Social Media Apps Programming

jQuery Mobile/Phonegap

Min-Yuh Day, Ph.D.
Assistant Professor
Department of Information Management
Tamkang University

http://mail.tku.edu.tw/myday

2013-11-28
<table>
<thead>
<tr>
<th>Week</th>
<th>Date</th>
<th>Subject/Topics</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2013/09/19</td>
<td>Mid-Autumn Festival (Day off)</td>
</tr>
<tr>
<td>2</td>
<td>2013/09/26</td>
<td>Course Orientation and Introduction to Social Media and Mobile Apps Programming</td>
</tr>
<tr>
<td>3</td>
<td>2013/10/03</td>
<td>Introduction to Android / iOS Apps Programming</td>
</tr>
<tr>
<td>4</td>
<td>2013/10/10</td>
<td>Double Tenth Day (Day off)</td>
</tr>
<tr>
<td>5</td>
<td>2013/10/17</td>
<td>Developing Android Native Apps with Java (Eclipse) (MIT App Inventor)</td>
</tr>
</tbody>
</table>
| 6    | 2013/10/24 | Developing iPhone / iPad Native Apps with Objective-C (Xcode) }
<table>
<thead>
<tr>
<th>Week</th>
<th>Date</th>
<th>Subject/Topics</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>2013/10/31</td>
<td>Mobile Apps using HTML5/CSS3/JavaScript</td>
</tr>
<tr>
<td>8</td>
<td>2013/11/07</td>
<td>jQuery Mobile</td>
</tr>
<tr>
<td>9</td>
<td>2013/11/14</td>
<td>Create Hybrid Apps with PhoneGap</td>
</tr>
<tr>
<td>10</td>
<td>2013/11/21</td>
<td>Midterm Exam Week (Midterm Project Report)</td>
</tr>
<tr>
<td>11</td>
<td>2013/11/28</td>
<td>jQuery Mobile/Phonegap</td>
</tr>
<tr>
<td>12</td>
<td>2013/12/05</td>
<td>Invited Talk: Social, Mobile and Business Model in PIXNET [Invited Speaker: Dr. Rick Cheng-Yu Lu]</td>
</tr>
</tbody>
</table>
## Course Schedule (3/3)

<table>
<thead>
<tr>
<th>Week</th>
<th>Date</th>
<th>Subject/Topics</th>
</tr>
</thead>
<tbody>
<tr>
<td>13</td>
<td>2013/12/12</td>
<td>Google App Engine and Google Map API</td>
</tr>
<tr>
<td>14</td>
<td>2013/12/19</td>
<td>Facebook API (Facebook JavaScript SDK) (Integrate Facebook with iOS/Android Apps)</td>
</tr>
<tr>
<td>15</td>
<td>2013/12/26</td>
<td>Twitter API</td>
</tr>
<tr>
<td>16</td>
<td>2014/01/02</td>
<td>Case Study on Social Media Apps Programming and Marketing in Google Play and App Store</td>
</tr>
<tr>
<td>17</td>
<td>2014/01/09</td>
<td>Final Project Presentation</td>
</tr>
<tr>
<td>18</td>
<td>2014/01/16</td>
<td>Final Exam Week (Final Project Report)</td>
</tr>
</tbody>
</table>
Outline

• jQuery Mobile/PhoneGap
  – HTML5, CSS3, JavaScript
  – jQuery Mobile
  – PhoneGap
  – PhoneGap Build
    • index.html, css, js, png
    • config.xml

• Demo
  – Create Hybrid Apps with PhoneGap
  – Testing iOS and Android Devices
Android /iOS Apps Programming

Native Apps

Hybrid Apps

Mobile Web Apps
PhoneGap, jQuery Mobile, jQuery

jQuery Mobile
(User Interface)

jQuery
(Logic and Computing)

PhoneGap
(Hardware Interface)
PhoneGap App Architecture

HTML5/CSS3 Application

UI Framework e.g. jQueryMobile

PhoneGap API

Phone Gap Bridge

Camera

GPS

SQLite

SQLite

File System

Accelerometer

Compass

etc

PhoneGap API

API Reference

Accelerometer
Tap into the device's motion sensor.

Camera
Capture a photo using the device's camera.

Capture
Capture media files using device's media capture applications.

Compass
Obtain the direction that the device is pointing.

Connection
Quickly check the network state, and cellular network information.

Contacts
Work with the devices contact database.

Device
Gather device specific information.

Events
Hook into native events through JavaScript.

File
Hook into native file system through JavaScript.

Geolocation
Make your application location aware.

Globalization
Enable representation of objects specific to a locale.

InAppBrowser
Launch URLs in another In-app browser instance.

Media
Record and play back audio files.

Notification
Visual, audible, and tactile device notifications.

Splashscreen
Show and hide the application splash screen.

http://docs.phonegap.com/en/2.9.0/index.html
PhoneGap API: Camera

Quick Example

Take a photo and retrieve it as a base64-encoded image:

```javascript
navigator.camera.getPicture(onSuccess, onFail, { quality: 50, destinationType: Camera.DestinationType.DATA_URL });

function onSuccess(imageData) {
  var image = document.getElementById('myImage');
  image.src = 'data:image/jpeg;base64,' + imageData;
}

function onFail(message) {
  alert('Failed because: ' + message);
}
```

Take a photo and retrieve the image's file location:

```javascript
navigator.camera.getPicture(onSuccess, onFail, { quality: 50, destinationType: Camera.DestinationType.FILE_URI });

function onSuccess(imageURI) {
  var image = document.getElementById('myImage');
  image.src = imageURI;
}

function onFail(message) {
  alert('Failed because: ' + message);
}
```

Full Example

```
<!DOCTYPE html>
```

http://docs.phonegap.com/en/2.9.0/cordova_camera_camera.md.html#Camera
PhoneGap API: Geolocation

