Developing with Unity

→ COMS W4172 IA Page
→ Installing Unity and Deploying to iOS/Android

Learning Unity 1
  Unity Documentation 1
    Tutorials 1
  Unity Tips 2
    Reparenting a GameObject 2
    Quality Settings 2
    Physics Settings 2
    Physic Materials 3
    Managing Project size 3
    Networking 3

XR Development 4
  Important Notes 4
  Option 1: VR in Unity 4
    Desktop VR 4
    Mobile VR 5
  Option 2: AR in Unity 5

Learning Unity

Unity Documentation

The Unity Manual helps you understand Unity's features, structure while providing figures and examples. For scripting you will use Unity's Class Reference and Scripting Guide.

Tutorials

Follow the Roll-a-ball tutorial to quickly get started with Unity. Here you will find more tutorial videos by project and topic.

Suggested tutorial order:

1. Interface Tutorials: Complete parts 1-3
2. Introduction to Scripting in Unity
   a. For those familiar with C#: Start from the Awake/Start tutorials and complete to end
3. Creating Ragdolls (3D Physics Tutorial): 1-4

Suggested Documentation:
- Learn the Unity Interface and Creating Scenes
- Understand Unity Assets (Primitive objects, Importing Assets, Notes on supported formats)
- Transformations in Unity
- Unity Scripting
- Input devices in Unity

Unity Tips

Reparenting a GameObject

If you want to change the parent of a GameObject in a script, you can assign the transform component of the new parent to the Transform.parent member variable of the child GameObject. (See Transform.parent for more details.)

Quality Settings

Unity allows you to set the graphical quality of its rendering pipeline. Quality settings are customizable and can be enabled/disabled for any platform Unity supports. The Quality Settings Inspector panel can be opened by going to Edit→Project Settings→Quality. At the top of the Quality Settings Inspector panel you will find a matrix of checkboxes. The rows of the matrix represent the quality setting and the columns of the matrix represent the platform. Please see the Unity reference page on Quality Settings for a detailed breakdown of each setting.

To change the default quality setting, select the small black triangle in the default row under the platform you wish to deploy to, and in the drop down list, select the quality setting you want to use as default. Below the settings matrix, you will find options that define the settings for the currently selected row in the matrix.

Physics Settings

Unity has a robust physics engine, but you may need to adjust the default settings in order to tune performance to your particular needs. To open the Physics Manager Inspector Panel, go to Edit→Project Settings→Physics. Please see the Unity reference page on the Physics manager for a detailed breakdown of each setting.
Important Physics Settings:

Gravity:
- You may need to change the Gravity setting when using Tracking Targets in Vuforia.
- Determine which axis will represent “down” in your system and change the magnitude and axis in the Gravity setting here.

Bounce:
- The Bounce threshold will determine at what relative velocity bouncing will stop.
  - Setting the value too high may stop bouncing top quickly.
  - Setting the value too low may result in endless bouncing.
  - Determine what settings are appropriate for your system through experimentation.

Default Contact Offset:
- This sets how many meters two colliding GameObjects may overlap before the physics engine pushes them apart.
- As with the Bounce threshold, determine what is right for your system through experimentation.

Physic Materials

Unity allows you modify the effect of friction and “bounciness” on colliding GameObjects through Physic Materials (yup, there’s no final “s” in “Physic”). To add Physic Materials to your project, go to Assets→Create→Physic Material. You can change the settings of a Physic Material by selecting the material in Project View and editing the properties in the Inspector. See the Unity Reference page on Physic Material for additional details.

To add a Physic Material to a GameObject:
- Drag the Physic Material onto the GameObject name in the Hierarchy View,
- Drag the Physic Material onto the GameObject in the Scene View, or
- Drag the Physic Material into (Inspector Panel) GameObject Inspector→Collider Component→Material.
  - If your GameObject does not have a Collider component, you will need to add one.

Managing Project size

The Unity Project may become very large as packages and assets are added:
- Consider removing unneeded packages.
- However, ensure that asset chains and required Prefabs are not removed.
  - Play it safe: Backup your project before attempting to clean it up!

Networking

Unity has several options for networking. We recommend using PUN2 Free. Please refer to their documentation for instructions on using networking.
If you are looking for a **client-server architecture solution** or other advanced features, we recommend using Mirror. Please refer to their documentation for more details. For comparisons between Unity networking solutions, please refer to this official post.

**XR Development**

You have 2 options for XR development in 4172: AR or VR (please see the appropriate sub-sections below). If you have time, try both! Lastly, if you would like to borrow one of the lab’s VR devices, let us know!

**Important Notes**

XR development libraries used by Unity went through significant changes over the last few years. If you are using online resources to learn VR and AR development, (tutorials, videos, guides...) double check that the Unity version and VR/AR libraries you are using in your project match the one in the tutorial. Known changes:

- Tutorials might point to the Unity library “OpenVR” in Unity. OpenVR is deprecated and replaced by OpenXR.

**Option 1: VR in Unity**

Feel free to deploy to either *Meta Devices*, SteamVR supported devices such as *HTC Vive or Valve Index*, *Windows Mixed Reality Devices* or to *Mobile VR* when the time comes. We will be making Meta Quest 2 and 3 devices available later this semester.

Unity's VR tutorials are a great introduction to VR development. We recommend **Create with VR** - Beginner’s guide (for Meta Quest, Valve Index, HTC Vive).

If you are using one of the lab’s Meta Quests, please check out the following pages:

- [Oculus Quest Best Practices](#)
- [Meta Quest Getting Started](#)
Desktop VR

OpenXR, SteamVR (3rd party tutorial here), Meta, and Microsoft have their own tutorials/samples. Please try to follow one of them.

Note: in 2018 SteamVR drastically changed how it handles input. Please see the tutorial by Eric Van de Kerckhove on RayWenderlich.com here to learn how to use SteamVR 2.0 input.

Mobile VR

VR is also supported on mobile devices. Please see one of the following tutorials for mobile VR support.

- Unity + Meta + Mobile tutorial
- Unity + Google Cardboard SDK tutorial

Please come speak to us to discuss optimization techniques for VR and AR applications on mobile devices. You can deploy to a mobile device from your laptop.

Option 2: AR in Unity

There are a number of ways of creating AR applications using Unity. The most straightforward, ARFoundation, which is part of Unity, relies on platform-native AR libraries: ARCore and ARKit (Android and Apple, respectively, although ARCore will also run on Apple devices). While it is possible to use ARCore and ARKit directly, we will not be doing that in this course.