Unity with optional VR controls

- Initializing non-VR player character
- Switching from VR to non-VR mode in the Editor
- Switching from VR to non-VR mode at runtime

Initializing non-VR player character

In this tutorial, we will assume that your Unity project has already been set up for VR use with the XR Interaction Toolkit, as well as the Input System. To initialize a non-VR player character in an existing VR project in Unity, follow these steps:

1. Create a new GameObject: In the Unity Editor, right-click in the Hierarchy panel and select "Create Empty." This will create a new empty GameObject. Name this object "Player".

2. Attach the VR Mode Switcher script to the "Player" GameoObject. This script will allow you to enable/disable the VR Mode at runtime.

3. Create a child GameObject: right-click on the Player GameObject in the Hierarchy panel and select "Create Empty". Name this object "Player Controller".

4. Attach a Rigidbody: Select the "Player Controller" GameObject, then click on the "Add Component" button in the Inspector panel. Search for "Rigidbody" and add it to the GameObject. Set the "Interpolate" property to "None". The Rigidbody controller will handle physics for your player object.

   > **Be careful with Mesh Colliders!**
   > GameObjects that have a Rigidbody component only support Mesh Colliders that have Convex option enabled: the physics engine can only simulate convex mesh colliders.

5. Attach a character controller: Select the "Player Controller" GameObject, then click on the "Add Component" button in the Inspector panel. Search for "Character Controller" and add it to the GameObject. The Character Controller component will handle the player's movement and collision.

6. Add a 3D model or sprite: If you want a visual representation for your player character, you can add a 3D model or sprite. Import your desired model or sprite into Unity by dragging it into the Project panel. Then, drag and drop the model or sprite onto your Player GameObject in the Scene view or Hierarchy panel.

7. Set up player controls: Attach a Player Input object and the Input System Test to the Player Controller. Open the "Events" drop-down menu, and add the corresponding

function from Input System Test to each Event included in the Input Action Asset. Some example functions for things such as camera movement, player movement, and jumping are already included in the Input System Test script.

8. Add a raycast for UI interaction: To enable the player character to interact with UI elements such as buttons or menus, you'll need to add a raycast. Attach the Raycaster script to the player character GameObject. This script uses the Physics.Raycast method to detect if the ray hits any UI elements and perform the desired actions accordingly, such as button clicks or menu selections.

9. Test and refine: Save your script and go back to the Unity Editor. Disable the Xr Origin object to make sure that the game won't start in VR mode. Press the Play button to enter Play mode and test your player character initialization and movement. Make any necessary adjustments to the code or settings until you are satisfied with the result.

That's it! By following these steps, you can initialize a player character in Unity and have them ready for movement and interaction in your game.

Switching from VR to non-VR mode in the Editor

Here's how to change from a VR to a non-VR player view in the Unity editor:

1. Open your Unity project and make sure you have both the VR and non-VR player controllers set up. These controllers should be separate GameObjects, each with scripts and components attached to the player character.
2. In the Unity editor, locate the XR Origin. This GameObject is responsible for handling the input and movement in the VR environment.
3. Disable the XR Origin. You can do this by right-clicking on the script/component in the inspector and selecting "Disable" or "Remove". The XR Interaction Manager GameObject can remain active
4. Locate the non-VR player controller component. This component is responsible for handling the input and movement in the non-VR environment.
5. Enable the non-VR player controller component. If the component was already added but disabled, right-click on it in the inspector and select "Enable". If the component wasn't added yet, refer back to the previous tutorial on how to set up the non-VR player controller.
6. Save your changes by pressing Ctrl + S (Windows) or Command + S (Mac) or by navigating to File > Save Scene.
7. Enter Play mode by clicking the Play button at the top of the Unity editor or by pressing Ctrl + P (Windows) or Command + P (Mac).
8. You should now be able to see the non-VR player view in the Game view window instead of the VR view. You can use your regular input controls to interact with the game as a non-VR player.

That's it! You have successfully changed from a VR to a non-VR player view in the Unity editor. You can toggle between the two views by enabling/disabling the corresponding player controller scripts or components.

Switching from VR to non-VR mode at runtime

Enabling and disabling VR through keyboard input

1. Make sure you have both the XR Origin and non-VR player controllers set up in your Unity project. These controllers should be separate scripts or components attached to the player character.
2. Attach the "VRModeSwitcher" script to an empty GameObject in your scene.
3. In the Unity editor, drag and drop the XR Origin GameObject and the non-VR player controller GameObject into the appropriate fields of the "VRModeSwitcher" script component.
4. Save the script and return to the Unity editor.
5. Enter Play mode by clicking the Play button at the top of the Unity editor or by pressing Ctrl + P (Windows) or Command + P (Mac).
6. Press the (alphanumeric) 1 key to toggle between VR and non-VR modes at runtime. The corresponding player controller should be enabled or disabled based on the selected mode.

Enabling and disabling VR through an in-game button