ARCore Tutorial

2020.9.10

TA : Seo Young Oh, Woojin Cho

ARCore overview

ARCore is Google's platform for building augmented reality experiences.

ARCore enables your phone to sense its environment, understand the world and interact with information.

Some of the APIs are available across Android and iOS to enable shared AR experiences.

ARCore uses three key capabilities to integrate virtual content with the real world as seen through your phone's camera:

- **Motion tracking** allows the phone to understand and track its position relative to the world.
- **Environmental understanding** allows the phone to detect the size and location of all type of surfaces: horizontal, vertical and angled surfaces like the ground, a coffee table or walls.
- **Light estimation** allows the phone to estimate the environment's current lighting conditions.

Goal of Programming Assignments

- Learn how to utilize ARCore with Unity (for your projects)

- Explore functions of ARCore and learn how to find and use them

Programming HW Schedule

PHW	Announcement	Due	Topic
#1	9/10 THU	9/20 SUN	ARCore Setup, Plane Detection
#2	9/17 THU	9/27 SUN	Interaction
#3	9/24 THU	10/4 SUN	Augmented Images
#4	10/8 THU	10/18 SUN	Cloud Anchors

ARCore with Unity

1. **AR Foundation** with **ARCore XR Plugin**, and **ARCore Extensions** — Recommended
    AR Foundation is Unity's high-level, cross platform API to support Augmented Reality. AR Foundation lets you write your app once, and build for either Android or iOS.

    ARCore Extensions is a package that provides additional ARCore functionality that can be used with AR Foundation.

1. **ARCore SDK for Unity** — Not compatible with AR Foundation
    ARCore SDK for Unity is a standalone SDK that lets you use ARCore's features when targeting Android. It is provided as a standalone `*.unitypackage`.

ARCore with Unity

Which ARCore supported device do you have? (Check all)

21 responses
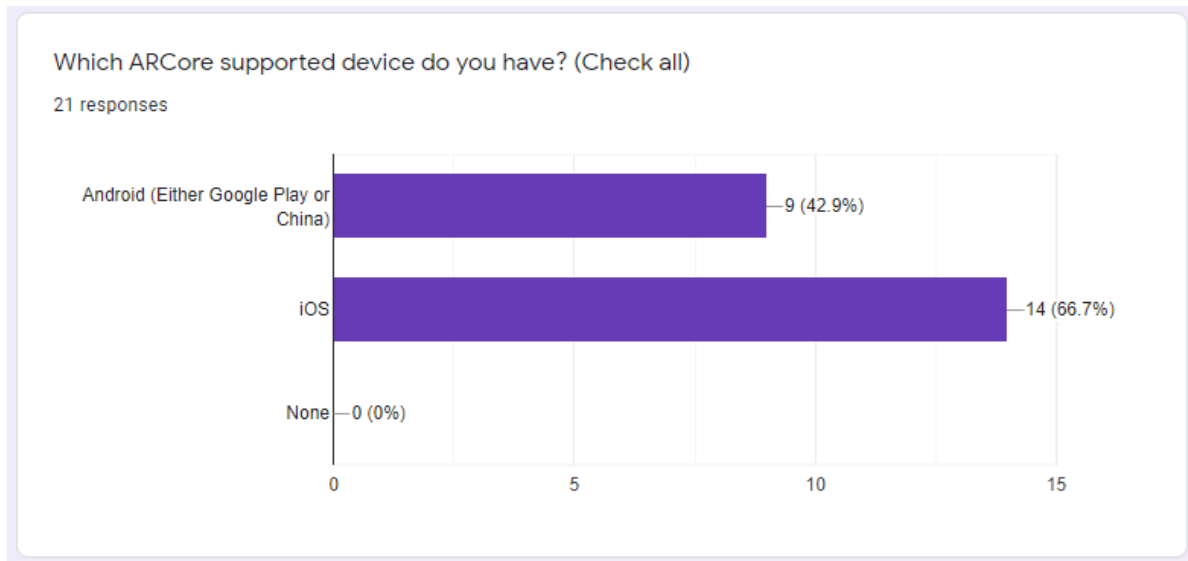
- Android development is strongly recommended if available, unless you have previous experiences on iOS development

Requirement (Android)

- Unity Hub

    - Unity 2019.4.3f1 or later (LTS versions are recommended)

- Android Studio

    - Android SDK 7.0 (API Level 24) or later (~10.0 is recommended)

- Java SE Development Kit

    - JDK 8 *(not JDK 9)* (Installation via Unity Hub is an easy way)

Requirement (iOS)

- Unity Hub
  - Unity 2019.4.3f1 or later (LTS versions are recommended)
- Apple Development
  - Xcode 9.3 or later
  - iMac or Macbook, of course (+ an Apple developer account)

