Super Mobile Controller

Functional Documentation

Version 3.0

BY AW Studio

Prefabs

Folder: SuperMobileController/Prefabs/Controller

• Joystick

A joystick component which output the amount and direction of the dialog stick continuously while dragging

Variable Name	**Description**
Input Name	The name if the input which passed to the result callback
Draggable Radius Modifier	The draggable area is calculated automatically. Use this variable to adjust the size of the draggable area if the calculation is not fit your images. *Negative value = Towards center
Relative Transform	The output value is relative to the player object by default. Set to camera object if you want the output relative to the camera.
Begin Drag Event	Call when start to touch the control Function(string InputName)
Drag Event	Call when dragging Function(string InputName, Vector2 result)
Analog Image	The sprite image of the analog. Not effective when set on runtime
Analog Area Image	The sprite image of the draggable area. Not effective when set on runtime

• Touch Input

An area to let user to drag. output the amount and direction continuously while dragging. Enable selection of "Touch Objects".

Variable Name	**Description**
Input Name	The name if the input which passed to the result callback
Relative Transform	The output value is relative to the player object by default. Set to camera object if you want the output relative to the camera.
Begin Drag Event	Call when start to touch the control Function(string InputName)

Dragging Result Relative To Initial Point	Always use the inital touch point as reference of output (good for character movement input). If as to false, the result will be relative to last Drag Event call (good for camera control)
Drag Event	Call when dragging Function(string InputName, Vector2 result)
End Drag Event	Call when drag ended Function(string InputName, Vector2 result)
Hold Second	The time that we consider the action as a hold action
Stop Drag End When Hold	If it is a hold action, don't call drag end callback
Stop Drag When Hold	If it is a hold action, don't call dragging callback
Hold Begin Event	Call when holded for {Hold Second} Function(string InputName, Vector2 result)
Hold End Event	Call when hold ended Function(string InputName, Vector2 result)
Drag Threshold	The threshold that we would recognize it as a drag action
Enable Drag Selection Box	Enable selection box for "Touch Objects" selection
Drag Selection Box Colour	The colour of the selection box

- **Touch Object**

An object that could be selected by a touch input.

Variable Name	**Description**
Tap Selection Event	Call when select an unselected touch object by a single tap Function(TouchObject objectSelected, Vector3 interactionPoint)
Tap Reselect Event	Call when reselect a selected touch object by a single tap Function(TouchObject objectSelected, Vector3 interactionPoint)

Area Select Event	Call when select a touch object by a area select Function(TouchObject objectSelected)
Deselect Event	Call when deselect a touch object Function(TouchObject objectSelected)
Selection Type	The type of this selection object
Area Selectable	Can this object be selected by area select
Tap Selectable	Can this object be selected by single tap
Deselect When Tap Select	Deselect this object when tapping the "Touch Input"
Deselect When Area Select	Deselect this object when doing area select in a "Touch Input"
Deselect When Reselect	Deselect the object when reselect
Unique Select By Type When Tap	Can only select one object of this selection type, other objects of this tyoe will be deselected
Touch Select Status	The status of this touch object
Selected Marker Tex	The texture used of the projector when this object is selected
Selected Marker Size	The size of the selection marker
Marker Ignore Layers	The Layers Ignored by the projector
Selected Market Colour	The colour of the marker
Block Lower Priority Tap	Only select the smaller priority object when tap
Tap Priority	The priority of the object
Selected Objects	A static List that stored all selected objects

- **Simple Skill Button**

A simple touch button input. Also allow to set quantity (used as item button)

Variable Name	**Description**
Input Name	The name if the input which passed to the result callback
Result Event	Call when touch ended Function(string InputName, Vector2 result)

Begin Touch Event	Call when touch begin Function(string InputName, Vector2 result)
Cooldown Second	The length of button cooldown. Set to 0 if no cooldown
Can Cancel	Cancel area will appear If set to yes. Drop at cancel area will cancel the action.
Quantity Left	Quantity of the button. Deduct by one when used. Button will be disabled if become 0. Set to －1 if unlimited.
Button Image	The sprite image of the button. Not effective when set on runtime. Use “SetButtonImage” method instead during runtime.

