

产品名称ShadowSDK快速入门	
产品版本ShadowSDK快速入门	Total 17 pages 共 17页

ShadowSDK快速入门

Prepared by		Date	
拟制	王超群	日期	2020/04/18
Reviewed by		Date	
审核		日期	
Approved by		Date	
批准		日期	



SHADOW CREATOR

Shadow Creator Information Technology Co.,Ltd.

上海影创信息科技有限公司

All rights reserved
版权所有 侵权必究



Revision Record 修订记录

Date 日期	Revision Version 修订 版本	Release Notes	Change Description 修改描述	Author 作者
2020/04/18	V1.0.0	ShadowSDK快速 入门	初稿完成	王超群



Catalog 目 录

1	ShadowSDK简介	4
1.1	简介	4
2	快速入门	4
2.1	ShadowSDK下载	5
2.2	制作一个简单的例子	5
2.3	打包安装APK	13
2.4	获取本例程项目源码	17
3	开发者文档	17
3.1	简介	17



1 ShadowSDK简介

1.1 简介

开发者可以利用ShadowSDK在Unity3d中开发影创MR设备应用，ShadowSDK提供简单的影创MR设备开发环境和最优化的设备访问，使开发者快速高效的完成应用开发
本文档旨在让开发者快速上手使用ShadowSDK开发应用

2 快速入门

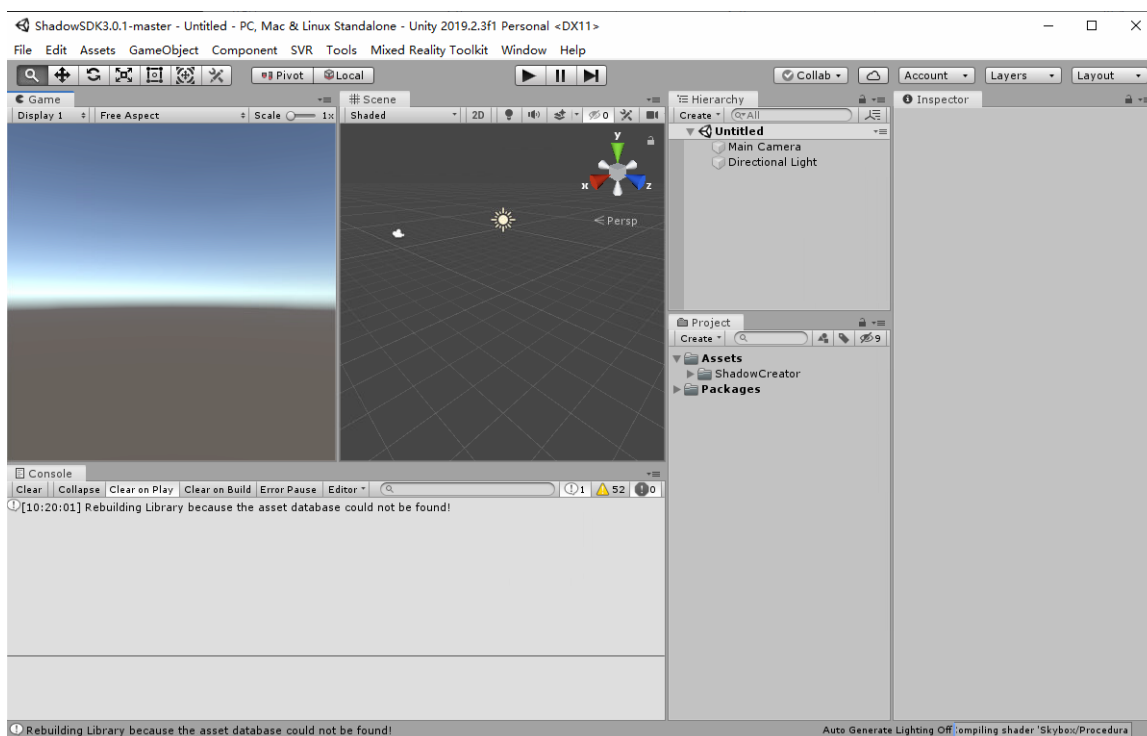
2.1 ShadowSDK下载

[点击进入ShadowSDK下载页面](#)

根据需要的ShadowSDK版本进行下载即可，ShadowSDK下载后解压即可得到ShadowSDK Unity工程，用ShadowSDK建议的Unity版本打开此工程

[点击查看ShadowSDK各版本建议的Unity版本](#)

效果如下：





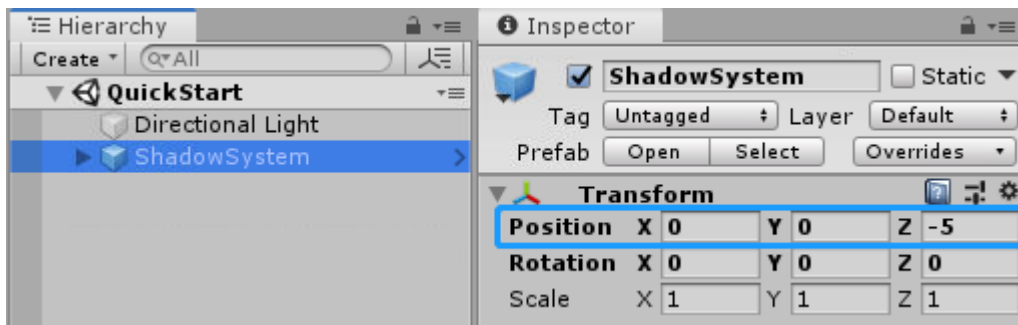
2.2 制作一个简单例子

如下快速集成一个简单的例子，包含功能如下：

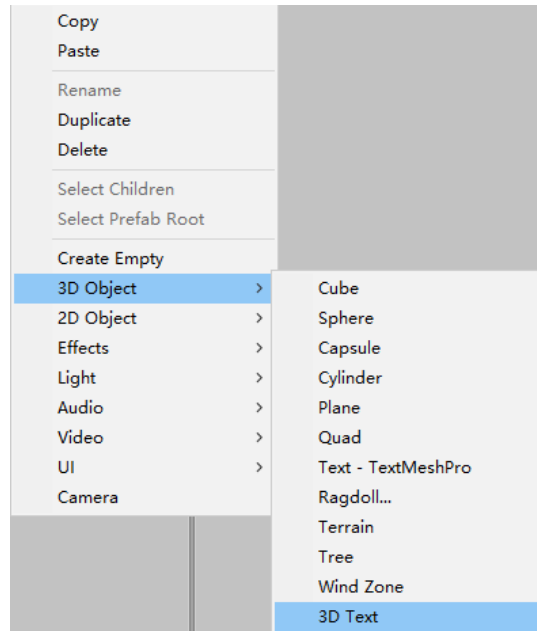
- a) 显示文字：Helloworld
- b) 通过点击Cube，更改文字显示”点击了Cube”
- c) 通过点击UIButton，更改文字显示”点击了UIButton”
- d) 自动点击UIButton，更改文字显示”自动点击了UIButton”
- e) 拖拽(快速或缓慢)一个Cube

