**To build for Android GearVR on Unity:**

1. In Unity go to **File->Build Settings** Find **Android** in the **Platform** list and download the **Android Module** if you haven't already.
2. Once that is downloaded, set Android as the Platform and click '**Switch Platform**'.
3. Check **Development Build**
4. Then click the '**Player Settings**' button and check your **Inspector**. Here you will have to set a **Company Name** and **Product Name** (Don't use spaces). *These can by anything that follows the Java package naming convention. See*[*this*](https://developer.android.com/guide/topics/manifest/manifest-element#package)*if you need help with that.*
5. Below open up **Other Settings** and change the **Package Name** from its default value to whatever you entered for Company Name and Product Name in the previous step. They must match what you used exactly.
6. If you are developing for the **OculusGO**, **GearVR**, or **Google CardBoard** you must set the **minimum API level to API 16 - Android 4.4 'Kit-kat'**. If you are developing for **DayDream** you must set the min **API to 24**.
7. Finally, open **XR Settings** at the bottom of the list and check **Virutal Reality Supported**. Then add to the list whichever you want to develop for.  
   ***Note: For GearVR choose Oculus***

**Running the application on your device:**

1. Before you are going to be able to run the app on your device you MUST get a osig file. Follow the instructions on [this page](https://dashboard.oculus.com/tools/osig-generator/) to get that file and add it to your project's **Assets/Plugins/Android/assets** folder.  
   2.Once you have followed above instructions, when you want to test your app on your device, you will have to click **File->Build Settings->Build**.
2. Then give your apk file a name and save it.
3. Once it builds, download that file onto your device and as long as you already setup the GearVR app on your phone you'll be good to go.

**If you are having issues with your first Android apk build on Unity make sure you are doing the following:**

* Using the JDK 8 not the most recent JDK 10. There are problems with it working correctly.
* Set the locations of your Android SDK and Java JDK in Unity. *It usually can find it on its own but sometimes it doesn't work properly.*  
  In Unity: Edit->Preferences->External Tools->Android.  
  Now set your JDK/SDK locations.  
  ***Note: You must first have installed the Unity modules to build for Android or you will not see the options.***
* **If you are getting an error "....Unable to list target platforms...":** go to your Android SDK root folder (probably C:\Users\YourUsername\AppData\Local\Android\sdk) and inside there remove the 'tools' folder completely. Then download [this](http://dl-ssl.google.com/android/repository/tools_r25.2.5-windows.zip) and extract it to the root SDK folder.

**Debugging using the command prompt and Android logcat (recommended):**

* Navigate to ***C:\Users\%USER%\AppData\Local\Android\Sdk\platform-tools* (or add to your PATH)** so you can use the **adb** ([Android Debug Bridge](https://developer.android.com/studio/command-line/adb)). Then type the following commands:
  + Check to make sure your device is connected via USB
    - adb devices
  + Restart in TCP mode on a given port
    - adb tcpip [port\_number]
  + Connect to the device over Wifi
    - adb connect [device\_IP]:[port\_number]

Note: The device and the PC must be on the same wifi network. You can find the IP address of the device by going to ***Settings > About Device > Status > IP Address***

Example:

C:\Users\poppy>adb devices

List of devices attached

ce07171751eadd34027e device

C:\Users\poppy>adb tcpip 1337

restarting in TCP mode port: 1337

C:\Users\poppy>adb connect 192.168.1.27:1337

connected to 192.168.1.27:1337

C:\Users\poppy>

**Finally use this command to filter out Unity related messages (it is also recommended to save this in an environmental variable for each use):**

adb logcat -s Unity ActivityManager PackageManager dalvikvm DEBUG