[Nível 1: 1](#_Toc1940946061)

[Nível 2: 2](#_Toc1417560612)

[Nível 3: 3](#_Toc826600872)

[Nível 4: 4](#_Toc1852791604)

[Nível 5: 5](#_Toc1251321901)

**DPJ\_02**

# Nível 1:

In Level 1, we installed the Input System through the Package Manager. Then, we created an Action Map dedicated to movement, defining the directional keys (W, A, S, D) that we would use to control the character (Fig.1).

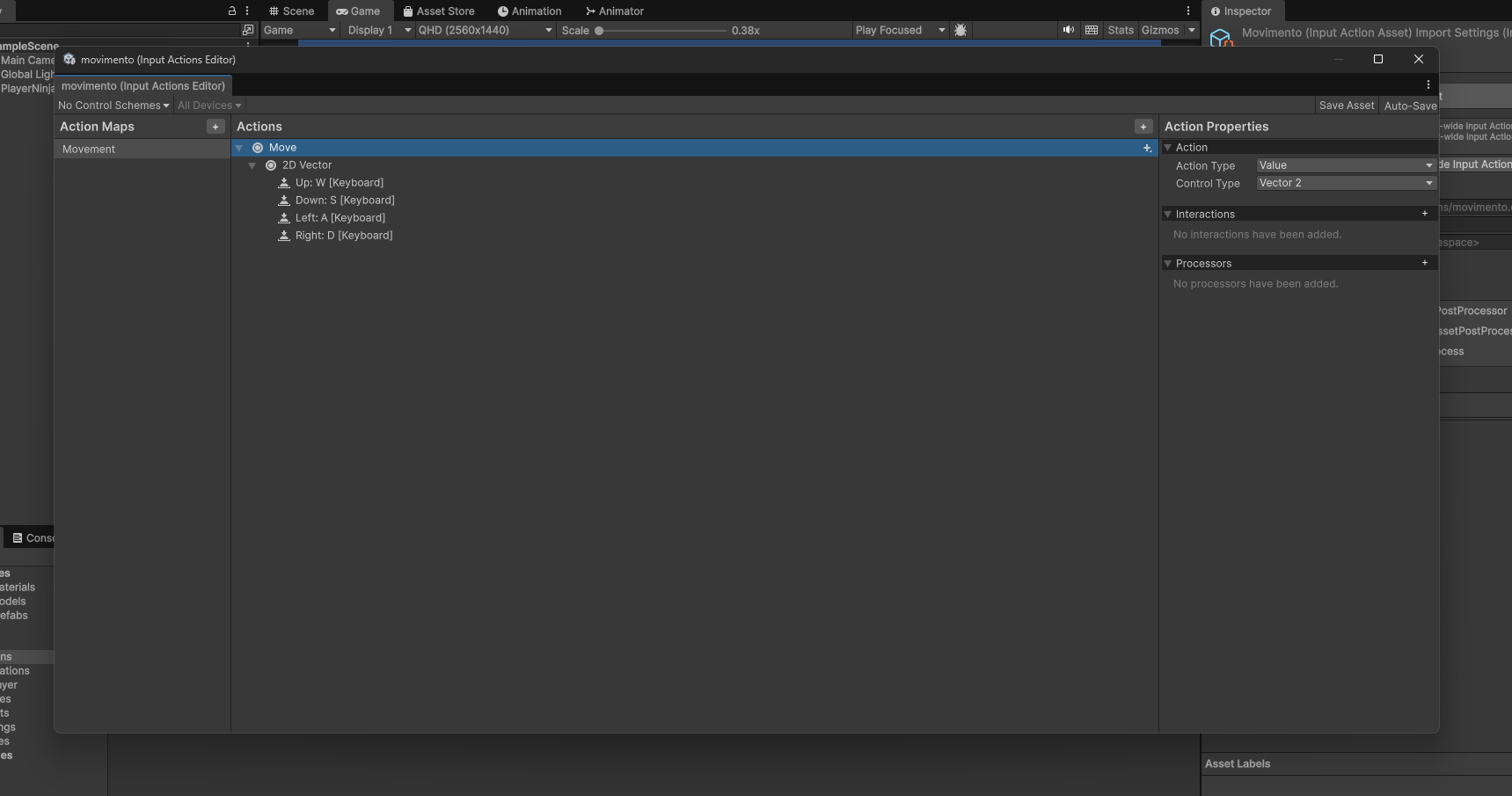


Fig.1

# Nível 2:

In Level 2, in the Green Ninja character sprites, we adjusted properties such as Pixels per Unit so that the character has the correct size within the scene (Fig.2). We used the Sprite Editor to slice the spritesheets into individual frames in order to create the character’s movement animations as they walk through the scene.



