using System;

using UnityEngine;

using UnityEngine.EventSystems;

namespace UnityStandardAssets.CrossPlatformInput

{

public class Joystick : MonoBehaviour, IPointerDownHandler, IPointerUpHandler, IDragHandler

{

public enum AxisOption

{

// Options for which axes to use

Both, // Use both

OnlyHorizontal, // Only horizontal

OnlyVertical // Only vertical

}

public int MovementRange = 100;

public AxisOption axesToUse = AxisOption.Both; // The options for the axes that the still will use

public string horizontalAxisName = "Horizontal"; // The name given to the horizontal axis for the cross platform input

public string verticalAxisName = "Vertical"; // The name given to the vertical axis for the cross platform input

Vector3 m\_StartPos;

bool m\_UseX; // Toggle for using the x axis

bool m\_UseY; // Toggle for using the Y axis

CrossPlatformInputManager.VirtualAxis m\_HorizontalVirtualAxis; // Reference to the joystick in the cross platform input

CrossPlatformInputManager.VirtualAxis m\_VerticalVirtualAxis; // Reference to the joystick in the cross platform input

void OnEnable()

{

m\_StartPos = transform.position;

CreateVirtualAxes();

}

void UpdateVirtualAxes(Vector3 value)

{

var delta = m\_StartPos - value;

delta.y = -delta.y;

delta /= MovementRange;

if (m\_UseX)

{

m\_HorizontalVirtualAxis.Update(-delta.x);

}

if (m\_UseY)

{

m\_VerticalVirtualAxis.Update(delta.y);

}

}

void CreateVirtualAxes()

{

// set axes to use

m\_UseX = (axesToUse == AxisOption.Both || axesToUse == AxisOption.OnlyHorizontal);

m\_UseY = (axesToUse == AxisOption.Both || axesToUse == AxisOption.OnlyVertical);

// create new axes based on axes to use

if (m\_UseX)

{

m\_HorizontalVirtualAxis = new CrossPlatformInputManager.VirtualAxis(horizontalAxisName);

CrossPlatformInputManager.RegisterVirtualAxis(m\_HorizontalVirtualAxis);

}

if (m\_UseY)

{

m\_VerticalVirtualAxis = new CrossPlatformInputManager.VirtualAxis(verticalAxisName);

CrossPlatformInputManager.RegisterVirtualAxis(m\_VerticalVirtualAxis);

}

}

public void OnDrag(PointerEventData data)

{

Vector3 newPos = Vector3.zero;

if (m\_UseX)

{

int delta = (int)(data.position.x - m\_StartPos.x);

delta = Mathf.Clamp(delta, - MovementRange, MovementRange);

newPos.x = delta;

}

if (m\_UseY)

{

int delta = (int)(data.position.y - m\_StartPos.y);

delta = Mathf.Clamp(delta, -MovementRange, MovementRange);

newPos.y = delta;

}

transform.position = new Vector3(m\_StartPos.x + newPos.x, m\_StartPos.y + newPos.y, m\_StartPos.z + newPos.z);

UpdateVirtualAxes(transform.position);

}

public void OnPointerUp(PointerEventData data)

{

transform.position = m\_StartPos;

UpdateVirtualAxes(m\_StartPos);

}

public void OnPointerDown(PointerEventData data) { }

void OnDisable()

{

// remove the joysticks from the cross platform input

if (m\_UseX)

{

m\_HorizontalVirtualAxis.Remove();

}

if (m\_UseY)

{

m\_VerticalVirtualAxis.Remove();

}

}

}

}