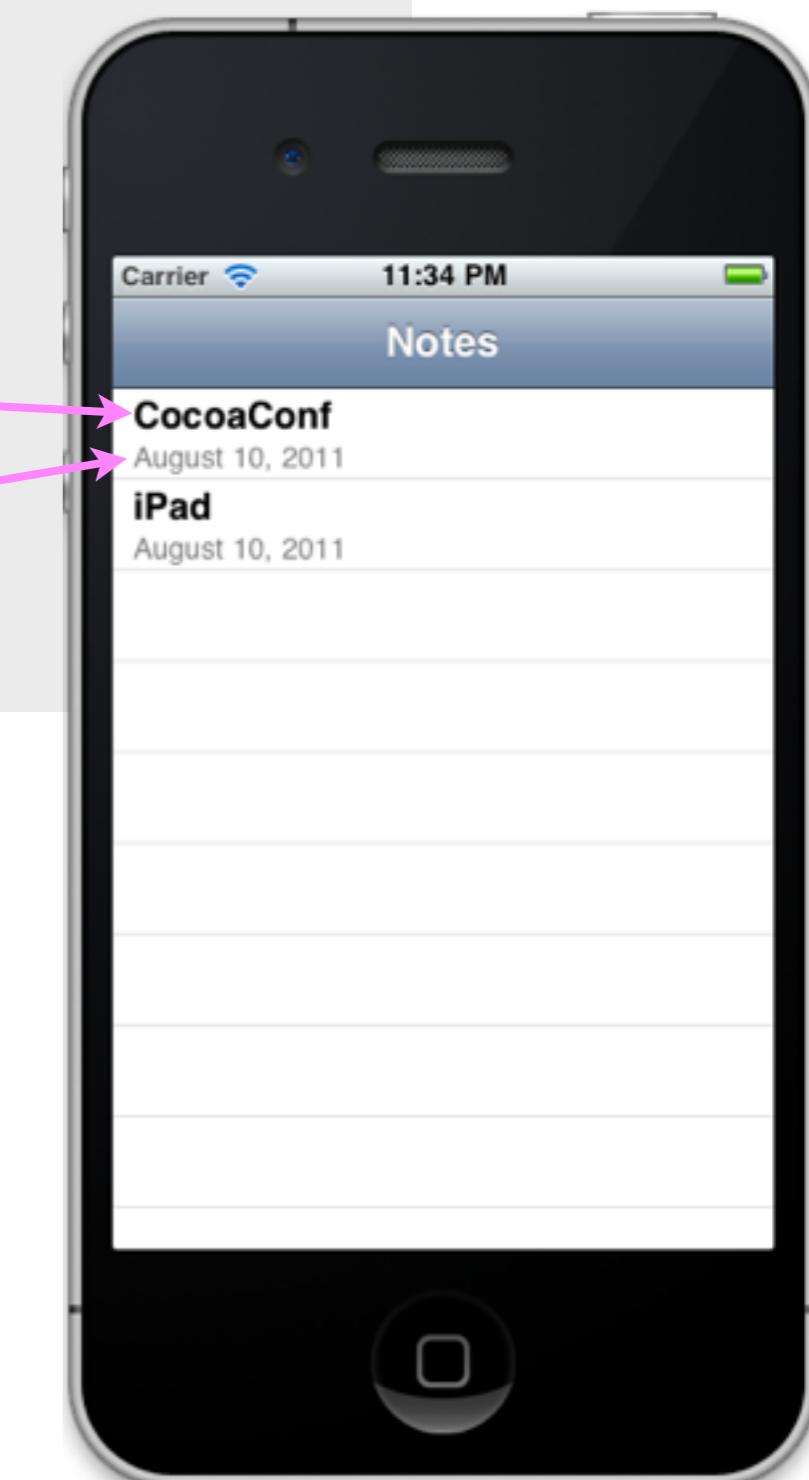
@end
```

```
- (UITableViewCell *)tableView:(UITableView *)tableView cellForRowAtIndexPath:(NSIndexPath *)indexPath
{
    static NSString *CellIdentifier = @"Cell";
    UITableViewCell *cell =
    [tableView dequeueReusableCellWithIdentifier:CellIdentifier];
    if (cell == nil) {
        cell = [[[UITableViewCell alloc] initWithStyle:UITableViewCellStyleSubtitle
                                         reuseIdentifier:CellIdentifier] autorelease];
    }

    // Set up the cell...