详细步骤如下：

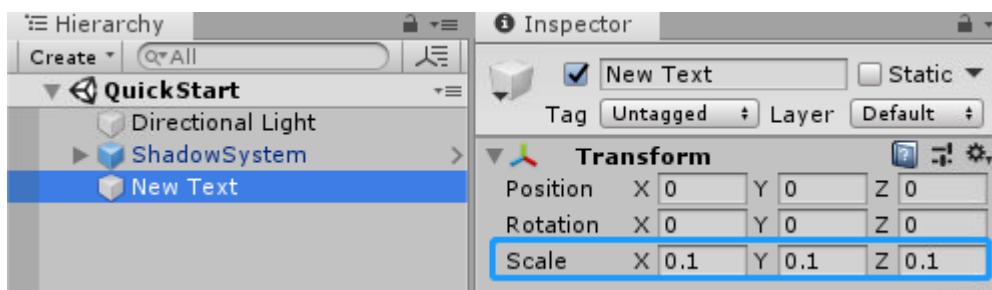
- 1：新建一个Scene，保存为QuickStart.scene
- 2：删除场景中的MainCamera，从Assets/Resources/Prefabs/ShadwoSystem拖拽到Hierarchy视图，并设置其Transform为如下状态



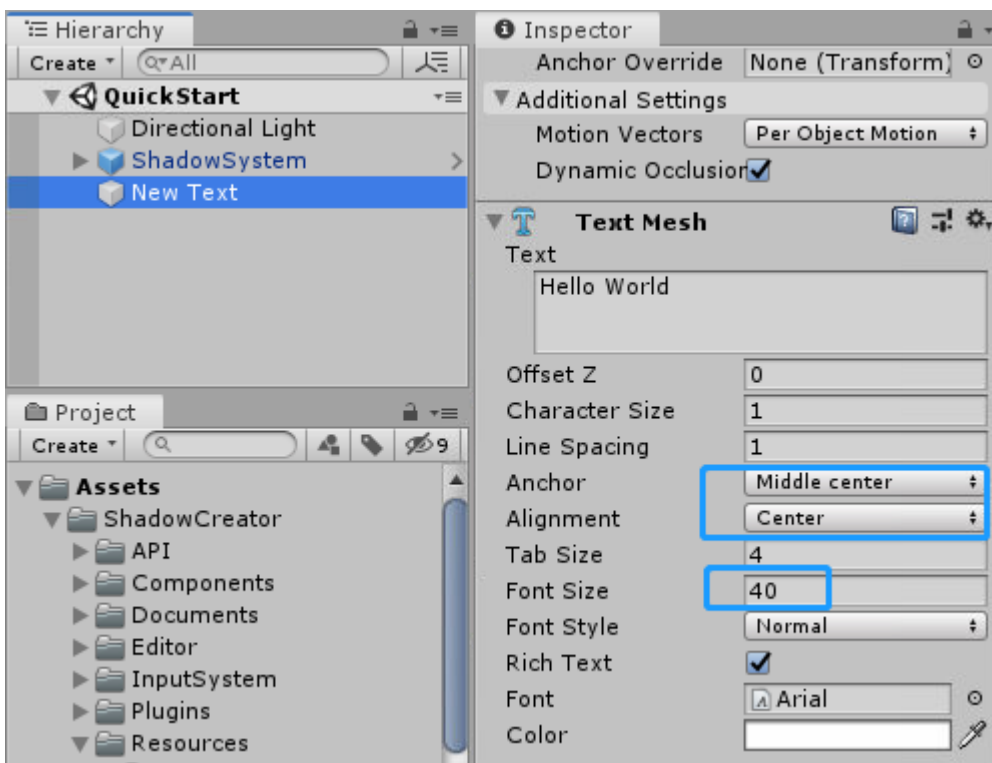
- 3：创建一个3D Text,并设置其属性



设置Transform

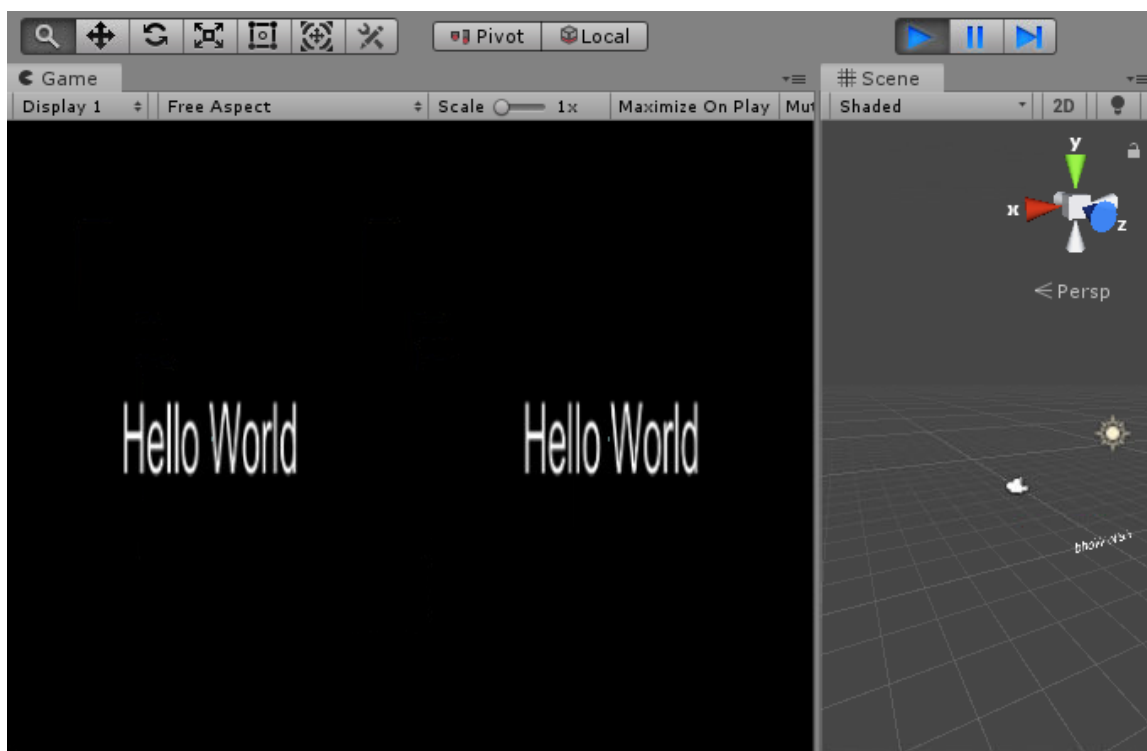


设置Text Mesh

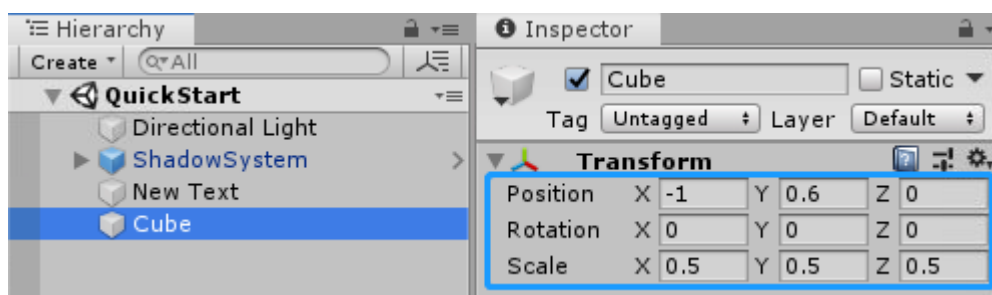




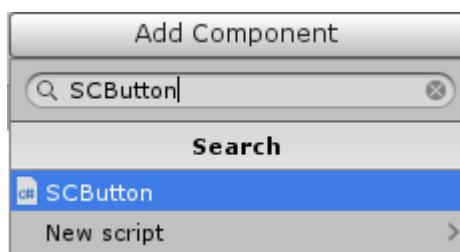
点击Play按钮，效果如下：



4: 创建一个Cube，并设置其属性



给Cube添加SCButton组件



编写一个ChangeText.cs脚本，内容如下：

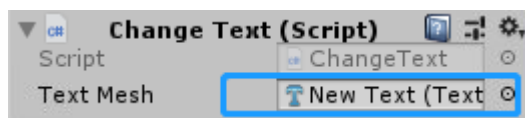
