

iPhone App Dev

Lesson 10

Source Codes

<https://github.com/makzan/ios-dev-course-example>

Contact

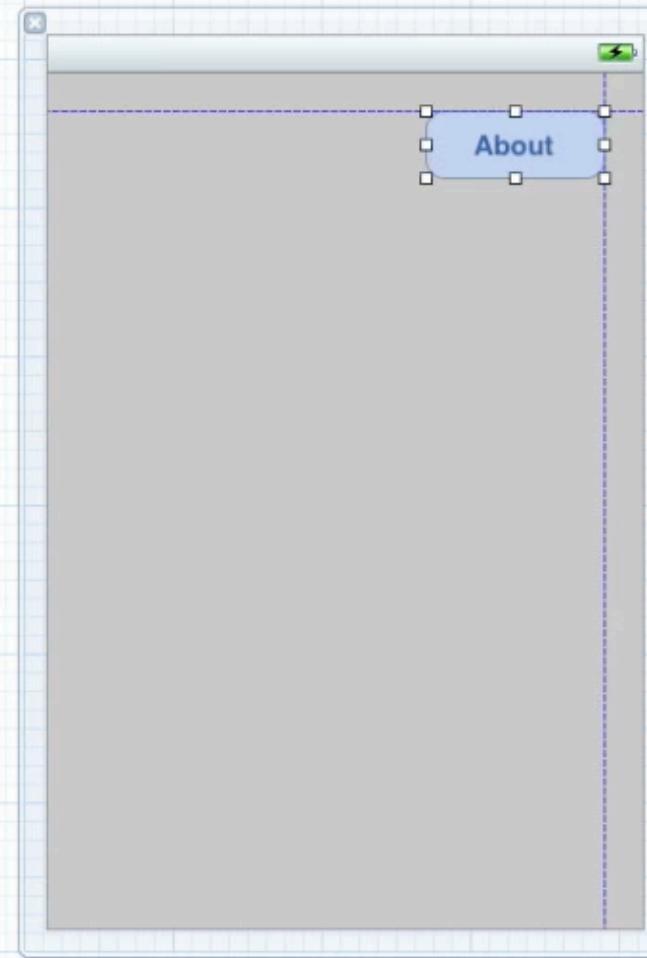
makzan@42games.net

Summary

- Using web view
- Rotation
- Submitting app to App Store
- Handling app approval and rejecting
- Other 3rd parties code worth checking out.
- Last one hour - Multiple Choices Quiz