    Note *note = [_notes objectAtIndex:indexPath.row];
    cell.textLabel.text = note.name; →

    NSDateFormatter *dateFormat = [[NSDateFormatter alloc] init];
    [dateFormat setDateFormat:@"MMMM d, YYYY"];
    cell.detailTextLabel.text = [dateFormat stringFromDate:note.createdDate]; →
    [dateFormat release];

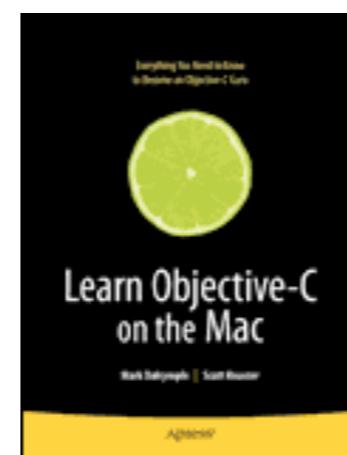
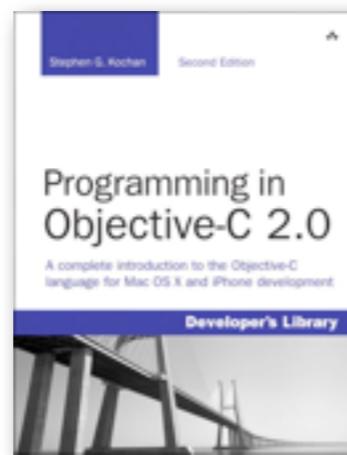
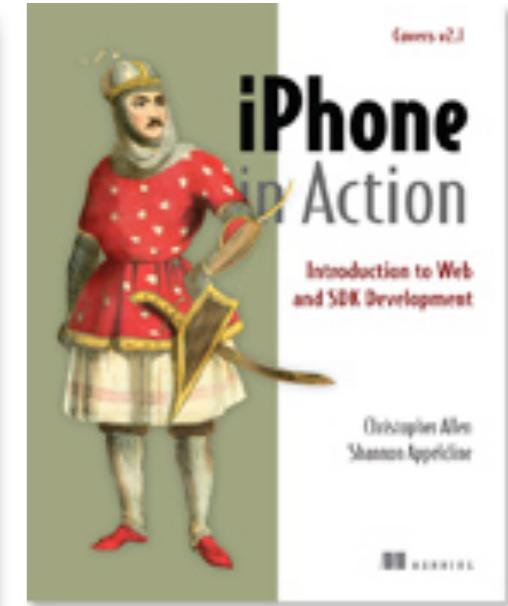
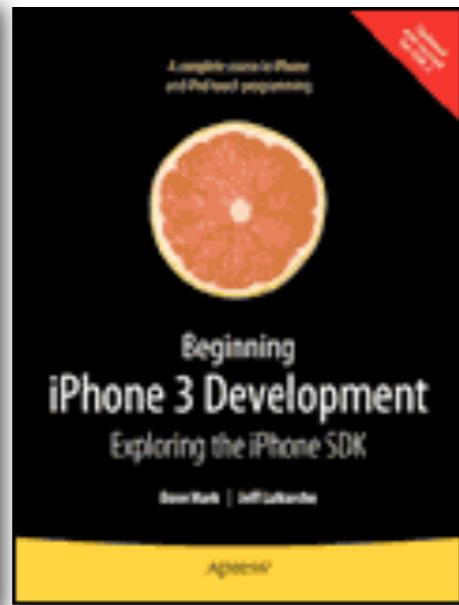
    return cell;
}
```



# *Lab 7*

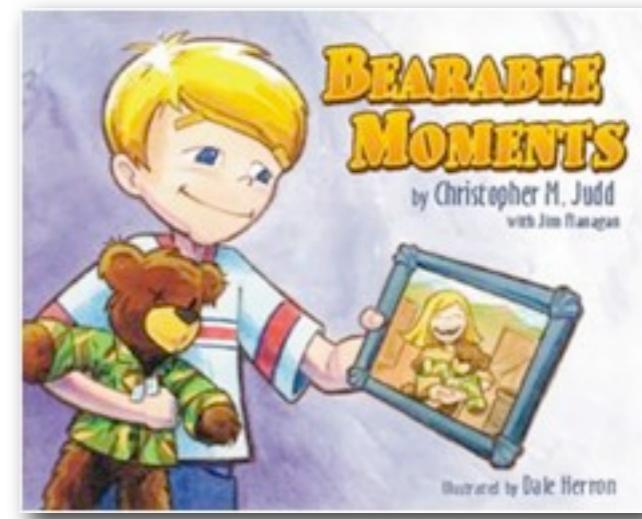
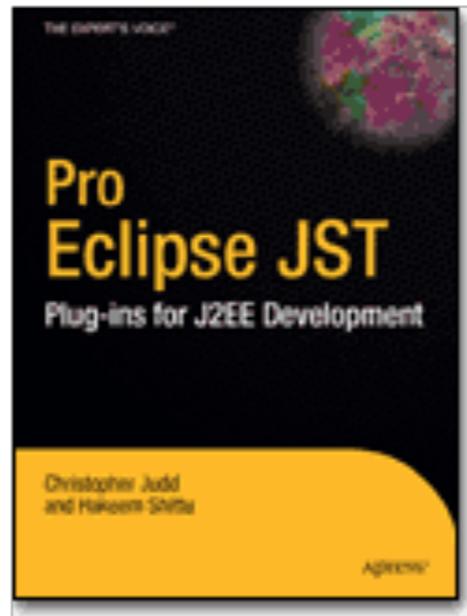
