

Getting Started with Unity AR Foundation AR Core Android

Requires:

Unity Version 2020.3.24f1 with Android Build Support modules

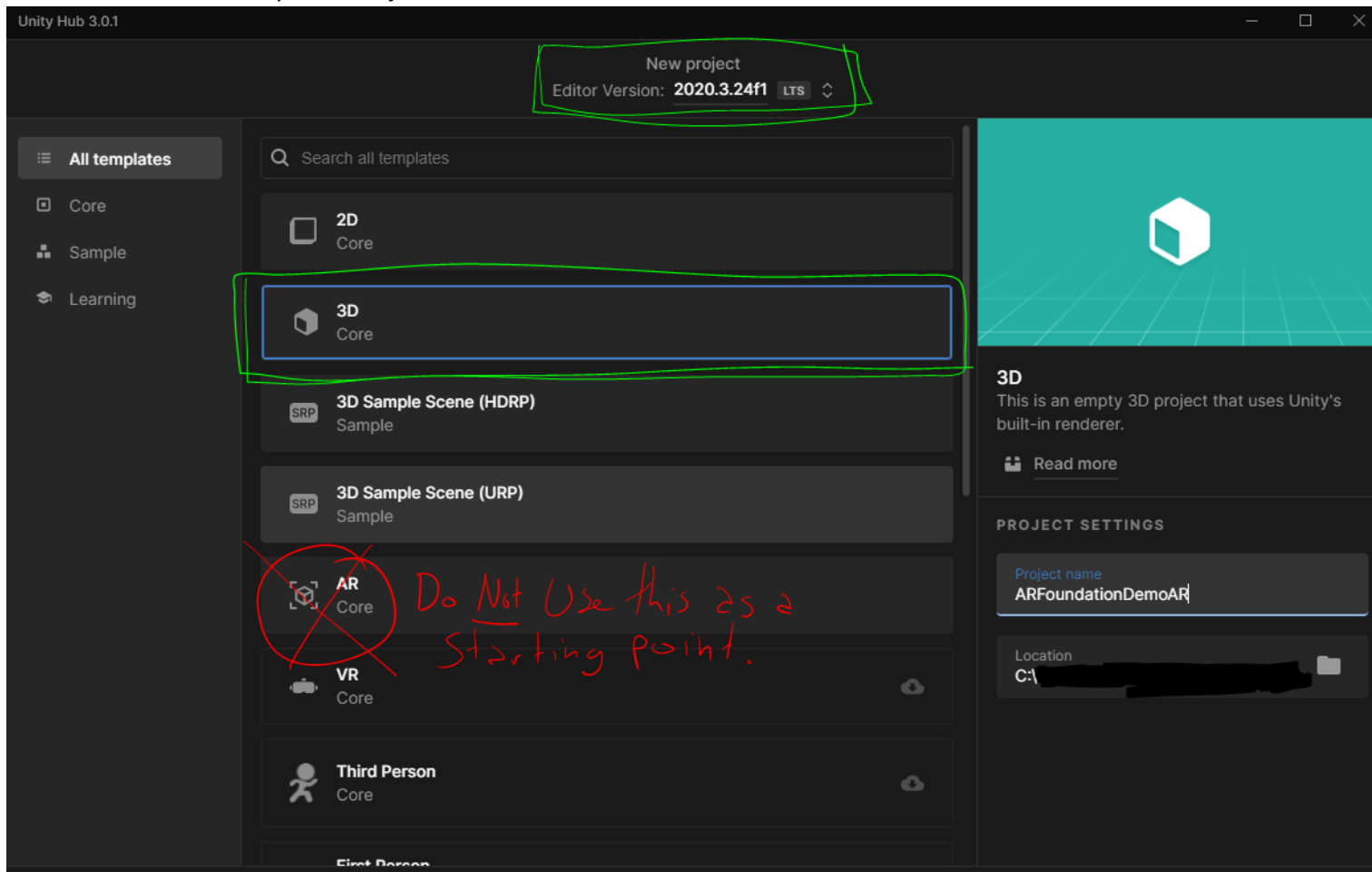
- Unity Hub -> Installs -> Settings cog next to installation -> Add Modules -> Android Build Support