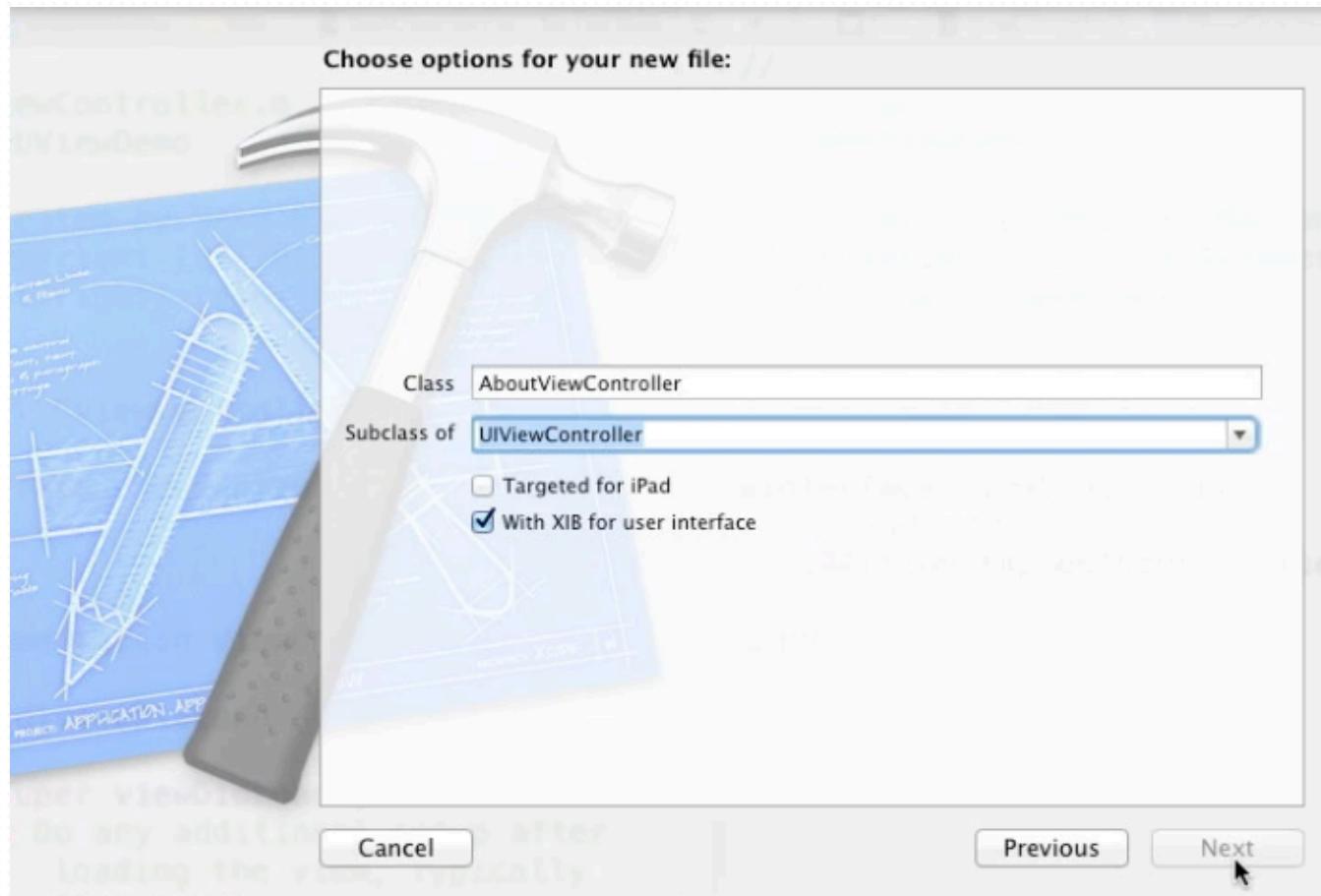
Using UIWebView

Using UIWebView



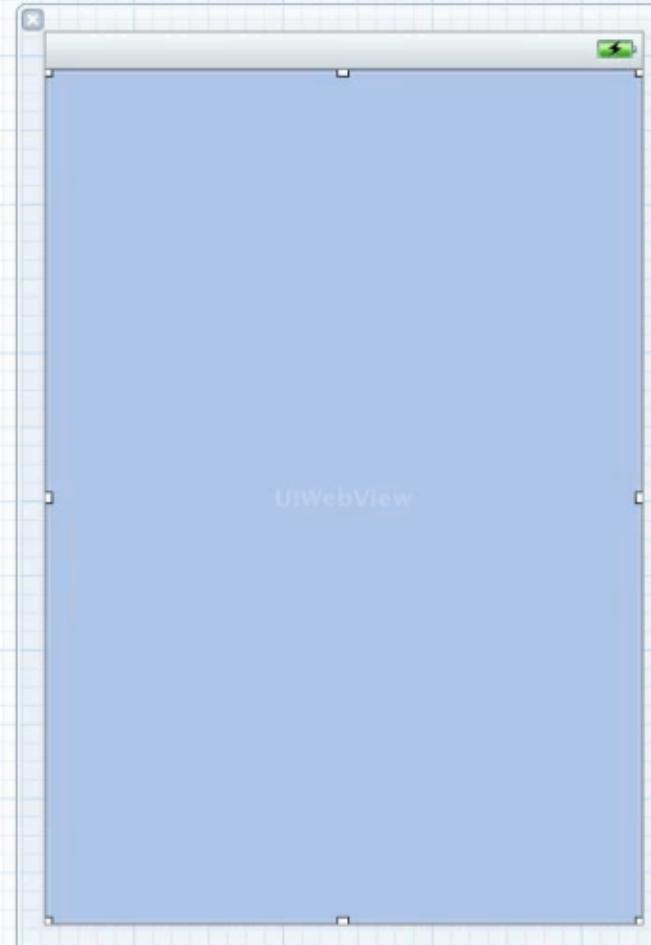
Let's prepare the main view with About button.

Using UIWebView



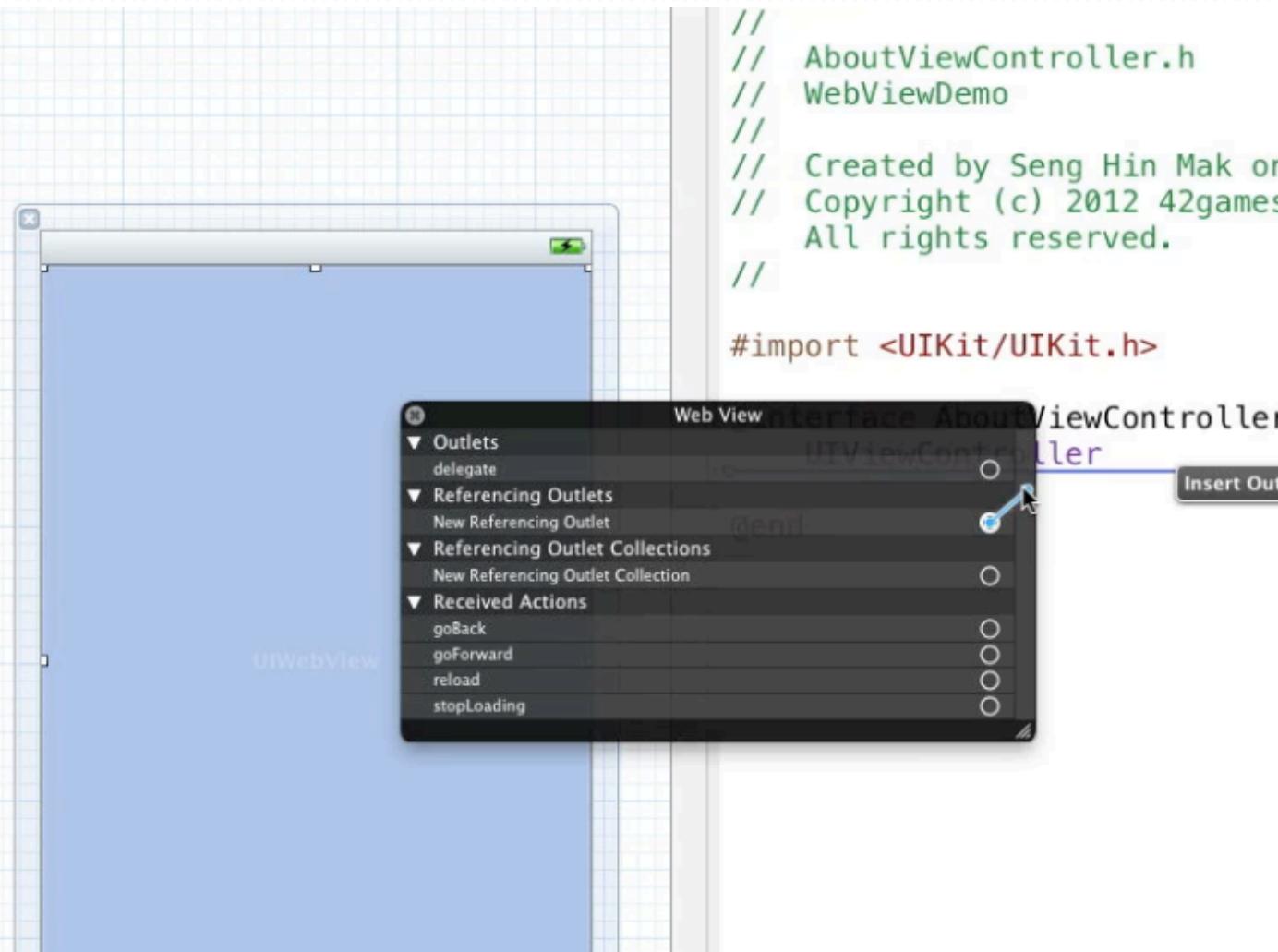
Create an AboutViewController class.

