# Touch Manager 2.1 Documentation

Welcome! Select the desired paragraph in the table of contents:

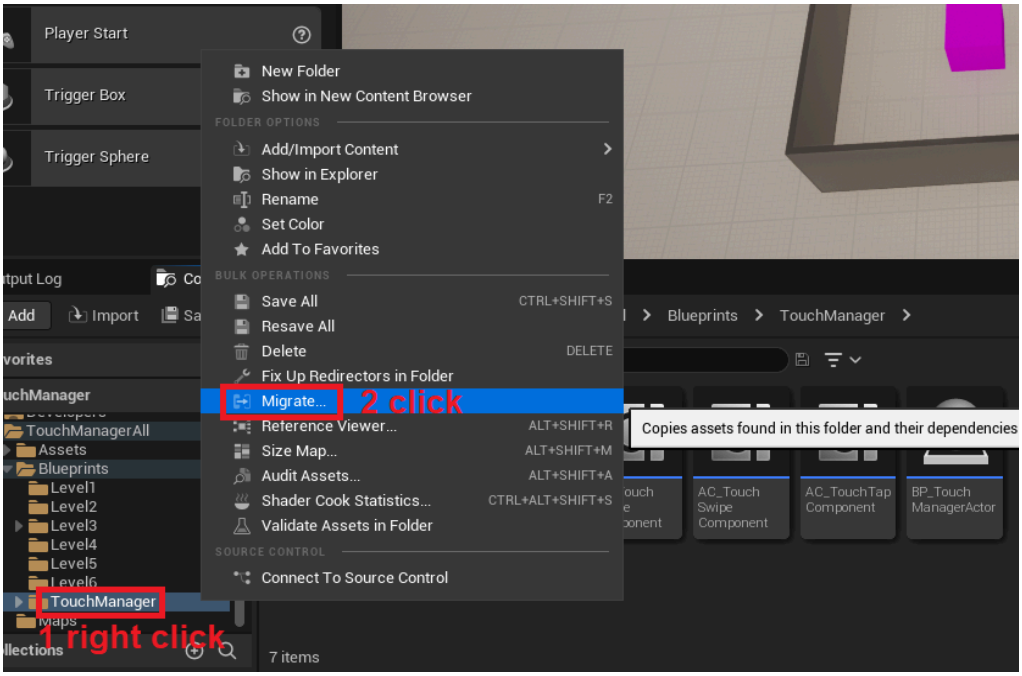


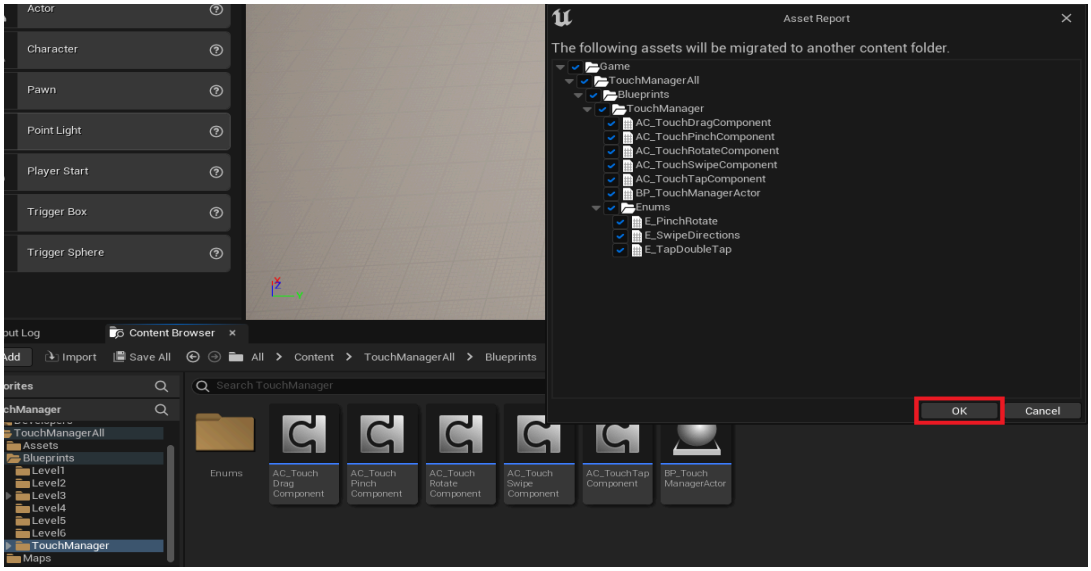
# Video tutorials

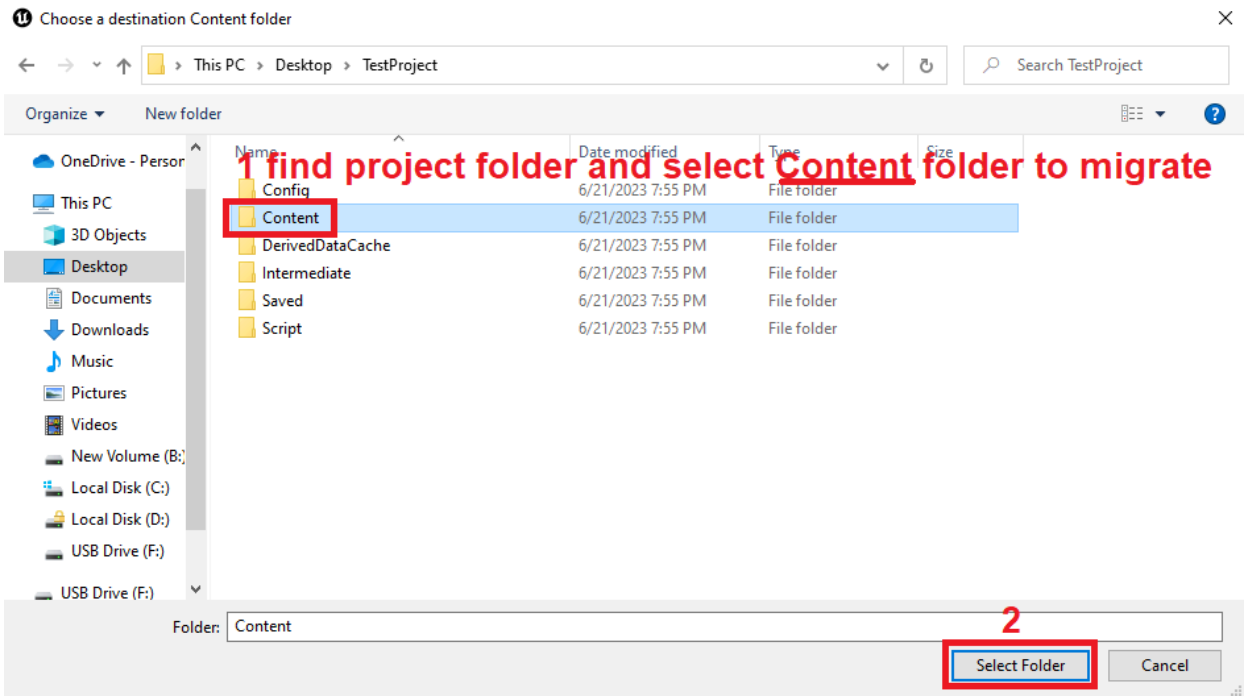
https://www.youtube.com/playlist?list=PL7K9vb0AOExHv-_3qfQQAHJO4d_Q4wveq

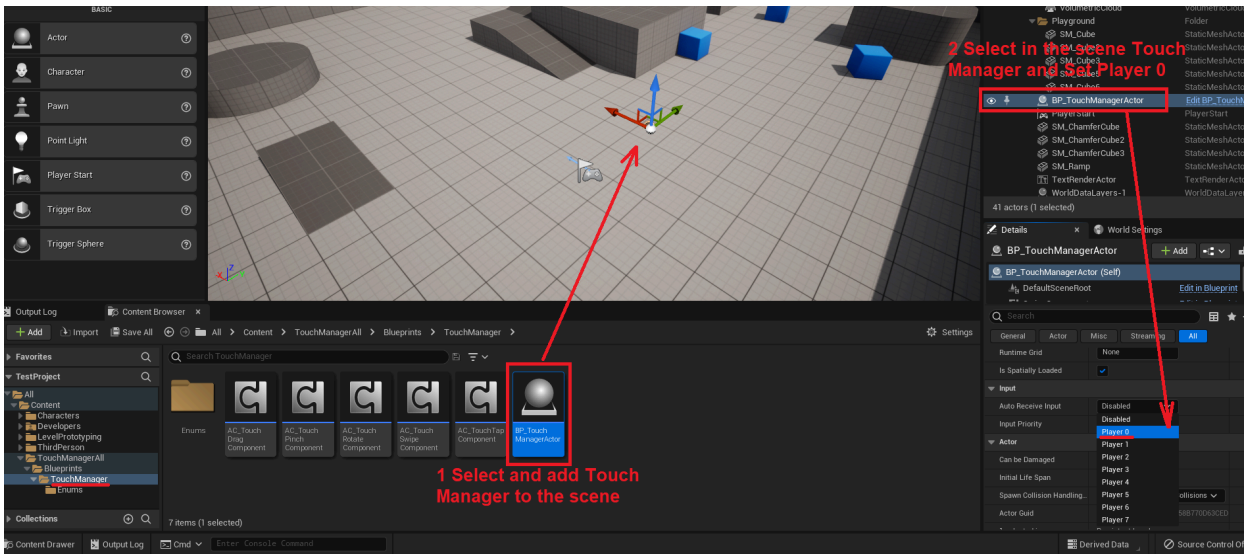
# How to add Touch Manager to your project

You need to make a few steps, on Youtube:
Also, you can use a new **AC_TouchManager2** component in multiplayer.
1. Migrate the Touch Manager folder to **your project** in a **Content** folder:


Player Start
Trigger Box
Trigger Sphere
New Folder
Show in New Content Browser
FOLDER OPTIONS
Add/Import Content
Show in Explorer
Rename F2
Set Color
Add To Favorites
BULK OPERATIONS
Save All CTRL+SHIFT+S
Resave All
Delete DELETE
Fix Up Redirectors in Folder
Migrate...
2 click
Reference Viewer... ALT+SHIFT+R
Size Map... ALT+SHIFT+M
Audit Assets... ALT+SHIFT+A
Shader Cook Statistics... CTRL+ALT+SHIFT+S
Validate Assets in Folder
SOURCE CONTROL
Connect To Source Control
Copies assets found in this folder and their dependencies
Blueprints > TouchManager
TouchManagerAll
Assets
Blueprints
Level1
Level2
Level3
Level4
Level5
Level6
TouchManager
Maps
1 right click
7 items
AC_Touch Swipe Component
AC_TouchTap Component
BP_Touch ManagerActor