Developer mode USB debugging enabled Android Device and USB connection

- Open settings on Android device, go to About Phone, go to Software Info, tap Build Number until Developer Mode unlocks
- In Developer Options enable USB debugging, allow permissions when connecting to computer

Create Project:

Start a new 3D Core Unity 2020 Project



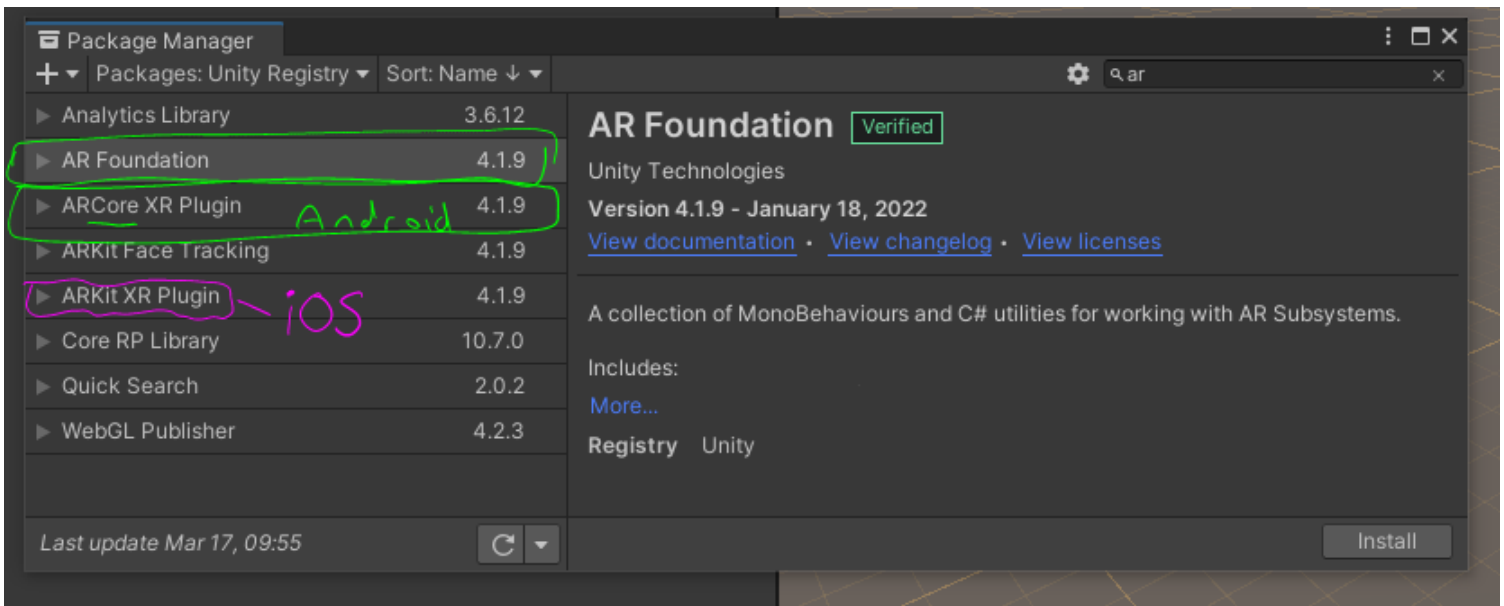
Package Management:

Optional: Remove unnecessary packages.

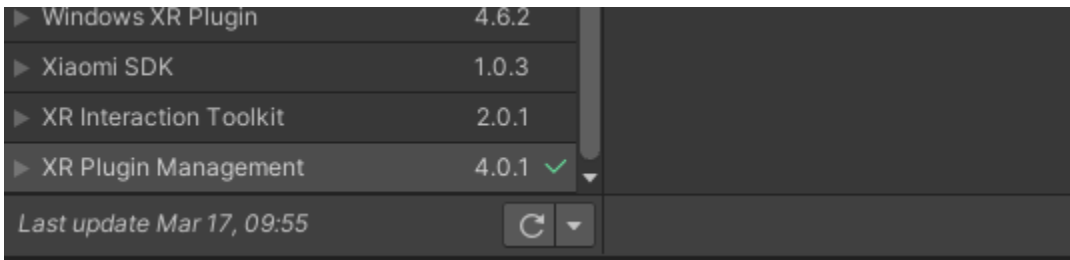
Add Required packages.

