**Huvle SDK Unity3D Install Guide(Unity 2022.10.18)**

**Common Requirements**

* Use above Unity 2020.3.25f1
* Above Google Play Service 20.5.0
* Target Android API 31

**Huvleview Install Guide ↓↓↓↓↓↓**

**Download HuvleSDK aar**

https://sdk.huvle.com/repository/internal/com/byappsoft/sap/HuvleSDK/6.0.1/HuvleSDK-6.0.1.aar

1. **Create HuvleSdkObject to Hierarchy**
   1. Click [+] on Hierarchy panel
   2. Click [Create Empty]

3. Change the name of created Object to HuvleSdkObject (No matter what the name is)

1. **Register Huvle SDK to Project**
   1. If ‘Plugins/Android’ is not existing below art of Assets, create the folder (Should distinguish name and uppercase lowercase actually)
   2. Drag and drop the downloaded file to Assets/Plugins/Android folder and register it (HuvleSDK-6.0.1.aar is the file)
2. **Create Huvle Script to Project**
   1. Click [+] on Project panel
   2. Click [C# Script]
   3. Change the name of created Script to HuvleSdkController (No matter what the name is)
3. **Apply to Huvle Script**
   1. Open the HuvleSdkController.cs file, which was created in 3, with an editor and code it as follows.

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

using System;

public class HuvleSdkController : MonoBehaviour

{

void OnApplicationFocus(bool hasFocus) {

if (hasFocus) {

CallHuvleOnResume();

}

}

private void CallHuvleOnResume() {

try {

AndroidJavaClass unity\_player = new AndroidJavaClass("com.unity3d.player.UnityPlayer");

AndroidJavaObject activity = unity\_player.GetStatic<AndroidJavaObject>("currentActivity");

AndroidJavaClass share\_plugin = new AndroidJavaClass("com.byappsoft.sap.UnityInterface");

activity.Call("runOnUiThread", new AndroidJavaRunnable(() => {

share\_plugin.CallStatic("onResume", activity);

}));

} catch (Exception ex) {

Debug.Log("CallHuvleOnResume error:" + ex.Message);

}

}

}

1. **Connect Huvle Script to Huvle Object**
   1. Drag Huvle Script created in 4 to Huvle Object created in 1
2. **Unity Gradle Template**
   1. [File]->[Build Settings]->[Player Settings…] and click the menu
   2. Open the Art [Publishing Settings] from the Art [Player]
   3. Check the art of [Custom Main Gradle Template] , [Custom Gradle Properties Template] on [Build] category.
   4. Choose [Other settings] **Minimum API Level 4.4**, **Target API Level android12(API Level31)**

텍스트이(가) 표시된 사진

자동 생성된 설명

1. **Revise Custom Main Gradle Template** 
   1. When you create Unity Gradle Template, mainTemplate file is created in Assets/Plugins/Android folder of Project panel.
   2. Open mainTemplate file with an editor and add it to dependencies as follows. (appcompat, support-v4, play-services-ads)

apply plugin: 'com.android.library'

\*\*APPLY\_PLUGINS\*\*

dependencies {

implementation fileTree(dir: 'libs', include: ['\*.jar'])

implementation 'com.android.support:appcompat-v7:28.0.0'

implementation 'com.android.support:support-v4:28.0.0'

implementation 'com.google.android.gms:play-services-ads:20.5.0'

\*\*DEPS\*\*}

android {

compileSdkVersion \*\*APIVERSION\*\*

buildToolsVersion '\*\*BUILDTOOLS\*\*'

compileOptions {

sourceCompatibility JavaVersion.VERSION\_1\_8

targetCompatibility JavaVersion.VERSION\_1\_8

}

defaultConfig {

minSdkVersion \*\*MINSDKVERSION\*\*

targetSdkVersion \*\*TARGETSDKVERSION\*\*

ndk {

abiFilters \*\*ABIFILTERS\*\*

}

versionCode \*\*VERSIONCODE\*\*

versionName '\*\*VERSIONNAME\*\*'

consumerProguardFiles 'proguard-unity.txt'\*\*USER\_PROGUARD\*\*

}

lintOptions {

abortOnError false

}

aaptOptions {

ignoreAssetsPattern = "!.svn:!.git:!.ds\_store:!\*.scc:.\*:!CVS:!thumbs.db:!picasa.ini:!\*~"

}\*\*PACKAGING\_OPTIONS\*\*

}\*\*REPOSITORIES\*\*\*\*SOURCE\_BUILD\_SETUP\*\*

\*\*EXTERNAL\_SOURCES\*\*

1. **Revise Custom Gradle Properties Template (For androidX)**
   1. gradleTemplate file is created in Assets/Plugins/Android folder of Project panel.
   2. Open GradleTemplate with editor and add it as follows.

org.gradle.jvmargs=-Xmx\*\*JVM\_HEAP\_SIZE\*\*M

org.gradle.parallel=true

android.enableJetifier=true

android.useAndroidX=true

android.enableR8=\*\*MINIFY\_WITH\_R\_EIGHT\*\*

unityStreamingAssets=.unity3d\*\*STREAMING\_ASSETS\*\*

\*\*ADDITIONAL\_PROPERTIES\*\*

1. **Notice**

* When you implement and exit your app after integration of HuvleSDK with Unity, Notification-bar will be re-shown up. Please refer to this Notice.
* Since this is based on the version after Unity 2020.3.25f1, integration way with the other below versions could be different. If you have any issue to integrate, please don’t hesitate to contact us.