```
using System.Collections;  
using System.Collections.Generic;  
using UnityEngine;
```



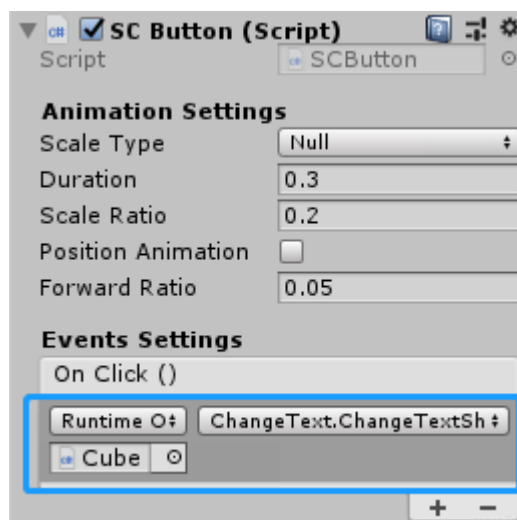
```
public class ChangeText : MonoBehaviour
{
    public TextMesh textMesh;

    public void ChangeTextShowCube() {
        if(textMesh) {
            textMesh.text = "点击了Cube";
        }
    }
}
```

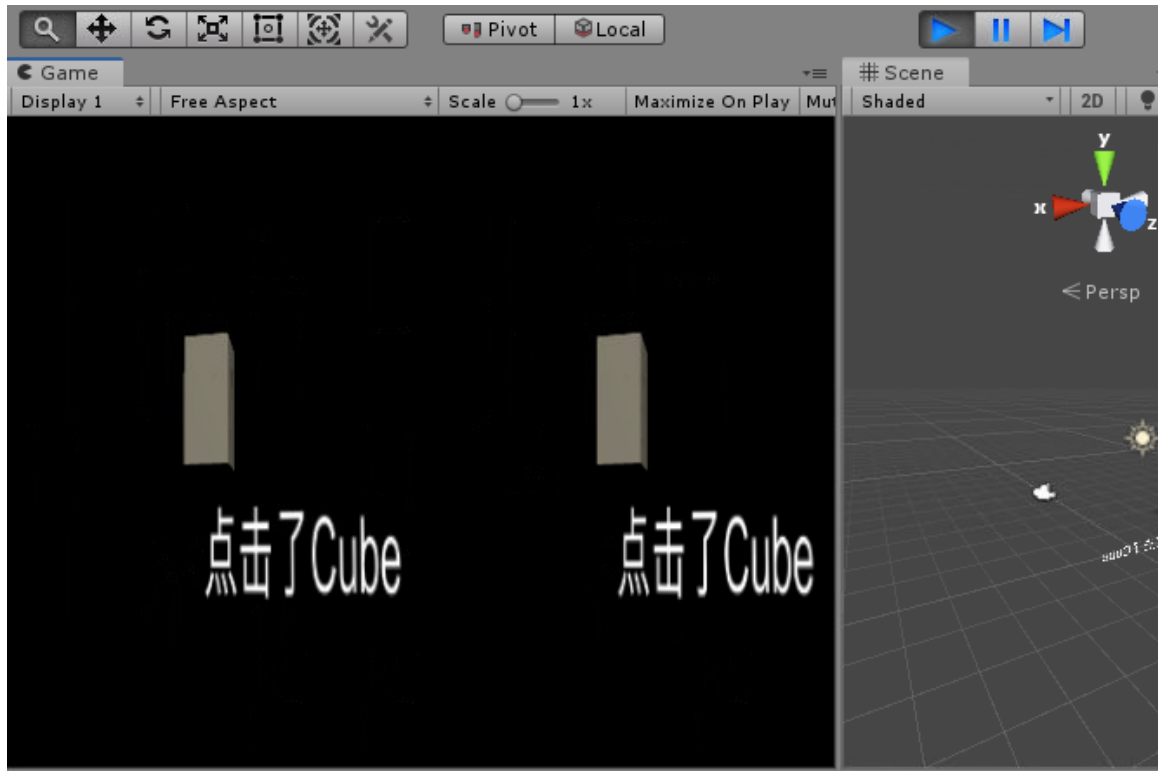
将ChangeText挂载到Cube上，并拖拽New Text赋值给textMesh



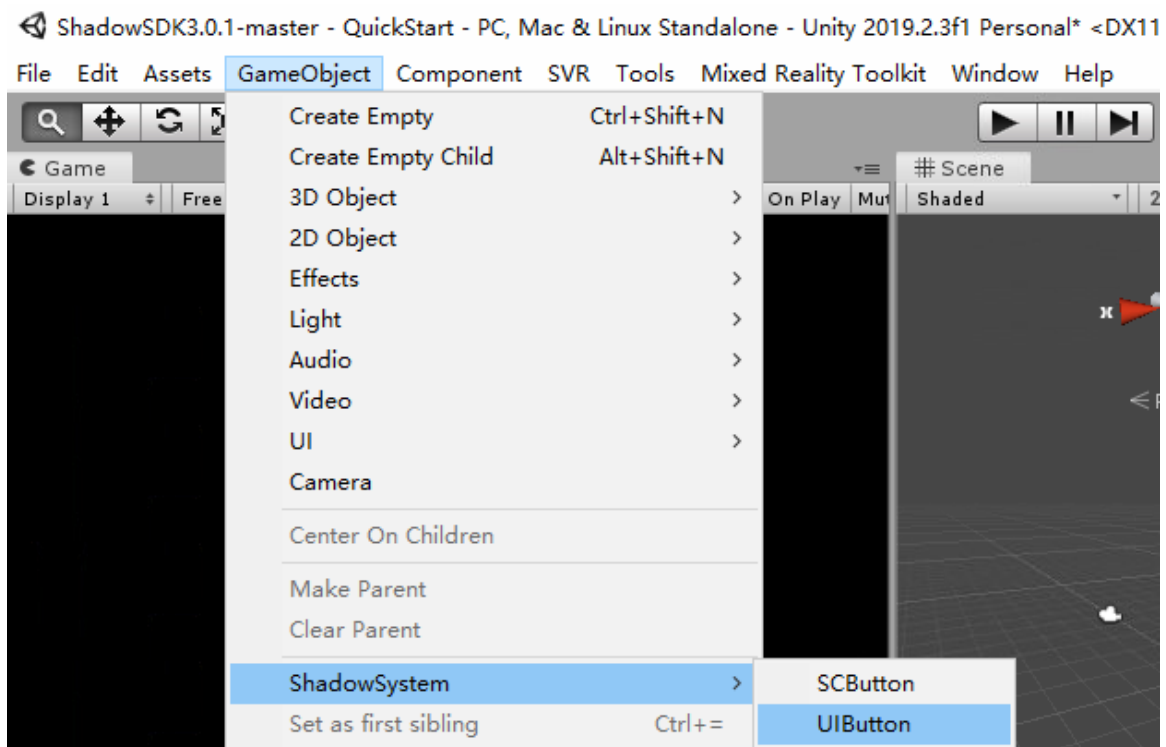
给Cube上的Scbutton组件添加点击触发方法（ChangeText脚本中的ChangeTextShowCube方法）