Using UIWebView



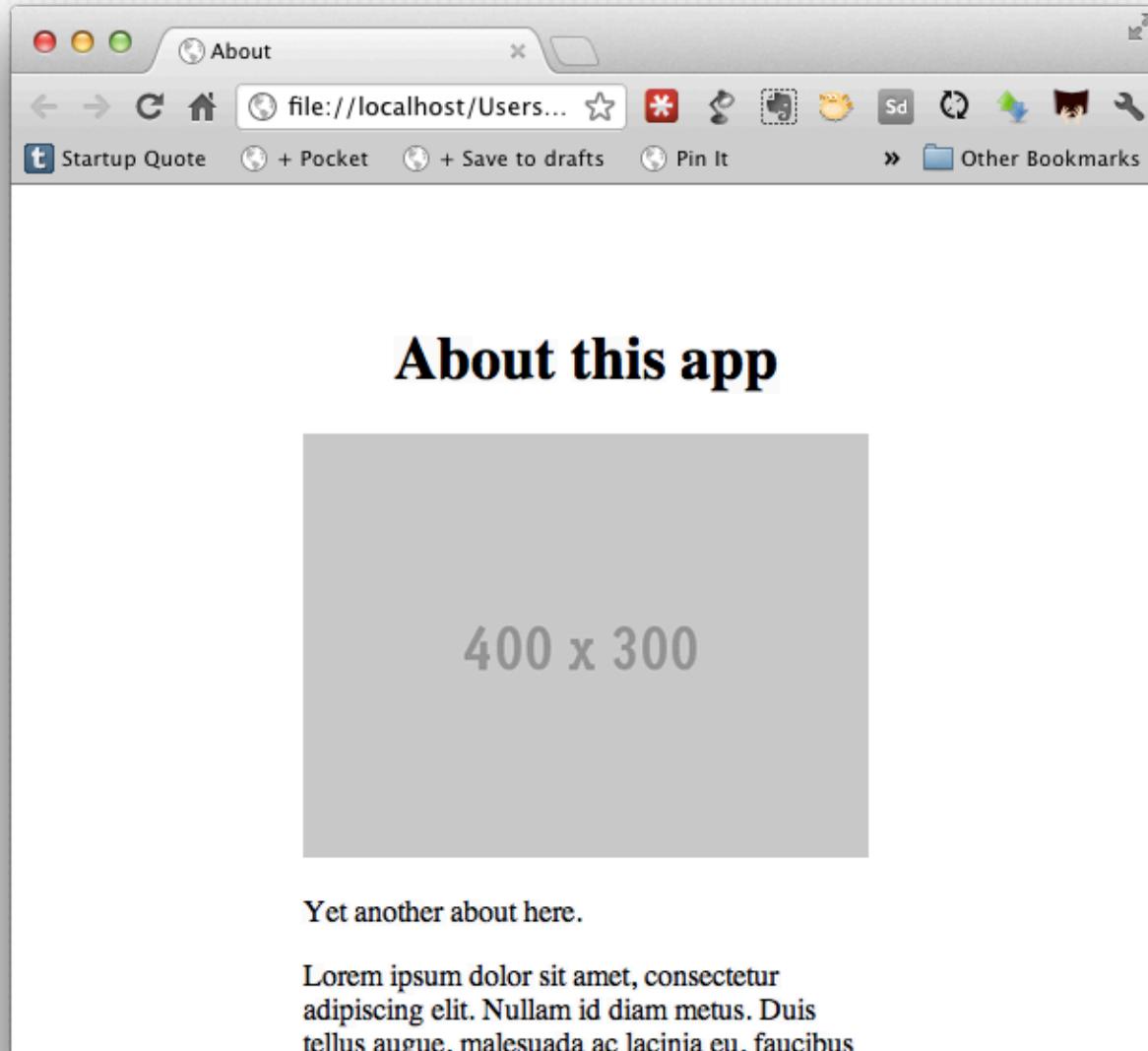
Drag a UIWebView component into the
AboutViewController.

Using UIWebView



Connect the UIWebView with IBOulet

Using UIWebView



Beforehand, we prepare a web page named `about.html`

Using UIWebView

AboutViewController.m

```
- (void)viewDidLoad
{
    [super viewDidLoad];
    // Do any additional setup after loading the view from its nib.

    NSURL *aboutURL = [NSURL fileURLWithPath:[[NSBundle mainBundle]
        pathForResource:@"about" ofType:@"html"]isDirectory:NO];

    [self.webview loadRequest:[NSURLRequest requestWithURL:aboutURL]];
}
```

Load the about html into the web view.

Using UIWebView

ViewController.m

```
#import "ViewController.h"
#import "AboutViewController.h"

- (IBAction)tappedAbout:(id)sender {
    AboutViewController *aboutVC = [[[AboutViewController alloc] init]
        autorelease];
    [self presentModalViewController:aboutVC animated:YES];
}
```

First, let's identify the table view in interface builder.

Using UIWebView

The result with web view
showing about.html



Using UIWebView

We can also load external website.



Using UIWebView

What's more?

We can actually communicate between
Objective-C and UIWebView JavaScript.

<https://github.com/ochameau/NativeBridge>

Auto Rotation

- Use the following code In each View Controller

```
- (BOOL)shouldAutorotateToInterfaceOrientation:  
    (UIInterfaceOrientation)interfaceOrientation  
{  
    return (interfaceOrientation == UIInterfaceOrientationPortrait);  
}
```

Return the orientation value that you want it to rotate.

Rotation

ViewController.m

```
- (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
{
    return (interfaceOrientation != UIInterfaceOrientationPortraitUpsideDown);
}
```

To control rotation, we use the above delegate method.

