PicoVRAndroidSDK_UE4

Development Document

Version：v_1.1.6

Beijing Pico Technology Co., Ltd

Directory

1 SDK introduction

This document describes the use of PicoVRAndroid SDK_UE4 (hereinafter referred to as SDK) to make VR/AR applications running on Pico integrated machine equipment under the environment of Unreal game development engine. SDK mainly provides: binocular stereo rendering, optical distortion correction, sensor fusion, asynchronous time distortion, single buffer rendering, multi-interactive support (head-hand 3D OF/6DOF), multiple peripheral support, power and heat management, account number and display. Payment management and other functions.

The SDK is provided in the form of an engine plug-in that implements UE4's VR abstraction layer bridge engine and Pico virtual reality hardware device.

This causes the position/orientation of the Camera component in the engine to follow the Pico virtual reality helmet movement, so that the engine's MotionController component will follow the Pico motion controller motion.

Certainly, the engine API can be used to control the Pico VR hardware device. For example, the Input/Head Mounted Display/Reset Orientation and Position can be used to implement the gesture/position reset function of the HMD. For some Pico VR-specific features, we also provide blueprint interfaces, with call methods attached.

2 Supported Devices

2.1 VR all-in-one Devices

Manufacturer	Product
Pico VR	Pico Goblin、Pico G2、Pico Neo

The SDK does not support normal Android phones, but only the VR all-in-one devices listed above.

3 Developing Environment

Software	Version Requirement
Unreal Engine	4.18.3、4.19.2、4.20.3、4.21.2
Visual Studio	2015 or later (for UE4.20+, Please ensure that the latest version of VS2017 has been installed)
JDK	jdk1.7.0_01 and later
Android Works	The latest version is recommended

Installation steps of Android Works please reference:

https://docs.unrealengine.com/en-US/Platforms/Mobile/Android/InstallingAndroidCodeWorksAndroid。

When installing Visual Studio 2015, the box “Common Tools for Visual C++ 2015” must be checked, otherwise the project cannot be compiled.

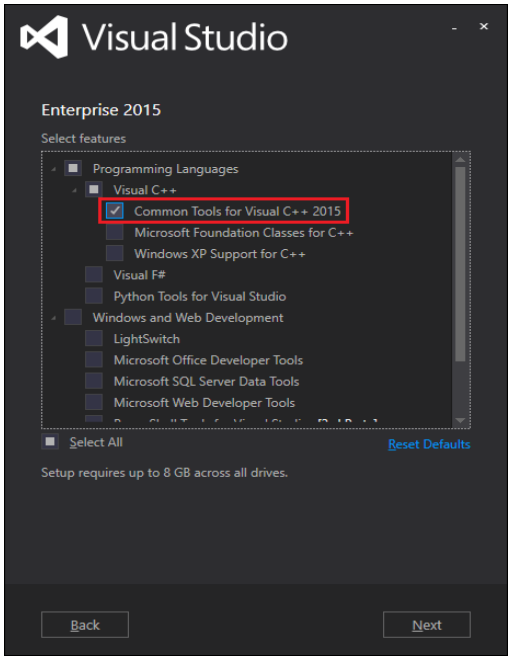


Figure 3.1 Visual Studio 2015 install features

In addition，when installing Visual Studio 2017,please check“Game development with C++”：

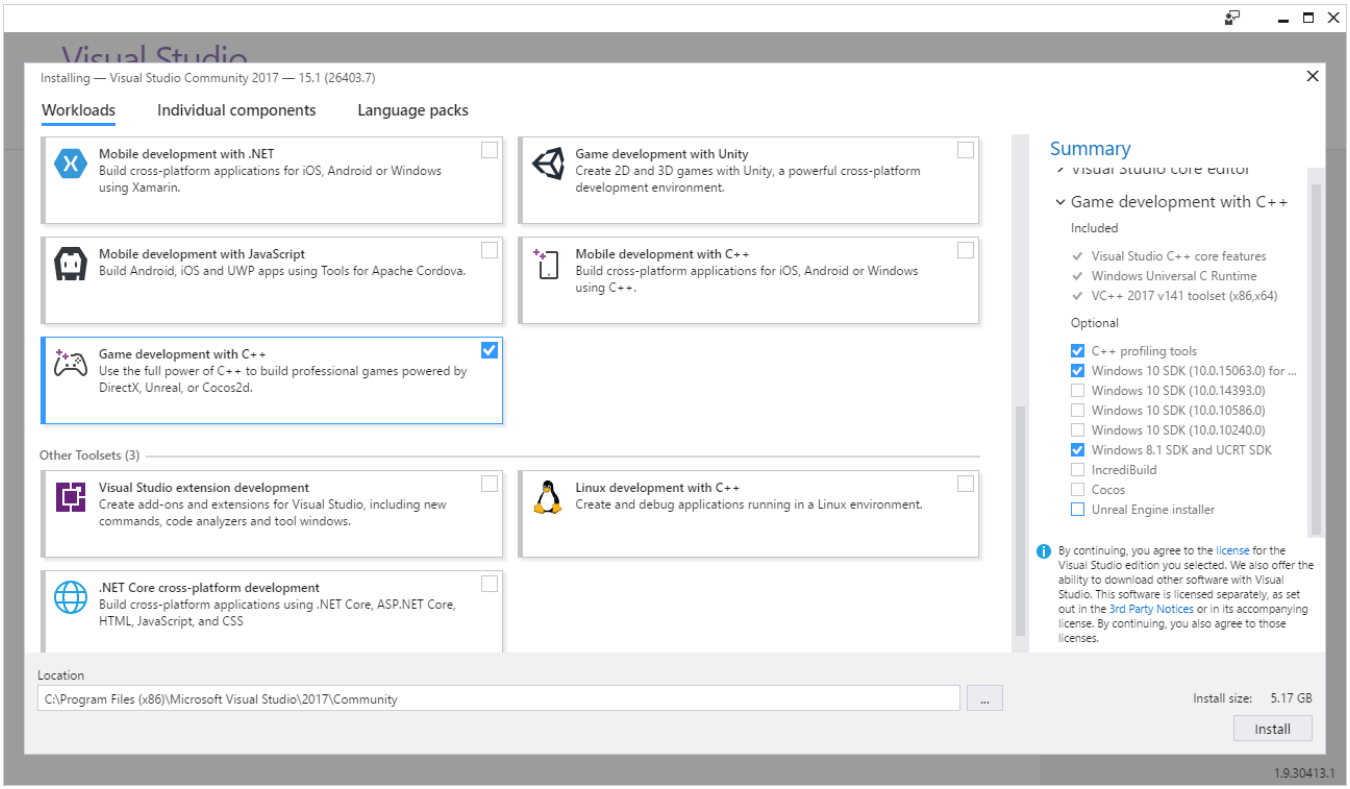


Figure 3.2 Visual Studio 2017 option of install

4 Quick Start

4.1 Project Creation

Our SDK supports both blueprint projects and C ++ projects. Here we take blueprint project of blank as an example.

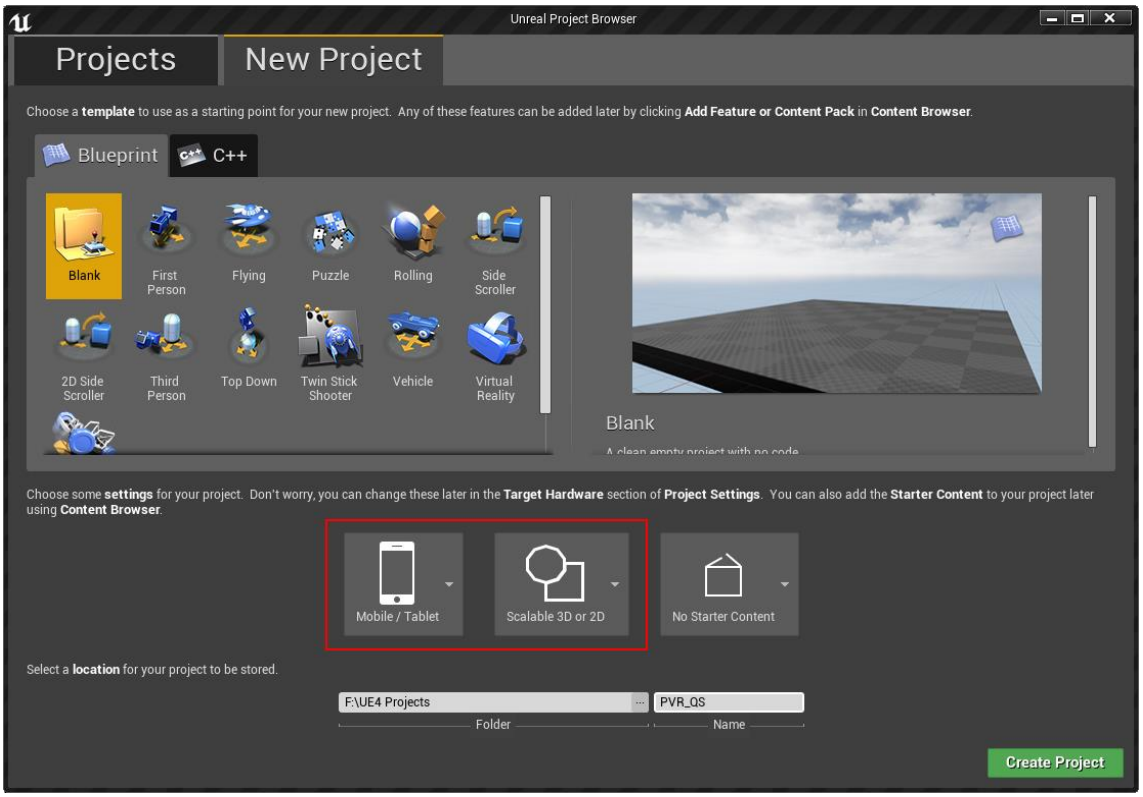


Figure 4.1 Project Creation

The target hardware should be Mobile / Tablet and the target graphic level should be Scalable 3D or 2D.

4.2 Project Settings

You need to set the project to compatible with our SDK. The steps are as follows:

1、Save the current map and open Edit-> Project Setting-> Project-> Maps and Modes to set it as the project Editor Start Map and Game Default Map.

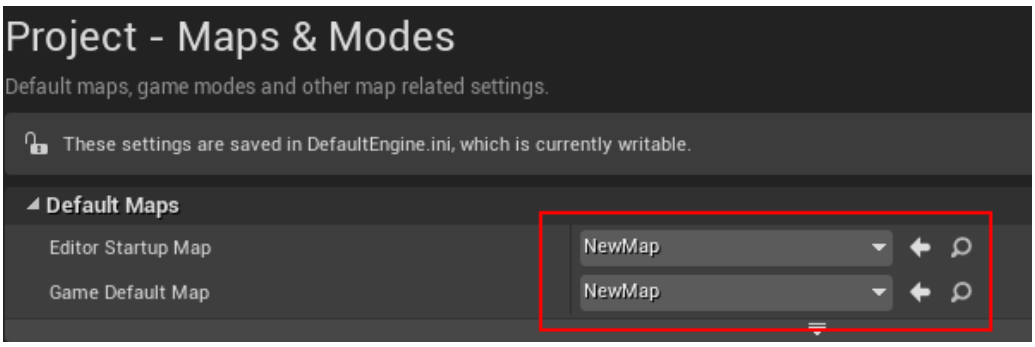


Figure 4.2 Set default maps

2、Open Engine ->Input->Mobile to clear Default Touch Interface:

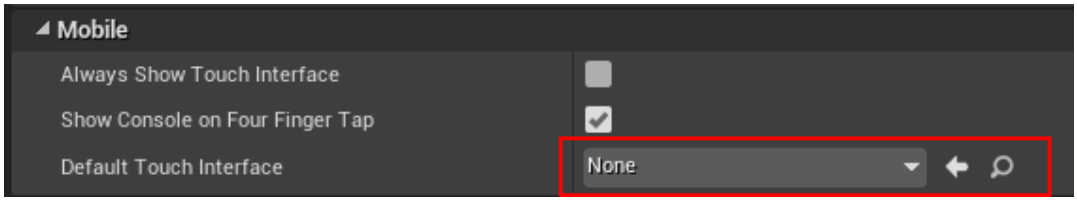


Figure 4.3 Clear Default Touch Interface

Then Open Platforms->Android->APKPackaging to check Enable FullScreen Immersive on KitKat and above devices:

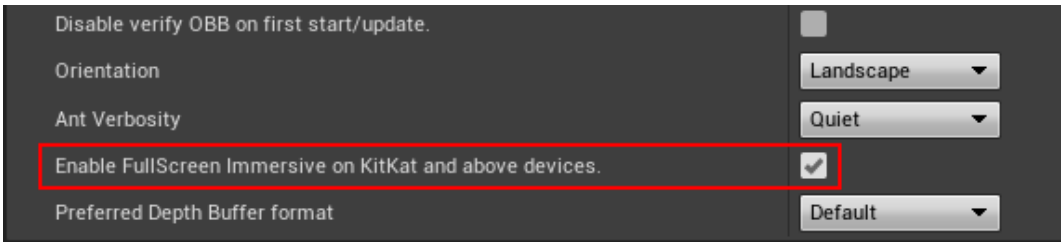


Figure 4.4 check Enable FullScreen Immersive on KitKat and above devices

3、Enter Platforms->Android to set both Minimum SDK Version and Target SDK Version to 19:

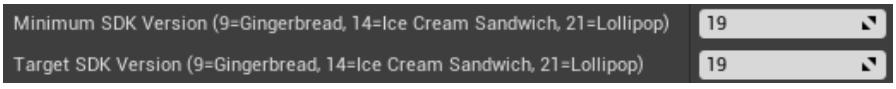


Figure 4.5 Select SDK Version

Then Enter Platforms->Android SDK to set SDK API Level as matchndk and ensure NDK API Level as android-19:

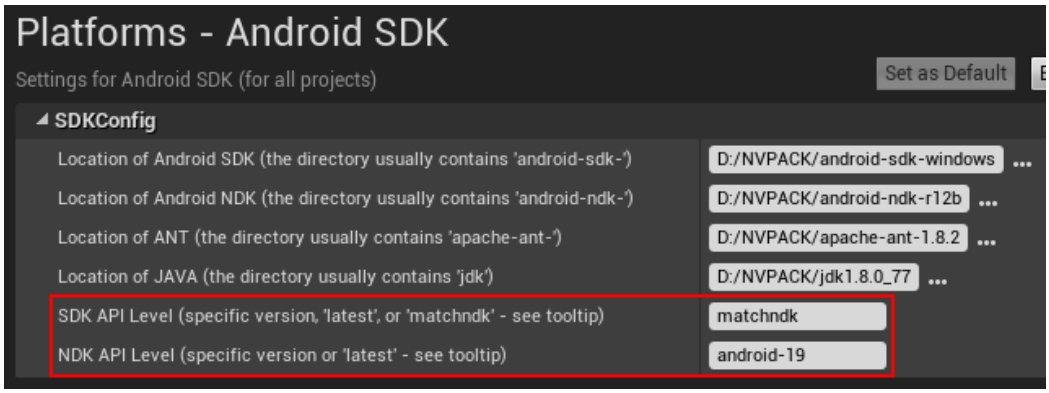


Figure 4.6 Select NDK Level

4、Enter Platforms->Android->Build to ensure the items of Support armv7 has checked:

Figure 4.7 CPU

5、Open the project and enter UE editor, go to Menu/Edit/Plugins, uncheck all the options under Built-in/Virtual Reality label. At this point the editor will prompt to restart. **You should close the editor Instead of restart it.**

4.3 Import SDK

Be sure to keep the editor closed, and then copy the Plugins directory to the project's root directory:

Figure 4.8 Import the SDK into the directory

Then double-click on the project name to reopen it.