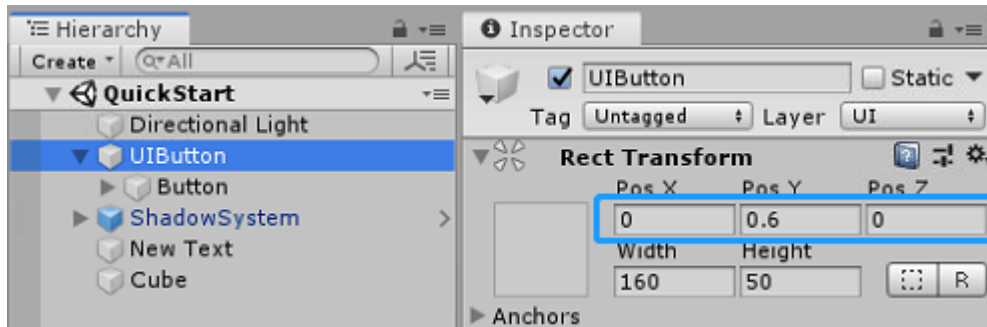
点击Play按钮，鼠标右键长按并拖动，将光圈对准Cube，点击鼠标左键，效果如下：



5: 添加一个UIButton, 如下图:



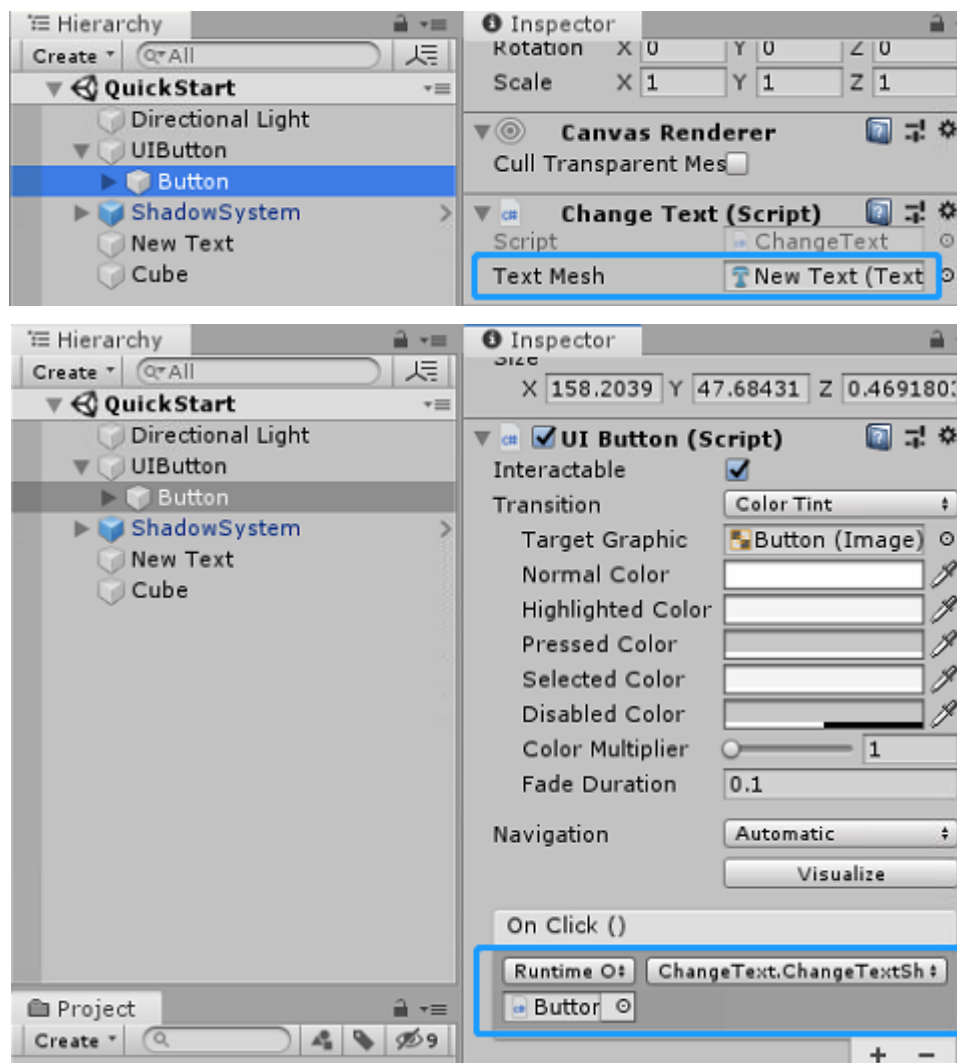
设置其Transform属性



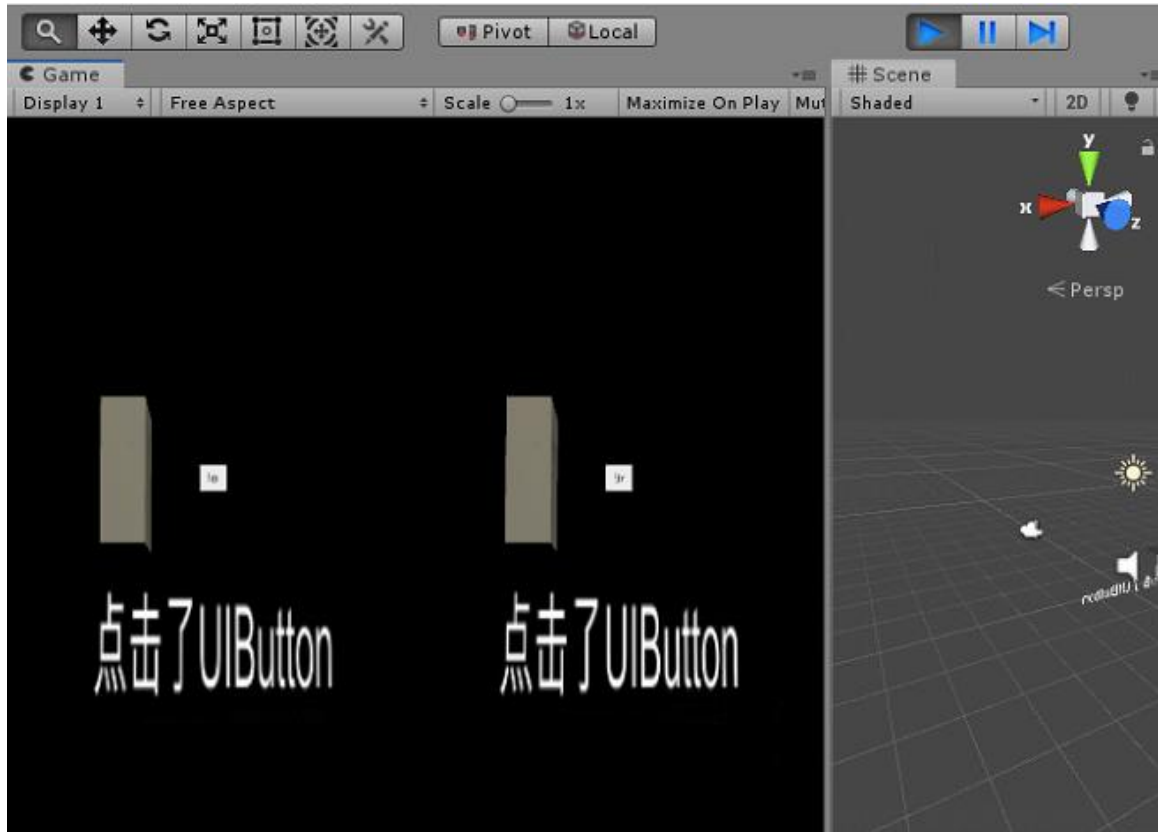
修改ChangeText.cs脚本，增加内容如下：

```
public void ChangeTextShowUIButton() {  
    if(textMesh) {  
        textMesh.text = "点击了UIButton";  
    }  
}
```

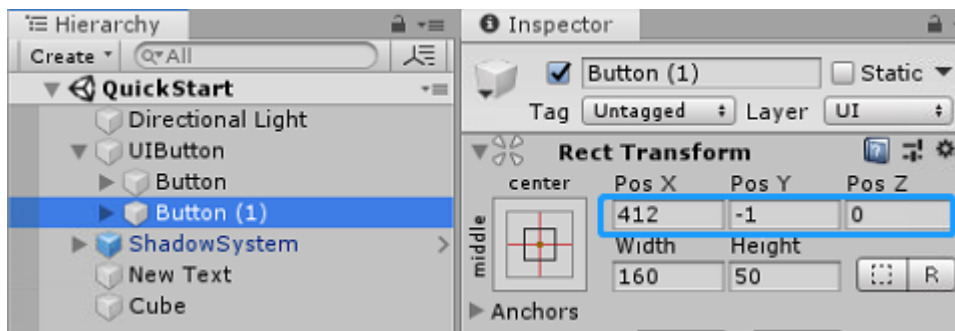