Quick Example

```javascript
// onSuccess Callback
// This method accepts a Position object, which contains the
// current GPS coordinates

var onSuccess = function(position) {
  alert('Latitude: ' + position.coords.latitude + '
' + 'Longitude: ' + position.coords.longitude + '
' + 'Altitude: ' + position.coords.altitude + '
' + 'Accuracy: ' + position.coords.accuracy + '
' + 'Altitude Accuracy: ' + position.coords.altitudeAccuracy + '
' + 'Heading: ' + position.coords.heading + '
' + 'Speed: ' + position.coords.speed + '
' + 'Timestamp: ' + position.timestamp + '
');
};

// onError Callback receives a PositionError object

function onError(error) {
  alert('code: ' + error.code + '
' + 'message: ' + error.message + '
');
}

navigator.geolocation.getCurrentPosition(onSuccess, onError);
```

Full Example

```html
<!DOCTYPE html>
<html>
<head>
  <title>Device Properties Example</title>
  <script type="text/javascript" charset="utf-8" src="cordova-x.x.x.js"></script>
  <script type="text/javascript" charset="utf-8">(function() {
    // Wait for device API libraries to load
    // document.addEventListener("deviceready", onDeviceReady, false);
  })();
</head>
```

http://docs.phonegap.com/en/2.9.0/cordova_geolocation_geolocation.md.html#Geolocation
PhoneGap API: Storage

Quick Example

```javascript
var db = window.openDatabase("test", "1.0", "Test DB", 1000000);
```

Full Example

```html
type="text/javascript" charset="utf-8" src="cordova-x.x.x.js"></script>

// Wait for device API libraries to load
//
document.addEventListener("deviceready", onDeviceReady, false);

// device APIs are available
//
function onDeviceReady()
{
    var db = window.openDatabase("test", "1.0", "Test DB", 1000000);
}

</script>
</head>
<body>
<h1>Example</h1>
<p>Open Database</p>
</body>
</html>
```

http://docs.phonegap.com/en/2.9.0/cordova_storage_storage.md.html#Storage
<html>
<head>
    <title>Storage Example</title>
    <script type="text/javascript" charset="utf-8" src="cordova-x.x.x.js"></script>
    <script type="text/javascript" charset="utf-8">

    // Wait for device API libraries to load
    //
    document.addEventListener("deviceready", onDeviceReady, false);

    // device APIs are available
    //
    function onDeviceReady() {
        var db = window.openDatabase("Database", "1.0", "Cordova Demo", 200000);
        db.transaction(populateDB, errorCB, successCB);
    }

    // Populate the database
    //
    function populateDB(tx) {
        tx.execSQL('DROP TABLE IF EXISTS DEMO');
        tx.execSQL('CREATE TABLE IF NOT EXISTS DEMO (id unique, data)');
        tx.execSQL('INSERT INTO DEMO (id, data) VALUES (1, "First row")');
        tx.execSQL('INSERT INTO DEMO (id, data) VALUES (2, "Second row")');
    }

    // Transaction error callback
    //
    function errorCB(tx, err) {
        alert("Error processing SQL: "+err);
    }

    // Transaction success callback
    //
    function successCB() {
        alert("success!");
    }

    </script>
</head>
<body>
    <h1>Example</h1>
    <p>Database</p>
</body>
</html>
iOS Certificate (p12) file
Provisioning profile
iOS Certificate (p12) file
Provisioning profile
MydayMobileApp.ipa

Myday Mobile App
Myday Mobile App for Prof. Min-Yuh Day, Department of Information Management, Tamkang University

iOS Myday Mobile App Rebuild ipa

No key selected

3.2/3.4 MB, 7 secs left
App Distribution on App Store

About App Distribution

Note: This guide describes Xcode 5. If you're using Xcode 4, read App Distribution Guide for Xcode 4 in the Retired Documents Library.

This guide contains everything you need to know to distribute an app through the App Store or Mac App Store.

- Get step-by-step guidance for enrolling in an Apple Developer Program and building, testing, and submitting your app.
- Configure technologies that are available only to apps submitted to the App Store or Mac App Store.
- Test your app on multiple devices and system versions, or offer testers a preview of your next release.
- Upload metadata about your app so the store can present it to customers.
- Verify that you've prepared your app correctly, and submit it to the store.
- Learn how to release and maintain your app after submission.
Test on iOS Devices (iPhone/iPAD)

Testing iOS App

- Provisioning Profile
- Developer Portal
- Code Signing
- Bundle ID
- Keychain Access
- App ID
- Certificates
- Xcode Organizer

Source: Provisioning Profile Overview for Non-Programmer [http://www.youtube.com/watch?v=hpzej8m_LII](http://www.youtube.com/watch?v=hpzej8m_LII)
Basics of iOS Device Testing

1. You have PERMISSION

2. Your App has PERMISSION

3. Your Device has PERMISSION

Source: Provisioning Profile Overview for Non-Programmer [http://www.youtube.com/watch?v=hpzej8m_LII](http://www.youtube.com/watch?v=hpzej8m_LII)
You have PERMISSION

Source: Provisioning Profile Overview for Non-Programmer  
http://www.youtube.com/watch?v=hpzej8m_L1I
You have PERMISSION

1

Keychain Access

Certificates

Source: Provisioning Profile Overview for Non-Programmer  http://www.youtube.com/watch?v=hpzej8m_L1I
Your App has PERMISSION

com.myDomain.myApp

Source: Provisioning Profile Overview for Non-Programmer  http://www.youtube.com/watch?v=hpzej8m_LlI
3 Your Device has PERMISSION

Source: Provisioning Profile Overview for Non-Programmer  http://www.youtube.com/watch?v=hpzej8m_L1l
Your Device has PERMISSION

Getting Started with iOS Devices

To install apps and pre-release versions of iOS on a development device, you must first register that device using its unique device identifier (UDID).

Register Devices with Xcode
Xcode is the easiest way to register your iOS device. Connect your device to your Mac and click Use for Development in Xcode's Organizer window. Sign in with the Apple ID associated with your iOS Developer Program membership and Xcode will register the UDID of your device to your iOS development team.

Register Devices Manually
If you wish to upload multiple device IDs, team agents and admins can manually register devices.

