

So far, you've run the app on the Simulator. That's nice and all but probably not why you're learning iOS development. You want to make apps that run on real iPhones and iPads! There's hardly a thing more exciting than running an app that you made on your own phone. And, of course, to show the fruits of your labor to other people!

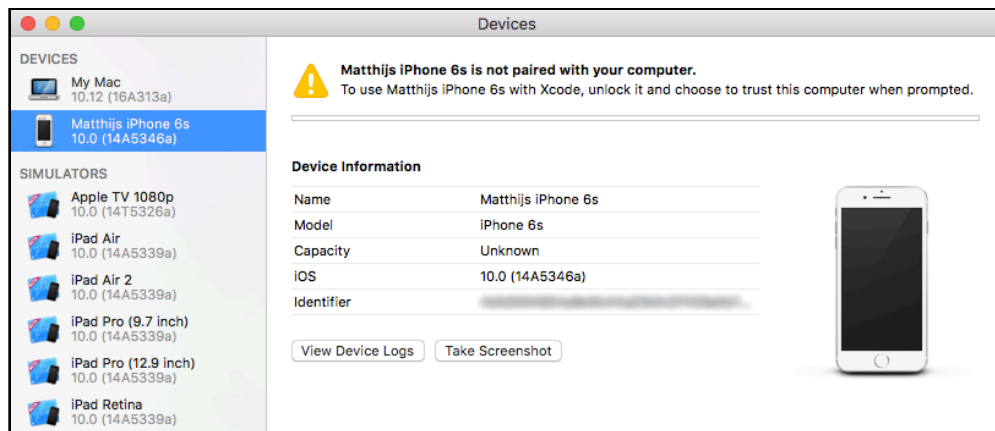
Don't get me wrong: developing your apps on the Simulator works very well. When developing, I spend most of my time with the Simulator and only test the app on my iPhone every so often.

The Simulator is great, but you do need to run your creations on a real device in order to test them properly. Some things the Simulator simply cannot do. If your app needs the iPhone's accelerometer, for example, you have no choice but to test that functionality on an actual device. Don't sit there and shake your Mac!

Until recently you needed a paid Developer Program account to run apps on your iPhone. Since Xcode 7, however, you can do it for free. All you need is an Apple ID. And Xcode 8 makes it easier than ever before.

- Connect your iPhone, iPod touch, or iPad to your Mac using the USB cable.
- From the Xcode menu bar select **Window** → **Devices** to open the Devices window.

Mine looks like this (I'm using an iPhone 6s):



The Xcode Devices window

On the left is a list of devices that can be used for development.

- Click your device name to select it.

If this is the first time you're using the device with Xcode, the Devices window will say something like, "iPhone is not paired with your computer." To pair the device with Xcode, you need to unlock the device first (hold the home button). After unlocking, an alert will pop up on the device asking you to trust the computer you're trying to pair with. Tap on **Trust** to continue.

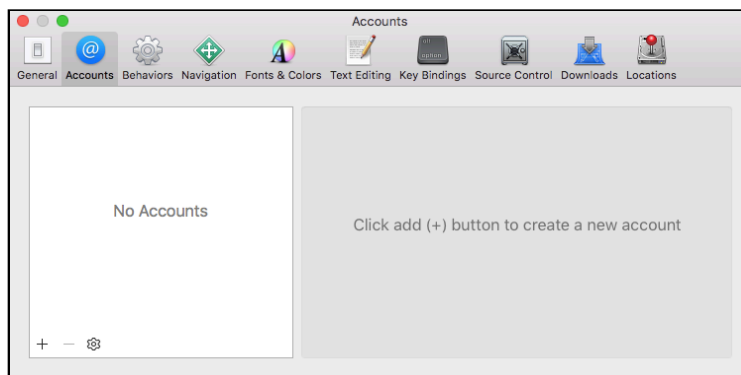
Xcode will now refresh the page and let you use the device for development. Give it a few minutes (see the progress bar in the main Xcode window). If it takes too long, you may need to unplug the device and plug it back in first.

At this point it's possible to get the error message, "An error was encountered while enabling development on this device." You'll need to unplug the device and reboot it. Make sure to restart Xcode before you reconnect the device.

Cool, that is the device sorted.

The next step is setting up your Apple ID with Xcode. It's OK to use the same Apple ID that you're already using with iTunes and your iPhone, but if you run a business you might want to create a new Apple ID to keep these things separate. Of course, if you've already registered for a paid Developer Program account, you should use that Apple ID.