Actor
Character
Pawn
Point Light
Player Start
Trigger Box
Trigger Sphere
Asset Report
The following assets will be migrated to another content folder.
Game
TouchManagerAll
Blueprints
TouchManager
AC_TouchDragComponent
AC_TouchPinchComponent
AC_TouchRotateComponent
AC_TouchSwipeComponent
AC_TouchTapComponent
BP_TouchManagerActor
Enums
E_PinchRotate
E_SwipeDirections
E_TapDoubleTap
OK
Cancel
Content Browser
Import
Save All
All > Content > TouchManagerAll > Blueprints
Search TouchManager
Enums
AC_Touch Drag Component
AC_Touch Pinch Component
AC_Touch Rotate Component
AC_Touch Swipe Component
AC_TouchTap Component
BP_Touch ManagerActor



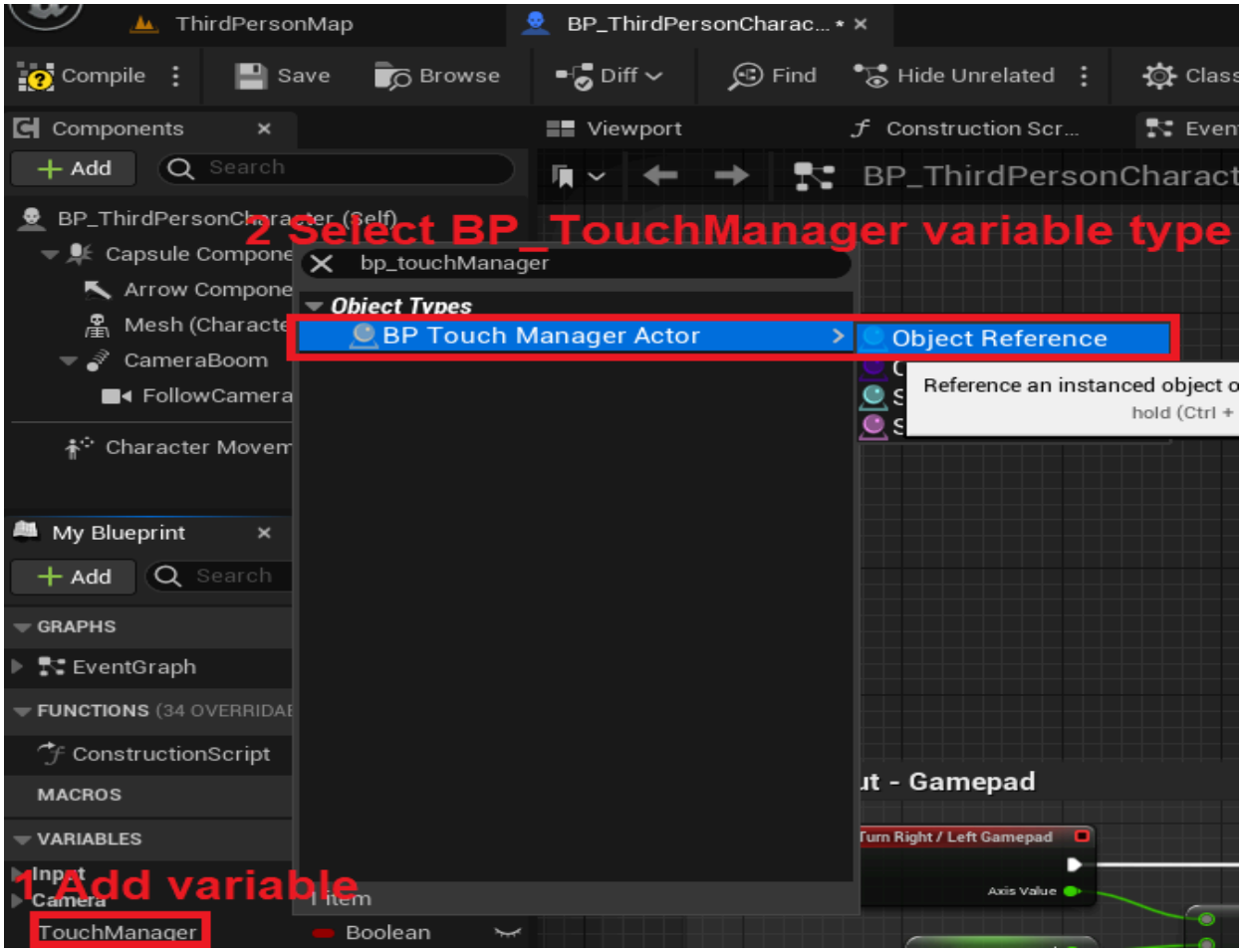




## Add Touch Manager PTR to your Character or Actor

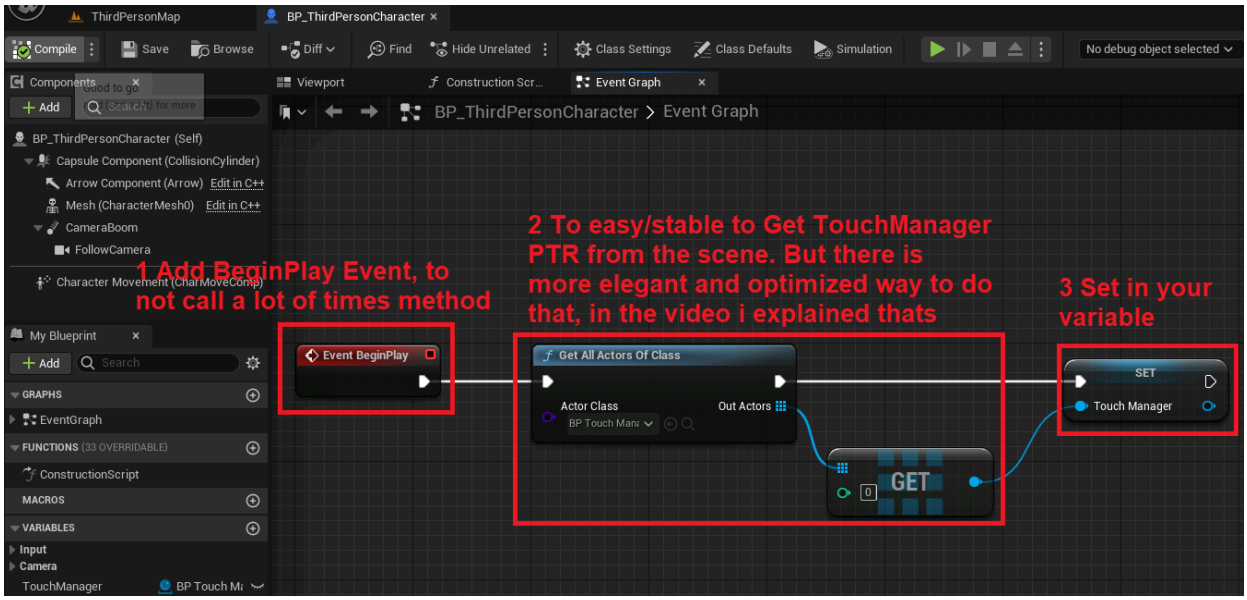
After Touch Manager setup in the scene, we need to take a PTR in your Character or Actor class where you will have Touch Events.

You can take PTR in different ways, the main one at the video: https://youtu.be/pM7qqi4nLAQ

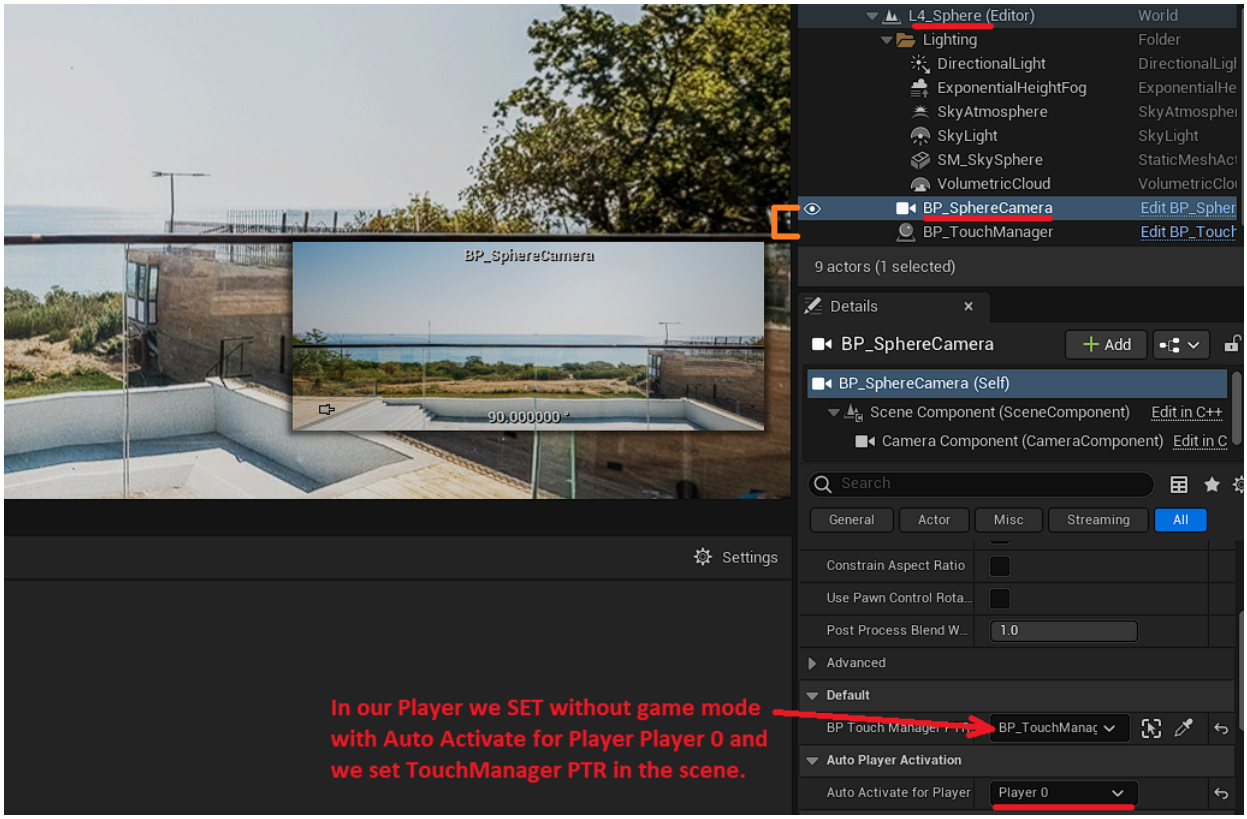
Other Simple to GetAllActors of class to take that in any AActor, widgets and other classes, but WARNING it's not optimized way.




