Image And Video Picker

ImageAndVideoPicker available for android and iOS, this plugin is very useful for developer who are developing application or game, by using this plugin you can browse image, video from gallery, you don't need to write native code for that.

Below is the integration guide and functions details for you.

Integration : Follow the given step to integrate plugin in your project:

1). Import “ImageAndVideoPicker" unitypackage into your project

2). Configure **AndroidManifest.xml** for Android

Add this to main activity

```
<meta-data android:name="unityplayer.SkipPermissionsDialog" android:value="true" />
```

Add this line for a activity

```
<activity
  android:name="com.astricstore.imageandvideopicker.AndroidPickerActivity"
    android:configChanges="orientation|keyboardHidden|screenSize">
    </activity>
     <activity
    android:name="eu.janmuller.android.simplecropimage.CropImage"
    android:configChanges="orientation|keyboardHidden|screenSize">
  </activity>
```

Notes:

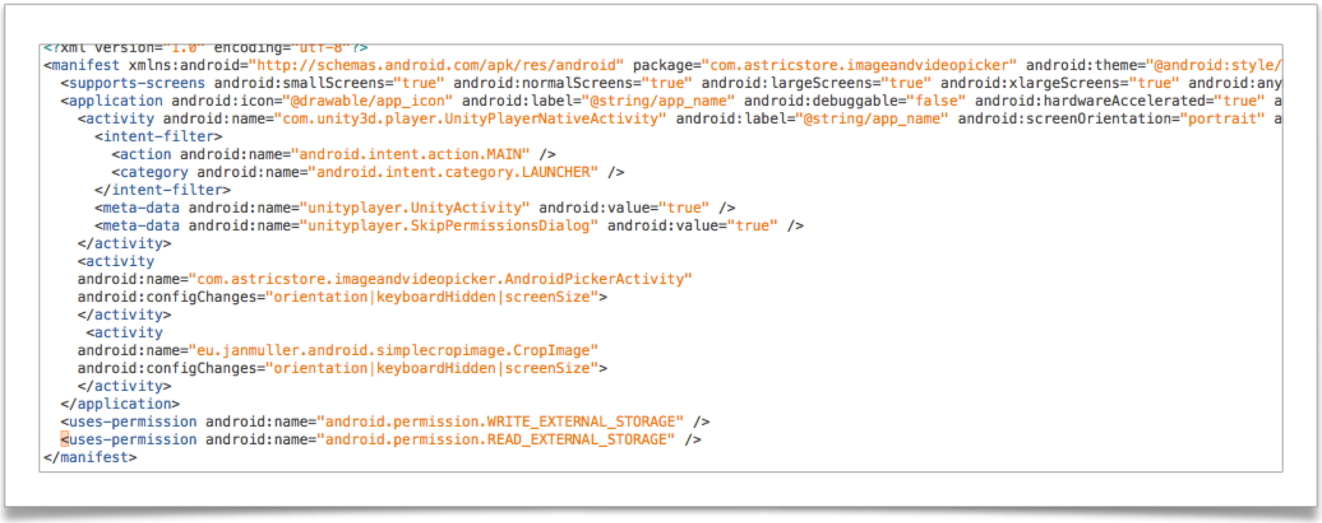
1). You can use provided AndroidManifest.xml file in your project, you have to change bundle identifier as per your projects identifier.

2). Also you can generate AndroidManifest.xml by following these steps:

- Simply build your project for android after setting all parameters in player setting.

- You will get your AndroidManifest.xml file in **YourProject \ Temp \ StagingArea \ AndroidManifest.xml**

Add require two activities in this xml in proper place (after any </activity> tag).



```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android" package="com.astricstore.imageandvideopicker" android:theme="@android:style/
  <supports-screens android:smallScreens="true" android:normalScreens="true" android:largeScreens="true" android:xlargeScreens="true" android:any
  <application android:icon="@drawable/app_icon" android:label="@string/app_name" android:debuggable="false" android:hardwareAccelerated="true" a
    <activity android:name="com.unity3d.player.UnityPlayerNativeActivity" android:label="@string/app_name" android:screenOrientation="portrait" a
      <intent-filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
      </intent-filter>
      <meta-data android:name="unityplayer.UnityActivity" android:value="true" />
      <meta-data android:name="unityplayer.SkipPermissionsDialog" android:value="true" />
    </activity>
    <activity
    android:name="com.astricstore.imageandvideopicker.AndroidPickerActivity"
    android:configChanges="orientation|keyboardHidden|screenSize">
    </activity>
     <activity
    android:name="eu.janmuller.android.simplecropimage.CropImage"
    android:configChanges="orientation|keyboardHidden|screenSize">
    </activity>
  </application>
  <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
  <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
</manifest>
```