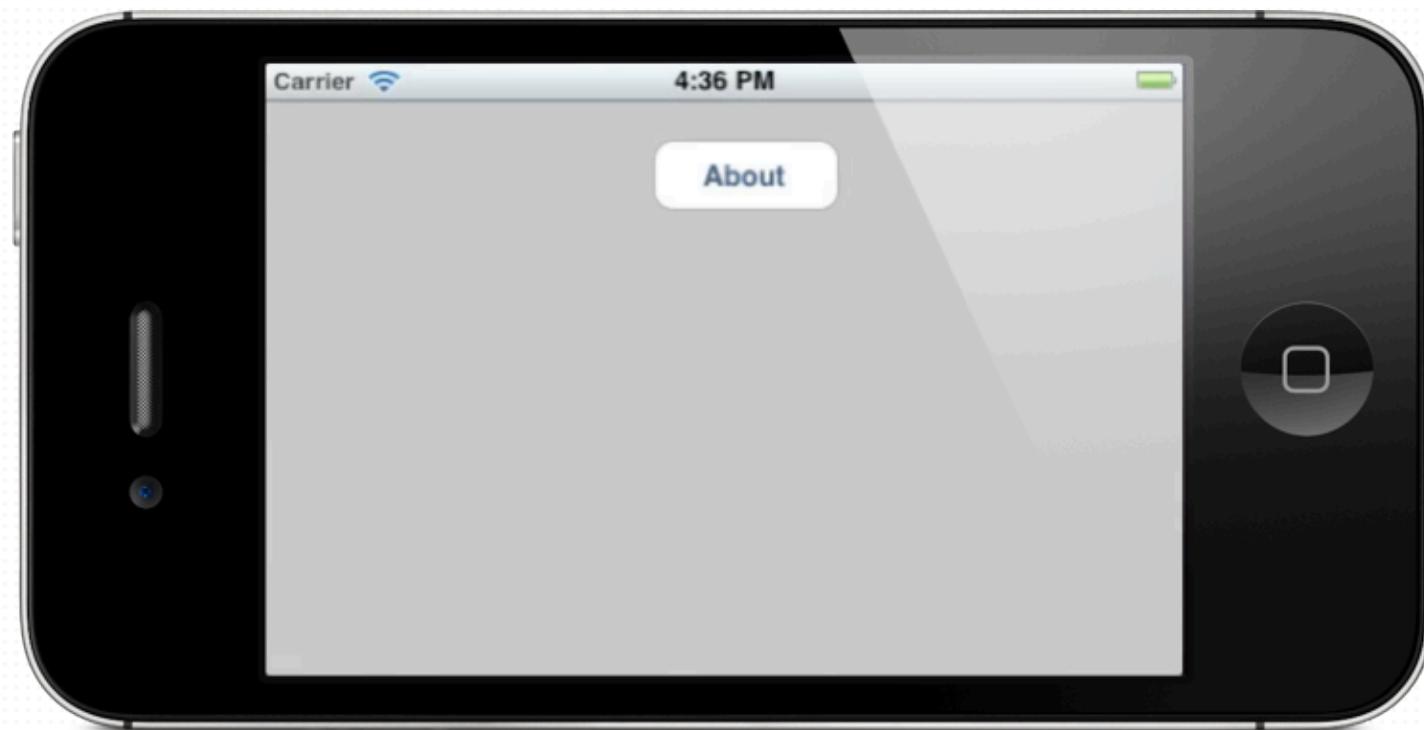
Rotation

ViewController.m

```
- (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)interfaceOrientation
{
    return YES;
//    return (interfaceOrientation != UIInterfaceOrientationPortraitUpsideDown);
}
```

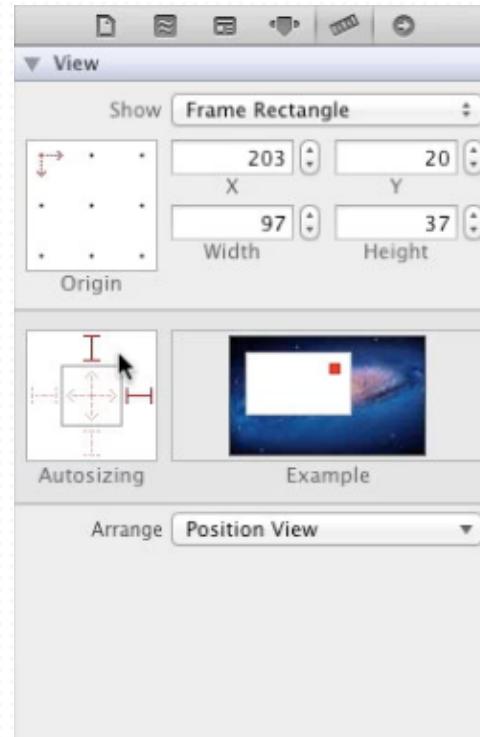
Please note that this rotation setting is individual and you need to make sure each view controller has its correct value.