Configuring Unity Project

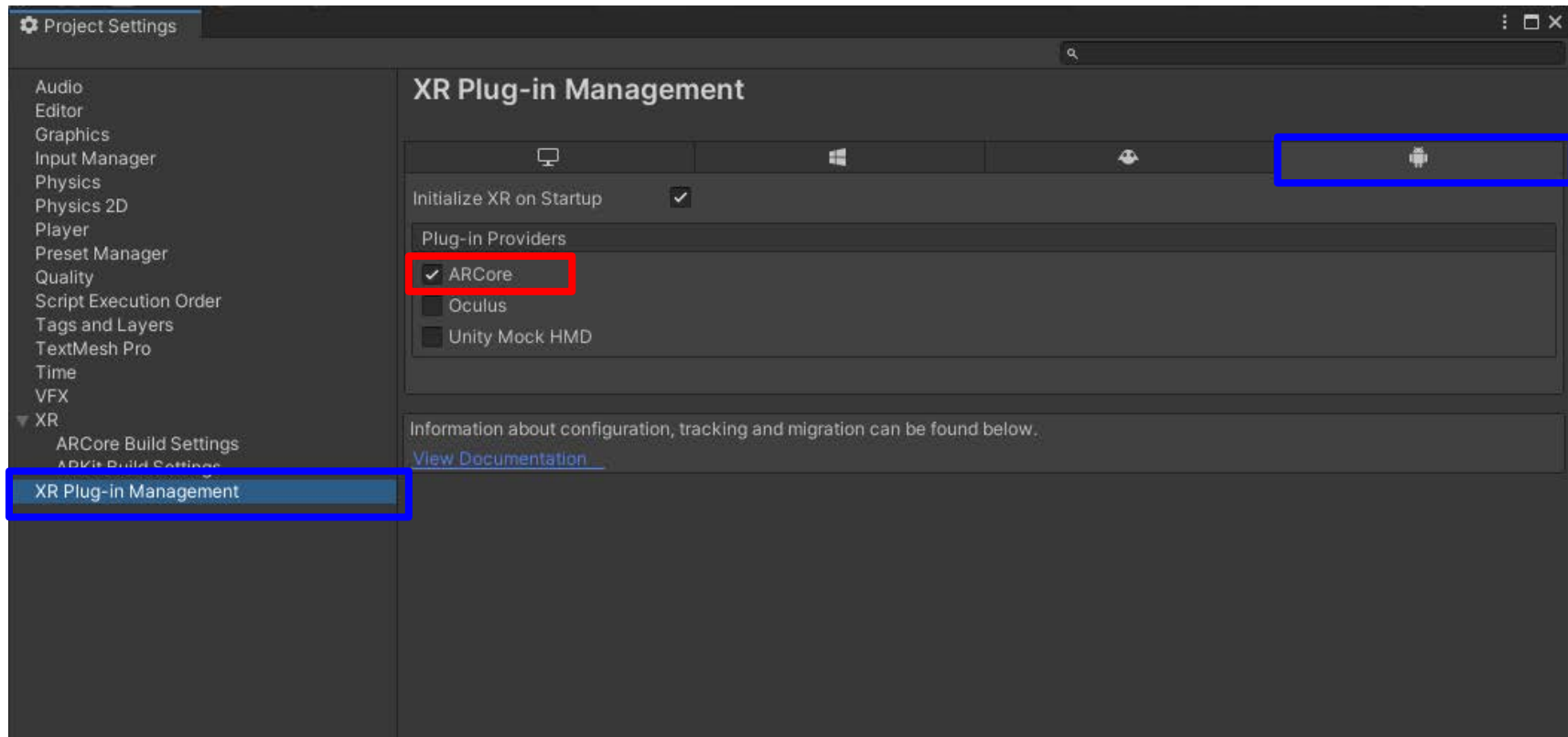- You will find the project file for PHW is already set up (mostly), but to be