If you use UE4.20+, it will prompt the SDK module is not compiled, click "Yes (Y)" to continue:

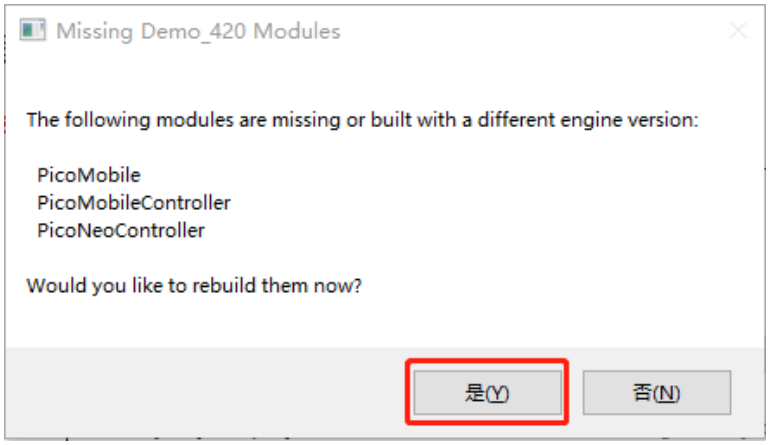


figure4.9 Prompt not compiled

Under the Plugins page, you can see our plugin:

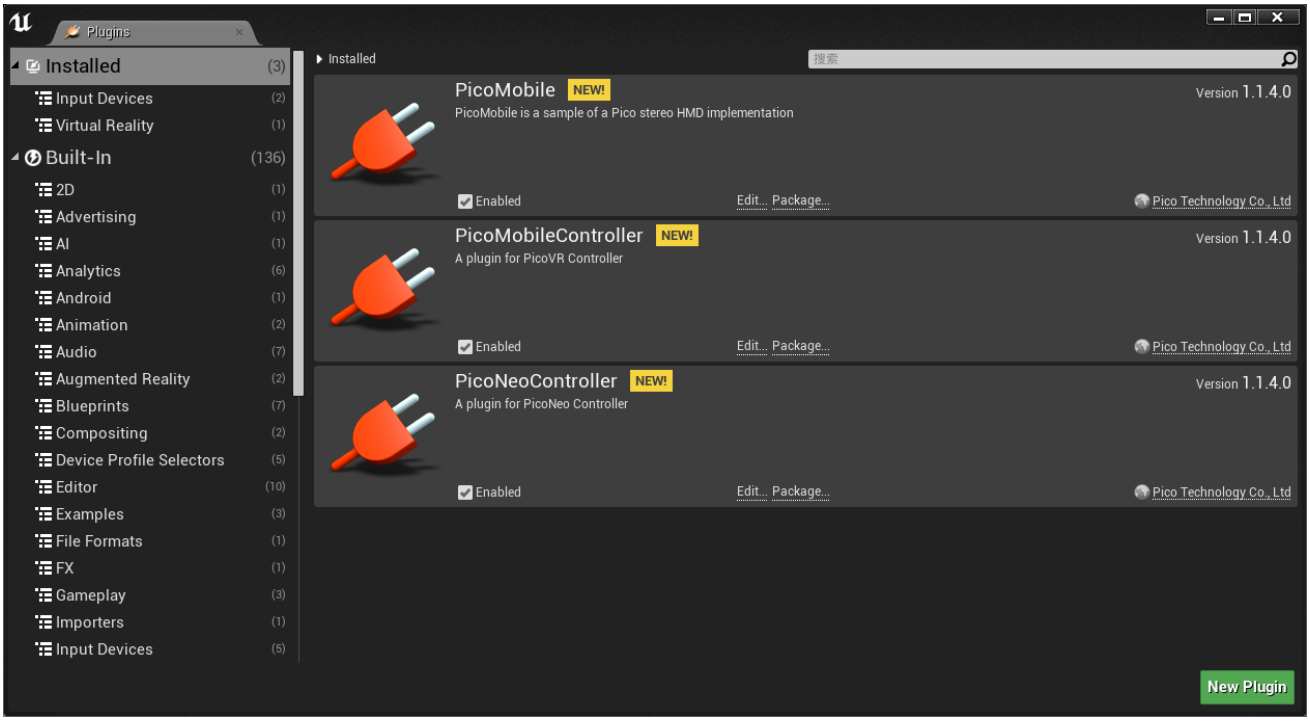


Figure 4.10 Pico Plugins

4.4 Project Completion

1、Create a Pawn blueprint class and add a Scene component to it. Then add a Camera component to

Scene component.

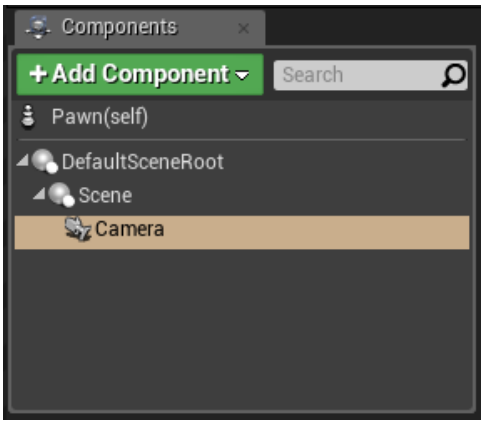


Figure 4.11 Hierarchy of pawn

After being packaged and mounted to the helmet, the relative position/pose of the Camera will be refreshed in real time according to the helmet, completing head tracking and stereo rendering.

2、Drag the Pawn into the scene and set its Auto Possess Player to Player0:

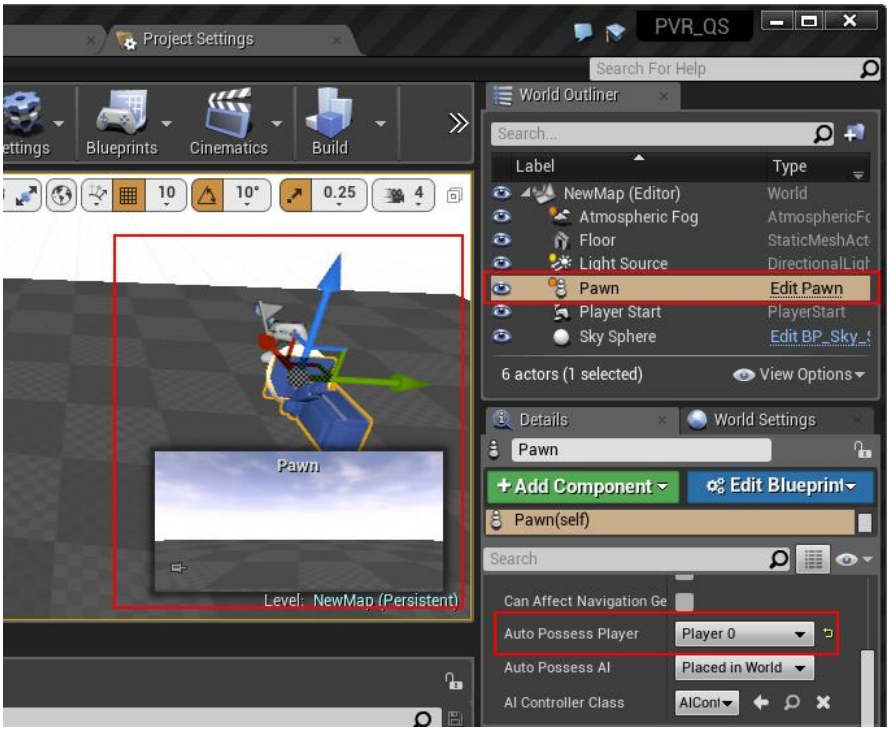


Figure 4.12 Drag Pawn to level

4.5 Package Project

The texture compression format Pico Neo、Pico Goblin support is ASTC, so please select Android(ASTC) when packaging project. The specific packaging process is: in editor, File->Package Project->Android-> Android(ASTC) (It is recommended to enter the "Project Settings" first, check the "Package game data inside .apk?" in the Platforms sub items Android, in order to package the data into apk) .

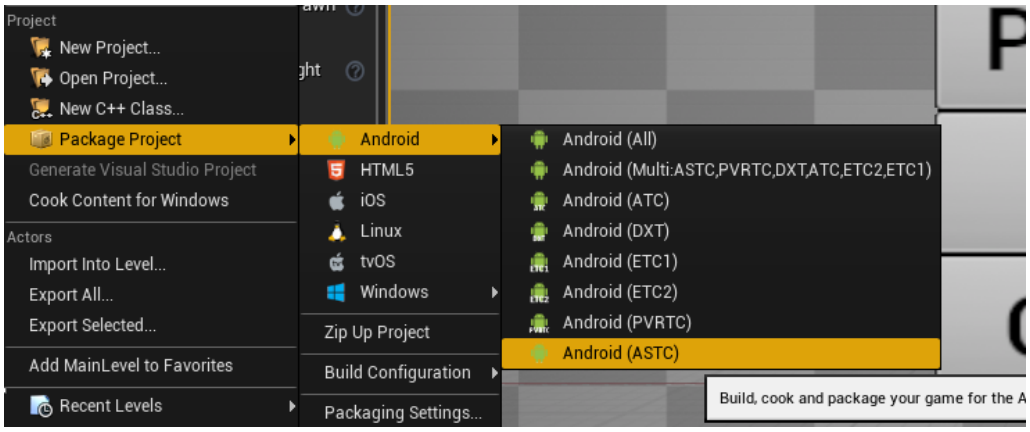


Figure 4.13 Package Project

In addition, please uncheck before "Enable Gradle instead of Ant" before packing.

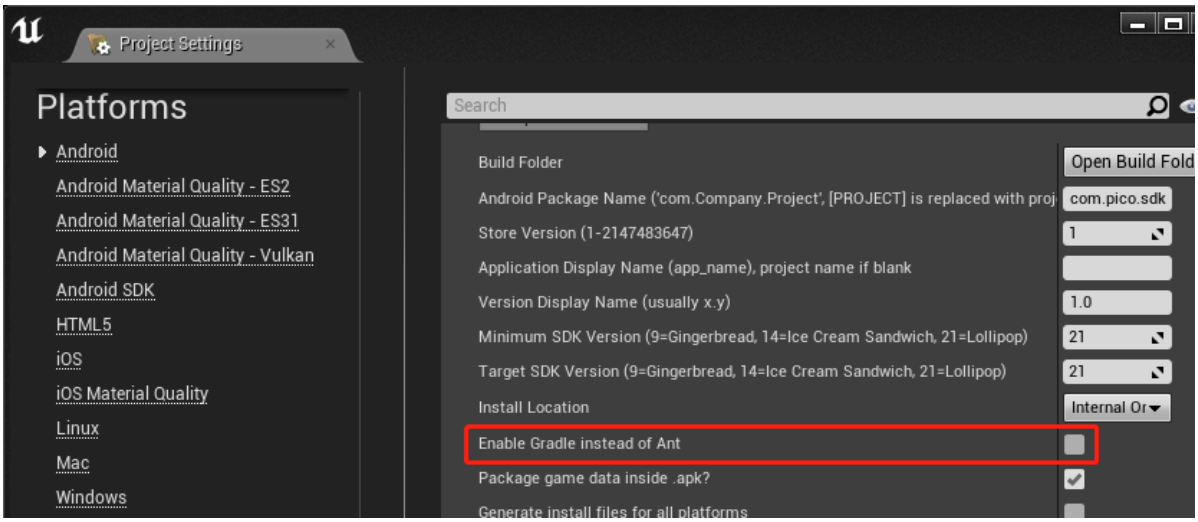


Figure 4.14 Uncheck "Enable Gradle instead of Ant"

After packaging, double-click the "Install_project name_compiler configuration -armv7-es2.bat for

installation":

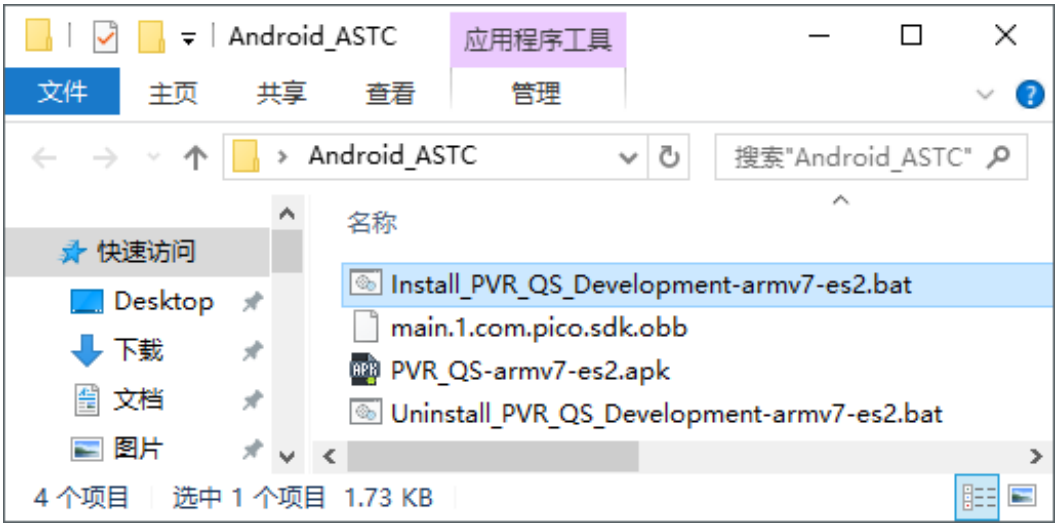


Figure 4.15 Install apk

5 Guide to use Pico controllers

5.1 Pico Goblin Controller

5.1.1 Instruction

Figure 5.1 Pico Goblin Controller

To use Pico Goblin controller，please follow the steps below：

1、Add MotionController component to the default Pawn in your project, in the same level as the Camera.