- **Target Skill Button**

A draggable button allow player to select target

Variable Name	**Description**
Input Name	The name if the input which passed to the result callback
Draggable Radius Modifier	The draggable area is calculated automatically. Use this variable to adjust the size of the draggable area if the calculation is not fit your images. *Negative value = Towards center
Relative Transform	The output value is relative to the player object by default. Set to camera object if you want the output relative to the camera.
Input Event	Call when drag ended Function(string InputName, Vector2 result)
Begin Drag Event	Call when start to touch the control Function(string InputName)
Dragging Event Call Interval	The interval in second that the dragging event being called once
Dragging Event	Call continuously when dragging based on Dragging Event Call Interval

Cooldown Second	The length of button cooldown. Set to 0 if no cooldown
Can Cancel	Cancel area will appear If set to yes. Drop at cancel area will cancel the action.
Quantity Left	Quantity of the button. Deduct by one when used. Button will be disabled if become 0. Set to －1 if unlimited.
Player	The player object (Must be set)
Skill Area Texture	The texture of the ground skill area marker. Black if transparent.
Skill Area Size	The size of the skill area. 1 is the original size of the Skill Area Texture
Skill Area Colour	Colour of the skill area
Skill Target Marker Texture	The texture of the ground target marker. Black if transparent.
Skill Marker Size	Size of the skill marker
Skill Marker Colour	Colour of the skill marker
Marker Ignore Layers	The Layers Ignored by the projector
Button Image	The sprite image of the button. Not effective when set on runtime. Use “SetButtonImage” method instead during runtime.
Analog Image	The sprite image of the analog. Not effective when set on runtime
Analog Area Image	The sprite image of the draggable area. Not effective when set on runtime
Center Offset	The offset of the center of the targer area

- **Directional Skill Button**

A draggable button allow player to select target direction

Variable Name	**Description**
Input Name	The name if the input which passed to the result callback

Draggable Radius Modifier	The draggable area is calculated automatically. Use this variable to adjust the size of the draggable area if the calculation is not fit your images. *Negative value = Towards center
Relative Transform	The output value is relative to the player object by default. Set to camera object if you want the output relative to the camera.
Input Event	Call when drag ended Function(string InputName, Vector2 result)
Begin Drag Event	Call when start to touch the control Function(string InputName)
Dragging Event Call Interval	The interval in second that the dragging event being called once
Dragging Event	Call continuously when dragging based on Dragging Event Call Interval
Cooldown Second	The length of button cooldown. Set to 0 if no cooldown
Can Cancel	Cancel area will appear If set to yes. Drop at cancel area will cancel the action.
Quantity Left	Quantity of the button. Deduct by one when used. Button will be disabled if become 0. Set to －1 if unlimited.
Player	The player object (Must be set)
Skill Target Marker Texture	The texture of the ground target marker. Black if transparent. Default set to *SuperMobileController/Prefabs/Controller/Image/SkillMarkerTextures/SkillMarker*
Skill Marker Aspect Ratio	The aspect ration of the texture
Skill Marker Size	The sizee of the skill marker
Skill Marker Colour	The Colour of the Skill Makrer
Marker Ignore Layers	The Layers Ignored by the projector

Button Image	The sprite image of the button. Not effective when set on runtime. Use “SetButtonImage” method instead during runtime.
Analog Image	The sprite image of the analog. Not effective when set on runtime
Analog Area Image	The sprite image of the draggable area. Not effective when set on runtime

• Cancel Area

An area only appear when dragging an element with Can Cancel = true

Variable Name	**Description**
Area Image	The sprite image of the area. Not effective when set on runtime
Active Area Image	The sprite image shows when dragging over the cancel button

• Dynamic Area

You may place any controls which inherited ControlBase (Joysick / SimpleTouchButton / DirectionSkillButton / TargetSkillButon) as a child object of a Dynamic Area. Please be noted that only one control is allowed in one Dynamic Area. The control will become a dynamic control which will move to the touch point when user touch any point inside the area.

Variable Name	**Description**
Hide Element on Release	Hide the element when no touching the area. Default is false

• Drag Button

A button that allow player to drag objects into the game

Variable Name	**Description**
Drop Group	The group name of this drop object
Begin Drag Event	Call when start to drag Function(string groupName)
Drag Event	Call when dragging Function(string groupName, Vector3?, interactionPoint, GameObject DraggingObject, GameObject dropTarget)

End Drag Event	Call when drop Function(string groupName, Vector3?, interactionPoint, GameObject DropObject, GameObject dropTarget)
Button Image	The image of the button
Show Dragging Object	Do we show dragging object
Show Dragging Object When Undropable	Do we show dragging object when we can't drop it
Dragging Object	The object to be created when dragging
Drag Y Offset	The Y offset of the dragging object
Dragging Object Rotation	The euler rotation of the dragging object
Create Drop Object	Do we show drop object
Drop Y Offset	The Y offset of the drop object
Drop Object Rotation	The euler rotation of the drop object
Drop Object	The object to be dropped

