Mac OS X Setup Checklist

You must be running Mac OS X 10.10 (Yosemite) or above.

1. Install Xcode. You can either download Xcode from the developer.apple.com site if you have a developer account, or from the Mac AppStore.
2. Install Xamarin tools onto your Mac. You can download the Xamarin unified installer from xamarin.com/download. The installer will automatically download any missing tools and SDKs needed for Android development.
    a. Note: this step will likely take a while depending on what is already installed on the computer since it has to download fairly large SDK installs to configure the machine.
3. Launch Xamarin Studio - it will be in your application folder, or you can use Spotlight to locate and launch the application.
4. Verify you are on the *stable* branch (this is the default). You can check this through the Xamarin Studio > Check For Updates system menu option.

Setup iOS Simulator

The iOS Simulator is built into Xcode and will be automatically launched by Xamarin Studio when necessary. You can choose which simulator to launch (iPhone vs. iPad, and iPhone variations) through the toolbar in Xamarin Studio. You can also deploy to a physical device as long as it has been configured to be used as a development device through Xcode.

Setup Android Emulators

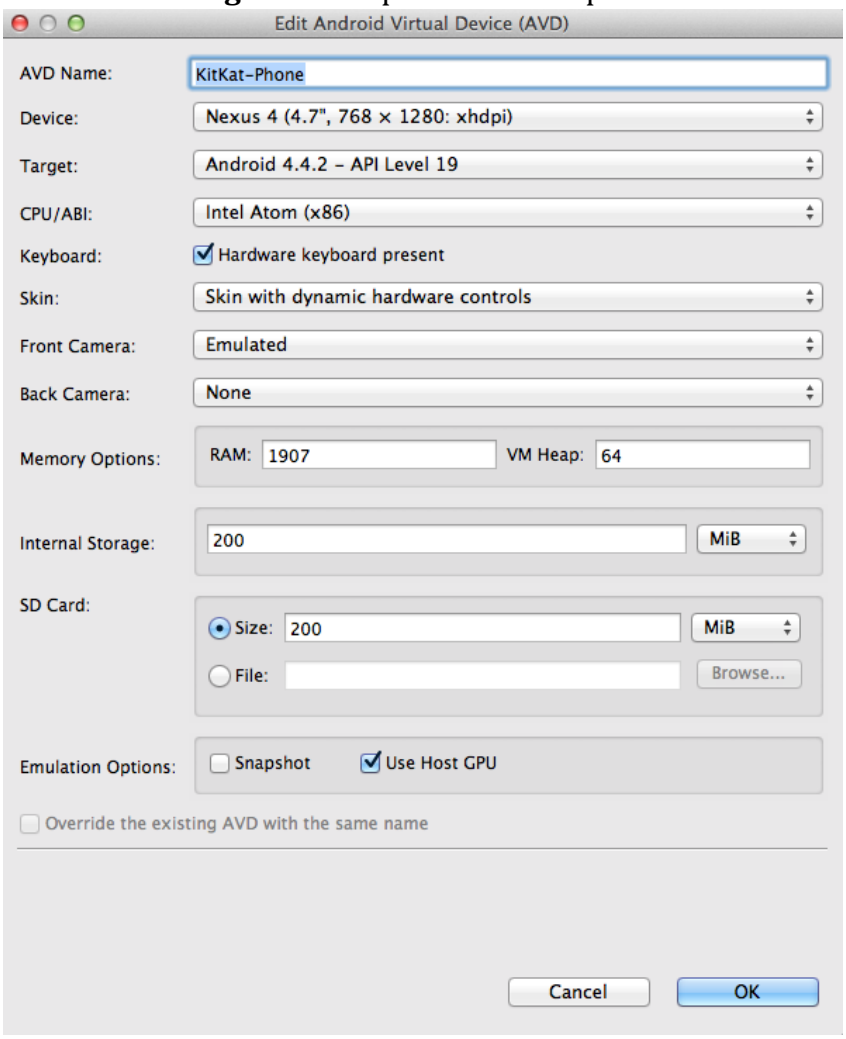
To test Android applications, you can either deploy to a physical device (preferred), or use an Android emulator. If you plan to use an emulator, we recommend you use **Xamarin Android Player** as it delivers higher performance over the Android SDK emulator.