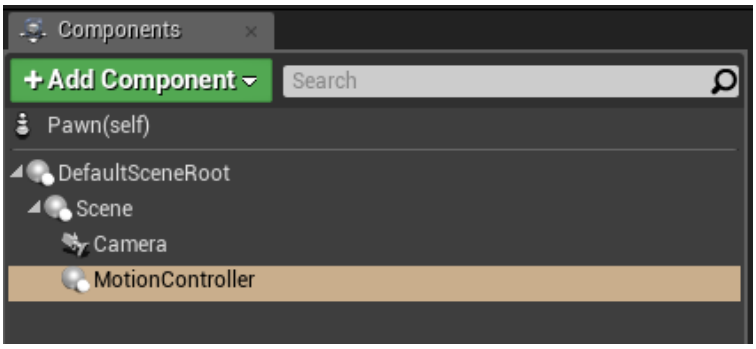


Figure 5.2 Add MotionController component

This component is not affected by the Hand property. For example, if you change the left and right hands, please go to the system settings to modify it. And please make sure to enable “Disable Low Latency Update” option, so that the controller model won’t twinkle as the result of dual update per

frame.

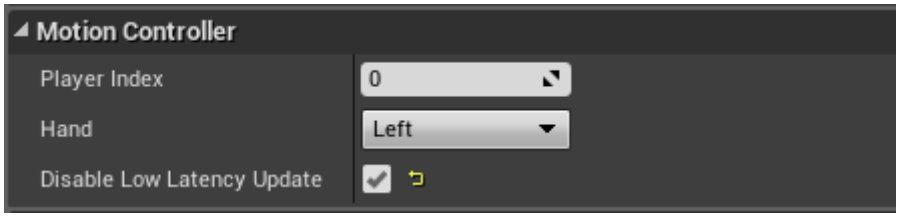


Figure 5.3 Check Disable Low Latency Update

After packaging and installation, the component will follow the position/orientation movement of the Pico Goblin handle.

2、Add model for MotionController:

For UE4.18，please add StaticMesh component under MotionController.

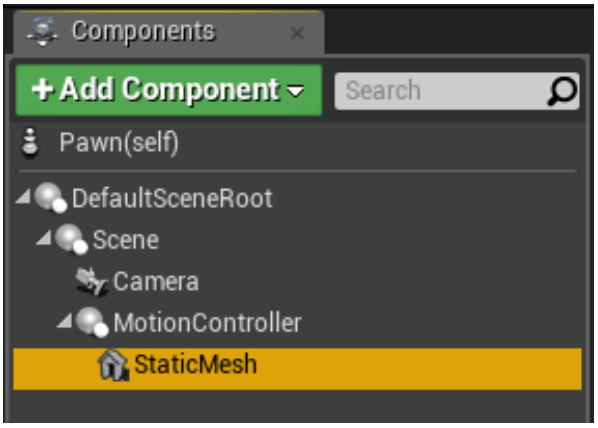


Figure5.4 Add StaticMesh component

In StaticMesh property panel, please select “Show Plugin Content” and then choose ppcontroller.

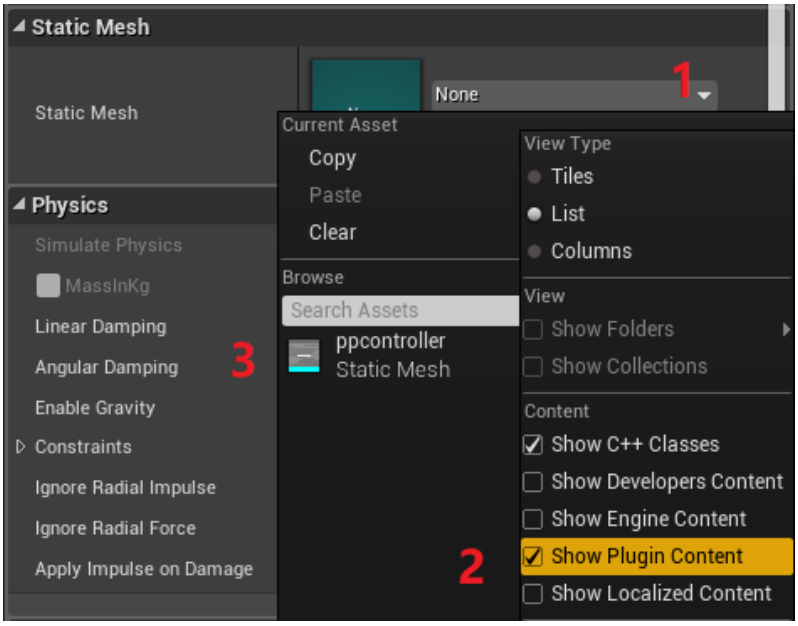


Figure 5.5 Add Pico Goblin controller model

For UE 4.19 and higher, add the model under the Visualization sub-menu of the MotionController details panel (also check "Show Plugin Content" to display):

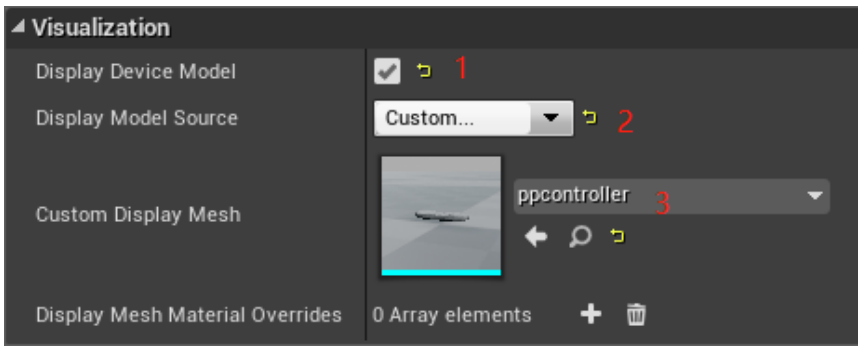


Figure 5.6 UE4.19+ adds Pico Goblin controller model

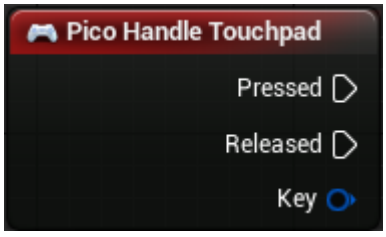
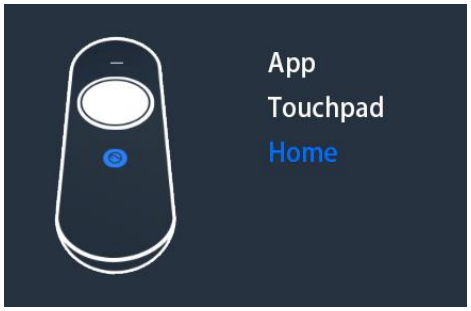
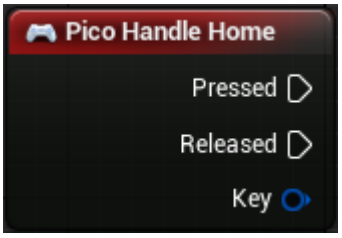
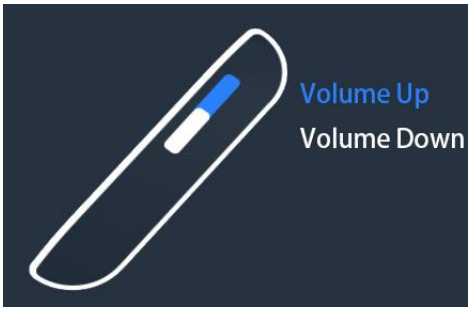
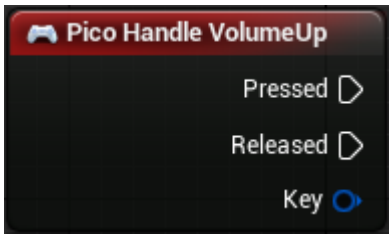
It should be noted that in our plugin directory, we package the handle with button animation into an Actor. If you want to reuse it, attach it to the Pawn or Character in your level.

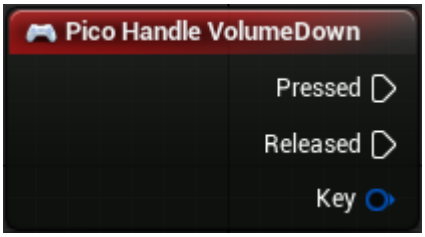
5.1.2 Input instruction

5.1.2.1 Buttons

The buttons used by the Goblin controller are not predefined by the engine. When developing,

please set the input binding according to the following table, or drive the event through the corresponding blueprint node:

Goblin Controller buttons	Input Binds	Blueprint Node
		
		
		
		

Goblin Controller buttons	Input Binds	Blueprint Node
		

5.1.2.2 Touchpad Axis

Please see the picture below:

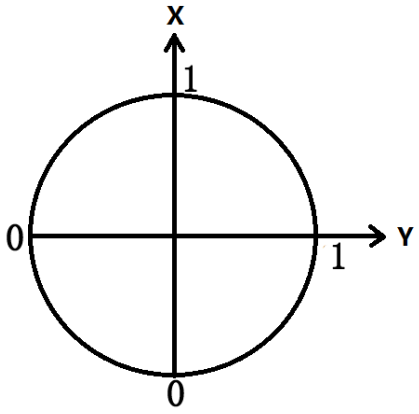


Figure 5.7 Goblin Controller touchpad axis

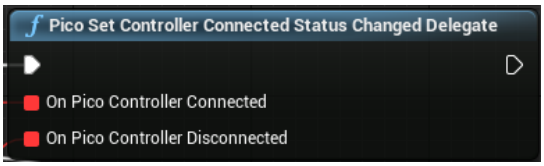
Please notice that the value of X and Y axis are range from 0 to 1, different from the touchpad of Steam, Oculus and Google. If there are functions related to touchpad axis in your project, please remember to map the value [0,1] to [-1,1].

Use the nodes below to get the axis value:

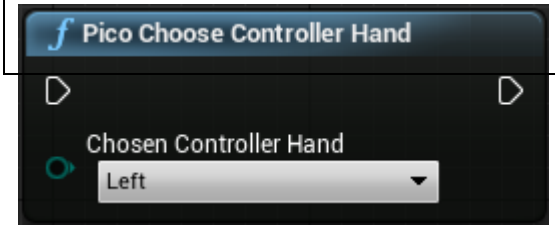
Figure 5.8 How to get the touchpad axis value

5.1.3 Blueprint nodes

There are some other blueprint interfaces of the Goblin controller:

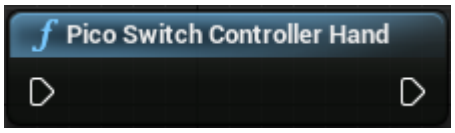


Description	Set the operation when connection of motion controller disconnect and reconnect the operation you want do
Input	OnPicoControllerConnected : The delegate executed when the handle is reconnected OnPicoControllerDisconnected : Commissioned when the handle connection is accidentally disconnected
Output	None
Return	None

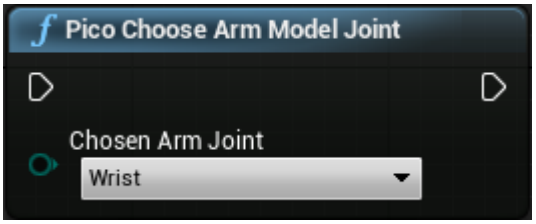


Description	Select a suitable hand for motion controller

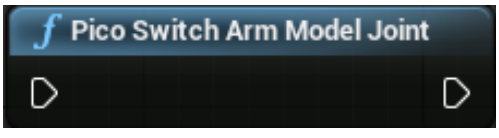
Input	Left——left hand, Right——right hand
Output	None
Return	None



Description	Switch the suitable hand for motion controller
Input	None
Output	None
Return	None



Description	Select the current tracking joint for motion controller
Input	Wrist，Elbow，Shoulder
Output	None
Return	None



Description	Cycle through the tracking joints of the current motion controller, in order of wrists - elbows - shoulders

	Input	None
	Output	None
	Return	None

5.2 Pico G2 Controller

Figure 5.9 Pico G2 Controller

The Pico G2 controller buttons' corresponding relationship is as follows.:

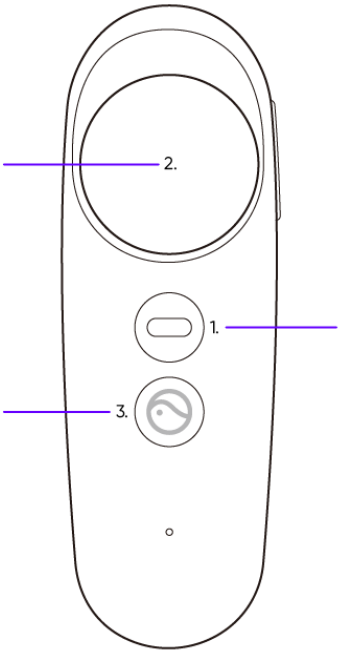


Figure 5.10 Pico G2 Controller Button Correspondence Diagram 1

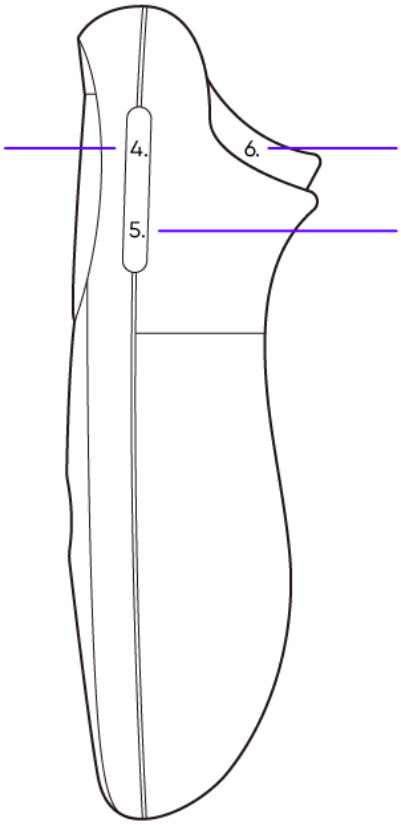
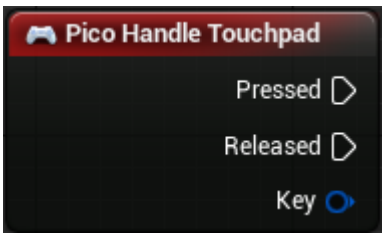
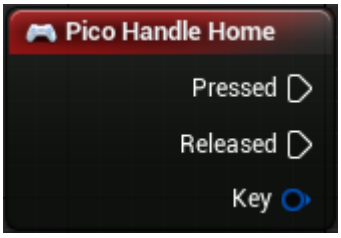
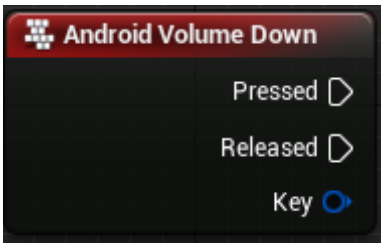


Figure 5.11 Pico G2 Controller Button Correspondence Diagram 2

G2 Controller buttons	Input Binds	Blueprint Node
1、APP		
2、Touchpad		

G2 Controller buttons	Input Binds	Blueprint Node
3、Home		
4、Volume Up		
5、Volume Down		
6、Trigger		

5.3 Pico Neo Controller

5.3.1 Instruction

Figure 5.12 Pico Neo controller

Regarding Pico Neo controller, there is a very important concept: main controller and sub controller. The main controller is one with the ray and can interact with UI, and the sub controller is the other one. When there is only one controller connect to the HMD, it must be the main controller.

