Code Signing Your Buzztouch Application
By Ed Goodall (@Ed Goodall)
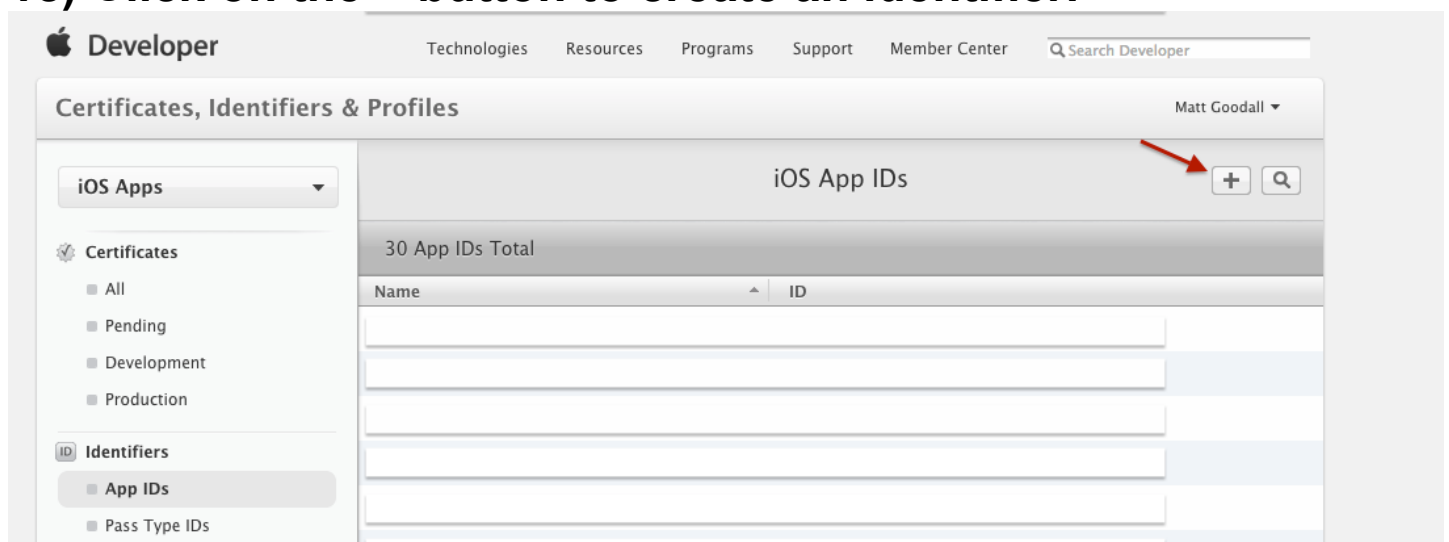
1a) Bring up the Certificates, Identities and Profiles portal in the Apple Developer Center.
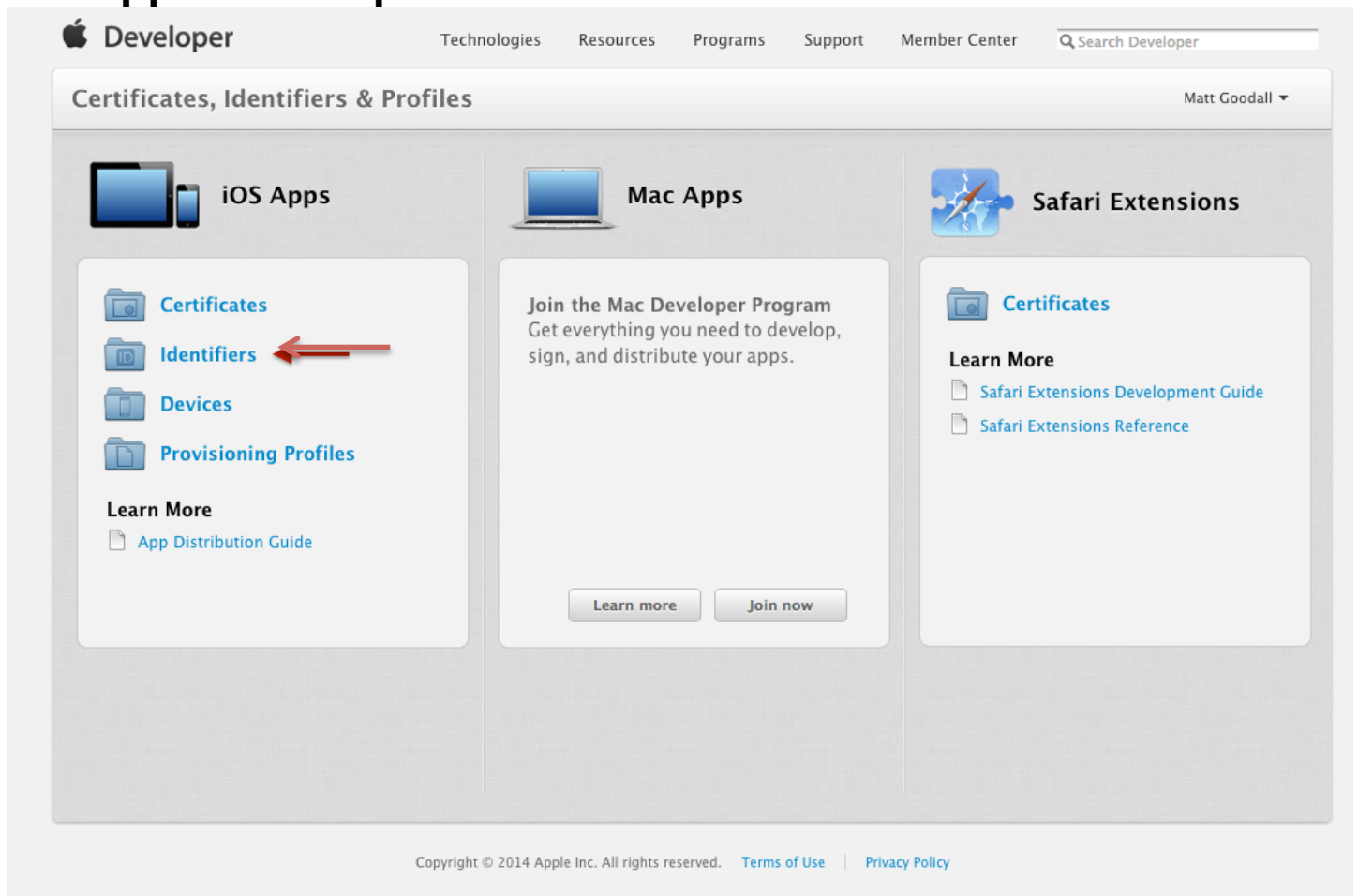
1b) Click on "Identifiers".
1c) Click on the + button to create an identifier.

