Nethereum Plugin For Unity3D

Accessing Ethereum Ecosystem Through Nethereum

May 15, 2019

The scope of this project is to create a Unity asset that will serve as an interface of sorts for the Nethereum library. The asset will expose some functionalities to make it easier to integrate Ethereum on an application developed using Unity.

1 The Nethereum Plugin

1.1 Plugin Installation

There are currently two ways to install the plugin on Unity base on what you have. To start off, run Unity Hub and open your project. Choose the installation method that suites you below.

1. Using Unity Asset Package

   - Open your unity project and double click the `NethereumAsset.unitypackage.gz`.
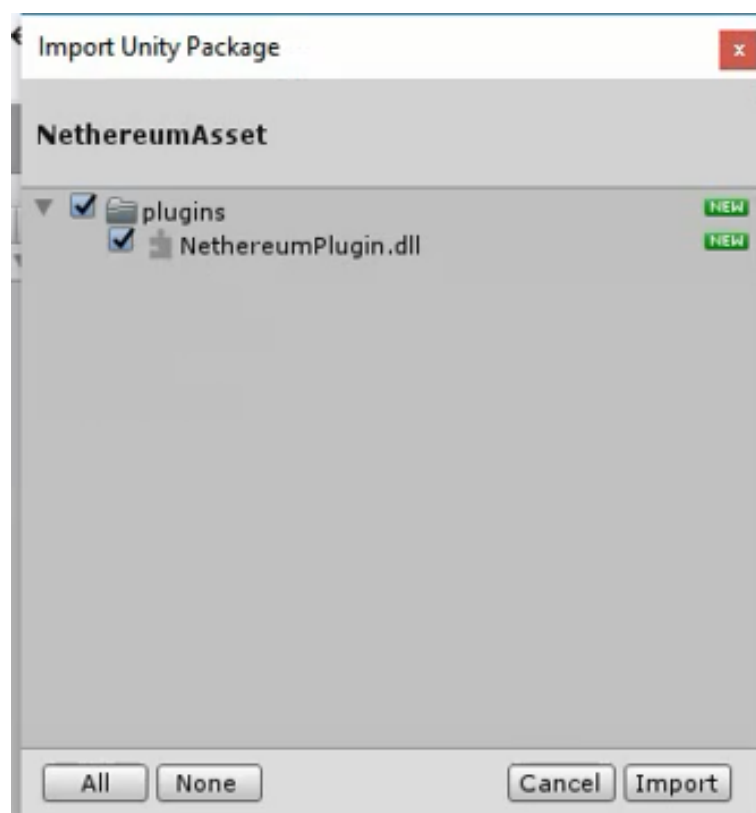   - On the window that opens, import all the available objects.

Figure 1: Opening the NethereumAsset package and importing objects.

   - After the installation is done, the plugin will be available inside the `plugins` folder that's been created.

2. Using Raw DLL

   - Create a new folder called `plugins` inside the `Assets` folder of your Unity project.
   - Place the **NethereumPlugin.dll** inside the `plugins` folder.

After finishing the plugin installation you must set now the .NET environment.

   - On the top menu of Unity, select `Edit -> Project Settings -> Player`.

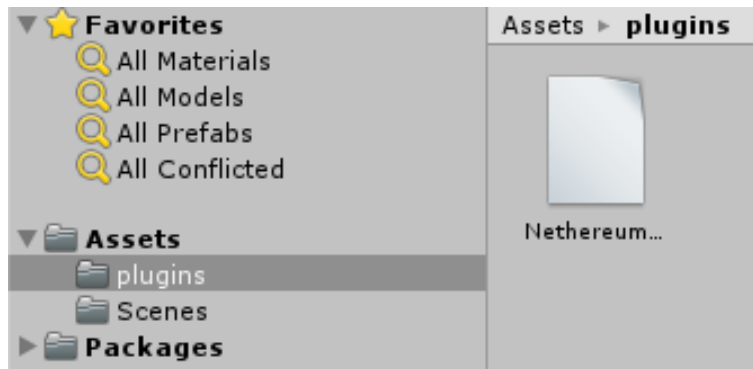Figure 2: Creating folder inside the Unity project directory.

Figure 3: Copying the plugin to the plugins folder.

- The inspector at the right will show several sections. Scroll to `Other Settings -> Configuration` and change the `Scripting Runtime Version` to `.NET 4.x Equivalent`. It will prompt you to restart Unity.
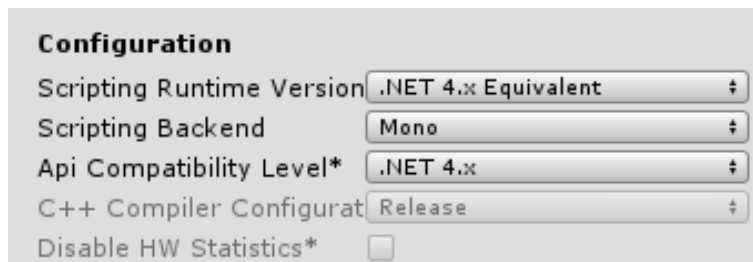

Figure 4: Changing the .NET environment.

1.2 Plugin Usage

Once the plugin has been correctly installed on your project, you can start using it. To do so, first you will have to create an instance of the plugin's class:

```
var pluginInstance = new NethereumPlugin();
```

The instance will expose all the available methods for you to use.

2 API Documentation

This are the functionalities offered by the plugin.

Table 1: API description.

Function Name	Description	Input	Input Description	Output
GenerateKeyPair	Method to retrieve the public address and private key.	None		Struct containing the public address and private key.
DeepLink	Method to deep link to another application	- AppUri - PublicAddress - TokenAddress - Amount - ReturnLink	Base URI of the app to communicate with. Public address of your app. The ensName to be sent to other app. Amount to be sent to other app. Callback method of your app.	None.
InitUser	Creates a new user.	- URL - IDFV - CoinId - Amount	URL of the backend. ID for vendor. ID of the coin to be use. Amount of coin to initialize user.	String containing the session token.
UpdateUser	Updates the coin amount of user.	- URL - Amount - SessionToken	URL of the backend. Coin amount to be added or subtracted to user. Session token received from InitUser.	None.
GetUser	Retrieves the user coins amount given its session cookie.	- URL - SessionToken	URL of the backend. Session token received from InitUser.	Returns the coin amount of the user.