Installing Unity and Deploying to iOS/Android

→ COMS W4172 IA Page
→ Developing with Unity

Tested Platforms

- Microsoft Windows 11 Pro running Unity LTS Release 2022.3.17f1 with Samsung Note 10 running Android 12
- MacBook Pro 2022, M2 chip, 14.2 (Sonoma) running Unity LTS Release 2022.3.17f1 with iPhone 13 running iOS version 17.2.1

Important Notes

All development in COMS W4172 will be done in Unity using one/some of a set of AR or VR libraries (to be announced soon) and deployed to a mobile device, and/or self-contained or tethered AR or VR head-worn display. Your mobile device can **run either iOS or Android**, as the tracking library will support both platforms.

Development Environment

We will be using Unity 2022.3 (**Note**: During the semester, we may announce changes to the versions of Unity or its associated libraries that we want you to use.)

- **If deploying to an iOS device**, Apple restricts deployment to be done from macOS only.
- **If deploying to an Android device,** you can deploy from either Windows or macOS. While it is technically possible to deploy to Android from Unity on Linux, we cannot provide support on this configuration (i.e., you are on your own and we will not accept Linux issues as an excuse for technical problems).

Mobile Device

You must have access to an Android or iOS device, noting again that you will *not* be able to use an iOS device unless you have a macOS computer from which to deploy to it. Please see the minimum system requirements in the following section to determine if your mobile device can be used. If you do not have a suitable mobile device, purchasing an Android (rather than iOS) device for use in the course would be the most cost-effective solution. Please let us know if you will need advice on what to purchase.

Minimum System Requirements

Mobile Device

Android	iOS
● OS: Android 6.0+ (API 23) ● Processor ARMv7 (Cortex family) or later ● GPU: Must support OpenGL ES 3.x ● Rear-facing camera ● See list of supported devices here. Make sure your device supports depth API	● OS: iOS 12+ ● XCode 12.5 or higher ● Unity requires **iOS 12 or higher,** and doesn't support earlier versions. This means that you MUST use an **iPhone 6s Plus** or better, as ARKit does not support any phone before that.

Computer

- OS:
  - Windows 10 or Windows 11, 64-bit versions only.
  - Mac OS X Sierra 10.13+ or later
- GPU: DX10, DX11, and DX12-capable GPUs.

→ Please see Manual: System requirements for Unity 2022 LTS for additional details.
→ If using ARCore, please see ARCore system requirements.
→ If using ARKit, please see ARKit system requirements.
→ If using Meta devices, please see Meta System Requirements.
→ If using Windows Mixed Reality devices, please see WMR System Requirements.

Step 1: Developer Account Registration/ Adding Apple ID to XCode (Mac/iOS)

You will need a **Unity developer account** to activate Unity and to run it with online services and resources. Additionally, if deploying from macOS to an iOS device,

- Unity ID Setup
- Additionally, if deploying from macOS to an iOS device
    - You will need a free Apple ID (**an Apple Developer account is not necessary**). Input your Apple ID to XCode's build configuration by navigating to *XCode → Preferences → Account*, and selecting your Apple ID. See iOS account setup

Step 2: Install Unity

- Download Unity Hub
- Follow the instructions in the install wizard.
- Open Unity Hub, and to install Unity 2022.3, start by selecting: ***Installs→Install Editor→Unity 2022.3* (recommended release)**
    - **If on OS X**: Depending on your chipset, install the 'Silicon' or 'Intel' version. More information, see here.

> In the "Choose Components" step, in addition to the components checked by default ("Unity..." and "Microsoft Visual Studio...") and select one or both of **"Android Build Support" and "iOS Build Support,"** depending on which one(s) you plan on using! XR packages will be available for download from within Unity.