sure check following settings
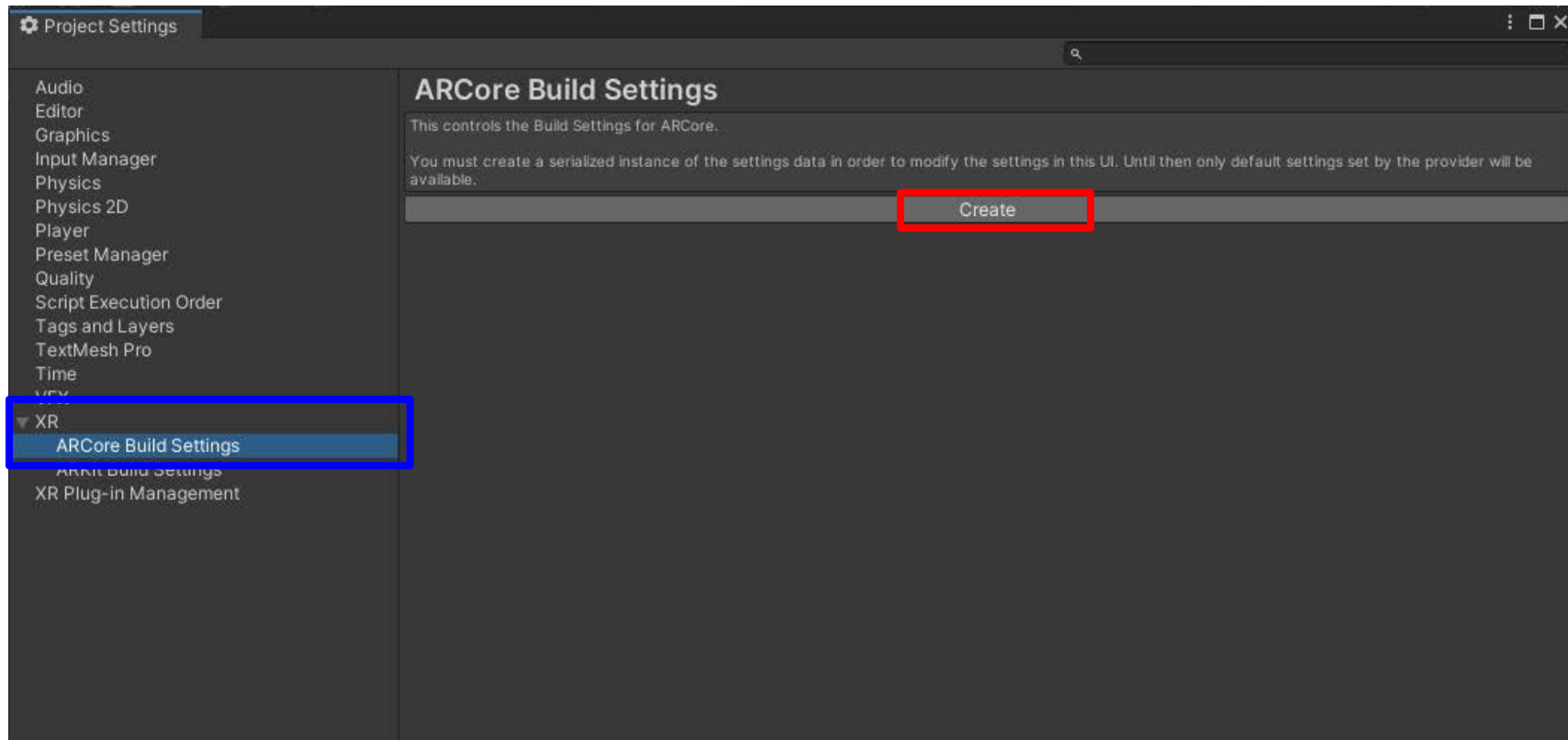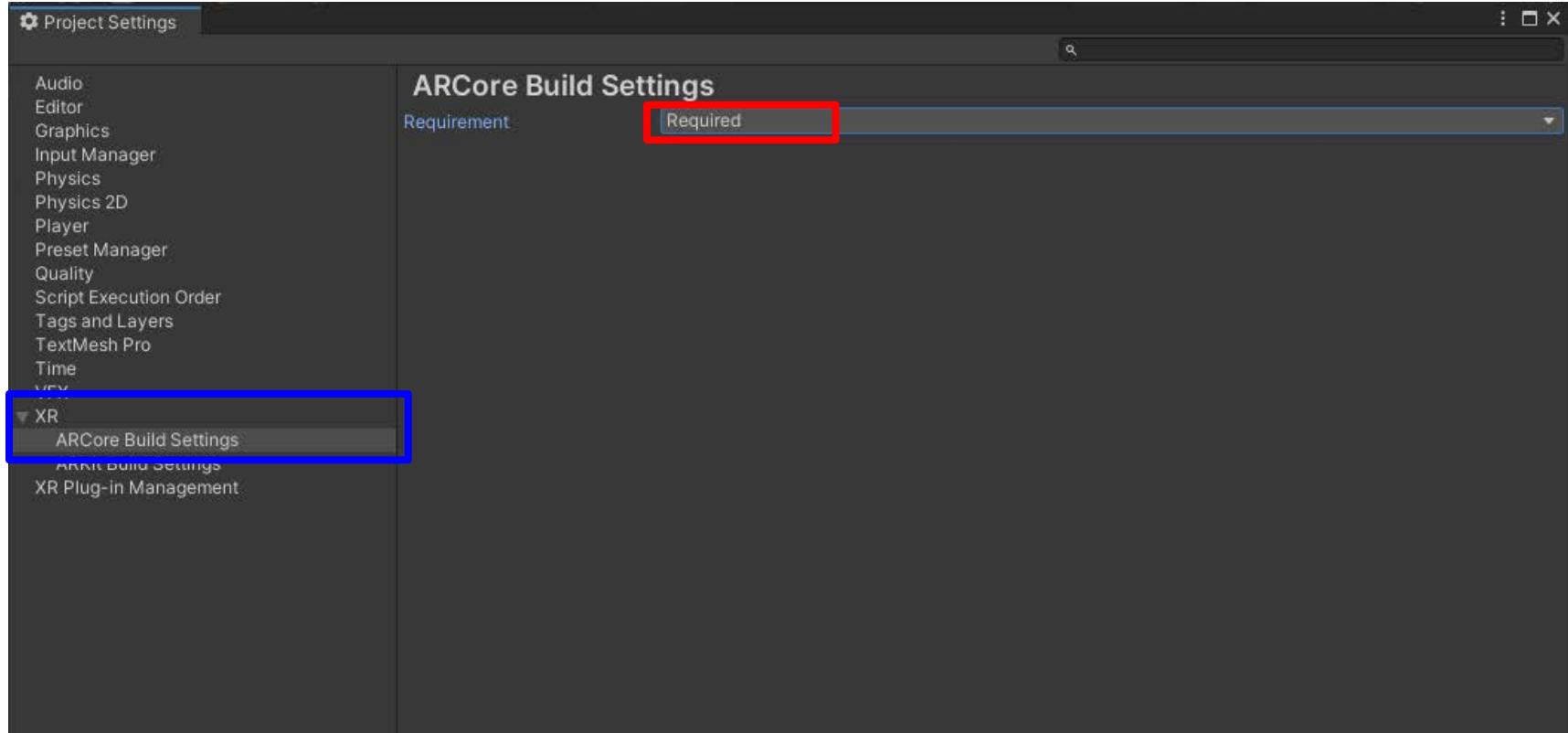
Configuring Unity Project (Android)