It is suggested to add WidgetInteraction component on the main controller, to make sure using the same controller to interact with UI in the system and in your own application.

5.3.2 Instruction

To use Pico Neo controller, please refer to the following steps:

Add two MotionController components, which should be in the same level as Camera, to the default Pawn in your project and rename them to MotionController_Main and MotionController_Sub.

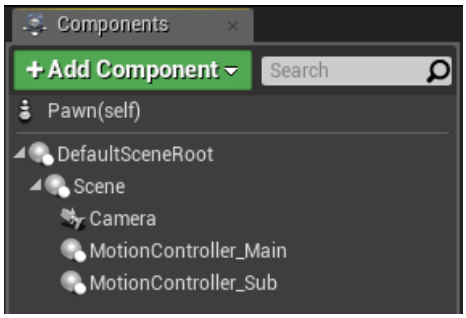


Figure 5.13 Add MotionController components

Select MotionController_Main, set the value of Hand to "Special 1" in its property panel. So the component will follow the main handle movement. With regard to MotionController_Sub, the value of Hand should be "Special 2".

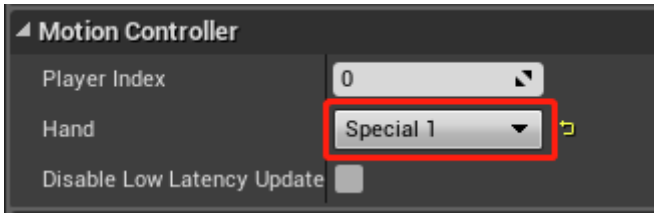


Figure 5.14 Set Hand property

4、Add StaticMesh component to MotionController:

For UE 4.18, first add the StaticMesh component to the MotionController component:

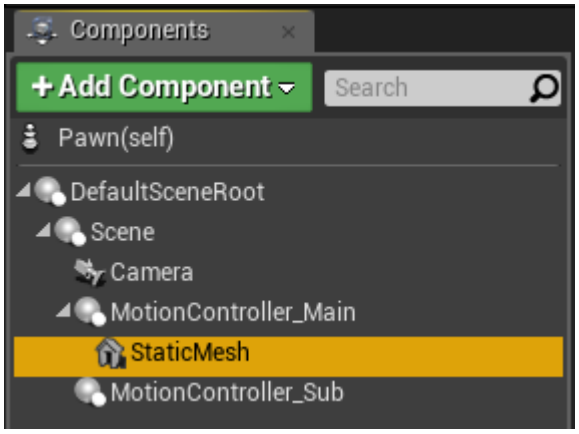


Figure 5.15 Add Static Mesh component

In StaticMeshcomponent panel, enable "Show Plugin Content" option, and then select Mesh_cvcontroller:

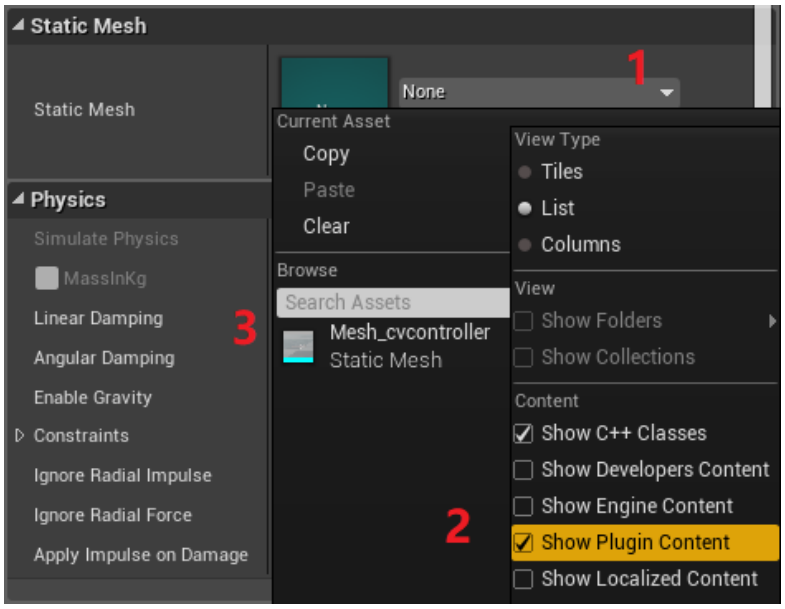


Figure 5.16 Add Pico Goblin Controller model

For UE 4.19 and higher, please add the model under the Visualization sub-menu of the MotionController details panel (also check "Show Plugin Content" to display):

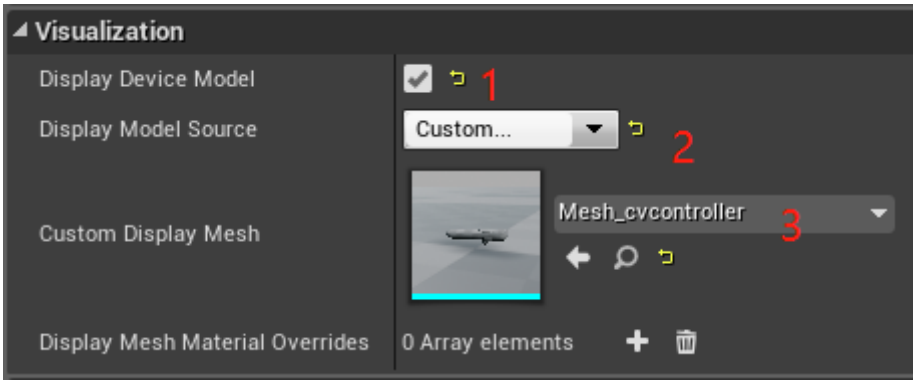


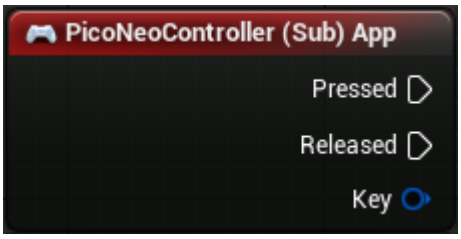
Figure 5.17 UE4.19+ Add Controller Model

It should be noted that in our plugin directory, we package the handle with button animation into an Actor. If you want to reuse it, attach it to the Pawn or Character in your level.

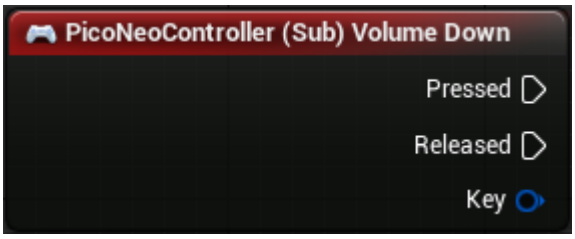
5.3.3 Input instruction

5.3.3.1 Buttons

The buttons used by the Goblin controller are not predefined by the engine. When developing, please set the input binding according to the following table, or drive the event through the corresponding blueprint node:

buttons	Input interfaces		
	Main		
	Sub		

buttons	Input interfaces		
	Main	PicoNeoController (Main) Touchpad	PicoNeoController (Main) Touchpad Pressed Released Key
	Sub	PicoNeoController (Sub) Touchpad	PicoNeoController (Sub) Touchpad Pressed Released Key
	Main	PicoNeoController (Main) Home	PicoNeoController (Main) Home Pressed Released Key
	Sub	PicoNeoController (Sub) Home	PicoNeoController (Sub) Home Pressed Released Key
	Main	PicoNeoController (Main) Volume Up	PicoNeoController (Main) Volume Up Pressed Released Key
	Sub	PicoNeoController (Sub) Volume Up	PicoNeoController (Sub) Volume Up Pressed Released Key

buttons	Input interfaces		
	Main		
	Sub		

5.3.3.2 Touchpad Axis

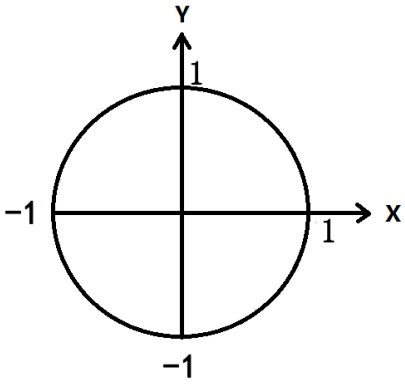


Figure 5.18 Pico Neo controller touchpad axis

Using the below blueprint nodes to achieve the value of the touchpad of the main and sub controllers:

Figure 5.19 Touchpad blueprint nodes

The value of the trigger range from 0 to 1, you can get the value with the following interfaces:

Figure 5.20 Trigger blueprint nodes

5.3.4 Blueprint nodes

Right click on the Pico Neo/Controller in the menu, you can see the APIs of the Pico Neo controller:

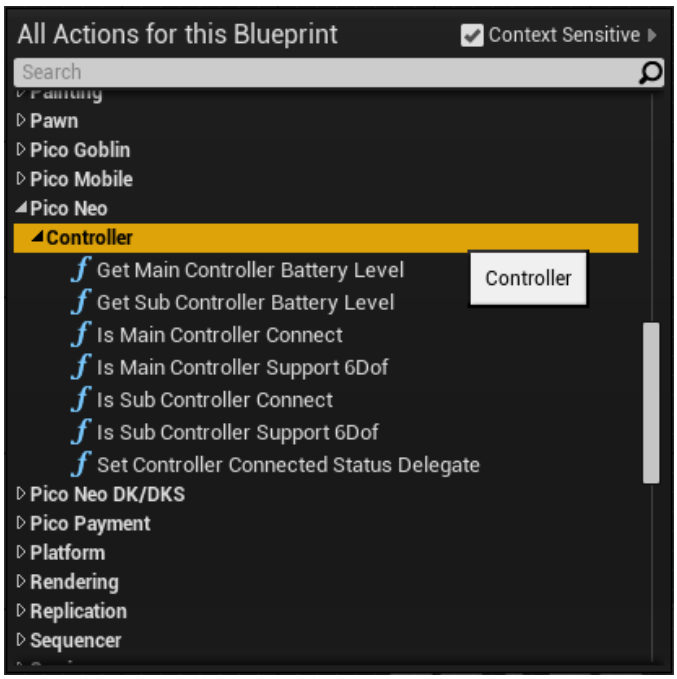
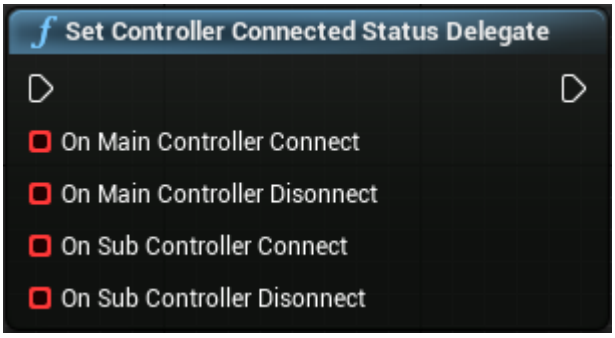


Figure 5.21 Pico Neo controller API

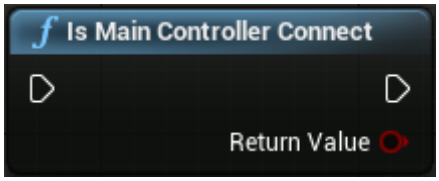
Please see the table below for more detail:

	Description	Set the callback event after the controller connection status changed
	Input	OnMainControllerConnect: Main controller connected OnMainControllerDisconnect: Main controller disconnected OnSubControllerConnect : Sub controller connected OnMainControllerDisconnect : Sub controller disconnected

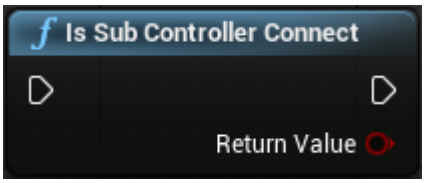
Output	None
Return	None

Description	Get main controller battery level
Input	None
Output	None
Return	main controller battery level，1~10

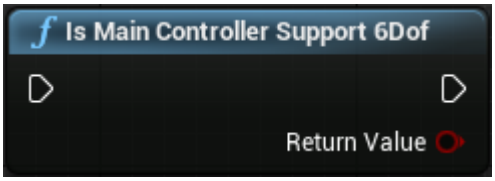
Description	Get sub controller battery level
Input	None
Output	None
Return	sub controller battery level，1~10



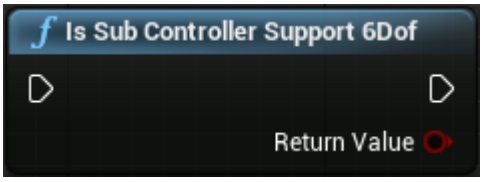
Description	Get the main controller connect state
Input	None
Output	None
Return	true-connected，false-disconnected



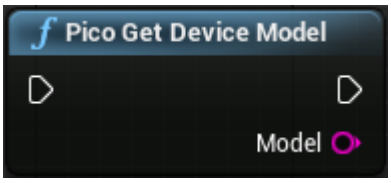
Description	Get the sub controller connect state
Input	None
Output	None
Return	true-connected，false-disconnected



Description	Judge whether the main controller support 6DoF
Input	None
Output	None
Return	true-Support，false-Not support（with only 3DoFtracking）



Description	Judge whether the sub controller support 6DoF
Input	None
Output	None
Return	true-Support，false-Not support (With only 3DoF tracking)



Description	Get the HMD type
Input	None
Output	HMD type
Return	None

6 API Reference

6.1 General API

The SDK supports VR generic functions in the following red dot-marked:

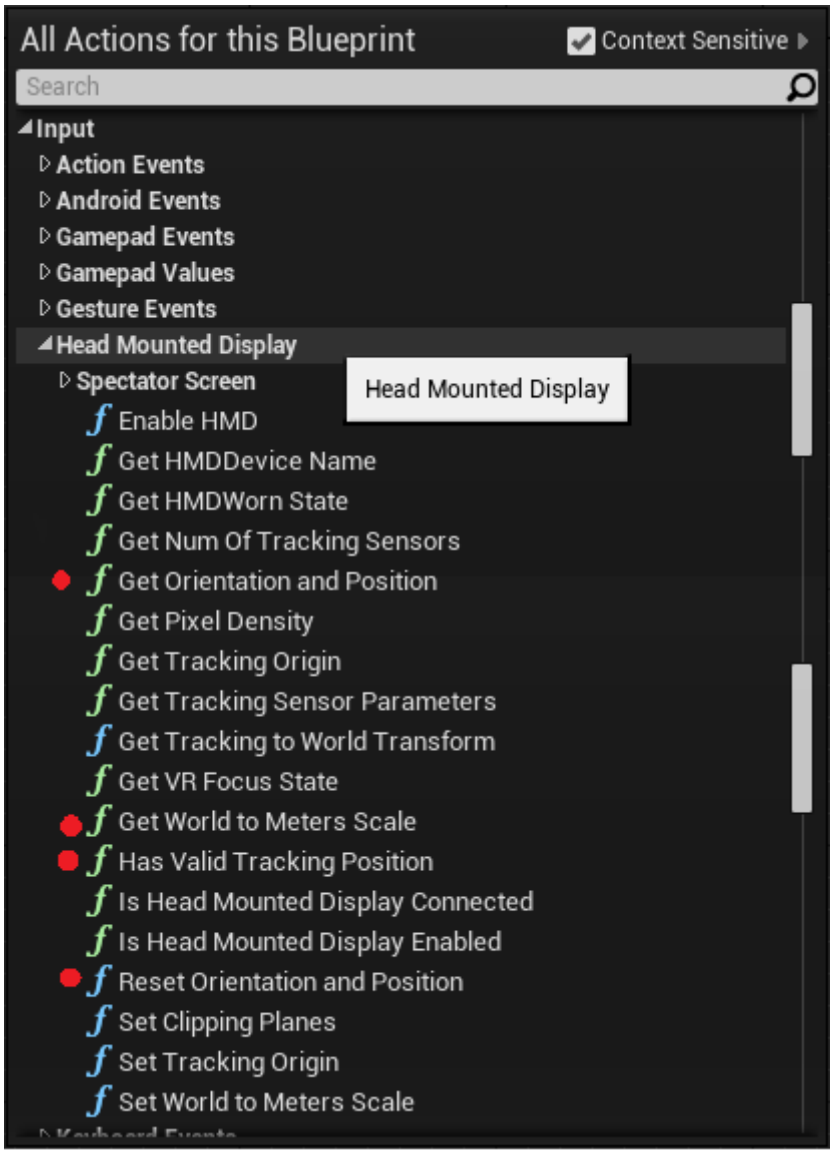


Figure 6.1 General API

For their detailed usage, please refer to the official UE4 document:

https://docs.unrealengine.com/en-us/。

The Reset Orientation and Position node only supports the function of resetting the positive

direction, ignore Yaw parameter, which is invalid.

6.2 Pico API

For system functions such as volume and brightness, the SDK also provides the corresponding API in the form of a blueprint node. Right-click in the event graph to enter the Pico Mobile sub-item, and you can see these APIs:

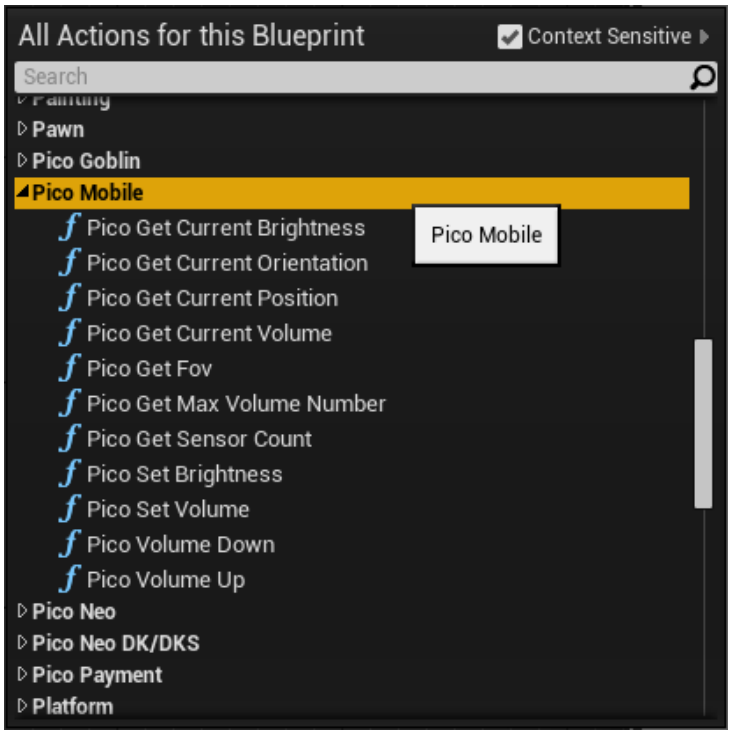
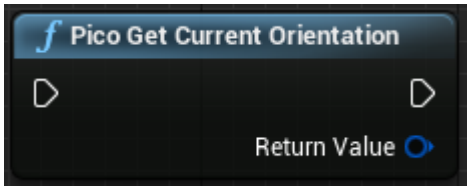
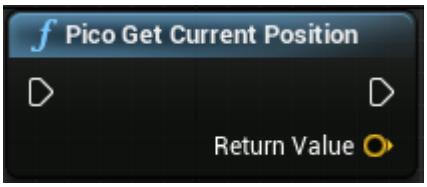


Figure 6.2 System interfaces

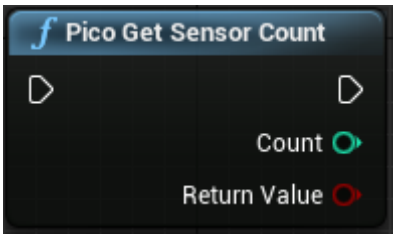
The details of these APIs is as follows:



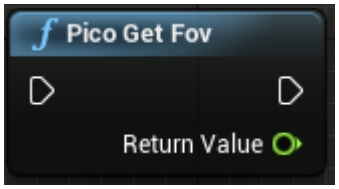
Description	to get the current orientation of HMD
Input	None
Output	None
Return	the current orientation of HMD



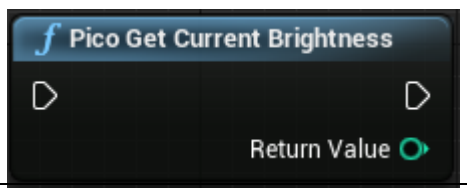
Description	to get the current position of HMD
Input	None
Output	None
Return	the current position of HMD



Description	to get the count of sensor in using
Input	None
Output	the count of sensor in using
Return	whether succeed in getting the value

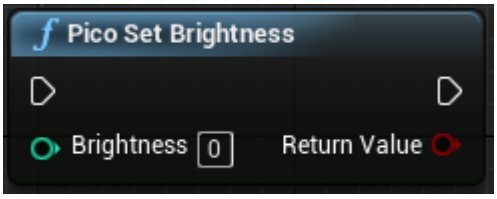


Description	to get FOV
Input	None
Output	None
Return	FOV



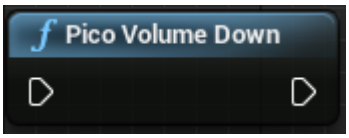
Description	to get the current system brightness
Input	None

Output	None
Return	the current system brightness（integer，0~255）



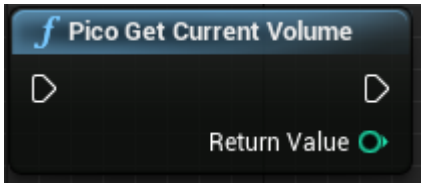
Description	to set system brightness
Input	the desired brightness（integer，0~255）
Output	None
Return	whether succeed in setting the brightness

Description	to increase the system volume （from 0~15，the value increases by 1 after calling）
Input	None
Output	None
Return	None

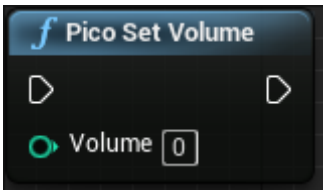


Description	to decrease the system volume （from 0~15，the value decreases by 1 after calling）
Input	None
Output	None

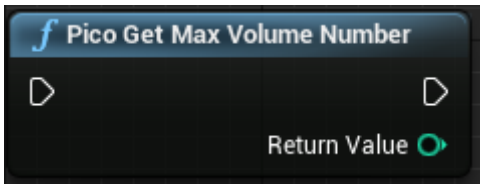
Return	None



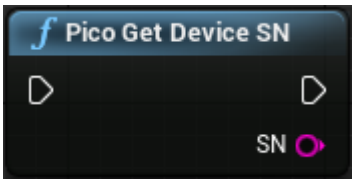
Description	to get the current system volume
Input	None
Output	None
Return	the current system volume



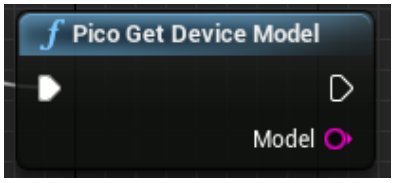
Description	to set the system volume
Input	the desired system value（integer，0~15）
Output	None
Return	None



Description	to get the max volume number
Input	None
Output	None
Return	the max volume number



Description	Get Device Serious Number
Input	None
Output	Serious Number
Return	None



Description	Get Device Model
Input	None
Output	Device Model String (Pico Neo --- Pico Neo) (Pico Goblin--- Pico Goblin) (Pico Goblin 2--- Pico G2)
Return	None

7 Payment System

Pico payment SDK is based on the Pico account system for game currency payment system, the settlement is under the current Pico company game currency unit (P-coin).

7.1 Preparation

7.1.1 Apply and fill in APPKEY，APPID、SCOPE、DEVELOPERID、APP SECRET

When accessing the payment SDK, developers need to create applications and obtain corresponding

strings on the developer platform. Below are the application steps:

- 1.Log on to the developer platform and register as Pico member（http://dev.picovr.com/）
- 2.Apply to become a developer

Developers are divided into individual developers and enterprise developers, please apply according to the actual situation. Upon review, we will provide feedback within 3 working days. Please review the developer platform status in a timely manner.

- 3. View developer ID

After becoming a developer, click on the upper right corner nickname, you can see the developer ID:

Figure 7.1 Developer ID

- 4. Obtain corresponding strings

Developers can enter the creating the application phase from the management center.

Figure 7.1 Create application

After you click create the application, you can choose which platform to publish:

Figure 7.3 Choose platform

After entering the appropriate platform, you can improve the application of relevant information, Including whether the app is free or paid, how much P currency to pay when paying:

Figure 7.4 Improve information

Please pay attention to the red box of the figure above. Please fill in the application type carefully. Once filled, it cannot be modified!

After the application is successfully created, the developer platform assigns the strings. 包括 APPKEY，APPID、APP SECRET：

Figure 7.5 APP ID、APP KEY、APP Secret

For games, if there are items in the case of payment, we require developers to use the developer backend to use commodity code for unified management. Choose in-game payment configuration,

configure the in-game purchase information:

Figure 7.6 in-game purchase information

Note that the rules for commodity codes are defined as' first is the alphabet, and only letters and numbers are allowed, no more than 20 characters. The commodity code between different items cannot be repeated.

Items are divided into expendable items and non-expendable items. Expendable items are those which can be purchased repeatable, such as coins and other items; non-expendable items are one-time purchases of products, such as weapons and unlock game levels.

- 5. Fill in the string

Open Edit->Project Settings... and go to PicoMobile to check “Enable Payment Module”，Then check “Is Foreign” according to the actual situation. Fill in ID、APPID、APP KEY、APP secret:

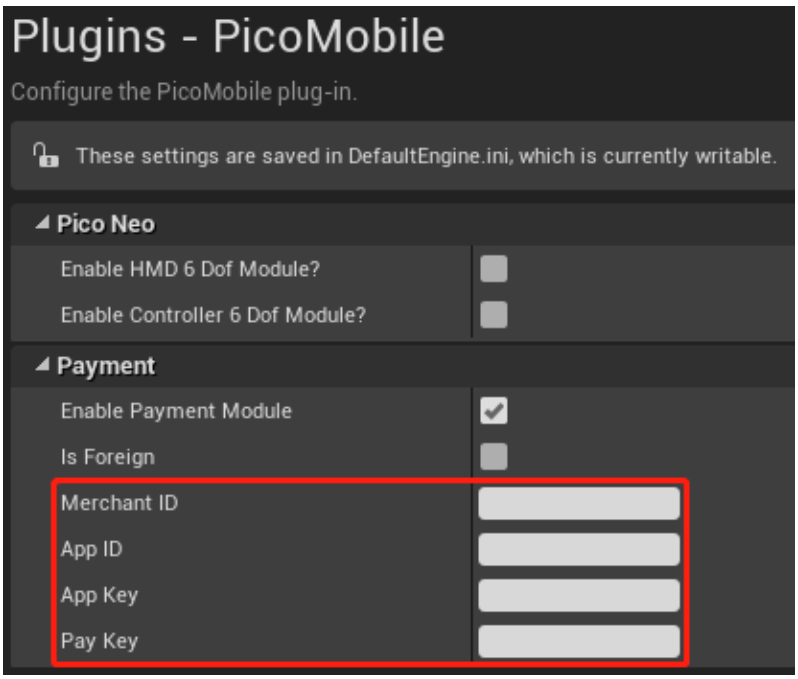
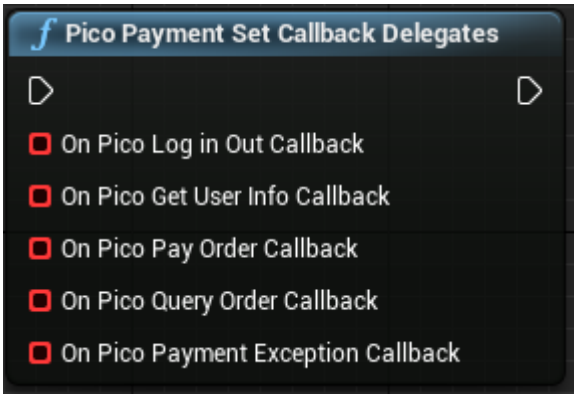


Figure 7.7 fill in string

7.1.2 Setting callback proxy event

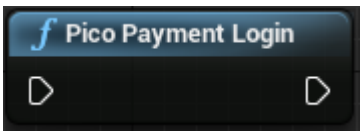
Before you use the payment, you should first set up the callback proxy event so that you can obtain the parameters of the callback function output and set the subsequent execution process. Please use the PicoPaymentSetCallbackDelegates node we provide here:



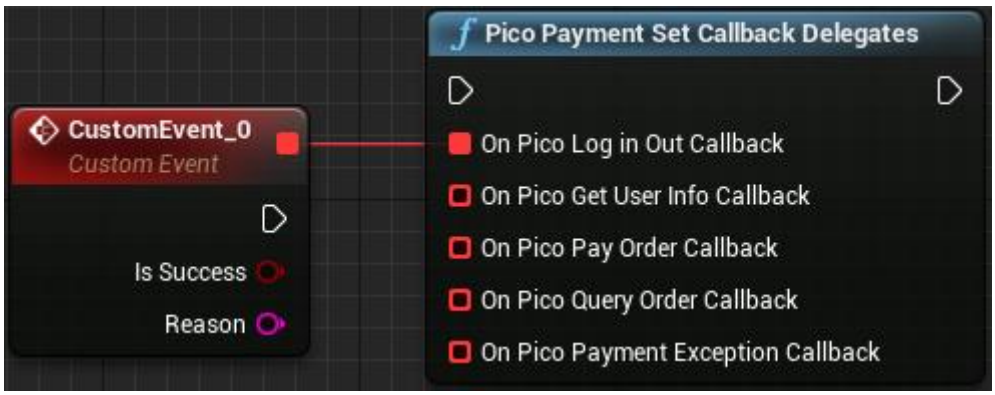
Among them, On Pico Payment Exception Callback is a callback for various exceptions. The exact meaning of the other callback function parameters will be introduced in the next section to introduce its related key functions.