1d) Enter an app name and an ID, usually laid out as a reverse domain name, e.g. if my app name was Ed's App and my website was www.fusion-technology.co.uk, I would use

co.uk.fusion-technology.EdsApp.

ID 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: My App Name

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

App ID Prefix

Value: AS3_____ (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: co.uk.fusion-technology.myappname ←

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

IMPORTANT: Take note of your App ID Prefix / Team ID, you'll need this later.

1e) Press Continue, and then you have completed creation of your App ID.

ID   **Registration complete.**

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

|	
App ID Description:	**My App Name**
Identifier:	**AS3_____.co.uk.fusion-technology.myappname**
App Groups (Beta):	○ Disabled
Associated Domains (Beta):	○ Disabled
Data Protection:	○ Disabled
Game Center:	● **Enabled**
HealthKit (Beta):	○ Disabled
HomeKit	○ Disabled
Wireless Accessory Configuration	○ Disabled
iCloud (Beta):	○ Disabled
In-App Purchase:	● **Enabled**
Inter-App Audio:	○ Disabled
Passbook:	○ Disabled
Push Notifications:	○ Disabled
VPN Configuration & Control (Beta):	○ Disabled

Step 2) Creating your Developer Certificates

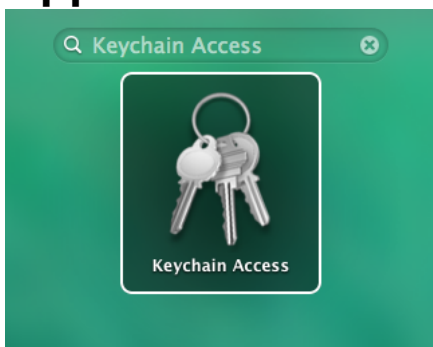## 2a) Go to the Certificates -> All page.
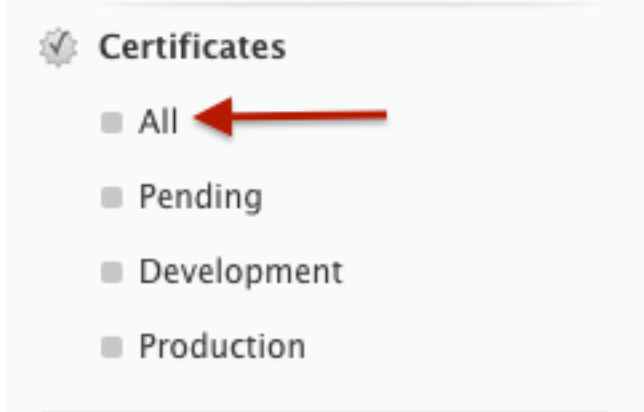
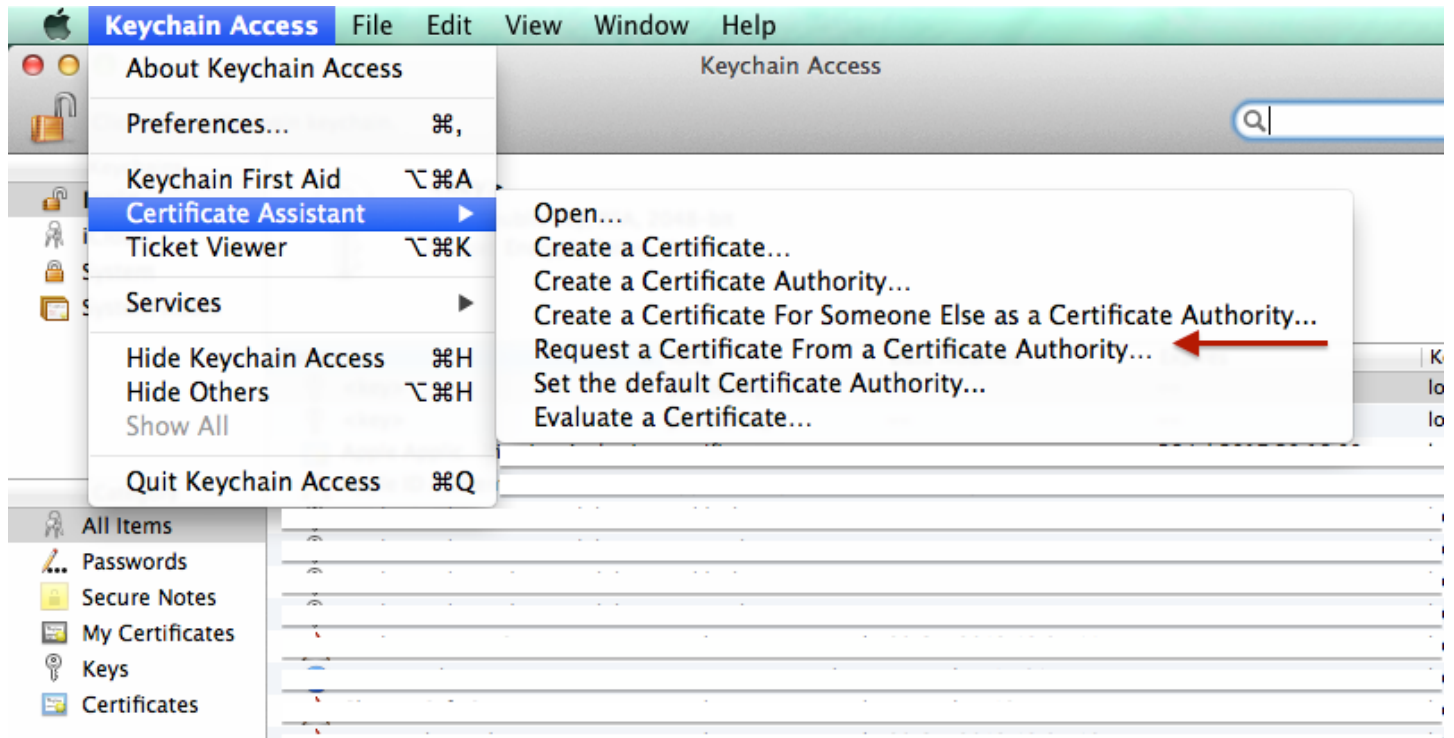## 2b) Press the + icon to create a new certificate.
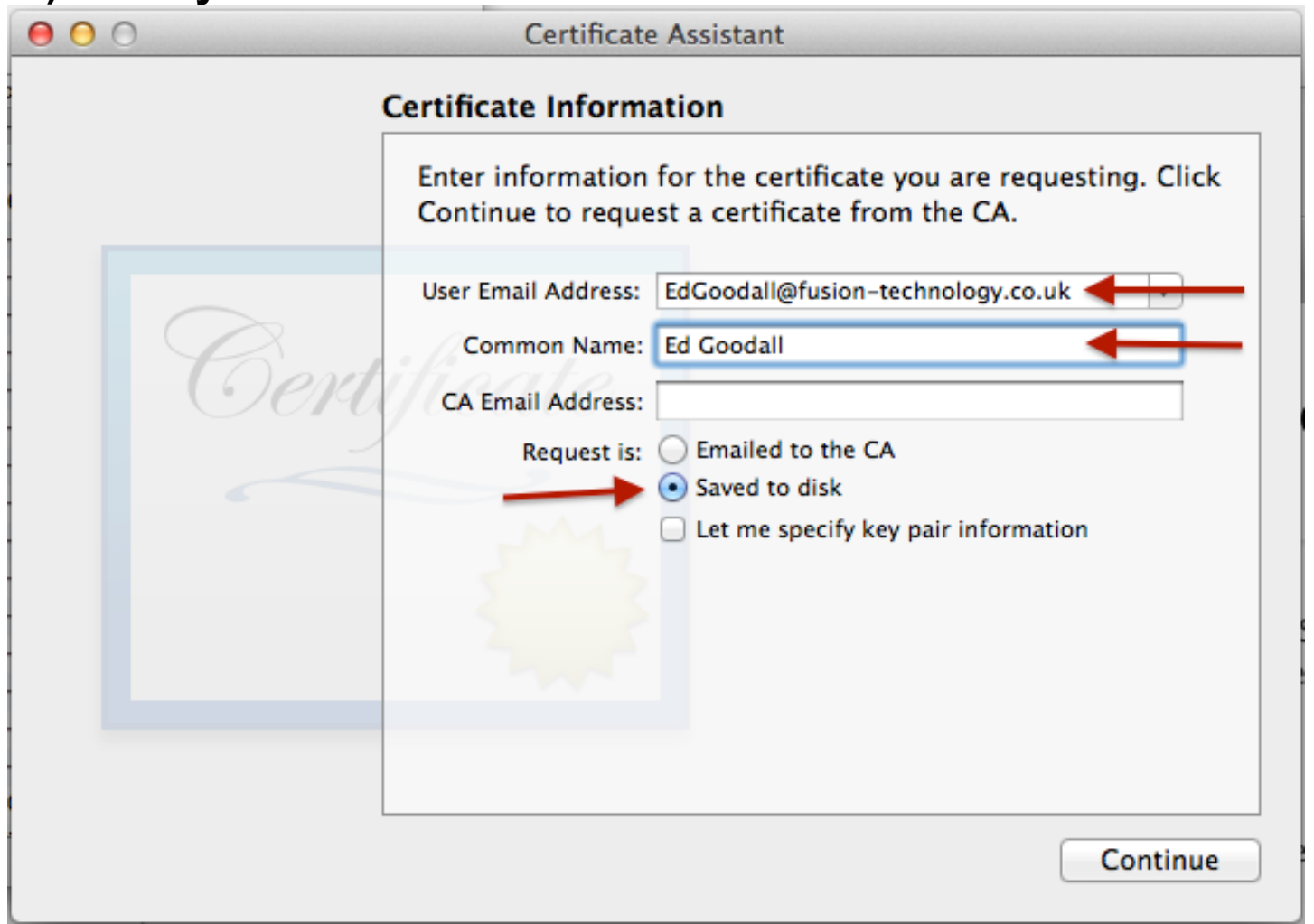## 2c) When the new screen appears, select iOS App Development.
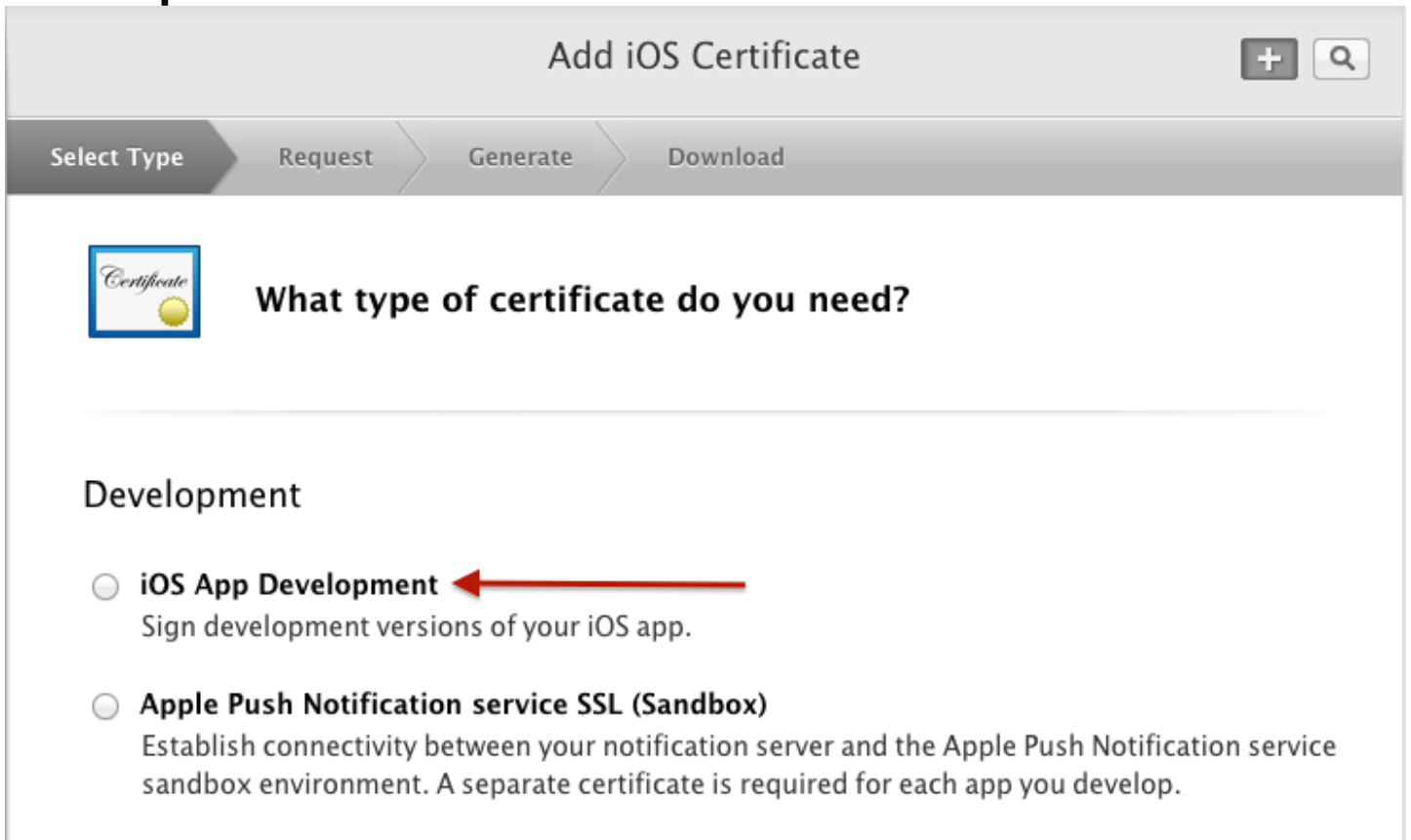
2d) Now open Keychain Access (this is in Utilities in your Applications folder).

