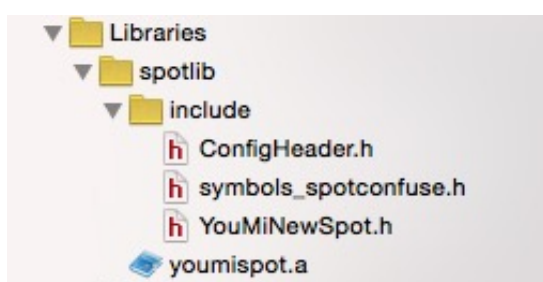


Unity3D项目嵌入YouMiSDK例子说明文档

本文档认为读者已经完成过起码一个Unity3D的项目

运行sdk中附带的unity3d的例子(YouMiUntiy3DSample)

1. 运行例子后,直接选导出xcode.
2. 在生成的xcode工程Libraries里添加上spotlib



3. 并添加所需的Framework

```
storekit.framework  
  
security.framework  
  
cfnetwork.framework  
  
systemconfiguration.framework  
  
ImageIO.framework  
  
libz.dylib  
  
libsqlite3.dylib
```

4. 在xcode工程中添加上UnityHandlerMiddle.h UnityHandlerMiddle.m(在例子中的readme文件夹有)



注意修改这几个值

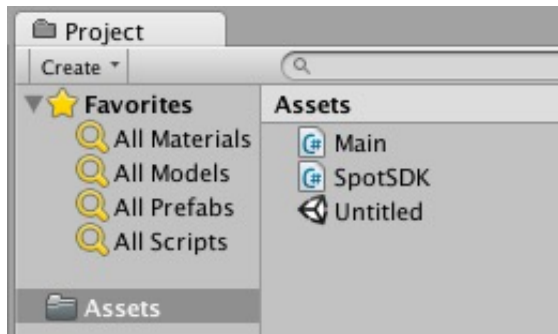
```
[YouMiNewSpot initYouMiDeveloperParams:@" APP_ID " YM_SecretId:@"  
APP_SECRETID "];//appid and appSecret
```

```
[YouMiNewSpot initYouMiDeveLoperSpot: SPOTTYPE ];//ad type
```

5. 编译运行xcode工程就可以看到两个按钮,一个初始化一个显示广告.上面都正常就可以看到广告正常显示.

要在自己的unity3d的工程中加入插屏的sdk,步骤如下:

1. 新建两个C#脚本Main, SpotSDK



2. Main用于显示按钮和响应事件。(具体可以按你的需要修改这个文件,或者不用这个文件)

```

using UnityEngine;
using System.Collections;

public class Main : MonoBehaviour {
    // Use this for initialization
    void Start () {

    }
    // Update is called once per frame
    void Update () {

    }

    void OnGUI() {
        if (GUI.Button(new Rect(20,44,150,60), "init wall")) {
            SpotSDK.InitSpot();
        }

        if (GUI.Button(new Rect(200, 44,150,60), "show wall")) {
            SpotSDK.ShowSpot();
        }

        GUI.Label (new Rect(100, 400, 320, 80), @"YouMi AD Unity3d
Sample");
    }
}

```

3. SpotSDK用于连接Native代码

```

using UnityEngine;
using System.Runtime.InteropServices;

public class SpotSDK : MonoBehaviour{

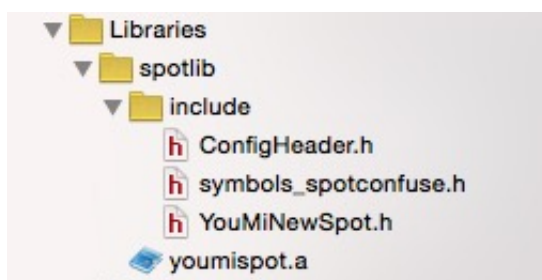
    [DllImport("__Internal")]
    private static extern void _initSpot();
    [DllImport("__Internal")]
    private static extern void _showSpot();

    public static void InitSpot()
    {
        if (Application.platform != RuntimePlatform.OSXEditor)
        {
            _initSpot();
        }
    }

    public static void ShowSpot()
    {
        if (Application.platform != RuntimePlatform.OSXEditor)
        {
            _showSpot();
        }
    }
}

```

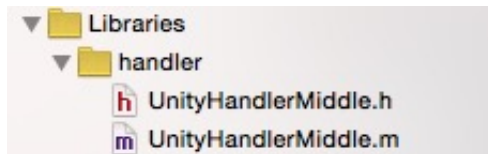
4. 选择build,导出xcode工程.
5. 在生成的xcode工程Libraries里添加上spotlib



6. 并添加所需的Framework

```
storekit.framework  
security.framework  
cfnetwork.framework  
systemconfiguration.framework  
ImageIO.framework  
libz.dylib  
libsqlite3.dylib
```

7. 在xcode工程中添加上UnityHandlerMiddle.h UnityHandlerMiddle.m(在例子中的readme文件夹有)



注意修改这几个值

```
[YouMiNewSpot initWithDeveloperParams:@" APP_ID " YM_SecretId:@"  
APP_SECRETID "]; //appid and appSecret
```

```
[YouMiNewSpot initWithDeveloperSpot: SPOTTTYPE ]; //ad type
```

- UnityHandlerMiddle.h

```
#import <Foundation/Foundation.h>  
  
@interface UnityHandlerMiddle : NSObject  
  
@end
```

- UnityHandlerMiddle.m

```
#import "UnityHandlerMiddle.h"  
#import "ConfigHeader.h"  
  
@implementation UnityHandlerMiddle  
  
+ (UnityHandlerMiddle *)sharedInstance {
```

```

        static UnityHandlerMiddle *instance = nil;
        static dispatch_once_t onceToken;
        dispatch_once(&onceToken, ^{
            instance = [UnityHandlerMiddle new];
        });
        return instance;
    }

- (id)init {
    self = [super init];
    if (self) {
        // **YouMi**
        // appid 设置
        [YouMiNewSpot initWithYouMiDeveloperParams:@"APP_ID"
        YM_SecretId:@"APP_SECRETID"]; //appid and appSecret

        [YouMiNewSpot initWithYouMiDeveloperSpot:SPOTTYPE]; //ad
type

        [YouMiNewSpot clickYouMiSpotAction:^(BOOL flag){
        }];

    }
    return self;
}

- (void)dealloc {
    [super dealloc];
}

void _initSpot() {
    [UnityHandlerMiddle sharedInstance];
    NSLog(@"初始化成功");
}

void _showSpot() {
    [YouMiNewSpot showYouMiSpotAction:^(BOOL flag){
        if (flag) {
            NSLog(@"Show success. Do success thing");
        }
        else{
            NSLog(@"Show error. Do error thing");
        }
    }];
}

@end

```

8. Run