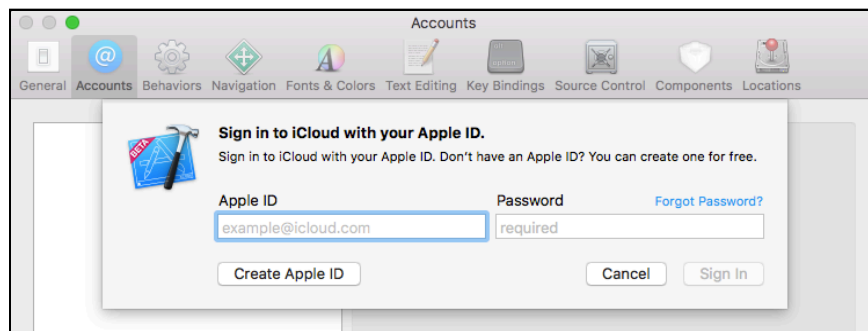
► Open the **Accounts** pane in the Xcode Preferences window:



The Accounts preferences

► Click the **+** button at the bottom and choose **Add Apple ID**.

Xcode will ask for your Apple ID:



Adding your Apple ID to Xcode

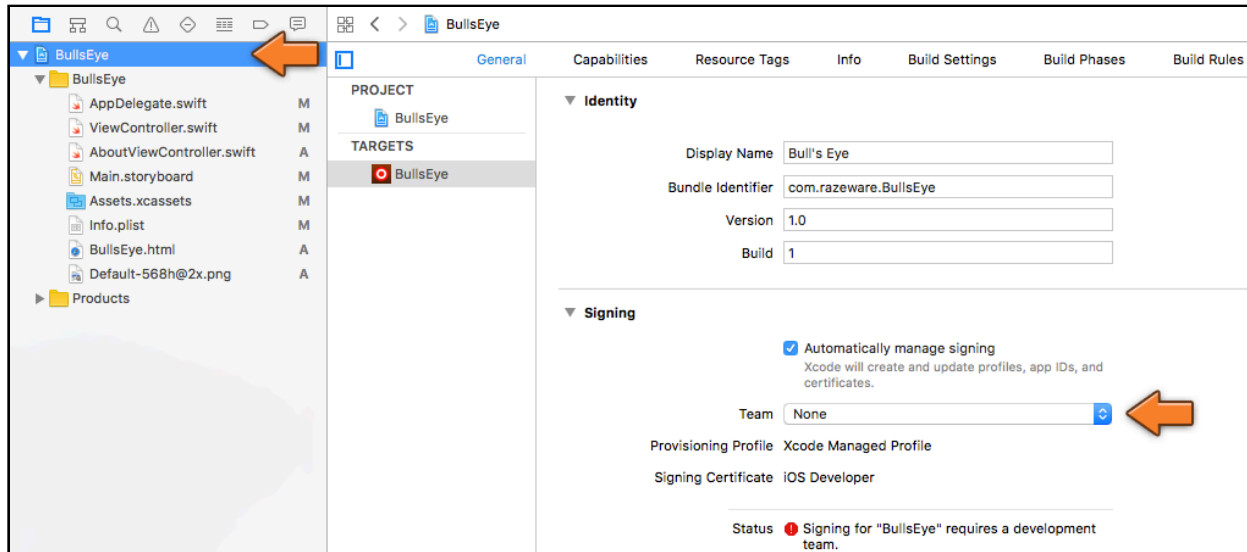
► Type your Apple ID username and password and click **Sign In**.

Xcode verifies your account details and adds them to the accounts window.

Note: It's possible that Xcode is unable to use the Apple ID you provided, for example if it has been used with a Developer Program account in the past that is now expired. The simplest solution is to make a new Apple ID. It's free and only takes a few minutes. <https://appleid.apple.com>

You still need to tell Xcode to use this account when building your app.

► Go to the **Project Settings** screen. In the **General** tab go to the **Signing** section.



The Signing options in the Project Settings screen

In order to allow Xcode to put an app on your iPhone, the app must be *digitally signed* with your **Development Certificate**. A *certificate* is an electronic document that identifies you as an iOS application developer and is valid only for a limited amount of time.

Apps that you want to submit to the App Store must be signed with another certificate, the **Distribution Certificate**. To use the distribution certificate you must be a member of the paid Developer Program but using the development certificate is free.