2e) Go to Keychain Access -> Certificate Assistant -> Request Certificate

2f) Enter your information and select 'Saved to Disk'

2g) After saving the Request to a folder, return to the Apple Website and upload it.

Add iOS Certificate   [+] [Q]

Select Type   Request   Generate   Download

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.

Choose File...   ☐ CertificateSigningRequest.certSigningRequest

Cancel   Back   **Generate**

2h) Generate your certificate, download and double click to install. You've now made your Developer Certificate. Repeat the steps from 2b to 2g again, but for Distribution. This will generate the certificate for when you put the app on the app store. Only use the developer certificate when you are testing on your own devices.
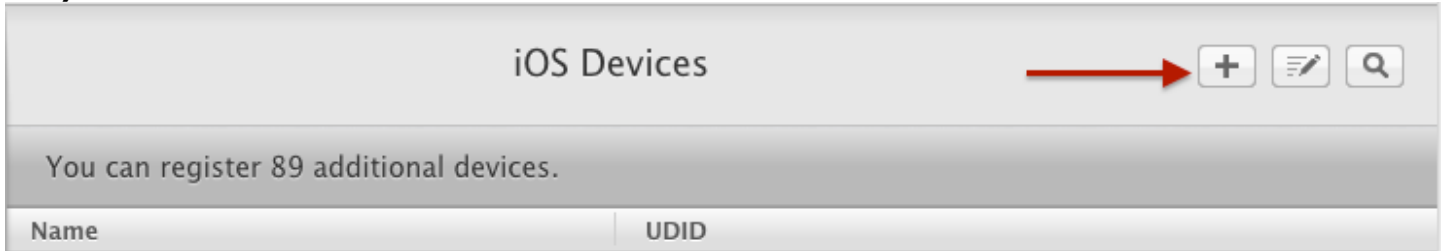
Step 3) Add your own device to the Developer Centre

When testing on your own device, you need to let Apple know what devices you are using.

3a) Go to devices.

3b) Press the + icon.

3c) Give a friendly name for your device. NOTE: This does not necessarily have to be the device name.

3d) To get your devices UDID (Unique Device Identifier), go to http://whatsmyudid.com/ and follow the instructions. Input the UDID you get (25 characters) into the UDID box and continue.

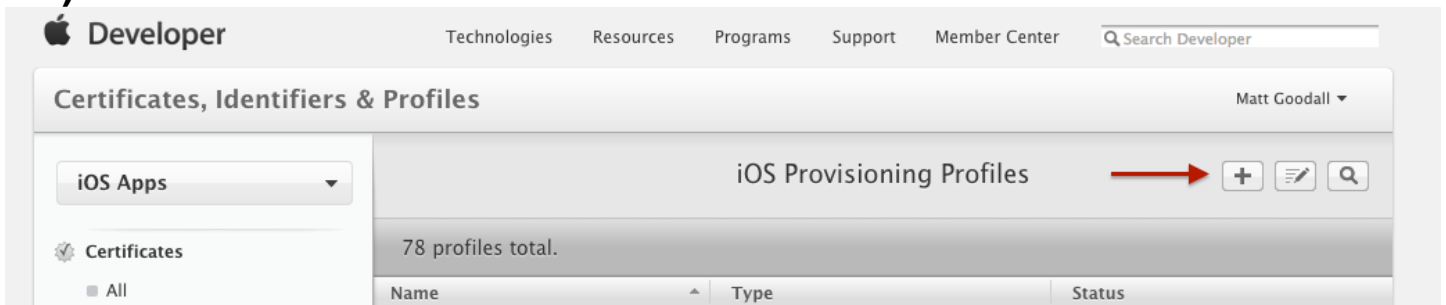## You have now added your device to the Developer Center.

Step 4) Create the Provisioning Profiles

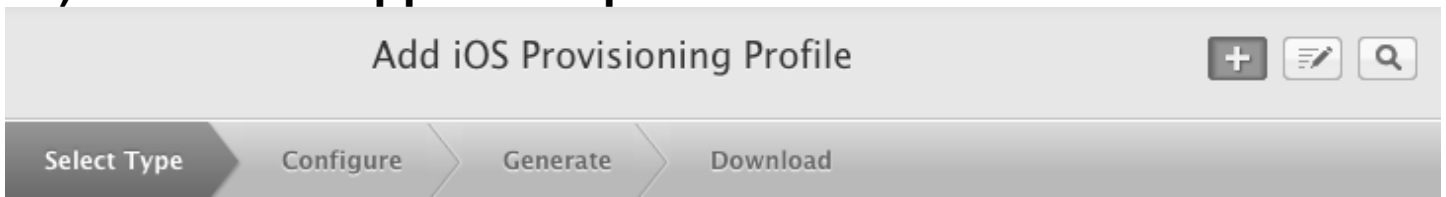## In order to install or compile your app, you need to create Provisioning Profiles.

4a) Go to Provisioning Profiles -> All.
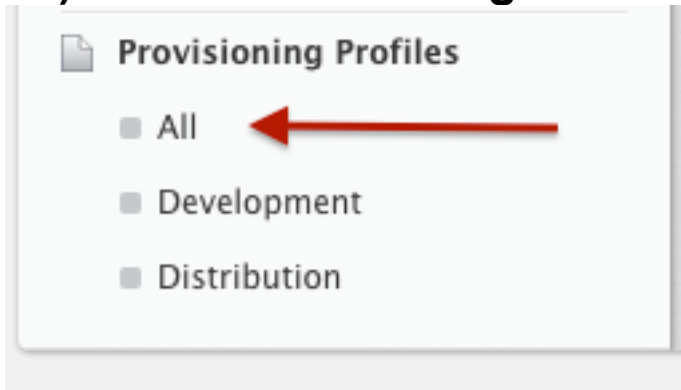
4b) Press the + button.

