Keith Erkert

kerkert@asu.edu

Updated 05/26/2022

Abstract

This guide will walk the user to setup a development environment for the phone application. The phone application uses Xamarin Forms as a cross platform solution. This guide is geared to development on a Windows 10+ based PC and using a Mac as a middle man for iOS development.

Development Setup

Table of Contents

[Introduction 2](#_Toc104885144)

[Hardware/Software Requirements 2](#_Toc104885145)

[Visual Studio Setup (for Windows) 2](#_Toc104885146)

[XCode Setup (for Mac) 4](#_Toc104885147)

[Pairing XCode/Mac with Visual Studios 7](#_Toc104885148)

[Deploying to iPhone First Time Setup 8](#_Toc104885149)

[Deploying to Android 13](#_Toc104885150)

[Running Simulators 15](#_Toc104885151)

# Introduction

This development guide is for Windows 10+ based machines. For iOS development from your windows machine, a virtual machine (or other Mac device on same local area network) with Mac Monterey (v12+) is needed for free deployment (no Apple Developer account needed). For Android development, only the windows pc is needed.

# Hardware/Software Requirements

* Windows 10+ machine (10.0.18362+, 2019 recommended), 16+ GB of RAM, 16+ GB HDD space for all software (excluding VMs)
* Android 4.4 minimum for Android development (this guide using Android 9, but is not needed; some tweaking may be needed)

For iOS development (from windows pc)

* iOS 9 minimum for iPhone development (this guide uses iOS 14, but is not needed; some tweaking may be needed)
* Mac (12+) running on either a physical device on your network (same as windows pc), or in a virtual machine on your windows PC (this guide does not cover how to run a Mac on a VM, due to legal reasons)

# Visual Studio Setup (for Windows)

1. Download the latest version of Visual Studio <https://visualstudio.microsoft.com/vs/> (2022 was used at the time of writing)
2. Select the following based off what you want to work on (select multiple if needed)

**Web API** – ASP.NET and web development

**Backend Helper App** - .NET desktop development

**Android/iOS App** – Mobile Development with .NET

1. Once the installation is complete, open visual studio

Graphical user interface, text

Description automatically generated

1. Insert the following you wish to modify under the ‘Repository location’

**Web API** – <https://github.com/ASU-IoT-ResearchProject/IotBackendAPI.git>

**Backend Helper App** - <https://github.com/ASU-IoT-ResearchProject/IoTBackendHelper.git>

**Android/iOS App** – <https://github.com/ASU-IoT-ResearchProject/IoTCrossPlatform.git>

**Raspberry Pi App** – <https://github.com/ASU-IoT-ResearchProject/IoTRaspberryPi.git>

# XCode Setup (for Mac)

* This is needed for deployment to iPhones for free. (If you have an Apple Developer account, a Mac should not be needed)
* macOS Monterey is recommended (latest version of visual studio requires , however older version of visual studio can use older macOS versions)

1. Download XCode 13+ from the App Store

Graphical user interface, application, website

Description automatically generated

1. Turn on remote login

Graphical user interface, application

Description automatically generated

1. Once the installation is complete, open visual studio

Graphical user interface, text

Description automatically generated

1. If prompted, allow your firewall to accept incoming connections for remote login

**Graphical user interface, text

Description automatically generated**

1. Your mac should now be discoverable by visual studio on your windows pc (if on same network)

# Pairing XCode/Mac with Visual Studios

This is only needed if using a windows pc to code on in visual studio.

1. Open the IoTCrossPlatform project in Visual Studio
2. Click the pair to mac button

Qr code

Description automatically generated

1. Visual Studio should auto discover the mac, if it does not click Add Mac and manually type the local ip address

Graphical user interface, application

Description automatically generated

1. Insert your mac user credentials to create a secure connection to the mac

# Deploying to iPhone First Time Setup

**Apple Developers Account (Paid Option)**

1. Right click IotCrossPlatform.iOS > select Properties > go to iOS Bundle Signing

Graphical user interface, text, application, chat or text message

Description automatically generated

1. Choose Automatic Provisioning
2. Add Account and sign in to your apple developer account

**Free Option (no Apple developer account needed)**

* App certificates expires after 7 days (every 7 days, developer needs to re-deploy dummy app to update the certificate, then redeploy our app)
* iPhone needs to be tricked to run our app

1. On Mac, Open XCode 13 and create a new project
2. For the template, choose App and click next

Graphical user interface, application

Description automatically generated

1. Keep all of the properties default and give the app any name you want such as IoTDummyApp.
2. In the title bar go to Xcode > Preferences > Accounts and add a new account by clicking the +
3. Choose Apple ID and click continue and sign in using your regular apple account
4. With your Personal Team selected > Manage Certificates > + > Apple Development, this will create a certificate for your app to use, close this window
5. Double Click the top most Folder in the tree view Graphical user interface, application

   Description automatically generated
6. Go to signing and capabilities and change the Bundle Identifier and change it to our app identifier to “com.companyname.IoTCrossPlatform”

Graphical user interface, text, application

Description automatically generated

1. By default XCode sets the iOS target version to the latest version, go to the General tab > and click the version number and change its to whatever version your iphone is using (minimum is 13+ for Swift language, which the default settings uses)

Graphical user interface, application

Description automatically generated

1. Change the deployment target to your connected iPhone (the iPhone should be connected to your mac not windows pc. If using a macOS in VM, make sure it is connected to guest, not host)
2. At the top of the XCode window click the iPhone/iPod simulator text and select your connected phone

Graphical user interface, application

Description automatically generated

1. Click the play button

Graphical user interface, application

Description automatically generated

1. XCode may prompt you for a password (your mac sign-in password not apple account password)

*If this is your first time deploying an app, the deployment will fail because your iPhone must trust your personal developer certificate.*

1. On you iPhone go to Settings > General > VPN & Device Management > Your Developer Certificate (should have your email) and trust the app/certificate
2. Redeploy to your phone and it should show up now
3. Every 7 days this app will need to be redeployed to update the certificate, there is no free way to workaround this.

Deploying our App (IoTCrossplatform project)

1. Back in Visual Studio, choose you iPhone from the dropdown deployment options

Graphical user interface, application

Description automatically generated

*If you do not see your iPhone, try restarting visual studio, and verify your mac is connected to it.*

1. Press the deploy button (filled in green arrow) to run the app

*building to your iPhone may take a minute or two and it uses the mac as a middleman to deploy to your iPhone.*

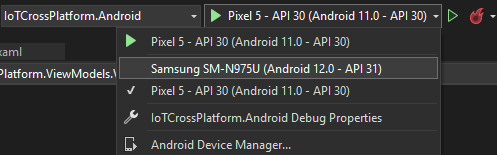
# Deploying to Android

1. Enable developer mode for your phone, each version of android is slightly different, please google your phone make/model to find the exact way to enable it.
2. Under the settings menu go to the Developer Options and toggle on Developer Options and scroll down to USB debugging and toggle that on

Graphical user interface, application

Description automatically generated

1. Plug your Android phone into your computer with Visual Studio on it. Accept any prompt on your phone asking you to verify you trust your phone.
2. In Visual Studio select the Android version of the app and click the dropdown on the right to select your phone.



1. And click the deploy button (filled green in arrow) to deploy to your android phone

# Running Simulators

To Be Added (having some issue currently)