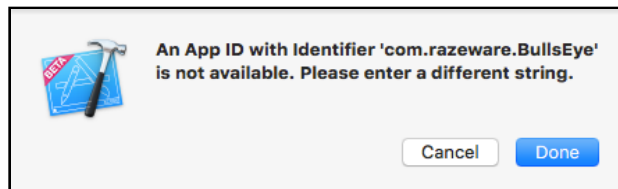
In addition to a valid certificate, you also need a so-called **Provisioning Profile** for each app you make. Xcode uses this profile to sign the app for use on your device. The specifics don't really matter, just know that you need a provisioning profile or the app won't go on your device.

Making the certificates and provisioning profiles used to be frustrating and error-prone. Fortunately, those days are over: Xcode 8 makes it really easy. When the **Automatically manage signing** option is enabled, Xcode will take care of all this business with certificates and provisioning profiles and you don't have to worry about a thing.

► Click on **Team** to select your Apple ID.

Xcode will now automatically register your device with your account, create a new Development Certificate, and download and install the Provisioning Profile on your device. These are all steps you would have had to do by hand in the past but now Xcode 8 takes care of all that.

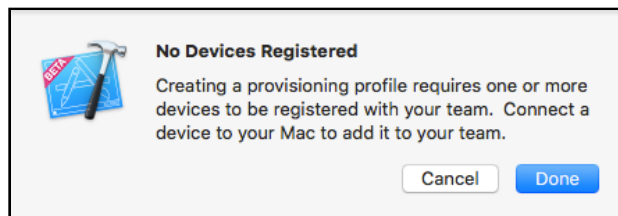
It's possible you get the following error:



The bundle identifier is already in use

The app's Bundle Identifier – or App ID as it's called here – must be unique. If another app is already using that identifier, then you cannot use it anymore. That's why you're supposed to start the Bundle ID with your own domain name. The fix is easy: change the Bundle Identifier field to something else and try again.

It's also possible you get this error:

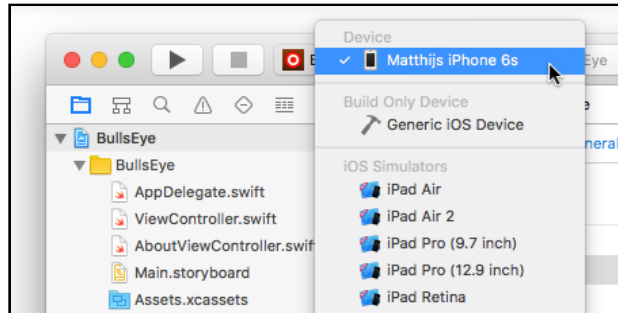


No devices registered

Xcode must know about the device that you're going to run the app on. That's why I asked you to connect your device first. Double-check that your iPhone or iPad is still connected to your Mac and that it is listed in the Devices window.

If all of that checks out, go back to Xcode's main window and click on the box in the toolbar to change where you will run the app. The name of your device should be in that list somewhere.

On my system it looks like this:



Changing where the app will be run

You're all set up and ready to go!

► Press **Run** to launch the app.

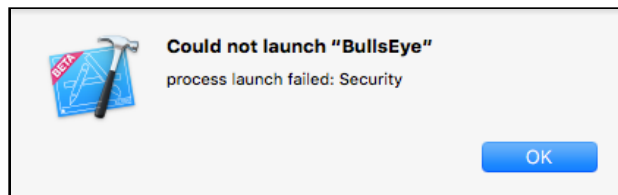
At this point you may get a popup with the question "codesign wants to sign using key ... in your keychain". If so, answer with **Always Allow**. This is Xcode trying to use the new Development Certificate you just created but you need to give it permission first.

Does the app work? Awesome! If not, read on...

There are a few things that can go wrong when you try to put the app on your device, especially if you've never done this before, so don't panic if you run into problems.

The device is not connected. Make sure your iPhone, iPod touch, or iPad is connected to your Mac. The device must be listed in Xcode's Devices window and there should not be a yellow warning icon.

The device does not trust you. You might get this warning:

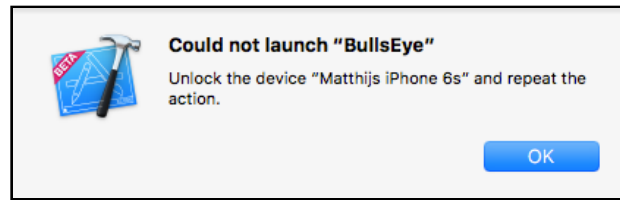


Quick, call security!

On the device itself there will be a popup with the text, "Untrusted Developer. Your device management settings do not allow using apps from developer ...".

If this happens, open the Settings app on the device and go to General, Profile. Your Apple ID should be listed in that screen. Tap it, followed by the Trust button. Then try running the app again.

