String™

Augmented Reality SDK for iOS Unity

String for iOS Unity Tutorial

Table of Contents

This is a step-by-step guide taking you through replicating, from scratch, the Unity tutorial project that comes with the String™ SDK. It requires basic knowledge of Unity and Xcode.

1. Prerequisites

- Register as an iOS developer in the iOS Dev Center.
- Install the latest Xcode and iOS SDK.
- Obtain and install a valid development provisioning profile from the iOS Dev Center which includes all the devices you're planning on using.
- Install the latest Unity with a valid Unity iOS license. You need a Pro license to use String's in-editor preview tool.
- If you're planning on using String's preview tool, please install Unity Remote for iOS.
- Download (if necessary) the String™ SDK and unzip it.

2. Getting started

- Open Unity.
- In the Project Wizard, select the Create new Project tab.
- Choose a folder and press Create Project.
- In the Project view, create folders Plugins, Scripts, StreamingAssets, Materials and Scenes. Please make sure the Plugins and StreamingAssets folders are spelled correctly and placed in your project root, as these folders have special meaning in Unity.
- Drag and drop String.bundle from the Libraries folder in the String™ SDK to the Plugins folder in your project.
- Drag and drop StringWrapper.cs and CameraCentricARManager.cs from the Assets folder in the String™ SDK to the Scripts folder in your project.
- Drag and drop Marker 1.png, Marker 2.png and Marker 3.png from the Assets folder in the String™ SDK to the StreamingAssets folder in your project.
- Select File → Build Settings.
- Select iOS and press Switch Platform.
- Close the Build Settings window.

3. Creating a directional light

- In the Unity menu, select GameObject → Create Other → Directional Light.
- Drag and drop it onto Main Camera.

4. Creating materials

- Right-click the Materials folder in your Project view, and select Create → Material.
- Rename the new material Red.
- Click the material's Main Color and select a red color.
- Repeat the above three steps with blue and green.

5. Creating objects

- In the Unity menu, select GameObject → Create Empty.
- Rename the object to Root Object 1.
- Select GameObject → Create Other → Cube.
- Drag and drop Cube onto Root Object 1.
- Make sure Cube's rotation is 0, 0 and 0.
- Set its position to 0, 0 and 0.2.
- Set its scale to 0.4, 0.4 and 0.4.
- Right-click Root Object 1 and select Duplicate.
- Rename the new object Root Object 2.
- Duplicate Root Object 1 again, and rename the new object to Root Object 3.
- Select the Cube attached to Root Object 1.
- In the Inspector view, expand the Mesh Renderer's Materials array.
- For Element 0, select Red.
- Repeat the above three steps with Root Object 2 and Root Object 3, setting their materials to Green and Blue, respectively.

6. Setting up CameraCentricARManager

- Drag and drop the CameraCentricARManager script onto Main Camera.
- Select Main Camera.
- In the Inspector view, under CameraCentricARManager, expand the Root Objects array.
- Set its size to 3.
- Set its elements to Root Object 1, 2 and 3.
- Press cmd+S to save the scene. Save it as Main in the Scenes folder.

7. Running in the editor

- If you have a Pro license, and have a camera attached, press cmd+P to run.
- You should now see a video feed in your Game view, with a String overlay.
- If you have a Pro license and a camera, and the video feed fails to appear, please select Window → Console and look for errors in the log.
- Print out one or more of the image markers in Tutorial Image Targets.pdf in the Assets folder of the String™ SDK.
- Hold the printed image marker in front of your camera.

8. Building for an iOS device

- Select Edit → Project Settings → Player.
- In the Inspector view, select Other Settings.
- Set Bundle Identifier to com.yourcompany.YourApp. If you're using an Indie, Pro or Campaign build of the String™ SDK, this needs to match the bundle ID you registered with the String Developer Portal.
- Set Target Device to iPhone + iPad.
- Set Target Platform to Universal armv6+armv7.
- Set SDK Version to iPhone iOS latest.
- Press cmd+B and click the Build button.
- Choose a folder for your build and press Save.

9. Setting up your Xcode project

- In Xcode, stop the build.
- In the Project navigator, select Unity-iPhone → target Unity-iPhone → Build Phases tab, and expand Link Binary With Libraries.
- Click the plus button, then press and hold cmd and select the AVFoundation, CoreGraphics, CoreMedia and CoreVideo frameworks.
- Click Add.
- Right-click Unity-iPhone and select Add Files to "Unity-iPhone".
- Navigate to the unzipped String™ SDK and select libStringUnity*.a from the Libraries folder.
- Click Add.
- Make sure the active scheme is iOS Device.
- Press cmd+B to build.
- If you get any errors, please make sure you went through all the steps in this section.
- Attach an iOS device with a camera that's enabled for development. See the iOS Dev Center for more.
- Press cmd+R to run.
- After the app has loaded, point your device's camera at one of the printed markers.