- **Drop Target**

A object that allow player to drop object from a drag button

Variable Name	**Description**
Drop Priorty	The priority of the drop target. When we have more then one droppable target, we drop on the one with smallest priority
Allowed Drop Group	What drop groups do we allow to drop on this target
Allowed Drop Tint	When dragging, we add a tint colour to this object if it is allow to drop. White colour mean no color
Disallowed Drop Tint	When dragging, we add a tint colour to this object if it is disallow to drop. White colour mean no color
Drop At Fixed Point	Do we force to frop on a fixed point
Fixed Point	The fixed point to be dropped (local position space)

- **Button Collection**

A collection of simple touch buttons arranged in round shape. You may put any number of simple touch buttons as children of this control.

Variable Name	**Description**

Radius	The radius of the circle
Degree In Between	The degree between two buttons
Degree Abjust	Rotate the whole set of buttons by this degree value

Scripts

Folder: SuperMobileController/Script

- **Common.cs**

Define all common functions

- **Enums.cs**

Define all enums

- **Event.cs**

Define all events

- **ControlBase.cs**

Base class of the controls. Some common variables are defined in it.

- **Joystick.cs**

A joystick for character movement

Accessibility	**Method Name**	**Description**
private	OnPointerDown	Handler for touch down event
private	OnPointerUp	Handler for touch ended event
private	OnDrag	Handler for dragging event
private	OnEndDrag	Handler for drag ended event
private	ReturnResult	Function to invoke "Input Event"
private	GetFinalResult	Function to calculate final result vector
private	GetRelativeTransformedPosition	Function to transform the vector relative to the "Relative Transform" setting

- **TouchInput.cs**

Allow to perform drag, fast drag, hold and select touch objects

Accessibility	**Method Name**	**Description**
private	OnPointerDown	Handler for touch down event
private	OnPointerUp	Handler for touch up event
private	OnDrag	Handler for dragging event
private	OnEndDrag	Handler for dragging ended event
private	ReturnResult	Function to invoke "Input Event"
private	GetResultPosition	Function to get result position

private	GetFinalResult	Function to calculate final result vector
private	Set Hold	Function to determine if action a "Hold" action

- **TouchObject.cs**

Object that could be selected by a touch input

Accessibility	Method Name	Description
public	TapSelect	Called by Touch Input when select an unselected object with tap select
public	AreaSelecting	Called by Touch Input when selecting an object with area (Still Dragging)
public	UndoAreaSelecting	Called by Touch Input when deselecting an object with area (Still Dragging)
public	AreaSelect	Called by Touch Input when selected an object with area (selection done)
public	Deselect	Call to deselect the object

- **SimpleTouchButton.cs**

A simple touch button

Accessibility	Method Name	Description
private	OnPointerDown	Handler for touch down event
private	OnPointerUp	Handler for touch ended event
private	SetCancelAreasEnabled	Enable / disable cancel areas
private	UpdateCooldown	Timer for cooldown countdown
private	ReturnResult	Function to invoke “Input Event”
public	SetEnabled	Enable / Disable the button
public	SetQuantity	Set “Quantity Left”, button will be reenabled when Quantity is not 0 and disabled when Quantity is 0
public	SetCoolDown	Set cooldown of the button in seconds
public	SetButtonImage	Set sprite to replace the button image

- **TargetSkillButton.cs**

Button allow you to output the input poisiotn and set skill marker to the player

Accessibility	Method Name	Description
private	OnPointerDown	Handler for touch down event
private	OnPointerUp	Handler for touch ended event
private	OnBeginDrag	Handler for drag started
private	OnDrag	Handler for dragging event
private	OnEndDrag	Handler for drag ended event
private	ReturnResult	Function to invoke “Input Event”

private	DraggingCallBack	Call between drag begin and drag end every “draggingEventCallInterval” Seconds
private	SetCancelAreasEnabled	Enable / disable cancel areas
private	UpdateCooldown	Timer for cooldown countdown
private	LateUpdate	Call on every draw to update skill marker’s position
public	SetEnabled	Enable / Disable the button
public	SetQuantity	Set “Quantity Left”, button will be reenabled when Quantity is not 0 and disabled when Quantity is 0
public	SetCoolDown	Set cooldown of the button in seconds
public	SetSkillMarkerSize	Set for size of the ground skill marker
public	SetSkillAreaSize	Set for size of the ground area marker
public	SetButtonImage	Set sprite to replace the button image