4c) Select iOS App Development.

4d) From the dropdown, select the App ID we created earlier.

Add iOS Provisioning Profile

Select Type   Configure   Generate   Download

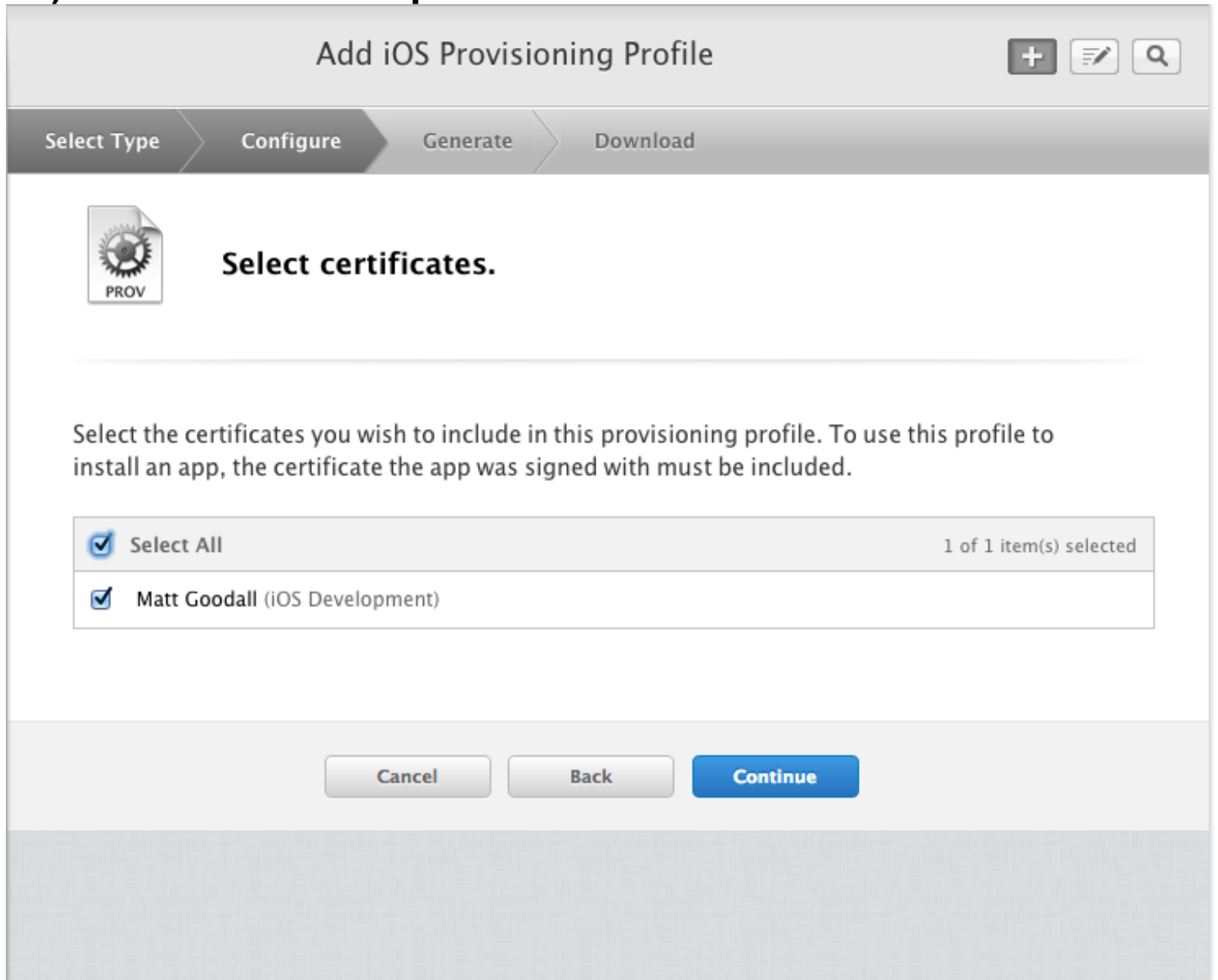### Select App ID.

PROV

If you plan to use services such as Game Center, In-App Purchase, and Push No
or want a Bundle ID unique to a single app, use an explicit App ID. If you want t
provisioning profile f
App ID. Wildcard App
note that iOS App ID:

App ID

My App Name (AS3   .co.uk.fusion-technology.myappname)

4e) Select the Development Certificate we created earlier.

4f) Select the devices you imported earlier.
NOTE: The devices selected here are the only devices that can test this app. Be careful with your choosing!

Add iOS Provisioning Profile

Select Type → Configure → Generate → Download

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   6 of 11 item(s) selected

☐

☐

☑ Eds iPad

☑ Eds iPhone

☑ Edward Goodall's iPhone

4g) Give a friendly name for this profile. Choose whatever you want (typically, I use <app name> <profile type>, for example "My Amazing App – Development")

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.