1. Download the Xamarin Android Player install from www.xamarn.com/android-player, for Mac OS X you can download the standalone package, which includes Virtual Box.
2. Once installed, run the application and download a pre-configured image, there are several to choose from.
3. Launch the image from the Xamarin Android Player app by selecting it in the list and clicking the Play button.
4. It should then show up in the emulators list in Xamarin or Visual Studio.

Android SDK Emulators

The Xamarin installer will create a few emulated devices for you, but you will want to adjust these and create one with a newer version of Android. We recommend a phone device running at least 4.4.2.

To create or adjust an Android emulator, you must run the Android SDK Emulator Manager. This can be started in Xamarin Studio through **Tools > Open Android Emulator Manager**, or in Visual Studio with **Tools > Android > Open Android Emulator Manager** menu option. An example definition is shown below:



Out of the box Google Android Emulators are very slow. To improve this, you can install the **Intel HAXM Drivers.** This additional install provides hardware acceleration for x86-based emulators on Intel VT-enabled systems. The HAXM drivers are free to use and published by Intel. If you are using the Android SDK emulator, this is a must.

1. Close any running emulators

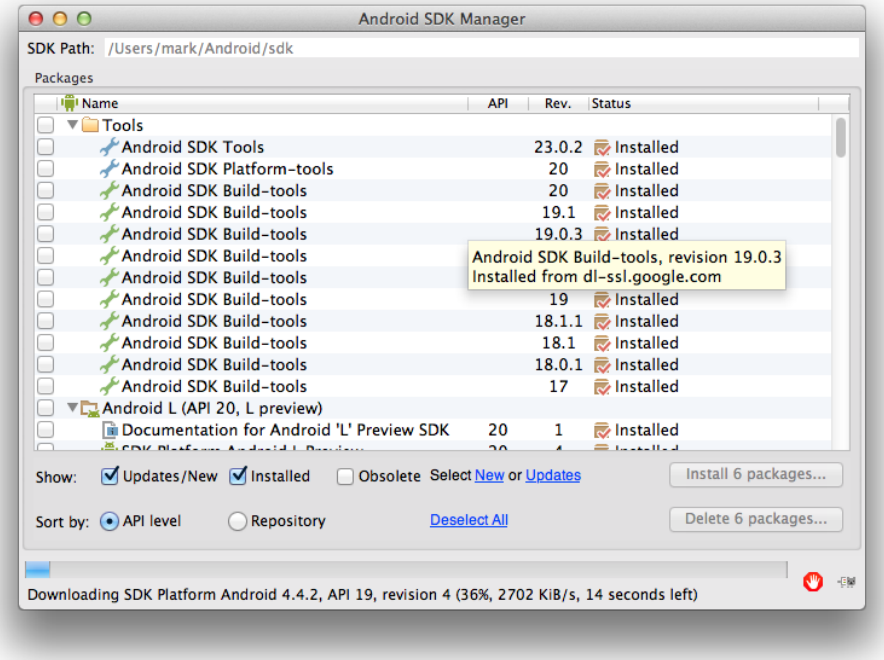
2. From the Intel website, download the latest HAXM virtualization engine from https://software.intel.com/en-us/android/articles/intel-hardware-accelerated-execution-manager.
3. Install the HAXM engine, and restart your computer if prompted.
4. Change any Android image you have created to use the Intel Atom (x86) CPU image as shown in the above screenshot – the acceleration only works for x86 images.

Warning!

HAXM run simultaneously with VirtualBox can cause stability issues. Both can exist together on the same machine, but it is best not to run both Xamarin Android Player and HAXM-based emulators at the same time.

Verify your Setup

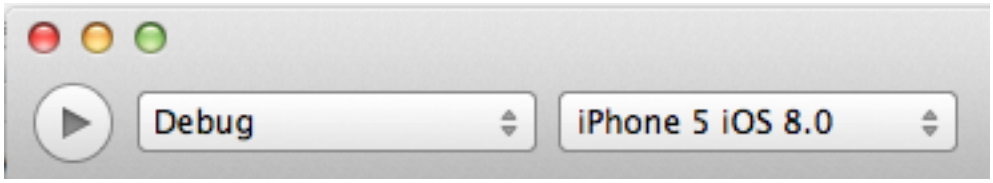
1. Launch Xamarin Studio, it is in your applications folder (or you can use Spotlight to quickly open it).
2. The IDE should prompt you to either start a trial with Xamarin, or enter to your registered Xamarin account. Go ahead and do one of these two steps so you be able to build full applications, including the T-shirt test app.
3. Verify that you have all the Android SDK versions you will need. We recommend installing 4.0 through 4.4 (or even Android "L" preview if you want to try some of the new features). You can get to the Android SDK options using the **Tools > Open Android SDK Manager** menu option.