● Build Settings
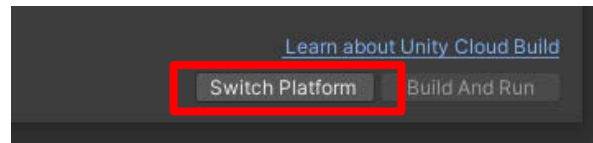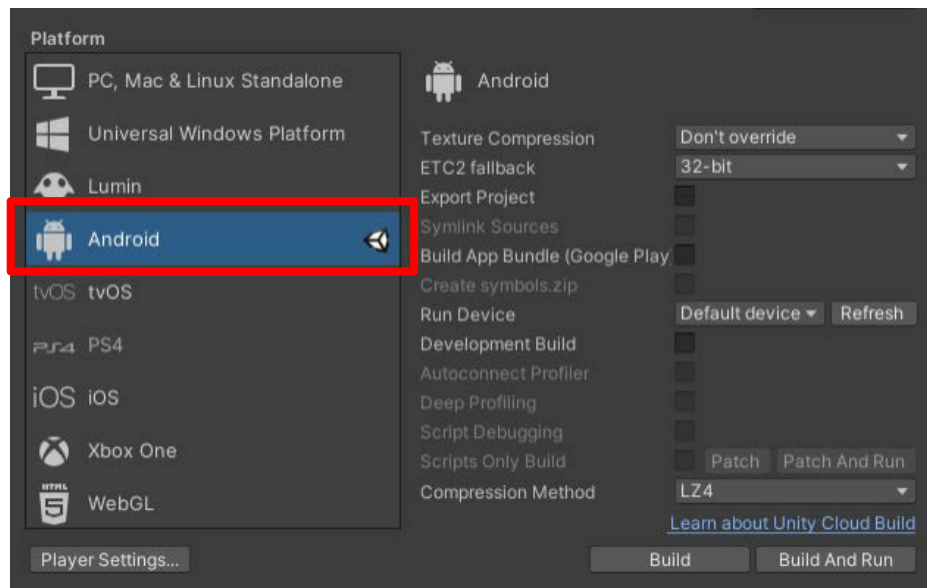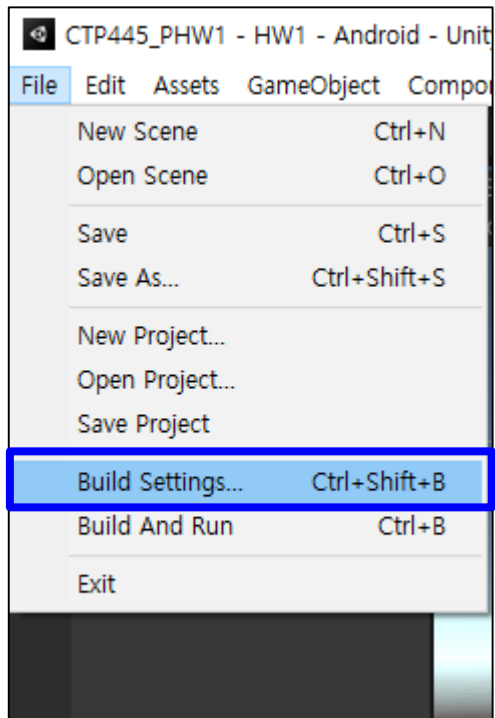
Configuring Unity Project (Android)
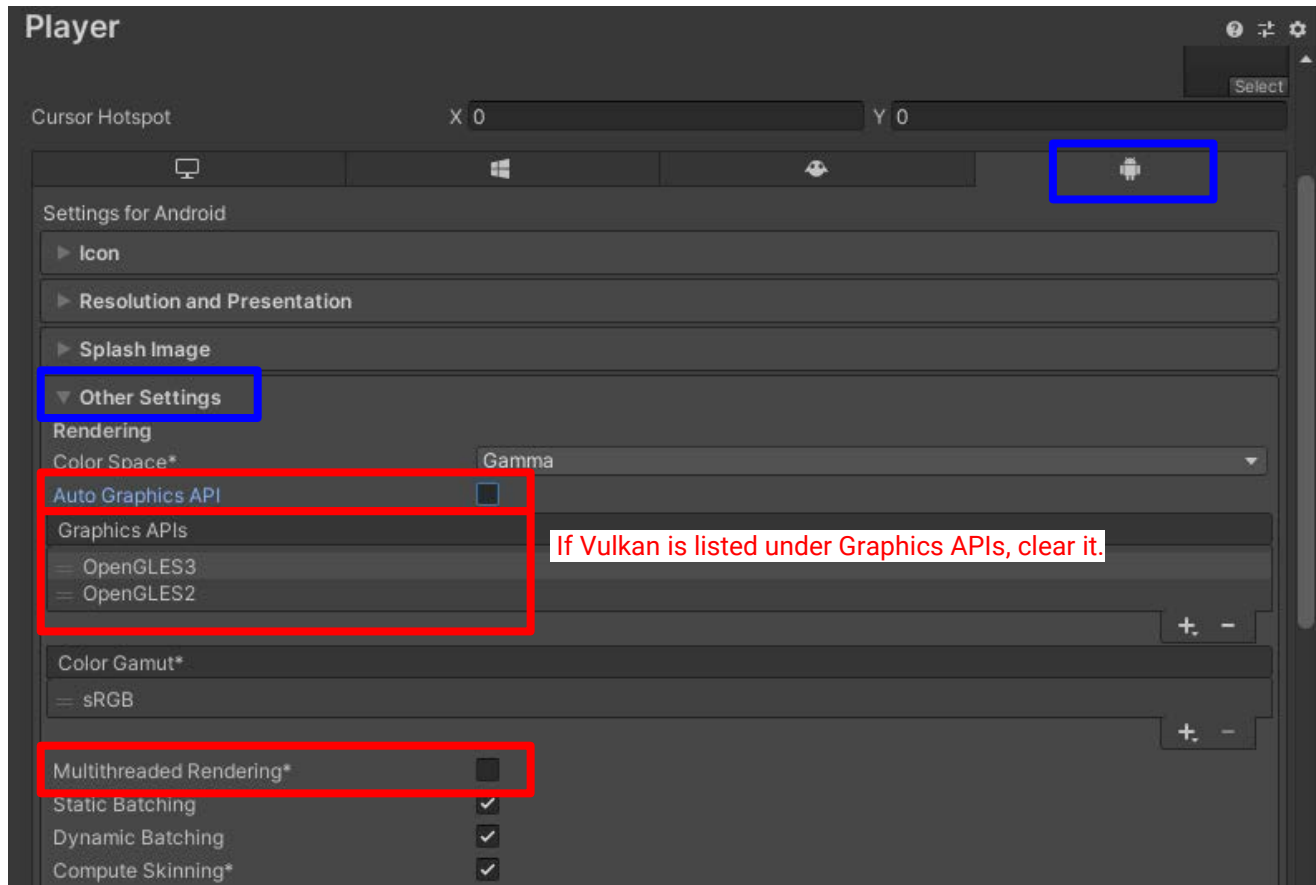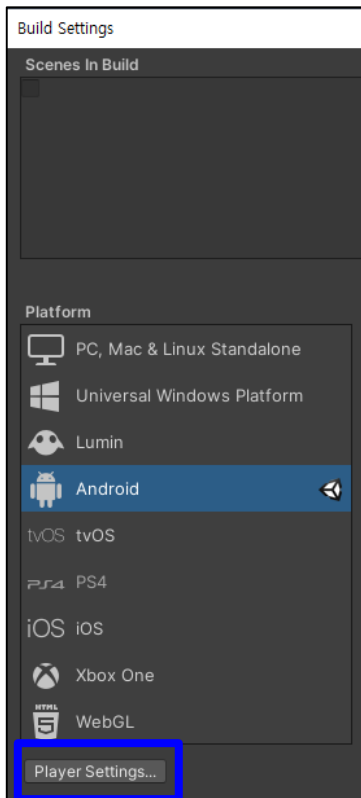
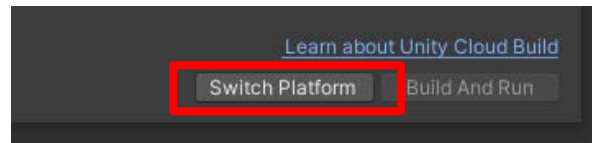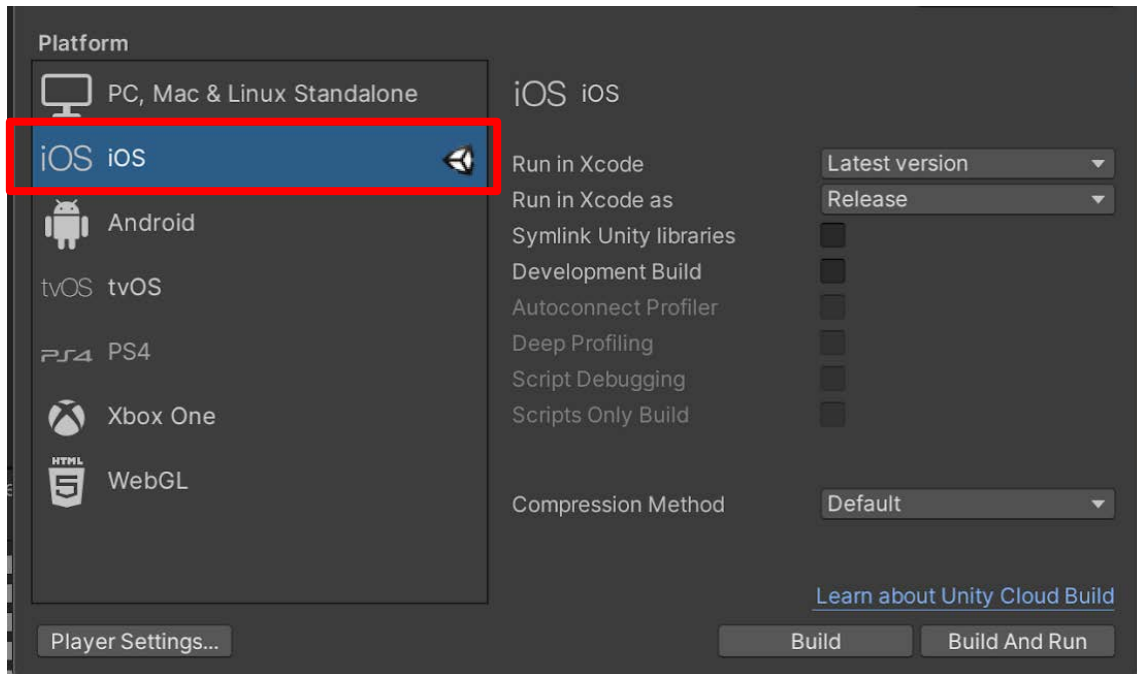- Player Settings

