

McGillXR Workshop 2

Overview

Workshop Hosts: Annabel Wing-Yan Fan, Eden Redman

Summary: This workshop focuses on how to set up your own XR Unity Project. We will create a basketball arcade game where you can interact with various objects in the scene using a virtual head-mounted device (HMD) or headset (Oculus Quest 2).

Date: 22/03/2023

Pre-Workshop

You should have the following items prepared:

- Install Unity Hub
- Install Unity (version 2022.2.11f or any current version)
- Workshop Assets: https://github.com/McGillXR/Winter-2023-Workshop-Series/tree/main/Day_2

Set up an XR Compatible Project

1. Create a New Project based on the VR Core Template
2. Changing the of the windows (optional):
 - a. On the top right-hand corner, change the Layout to <2 by 3>
 - b. On the project tab, select the drop-down menu, and the <One Column Layout>
3. Go to Edit > Project Settings > XR Plug-in Management:
 - a. Check Unity Mock HMD
4. Go to Window > Package Manager:
 - a. Change <Packages: In Project> drop-down to <Packages: Unity Registry>
 - b. Search for and install: XR Interaction Toolkit
 - c. Choose "Go Ahead..." when prompted
 - d. From the Samples tab options, import: Starter Assets and XR Device Simulator
5. Under Project > Assets > Samples > Starter Assets:
 - a. Under the Inspector > Add all XRI Default Inputs to <ActionBasedContinuousMoveProvider default>

Import Workshop Asset Package

Go to where you downloaded workshop 2's asset package from McGillXR's GitHub:

1. Double-click on the asset package (has a unity icon next to it)
2. In the pop-up within Unity click "Import All"
 - a. Take a moment to review the imported assets
3. Under BasketBallArcade_Assets > Scenes:
 - a. Double-click StartScene
 - b. Take a moment to review the GameObjects in StartScene
4. Under Hierarchy > uncheck/deactivate XRRig
5. Right-click in Hierarchy panel > XR > add XR Origin (VR)
6. Right-click in Hierarchy panel > XR > add Locomotion System (Action-based)
7. With Locomotion System selected > Under Inspector:
 - a. Uncheck Teleportation Provider and Snap Turn Provider
 - b. Add component Continuous Move Provider (Action-based)
 - i. Uncheck Right Hand Move Action > Use Reference
 - c. Add component Continuous Turn Provider (Action-based)
 - i. Uncheck Left Hand Turn Action > Use Reference
8. Under Project > Assets > Samples > XR Interaction Toolkit > 2.2.0 > XR Device Simulator:
 - a. Drag and drop the XR Device Simulator Prefab into the Hierarchy panel
9. Under XR Origin > Camera Offset > LeftHandController:
 - a. Add a Cube
 - b. Under Inspector > Transform > change scale to (x = 0.1, y = 0.1, z = 0.15)
 - c. Duplicate Cube (ctrl+D) and add to RightHandController

Okay, time to hit the Play button for the first time!

MOCK HMD CONTROLS

Very unintuitive controls :)

- Right-click with mouse to move the camera
- Scroll wheel with mouse to move camera forwards and backwards
- Push down on mouse-wheel to aim

While holding shift:

- WASD to move
- Q, E to rotate
- G to grab
- left-click as trigger button

While holding space:

- controls the right controller

While holding left-shift:

- controls the left controller

Create Grab Interactable Basketball

Now we will create a basketball Prefab GameObject to interact with:

1. Right-click in Hierarchy panel > 3D Object > Cube:
 - a. Re-name to SpawnPoint
 - b. Under Inspector > Transform > change position to (x = 0, y = -1.14, z = 0.48), change scale to (x = 0.36, y = 0.36, z = 0.36)
 - c. Under Inspector > uncheck Mesh Renderer and Box Collider

- d. Under Project > BasketballArcade_Assets > Scripts:
 - i. Drag and drop the SpawnBasketball script under SpawnPoint in the Hierarchy panel
2. Under Project > BasketballArcade_Assets > 3DModels:
 - a. Drag and drop the basketball model under SpawnPoint in the Hierarchy panel
 - b. Under Inspector > Transform:
 - i. Select <Reset> from the drop-down menu
 - ii. Change the scale to (x = 90, y = 90, z = 90)
 - c. Under the Inspector, click on the <Add Component> button:
 - i. Add a Rigidbody
 - ii. Add a sphere collider
 - iii. Add XR Grab Interactable, change the Throw Velocity Scale to 7
 - iv. Add the DestroyOnCollision Script
3. Drag and drop the basketball under the Project > BasketballArcade_Assets > Prefabs folder:
 - a. Say yes to creating a basketball Variant
4. Drag and drop this new basketball Variant prefab into the Basketball field of SpawnPoint's Spawn Basketball Script
5. Under the Scene or Hierarchy tab, click on the basketball:
 - a. Duplicate the basketball two more times, and move the basketballs around so they are not overlapping (make sure they are children of the SpawnPoint)

Okay, time to hit the Play button!

SHOOTING THE BASKETBALL

- Hold left shift to control the left controller
- Press down on the mouse wheel to aim the controller (or use the mouse to move the controller) until you see the red ray beam turn white
- Press and Hold G to grab the basketball
- Holding shift, use the WASD keys to move towards the hoop
- Release G to throw the basketball

This is finicky, give it some practice!

To make things easier to see you can also:

- Deactivate the RightHandController under Hierarchy > XR Origin > Camera Offset
- Select the <Left Eye> under the Game view during play mode