Install AR Foundation -> this brings in AR Foundation and all dependencies

Install AR Core XR Plugin -> plugin to build for Android

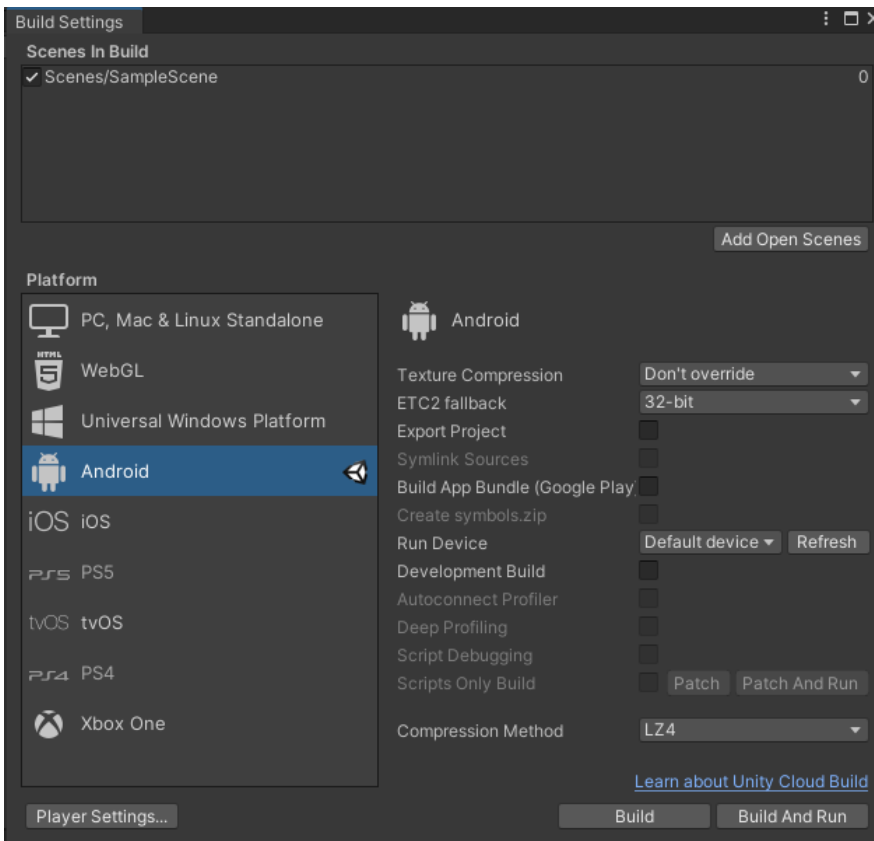


XR Plugin Management is installed with AR Foundation



Build Settings:

Switch target platform to Android

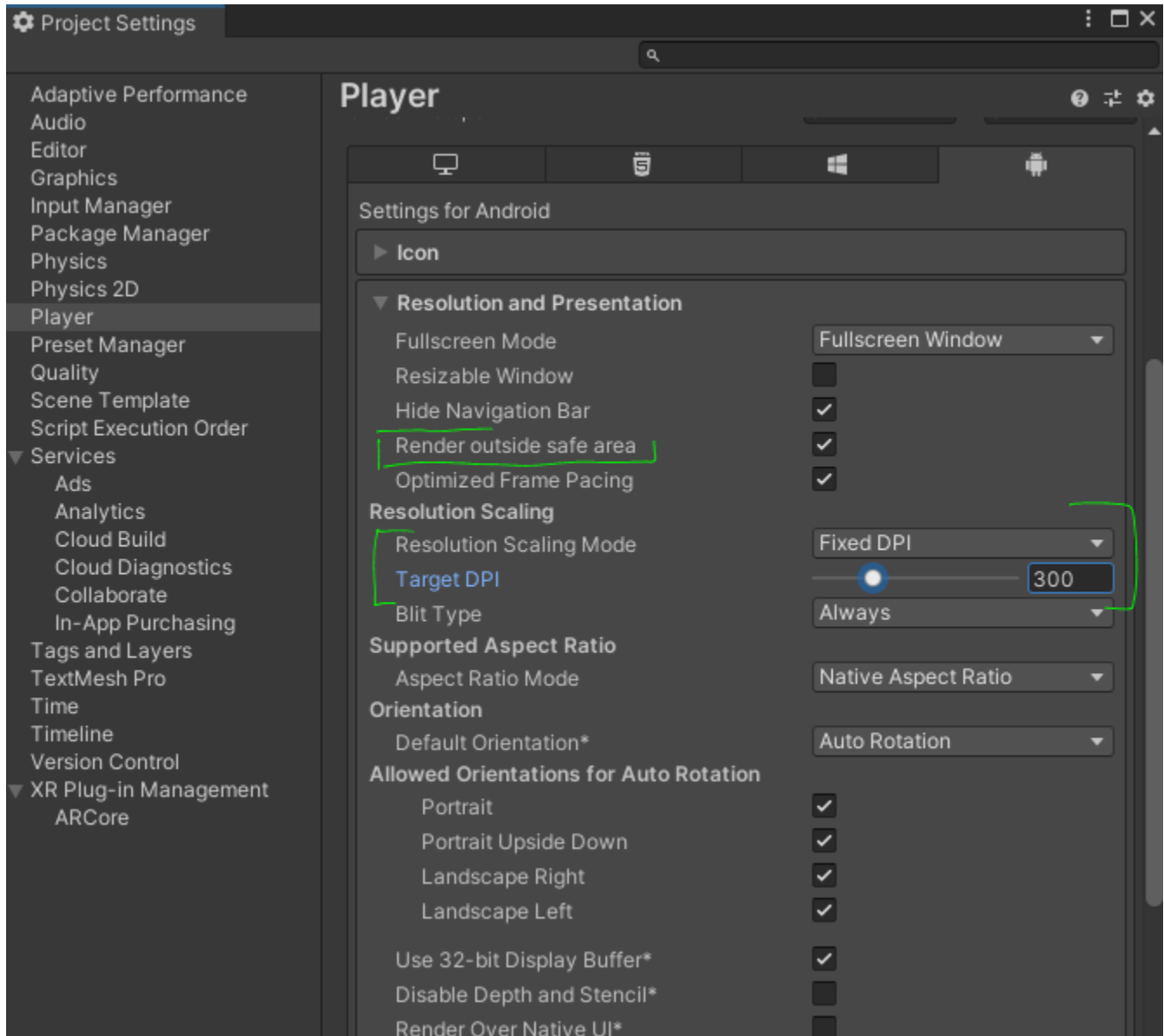


Player Settings:

Resolution and presentation:

Render outside safe area to use full phone screen, disable to not render inside of phone notches

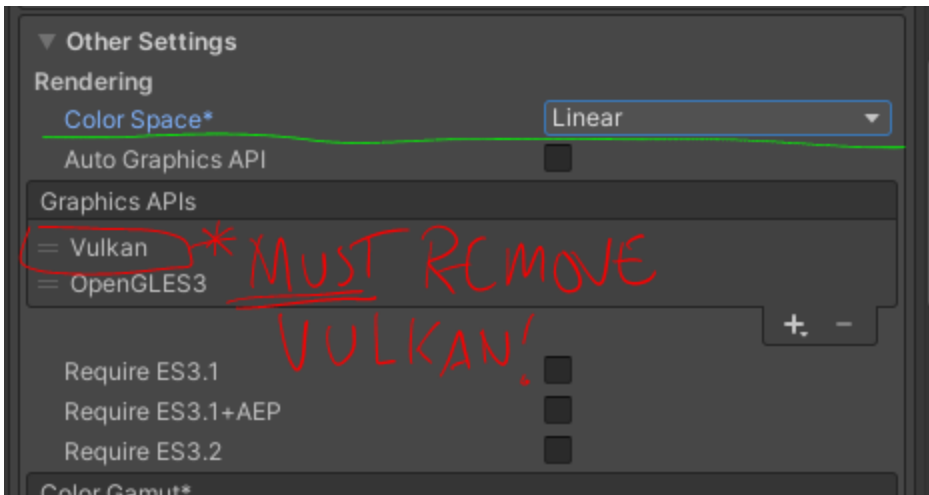
Resolution Scaling Mode -> set to Fixed DPI. Target DPI 300. This optimizes against battery drain on high resolution devices.