- Once the install completes, open Unity Hub.
- Enter your Unity ID account credentials (click **Create Account** and see Step 1 if you do not have a Unity account). Click **OK** after entering your credentials.
- You will be prompted to select the version of Unity you wish to activate. Please check the **Unity Personal Edition** box to activate the free version of Unity. Then click **OK**.
    - Click "Start Using Unity" to proceed.
- The application should open on the "Projects" dialog window. Click **New**.
    - Select a location and create a name for your project.
    - Choose "3D".
    - Click **Create project**
    - If the application crashes, please consult Unity's troubleshooting page or contact an IA.

Step 3: Install iOS/Android development tools

If deploying to an iOS device:

- Upgrade your Operating System and iTunes installation to the most recent versions.
  - Download and install Xcode.
  - You can download the latest Xcode directly from the App Store for free.
  - Deploying to an iOS device from **Unity 2022.3** requires **Xcode 14.1** or later.
  - Please see Unity's Getting started with iOS development for more details.

If deploying to an Android device:

- Install the correct drivers for your Android device.
  - **If using Windows**, please visit OEM USB Drivers.
  - **If on OS X**, you can **skip** this step.
- Enable USB debugging on your Android device.
  - You will need to access **Settings→Developer Options**, which is hidden by default. To show Developer options, go to **Settings→About Phone→Software Information** and tap **Build Number** seven times (yes, that sounds ridiculous). Then, go back up to **Settings** to find that **Developer Options** is now visible.
  - Enable **USB debugging** in **Developer Options**. (On Samsung devices, this dialogue will indicate that its connection type is "MTP.")
  - Please see Using Hardware Devices if you want more information.
- Connect your Android device to your computer through USB.
  - When your device is connected by USB, its connection type, as specified in **Developer Options**, should be either PTP or MTP, if it does not already have one of these values, you will need to change it. (This step will probably not be necessary! On Samsung devices, this information is provided when you enable USB debugging, as described above, and will be "MTP.")
  - You will need to confirm through a dialogue box on your device that USB debugging is allowed the first time you connect your device to a specific computer.
- When building your first Android project in Unity, you may be asked to provide the path to the Android SDK if Unity can't find it on its own.
  - If this happens, make sure you have installed "Android SDK & NDK Tools" and "OpenJDK" from Unity Hub (under the "Android Build Support" section) and have the path set up correctly in the editor settings (under edit=>preferences=>external tools)
- Please see Unity's Android SDK Setup for additional details.

Step 4: Test Unity + Android/iOS

If deploying to an iOS device:

1. Attach your iOS device via USB.
2. Open Unity
3. Create a new project.
4. Select **File→Build Settings**.
5. Select iOS from the list of available platforms.
6. Select **Player Settings.**
    a. Note: When you actually deploy, don't forget to click "add open scenes" on the upper right to actually add your scenes into the build.
    b. When you click "add open scenes", it will ask you to name the new scene. Just name it whatever you'd like.
    c. Go down to Identification and change the Bundle Identifier. It must fit the format, com.Company.ProductName, so you can use something like com.yourname.game
7. Select **Build and Run**
8. When the build process completes, XCode will close any open Unity XCode project and will start the XCode deployment process.
    a. **You may find an error message from XCode saying "Signing for Unity-iPhone requires a development team".**If this occurs, make sure your Apple ID is selected in your XCode preferences (See step 1).
        i. In XCode, click the little folder icon on the upper left corner. Change the "Bundle Identifier" to match the one you specified in the Unity Player Settings.
        ii. Change Team to match the personal team associated with your Apple ID
        iii. It should create a certificate for you if you don't have one, and everything should work after that.
        iv. Also, make sure you've chosen your iOS device as the device to use (rather than "Generic iOS Device").
9. When the XCode deployment process completes, your iOS device should launch the Unity application.

If deploying to an Android device:

1. Attach your Android device via USB.
2. Open Unity and **create a new project.**
3. Select **File→Build Settings**.
4. Select Android from the list of available platforms.
5. Select **Build and Run**
    a. Unity may ask for the path to your Android installation. Select **Browse** and locate the root folder of your Android SDK directory.
6. When the build process completes, your Android device should launch the Unity application.

Troubleshooting

→ **Unity** troubleshooting
→ **Android** troubleshooting