4. Download the Xamarin T-shirt application from https://xamarin.com/c-sharp-shirt, unzip it onto your desktop (or some other easily accessible location) and open the solution.
5. Open the solution (.sln) using Xamarin Studio and following the instructions below for iOS, Android or both.

Testing iOS on OS X with Xamarin Studio

1. Make sure the **XamarinStore.iOS** project is the active project in Xamarin Studio (you can right-click and choose *Set As Startup Project* to change it if necessary).
2. Select the **Debug** configuration and one of the **iOS Simulator** choices in the toolbar, for example the iPhone 5 iOS 8.0 choice:



1. Click the "Run" button in the toolbar (looks like a "Play" icon).
2. It should build the application and then run it in the simulator.

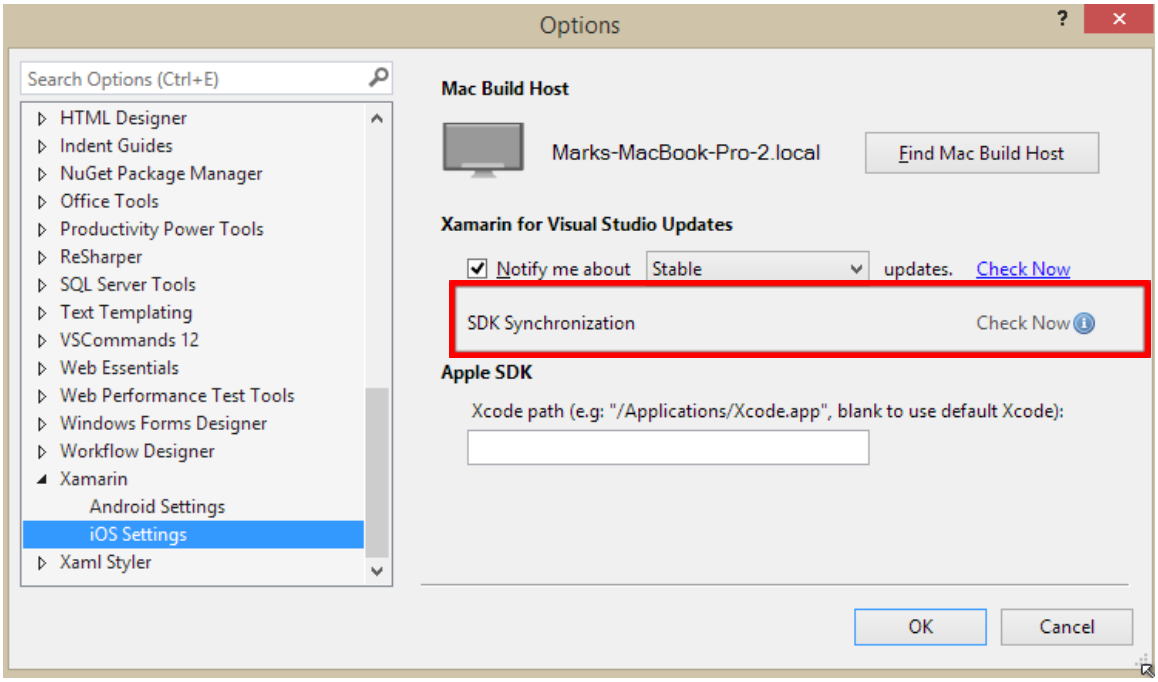
Testing Xamarin.Android on OS X with Xamarin Studio

1. Make sure the **XamarinStore.Droid** project is the active project in Xamarin Studio (you can right-click and choose *Set As Startup Project* to change it if necessary).
2. Select the **Debug** configuration and one of the choices in the emulator drop-down. If you don't see any choices, or you would like to configure the choices available, use the **Tools > Open Android Emulator Manager** to work directly with the Android SDK tool to create or delete emulators as described above.
3. Click the "Run" button in the toolbar (looks like a "Play" icon). It should build the application and then run it in the selected emulator.