将ChangeText.cs拖拽到UIButton/Button上，并绑定点击时触发的方法：



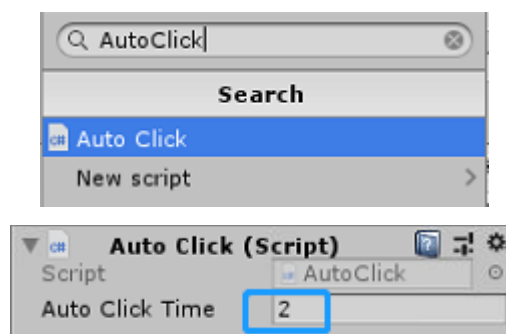
点击Play按钮，鼠标右键长按并拖动，将光圈对准UIButton，点击鼠标左键，效果如下：



6: 复制UIButton/Button, 并设置其属性如下图:



给Button(1)添加组件AutoClick,并设置AutoClickTime为2



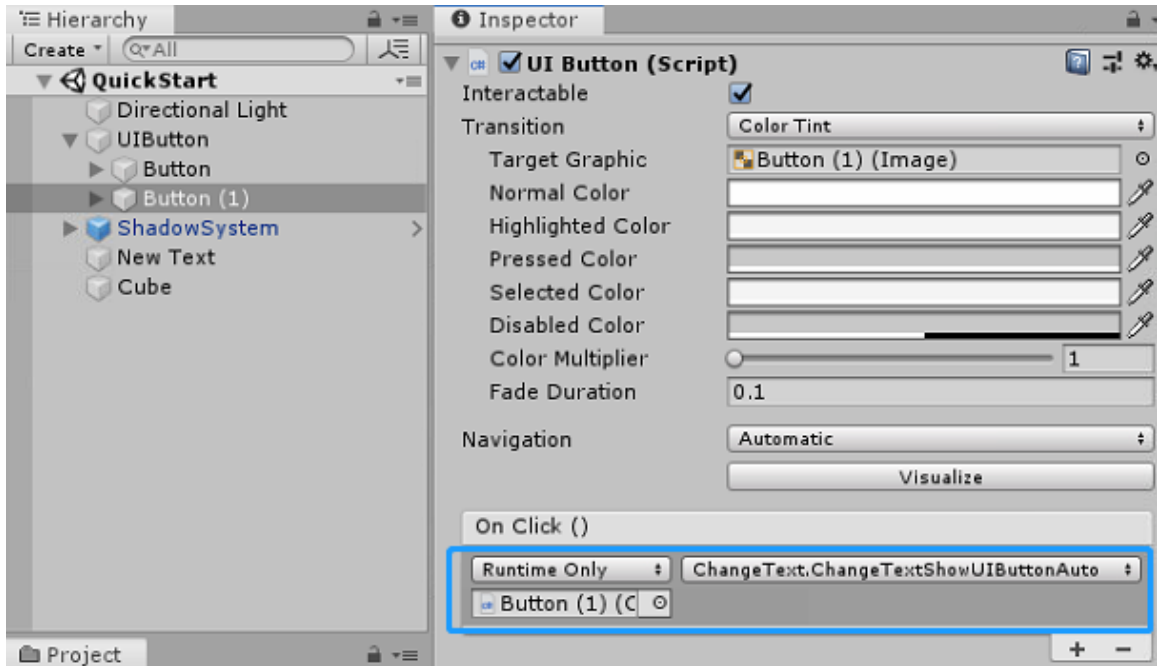
修改ChangeText.cs脚本, 增加内容如下:

```
public void ChangeTextShowUIButtonAuto() {  
    if(textMesh) {
```