After that variable, you can easily add BeginPlay event and use very simply node GetAllActors of class, but not optimized, more elegant solutions, storing i will explain in the video about that part. Now you can get PTR:



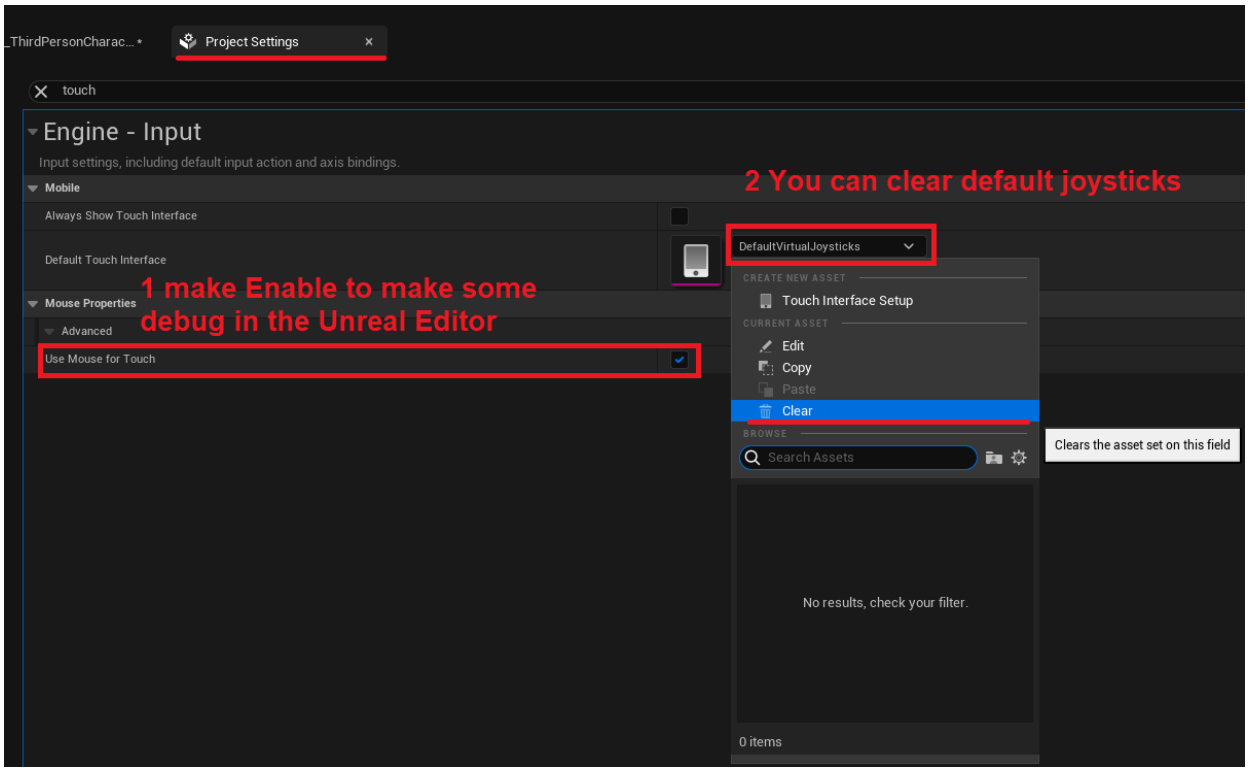
**Another method** can be done, if your player is on the scene with **AutoPlayerActivation “Player 0”** and **Auto Activate For Player “Player 0”**

