

Go Ahead and Jump – Exercise 5

Dr. David Israel Flores Granado

10/18/2023

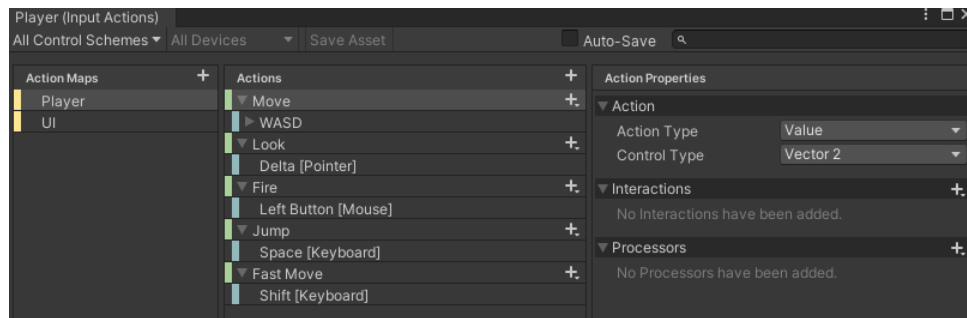
Team	Responsibility
Ariel	Jump and Horizontal/Vertical Double Displacement C# Script
Yadiel	Jump Horizontal/Vertical and Double Displacement Visual Script
Omar	PDF file

ACTION MATRIX

ACTIONS	MAPPINGS
Horizontal Movement	A and D keys, Left and Right arrows, gamepad left stick
Vertical Movement	W and S keys, Up and Down arrows,
Shoot	Left mouse click
Jump	Space Bar
Look	Mouse Look
Fast horizontal movement	Shift
Fast vertical movement	Shift

C# SCRIPTS

Movement Input



Script

```

public class shootermoveinput : MonoBehaviour
{
    3 references
    public float speed;
    1 reference
    public float rotationSpeed;
    2 references
    private float lookValue;
    1 reference
    private float jumpForce = 300;
    1 reference
    public Transform groundChecker;
    2 references
    public bool isGrounded;
    1 reference
    public LayerMask _layer;
    3 references
    private Vector2 movementValue;
    1 reference
    public GameObject prefab;
    2 references
    public GameObject shootPoint;
    0 references
    public Rigidbody rb;
    3 references
    public bool fastSpeed;

    0 references
    void Start()
    {
    }
}

```

```
private void Awake()
```

```
{
```

```
    Cursor.visible = false;
```

```
    Cursor.lockState = CursorLockMode.Locked;
```

```
}
```

0 references

```
void Update()
```

```
{
```

```
    isGrounded = Physics.CheckSphere(groundChecker.position, 0.3f, _layer);
```

```
    if (Input.GetKeyUp(KeyCode.LeftShift))
```

```
    {
```

```
        speed -= 5;
```

```
        fastSpeed = false;
```

```
    }
```

```
    transform.Translate(
```

```
        movementValue.x * Time.deltaTime, 0,
```

```
        movementValue.y * Time.deltaTime);
```

```
    transform.Rotate(0, lookValue*Time.deltaTime, 0);
```

```
}
```