- **DirectionSkillButton.cs**

Button allow you to output the input direction and set skill marker to the player

Accessibility	**Method Name**	**Description**
private	OnPointerDown	Handler for touch down event
private	OnPointerUp	Handler for touch ended event
private	OnBeginDrag	Handler for drag started
private	OnDrag	Handler for dragging event
private	OnEndDrag	Handler for drag ended event
private	ReturnResult	Function to invoke “Input Event”
private	DraggingCallBack	Call between drag begin and drag end every “draggingEventCallInterval” Seconds
private	SetCancelAreasEnabled	Enable / disable cancel areas
private	UpdateCooldown	Timer for cooldown countdown
private	LateUpdate	Call on every draw to update skill marker’s position
public	SetEnabled	Enable / Disable the button
public	SetQuantity	Set “Quantity Left”, button will be reenabled when Quantity is not 0 and disabled when Quantity is 0
public	SetCoolDown	Set cooldown of the button in seconds
public	SetSkillMarkerSize	Set for size of the ground skill marker
public	SetButtonImage	Set sprite to replace the button image

- **CancelArea.cs**

Drag a cancelable object to this area will cancel the action

Accessibility	Method Name	Description
public	SetAreaImage	Set sprite to replace the area image
public	SetActiveImage	Set sprite to replace the active area image

- **DynamicArea.cs**

Make the first ControlBase element become dynamic in position inside this area

Accessibility	Method Name	Description
private	OnPointerDown	Function to set child element position to touch position and fire OnPointerDown of child element
private	OnPointerUp	Function to reset child element position and fire OnPointerUp of child element
private	OnBeginDrag	Function to fire OnBeginDrag of child element
private	OnDrag	Function to fire OnDrag of child element
private	OnEndDrag	Function to fire OnEndDrag of child element

- **CircleButtonCollection.cs**

Make the children as a collection of SimpleTouchButtons arranged in circle shape.

Accessibility	Method Name	Description
public	Show	Function to show the child buttons
public	FireButtonAtPosition	Function to fire button click at the touch position
public	Hide	Function to hide the child buttons
private	getCirclePosition	Function for getting button position in a circle

- **DragButton.cs**

Allow to drag and create objects into the scene

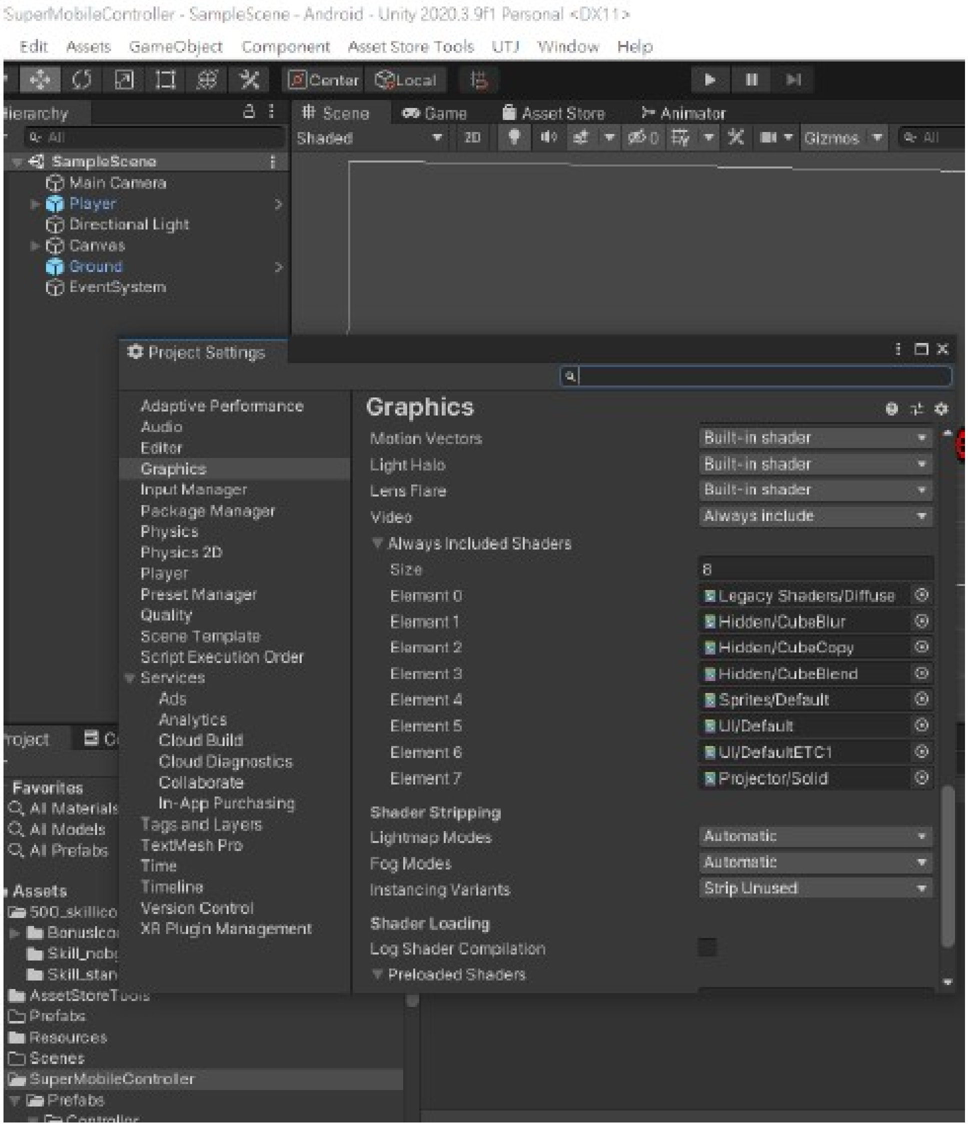
Accessibility	Method Name	Description
private	OnBeginDrag	Handler for drag started
private	OnDrag	Handler for dragging
private	OnEndDrag	Handler for drag ended