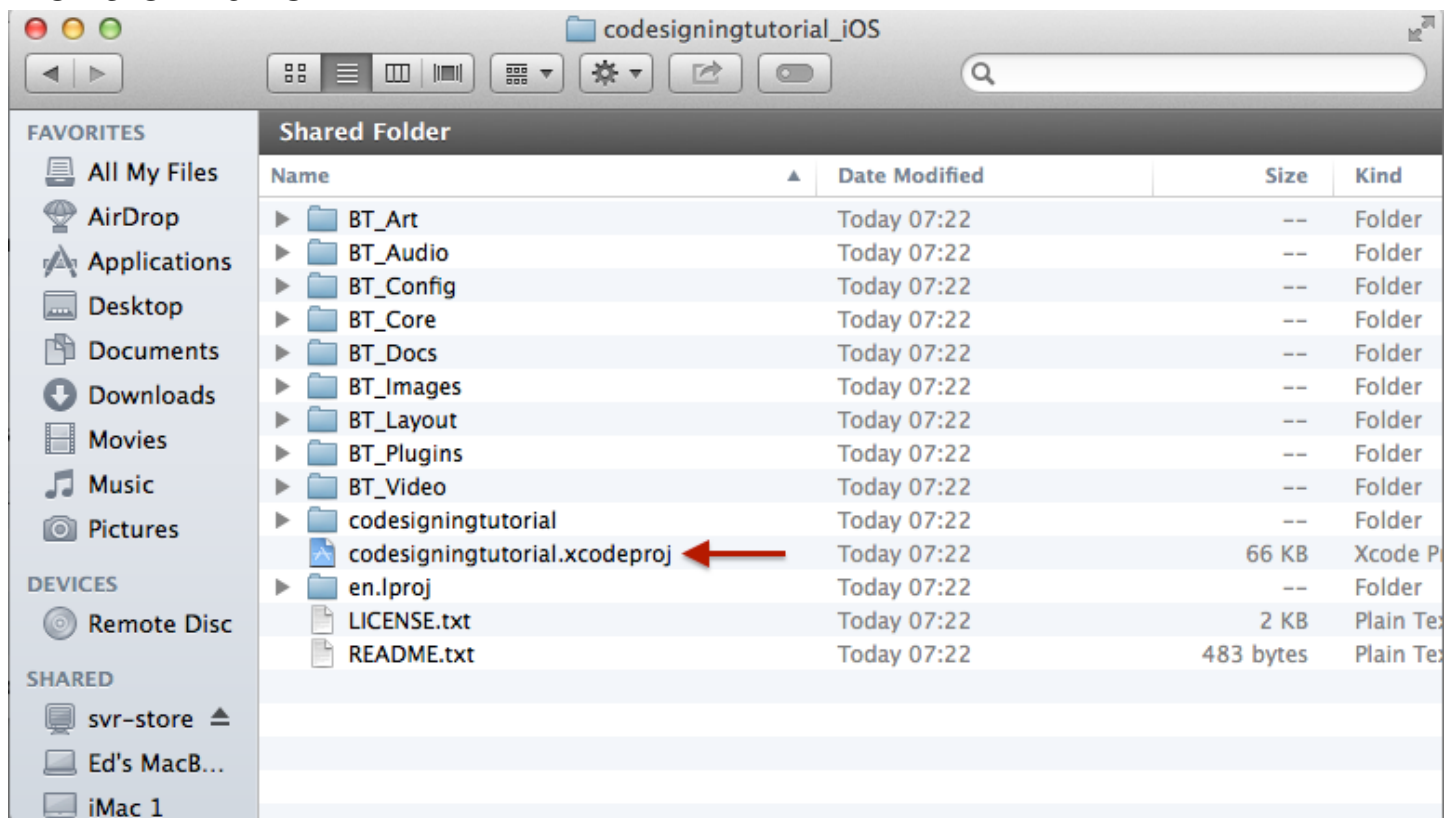Profile Name: | My App Name Development

Type: **Development**

4h) Download your provisioning profile, and double click to install.
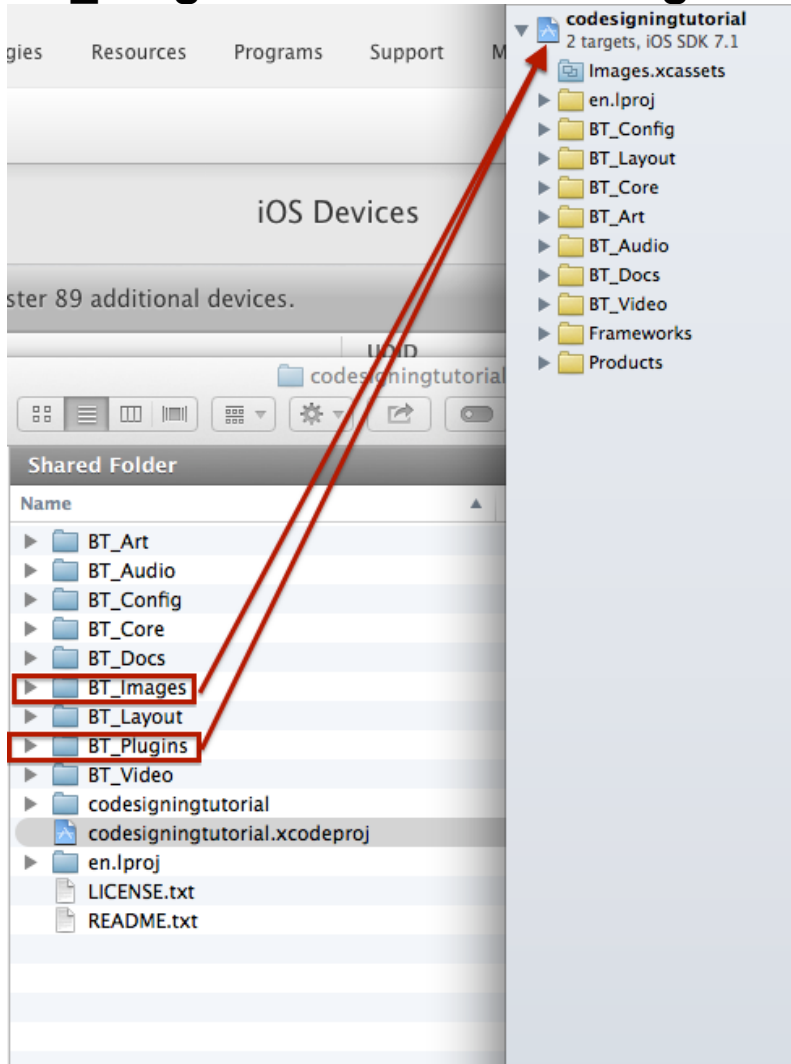

4i) Repeat steps 4b to 4h but using Production / Distribution.