Source: Provisioning Profile Overview for Non-Programmer [http://www.youtube.com/watch?v=hpzej8m_LII](http://www.youtube.com/watch?v=hpzej8m_LII)
Demo
Create Hybrid Apps with PhoneGap
Testing iOS and Android Devices
https://build.phonegap.com/
PhoneGap Build

Source: https://build.phonegap.com/
Basics of iOS Device Testing

1. **You** have PERMISSION

2. **Your App** has PERMISSION

3. **Your Device** has PERMISSION

Source: Provisioning Profile Overview for Non-Programmer  [http://www.youtube.com/watch?v=hpzej8m_LII](http://www.youtube.com/watch?v=hpzej8m_LII)
You have PERMISSION
Here’s where it all happens for Apple developers.

iOS Developer Program
Get the resources you need to develop and distribute apps for iPad, iPhone, and iPod touch.

https://developer.apple.com/
iOS Developer Program
The fastest path from code to customer.

Enroll Now  $99/year

1. Develop
Develop your application with the iOS SDK and a wealth of technical resources in the iOS Dev Center.
Learn more

2. Test
Test and debug your code on iPad, iPhone and iPod touch to finalize your applications.
Learn more

3. Distribute
Distribute your apps on the App Store and reach millions of iPad, iPhone, and iPod touch users.
Learn more

iOS Developer Program

The fastest path from code to customer.

Enroll Now  $99/year

Enrolling in Apple Developer Programs

Get everything you need to develop and distribute apps for iOS and OS X.

It's easy to get started.

✓ Choose an enrollment type.
  
  Individual: choose this option if you are an individual or sole proprietor/single person business.

  Company/Organization: choose this option if you are a company, non-profit organization, joint venture, partnership, or government organization.

✓ Submit your information.

  Provide basic personal information, including your legal name and address. If you're enrolling as a company/organization, we'll need a few more things, like your legal entity name and D-U-N-S® Number, as part of our verification process.

✓ Purchase and activate your program.

  Once we verify your information, you can purchase your program on the Apple Online Store. After you have completed your purchase, we'll send you an email within 24 hours on how to activate your membership.

Continue
Sign in or create an Apple ID.

You can enroll in the iOS Developer Program or Mac Developer Program with the same Apple ID you use for other services like iCloud and the Apple Online Store. However, if you have an iTunes Connect account for distributing another media type (music, TV, movies, or books) or are enrolled in the iOS Developer Enterprise Program, you need to use a different Apple ID for your enrollment.

Existing Apple ID

Enroll in an Apple Developer Program with the same Apple ID you use for other services like iCloud and the Apple Online Store.

Sign In

New Apple ID

Create a new Apple ID if you have an existing iTunes Connect account, participate in the Volume Purchase Program, are enrolled in the iOS Developer Enterprise Program, or prefer to have an Apple ID dedicated to your business transactions.

Create Apple ID

Are you enrolling as an individual or organization?

**Individual**
Select this option if you are an individual or sole proprietor/single person company.

**Seller Name**
Your personal legal name will be listed as the seller of your apps on the App Store.

Example:
Seller: John Smith

**Individual Development Only**
You are the only one allowed access to program resources.

**You will need:**
- A valid credit card for purchase.
  We may also require additional personal documentation to verify your identity.

---

**Company/Organization**
Select this option if you are a company, non-profit organization, joint venture, partnership, or government organization.

**Seller Name**
Your organization’s legal entity name will be listed as the seller of your apps on the App Store.

Example:
Seller: ABC Company, Inc.

**Development Team**
You can add additional developers to your team who can access program resources. Companies who have hired a contractor to create apps for distribution on the App Store should enroll with their company name and add the contractors to their team.

**You will need:**
- The legal authority to bind your company/organization to Apple Developer Program legal agreements.
- An address for the company’s principal place of business or corporate headquarters.
Individual
Select this option if you are an individual or sole proprietor/single person company.

Seller Name
Your personal legal name will be listed as the seller of your apps on the App Store.
Example:
Seller: John Smith

Individual Development Only
You are the only one allowed access to program resources.

You will need:
- A valid credit card for purchase.
  We may also require additional personal documentation to verify your identity.

Company/Organization
Select this option if you are a company, non-profit organization, joint venture, partnership, or government organization.

Seller Name
Your organization's legal entity name will be listed as the seller of your apps on the App Store.
Example:
Seller: ABC Company, Inc.

Development Team
You can add additional developers to your team who can access program resources. Companies who have hired a contractor to create apps for distribution on the App Store should enroll with their company name and add the contractors to their team.

You will need:
- The legal authority to bind your company/organization to Apple Developer Program legal agreements.
- An address for the company's principal place of business or corporate headquarters.
- A D-U-N-S® Number assigned to a legal entity.
  D-U-N-S Numbers, available from D&B for free in most jurisdictions, are unique nine-digit numbers widely used as standard business identifiers. To learn more, read our FAQs. Before enrolling, check to see if D&B has assigned you a D-U-N-S Number. If not, please request one.
  Note: We do not accept DBAs, Fictitious Business, Trade names, or branches at this time.
- A valid credit card for purchase.
Certificates, Identifiers & Profiles

Certificates, Identifiers & Profiles

Manage your certificates, App IDs, devices, and provisioning profiles.
iOS Developer Program University License Agreement

View the iOS Developer Program University License Agreement PDF.

PLEASE READ THE FOLLOWING AGREEMENT TERMS AND CONDITIONS CAREFULLY BEFORE DOWNLOADING OR USING THE APPLE SOFTWARE. THESE TERMS AND CONDITIONS CONSTITUTE A LEGAL AGREEMENT BETWEEN UNIVERSITY AND APPLE. BY CLICKING ON THE "I AGREE" BUTTON, UNIVERSITY, THROUGH ITS AUTHORIZED LEGAL REPRESENTATIVE, IS agreeing TO BE BOUND BY AND IS BECOMING A PARTY TO THIS AGREEMENT. IF UNIVERSITY DOES NOT OR CANNOT AGREE TO THIS AGREEMENT, THEN CLICK THE "CANCEL" BUTTON. IF UNIVERSITY DOES NOT AGREE TO THIS AGREEMENT, THEN UNIVERSITY IS NOT PERMITTED TO PARTICIPATE IN THIS PROGRAM OR USE THE APPLE SOFTWARE.

iOS Developer Program University Agreement
(for teaching classes that allow students to load applications onto Apple-branded products running the iOS)

By checking this box I confirm that I have read and agree to be bound by the Agreement above. I am agreeing on behalf of my higher education institution, and I represent and warrant that I have legal authority to bind my higher education institution to the terms of the Agreement above. I also confirm that I am of the legal age of majority in the jurisdiction in which I reside (at least 18 years of age in many countries).