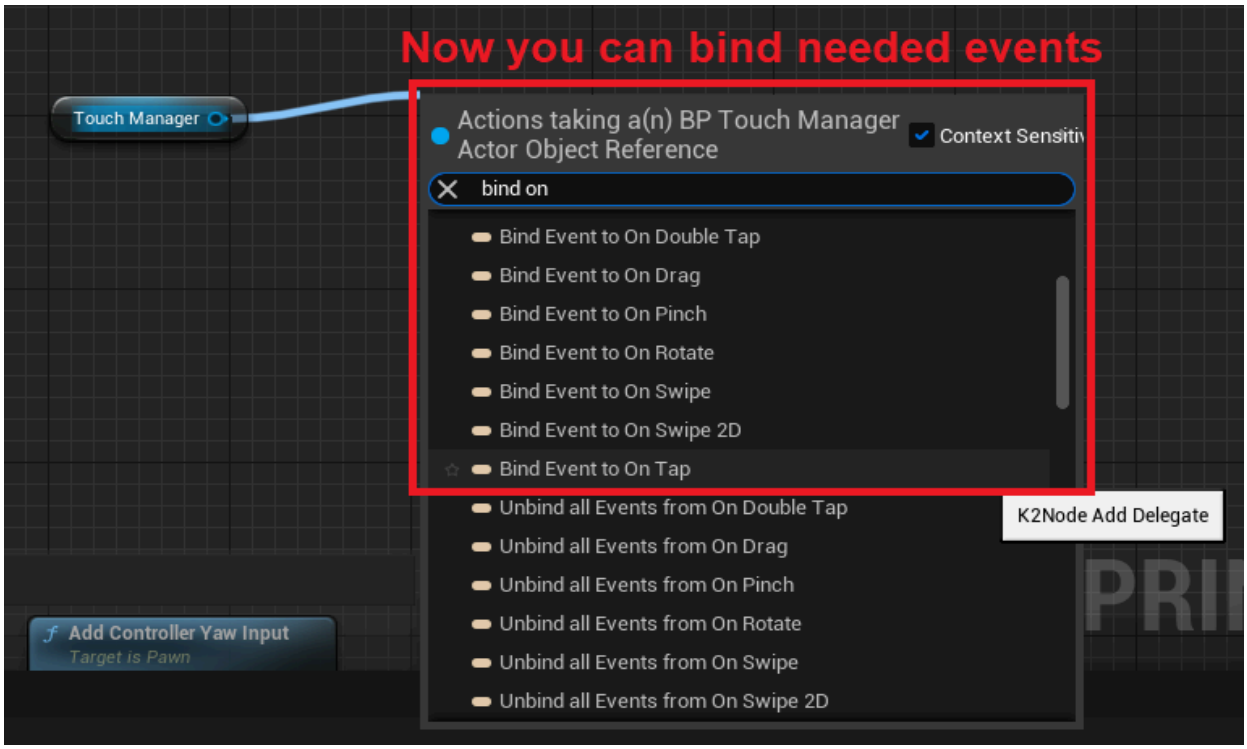


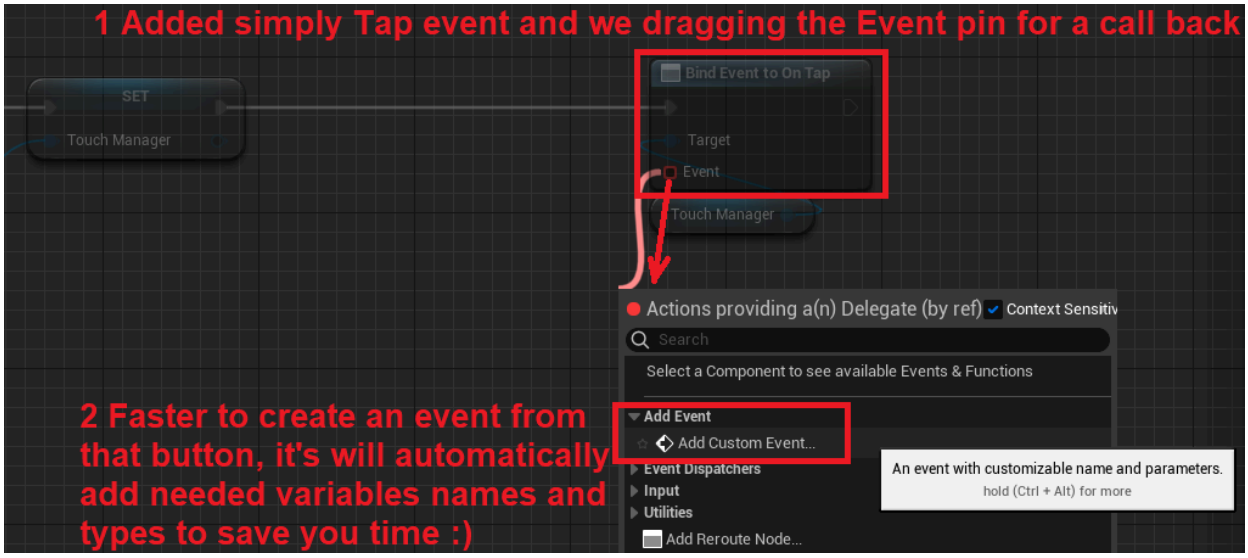

And we pass in player TouchManager PTR in the level public variable.

# Binding and Unbinding to the touch events

To easier debug in the Unreal Editor we can Enable [UseMouseForTouch]
And if you dont need default joists you can clear the default joysticks.




1 Added simply Tap event and we dragging the Event pin for a call back
SET
Touch Manager
Bind Event to On Tap
Target
Event
Touch Manager
Actions providing a(n) Delegate (by ref)
Context Sensitiv
Search
Select a Component to see available Events & Functions
Add Event
Add Custom Event...
Event Dispatchers
Input
Utilities
Add Reroute Node...
An event with customizable name and parameters.
hold (Ctrl + Alt) for more
2 Faster to create an event from that button, it's will automatically add needed variables names and types to save you time :)


Bind Event to On Tap
Target
Event
Touch Manager
CustomEvent_0
Custom Event
After event, simply add print string for test
Print String
In String
OnTapCallBack!
Development Only


Call back works! You can easily bind to other methods and check more in the Touch Manager marketplace project or Youtube video tutorials.