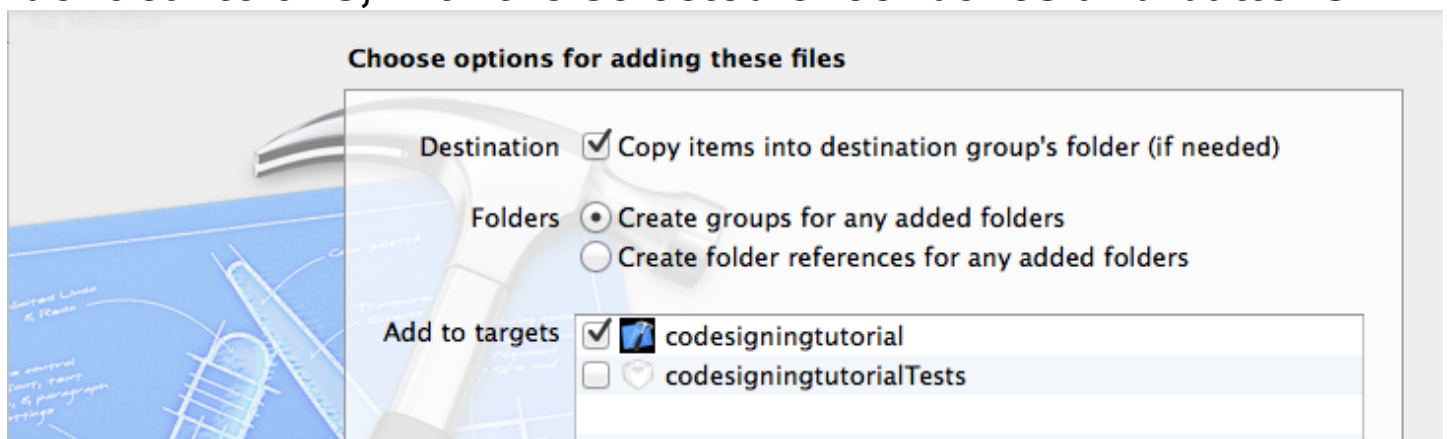# Step 5) Configure your Xcode project

5a) Open your xcode project downloaded from the Buzztouch Control Panel.
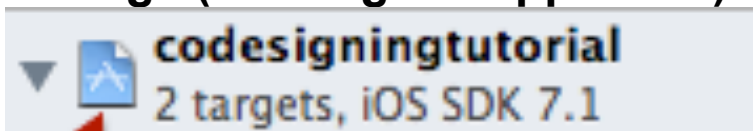
5b) Xcode projects don't include 2 folders, BT_Images and BT_Plugins. You need to drag these into your project.

5c) Make sure when you drag the folders, you see a dialog identical to this, with the selected checkboxes and buttons.

Choose options for adding these files

Destination ☑ Copy items into destination group's folder (if needed)

Folders ⦿ Create groups for any added folders
⦾ Create folder references for any added folders

Add to targets ☑ codesigningtutorial
☐ codesigningtutorialTests

5d) Insert the Build number, Version and App ID under General Settings (clicking the app name)

codesigningtutorial
2 targets, iOS SDK 7.1

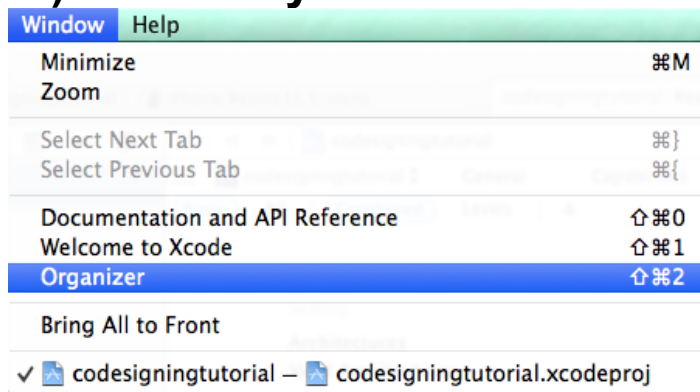**NOTE: With the App ID, put the prefix in with this. It tells Xcode what identity you're using.**

5e) Under Build Settings, change the Code Signing identity certificate to the Development one we selected earlier (or Distribution if compiling for the store).
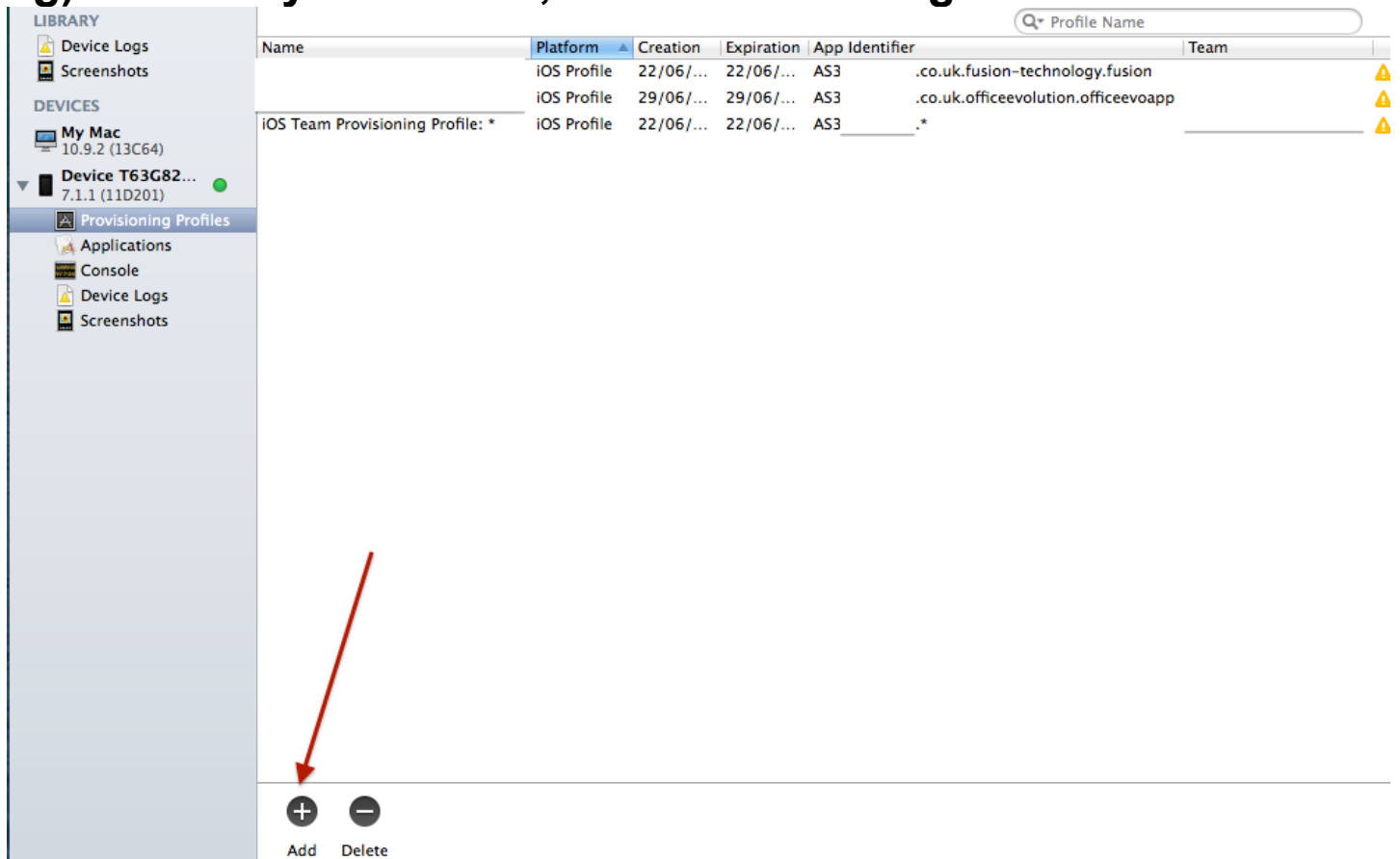