Other Settings:

Set Color Space to Linear

Remove Vulkan Graphics API

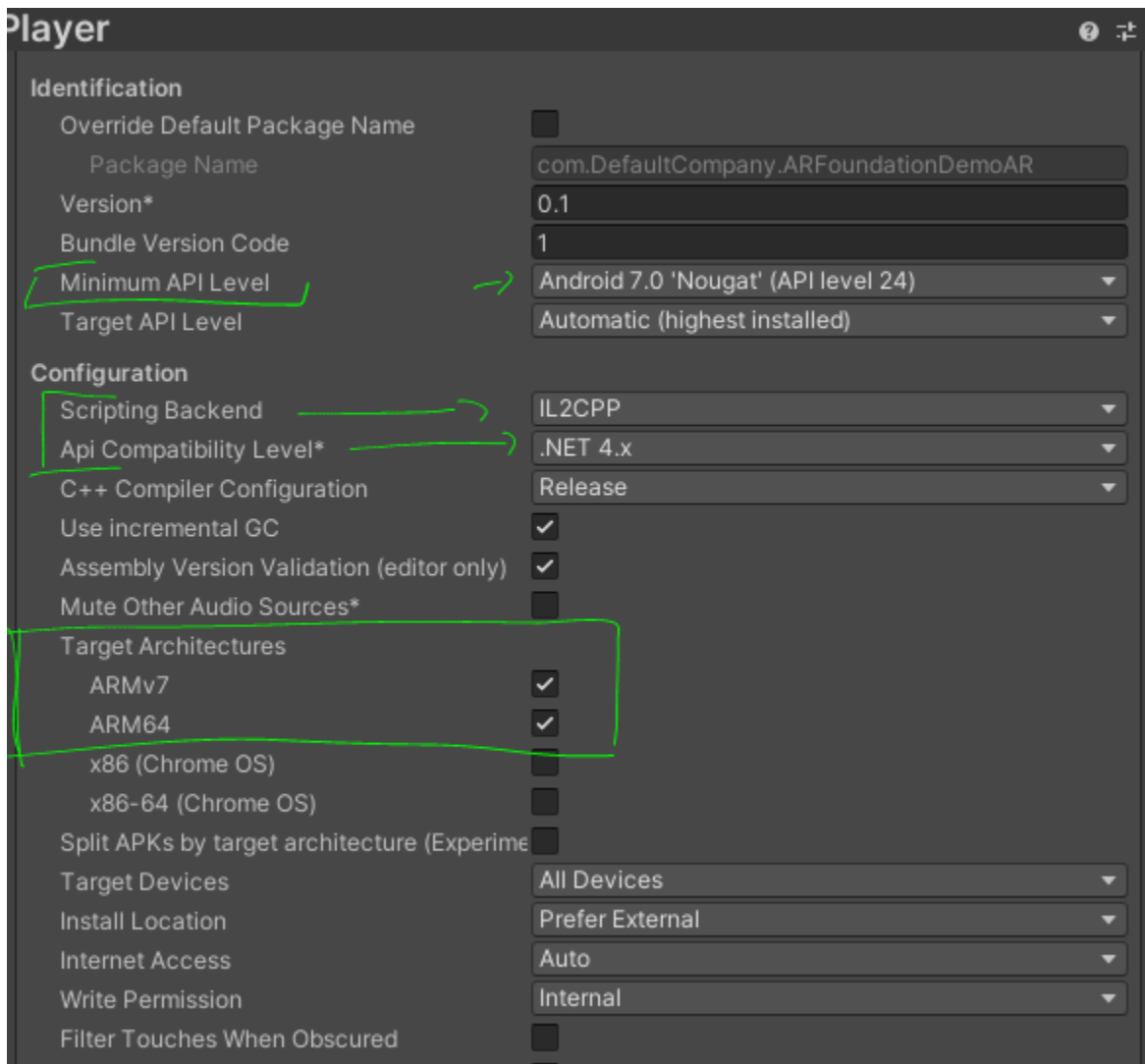


Set minimum API level to 24

Scripting Backend to IL2CPP

