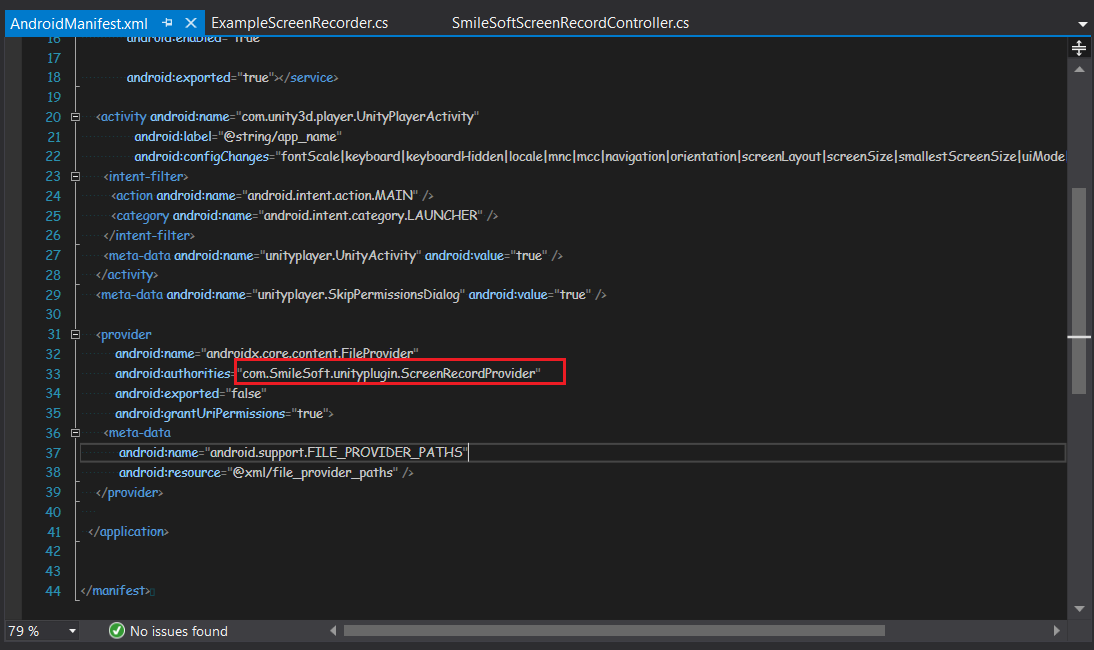
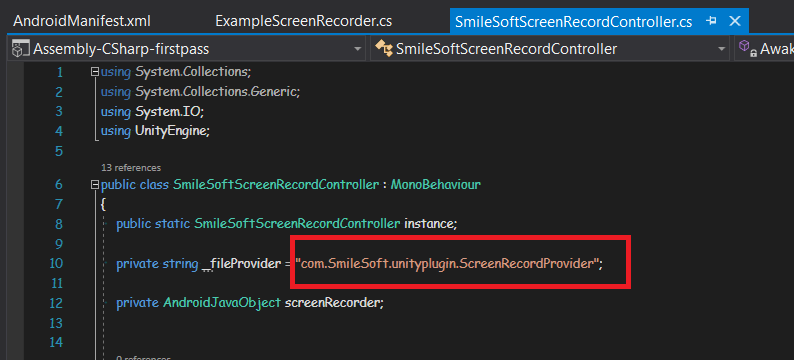
**SunShine Android Native Screen Recorder**

This plugin can record your gameplay without any issue. It uses android native **MediaRecorder** API and **Android Foreground Service** for record screen. So it will support from **Lolipop (5.1)** to newer versions even **android 10.**

**Instruction:** After importing the project please drag **Screen Recorder** prefab in Scene Hierarchy from **Assets > SunShine Android Native Screen Recorder > Prefab > Screen Recorder**.

If you want to share the video using native share dialog then please do the following settings.

1. Open Android manifest file from **Assets > Plugins > Android > AndroidManifest.xml** . Now change the authority name with some unique one. We highly recommend you to use the package name.
2. Again Open **SmileSoftScreenRecordController.cs** from **Assets > SunShine Android Native Screen Recorder > Scripts > SmileSoftScreenRecordController.cs**. Now change the \_fileProvider value which you set on the manifest file before.



You can also see this [video](https://www.youtube.com/watch?v=GuCw5plwxtI&feature=youtu.be&ab_channel=SmileSoft) For setting the file provider path value.

Now you are ready for calling the plugin API. The following functions are available in this plugin.

**Functions:**

1. **Start Record:** **SmileSoftScreenRecordController.instance.StartRecording()**
2. **Stop Record:**  **SmileSoftScreenRecordController.instance.StopRecording()** . It returns the recorded file path.
3. **Record Audio: SmileSoftScreenRecordController.instance.SetAudioCapabilities(isAudioRecording)**.

Where **isAudioRecording** is a **Boolean** type parameter. This will record audio from the device mic.

1. **Set File Destination:** **SmileSoftScreenRecordController.instance.SetVideoDestination(destination)**. Where **destination** is a **String** type value. By default the destination is **External Storage** directory.
2. **Set File Name:** **SmileSoftScreenRecordController.instance.SetVideoName(fileName)**. Where **fileName** is a **String** type value.
3. **Folder Name:** You can create a new folder and save your video there. Just call **SmileSoftScreenRecordController.instance.SetStoredFolderName(folderName)** function. Here **folderName** is a **String** type value.
4. **Set Gallery Add Capabilities:** **SmileSoftScreenRecordController.instance.SetGalleryAddingCapabilities(canAddIntoGallery)**. Where **canAddintoGallery** is a **boolean** type value.If it is true then video file will be added into device gallery.By default this value is **true.**
5. **Bit Rate:** **SmileSoftScreenRecordController.instance.SetBitRate(bitrate)**. **bitrate** is an **integer** type parameter. Higher bitrate means high quality video.
6. **Preview:** If you want to see the preview in the native video view then just call this function **SmileSoftScreenRecordController.instance.PreviewVideo(\_recordedFilePath)**.Here **recordedFilePath** is an **string** type parameter.
7. **Share Video:** You can also Share the video using native share dialog. Just call **SmileSoftScreenRecordController.instance.ShareVideo(\_recordedFilePath, message, share title)**. It takes three parameters and all of them are **string**. First one is the **filePath**, second one is the **share message** and third one is **share title**.
8. **Video Size:** **SmileSoftScreenRecordController.instance.SetVideoSize(width,height)**. Both **width** and **height** are integer type parameter.
9. **Video Encoder:** **SmileSoftScreenRecordController.instance.SetVideoEncoder(videoEncoder)**. Here **videoEncoder** is an integer type parameter. Before setting this please look [Android Developer official](https://developer.android.com/reference/android/media/MediaRecorder.VideoEncoder) site for supporting encoders in different Android API versions.

To get a clear idea please see the example scene from **Assets > SunShine Android Native Screen Recorder > Example > Example Scene.**

**\*\* Here 3-9 number functions are optional. You can use all of them or none. But please be sure that these (3-9) functions are called before calling the Start Record function. \*\***