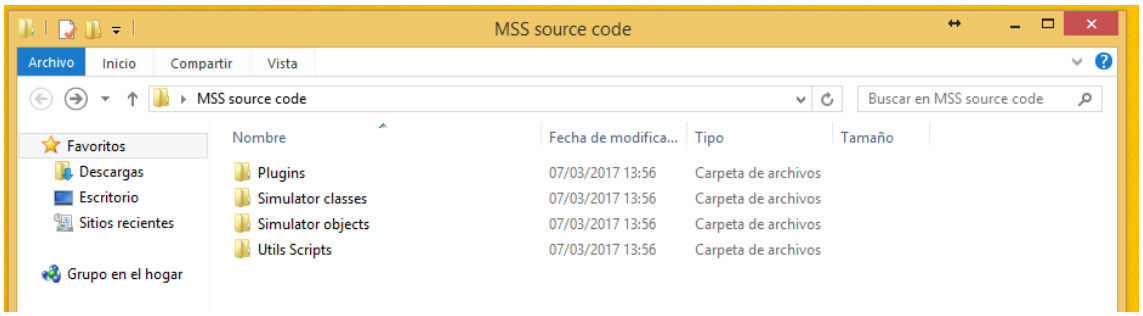
Multi Camera System Simulator (MSS)

Tutorial: Configure the Multi Camera System Simulator in a Unity Project.

In this document we explain how to configure the simulator in a new or an existing Unity project. To follow this tutorial, you need:

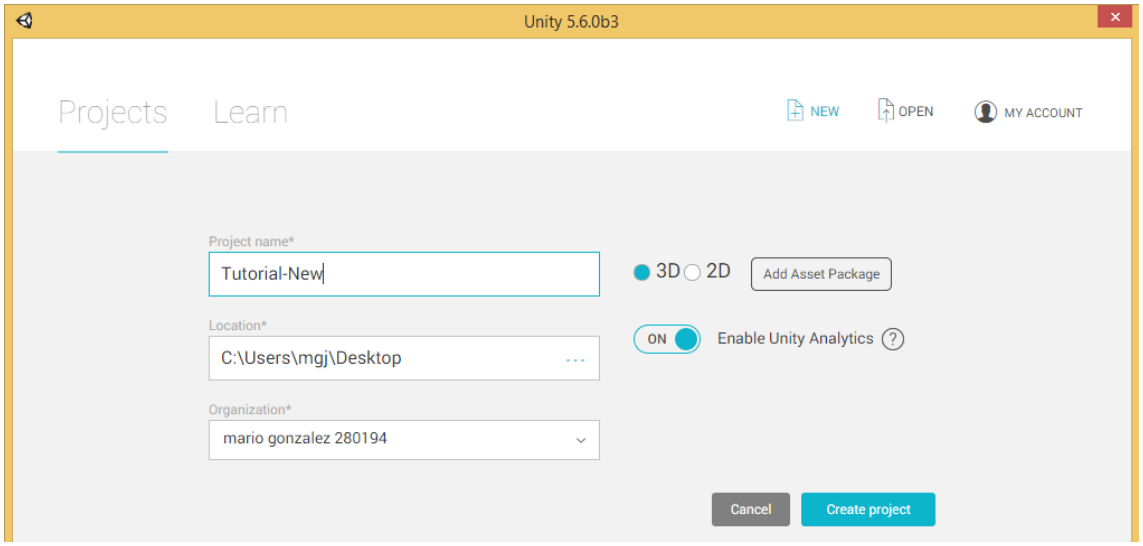
- Unity installed. You can download here for free.
- The MSS source code folder. You can download from our website.

The MSS source code folder contains some external plugins such as EmguCV, a .Net wrapper to the OpenCV library, the classes developed and some objects for Unity.

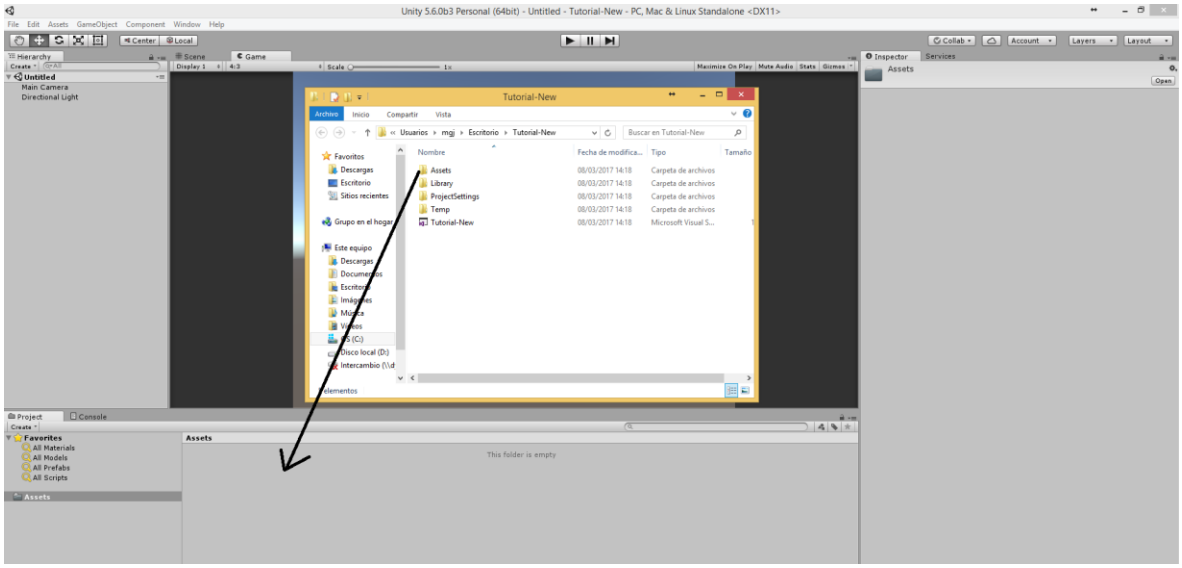


Project

At this tutorial, we create a new and empty project.

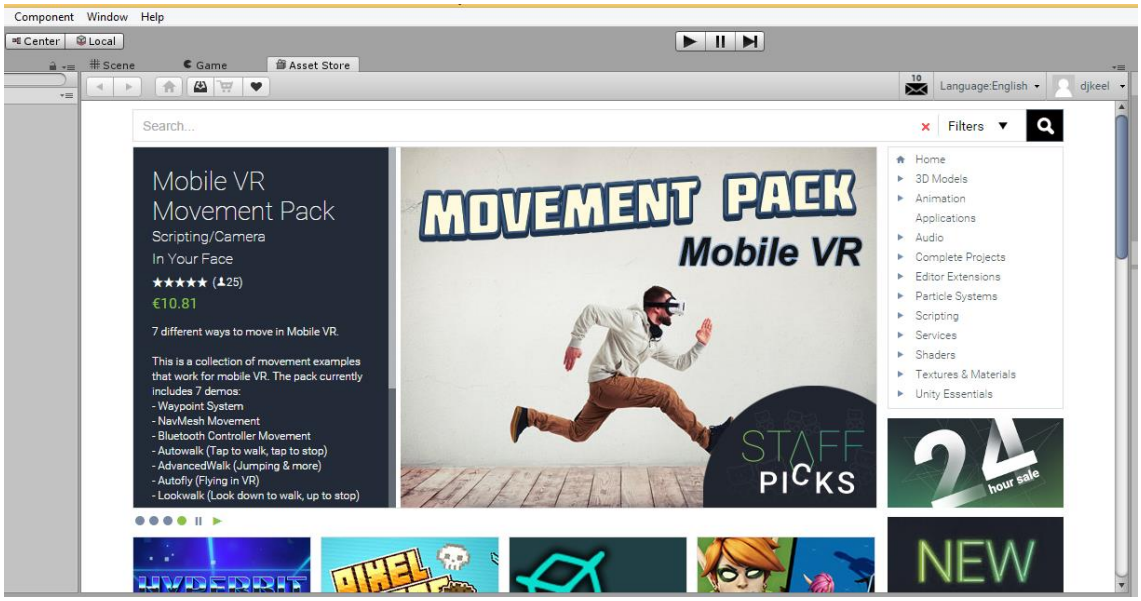


An empty project looks like this image. On the directory that you selected before, some folders has been created. In particular, ‘Assets’ is the folder which contains all files in you project (code, models 3d, images, etc.) organize in a structure how you want because Unity compiles all the files insaid.

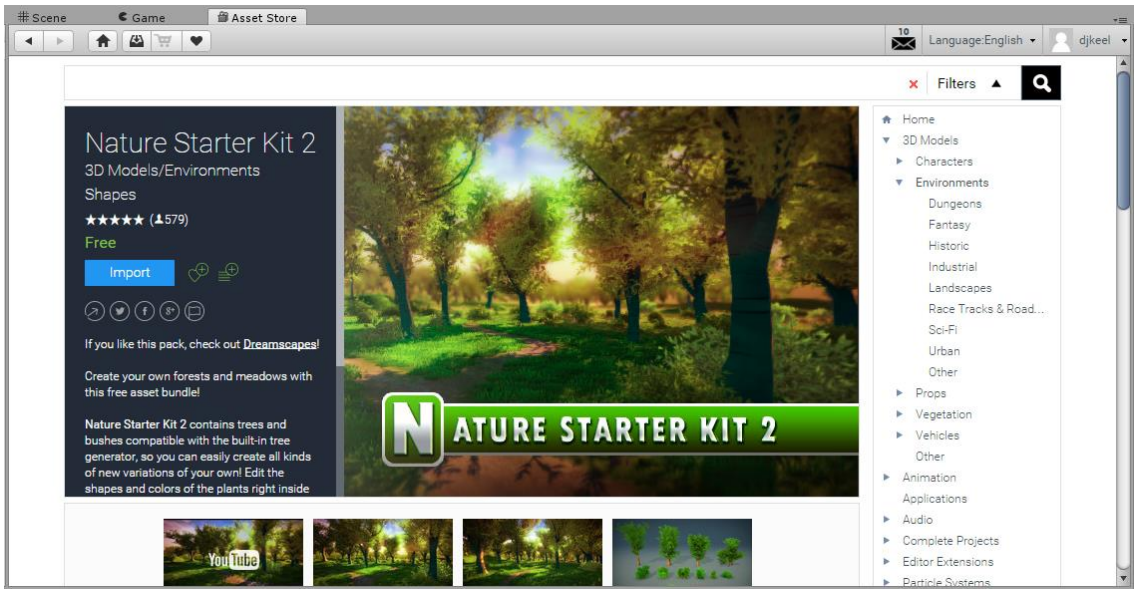


Scenario

To run the simulator, we need a scenario where we will place the cameras. There are many possibilities to do this. For example, you can download some for free in the Unity Asset store. Click on ‘Window’ menu, and later on ‘Asset Store’.



You can find scenarios in '3D Models' -> 'Environments' and 'Complete projects' sections. We download this free pack.



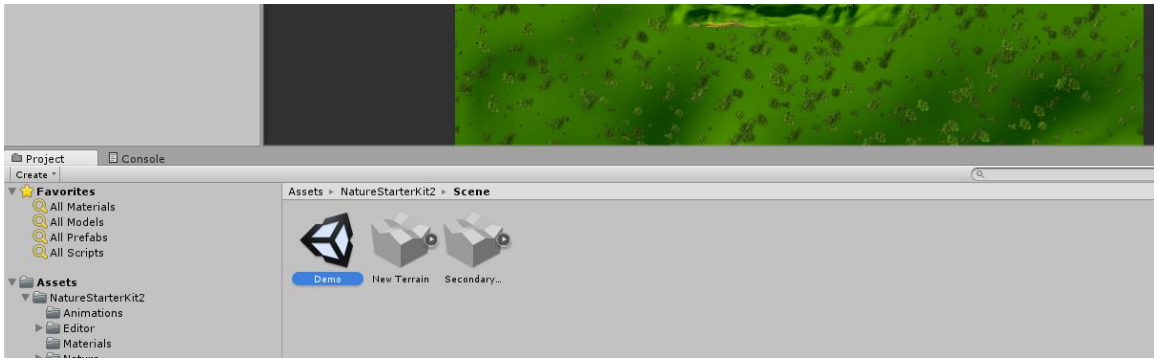
In the folder that we just add to the project, we have 3D models, textures, objects and more.

Creating new Scenarios

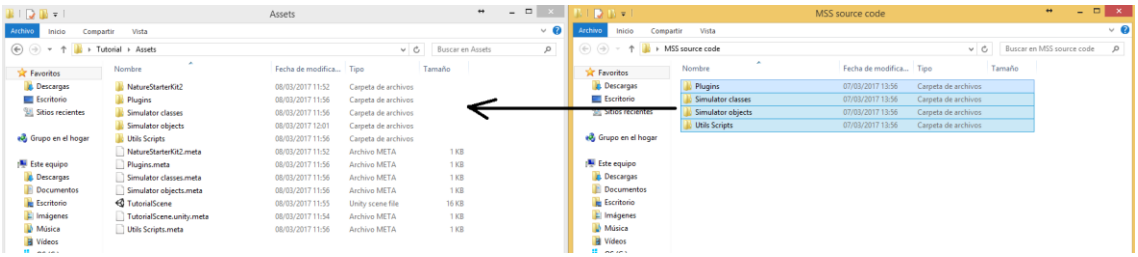
To design and to build a new scenario, first you need to create a scene.

Using existing scenarios

The pack that we have downloaded includes one scenario. Open 'Demo' in 'NatureStarterKit2/Scene/'.



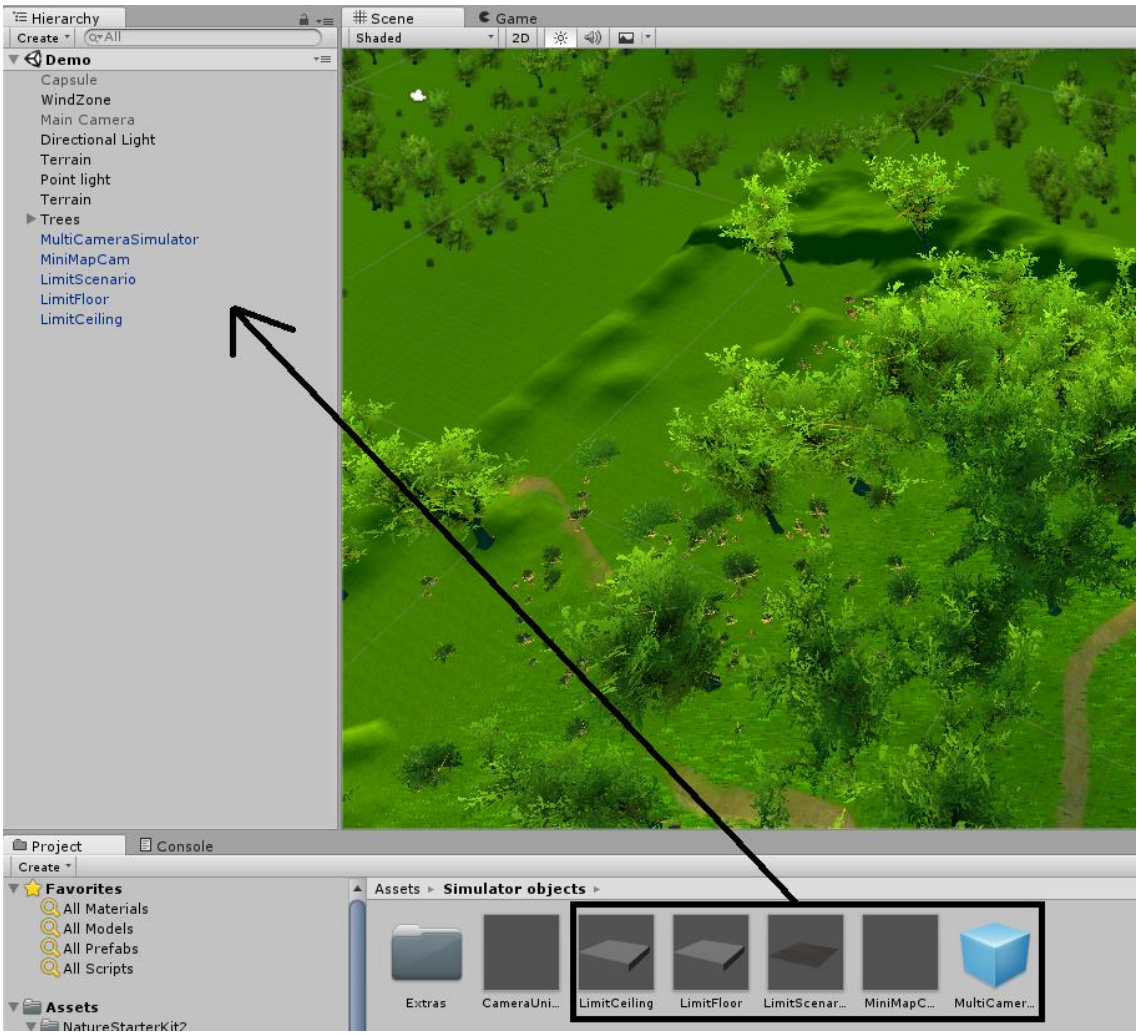
Now we have to copy the MSS folder in the 'Asset' folder of our project.



In 'Simulator objects' folder, you must check the following objects:

- 'CameraUnityObject' has the 'CameraSimulator' script attached. If not, or you have a missing reference, you need to add it. This step is showed in the video.
- 'MiniMapCam' has the 'AutoMipmap' script attached and 'Scale Image Map' variable is 1.5.
- 'MultiCameraSimulator' has the 'StartApp' script attached.

Now, add these 5 objects to our scenario with a simple drag and drop.



'MultiCameraSimulator' and 'MiniMapCam' objects don't need configuration, but 'LimitScenario', 'LimitFloor' and 'LimitCeiling' need adjustment. These objects are necessary for the system to insert cameras with a GUI. If you don't need this feature, you can ignore this step, although I recommend you to follow it.

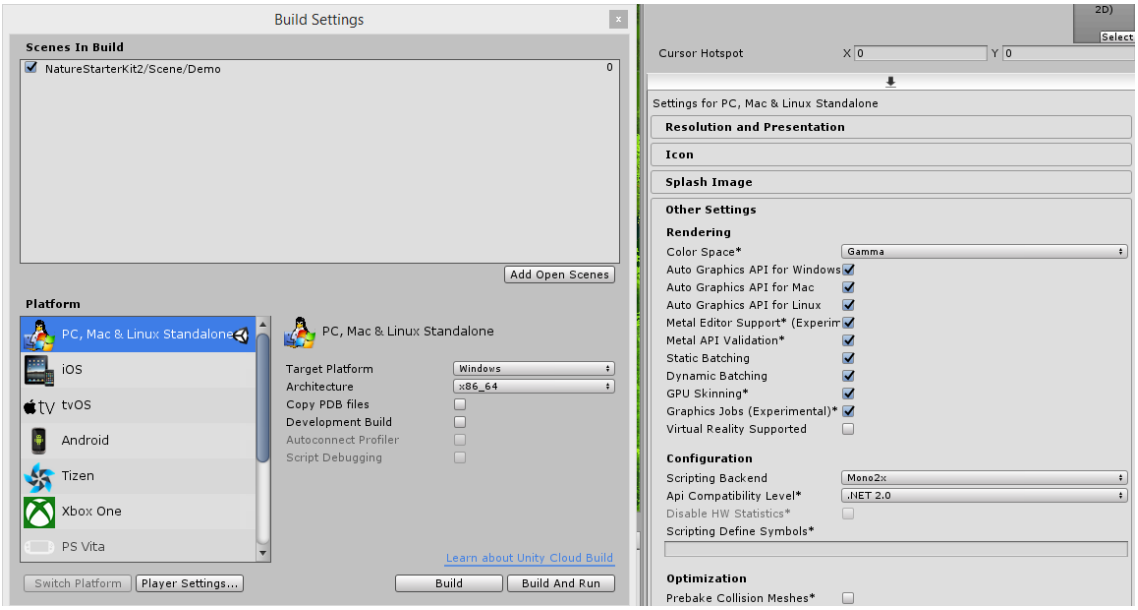
'LimitScenario' establishes the area where the cameras can be placed. 'LimitFloor' and 'LimitCeiling', the minimum and maximum coordinates in the Y axis. To adjust them, we only need to move them. You can see an example in the video.

Now we have ready to use the simulator. Press the 'Play' button to run the project, and later press 'Start' to initiate the simulator.

Export project (.exe)

In the 'File'->'Build Settings' option, you can export your project. Click on 'Add open scenes' and 'Player Settings'.

'Architecture', 'Platform', 'GPU Skinning', 'Graphics Jobs' and 'Api Compatibility Level' must be configured like this image.



Now, click on 'Build'. In the folder has been created, you need to copy the 'x64' folder ('Assets/Plugins') in the '_Data/Managed' folder.