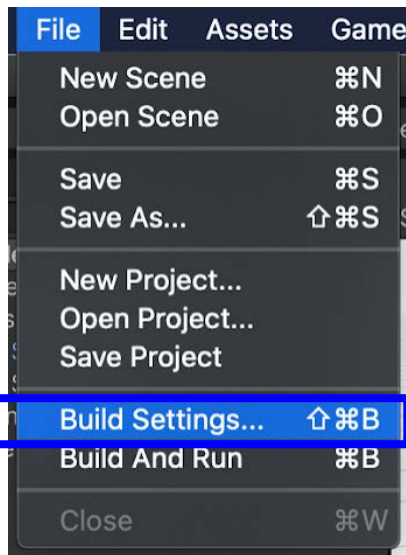# Configuring Unity Project (Android)

- Player Settings
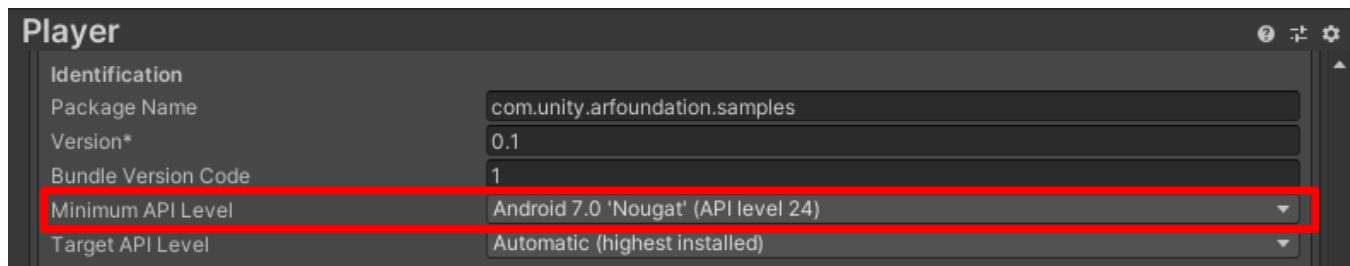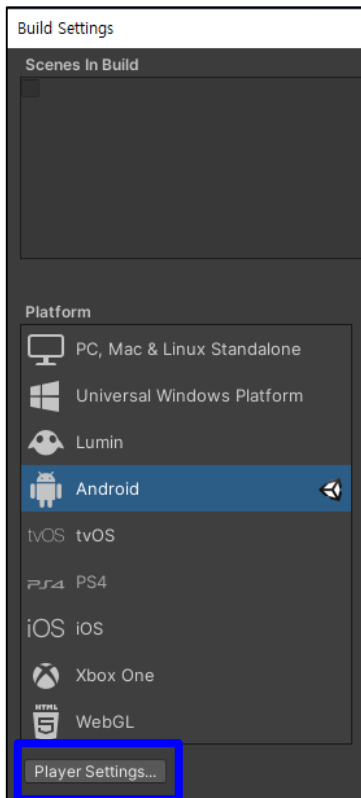
Configuring Unity Project (Android)

- Project Settings

Configuring Unity Project (Android)