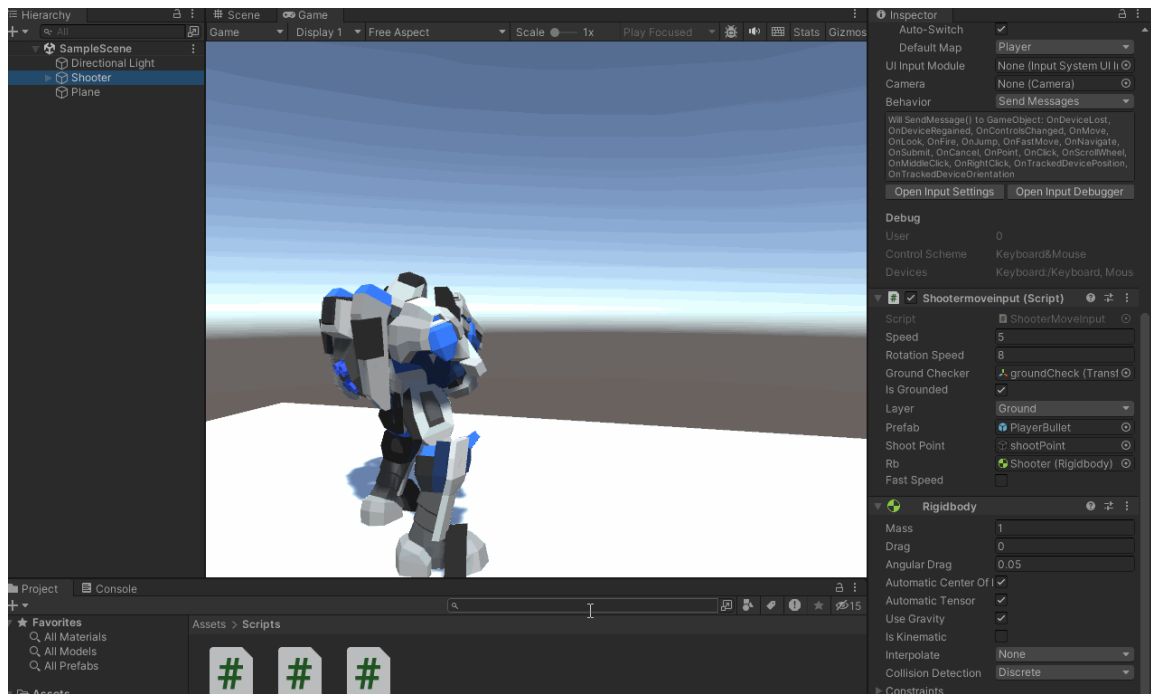
```

public void OnMove(InputValue value)
{
    movementValue = value.Get<Vector2>()*speed;
}
0 references
public void OnLook(InputValue value)
{
    lookValue = value.Get<Vector2>().x*rotationSpeed;
}
0 references
public void OnFire(InputValue value)
{
    if(value.isPressed)
    {
        GameObject clone = Instantiate(prefab);
        clone.transform.position = shootPoint.transform.position;
        clone.transform.rotation = shootPoint.transform.rotation;
    }
}
0 references
public void OnJump(InputValue value)
{
    if (value.isPressed && isGrounded)
    {
        GetComponent<Rigidbody>().AddForce(Vector3.up * jumpForce);
    }
}
0 references
public void OnFastMove(InputValue value)
{
    if (value.isPressed && fastSpeed == false)
    {
        speed += 5;
        fastSpeed = true;
    }
}
}

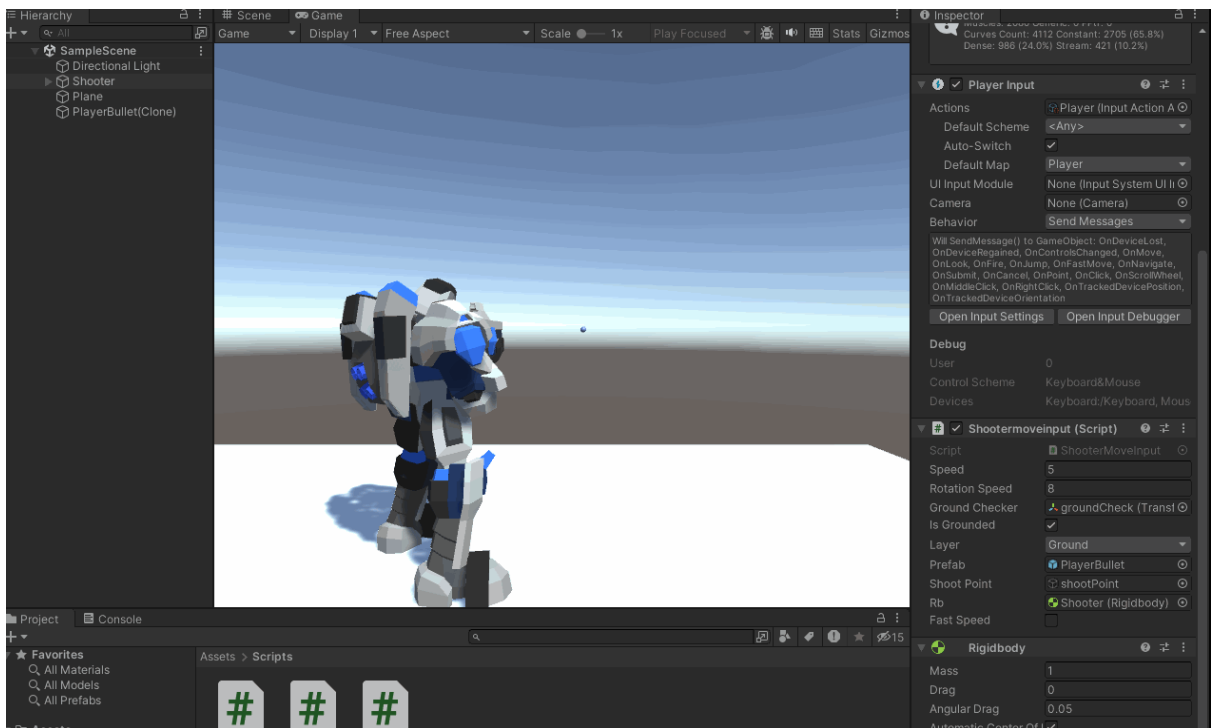
```

Gameplay

Jump

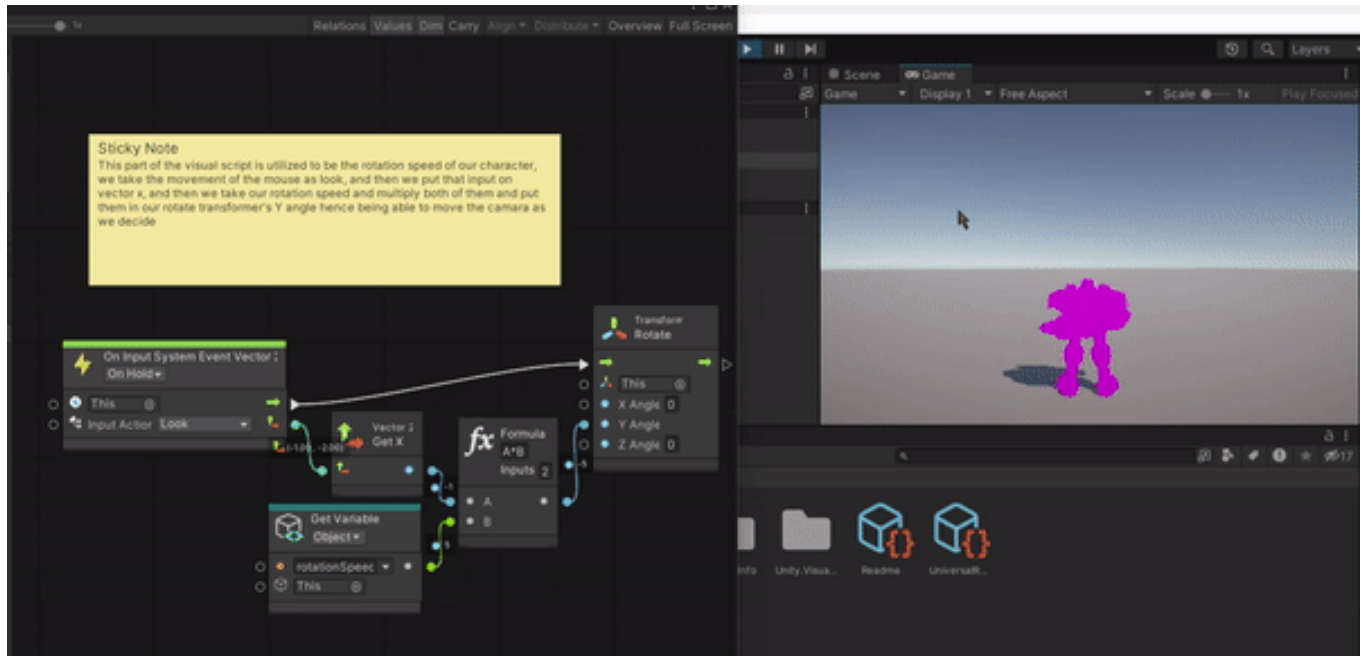


Double Movement

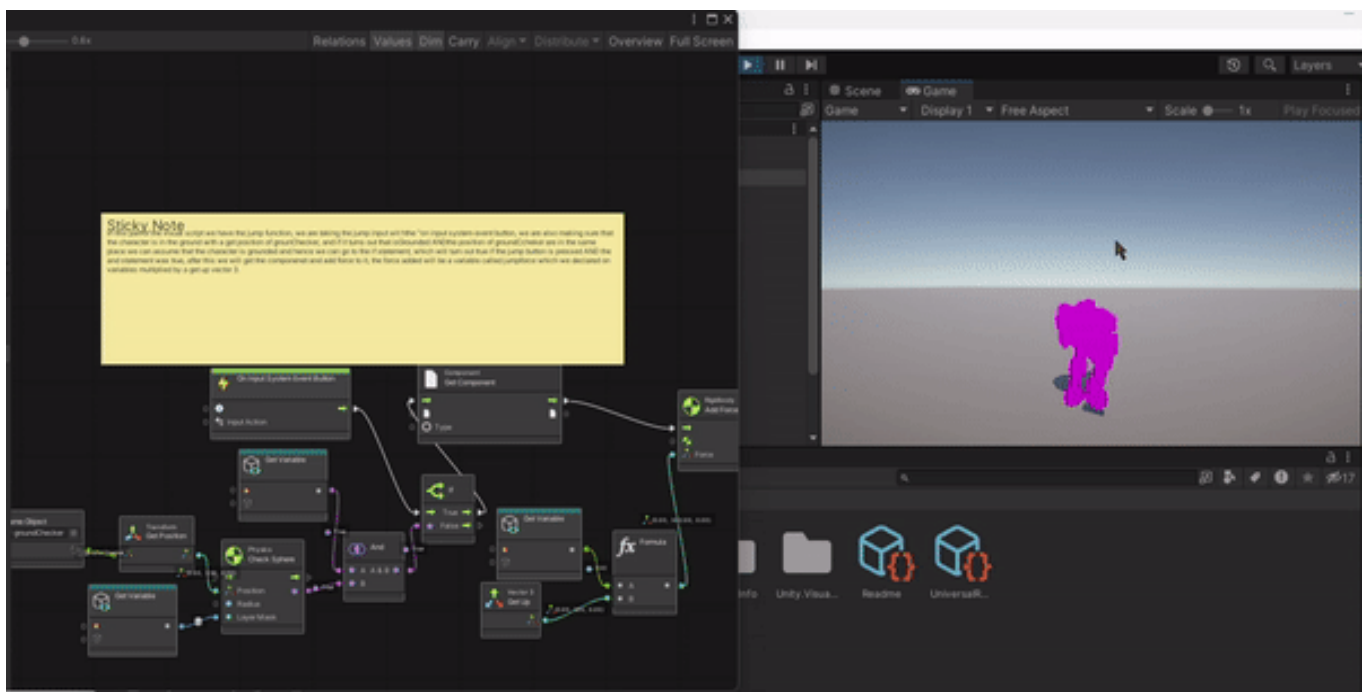


Visual Scripting/Gameplay

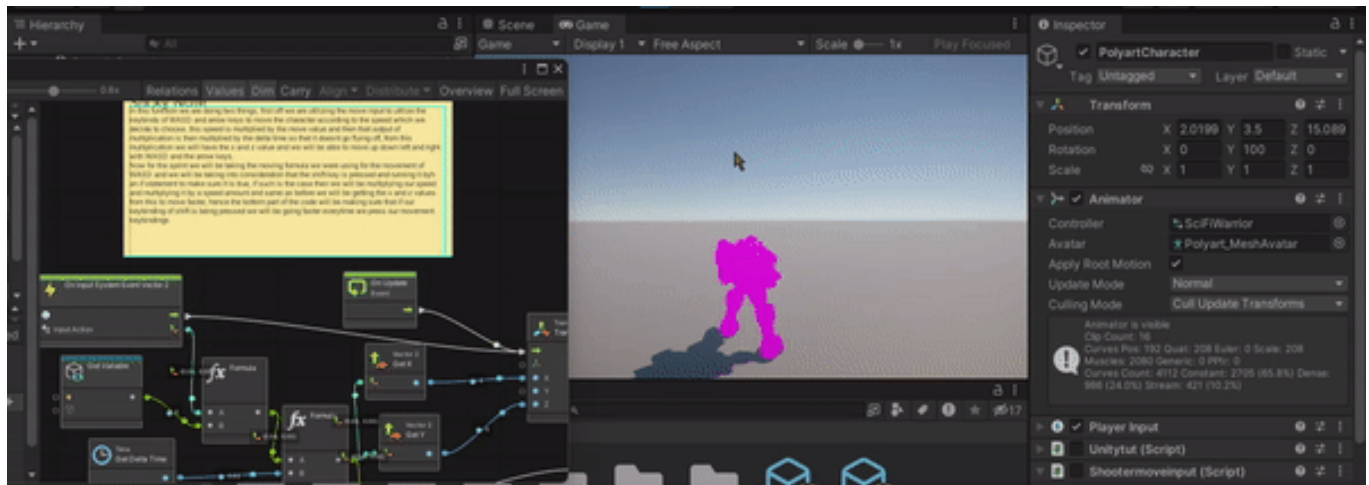