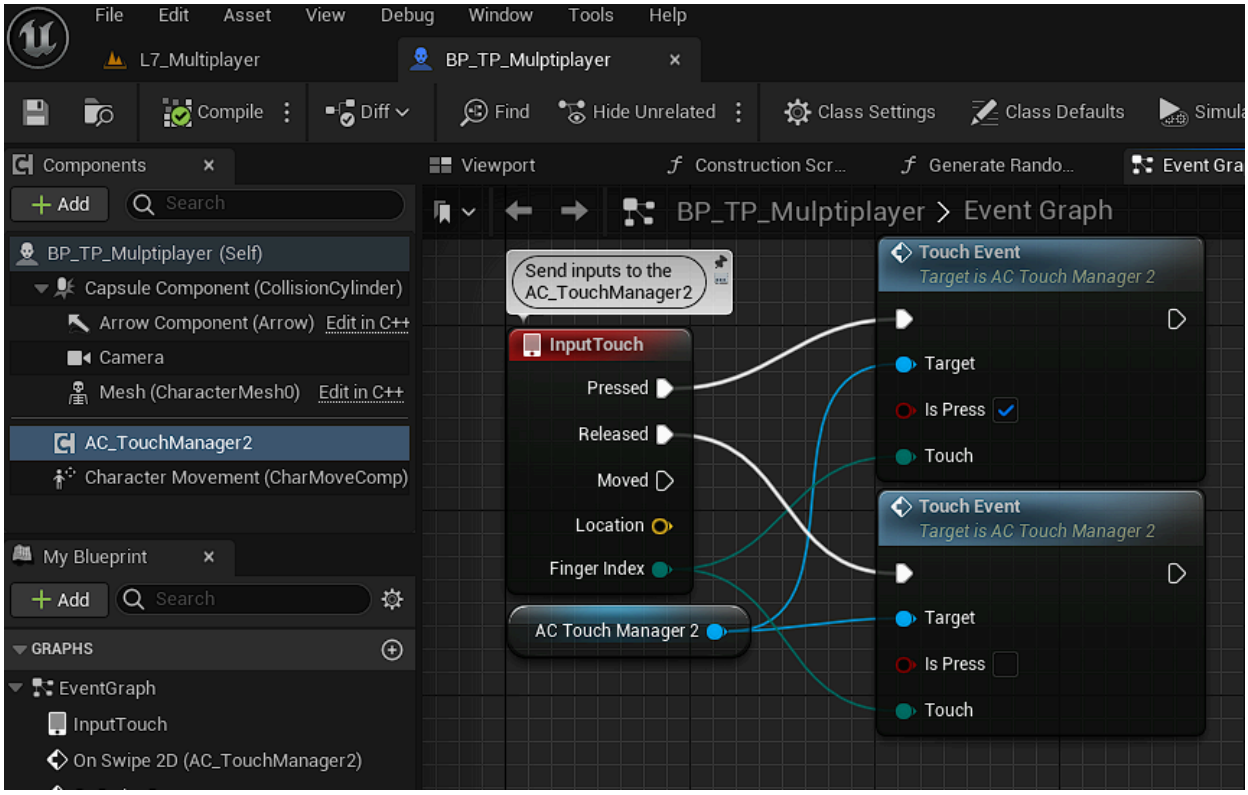
# Technical details, about Touch Manager

**Touch Manager** it's a blueprint Actor class with Touch inputs and updates from Other Actors Components methods (DragComponent, PinchComponent, RotateComponent, SwipeComponent, TapComponent).

- **Pinch/Rotate** enum variable to change Pinch and Rotate setup. By default Pinch And Rotate is selected, which means we can do Pinch and Rotate. If only Rotate or only Pinch that will work only for the single selected one. That can be helpful if you need sometimes block Rotate or Pinch, without Unbinding and Binding again.

# Touch Manager update 2

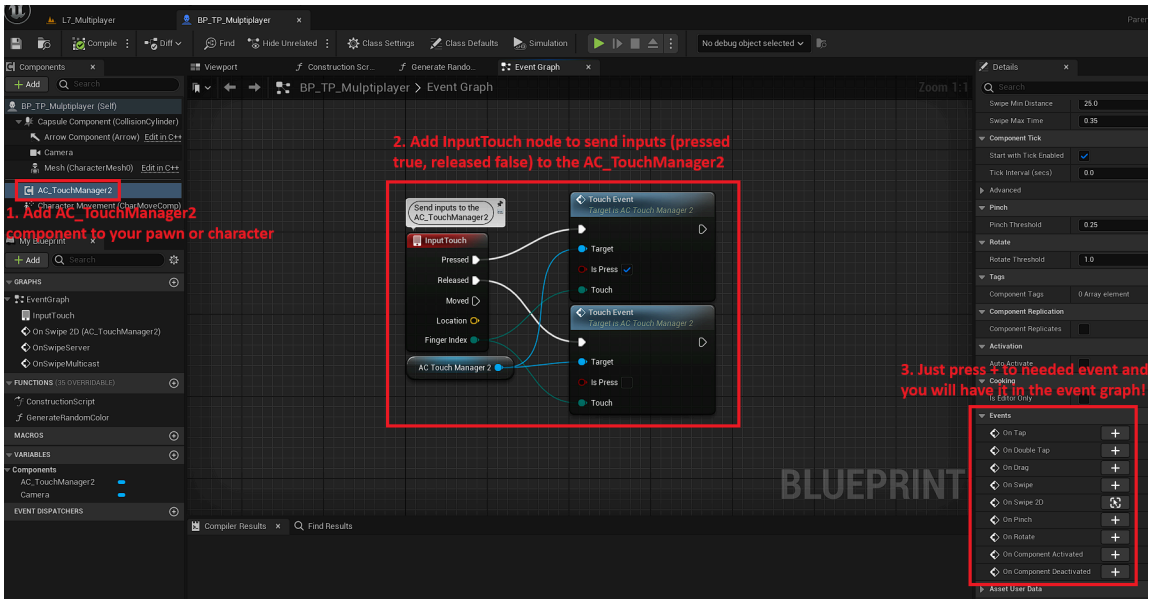
In the new update added multiplayer support only in **L7_Multiplayer.** Now you can add new component **AC_TouchManager2** to your pawn, and send inputs to work with events in one component and multiplayer image:




In the **AC_TouchManager2** you have the same methods and parameters like in the **BP_TouchManagerActor.** If you need to have it all in one place, possibly multiplayer (RPC) support use a new **AC_TouchManager2** setup. **Video:** [https://youtu.be/Osjb9holWso](https://youtu.be/Osjb9holWso)

## How to use new AC_TouchManager2:

1. Add **AC_TouchManager2** component to your pawn.
2. Add InputTouch event and connect to TouchEvent on press (IsPress true) and release (IsPress false) to handle inputs from touches.
3. Select needed events by pressing + and you will have it in the event graph.