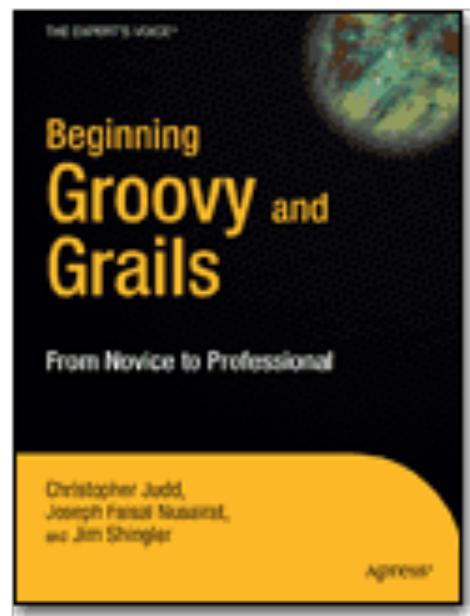
- I. Add a UITableViewController to list all the notes in the database.

# Resources





**Christopher M. Judd**  
**Judd Solutions**  
President/Consultant/Author  
email: [cjudd@juddsolutions.com](mailto:cjudd@juddsolutions.com)  
web: [www.juddsolutions.com](http://www.juddsolutions.com)  
blog: [juddsolutions.blogspot.com](http://juddsolutions.blogspot.com)  
twitter: [javajudd](https://twitter.com/javajudd)



# Attributions



<http://www.iconarchive.com/show/cristal-intense-icons-by-tatice/Notepad-Bloc-notes-2-icon.html>