Rotation



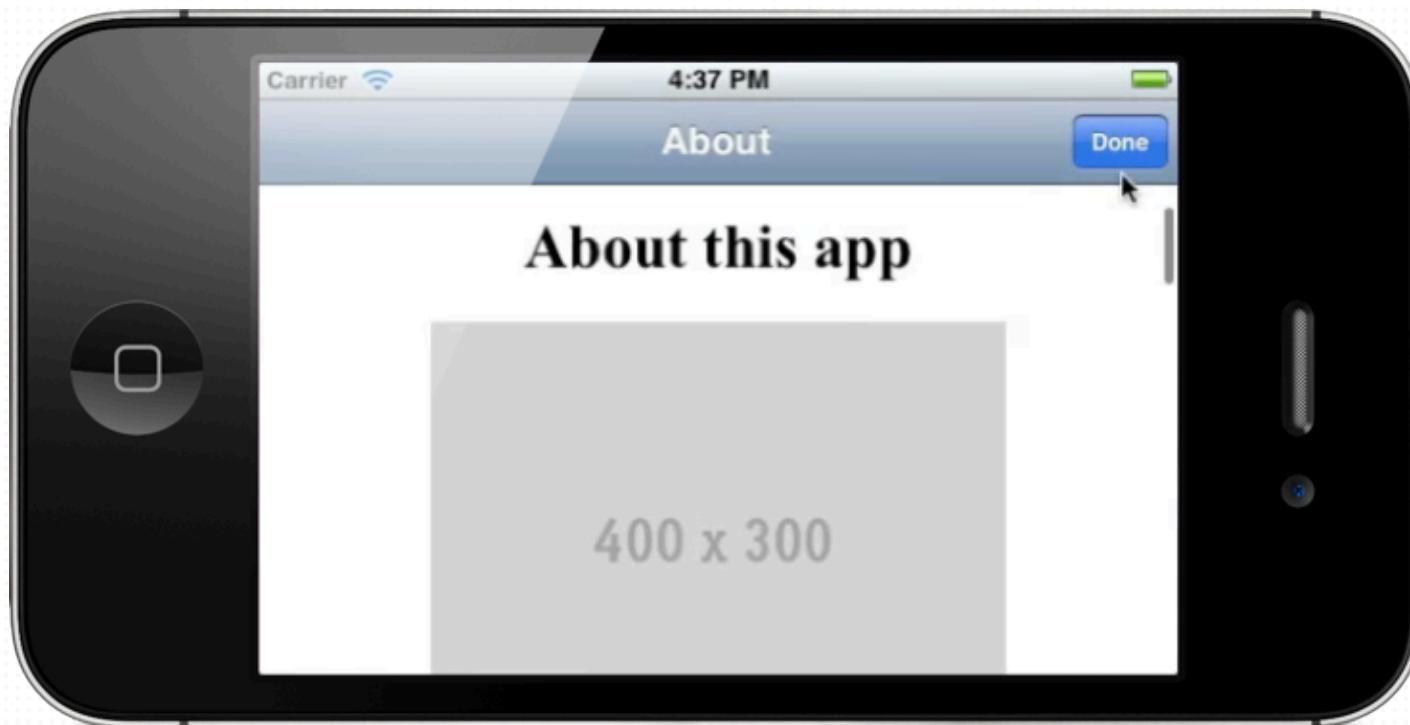
The result with device in landscape.

Rotation



We can control how the UI elements flows when the rotation changes in the interface builder.

Rotation

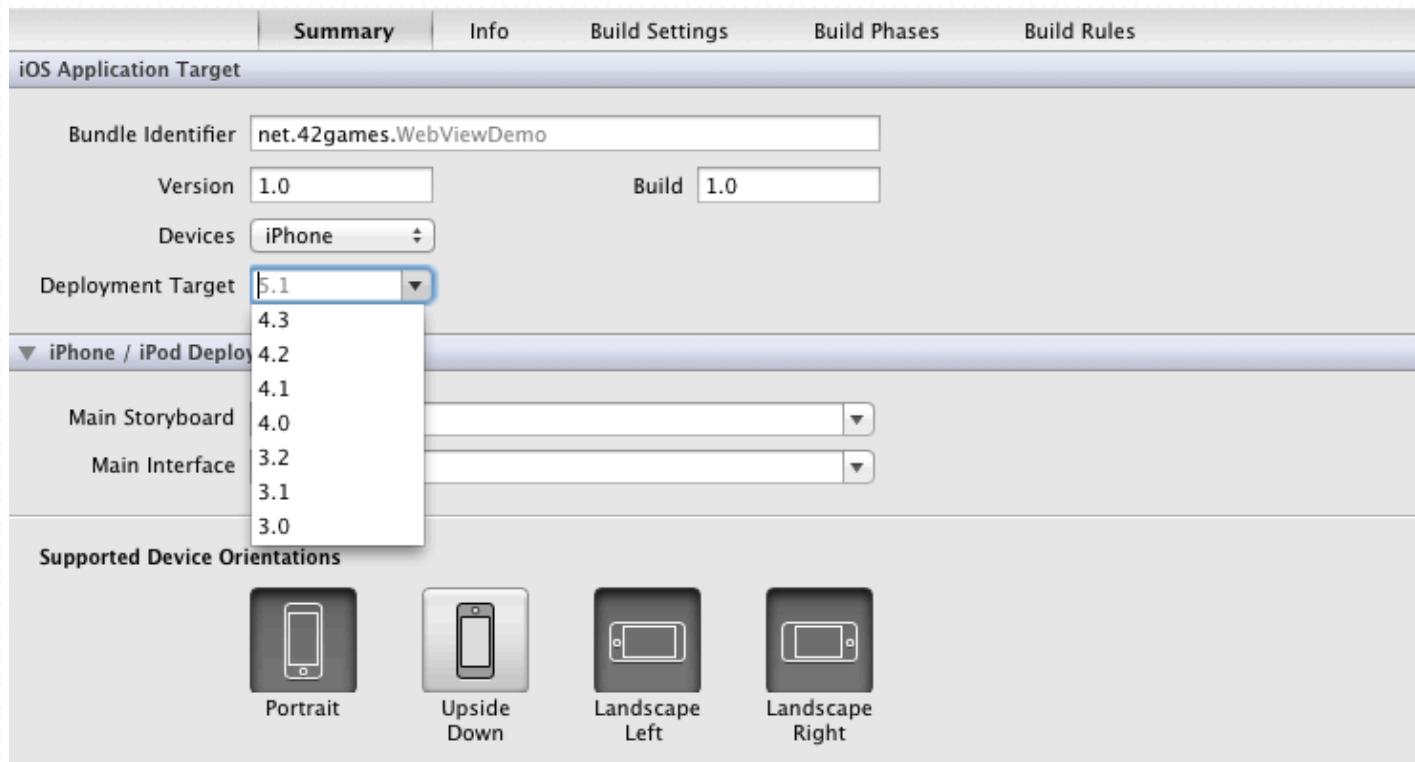


And a result with our above web view in landscape.

Before Submitting

- Ensure the Deployment Target is correct
- Ensure the Icon files are prepared.
 - iPhone/iPod Touch
 - 57x57, 114x114
 - iPad
 - 72x72, 144x144

Before Submitting



Deployment Target means the lowest iOS version this app is able to run.

Applying Developer Acc.