Look Around



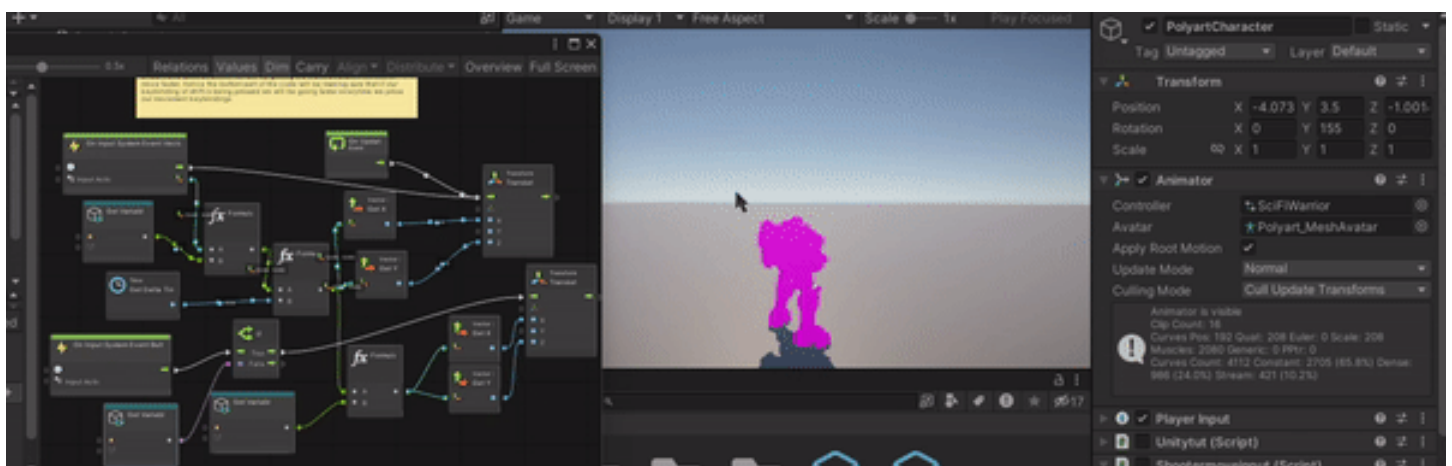
Jump



Movement



Shift



Shoot