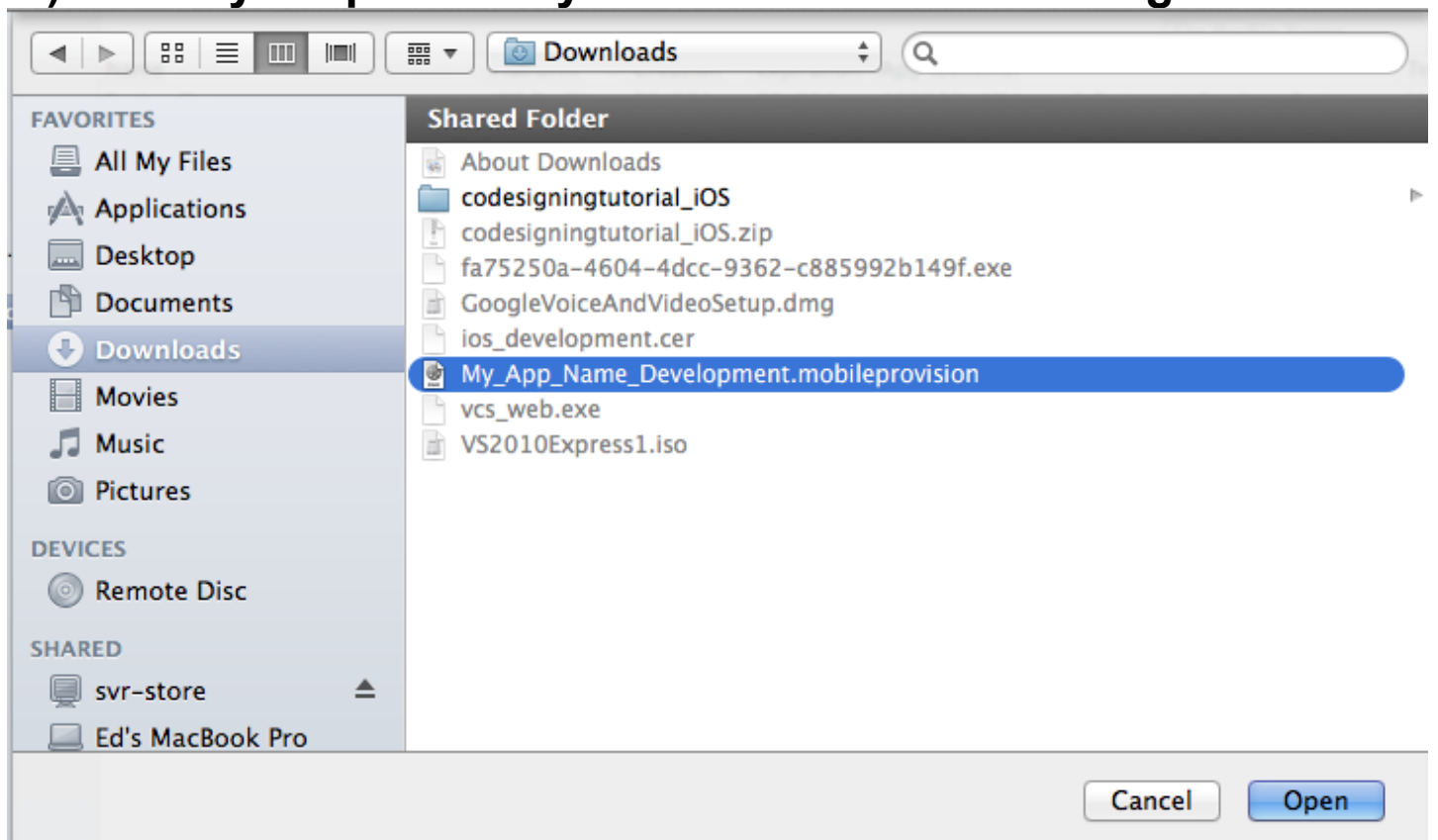**5f) Connect your device and head over to the Organizer.**

5g) Click on your device, then Provisioning Profiles.


5h) Click Add.
5i) Select your previously downloaded Provisioning Profiles.

This process will sync the profiles to your device. If they don't sync, or there is a red circle next to your device, click the

device name and click 'Use for Development'. Then go from 5h again.
5j) Close off the Organizer and go back to your project in Xcode. Change the target from Simulator to your device.
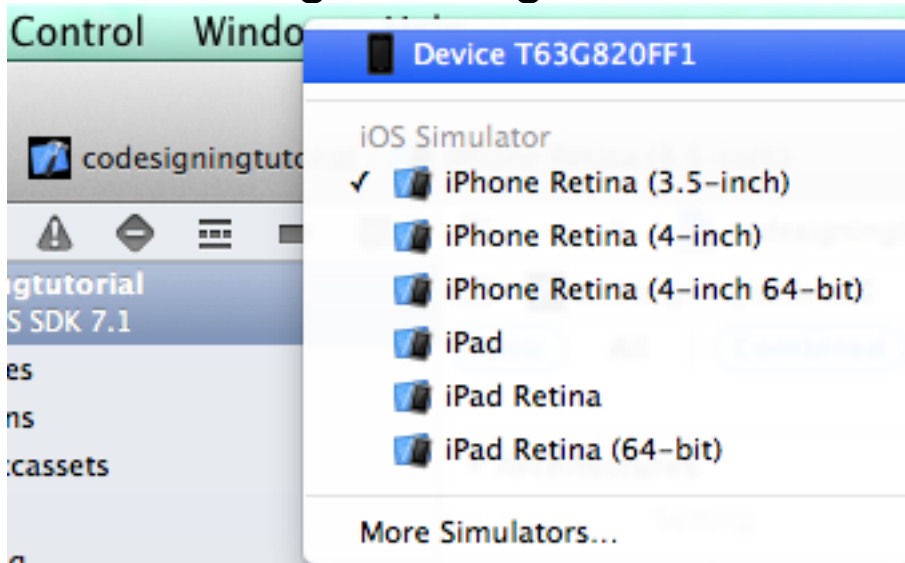
5k) Unlock your device with its passcode (or just swipe to unlock) and press Run. Your app should now run on your device. To keep it there, just disconnect your device after running.

When distributing, use the Distribution profile, and change the Code Signing Identities in Step 5e.