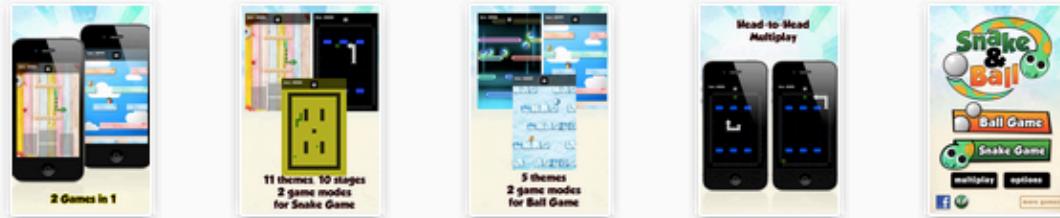
- Apply a free developer account at:
- <https://developer.apple.com/>
- Enroll in the iOS Developer Program:
- <https://developer.apple.com/programs/ios/>

Applying Developer Acc.

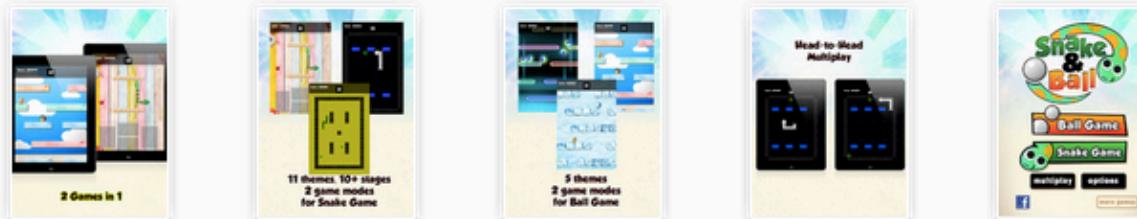
- A step by step guide:
- <http://www.flynax.com/iphone-ipod-app/create-apple-account-instruction.html>

Filling in App Information

iPhone and iPod touch Screenshots



iPad Screenshots



You need decide 5 screenshots and their orders.

Filling in App Information

Select the availability date and price tier for your app.

Availability Date [?](#)

Price Tier [?](#)
[View Pricing Matrix ▶](#)

Price Tier Effective Date [?](#)

Price Tier End Date [?](#)

Price Tier Schedule

Price Tier	Price Effective Date	Price End Date
Tier 1	Existing	None

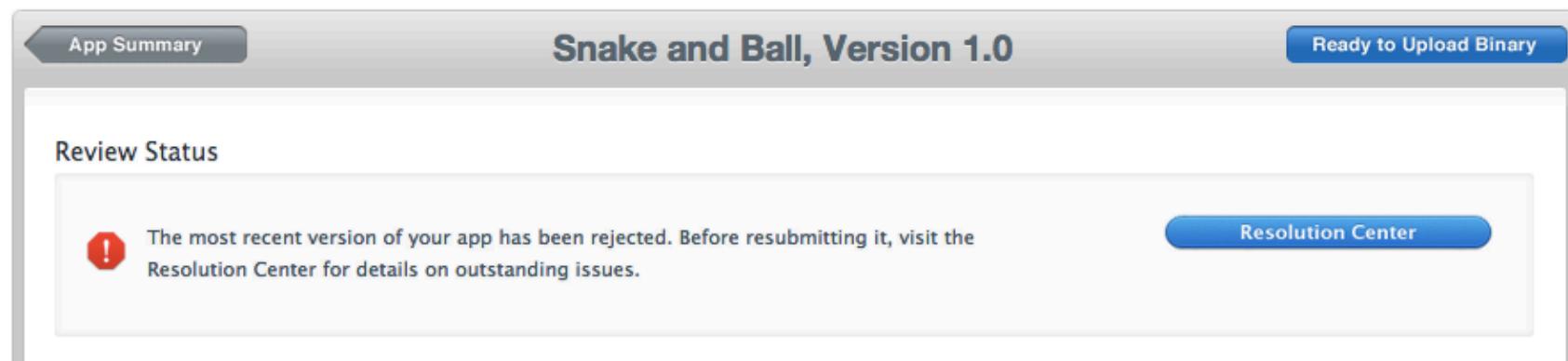
Discount for Educational Institutions [?](#)

Custom B2B App [?](#)

Unless you select [specific stores](#), your app will be for sale in all App Stores worldwide.

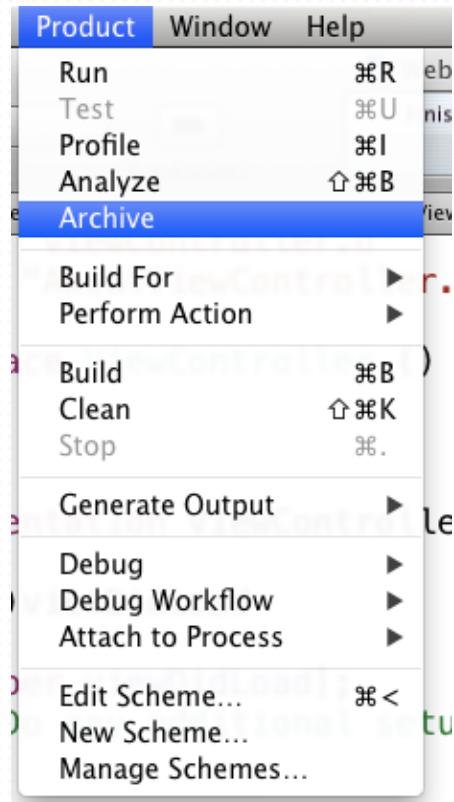
Set the pricing information.

Filling in App Information



Click the Ready to Upload Binary button.