BY CLICKING THE "AGREE" BUTTON AND CHECKING THE BOXES FOR EACH AGREEMENT YOU WANT TO ACCEPT, YOU ARE AGREEING TO BE BOUND BY SUCH AGREEMENT(S).

Cancel  I Agree
Certificates, Identifiers & Profiles

[Image of the Apple Developer Member Center webpage highlighting the Certificates, Identifiers & Profiles section]

Get started with your iOS Developer University Program
Learn how to take advantage of your program resources and benefits.

Assigning Team Roles
You can manage team members, assign roles to existing members, or invite others to join your team.

✓ Managing Developer Roles in the Member Center
The Member Center is where you can manage your development team, which can include up to 200 students. Team roles are used to assign certain responsibilities to a developer. While students can be added only as Team Members, faculty and staff are intended to assume the Admin role in order to be able to invite additional team members, approve digital certificates, create provisioning profiles, and more.
Get started in the Member Center

Quick Links to Resources
Member Center
Manage your development team, view legal agreements, and access technical resources.

Certificates, Identifiers & Profiles
Provision & test apps on your iPhone, iPad & iPod touch

Getting Started with iOS Certificates

You will need to set up digital certificates to develop and distribute iOS apps. To install your app on a development device or submit it to the App Store, it must first be signed with an Apple-issued certificate. Certificates allow the system to identify who signed the app.

Request Certificates with Xcode
Xcode is the easiest way to request certificates. Connect your device to your Mac and click Use for Development in Xcode's Organizer window. Sign in with the Apple ID associated with your iOS Developer Program membership and Xcode will automatically generate your certificates.

Request Certificates Manually
To request certificates, upload a Certificate Signing Request from your Mac.

Learn More
App Distribution Guide

What type of certificate do you need?

Development

- **iOS App Development**
  Sign development versions of your iOS app.

  - Apple Push Notification service SSL (Sandbox)
    Establish connectivity between your notification server and the Apple Push Notification service sandbox environment. A separate certificate is required for each app you develop.

Production

- **Pass Type ID Certificate**
  Sign and send updates to passes in Passbook.

- **Website Push ID Certificate**
  Sign and send updates for Websites.
Development

- **iOS App Development**
  Sign development versions of your iOS app.

- **Apple Push Notification service SSL (Sandbox)**
  Establish connectivity between your notification server and the Apple Push Notification service sandbox environment. A separate certificate is required for each app you develop.

Production

- **Pass Type ID Certificate**
  Sign and send updates to passes in Passbook.

- **Website Push ID Certificate**
  Sign and send updates for Websites.

Intermediate Certificates
To use your certificates, you must have the intermediate signing certificate in your system keychain. This is automatically installed by Xcode. However, if you need to reinstall the intermediate signing certificate click the link below:

About Creating a Certificate Signing Request (CSR)

To manually generate a Certificate, you need a Certificate Signing Request (CSR) file from your Mac. To create a CSR file, follow the instructions below to create one using Keychain Access.

Create a CSR file.
In the Applications folder on your Mac, open the Utilities folder and launch Keychain Access.

Within the Keychain Access drop down menu, select Keychain Access > Certificate Assistant > Request a Certificate from a Certificate Authority.
- In the Certificate Information window, enter the following information:
  - In the User Email Address field, enter your email address.
  - In the Common Name field, create a name for your private key (e.g., John Doe Dev Key).
  - The CA Email Address field should be left empty.
  - In the "Request is" group, select the "Saved to disk" option.
- Click Continue within Keychain Access to complete the CSR generating process.
Upload CSR file (CertificateSigningRequest)

Generate your certificate.

With the creation of your CSR, Keychain Access simultaneously generated a public and private key pair. Your private key is stored on your Mac in the login Keychain by default and can be viewed in the Keychain Access application under the "Keys" category. Your requested certificate will be the public half of your key pair.

Upload CSR file.
Select .certSigningRequest file saved on your Mac.

1 **You** have PERMISSION

Keychain Access

Certificates

Source: Provisioning Profile Overview for Non-Programmer  [http://www.youtube.com/watch?v=hpzej8m_LlI](http://www.youtube.com/watch?v=hpzej8m_LlI)
Applications\Utilities\Keychain Access
Request a Certificate From a Certificate Authority
Certificate Assistant

Certificate Information

Enter information for the certificate you are requesting. Click Continue to request a certificate from the CA.

User Email Address: imyday@gmail.com

Common Name: iMyday

CA Email Address:

Request is:
- Emailed to the CA
- Saved to disk
- Let me specify key pair information

Continue
Request a Certificate From a Certificate Authority
Key Pair Information

Specify the key size and algorithm used to create your key pair.

The key pair is made up of your private and public keys. The private key is the secret part of the key pair and should be kept secret. The public key is made publicly available as part of the digital certificate.

Key Size: 2048 bits
Algorithm: RSA

Learn More...

Continue
Your certificate request has been created on disk.

Show In Finder...
CertificateSigningRequest
CSR
To add an iOS certificate, locate the certificate file on your Mac and upload it. Here are the steps:

1. Navigate to the location where you saved the certificate file.
2. Select the certificate file.
3. Click the "Choose File" button to select the certificate file.
4. The selected certificate file will be uploaded and used to create your iOS certificate.

Note: Ensure the certificate file is a valid certificate signing request (.csr) file.
Generate your certificate.

With the creation of your CSR, Keychain Access simultaneously generated a public and private key pair. Your private key is stored on your Mac in the login Keychain by default and can be viewed in the Keychain Access application under the "Keys" category. Your requested certificate will be the public half of your key pair.

Upload CSR file.
Select .certSigningRequest file saved on your Mac.
Generate your certificate.

With the creation of your CSR, Keychain Access will simultaneously generated a public and private key pair. Your private key is stored on your Mac in the login Keychain by default and can be viewed in the Keychain Access application under the "Keys" category. Your requested certificate will be the public half of your key pair.

Upload CSR file.
Select .certSigningRequest file saved on your Mac.
Your certificate is ready.

Download, Install and Backup
Download your certificate to your Mac, then double click the .cer file to install in Keychain Access. Make sure to save a backup copy of your private and public keys somewhere secure.

Documentation
For more information on using and managing your certificates read:

App Distribution Guide
Your iOS Certificate:
ios_development.cer
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Expires</th>
</tr>
</thead>
<tbody>
<tr>
<td>Day Min Yuh</td>
<td>iOS Development</td>
<td>Nov 23, 2014</td>
</tr>
</tbody>
</table>
Your App has PERMISSION

com.myDomain.myApp

Source: Provisioning Profile Overview for Non-Programmer http://www.youtube.com/watch?v=hpzej8m_Lll
Apple ID

Bundle ID

Bundle ID vs. App ID

App ID, Device ID

iOS Team Provisioning Profile: *

Devices

App ID

Development Certificates

Xcode: iOS Wildcard App ID

iOS Development:
- Ann Johnson
- Tom Clark
- Mei Chen

Ann Johnson’s iPad
Tom Clark’s iPhone
Mei Chen’s iPod touch

App ID, Device ID

Mac Team Provisioning Profile: com.example.bjamesinc.trackmix

Devices
- Bill James’s iMac
- Juan Chavez’s MacBook Air
- Maria Ruiz’s MacBook Pro

App ID
- Xcode Mac App ID com example bjamesinc TrackMix

Development Certificates
- Mac Development: Bill James
- Mac Development: Juan Chavez
- Mac Development: Maria Ruiz

iOS App ID

Getting Started with iOS App IDs

An App ID is a string that defines both a keychain identity and a set of apps you are developing. Its primary use is as part of a provisioning profile; it specifies which apps are authorized by the profile to be signed and launched.

Register App IDs with Xcode
When you use Xcode to automatically provision your devices, you will receive a wildcard App ID by default. This App ID can be used for all of your iOS apps that do not require app services such as push notifications, iCloud, Game Center, In-App Purchase, Data Protection, and Passbook.

Register App IDs Manually
Team agents and admins can manually register App IDs and enable them for app services. Register your App ID.

Learn More
App Distribution Guide
Registering an App ID

The App ID string contains two parts separated by a period (.)—an App ID Prefix that is defined as your Team ID by default and an App ID Suffix that is defined as a Bundle ID search string. Each part of an App ID has different and important uses for your app. Learn More

App ID Description

Name: [Input Field]
You cannot use special characters such as @, &, *, ", '

App ID Prefix

Value: 8369C9ES2Z (Team ID)

App ID Suffix
Myday Mobile App

com.imtku.smap.mydaymobileapp

---

**App ID Description**

**Name:** Myday Mobile App

You cannot use special characters such as @, &, *, +, "

---

**App ID Prefix**

**Value:** 8369C9E52Z (Team ID)

---

**App ID Suffix**

**Explicit App ID**

If you plan to incorporate app services such as Game Center, In-App Purchase, Data Protection, and iCloud, or want a provisioning profile unique to a single app, you must register an explicit App ID for your app.

To create an explicit App ID, enter a unique string in the Bundle ID field. This string should match the Bundle ID of your app.

**Bundle ID:** com.imtku.smap.mydaymobileapp

We recommend using a reverse-domain name style string (i.e., com.domainname.appname). It cannot contain an asterisk (*).

---

**Wildcard App ID**

This allows you to use a single App ID to match multiple apps. To create a wildcard App ID, enter an asterisk (*) as the last digit in the Bundle ID field.

**Bundle ID:**

Example: com.domainname.*

---

Bundle ID: com.instrument.app

We recommend using a reverse-domain name style string (i.e., com.domainname.appname). It cannot contain an asterisk (*).

- **Wildcard App ID**
  This allows you to use a single App ID to match multiple apps. To create a wildcard App ID, enter an asterisk (*) as the last digit in the Bundle ID field.

  **Bundle ID:**
  
  Example: com.domainname.

---

**App Services**
Select the services you would like to enable in your app. You can edit your choices after this App ID has been registered.

**Enable Services:**
- Data Protection
  - Complete Protection
  - Protected Unless Open
  - Protected Until First User Authentication

- Game Center
- iCloud
- In-App Purchase
- Inter-App Audio
- Passbook
- Push Notifications

---

[Continue button highlighted]
Confirm your App ID.

To complete the registration of this App ID, make sure your App ID information is correct, and click the submit button.

App ID Description: Myday Mobile App
Identifier: 8369C9ES2Z.com.imtku.smaph.mydaymobileapp
Data Protection: ◯ Disabled
Game Center: ◯ Enabled
iCloud: ◯ Disabled
In-App Purchase: ◯ Enabled
Inter-App Audio: ◯ Disabled
Passbook: ◯ Disabled
Push Notifications: ◯ Disabled

Submit
Registration complete.

This App ID is now registered to your account and can be used in your provisioning profiles.

App ID Description: Myday Mobile App
Identifier: 8369C9ES2Z.com.imtku.smap.mydaymobileapp
Data Protection: Disabled
Game Center: Enabled
iCloud: Disabled
In-App Purchase: Enabled
Inter-App Audio: Disabled
Passbook: Disabled
Push Notifications: Disabled
### iOS App IDs

<table>
<thead>
<tr>
<th>Name</th>
<th>ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Myday Mobile App</td>
<td>com.imtkusmap.mydymobileapp</td>
</tr>
</tbody>
</table>
3 Your Device has PERMISSION

Source: Provisioning Profile Overview for Non-Programmer  http://www.youtube.com/watch?v=hpzej8m_LII
Getting Started with iOS Devices

To install apps and pre-release versions of iOS on a development device, you must first register that device using its unique device identifier (UDID).

Register Devices with Xcode
Xcode is the easiest way to register your iOS device. Connect your device to your Mac and click Use for Development in Xcode's Organizer window. Sign in with the Apple ID associated with your iOS Developer Program membership and Xcode will register the UDID of your device to your iOS development team.

Register Devices Manually
If you wish to upload multiple device IDs, team agents and admins can manually register devices.

Learn More
- App Distribution Guide
Unique Device Identifier (UDID)
Unique Device Identifier (UDID)

Registering a New Device or Multiple Devices

Pre-Release Software Reminder
You may only share Apple pre-release software with employees, contractors, and members of your organization who are registered as Apple developers and have a demonstrable need to know or use Apple software to develop and test applications on your behalf.

Unauthorized distribution of Apple confidential information (including pre-release software) is prohibited and may result in the termination of your Apple Developer Program. It may also subject you to civil and criminal liability.

Register Device
Name your device and enter its Unique Device Identifier (UDID).

Name: [Field]

UDID: [Field]
Unique Device Identifier (UDID)

Registering a New Device or Multiple Devices

Pre-Release Software Reminder
You may only share Apple pre-release software with employees, contractors, and members of your organization who are registered as Apple developers and have a demonstrable need to know or use Apple software to develop and test applications on your behalf.

Unauthorized distribution of Apple confidential information (including pre-release software) is prohibited and may result in the termination of your Apple Developer Program. It may also subject you to civil and criminal liability.

Register Device
Name your device and enter its Unique Device Identifier (UDID).

Name: iMydey iPhone

UDID: 00f3558b78ecfa5242fe648a04da11cd140c4a27
Unique Device Identifier (UDID)

Register Device
Name your device and enter its Unique Device Identifier (UDID).

Name: iMydog iPhone

UDID: 00f3558b78ecfa5242fe646a04da11cd140c4a27

Register Multiple Devices
Upload a file containing the devices you wish to register. Please note that a maximum of 100 devices can be included in your file and it may take a few minutes to process.

Download sample files
**Review and register.**

Confirm the device information is correct. Once this device is registered, you will not be able to edit the UDID and can only edit the name or disable it.

**Name:** iMyday iPhone  
**UDID:** 00f3558b78ecfa5242fe648a04da11cd140c4a27

---

**You can register 200 devices.**
The maximum number of devices you can register per membership year is 200. You may reset your device list at the start of your next membership year.
Registration complete.

Your device has been registered and can now be included in provisioning profiles for app development and installation. Registered devices are also eligible to install pre-release versions of iOS. For step-by-step instructions review the following:

- App Distribution Guide
## iOS Devices

Reset your device list before adding any new devices.

You can register 199 additional devices.

<table>
<thead>
<tr>
<th>Name</th>
<th>UDID</th>
</tr>
</thead>
<tbody>
<tr>
<td>iMyday iPhone</td>
<td>00f3558b78ecfa542f648a04da11cd140c4a27</td>
</tr>
</tbody>
</table>
Getting Started with iOS Provisioning Profiles

Provisioning profiles allow you to install apps onto your iOS devices. A provisioning profile includes signing certificates, device identifiers, and an App ID. Development provisioning profiles are used to build and install versions of your app during the development cycle, while distribution provisioning profiles are used to submit your apps to the App Store and distribute them to beta testers.

Generate Provisioning Profiles with Xcode

Xcode is the easiest way to create an iOS provisioning profile. Connect your device to your Mac and click Use for Development in Xcode’s Organizer window. Sign in with the Apple ID associated with your iOS Developer Program membership and Xcode will generate a provisioning profile containing your certificate and device identifier.

Generate Provisioning Profiles Manually

Team agents and admins can manually generate profiles.
What type of provisioning profile do you need?

Development

- iOS App Development
  Create a provisioning profile to install development apps on test devices.
What type of provisioning profile do you need?

Development

- **iOS App Development**
  Create a provisioning profile to install development apps on test devices.

[Continue button]
Select App ID.

If you plan to use services such as Game Center, In-App Purchase, and Push Notifications, or want a Bundle ID unique to a single app, use an explicit App ID. If you want to create one provisioning profile for multiple apps or don't need a specific Bundle ID, select a wildcard App ID. Wildcard App IDs use an asterisk (*) as the last digit in the Bundle ID field. Please note that iOS App IDs and Mac App IDs cannot be used interchangeably.

App ID: Myday Mobile App (8369C915Z2Z.com.imtku.smap.mydynamobileapp) ✗
Select certificates.

Select the certificates you wish to include in this provisioning profile. To use this profile to install an app, the certificate the app was signed with must be included.

- [ ] Select All
- [x] Day Min Yuh (iOS Development)

Continue
Select certificates.

Select the certificates you wish to include in this provisioning profile. To use this profile to install an app, the certificate the app was signed with must be included.

- [ ] Select All
- [ ] Day Min Yuh (iOS Development)

(Click on the "Continue" button)
Select devices.

Select the devices you wish to include in this provisioning profile. To install an app signed with this profile on a device, the device must be included.

- [ ] Select All
- [ ] iMyday iPhone
Select the devices you wish to include in this provisioning profile. To install an app signed with this profile on a device, the device must be included.

- Select All
- iMyday iPhone

Continue
Name this profile and generate.

The name you provide will be used to identify the profile in the portal. You cannot use special characters such as @, &, *, ', " for your profile name.

Profile Name: [Input Field]

Type: Development
App ID: Myday Mobile App
   (8369C9ES2Z.com.imtku.smap.mydaymobileapp)
Certificates: 1 Included
Devices: 1 Included

[Generate Button]
Add iOS Provisioning Profile

Name this profile and generate.

The name you provide will be used to identify the profile in the portal. You cannot use special characters such as @, &, *, ', " for your profile name.

Profile Name: MydayMobileApp

Type: Development

App ID: Myday Mobile App (8369C9ES2Z.com.imtku.smapp.mydaymobileapp)

Certificates: 1 Included

Devices: 1 Included

Generate
Your provisioning profile is ready.

Download and Install
Download and double click the following file to install your Provisioning Profile.

Name: MydayMobileApp
Type: Development
App ID: 8369C9E52Z.com.imtku.smap.mydaymobileapp
Expires: Nov 23, 2014

Download

Documentation
For more information on using and managing your Provisioning Profile read:

App Distribution Guide
MydayMobileApp.mobileprovision
Certificates.p12

File Format: Personal Information Exchange (.p12)
Enter a password which will be used to protect the exported items:

Password: 
Verify: 

Password Strength: Weak

Cancel  OK
Enter a password which will be used to protect the exported items:

Password: ************
Verify: ************

Password Strength: Excellent
Keychain Access wants to export key “iMyday” from your keychain.

To allow this, enter the “login” keychain password.

Password: **********

[Buttons: Always Allow, Deny, Allow]
Certificates.p12
MydayMobileApp.mobileprovision
Certificates.p12
MydayMobileApp.mobileprovision
certificate password
## Myday Mobile App

Myday Mobile App for Prof. Min-Yuh Day, Department of Information Management, Tamkang University

<table>
<thead>
<tr>
<th>Platform</th>
<th>App Name</th>
<th>Build Status</th>
<th>Build Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>iOS</td>
<td>Myday Mobile App</td>
<td>Rebuild</td>
<td>Error</td>
</tr>
<tr>
<td>Windows</td>
<td></td>
<td>Rebuild</td>
<td>xap</td>
</tr>
<tr>
<td>Linux</td>
<td></td>
<td>Rebuild</td>
<td>ipk</td>
</tr>
</tbody>
</table>

For Android, there is a No key selected message, creating an OTA install option.

### Build Details
- **App ID**: 573842
- **Version**: 1.0.0
- **PhoneGap Version**: 2.9.0
- **Owned by**: myday@gmail.com
- **Last built**: 10 days
- **Source**: zip package
Download MydayMobileApp.ipa
Install on iOS Devices (iPhone/iPAD)
Install Myday Mobile App 1.0.0

Android apk

BlackBerry OTA install

iOS ipa

symbian wgz

webOS ipk

Windows Phone xap
MydayMobileApp.ipa
Test app.ipa on iOS Devices (iPhone)
Test app.ipa on iOS Devices (iPhone)
Test app.ipa on iOS Devices (iPhone)

MIN-YUH DAY
ASSISTANT PROFESSOR
DEPARTMENT OF
INFORMATION
MANAGEMENT,
TAMKANG UNIVERSITY

MOBILE: 886-9-33282012
TEL: 886-2-26215656 EXT. 2846
FAX: 886-2-26209737
OFFICE: B929
ADDRESS: NO.151, YINGZHUAN RD., DANSHUI
DIST., NEW TAIPEI CITY 25137, TAIWAN (R.O.C.)
EMAIL: MYDAY@MAIL.TKU.EDU.TW
WEB: HTTP://MAIL.TKU.EDU.TW/MYDAY/

GOOGLE MAP

HOME RESEARCH TEACHING ABOUT
Demo

Testing and Refining Your App

https://build.phonegap.com/
index.css

```css
body {
  -webkit-touch-callout: none;
  -webkit-text-size-adjust: none;
  -webkit-user-select: none;
  background-color: #E4E4E4;
  background-image: linear-gradient(top, #A7A7A7 0%, #E4E4E4 51%);
  background-image: -webkit-linear-gradient(top, #A7A7A7 0%, #E4E4E4 51%);
  background-image: -ms-linear-gradient(top, #A7A7A7 0%, #E4E4E4 51%);
  background-image: -webkit-gradient( linear, left top, left bottom, color-stop(0, #A7A7A7), color-stop(0.51, #E4E4E4) );
  background-attachment: fixed;
  font-family: 'Helvetica Neue-Light', 'Helvetica Neue', Helvetica, Arial, sans-serif;
  font-size: 12px;
  height: 100%;
  margin: 0px;
  padding: 0px;
  text-transform: uppercase;
  width: 100%;
}

/* prevent callout to copy image, etc when tap to hold */
/* prevent webkit from resizing text to fit */
/* prevent copy paste, to allow, change 'none' to 'text' */

<link rel="stylesheet" type="text/css" href="css/index.css" />

<--PhoneGap-->
CSS code for index.css:

```css
body {
  -webkit-touch-callout: none;
  -webkit-text-size-adjust: none;
  -webkit-user-select: none;
  background-color: #E4E4E4;
  background-image: linear-gradient(top, #A7A7A7 0%, #E4E4E4 51%);
  background-image: -webkit-linear-gradient(top, #A7A7A7 0%, #E4E4E4 51%);
  background-image: -ms-linear-gradient(top, #A7A7A7 0%, #E4E4E4 51%);
  background-image: -o-linear-gradient(top, #A7A7A7 0%, #E4E4E4 51%);
  background-image: linear-gradient( linear, left top, left bottom, color-stop(0, #A7A7A7), color-stop(0.51, #E4E4E4));
  background-attachment: fixed;
  font-family: 'HelveticaNeue-Light', 'HelveticaNeue', Helvetica, Arial, sans-serif;
  font-size: 12px;
  height: 100%;
  margin: 0px;
  padding: 0px;
  text-transform: none;
  width: 100%;
}
```

HTML code with PhoneGap:

```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>Social Media</title>
  <!-- PhoneGap -->
  <link rel="stylesheet" type="text/css" href="css/index.css"/>
</head>
<body>
  <!-- Body content here -->
</body>
</html>
```
Fall 2013 (2013.09-2014.01)

Social Media Apps Programming
(Fall 2013)

Case Study for Information Management
(Fall 2013)

Digital Information Services
(Fall 2013)

Spring 2013 (2013.02-2013.06)

Home  Research  Teaching  About
Removing Unnecessary Files

Once you've included the necessary assets, remove the `phonegap.js` (cordova.js) as Build will automatically inject it during compile time.

Why must you delete the `phonegap.js`?

PhoneGap requires a different JavaScript file for each platform and using an incompatible `phonegap.js` will result in errors when running your application.

Making Sure You can Still Access the PhoneGap API

Once you've deleted the `phonegap.js` you'll need to make sure that your application can still access the PhoneGap API.

To do so, simply ensure that the following reference is made in your `index.html`:

```html
<script src="phonegap.js"></script>
```
Usage and Additional Information:

Unless otherwise specified in a config.xml, each platform will try to use the default splash.png during compilation. To define platform specific splash screens please use the guide provided below.

Splash files should be the file formats specified in the examples below. Any other file type is not guaranteed to work across platforms.

Warning:

If you do not supply the gap:platform attribute, the referenced image will be copied to ALL platforms, increasing the size of their application packages.

Default

The default splash must be named splash.png and must reside in the root of your application folder.

