using System;

using UnityEngine;

namespace UnityStandardAssets.CrossPlatformInput.PlatformSpecific

{

public class MobileInput : VirtualInput

{

private void AddButton(string name)

{

// we have not registered this button yet so add it, happens in the constructor

CrossPlatformInputManager.RegisterVirtualButton(new CrossPlatformInputManager.VirtualButton(name));

}

private void AddAxes(string name)

{

// we have not registered this button yet so add it, happens in the constructor

CrossPlatformInputManager.RegisterVirtualAxis(new CrossPlatformInputManager.VirtualAxis(name));

}

public override float GetAxis(string name, bool raw)

{

if (!m\_VirtualAxes.ContainsKey(name))

{

AddAxes(name);

}

return m\_VirtualAxes[name].GetValue;

}

public override void SetButtonDown(string name)

{

if (!m\_VirtualButtons.ContainsKey(name))

{

AddButton(name);

}

m\_VirtualButtons[name].Pressed();

}

public override void SetButtonUp(string name)

{

if (!m\_VirtualButtons.ContainsKey(name))

{

AddButton(name);

}

m\_VirtualButtons[name].Released();

}

public override void SetAxisPositive(string name)

{

if (!m\_VirtualAxes.ContainsKey(name))

{

AddAxes(name);

}

m\_VirtualAxes[name].Update(1f);

}

public override void SetAxisNegative(string name)

{

if (!m\_VirtualAxes.ContainsKey(name))

{

AddAxes(name);

}

m\_VirtualAxes[name].Update(-1f);

}

public override void SetAxisZero(string name)

{

if (!m\_VirtualAxes.ContainsKey(name))

{

AddAxes(name);

}

m\_VirtualAxes[name].Update(0f);

}

public override void SetAxis(string name, float value)

{

if (!m\_VirtualAxes.ContainsKey(name))

{

AddAxes(name);

}

m\_VirtualAxes[name].Update(value);

}

public override bool GetButtonDown(string name)

{

if (m\_VirtualButtons.ContainsKey(name))

{

return m\_VirtualButtons[name].GetButtonDown;

}

AddButton(name);

return m\_VirtualButtons[name].GetButtonDown;

}

public override bool GetButtonUp(string name)

{

if (m\_VirtualButtons.ContainsKey(name))

{

return m\_VirtualButtons[name].GetButtonUp;

}

AddButton(name);

return m\_VirtualButtons[name].GetButtonUp;

}

public override bool GetButton(string name)

{

if (m\_VirtualButtons.ContainsKey(name))

{

return m\_VirtualButtons[name].GetButton;

}

AddButton(name);

return m\_VirtualButtons[name].GetButton;

}

public override Vector3 MousePosition()

{

return virtualMousePosition;

}

}

}