- **DropTarget.cs**

The drop target for drag button.

Setup

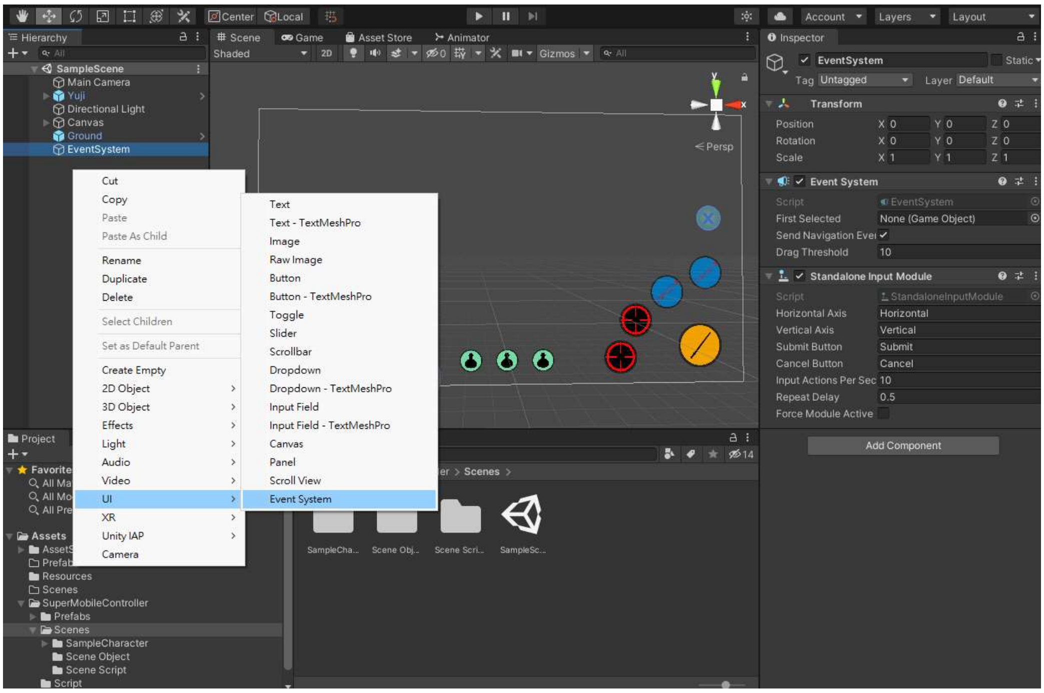
1. Go to Edit -> Project Settings -> Graphics. A SolidProjector to Always Included Shader **