7.1.3 User login

Pico provides Oauth2.0 based authentication authorization for developers, therefore, users need to log in before they pay. Please use the PicoPaymentLogin nodes we provide



Callback function: OnPicoLogInOutCallback, parameters are as following:

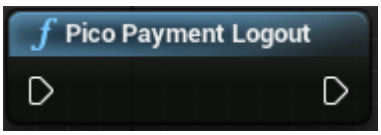


- IsSuccess: whether succeed in logging in or logging out (Boolean)
- Reason: The reason of success or failure for login and logout.

Users only need to login once, then the payment can be used directly. Two weeks after login the function will return a code (login expired code). After login is expired, users need to log in again

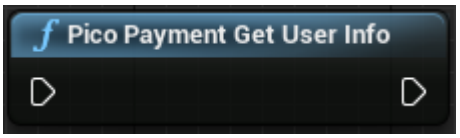
7.2 Other related APIs

7.2.1 PicoPaymentLogout



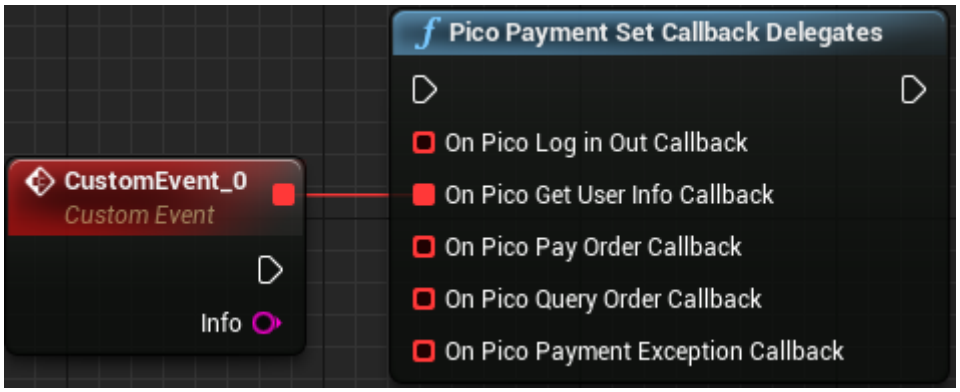
- Description: to log out
- Callback function: OnPicoLogInOutCallback as introduced above

7.2.2 PicoPaymentGetUserInfo



- Description：to get user info

- Callback function: OnPicoGetUserInfoCallback,



Info：An unprocessed Json string (string), query succeeded example is as following：


```
{"ret_code":"0000",
"data" : {
    "aboutme":"",
        "birthday" : 1460476800000,
        "phone" : "13100000000",
        "username" : "Admin",
        "email" : "",
        "gender" : "male",
        "lastname" : "",
        "openid" : "4f3148bdc34d9bca104927729a173b64",
        "firstname" : "",
        "avatar" : "http://172.31.83.11/upload/6dd6ee103714e967846c3d38ae48d511",
        "signature" : "14a25d7219d8dfc91e55f63286ae5c0a",
        "country" : "China",
        "city" : ""
},
"ret_msg":"调用成功"

}
```


Query failed example is as following：

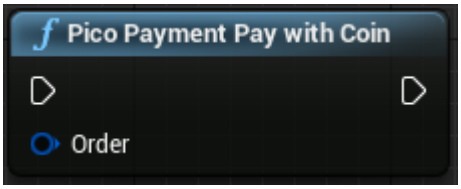
```
{
"ret_code":"00003000",
"ret_msg" : "签名验证失败"

}
```

Other ret_code codes and ret_msg are as listed:

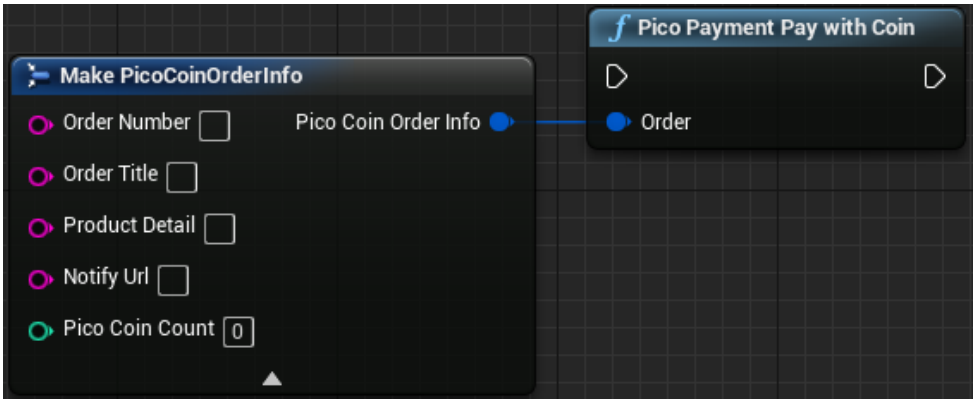
List 7.1 OnPicoGetUserInfoCallback output code ret_code and ret_msg

ret_code	ret_msg	meaning
0000	请求成功	Request success
00020000	数据库操作失败	Database operation failed
9999	系统错误	System error
00001000	参数错误	Parameter error
00002000	数据解析失败	Data parsing failed
00003000	签名验证失败	Signature verification failed
00003001	时间验证失败	Time validation failed
00060000	用户未找到	User not found
00060001	用户密码错误	User password error
00060002	用户登录未知错误	User logon unknown error
00061000	用户 token 未找到失败	Cannot find user token
00061001	用户 token 验证失败	User token validation failed
00061002	用户 token 未知错误	User token unknown error
00070001	应用验证失败	Application validation failed
00071001	应用密钥验证失败	Application key authentication failed
00080001	OAUTH_CODE 验证失败	OAUTH_CODE validation failed
00090001	REFRESH_TOKEN 验证失败	REFRESH_TOKEN validation failed
00100001	ACCESS_TOKEN 验证失败	ACCESS_TOKEN validation failed
00110001	SCOPE 验证失败	SCOPE validation failed

7.2.3 PicoPaymentPaywithCoin



- Description：pay with P-coin
- Input Value: Order



OrderNumber：The order number generated by the merchant itself, within 32 characters, can contain letters and numbers;

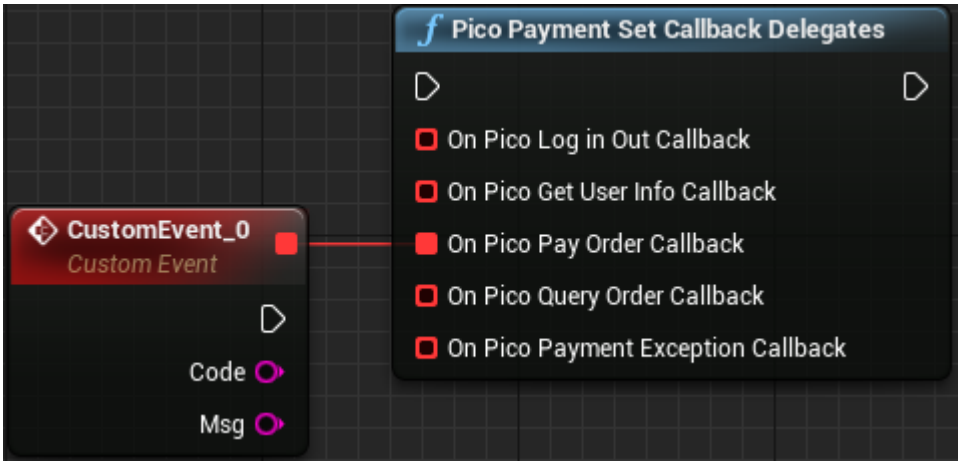
OrderTitle：order title

ProductDetail：the description of the product

Notify Url：the notify URL（not obligated）， **it must be URL for direct access, and cannot contain parameters**;

PicoCoinCount：coin count.

- Callback function: OnPicoPayOrderCallback

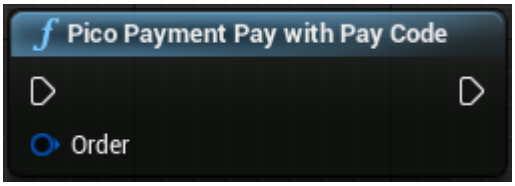


Code and Msg are as listed：

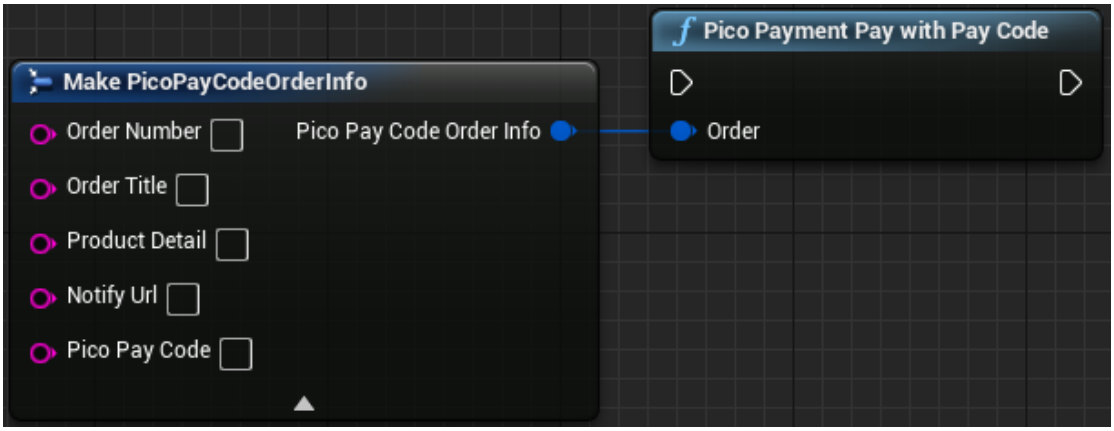
Code	Message	Meaning

Code	Message	Meaning
00000	网络异常	Network anomaly
10000	登录成功	Login successful
10001	用户未登陆	User not logged in
10002	请输入正确金额	Please enter the correct amount
10003	登陆过期，请重新登陆	Login expired, please re-login
11000	商户验证成功	Merchant verification successful
11001	商户验证失败	Merchant authentication failed
11002	用户验证参数错误或请求过期	User authentication parameter error or request expired
11003	商户未验证	Merchant not validated
12000	支付成功	Successful payment
12001	支付失败	Payment failure
12003	P 币不足	P coins are insufficient
12004	余额可用	Balance available
13000	生成订单	Generate order
13001	获取数据失败	Failed to obtain data
13002	生成订单失败	Failed to generate order
14000	查询订单成功	Query for order success
14001	订单不存在/有误	The order does not exist / is incorrect
14002	用户取消支付操作	User cancelled payment operation
15000	未输入商品信息	No input commodity information
15001	未输入预付 ID	No prepaid ID entered
15002	请输入 Pico 支付订单号或商户订单号	Eenter Pico order number or merchant order number
NOAU	商户无此接口权限	Merchants do not have this interface authority
SYST	系统错误	System error
APP_I	APP_ID 不存在	APP_IDdoes not exist
MCHI	MCHID 不存在	MCHIDdoes not exist
APP_I	app_id 和 mch_id 不匹配	app_id&mch_id not match
LACK	缺少参数	Parameter missing
SIGNE	签名错误	Signature error
NO_D	没有查询到数据/用户未充值	No query to data / user not prepaid
ORDE	订单已存在	The order already exists
PAY_C	消费代码不存在	The consumer code does not exist
PAY_C	用户已对商品代码消费	The consumer has consumed the commodity code

7.2.4 PicoPaymentPayWithPayCode[1]



- Description：pay with pay code
- Input value: Order



OrderNumber：The order number generated by the merchant itself, within 32 characters, can contain letters and numbers;

OrderTitile：order title ；

Product Detail：description of product;

Notify Url：he notify URL（not obligated）， **it must be URL for direct access, and cannot contain parameters;**

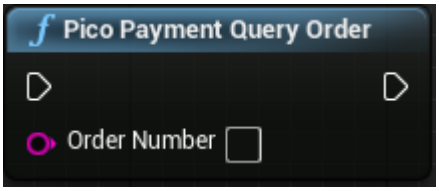
PicoPayCode：item code，user can obtain from 7.1.1 configuration.

- Callback function ：OnPicoPayOrderCallback，same as P-coin payment.