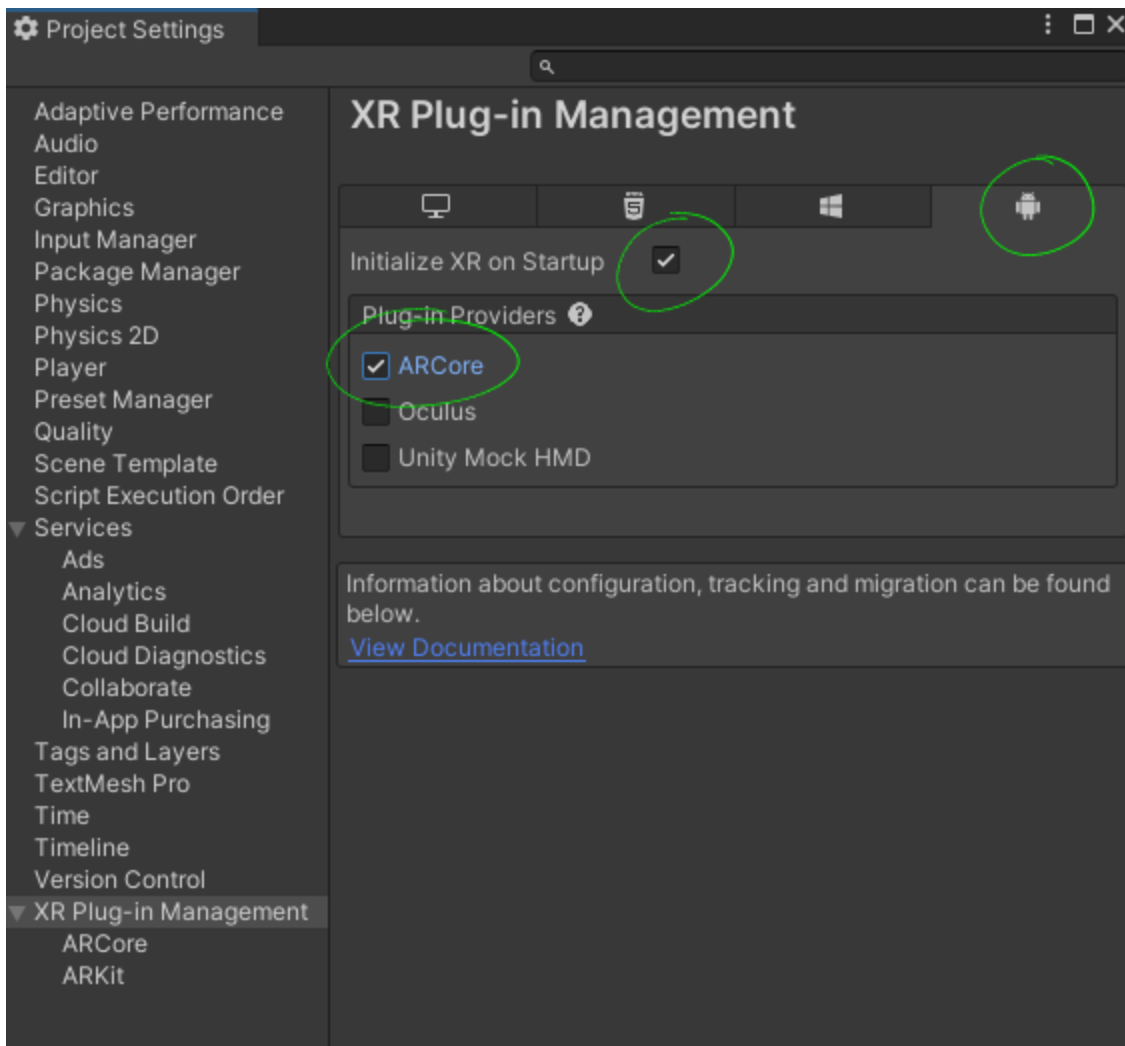
Api Compatibility Level .NET 4.x

Target Architectures ARMv7 and ARM64

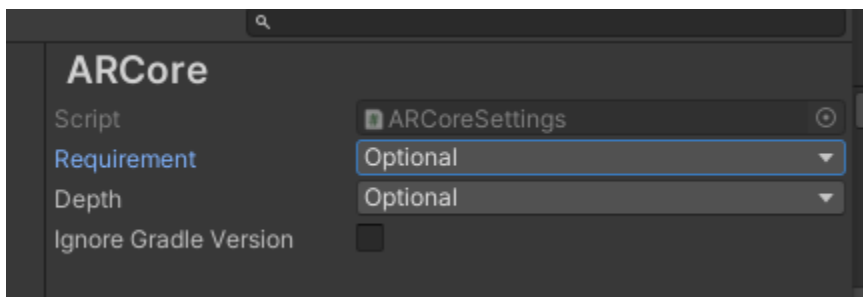


XR Plugin Management Settings:

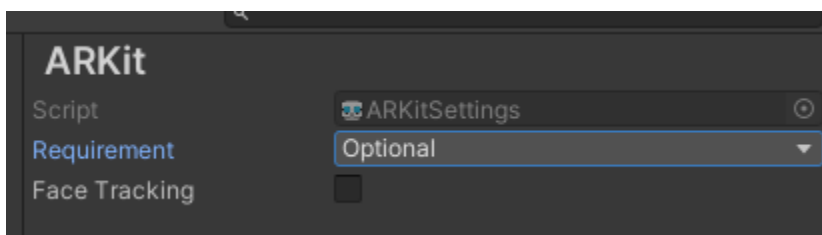
Enable ARCore



Set AR Core requirement and depth to optional



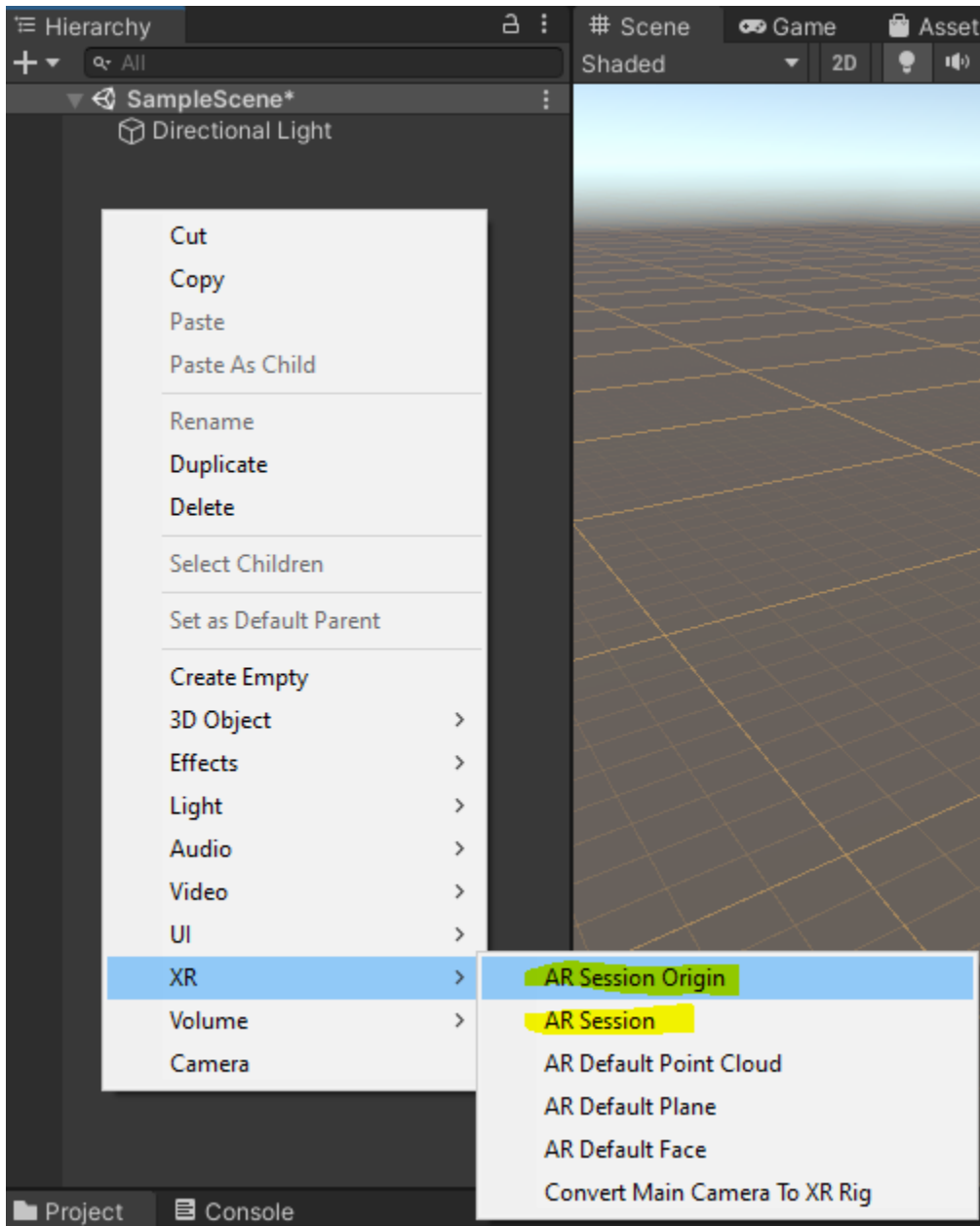
(If present) Set AR Kit to optional. (Add AR Kit package and the Unity build support for iOS to do cross platform development, with that changing the platform should be all it takes to build your Unity project for iOS.)