- Project Settings

Configuring Unity Project (Android)
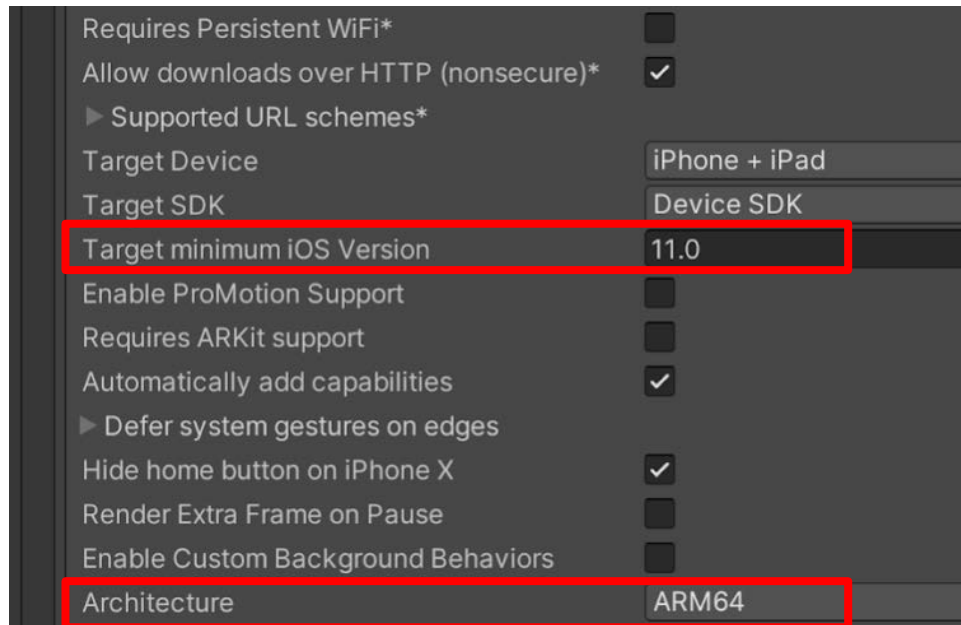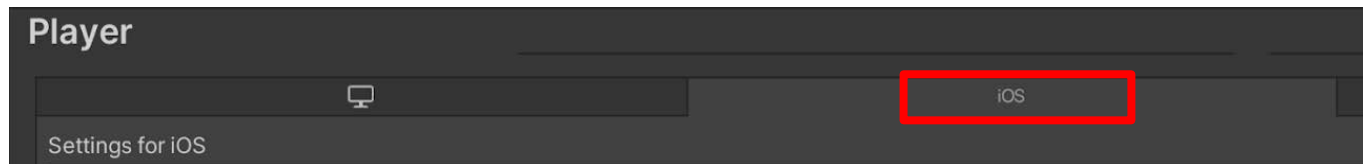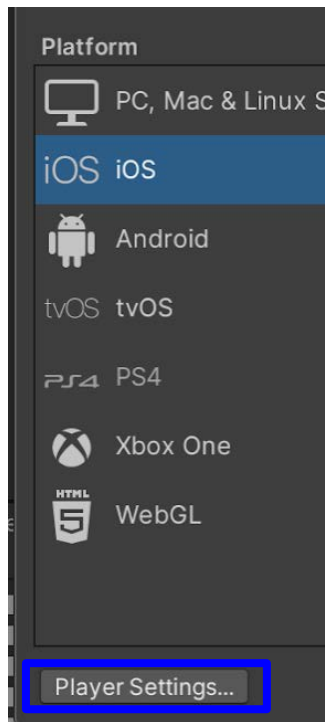
- Project Settings

Configuring Unity Project (iOS)
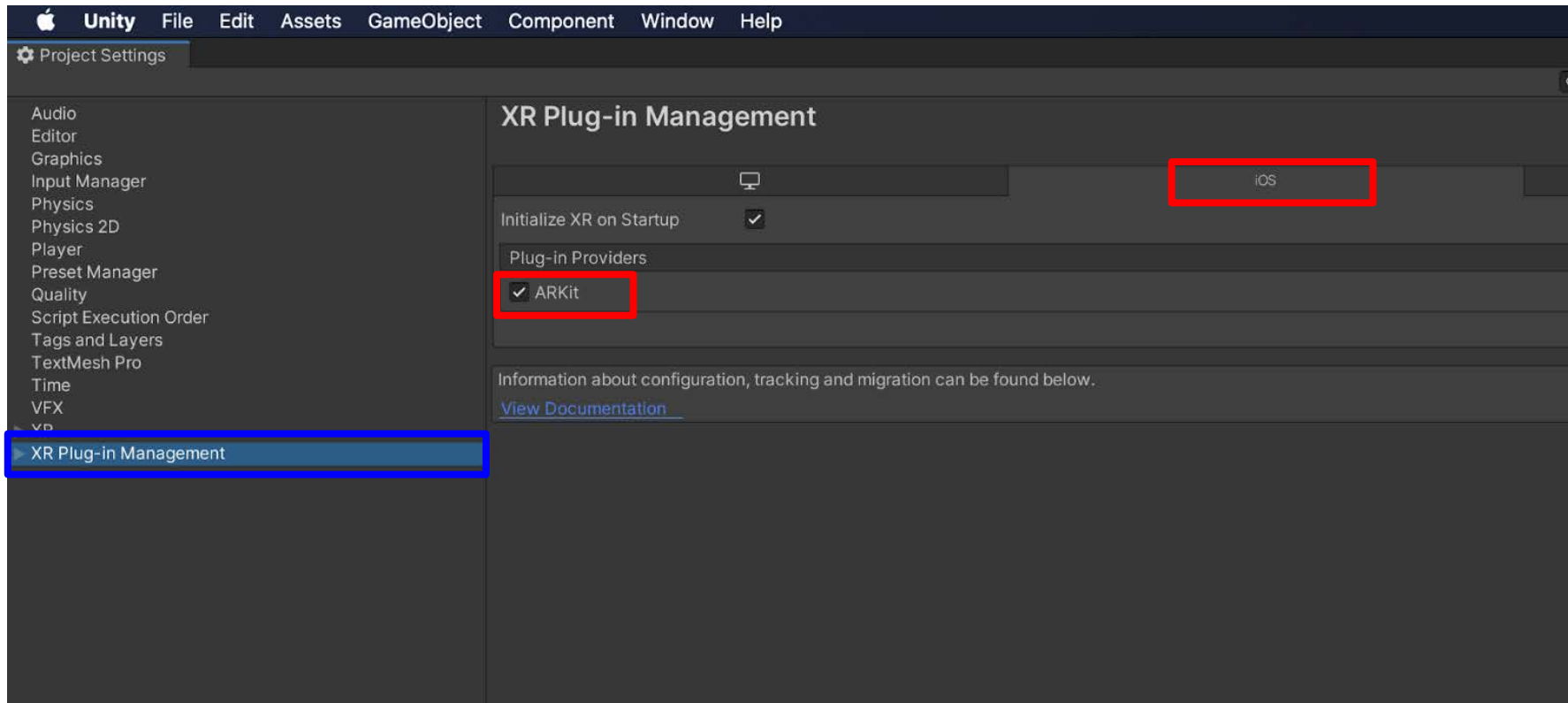
- Build Settings