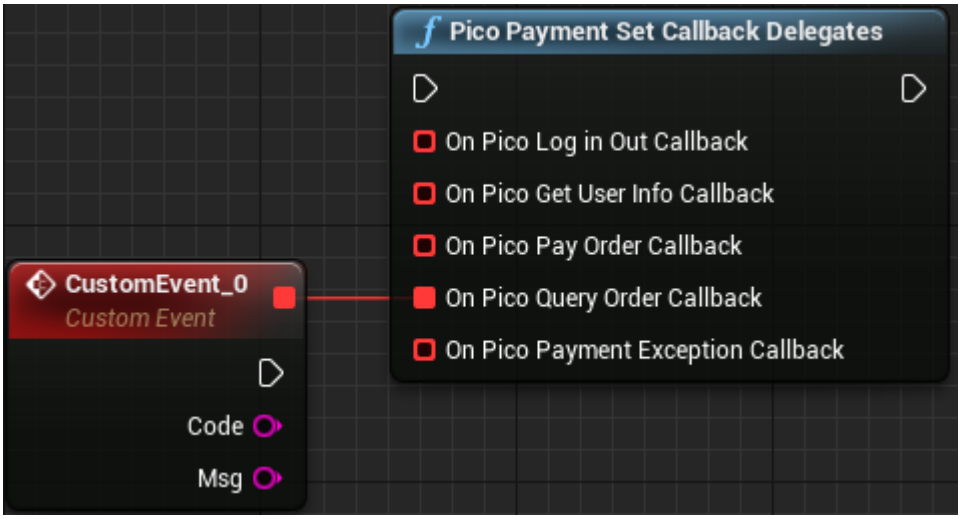
[1] Note: commodity code payment is a new payment system designed by developers' platform, and developers need to create different products under the game of developer platform, and fill in the commodity code. In the development of the game, you do not have to fill in the amount of the goods, directly fill in the corresponding commodity code, you can call the corresponding payment interface for payment.

- Callback function：OnPicoPaymentExceptionCallback，same as P-coin payment.

7.2.5 PicoPaymentQueryOrder



- Description：to query the order
- Input value : OrderNumber,order number（String）
- Callback function: OnPicoQueryOrderCallback：



Parameter is the same as OnPicoPayOrderCallback

7.3 Developer server interaction

After payment is completed, the payment system will send the relevant payment results and user information to the game developer, and the developer needs to receive, process, and response.

When interacting with backend, if the response that the payment system received is not either successful or overtime, it will be regarded as failed. The payment system will periodically resend

notifications with certain strategies to increase the success rate of the notification, but it does not guarantee that the notification will eventually succeed.

The same notification may be sent to the developer ‘s system several times, so the developer’ s system must be able to process duplicate notifications correctly. We suggest to check the status of corresponding business data and determine whether the notification has been dealt with or not after receiving the notification of processing. If not, the system will start processing, otherwise, it will return succeeded directly. Before check and process the business data status, concurrency control using data locks should be performed to avoid data confusion caused by reentrance functions.

The developer server needs to implement the following interface for receiving requests from the Pico server to obtain payment results and user information from the Pico payment system:

List 7.2 The interface that the merchant server needs to implement

<table>
<tr><td>Name</td><td colspan="3">Payment results callback interface</td></tr>
<tr><td>Request type</td><td colspan="3">POST</td></tr>
<tr><td>Request URL</td><td colspan="3">payment，PayOrder input value notify_url</td></tr>
<tr><td>Request format</td><td colspan="3">JSON</td></tr>
<tr><td>Return format</td><td colspan="3">JSON</td></tr>
<tr><td>Require Login</td><td colspan="3">Yes</td></tr>
<tr><td>Parameters</td><td colspan="3">Check details in “List 7.3 Notification parameters in the payment results notification” .</td></tr>
<tr><td>Return value</td><td>name</td><td>type</td><td>description</td></tr>
<tr><td></td><td>ret_code</td><td>string</td><td>Error code.</td></tr>
</table>

	ret_msg | string | Error information string. Check details in "List 7.4 Return result"

Example	{
	"ret_code": "SUCCESS",
	"ret_msg" : "OK"
	}

Update description	

List 7.3 Notification parameters in the payment results notification

Field name	Variable name	Required	Type	Description
Return status code	ret_code	Yes	String	SUCCESS/FAIL This is communication identification, not transaction identification, please use result_code to judge if the deal is finished
Return message	ret_msg	No	String	Return message, if not null, it contains error message Signature failed, parameter format check error

Field name	Variable name	Required	Type	Description
error code	sub_code	No	String	Error code
Error code message	sub_msg	No	String	Message description of error
Pico trade number	trade_no	Yes	String	Pico trade number
Merchant trade number	out_trade_no	Yes	String	Merchant trade number
App ID	app_id	Yes	String	App ID
Merchant ID	mch_id	Yes	String	Merchant ID
User ID	open_id	Yes	String	User ID under each app id
Device ID	device_id	No	String	Device ID
Random string	nonce_str	Yes	String	Random strings, not longer than 32 bits. A random number generation algorithm is recommended
signature	signature	Yes	String	signature, see the signature generation algorithm

Field name	Variable name	Required	Type	Description
Result code	result_code	Yes	String	SUCCESS/FAIL
Trade type	trade_type	Yes	String	The type user paid by.
Fee type	fee_type	Yes	String	The currency user chose.
Total fee	total_fee	Yes	String	Total fee of the order
Receipt fee	receipt_fee	Yes	String	Actual value of the order
Buyer pay fee	buyer_pay_fee	No	String	The fee user paid
Coupon fee	coupon_fee	No	String	Coupon fee
Merchant data package	attach	No	String	Merchant data package
Pay time	pay_time	Yes	String	Payment finish time, yyyy-MM-dd HH:mm:ss

List 7.4 Return result

Field name	Variable name	Required	Type	Description
Return code	ret_code	Yes	String	SUCCESS/FAIL SUCCESS: merchant received the notification and verify is successful

Return message	ret_msg	No	String	Return message, if not null, it contains error message: signature failed, parameter format check error

Special reminder: developer system must sign to verify the contents for payment results to prevent data leakage, which could result in "false notification" and loss of funds.

The signature check rule is:

1. For the returned list of parameters, remove the signature parameter, and adds key = "app_secret", value=paykey. Then, sort out the key value by natural order, separate the parameters by & if there are more than one, and encrypt by MD5.

2. Compare the encrypted string with the acquired signature

The signature function is as following:

```
public static String createSign(Map<String, Object> result, String paykey)
{
    if (result == null || result.size() == 0)
        return null;
    result.put("app_secret", paykey);
    String sign = result.get("signature");
    result.remove("signature");
    String[] tmp = new String[result.size()];
    int i = 0;
    for (String key : result.keySet())
    {
        tmp[i++] = key;
    }
    Arrays.sort(tmp);
    String sign = "";
    for (String string : tmp)
    {
        if (m.get(string) == null)
            continue;
```

```
            sign += string + "=" +
URLEncoder.encode(m.get(string).toString()
                    , "utf-8") + "&";
        }
        if (sign.endsWith("&"))
            sign = sign.substring(0, sign.length() - 1);
        Log.i(TAG, "createSign: " + sign);
        String localSign = MD5.MD5(sign);
        return localSign.equal(sign);

}
```

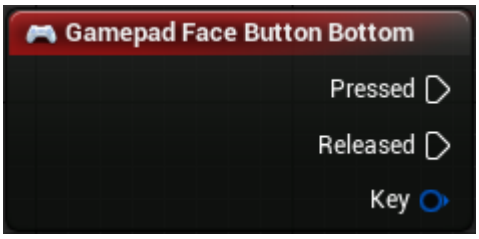
8 Other Instructions

8.1 Goblin HMD Buttons

Figure 8.1 Goblin HMD buttons

Table 8.1 GoblinHMD keys mapping

Goblin HMD buttons	Description
1	Android standard key code - POWER
2	Android standard key code - HOME

Goblin HMD buttons	Description
3	
4	Android standard key code - VOLUME_UP
5	Android standard key code - VOLUME_DOWN

8.2 G2 HMD Buttons

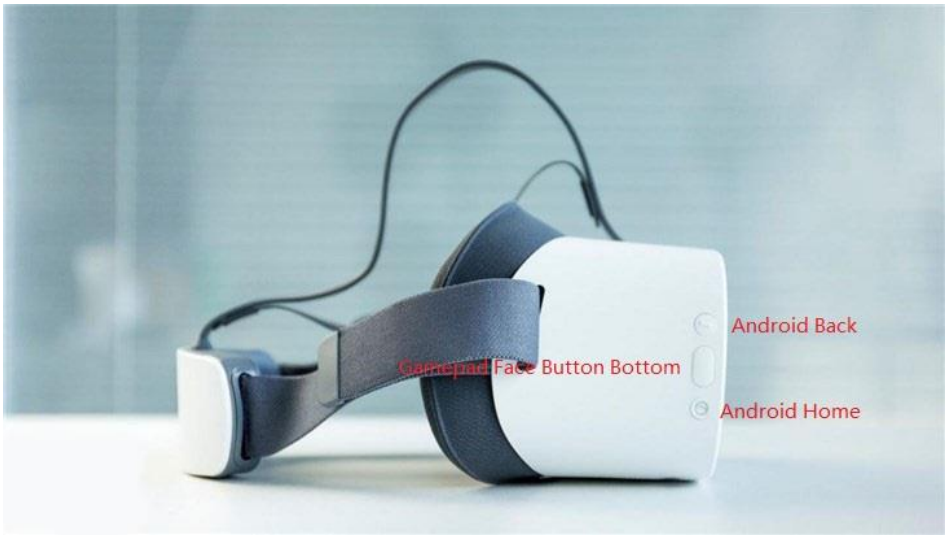


Figure 8.2 G2 HMD button

8.3 Pico Neo HMD Buttons

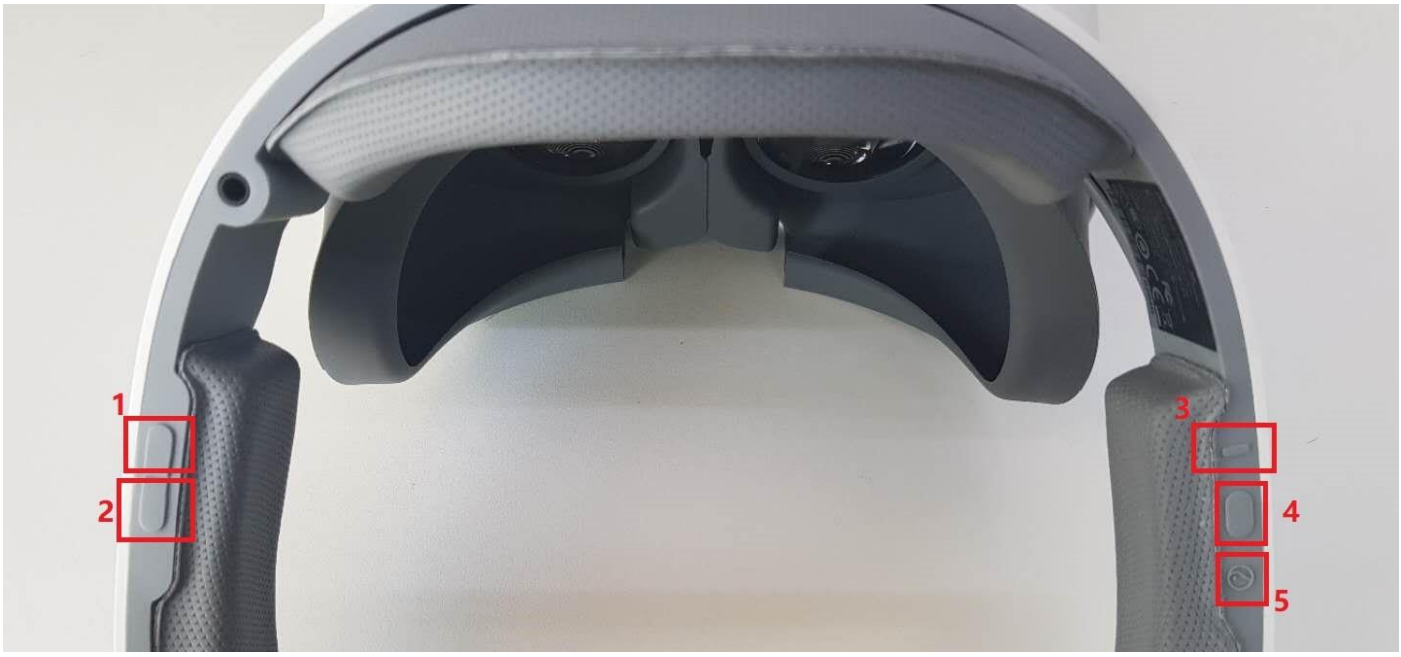
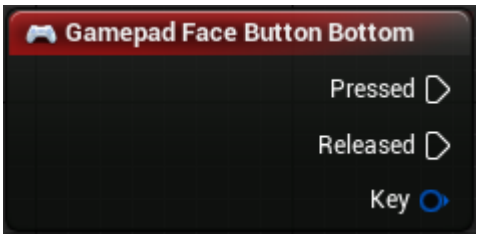


Figure 8.3 Pico Neo HMD buttons

Table 8.2 Pico Neo keys mapping

Goblin HMD Buttons	Description
1	Android standard key code - VOLUME_UP
2	Android standard key code - VOLUME_DOWN
3	Android standard key code - BACK
4	
5	Android standard key code - HOME

8.4 Pico Neo Safe Area

Pico Neo safe area is similar to the Oculus Rift Guardian System and SteamVR Chaperone System, will show the notice when the player is out of the safe range, in case the player run into the wall or something else. Even though the VR all-in-one device is restricted by the inside-out tracking and have no external sensor to input absolute safe area axis, we still suggest the developers to set a range value to make sure the player’s safety.