Photo : Example of AndroidManifest.xml

3.1) Add framework in xCode for iOS

- For Unity4 (or below) users

Build your project and make xCode project, follow these steps

- Select your project in xCode hierarchy
- Goto Build Phase
- Click add(+) button in “Link Binary With Libraries”
- Search MobileCoreServices.framework and add
- Search AssetsLibrary.framework and add

- For Unity5 or above users

- Select libImageAndVideoPicker.a in Plugins/iOS/ folder
- Check(tick) “MobileCoreServices” from framework dependencies in inspector.
- Check(tick) “AssetsLibrary” from Rarely used framework

3.2) Mark DLL for iOS

Select ImageAndVideoPicker_iOS.dll in Plugins/ folder and mark it for Editor and iOS in inspector.

4). Drag **PickerEventListener.prefab** into your Hierarchy.

5). Use ImageAndVideoPicker package into your script by writing

following line on the top of your script

```
using ImageAndVideoPicker;
```

6). Plugin is now ready to use, you can simply call function provided in plugin.

# Check for permissions : Add this code to Start() or OnEnable() function to check permission before picking image/video from gallery (for Android only).

```
#if UNITY_ANDROID
    AndroidPicker.CheckPermissions();
#endif
```

# Use these function when you want to browse image from gallery.

Android : AndroidPicker.BrowseImage();

iOS : IOSPicker.BrowseImage();

# Use these function when you want to browse and crop image from gallery.

Android : AndroidPicker.BrowseImage(bool cropping, int aspectX, int aspectY);

iOS : IOSPicker.BrowseImage(bool cropping);

# Use this function when you want to browse video from gallery.

Android : AndroidPicker.BrowseVideo();
IOS : IOSPicker.BrowseVideo();

Note: iOS compress video so it will take some to compress.

7). Event will be fired after successful/un-successful operation.

<u>Add Events like:</u>

# PickerEventListener.onImageSelect += OnImageSelect;

```
void OnImageSelect(string imgPath,ImageAndVideoPicker.ImageOrientation orientation)
{
	// imgPath : image path
}
```

# PickerEventListener.onImageLoad += OnImageLoad;

```
void OnImageLoad(string imgPath, Texture2D tex, ImageAndVideoPicker.ImageOrientation orientation )
{
	// imgPath : browsed image path
	// tex : image texture
}
```

# PickerEventListener.onVideoSelect += OnVideoSelect

```
void OnVideoSelect(string vidPath)
{
        // vidPath : video path
}
```

# PickerEventListener.onError += OnError;

```
void OnError(string errorMsg)
{
        // errorMsg : error message
}
```

# PickerEventListener.onCancel += OnCancel;

```
void OnCancel()
{
}
```

Remove Events like:

```
PickerEventListener.onImageSelect -= OnImageSelect;
PickerEventListener.onImageLoad -= OnImageLoad;
PickerEventListener.onVideoSelect -= OnVideoSelect;
PickerEventListener.onError -= OnError;
PickerEventListener.onCancel -= OnCancel;
```

8). **ImageAndVideoPicker.ImageOrientation :** Here is description about orientation of the image. Icon representing the orientation of the image. This will be very helpful to rotate image in current orientation.

```
public enum ImageOrientation
{

     UP = 0,

     DOWN = 1 ,   // 180 deg rotation

     LEFT  = 2,   // 90 deg Clock Wise

     RIGHT = 3,   // 90 deg Counter Clock Wise

     UP_MIRRORED = 4,    // as UP but image mirrored along

     DOWN_MIRRORED = 5,  // horizontal flip

     LEFT_MIRRORED = 6,  // vertical flip

     RIGHT_MIRRORED = 7, // vertical flip
}
```

For support email me at devesh.pandey19@gmail.com

Leave your comment if you have any problem or suggestions.

http://unitydevelopers.blogspot.com/2015/05/image-video-and-contact-picker-image.html