Scene Setup

From a basic scene setup, first delete the main Camera

Add an AR Session Origin then an AR Session object to the scene.



AR Camera Settings:

Open the AR Camera inside the AR Session object and set the tag to Main Camera

Change the background color to something low alpha and not black for troubleshooting app crashes vs. just not rendering.

In the AR Camera Manager Script, set Light Estimation to Everything.

Completed AR Starting Point

From Build Settings, select your device in the Run Device dropdown.

Build and Run.

If it works you should have the camera feed from the device displaying on the screen.

Congrats, you are now ready to start building AR Content.

Set Up Universal Render Pipeline for AR (optional):

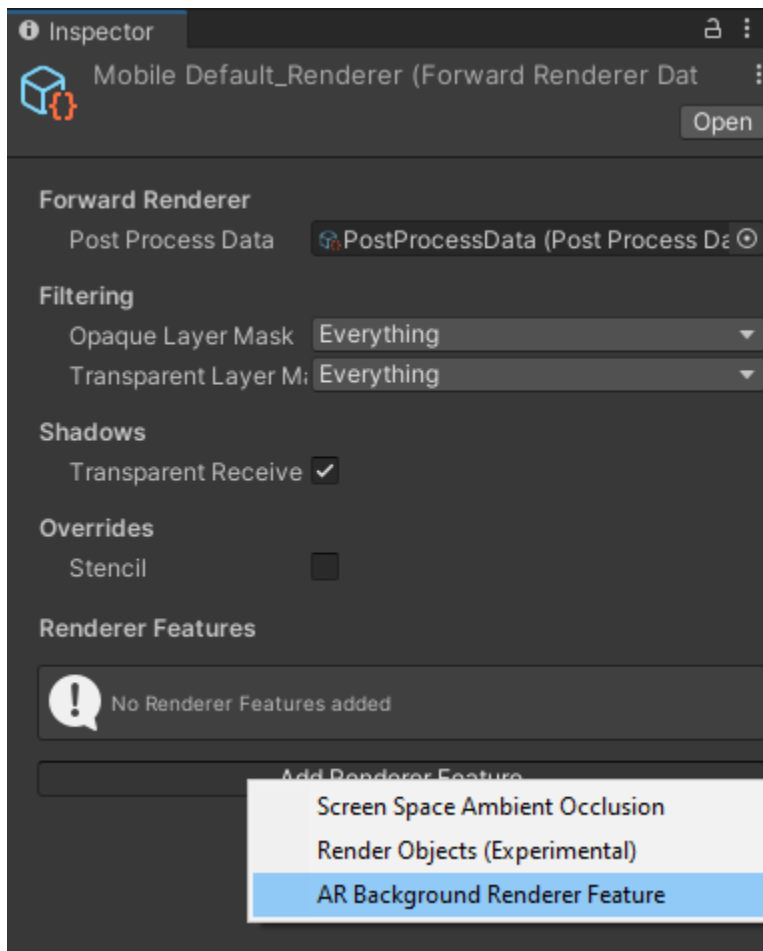
From package manager bring in Universal RP package

In project hierarchy create settings folder -> create URP settings subfolder

In URP Settings folder create > rendering > Universal Render Pipeline > Pipeline Asset (forward renderer)

Name it MobileDefault

Open the forward renderer (MobileDefault_Renderer) in the inspector, add the AR Background Render feature



Set MobileDefault render settings as desired

From Project Settings -> Graphics -> assign MobileDefault to Scriptable Render Pipeline Settings

Project Settings -> Quality -> remove everything except Medium quality, assign the MobileDefault URP.