Fig.2

# Nível 3:

In Level 3, we renamed the main sprite to PlayerNinja and created a Sorting Layer to control the rendering of the characters. We organized folders within our project and created the PlayerMovement script, linking it to PlayerNinja (Fig.3).

 Fig.3

# Nível 4:

In Level 4, within the script, we define variables such as the character’s speed and the Input Actions system (Fig. 4), writing methods to enable and disable the commands. This allows us to read keyboard input and set up the character’s movement based on a key press (Fig. 5).

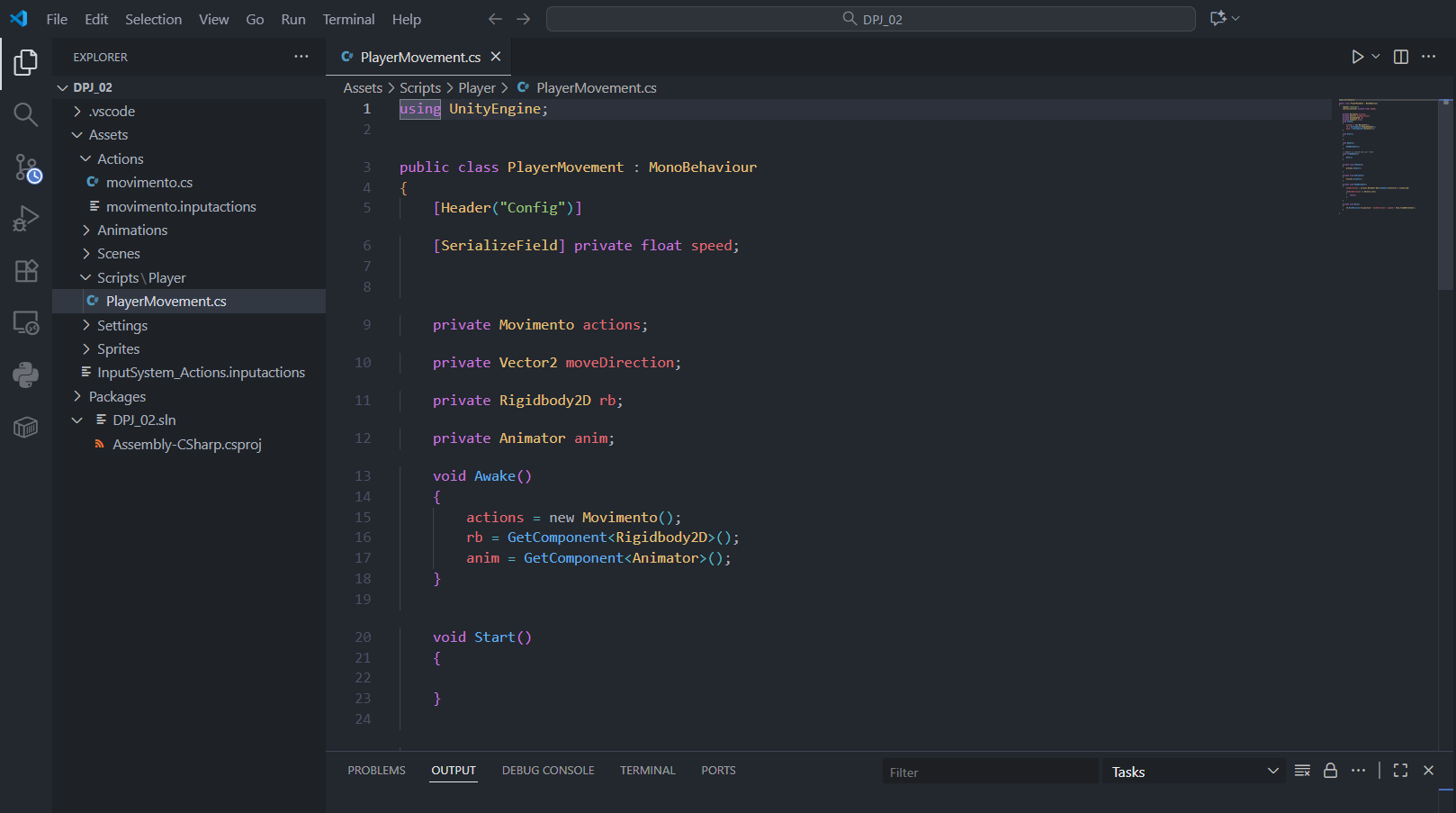


Fig.4

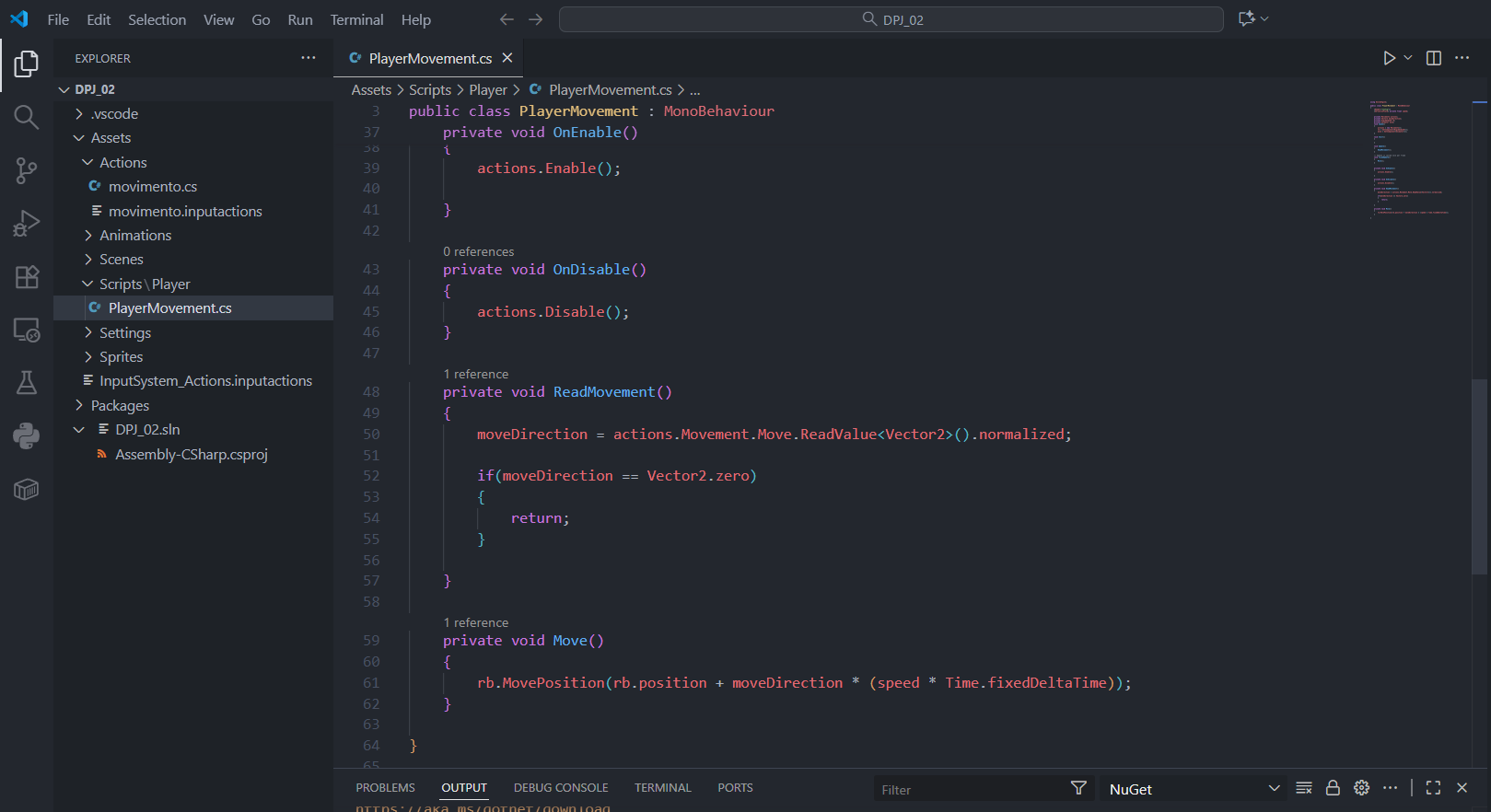
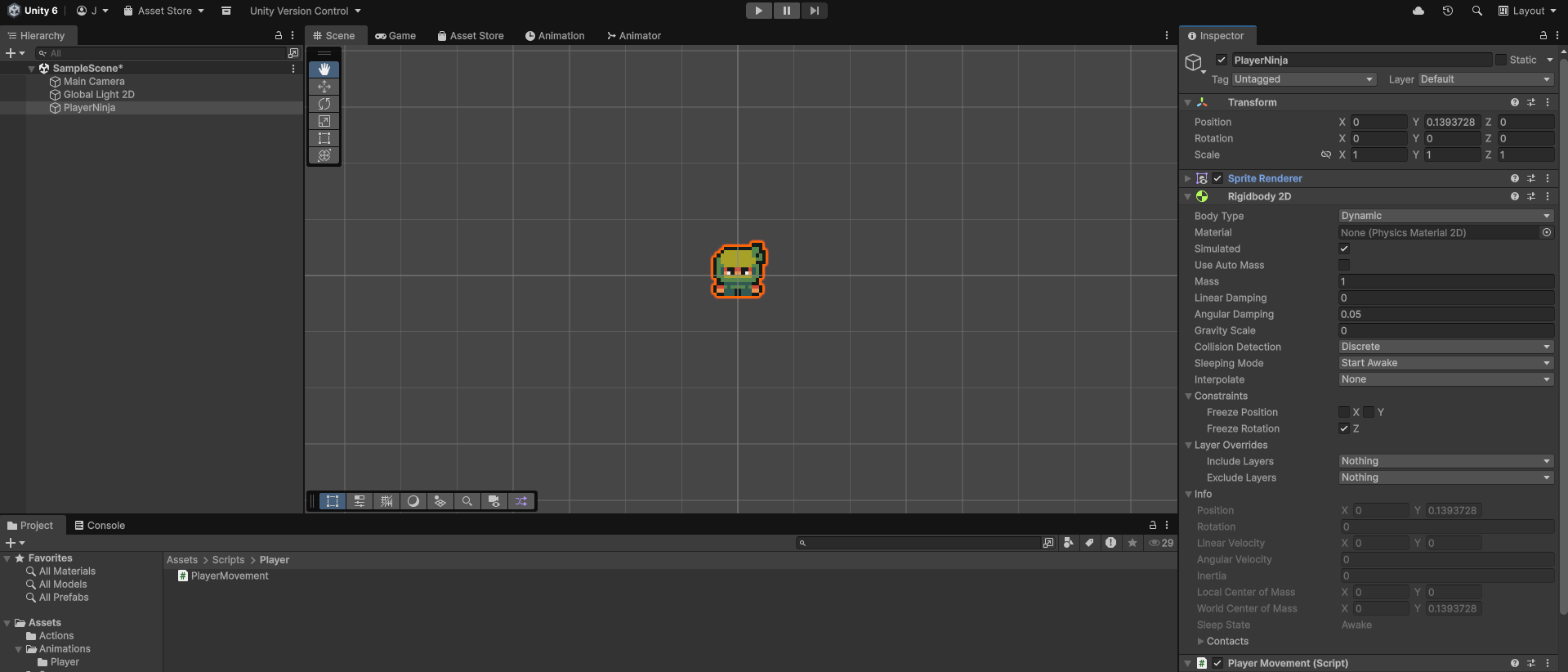


Fig.5

# Nível 5:

In Level 5 we added a component “RigidBody2D” to “PlayerNinja” so we can move forward, blocking the rotate point Y so the charapter can always stick to correct posicion(Fig.6). After we make some animations for different actions (move forward, move right, move down, move left), placing all in Animations folders for organize. We use the animater to configurate the “Blend Tree”, who allow to change automatually between the the animations and values (Fig7).