Old BP_TouchManagerActor is better in single-player and when you need to listen from a lot of different AActors your bindings, to not hardcode all only to a character.

# Touch Manager update 2.1

Update 2.1 includes different screen sizes support feature, from 1k to 8k screens, that add more accuracy OnDrag and other trachold values events.

| Resolution | Measurements (In pixels) | Pixel count |
|---|---|---|
| 8k (Ultra HD) | 7,680 × 4,320 | 33,177,600 |
| 4k (Ultra HD) | 3,840 × 2,160 | 8,294,400 |
| 1080p (Full HD) | 1,920 × 1,080 | 2,073,600 |
| 720p (HD) | 1,280 × 720 | 921,600 |
| 480p (SD) | 640 × 480 | 307,000 |

To enable it you need set True **ScreenMultiplierEnabled?** And recommend using the new **AC_TouchManager2** component.

# Debugging

Touch Manager has bool “**ShowDebug**?” variable. On **True** we will have helpful debug information with print springs, On **False** no debug message. Also in the new update, every debug will show a unique color 🙂

## AC_TouchTapComponent

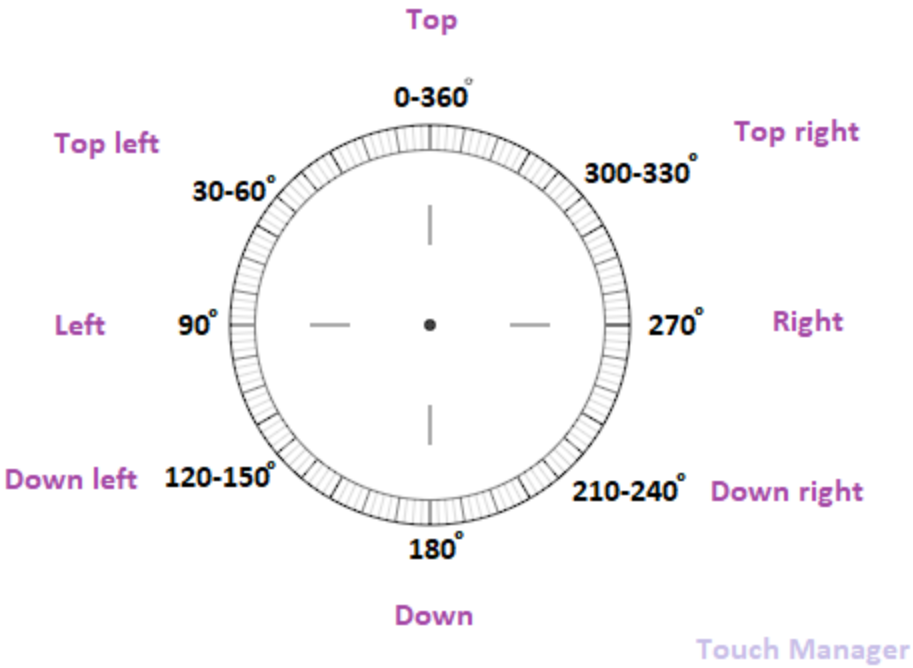
The tap component is used for single finger, tap or double tap.

- **Switch EnumToParameter** enum variable to change Tap and DoubleTap setup. By default Tap and DoubleTap setted, to be able Tap and Double tap events call. If only Tap or only Double that will work only for the single selected one.
- **DoubleTapDelay** float variable to have a limit for double tap call after first touch begins, default value 0.3. If the time is lesser you will have a DoubleTap Event call in the dispatcher.
- **DoubleTapDistance** float variable to set the maximum distance between the first tap and second tap, if the distance will be less than that, a double tap will be called. That is a threshold value for the maximum distance between the 1 and 2 touches, default value 20.

## AC_TouchSwipeComponent

The swipe component is used for single finger, to make fast swipes in different directions.

- **MinSwipeDistance** float variable to set a minimum swipe length. Default value 25 and if the swipe will be bigger than that value, we should make a swipe (and a time threshold from SwipeMaxTime).
- **SwipeMaxTime** float variable to set a maximum swiping time. Default value 0.35, which means if we swiping more time than that value Swipe events will not call.
- **CallOnSwipeEnum()** Function with math calculation to make 8 different directions of swipes, shown on the image:




To change angles you can modify **CallOnSwipeEnum()** function, video: https://youtu.be/JYbDf7dbxOA

## AC_TouchRotateComponent

The rotate component is using for two fingers to make the rotation.

- **RotateThreshold** float variable to not call rotation so frequently on the small angle values.
- **AngleBetweenVectors**(FirstVector vector2D, SecondVector vector2D) function to get an angle from touches.

## AC_TouchPinchComponent

The pinch component is using for two fingers to make zoom in and zoom out.

## AC_TouchDragComponent

The drag component is used for a single finger to move objects with a finger.

# How to make your own color Touch Interface

Helpful video on how to change joysticks color in Affinity Photo 2 to Unreal mobile:
https://youtu.be/yIXlyila3ME?si=ZU66wZIMIj93WRnw

# Support

If you need support or have questions please write to the support email: UnrealOwlHelp@gmail.com