Thus, we create an Actor blueprint class as the safe area. Actually it is a cylinder with the normals face to the inner side to remind the player of the range of the safe area:

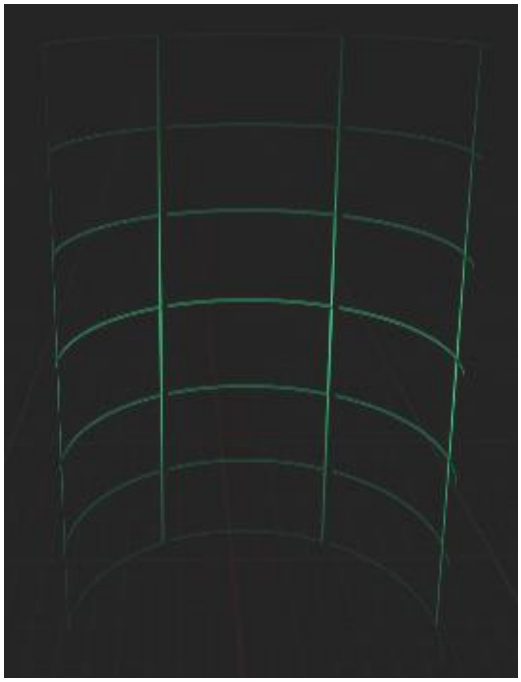


Figure 8.4 Safe area

It is suggested to add the below node after the BeginPlay event of the Default Pawn of the scene:

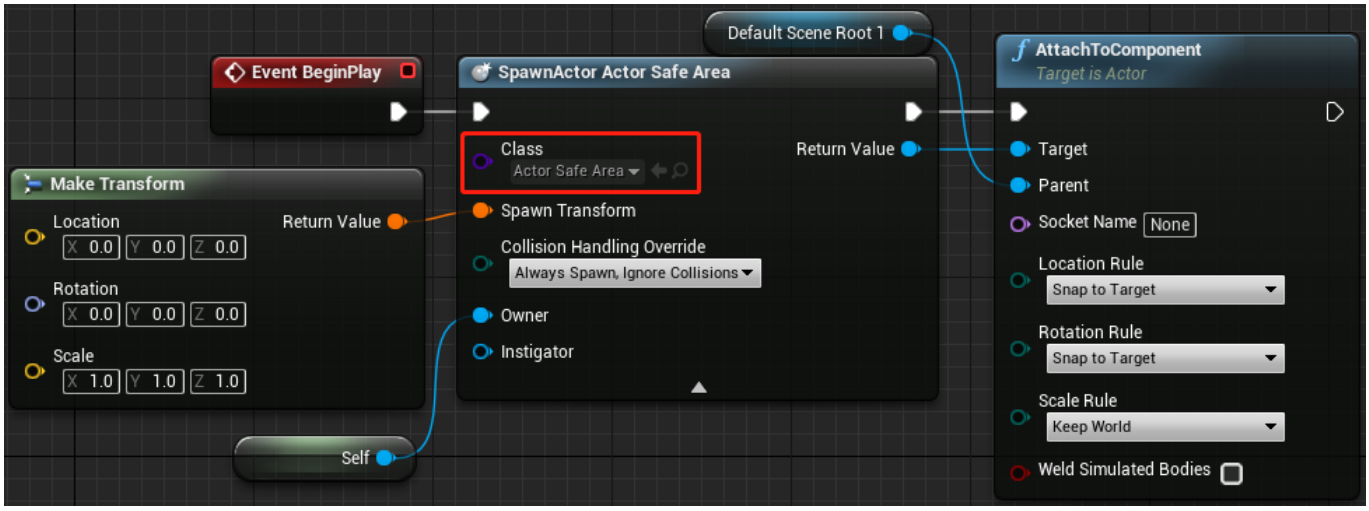


Figure 8.5 Add safe area

Then display and hide the safe area dynamically according to the Camera’s or MotionController’s relative position to parent component

Moreover, Actor_SafeArea blueprint class provides two methods to get and set the radius value of the safe area:

Figure 8.6 Functions of Safe Area

We suggest that develops would give the ability to users of setting the range of the safe area, so that the safe area could fit the actual environment better.

8.5 Enable Pico Neo 6DoF function

The default value of head and hands tracking in SDK is 3DoF. For the devices with 6DoF tracking function, such as Pico Neo, please refer to the following steps to enable 6DoF tracking:

Open the menu Edit->Project Settings->Plugins->Pico Neo, enable the options showed in below

picture:

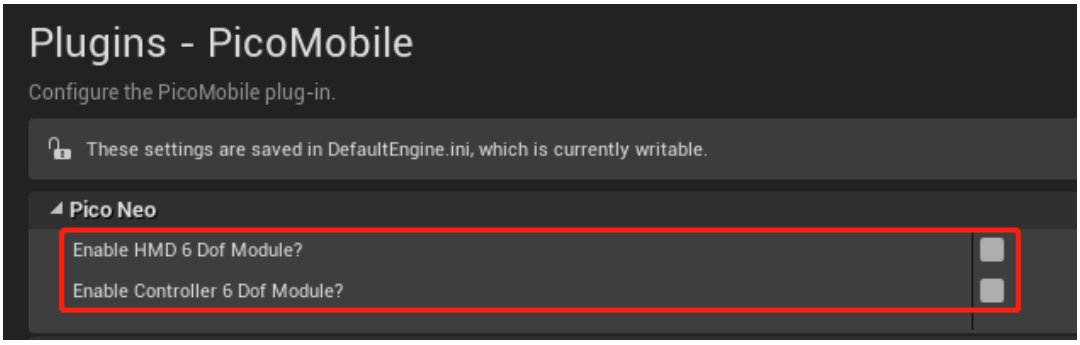


Figure 8.7 Pico Neo 6DoF options

9 Known Issues

- The SDK doesn’t support “VR Preview” in editor.

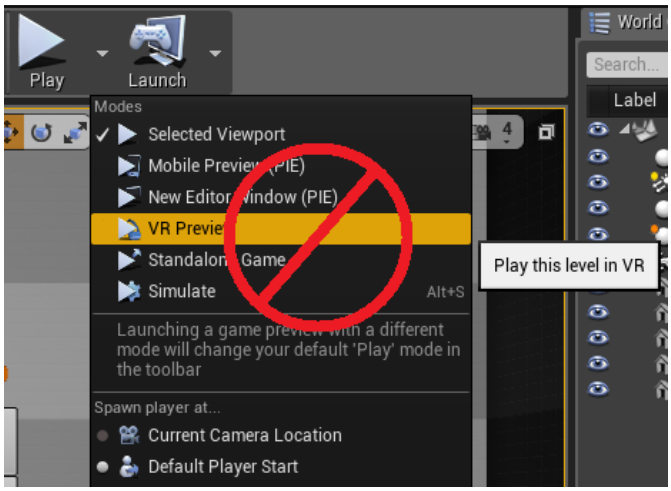


Figure 9.1 The SDK doesn’t support “VR Preview” in editor

- For UE4.18, before package, please disable the "Enable Gradle instead of Ant" option in Edit >>Project Settings >>Platform >>Android.

Figure 9.2 Uncheck Enable Gradle instead of Ant

10 FAQ

Q: Why there are virtual buttons for the Android system after Packaging the project?

A: Check: Project Setting→Platform→Android→APKPackaging→Enable FullScreen Immersive on KitKat and above devices:

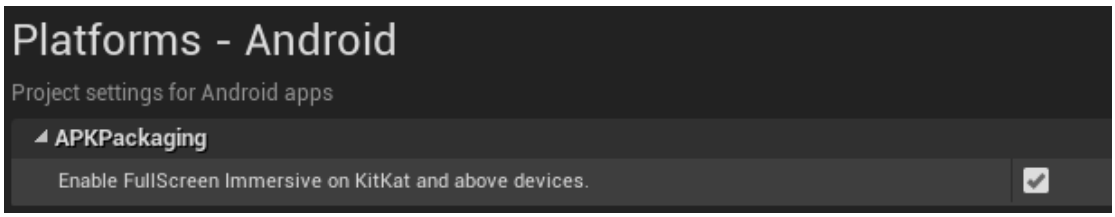


Figure 10.1 Check: Enable FullScreen Immersive on KitKat and above devices

Q: How to increase the FPS of a game?

A: To increase the FPS, there are two methods we suggest:

1、Uncheck Ambient Occlusion and Ambient Occlusion Static Fraction. At this stage, global illumination is not highly required in mobile VR games, so ambient light shielding can be turned off. Project Setting/Engine/Default Setting, uncheck Ambient Occlusion and Ambient Occlusion Static Fraction:

Figure 10.2 Uncheck Ambient Occlusion and Ambient Occlusion Static Fraction

2、Uncheck Mobile HDR。Project Setting/Engine/Rendering/Mobile，uncheck Mobile HDR：

Figure 10.3 Uncheck Mobile HDR

Q：How to reduce the size of the game package?

A：Please refer to the following methods：

1、Project Setting/ Project /package　check Create compressed cooked packages；

2、Uncheck the Plugins which are not in use；

3、Delete the resources that are not used in the content browser。

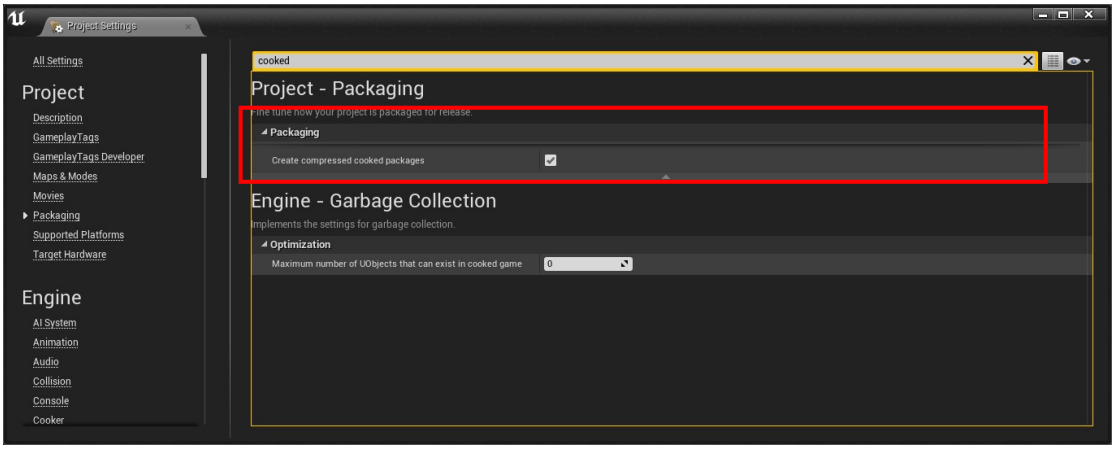


Figure 10.4 Create compressed cooked packages

Q：How to optimize the display effect?

A: Project Setting/Engine/Rendering/Mobile，Choose the appropriate Mobile MSAA level，which can effectively enhance the display effect by using Anti-Aliasing setting. The higher the multiplier is, the better the optimization effect will be, but it will increase performance consumption and reduce the FPS simultaneously.

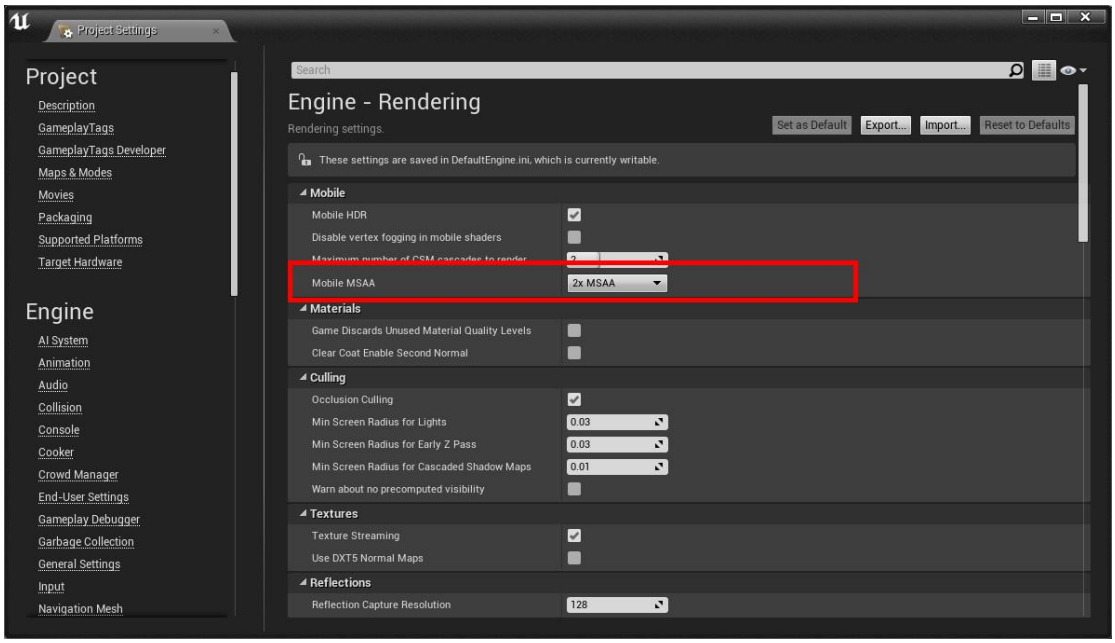


Figure 10.5 Choose the appropriate Mobile MSAA

Q: The virtual system buttons are displayed in the scene, how to disable them?

A: Set "Enable FullScreen ..." option enabled in UE Android project setting.

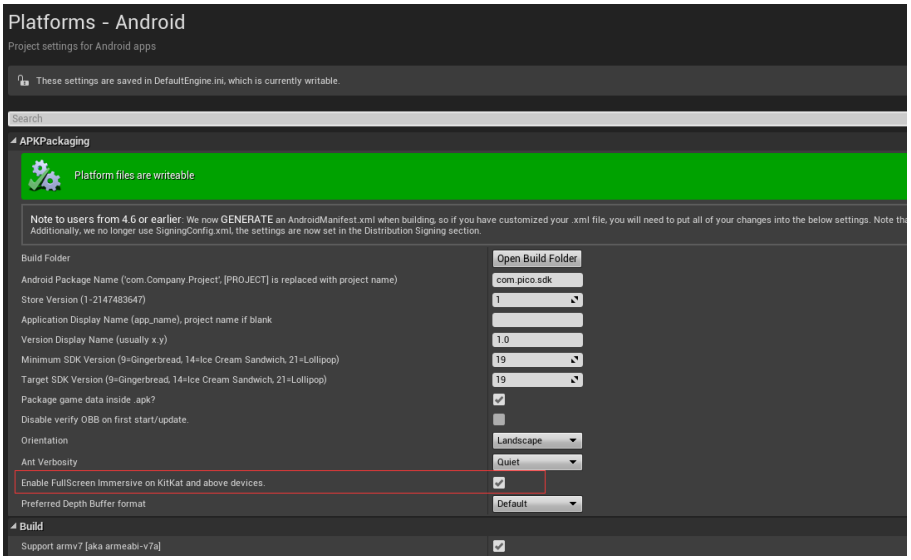


Figure 10.6 Enable full screen

Q: I used the SDK strictly according to this document for my project named "test". Why it still package failed?

A: The word "test" is a command keyword in Unreal Engine. Use command keyword name for project may cause failures in packaging. Thus, you'd better avoid these keywords when creating project as the following picture.

```
// Configuration names:
case "DEBUG":
    Configuration = UnrealTargetConfiguration.Debug;
    break;
case "DEBUGGAME":
    Configuration = UnrealTargetConfiguration.DebugGame;
    break;
case "DEVELOPMENT":
    Configuration = UnrealTargetConfiguration.Development;
    break;
case "SHIPPING":
    Configuration = UnrealTargetConfiguration.Shipping;
    break;
case "TEST":
    Configuration = UnrealTargetConfiguration.Test;
    break;
```

Q: Why Goblin motion controller flashes sometimes?

A: Epic moves the smooth motion controller by default and updates the motion controller's position and

pose by default, but this also creates a bug that the model sometimes flashes. To avoid this flicker, simply select the Motion Controller component and check Disable Low Latency Update in its detail panel: