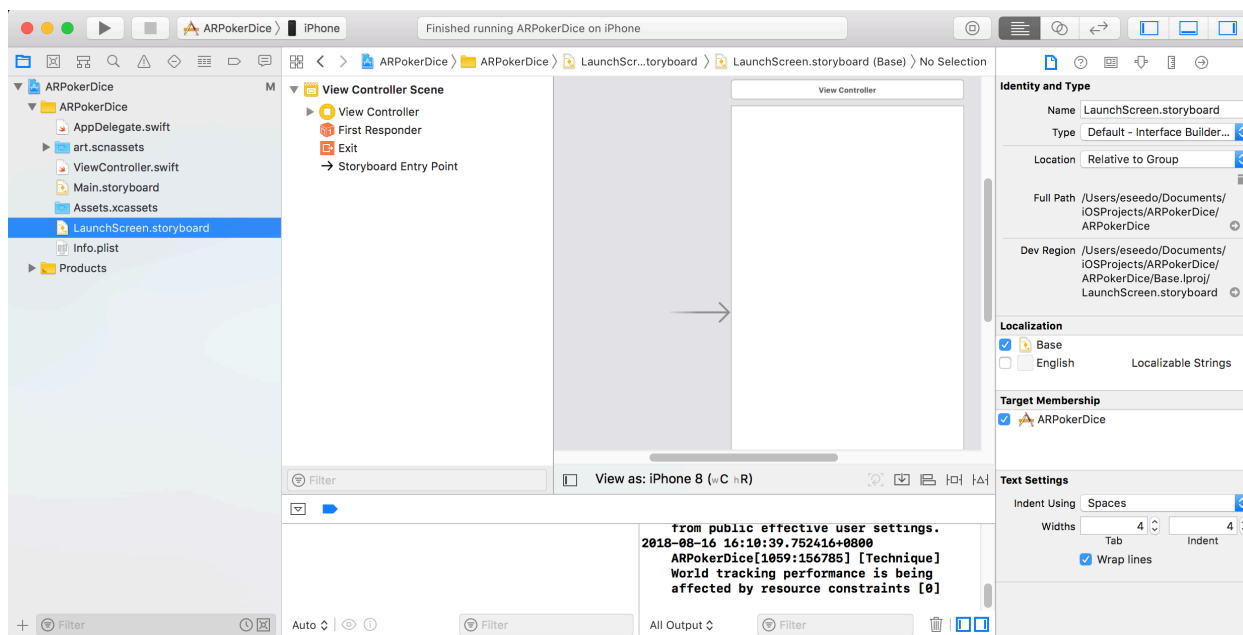


在继续进行其它操作之前，我们需要查看一下Xcode自动帮我们创建的项目文件。

1. LaunchScreen.storyboard

启动界面是每个应用都具备的标准部分。当用户打开应用的时候，首先看到的就是启动界面。

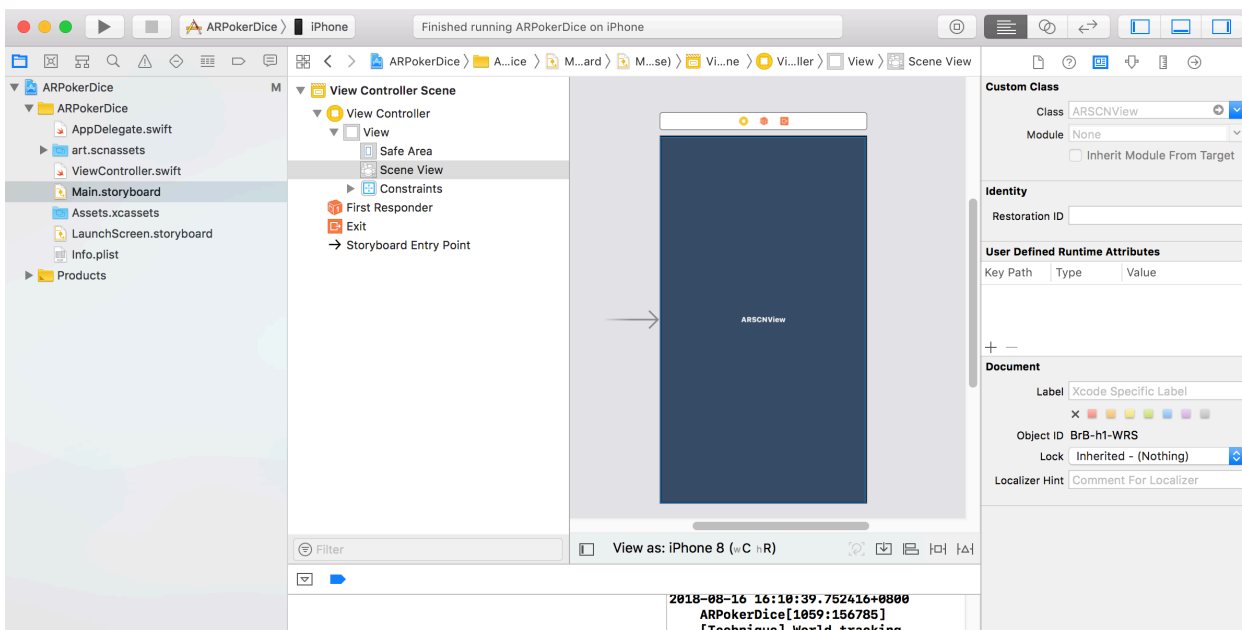
选中LaunchScreen.storyboard并将其打开。



在这里，我们可以放一个精美的启动图片作为AR应用的启动界面。

2. Main.storyboard

Main.storyboard是AR应用的视图部分，当应用运行的时候，这里也是用户看到的界面。



需要特别注意的是这里的ARSCNView 视图类，它可以让我们把3D的场景叠加在摄像头所获取的实时场景图片上。ARSCNView可以将ARKit和SceneKit无缝接合在一起。

注意：

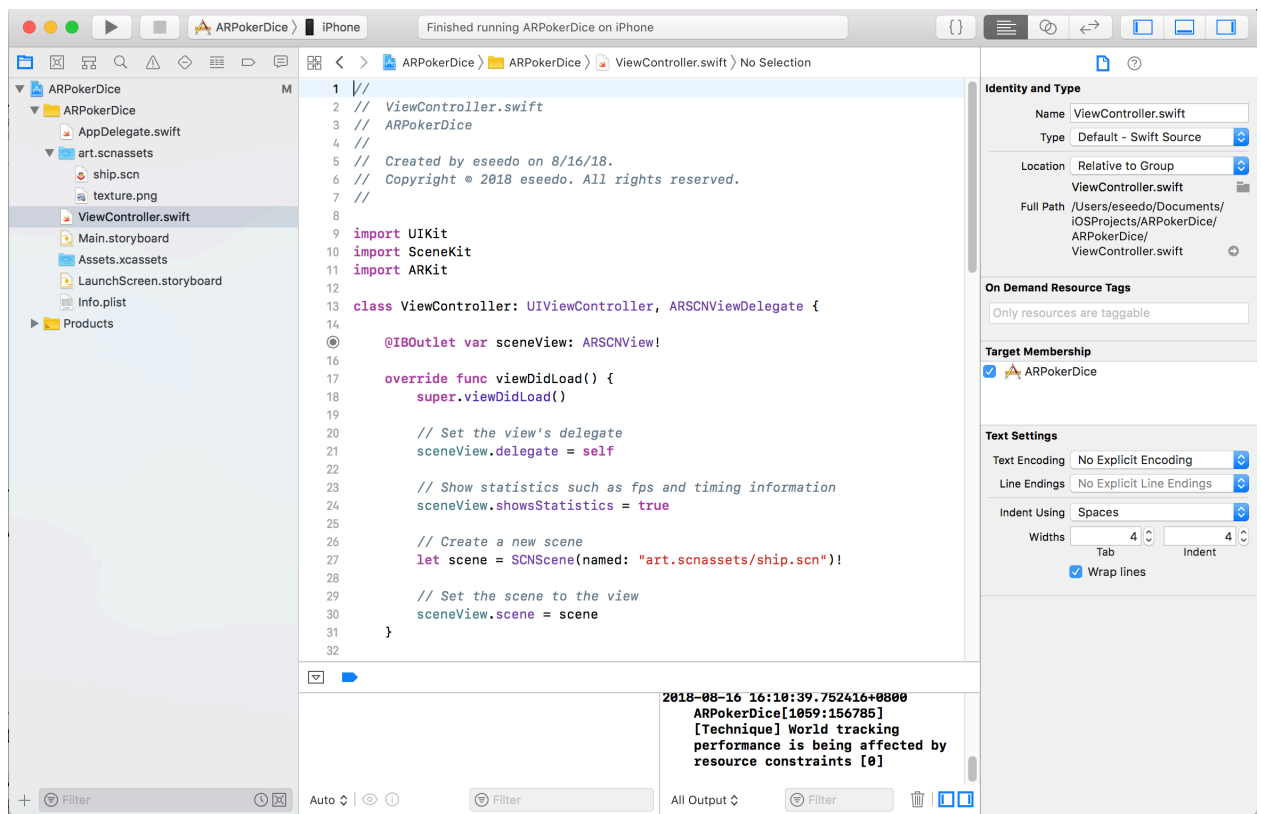
除了ARSCNView视图类，还有另外一个视图类也可以实现ARKit和SpriteKit的整合：

ARSKView。如果我们希望叠加2D的基于SpriteKit的内容，而不是3D的和SceneKit相关的内容，就可以考虑使用ARSKView。

3.ViewController.swift

可以把视图控制器类看做main storyboard的幕后指挥。

选择ViewController.swift文件，将其打开。



首先我们注意到import的三个重要框架：

- UIKit
- SceneKit
- ARKit

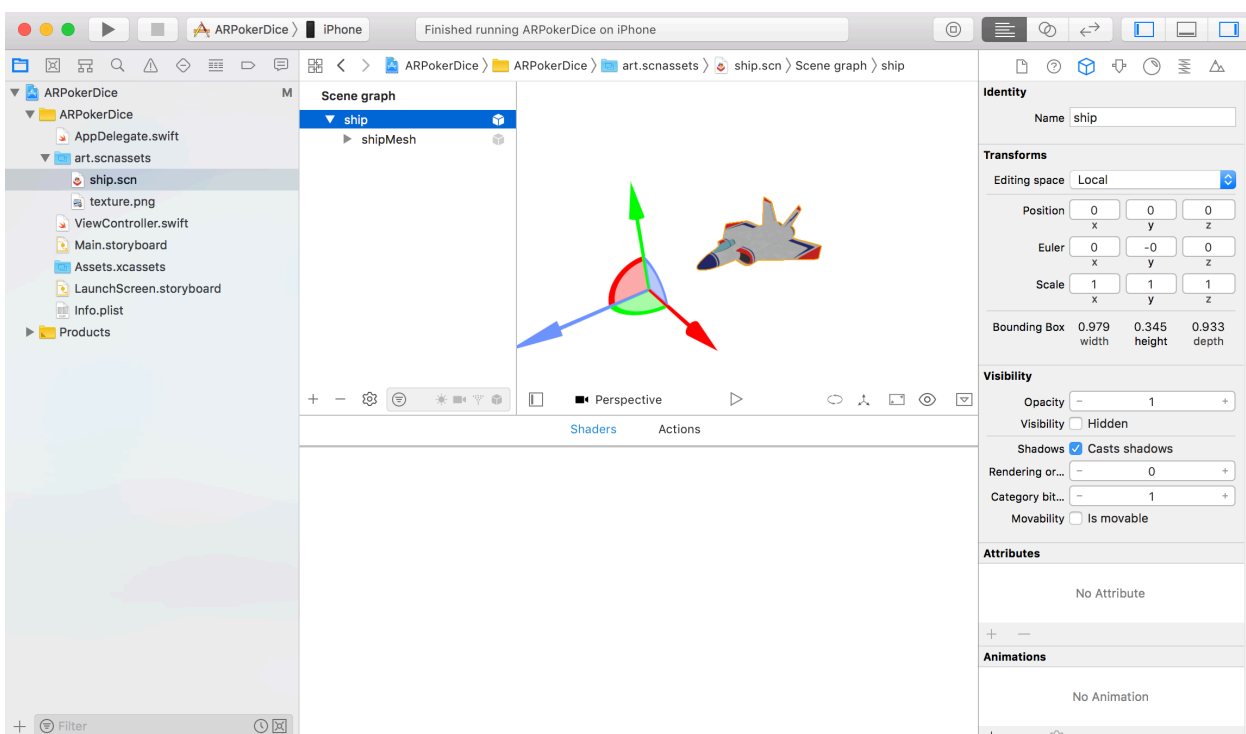
接下来可以看到，这里的ViewController类直接继承自标准的UIViewController，而标准的UIViewController则提供了用于管理基于UIKit应用的视图。

除此之外，这里的ViewController还遵循ARKit中提供的ARSCNViewDelegate协议。该协议中提供了若干方法，可以让我们将SceneKit中的内容和AR session同步。

4.SceneKit asset 目录

在项目文件中有一个名为art.scnassets的目录，这是一个为SceneKit资源特殊设计的目录。SceneKit是一个表现稳健的高等图像处理框架，可以用来创建和管理应用中的所有图像和音频内容。我们也将在这个文件夹中保存各种所需的资源，包括场景、纹理、3D模型、动画、音效和音乐等等。

双击art.scnassets/ship.scn，你可以看到之前编译成功后在手机屏幕上出现的酷炫的太空飞船。 .scn文件其实就是一个scene（场景），而这里我们看到的就是Xcode中内置的SceneKit编辑器。



除了ship.scn文件，在这个目录下还有一个名为texture.png的文件。它是飞船3D模型所对应的纹理贴图文件，关于这一点我们在后面再说。

5.Info.plist文件

在我们第一次编译运行应用的时候，首先看到一个提示，也即是否允许访问相机。对所有的ARKit应用来说，都必须获取设备相机的访问权限。否则的话ARKit就无法完成任何任务。我们可以在Info.plist配置文件中看到特定的设置。