The device is locked. If your phone locks itself with a passcode after a few minutes, you might get this warning:

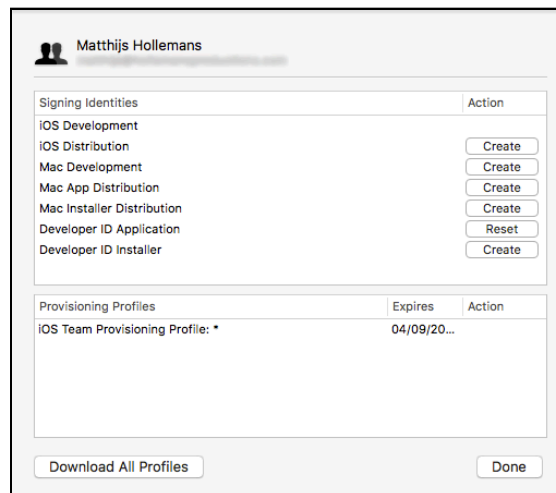


The app won't run if the device is locked

Simply unlock your device (hold the home button or type in the 4-digit passcode) and press Run again.

If you're curious about these certificates and provisioning profiles, then open the **Preferences** window and go to the **Accounts** tab. Select your account and click the **View Details...** button in the bottom-right corner.

This brings up another panel, listing your signing identities (the certificates) and the provisioning profiles:

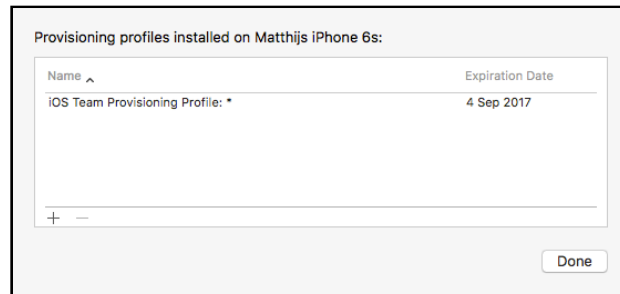


The account details panel

The "iOS Team Provisioning Profile: *" is the thing that allows you to run the app on your device. (By the way, they call it the "team" profile because often there is more than one developer working on an app and they can all share the profile.)

When you're done, close the Accounts window and go to the Devices window.

You can see the provisioning profiles that are installed on your device by right-clicking the device name and choosing **Show Provisioning Profiles**:

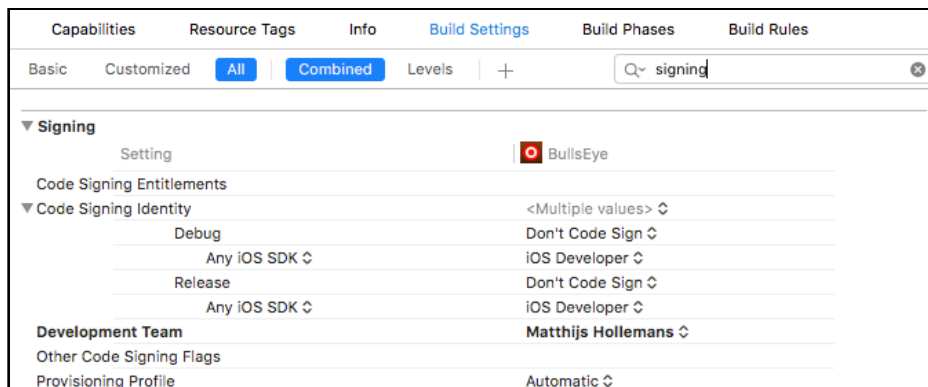


The provisioning profiles on your device

You can have more than one certificate and provisioning profile installed. This is useful if you're on multiple development teams or if you prefer to manage the provisioning profiles for different apps by hand.

To see how Xcode chooses which profile and certificate to sign your app with, go to the Project Settings screen and switch to the **Build Settings** tab. There are a lot of settings in this list, so filter them by typing **signing** in the search box. (Also make sure **All** is selected, not Basic.)

The screen will look something like this:



The Code Signing settings

Under **Code Signing Identity** it says **iOS Developer**. This is the certificate that Xcode uses to sign the app. If you click on that line, you can choose another certificate. Under **Provisioning Profile** you can change the active profile. Most of the time you won't need to change these settings, but at least you know where to find them now.

The end... or the beginning?

This has been a very long lesson – if you're new to programming, you've had to get a lot of new concepts into your head. I hope your brain didn't explode!

At least you should have gotten some insight into what it takes to develop an app.