Archive the XCode

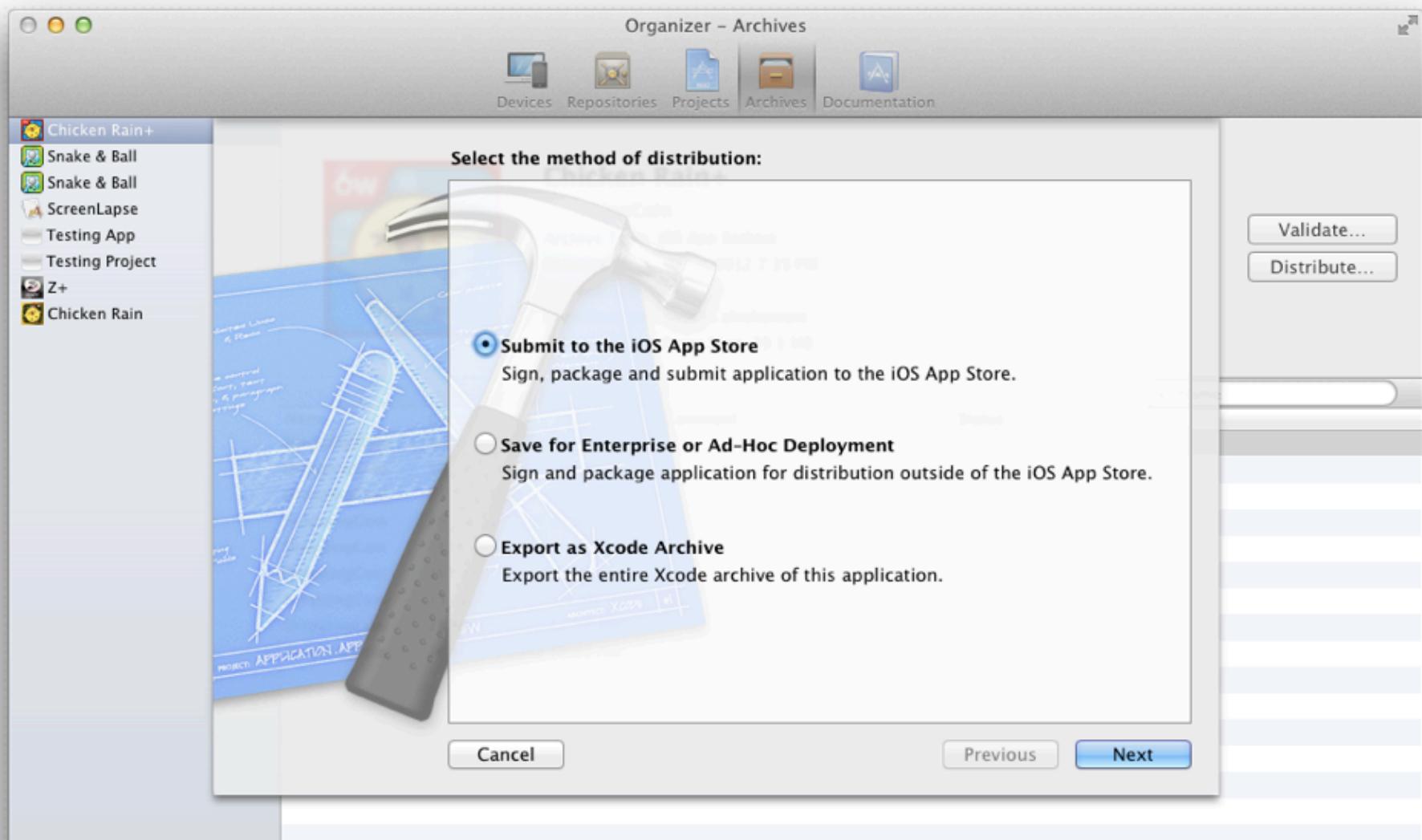


Be sure to select 'device' in build target,
otherwise the Archive button is disabled.

Archive the XCode



Archive the XCode



Archive the XCode

If you are submitting the app,
choose Submit to App Store.

If you are sending out the IPA to beta testers,
choose Save for AdHoc Deployment

Manage Testing Devices

iOS Provisioning Portal

Welcome, Thomas Mak | Edit Profile | Log out

Provisioning Portal : 42games Limited

Home Certificates Devices App IDs Pass Type IDs Provisioning Distribution

Manage History How To

Go to iOS Dev Center

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 47 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"/>	42games iPad	1b462b402b8b0b5fd1ac11c453...	1 (Details)	Edit
<input type="checkbox"/>	BBB's iPad	f0c4e470067f90a736fe1dcaf0...	18 (Details)	Edit
<input type="checkbox"/>	BChai 的 iPhone	9a3c34bf87749c6ef0fdbc97c3...	7 (Details)	Edit

Manage Provisioning

 **Developer** Technologies Resources Programs Support Member Center

iOS Provisioning Portal Welcome, Thomas Mak | Edit Profile | Log out

Provisioning Portal : 42games Limited Go to iOS Dev Center

Development Distribution History How To

 **Development Provisioning Profiles**

<input type="checkbox"/> Provisioning Profile	App ID	Status	Actions
 Alley Oop	DF2G58BF48.net.42games.alleyoo...	Expired	<input type="button" value="Modify"/>
 CP EBook dev1	28KB4BD43Q.net.42games.cp-eboo...	Invalid	<input type="button" value="Modify"/>
 CPak dev1	28KB4BD43Q.net.42games.cpak	Invalid	<input type="button" value="Modify"/>
 ColorPaint_dev1	28KB4BD43Q.net.42games.colorpa...	Expired	<input type="button" value="Modify"/>
 DroppingCoin_dev7	CCBTF7QZNE.net.42games.droppin...	Invalid	<input type="button" value="Modify"/>
 FourJewels	G93PVS94NZ.net.42games.fourjew...	Expired	<input type="button" value="Modify"/>
 Happy Blocks dev1	SQ8P4QP7X3.net.42games.happybl...	Expired	<input type="button" value="Modify"/>
 Lakoo Checkin SDK dev1	SDT4G75397.net.42games.lakoooch...	Expired	<input type="button" value="Modify"/>
 Lakoo Checkin Test Dev1	7W3HB8NRHH.net.42games.lakoo-c...	Expired	<input type="button" value="Modify"/>

Beta Testing

The screenshot shows a web browser window with the TestFlight dashboard for the user 'Thomas Mak'. The dashboard features a dark header bar with the TestFlight logo, a search bar, and various bookmark and account links. Below the header is a navigation menu with 'Dashboard' selected, followed by 'Builds', 'People', and 'Support'. On the right of the menu are 'Upload Build' and 'Add Teammate' buttons. The main content area is titled '42games Dashboard' and includes a promotional box for the 'TestFlight Desktop App'. It also features four main sections: 'Upload Your Builds' (with an icon of a box and arrow), 'Invite Testers' (with an icon of three people), 'Upload API' (with an icon of a power outlet), and 'TestFlight SDK' (with an icon of a toolbox). Each section has a brief description and a green 'Get' button.