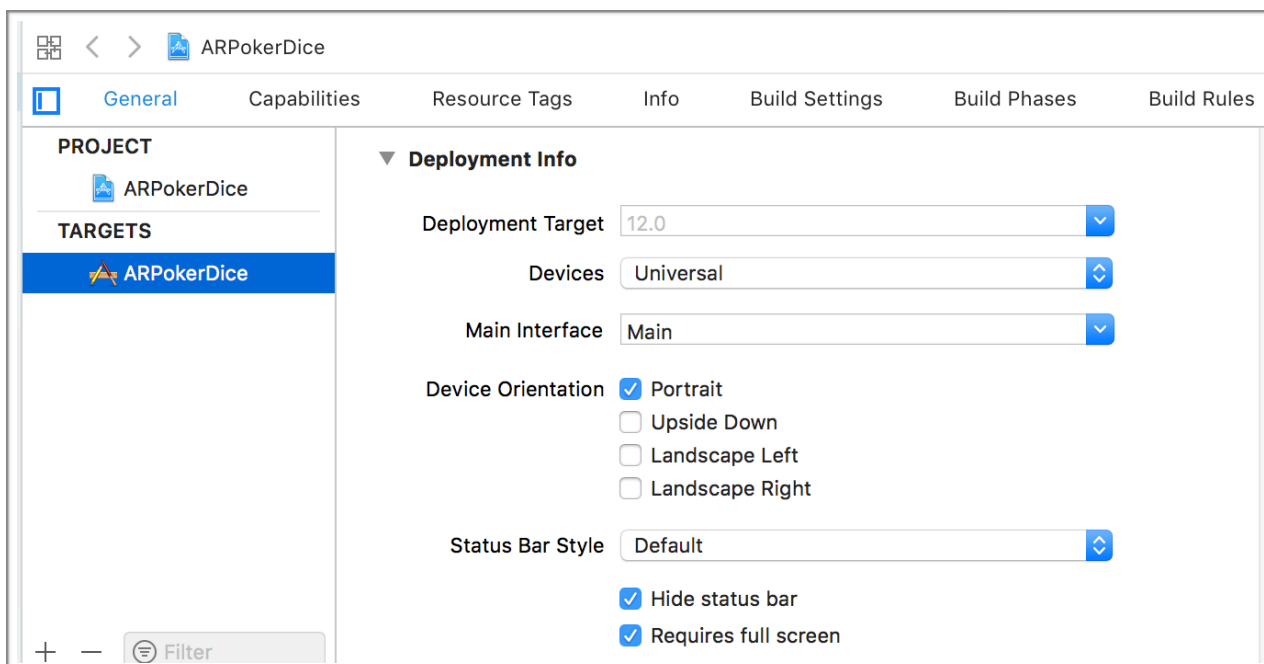
在Key这一栏下面找到Privacy - Camera Usage Description，然后可以把这个信息修改为你想要的。这个信息用在哪里呢？当打开应用，并向用户请求相机的访问权限时，就会显示这条信息。

| AR Poker Dice > AR Poker Dice > Info.plist > No Selection | | |
|---|------------|--|
| Key | Type | Value |
| ▼ Information Property List | Dictionary | (17 items) |
| Localization native development re... | String | \$(DEVELOPMENT_LANGUAGE) |
| Executable file | String | \$(EXECUTABLE_NAME) |
| Bundle identifier | String | \$(PRODUCT_BUNDLE_IDENTIFIER) |
| InfoDictionary version | String | 6.0 |
| Bundle name | String | \$(PRODUCT_NAME) |
| Bundle OS Type code | String | APPL |
| Bundle versions string, short | String | 1.0 |
| Bundle version | String | 1 |
| Application requires iPhone enviro... | Boolean | YES |
| Privacy - Camera Usage Description | String | We need to use the camera for Augmented Reality. |
| Launch screen interface file base... | String | LaunchScreen |
| Main storyboard file base name | String | Main |
| ► Required device capabilities | Array | (2 items) |
| UIRequiresFullScreen | Boolean | YES |
| Status bar is initially hidden | Boolean | YES |
| ► Supported interface orientations | Array | (1 item) |
| ► Supported interface orientations (i... | Array | (4 items) |

注意：当我们使用ARKit模板创建应用时，Xcode会自动帮我们添加一个提示信息设置。但是如果使用其它模板从零开始创建了一个ARKit项目，那么就必须在Info.plist中手动添加相关的信息。

项目设置

和其它iOS应用一样，我们也可以对项目进行个性化的设置。



从项目导航视图中选中AR Poker Dice，然后在TARGETS-AR Poker Dice-Development Info中，可以和其它iOS应用一样进行设置：

对我们这款应用来说，需要勾选Hide status bar和Requires full screen。

通过这样的设置，所创建的ARKit应用会一直以portrait mode（竖屏模式）显示，而这正是我们所需要的效果。