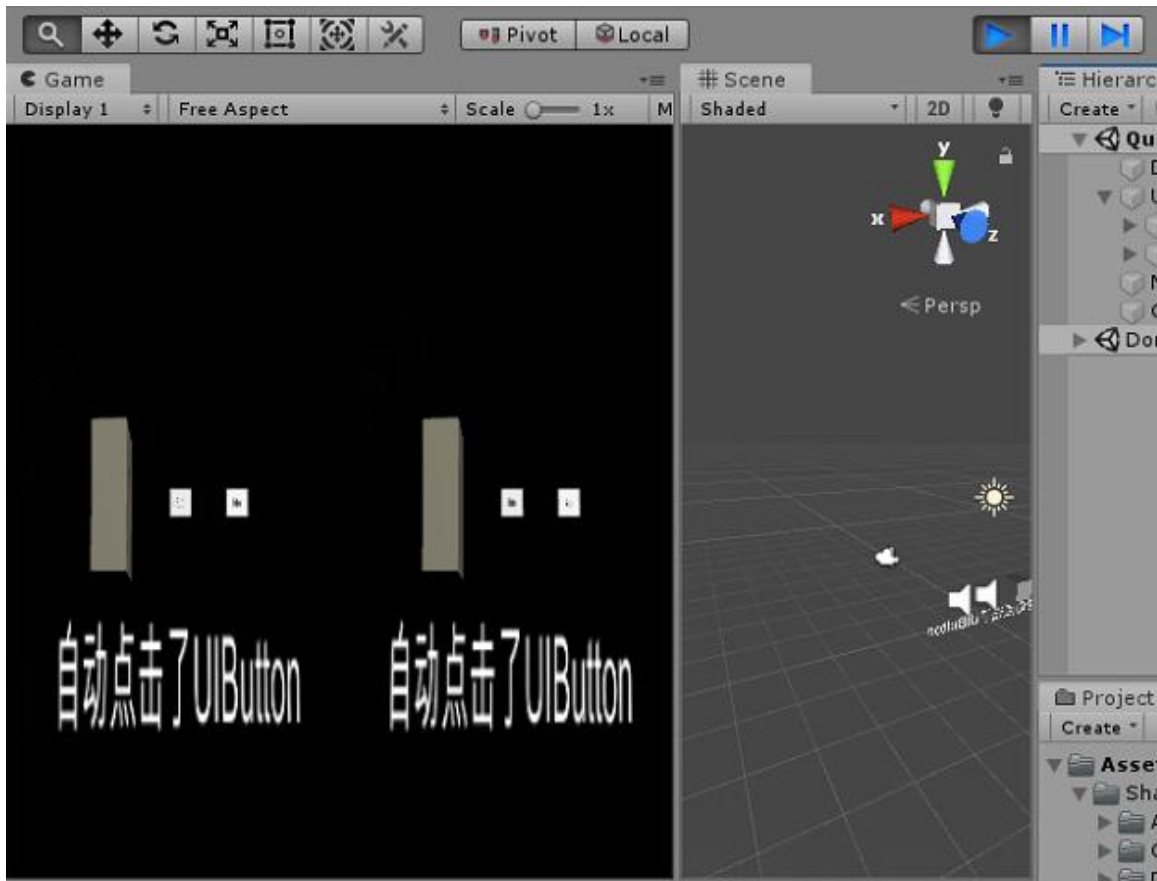


```
textMesh.text = "自动点击了UIButton";  
}  
}
```