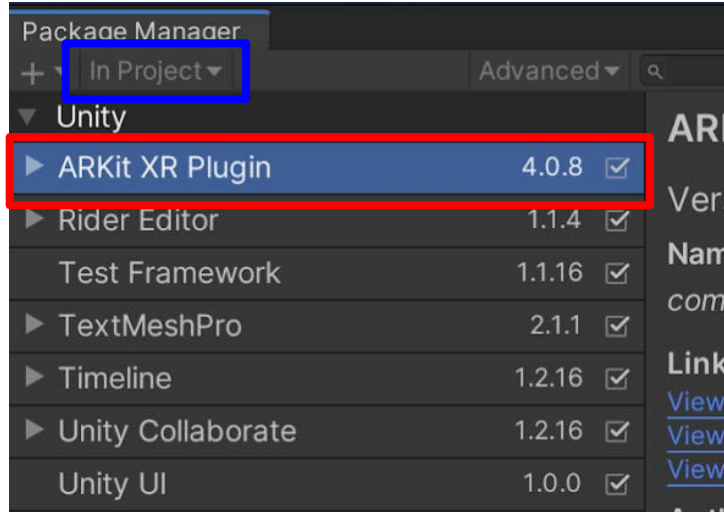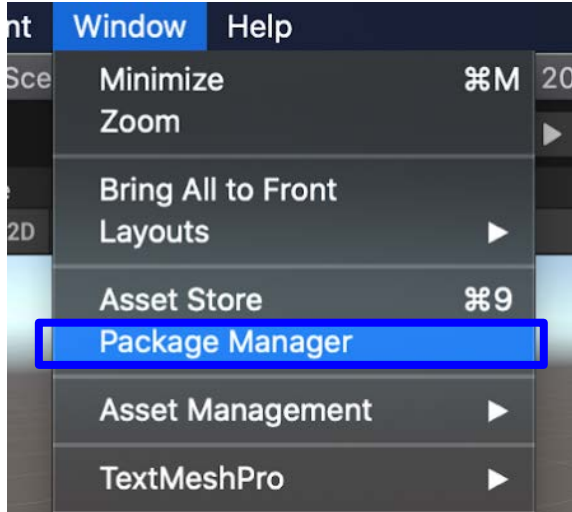# Configuring Unity Project (iOS)

- Player Settings

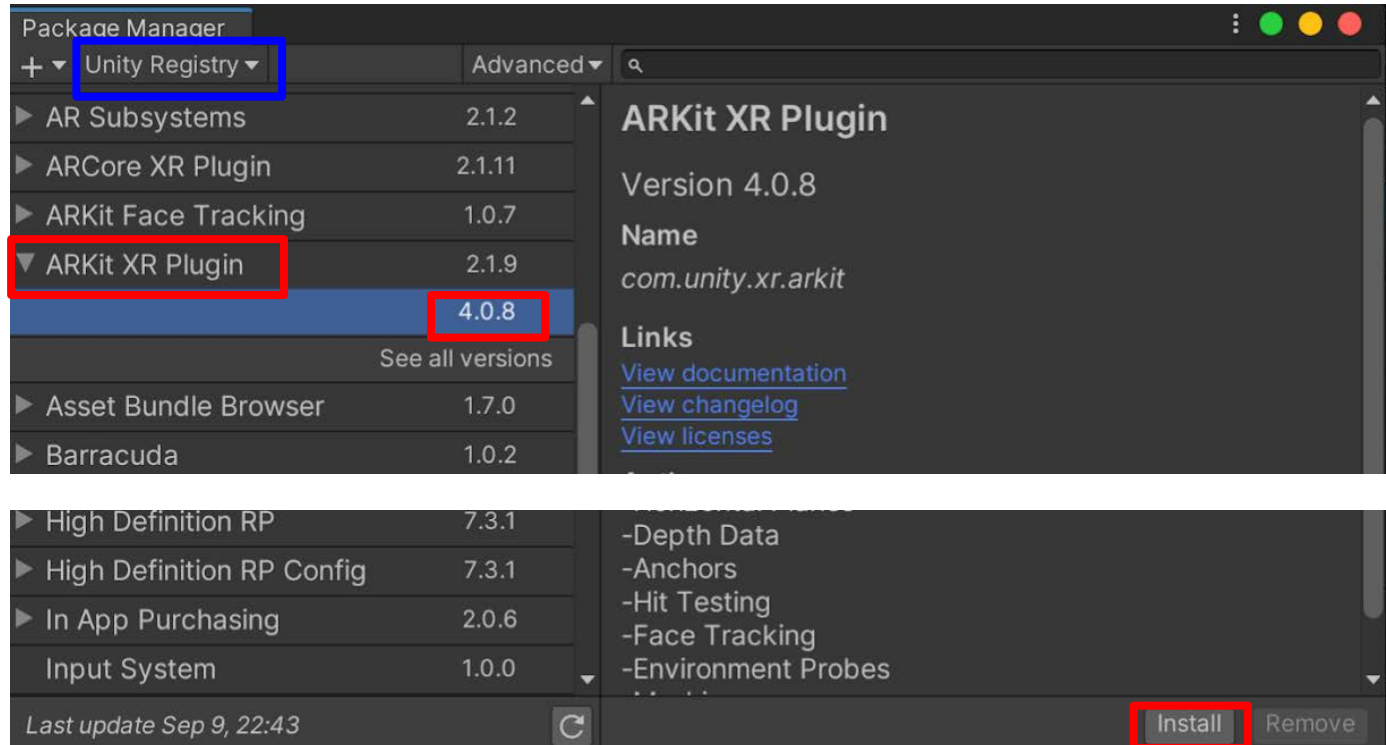# Configuring Unity Project (iOS)