Graphics

Motion Vectors	Built-in shader
Light Halo	Built-in shader
Lens Flare	Built-in shader
Video	Always include
Always Included Shaders	
Size	8
Element 0	Legacy Shaders/Diffuse
Element 1	Hidden/CubeBlur
Element 2	Hidden/CubeCopy
Element 3	Hidden/CubeBlend
Element 4	Sprites/Default
Element 5	UI/Default
Element 6	UI/DefaultETC1
Element 7	Projector/Solid
Shader Stripping	
Lightmap Modes	Automatic
Fog Modes	Automatic
Instancing Variants	Strip Unused
Shader Loading	
Log Shader Compilation	□
Preloaded Shaders	

2. Add Event System object to scene



Inspector

EventSystem

Tag Untagged Layer Default

Transform

Position X 0 Y 0 Z 0
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1

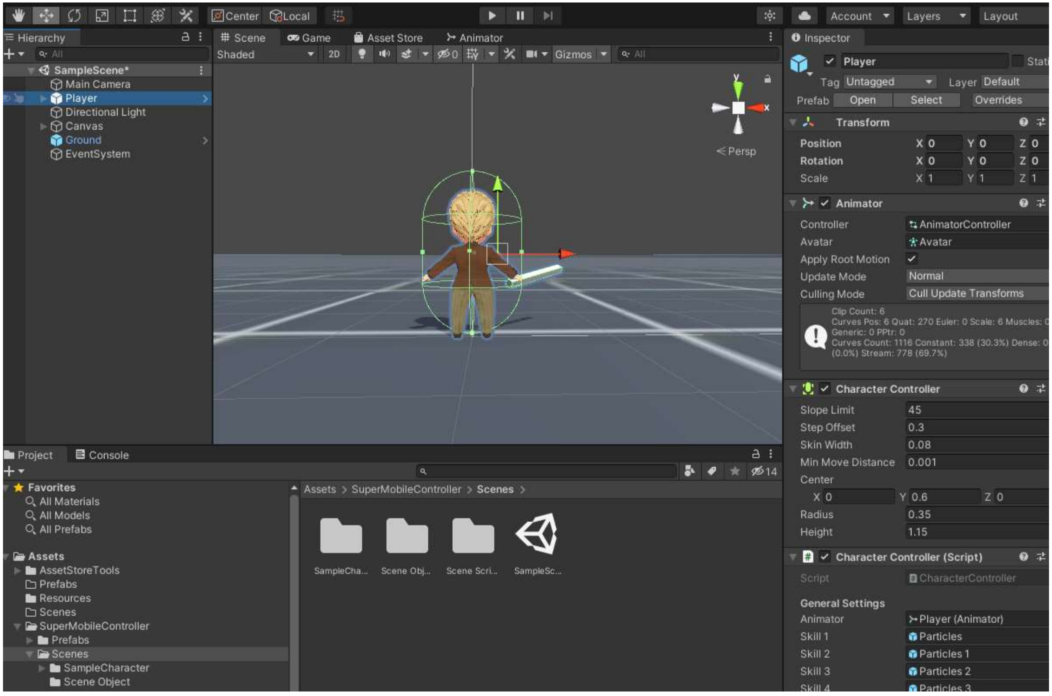
Event System

Script EventSystem
First Selected None (Game Object)
Send Navigation Events ✓
Drag Threshold 10

Standalone Input Module

Script StandaloneInputModule
Horizontal Axis Horizontal
Vertical Axis Vertical
Submit Button Submit
Cancel Button Cancel
Input Actions Per Sec 10
Repeat Delay 0.5
Force Module Active □

3. Add player object to scene



Inspector

Player

Tag Untagged Layer Default

Prefab Open Select Overrides

Transform

Position X 0 Y 0 Z 0
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1

Animator

Controller AnimatorController
Avatar Avatar
Apply Root Motion ✓
Update Mode Normal
Culling Mode Cull Update Transforms

Clip Count: 6
Curves Pos: 6 Quat: 270 Euler: 0 Scale: 6 Muscles: 0
Generic: 0 PPtr: 0
Curves Count: 1116 Constant: 338 (30.3%) Dense: 0
(0.0%) Stream: 778 (69.7%)