修改Button(1)绑定点击时触发的方法：

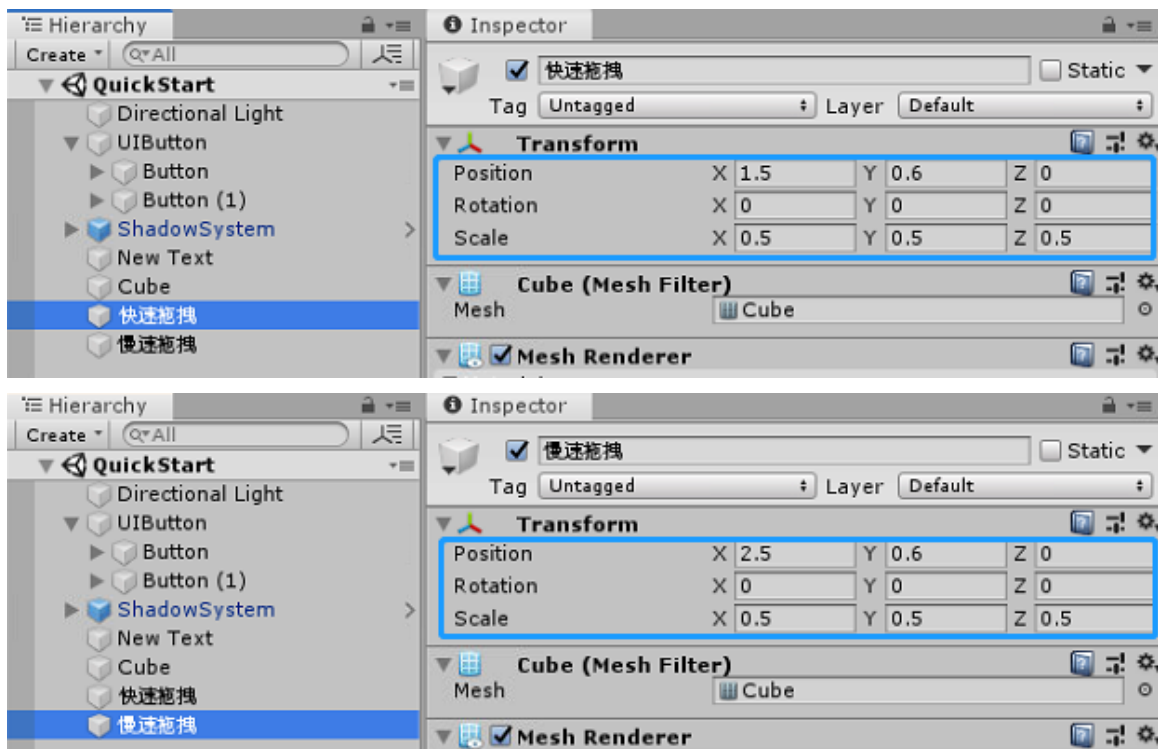


点击Play按钮，鼠标右键长按并拖动，将光圈对准UIButton，等待2秒，效果如下：

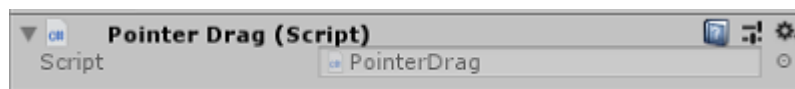




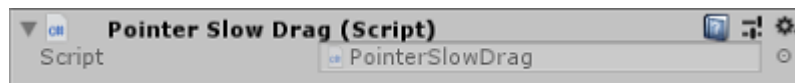
7: 创建两个Cube，名为快速拖拽，慢速拖拽，设置其Transform如下：



给快速拖拽的Cube添加组件PointerDrag



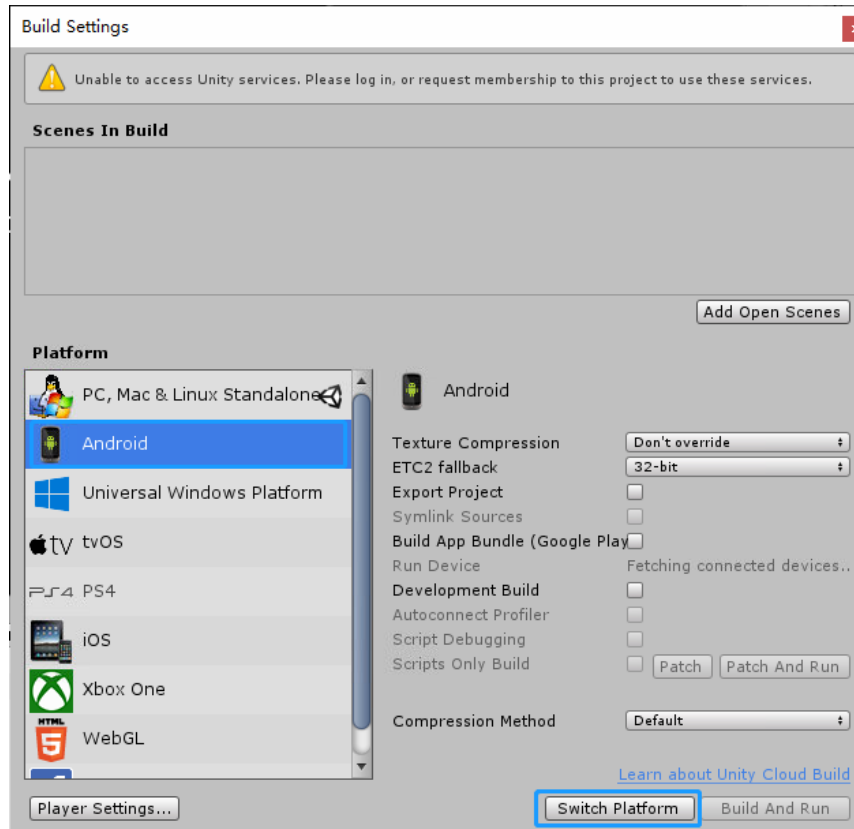
给慢速拖拽的Cube添加组件PointerSlowDrag



点击Play按钮，鼠标右键长按并拖动，将光圈对准快速拖拽或慢速拖拽，长按鼠标左键并拖动Cube

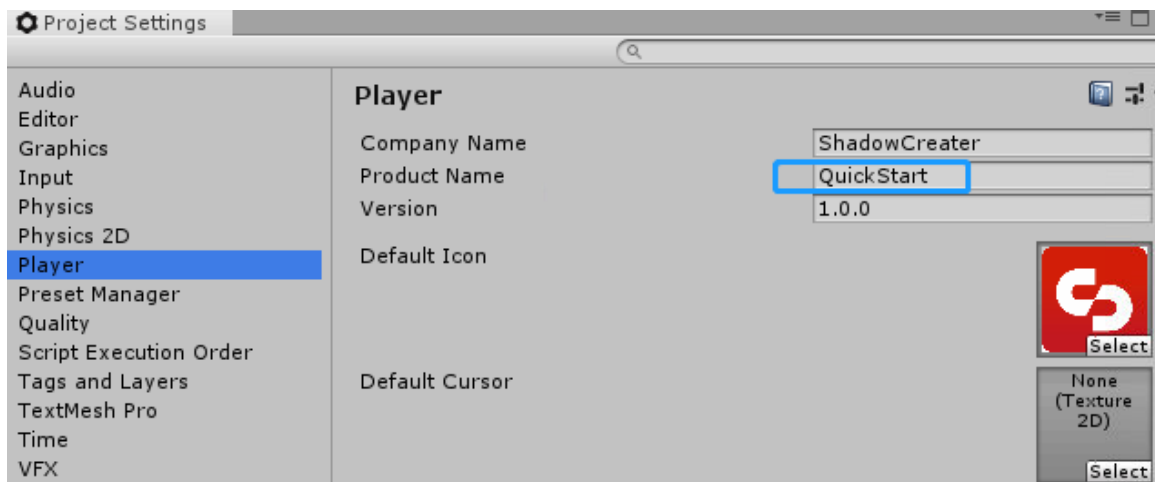
2.3 打包安装APK