- Project Settings

Configuring Unity Project (iOS)

- Package Manager

Configuring Unity Project (iOS)

- Package Manager

Install ARCore Extensions Package

- Download ARCore Extensions v1.19.0
  - https://github.com/google-ar/arcore-unity-extensions/releases/

▾ Assets 3

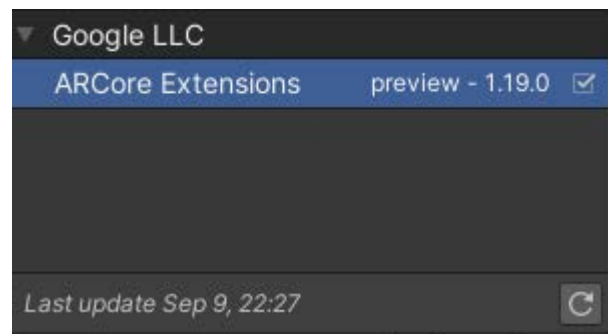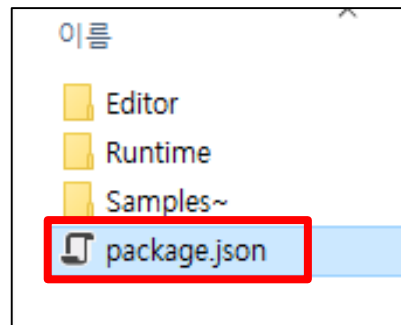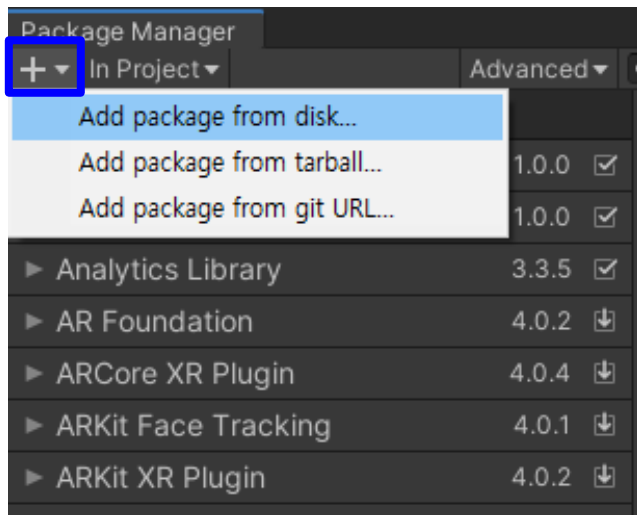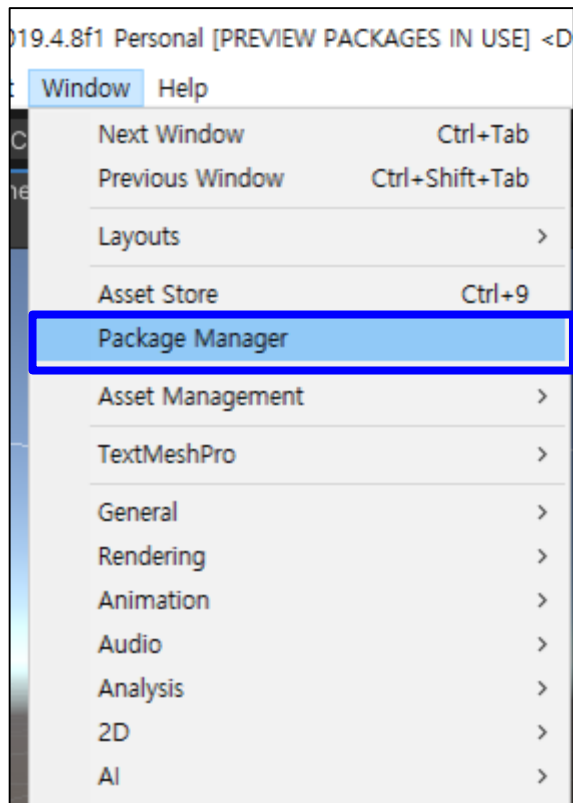|	
⬡ arcore-unity-extensions-1.19.0.tgz	1.24 MB
📄 Source code (zip)	
📄 Source code (tar.gz)	

- Extract the downloaded file

Install ARCore Extensions Package

● Open Package Manager

Programming Homework #1

- Enable ARCore Extensions on your device
- Detect planes in your environment
- Place an object on the plane

- Homework guidelines will be uploaded along with a Unity project (skeleton code)

Launching the app on devices

- Android
  - Just build on Unity and run the .apk file on the device.

- iOS
  - When you build on Unity, Xcode project will be generated.
  - .xcodeproj file doesn't contain the required iOS dependencies.
  - You should create .xcworkspace file from .xcodeproj file (ex. by using cocoapods)
  - Then run .xcworkspace file with your device connected.

Reference

ARCore

https://developers.google.com/ar/develop/unity

https://docs.unity3d.com/Packages/com.unity.xr.arcore@4.1/manual/

ARFoundation

https://docs.unity3d.com/Packages/com.unity.xr.arfoundation@4.1/manual/index.html

https://github.com/Unity-Technologies/arfoundation-samples/tree/4.0