Character Controller

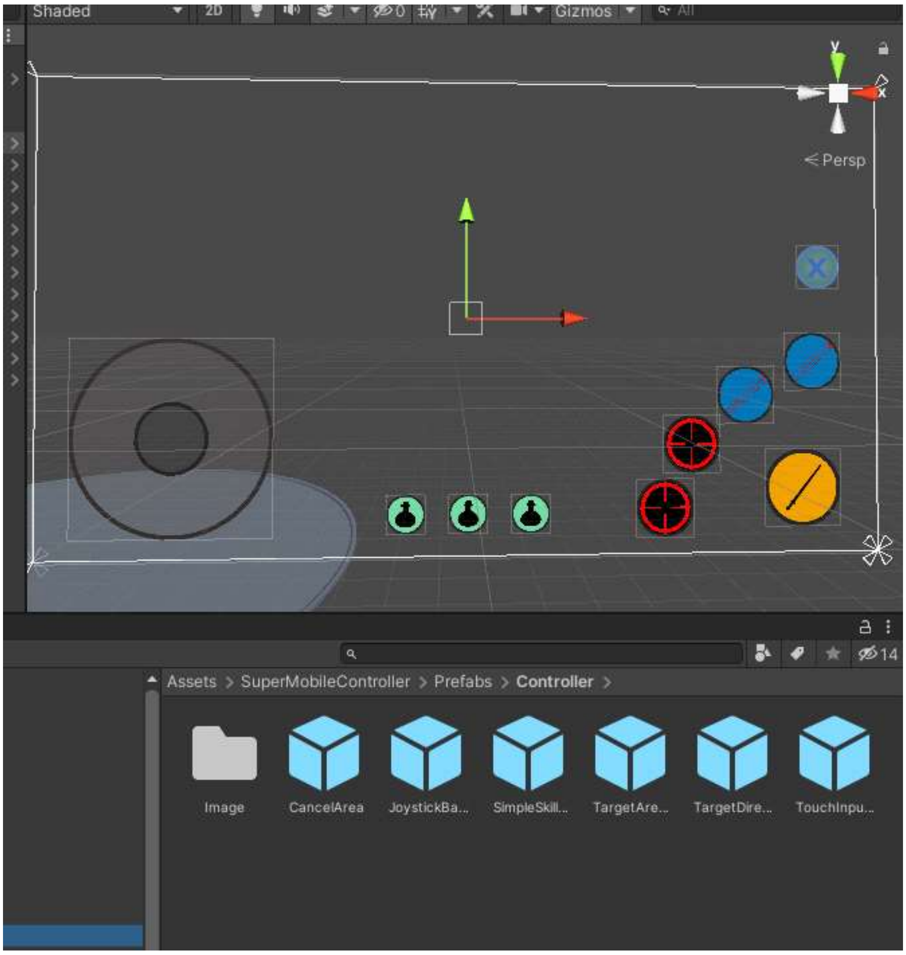
Slope Limit 45
Step Offset 0.3
Skin Width 0.08
Min Move Distance 0.001
Center X 0 Y 0.6 Z 0
Radius 0.35
Height 1.15

Character Controller (Script)

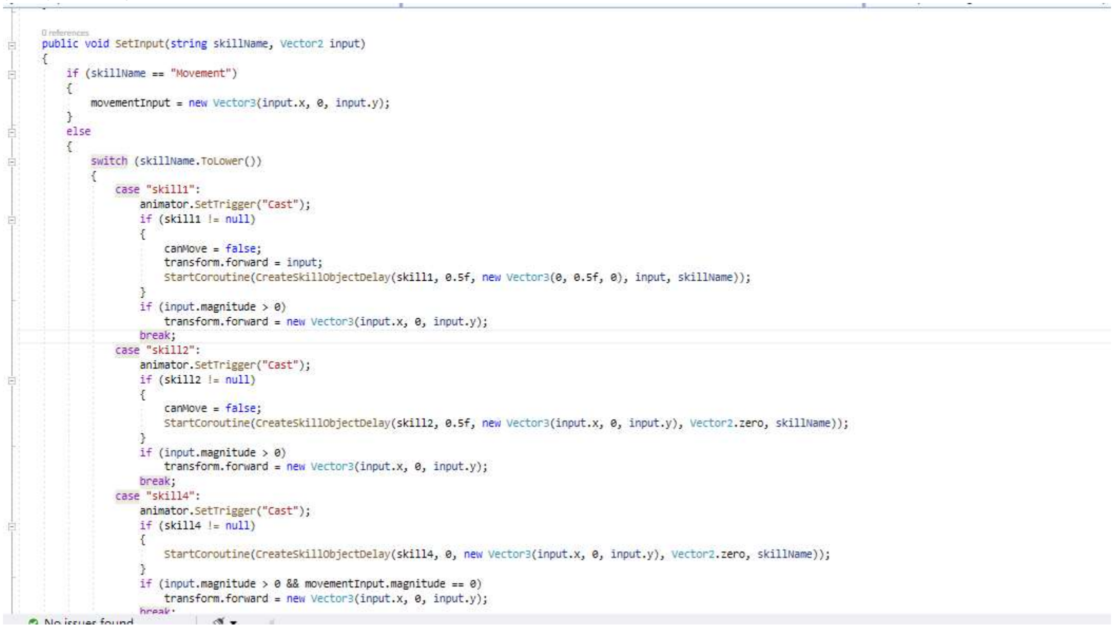
Script CharacterController

General Settings
Animator Player (Animator)
Skill 1 Particles
Skill 2 Particles 1
Skill 3 Particles 2
Skill 4 Particles 3