The TestFlight Desktop App
Want screaming fast build uploads? Try the TestFlight Desktop App. Download it now »

Upload Your Builds
Upload your .IPA and we'll send your app over-the-air to your testers.
Add a Build

Invite Testers
Create a team for your project and recruit new testers.
Invite Testers

Upload API
Plug into our API for automated IPA uploads.
Get the API

TestFlight SDK
Get insight into your beta testing.
v1.0 now available
Get the SDK

App Approval Status

Date	User	Status
July 04, 2012 12:04	Apple	Rejected
July 04, 2012 08:58	Apple	In Review
June 25, 2012 14:44	Apple	Waiting For Review
June 25, 2012 14:42	Apple	Upload Received
June 25, 2012 14:09	steven+dev1@mz-lab.com	Waiting For Upload
June 07, 2012 16:51	steven+dev1@mz-lab.com	Developer Rejected
June 07, 2012 16:50	Apple	Waiting For Review
June 07, 2012 16:47	Apple	Upload Received
June 07, 2012 10:20	steven+dev1@mz-lab.com	Waiting For Upload
May 31, 2012 17:42	steven+dev1@mz-lab.com	Prepare for Upload

Normally it takes 1-2 weeks to get respond from Apple

Handle AppStore Rejection

App Summary Resolution Center

Use the Resolution Center to correspond with App Review until all issues with your app version have been resolved.

Binary Rejected Jul 4, 2012 12:04 PM
Reasons for Rejection:

Jul 4, 2012 12:04 PM. From Apple.

We found that while your app offers In-App Purchase(s) that can be restored, it does not include the required "Restore" feature to allow users to restore the previously purchased In-App Purchase(s), as specified in Restoring Transactions section of the [In-App Purchase Programming Guide](#):

"...if your application supports product types that must be restorable, you must include an interface that allows users to restore these purchases. This interface allows a user to add the product to other devices or, if the original device was wiped, to restore the transaction on the original device."

To restore previously purchased In-App Purchase products, it would be appropriate to provide a "Restore" button and initiate the restore process when the "Restore" button is tapped by the user.

For more information about restoring transactions and verifying store receipt, please refer to the [In-App Purchase Programming Guide](#).

Reply
Use the field below to ask questions or provide additional information to the App Review team.

Common AppStore Rejections

- The app crashes at certain place and caught by Apple Staffs.
- The app contains information or links to external online payment.
- The app uses private APIs.
- The app violate the background service rules.

In-App Purchases

- Several types
 - Consumable - virtual coins, healing drinks
 - Non-Consumable - level unlock
 - Auto Renewable Subscription - Magazine, Newspaper
 - NonRenewing Subscription - Magazine, Newspaper with manual subscription extension.

Common AppStore Rejections

- Related to In-App Purchases
 - Using virtual currency in In-App Purchase to buy Non-Consumable things.
 - Using In-App Purchase to buy real products
 - Unable to restore purchased item if user changes device or restores device.

**App Approved?
This is just the beginning.**

After App Approval

- Build an app website.
- Start marketing.
- Prepare next app update.

Showcase of App Sites

The screenshot shows a web browser window displaying the [App Sites](http://www.appspot.com) website. The title bar reads "App Sites - Showcase of great iPhone, iPad & Mac app websites". The address bar shows the URL "www.appspot.com". The browser toolbar includes standard icons for back, forward, search, and bookmarks, along with specific buttons for "Startup Quote", "Pocket", "Save to drafts", "Pin It", "Preview and Edit Style", "Extract Style Sheets", "Preview Page", "WatchList", and "Other Bookmarks".

The main content area features the "APP SITES" logo. Below it is a sub-headline: "App Sites is a showcase of beautiful iPhone, iPad & Mac app websites. Stay up to date by following us on Twitter or by subscribing to the RSS feed." A navigation bar below the logo includes links for "All sites", "iPhone", "iPad", and "Mac", along with social sharing icons for Google+, Twitter, and RSS.

The page displays a grid of six mobile application screenshots:

- Wthr**: A weather application showing a smartphone screen with a circular dial interface and the text "Sunny & 80°".
- Just Landed**: An airport pickup service application showing a smartphone screen with flight information (IAH to CLE) and a button to "Get the App!".
- Gab**: A social networking application featuring a smartphone displaying a news feed with various posts and the text "GAB REINVENTS THE WAY WE USE FACEBOOK. IT MAKES SENSE OF THE WORLD'S LARGEST SOCIAL NETWORK".
- Say Hello to Transit**: A transit application showing a smartphone screen with a bus schedule for route 506 and a map.
- tiny**: A dating application showing a smartphone screen with the text "The easiest way to pick a date with friends and family".
- SPELLTOWER**: A word game application showing a smartphone screen with the word "LUNCH" highlighted and other options like "MEETING" and "PLAY TENNIS".

Other 3rd party codes worth check out

- cocos2D for game
- <http://cocos2d-iphone.org/>
- iOS Boilerplate Project
- <http://iosboilerplate.com/>

Other 3rd party codes worth check out

- Custom UI controls
- <http://www.cocoacontrols.com/>
- (Really) Quick TableView Form Generator
- <http://escoz.com/open-source/quickdialog>

Project

Submit it before 4th Aug, 2012

Project

- Develop an iOS app.
- Solve a problem you face.
- **No need** to use all the things from course.
- Less is ~~More~~ Less.

Project

- Send to makzan@42games.net
- If you find me not responding via email, the mail may not be delivered successfully.
- If so, just drop me a message, 6633 5344.

Project

You may encounter different problems.

Just contact me for any questions.

I will try my best to answer you.

The end?

It's the beginning, seriously.