点击File->BuildSetting，切换到Android平台

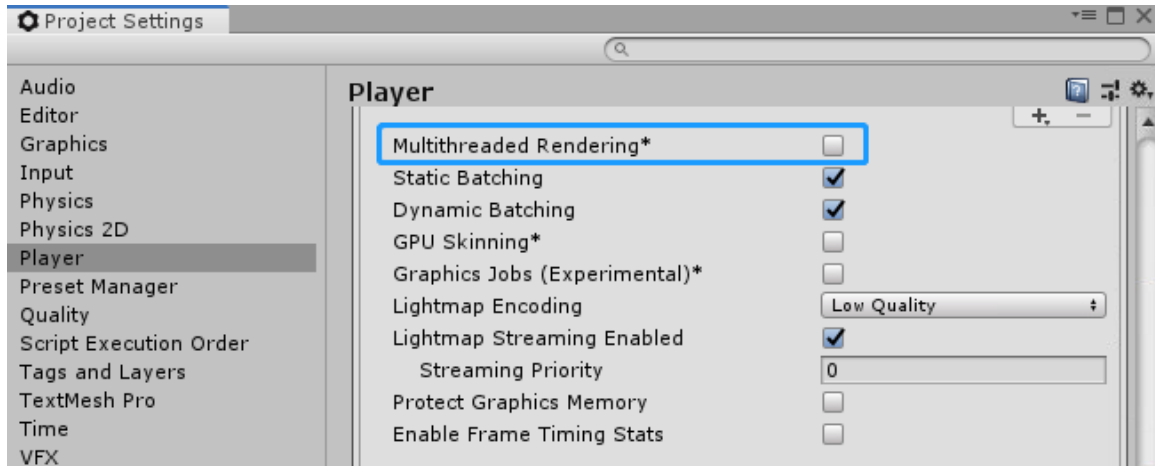


切换完成后，点击Player Settings，设置打包属性

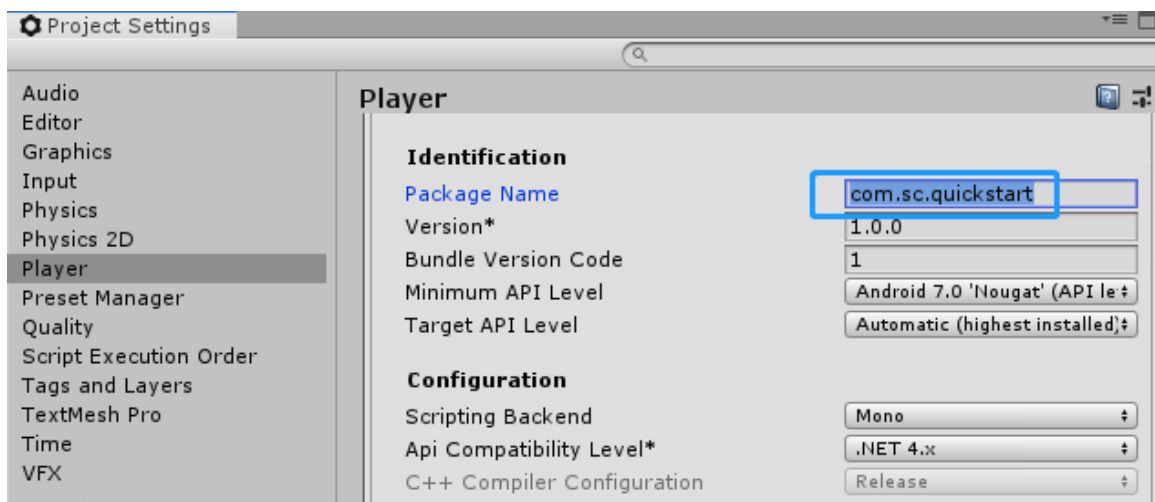
设置App名



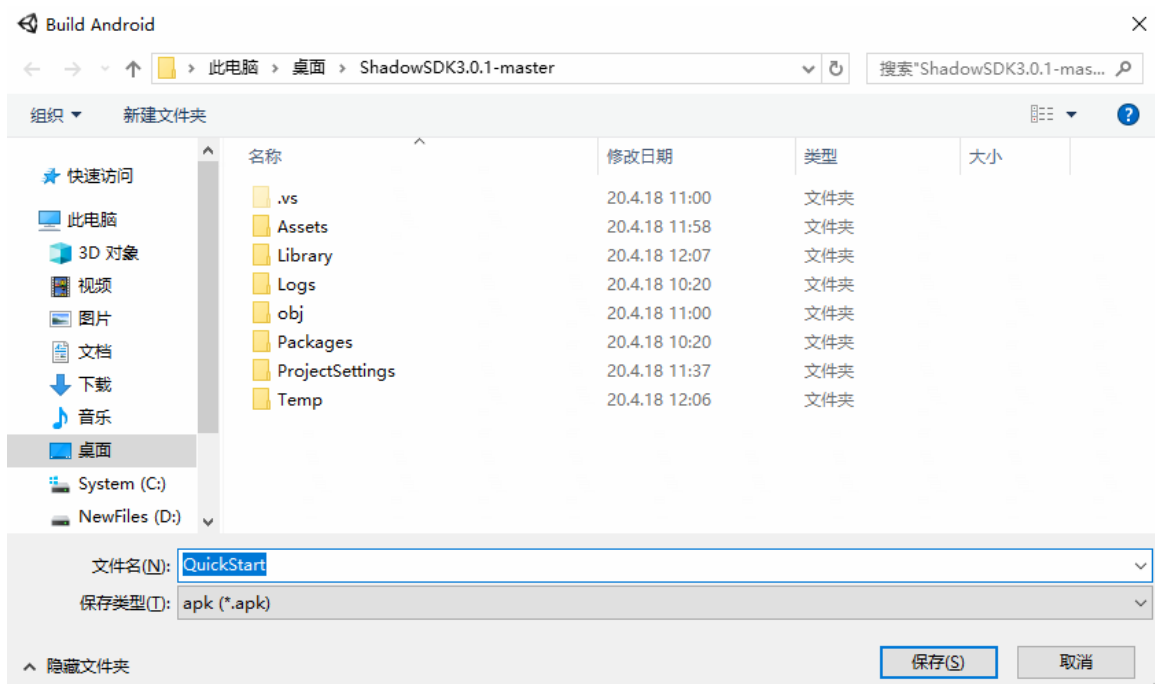
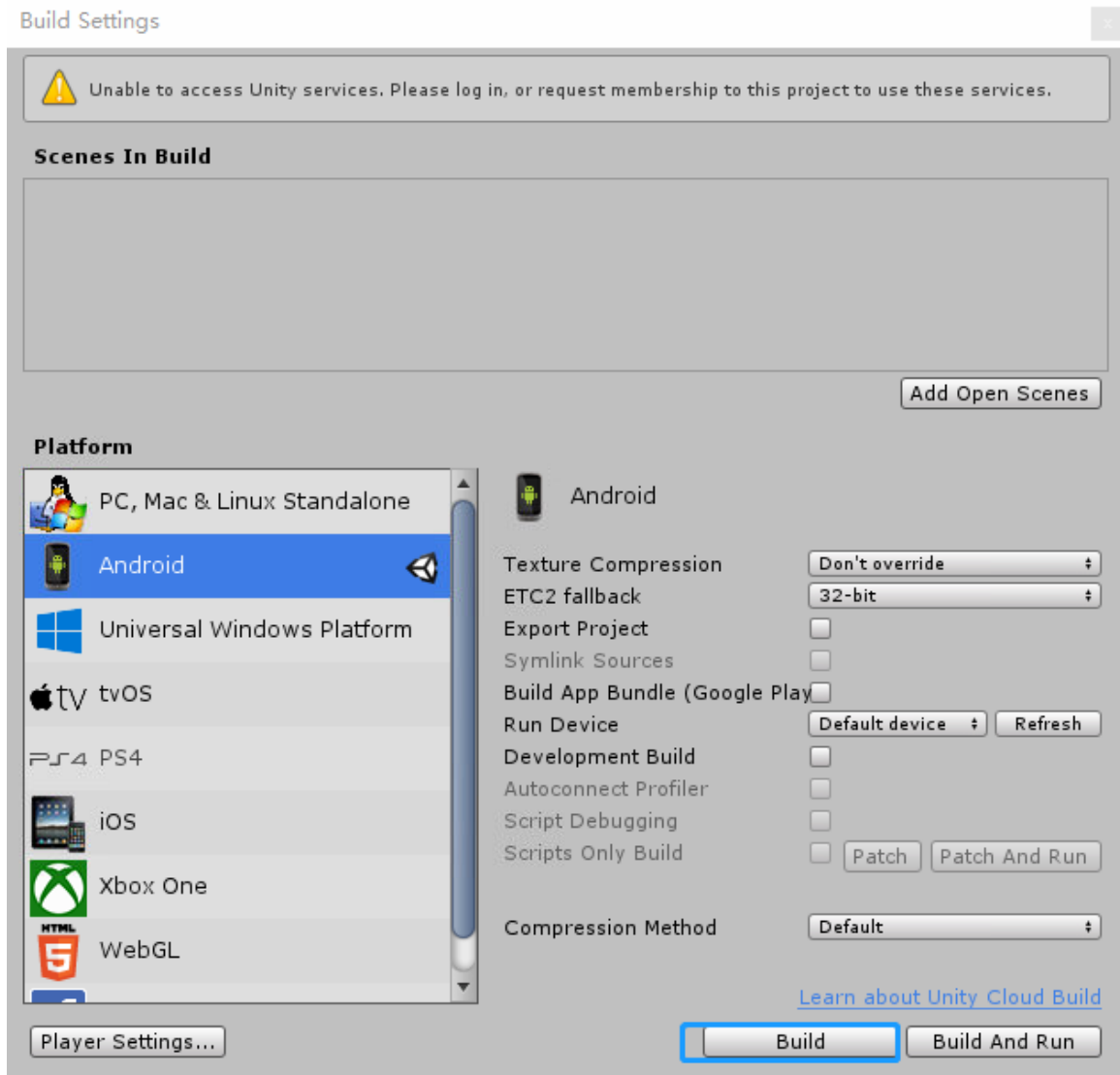
关闭多线程渲染(默认关闭)



设置APP包名



点击Build选择打包目录及Apk保存名（QuickStart.apk），打包Apk





QuickStart打包出后，安装到影创MR眼镜，即可查看眼镜中的效果

2.4 获取本例程项目源码

[点击下载本例程项目源码](#)

3 开发者文档

3.1 简介

开发者文档提供开发者一切所需资料

[点击进入开发者文档](#)