

Cross Platform Development

Lab # 5 Part B: *Using PhoneGap for Cross-Platform Development (iOS)*

The objective of this lab is to build upon the work completed in previous lab that taught you how to create an Android smartphone application using the PhoneGap Build service. This lab will show you how to build the application for iOS in order to deploy it on iPods, iPhones and iPads. We will test this application on an iOS device after we learn how to sign the application for testing. In order to complete this lab, you require access to a Mac running OS X to create your signing keys, an iOS device and an Apple iOS Developer account. Note that you need to work with your files from Lab4 or Lab5A.

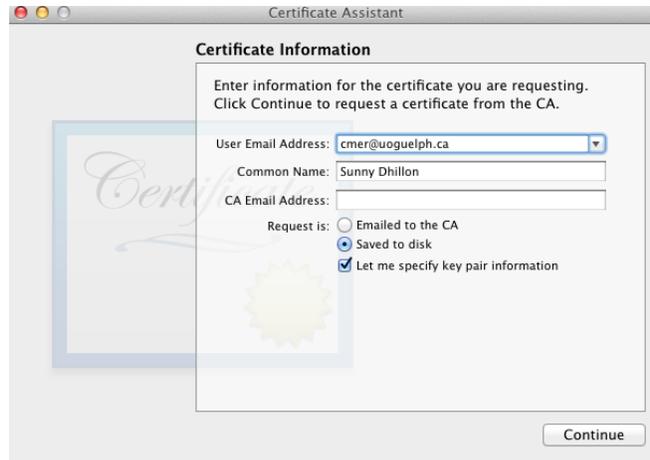
Building your Application for iOS with PhoneGap

Step 1: iOS Developer Account Registration

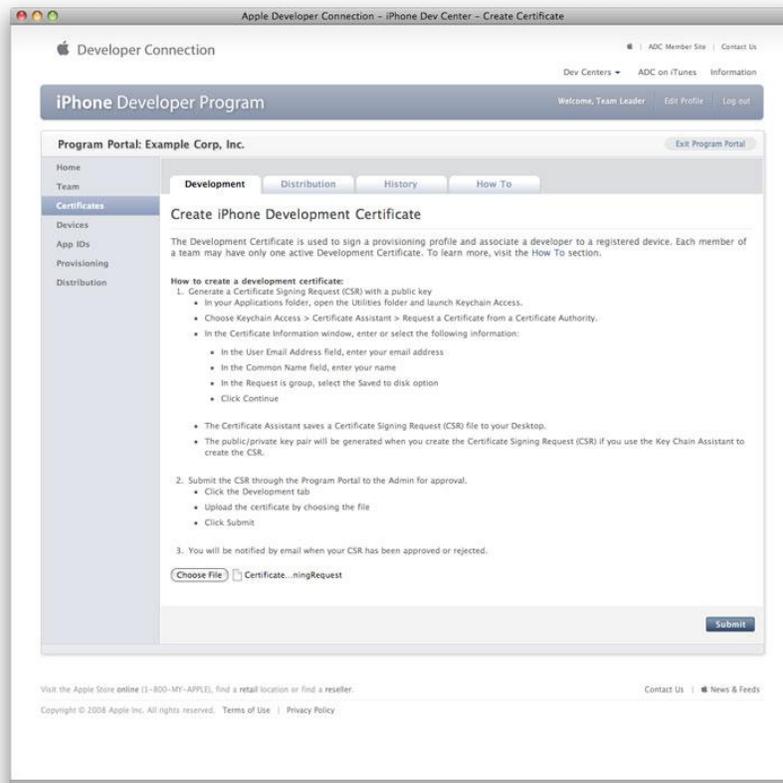
- In order to build and test iOS applications, you must have a developer account
- A standard account currently costs \$99 per year or an academic account is available free of charge through many educational institutions
- To enroll in either program, register at <https://developer.apple.com/programs/ios/>
- For all academic program registrations, it must be completed by an employee of the University and several checks will be required
- Without access to a developer account, you cannot build your application

Step 2: Creating Signing Keys

- Provided here will be an abridged version of the process found at the PhoneGap website: <https://build.phonegap.com/docs/ios-builds> along with the Apple Developer portal. More instructional materials and videos are available here <https://developer.apple.com/videos/ios/>
- The first step is to create a certificate signing request through the Keychain Access application in OS X.
 - In the menu bar, go to Keychain access and preferences
 - In the certificates tab, ensure OCSP and CRL are set to off
 - Go back to the menu and in the Certificate Assistant submenu, select, ‘Request a Certificate from a Certificate Authority’
 - Enter the name and user email, than select ‘Saved to Disk’. The CA Email address can remain blank. Ensure the ‘Let me specify key pair information’ box is checked.



- Select a location to save your signing request and click save
- Specify a 2048 bit, RSA key and continue for the request file to be created.
- Next in the iOS Provisioning Portal website, select the certificates tab and click ‘Add Certificate.’
 - This screen will allow you to select a file, choose the signing request file you previously created and click submit. Your Team Leader will now need to approve your request before it is available for download.



- Once approved, you will be able to download both your developer certificate and the WWDR intermediate certificate through the same Certificates tab
- On the local Mac machine, double click each certificate to install
- The final step is to save our private key by exporting to a .p12 file
 - In Keychain Access, a new item called ‘iPhone Developer’ or ‘iPhone Distribution’ followed by your name will be listed. You may need to select the ‘My Certificates’ category to easily locate it. Control click this and select export
 - Choose the .p12 file format and select a password that you will remember. Save this .p12 file.

Step 3: Compiling and Signing your iOS Application

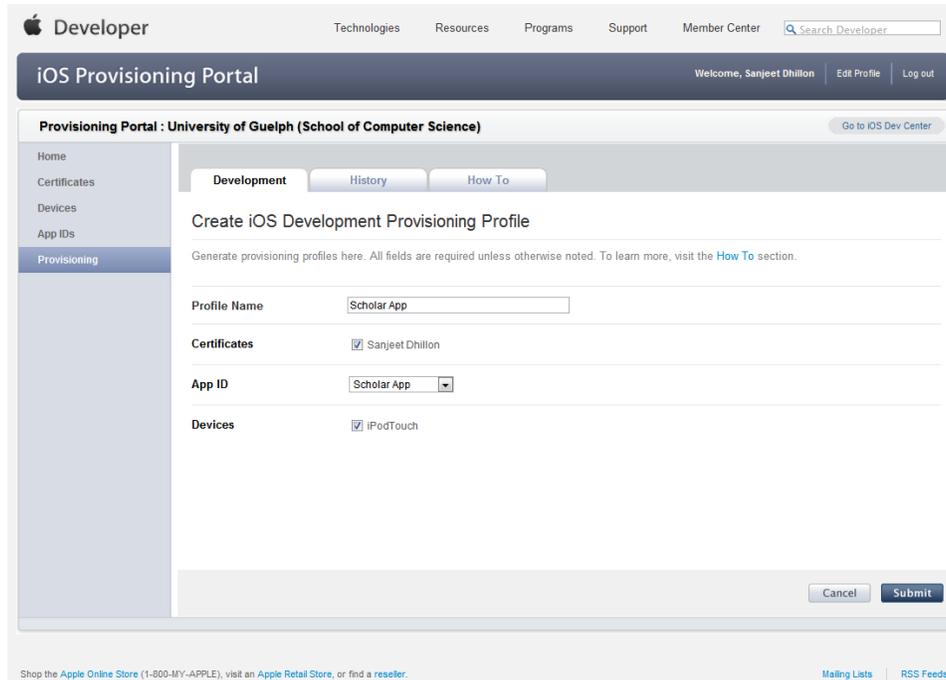
When an application is submitted to any application market, it must first be signed with a digital certificate but with iOS, we must have this certificate even for testing. We will create an Application ID, add a test device and finally build our application in this step.

- In the iOS Provisioning Portal, select the ‘App IDs’ tab
 - Press the New App ID button and enter the requested information. Instructions are given on how to choose a suitable bundle identifier.
- Select the ‘Devices’ tab
 - Press the add devices button and enter the name and Device ID of your iOS device. To find the device ID, open iTunes with the device connected, select the device and click on the serial number for the Identifier to be displayed.



- Now once again return to the provisioning portal and select the ‘Provisioning’ tab
 - Click the new profile button then enter a name for the profile (usually application name), check the box beside your certificate and device, then select the App ID

and click submit. Once the request is approved, you will be able to download this file and store it with your .p12 file.

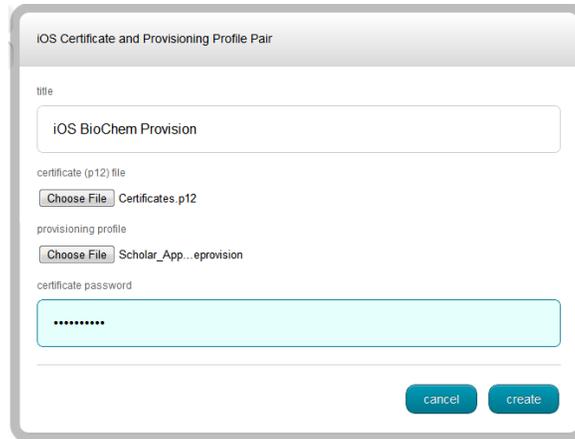


- Open Lab5b.zip and extract it to a folder of your choice or if you have completed Lab 5 Part A, you may use those files
 - Open the config.xml file and after the versionCode tag, add an id tag with the bundle identifier you chose in the previous step. For example:

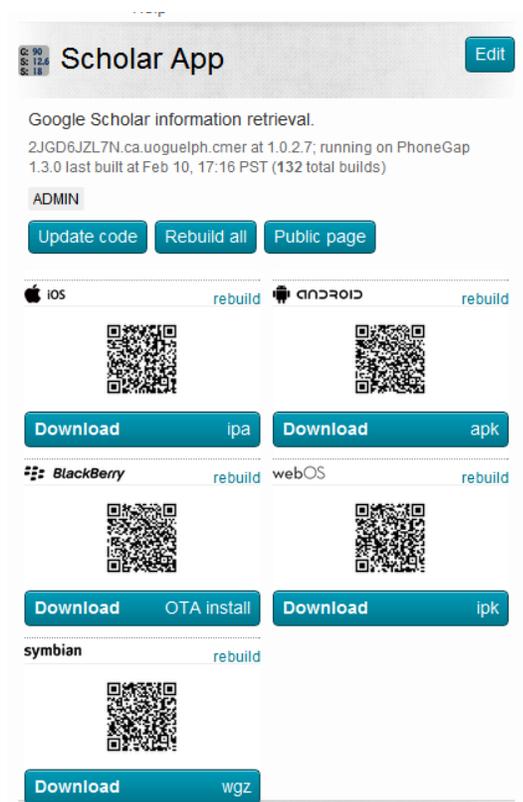
```
id = "ca.uoguelph.ScholarApp">
```

This line is required for proper iOS signing of the application. Without this change or an incorrect ID, an error will be given. This will not affect builds for Android so two versions of the config.xml file are **not** needed.

- Go to <https://build.phonegap.com> and login to the account you made in Lab 5 Part A
 - Once logged in, click your email address in the top right corner and select the signing tab. Under iOS, select Add a Key.
 - Create a title of your choosing and select your p12 file and provisioning profile. Next enter the certificate password created when exporting to the p12 and click create. Set this key as default. The provisioning profile is specific to this application and testing device, it will need to be modified for App Store distribution or use for other applications and devices.



- Returning to the main page, click the Scholar App button you created previously. Press the update code button and upload a new zip file containing the modified config.xml. The application will now compile for multiple platforms, including iOS. If you receive an error, ensure the signing key you have selected is being used by pressing the edit button and going to the signing tab. When finished you will be presented with a screen like this



You will now be able to download the .ipa file for testing on your device. You may also download for other platforms if you wish.

Step 4: Testing

- The process for installing and testing on iOS devices is quite simple if all steps were followed correctly
- Double click on your downloaded ipa file and install the application into iTunes
 - Sync your test device to iTunes to have the application transferred
 - You may now test it on the device

- If you find problems with the user interface on a certain devices, try going back into your code and optimizing the CSS to detect device resolution and aspect ratio, then display accordingly.
- The lab is now completed. You now have a fully functioning iOS application. Now that you have completed the translation of your application for BlackBerry, Android and iOS, you can attempt to download and try other versions that the PhoneGap Build service has provided you.