```xml
<gap:splash src="splash.png" />
```

iOS

We support classic, retina, iPhone 5 and iPad displays; the following will define splash screens for each of those. Standard iPads have two different splash screens, portrait, landscape. Retina iPads have two additional splash screens, retina portrait and retina landscape (PhoneGap 2.5.0+ only).

```xml
<gap:splash src="splash/ios/Default.png" gap:platform="ios" width="320" height="480" />
```
icon.png
(128x128)

<icon src="icon.png" />
logo.png
(172x200)
icon/ios/icon-57.png
(57x57)

cfg.xml

```
<icon src="icon.png" />
<icon gap:density="ldpi" gap:platform="android" src="res/icon/android/icon-36-ldpi.png" />
<icon gap:density="mdpi" gap:platform="android" src="res/icon/android/icon-48-mdpi.png" />
<icon gap:density="hdpi" gap:platform="android" src="res/icon/android/icon-72-hdpi.png" />
<icon gap:density="xhdpi" gap:platform="android" src="res/icon/android/icon-96-xhdpi.png" />

<icon gap:platform="ios" height="57" src="res/icon/ios/icon-57.png" width="57" />
<icon gap:platform="ios" height="72" src="res/icon/ios/icon-72.png" width="72" />
<icon gap:platform="ios" height="114" src="res/icon/ios/icon-57-2x.png" width="114" />
<icon gap:platform="ios" height="144" src="res/icon/ios/icon-72-2x.png" width="144" />
```
icon/ios/icon-57.png

(57x57)

```
<icon gap:platform="ios" height="57"
src="res/icon/ios/icon-57.png" width="57" />
```
icon/ios/icon-57.png

(57x57)

<icon gap:platform="ios" height="57" src="res/icon/ios/icon-57.png" width="57" />
# Pixel density

**Resolution**: 320x480 (2048x1536)  
**PPI**: 64 (326)

<table>
<thead>
<tr>
<th>Model</th>
<th>Generations</th>
<th>Diagonal cm (in)</th>
<th>Resolution</th>
<th>ppcm (PPI)</th>
</tr>
</thead>
<tbody>
<tr>
<td>iPad</td>
<td>1st Gen, 2</td>
<td>25 (9.7)</td>
<td>1024x768</td>
<td>52 (132)</td>
</tr>
<tr>
<td>iPad / iPad Air</td>
<td>3rd Gen, 4th Gen</td>
<td>25 (9.7)</td>
<td>2048x1536</td>
<td>104 (264)</td>
</tr>
<tr>
<td>iPad Mini</td>
<td>2nd Gen</td>
<td>20 (7.9)</td>
<td>2048x1536</td>
<td>128 (326)</td>
</tr>
<tr>
<td>iPhone 3GS</td>
<td>3 / 3rd Gen</td>
<td>8.9 (3.5)</td>
<td>320x480</td>
<td>64 (163)</td>
</tr>
<tr>
<td>iPhone 4 / iPod Touch</td>
<td>4 / 4th Gen</td>
<td>8.9 (3.5)</td>
<td>960x640</td>
<td>128 (326)</td>
</tr>
<tr>
<td>iPhone 5S</td>
<td>5S / 6th Gen</td>
<td>10 (4)</td>
<td>1136x640</td>
<td>128 (326)</td>
</tr>
</tbody>
</table>

<gap:splash gap:platform="ios" height="480" src="res/screen/ios/screen-iphone-portrait.png" width="320" />
<gap:splash gap:density="mdpi" gap:platform="android" src="res/screen/android/screen-mdpi-portrait.png" />

200x320
320x480
480x800
720x1280

(screen-mdpi-portrait.png)
<gap:splash gap:density="ldpi" gap:platform="android" src="res/screen/android/screen-ldpi-portrait.png" />
<gap:splash gap:density="mdpi" gap:platform="android" src="res/screen/android/screen-mdpi-portrait.png" />
<gap:splash gap:density="hdpi" gap:platform="android" src="res/screen/android/screen-hdpi-portrait.png" />
<gap:splash gap:density="xhdpi" gap:platform="android" src="res/screen/android/screen-xhdpi-portrait.png" />
<gap:splash gap:platform="blackberry" src="res/screen/blackberry/screen-225.png" />
<gap:splash gap:platform="ios" height="480" src="res/screen/ios/screen-iphone-portrait.png" width="320" />
<gap:splash gap:platform="ios" height="960" src="res/screen/ios/screen-iphone-portrait-2x.png" width="640" />
<gap:splash gap:platform="ios" height="1024" src="res/screen/ios/screen-ipad-portrait.png" width="768" />
<gap:splash gap:platform="ios" height="768" src="res/screen/ios/screen-ipad-landscape.png" width="1024" />
<gap:splash gap:platform="winphone" src="res/screen/windows-phone/screen-portrait.jpg" />
<access origin="http://127.0.0.1*" />
</widget>
Summary

• jQuery Mobile/PhoneGap
  – HTML5, CSS3, JavaScript
  – jQuery Mobile
  – PhoneGap
  – PhoneGap Build
    • index.html, css, js, png
    • config.xml

• Demo
  – Create Hybrid Apps with PhoneGap
  – Testing iOS and Android Devices
References

• Learn HTML5 and JavaScript for iOS: Web Standards-based Apps for iPhone, iPad, and iPod touch, Scott Preston, Apress, 2012
• PhoneGap
• PhoneGap Build
  – [https://build.phonegap.com/](https://build.phonegap.com/)
• jQuery Mobile
• Introduction to PhoneGap Build