4. Drag the prefab to the Canvas to create a gamepad

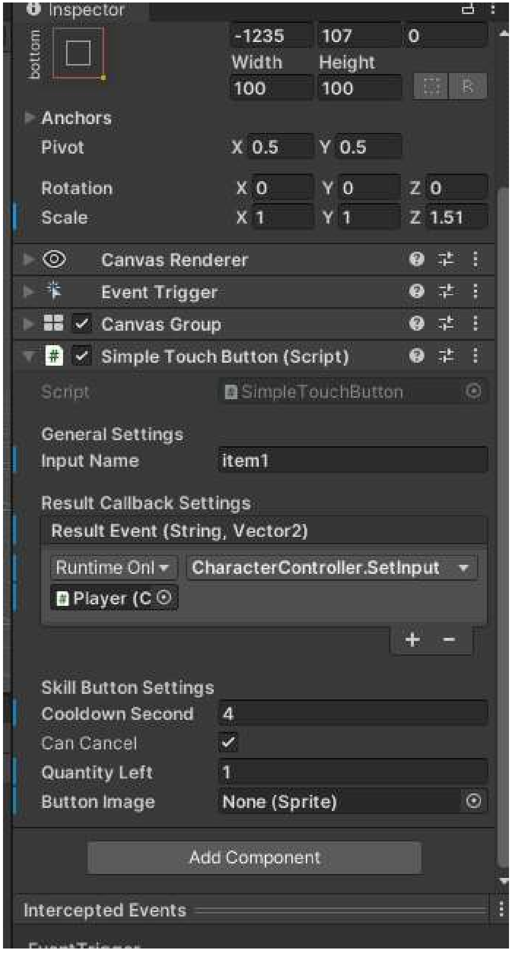


5. Write a handler function in a new PlayerController script and attach it to the player object



```
public void SetInput(string skillName, Vector2 input)
{
    if (skillName == "Movement")
    {
        movementInput = new Vector3(input.x, 0, input.y);
    }
    else
    {
        switch (skillName.ToLower())
        {
            case "skill1":
                animator.SetTrigger("Cast");
                if (skill1 != null)
                {
                    canMove = false;
                    transform.forward = input;
                    StartCoroutine(CreateSkillObjectDelay(skill1, 0.5f, new Vector3(0, 0.5f, 0), input, skillName));
                }
                if (input.magnitude > 0)
                    transform.forward = new Vector3(input.x, 0, input.y);
                break;
            case "skill2":
                animator.SetTrigger("Cast");
                if (skill2 != null)
                {
                    canMove = false;
                    StartCoroutine(CreateSkillObjectDelay(skill2, 0.5f, new Vector3(input.x, 0, input.y), Vector2.zero, skillName));
                }
                if (input.magnitude > 0)
                    transform.forward = new Vector3(input.x, 0, input.y);
                break;
            case "skill4":
                animator.SetTrigger("Cast");
                if (skill4 != null)
                {
                    StartCoroutine(CreateSkillObjectDelay(skill4, 0, new Vector3(input.x, 0, input.y), Vector2.zero, skillName));
                }
                if (input.magnitude > 0 && movementInput.magnitude == 0)
                    transform.forward = new Vector3(input.x, 0, input.y);
                break;
```

6. Add the handler to the corresponding callback event and setup other parameters



Inspector

-1235 107 0

Width 100 Height 100

Anchors

Pivot X 0.5 Y 0.5

Rotation X 0 Y 0 Z 0

Scale X 1 Y 1 Z 1.51

Canvas Renderer

Event Trigger

☑ Canvas Group

☑ Simple Touch Button (Script)

Script: SimpleTouchButton

General Settings

Input Name: item1

Result Callback Settings

Result Event (String, Vector2)

Runtime Onl ▾ CharacterController.SetInput ▾

Player (C

Skill Button Settings

Cooldown Second: 4

Can Cancel: ☑

Quantity Left: 1

Button Image: None (Sprite)

Add Component

Intercepted Events

7. Test the input