 Fig.6

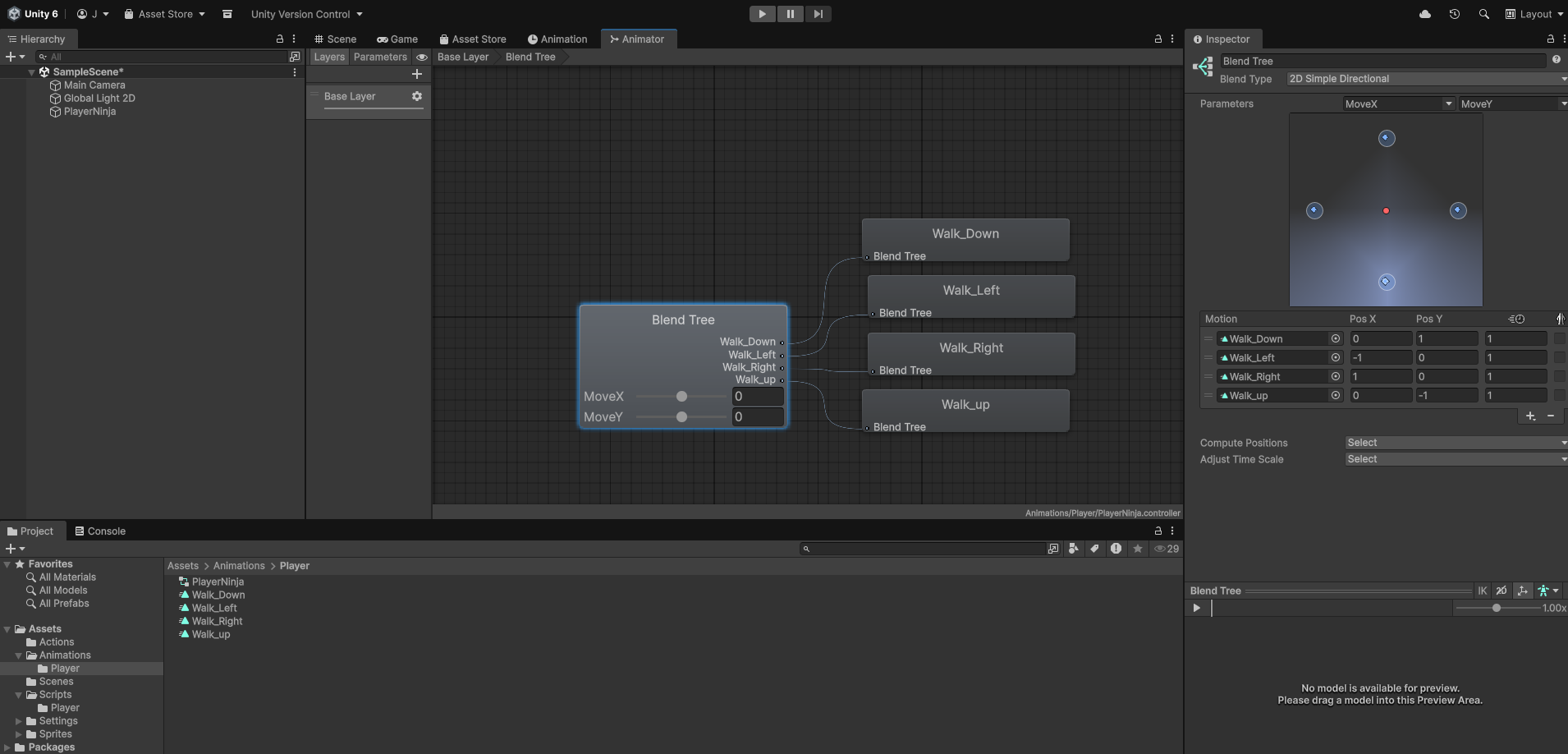


Fig.7