Troubleshooting

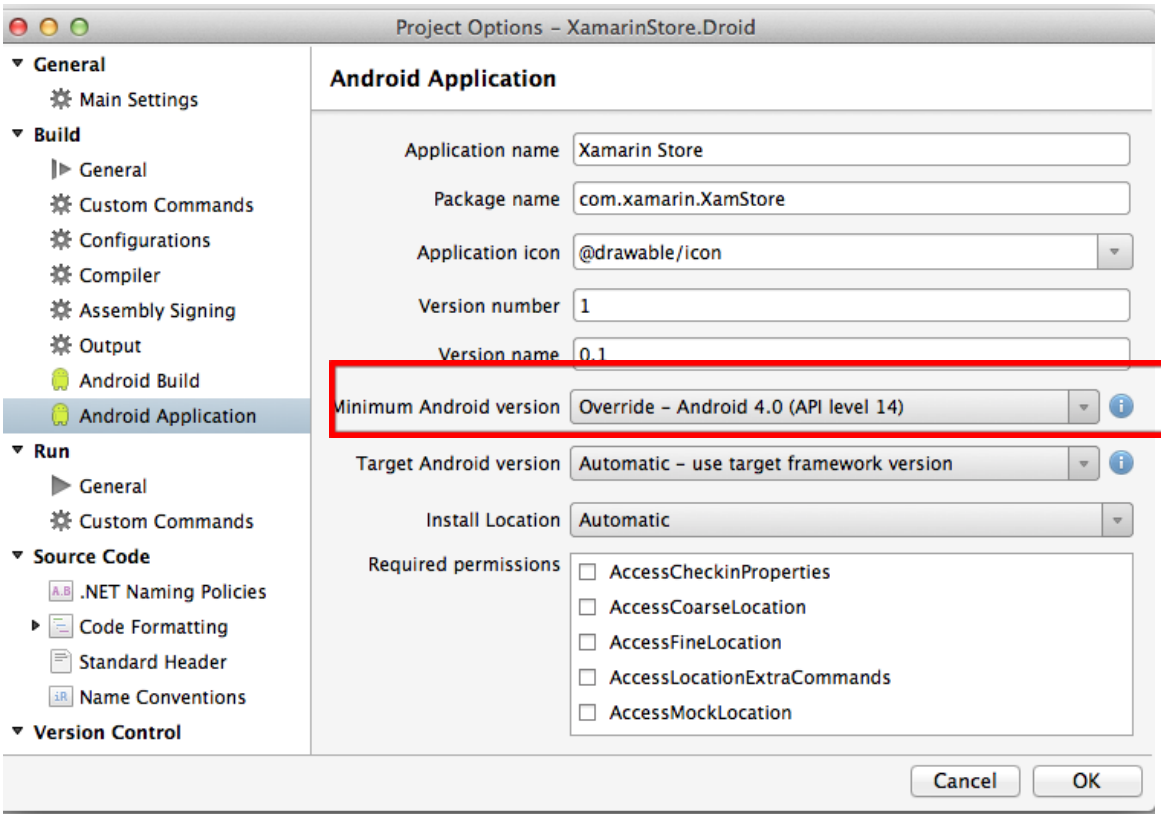
The most common issues encountered are: Missing SDK versions and Android emulator issues.

If the iOS Application fails to build on Visual Studio verify that the iOS SDK on the Mac and the iOS SDK on the Windows machine are the same – this means you are running in the same Xamarin channel (we recommend *stable*) and the two machines are synchronized. You can check this in the **Tools > Options > Xamarin > iOS** dialog – click the "Check Now" option next to SDK Synchronization.



If the Android application fails to build on either environment, then verify you have the proper Android SDK versions installed. You will often get an error that reads something like "The required Android SDK x.xx is missing…".

The T-Shirt application is setup to require Android 4.0, but you can change it to a newer version through the project options (just double-click on the XamarinStore.Droid project, or right click and select Properties). The version is in the **Android Application** section in Xamarin and Visual Studio:



Alternatively, you can use the Android SDK Manager to download the required SDK libraries for 4.0 and then you should be able to build the application.

For emulator issues, we recommend launching the Android emulator before you run your program and then *keep it running*. If Xamarin Studio fails to detect the emulator, then close and restart it. It should detect it on launch and then the emulator will show up in the available choices in the toolbar.