Blockade Labs - AI Skybox Generator

Create stunning AI-generated skybox assets within Unity for use as HDRIs in game dev and immersive projects.

Unity Versions Support

- >= 2021.x.x - works out of the box
- >= 2020.x.x - requires installing newtonsoft-json

If you are using a Unity version `2020.x.x` you would need to add a `newtonsoft-json` package to your project. Simplest way to install the package is to open `Packages/manifest.json` file of your project with your favourite editor and add the following in your `dependencies` (make sure to respect JSON commas):

```
{
 "dependencies": {
   "com.unity.nuget.newtonsoft-json": "3.0.2",
 }
}
```

Using Pusher

Scripts in the package can be used standalone or optionally together with a Pusher websockets package. If installed, the Pusher package will use websockets to listen for any changes in the Asset Generation Process on Runtime and make updates accordingly, which in return should improve performance for your games on Runtime.

Edit the project's `manifest.json` file

Again, easiest way to install the optional Pusher dependency is to open `Packages/manifest.json` file of your project with your favourite editor and add the following in your `dependencies` (make sure to respect JSON commas):

```
{
  "dependencies": {
    "com.pusher.pusherwebsocketunity": "https://github.com/pusher/pusher-websocket-unity.git#upm",
  }
}
```

Use git URL option

Alternatively you can go to your Unity Project, to `Window > Package Manager` and install the packages using the `Add package from git URL...` option.

When using Pusher make sure to use this URL (uses the UPM branch):

`https://github.com/pusher/pusher-websocket-unity.git#upm`

Use OpenUPM-CLI

If you are using `OpenUPM-CLI`, you can easily install the Pusher package using the command below:

```
openupm add com.pusher.pusherwebsocketunity
```

For known issues after installation check the section below.

Getting Started

Samples

After installing the Blockade Labs - AI Skybox Generator you can go to `Window > Package Manager` and switch to `Packages: In Project` tab to locate the package. On the Blockade Labs package page there are samples that can be imported in your Project. Samples contain some assets and a demo scene to get you started (Skybox Scene).

After importing the samples load the above mentioned demo scene inside your project which should be located in `Assets/Samples/Blockade Labs SDK/x.x.x/Scenes` folder.

Note: Demo scene uses Text Mesh Pro elements for runtime UI. If you haven't imported TMP Essentials you will be prompted to do so after you load the scene. When you are done importing TMP Essentials, reload the scene by either right clicking on it in the `Assets/Samples/Blockade Labs SDK/x.x.x/Scenes` folder and selecting `Reimport` or by loading some other scene and then loading the Skybox Scene again.

How to use

Pusher

On the sample scene you loaded there is game object named `Pusher` . If you don't plan on using Pusher you can delete it in order to avoid any future `The referenced script (PusherManager) on this Behaviour is missing!` warnings. If you plan on using Pusher on Runtime, you can leave it as it is.

Editor

Skyboxes

If you open the Skybox Scene sample, you will notice a game objects named `Skybox Sphere` . The Sphere has a Mesh Renderer component which uses a sample `Skybox Material` which in turn uses a shader of type `Skybox/Panoramic` . A texture generated with this package is assigned to the shader. You can generate a new texture that will replace the existing one on the sphere by following these steps.

1. Select the Sphere object.
2. Locate the `Blockade Labs Skybox` component.
3. Add your Blockade Labs' `public` API key in the designated field first.
4. Click the `Initialize` button in the `API` section.
5. After the plugin is successfully initialized some need fields will become available.
6. You'll notice that the object has an option `Assign to Material` ticked. Leave it as it is.
7. Select the desired style.
8. Fill in the `Prompt` field.
9. Click the `Generate Skybox` Button.
10. In about 20-30 seconds your texture will be replaced with a new texture you just created, and a folder located in `Assets/Blockade Labs SDK Assets` will now hold your newly created texture.
11. In the `Scene` tab of the Unity editor using the `View` Tool you can position yourself inside the sphere and check out the newly generated skybox.

Runtime

To be able to generate assets on runtime you just need to follow these simple steps:

1. Select the game object with the attached component of `Blockade Labs Skybox` .
2. Make sure that you added your Blockade Labs' `public` API key in the designated field.
3. Demo scene already has the necessary elements to display the UI for Skybox generation on runtime.
4. After you run the game UI will become active on top of your Game view.
5. Use the runtime UI in the same manner as you would in the editor (`Enter Prompt > Select a style > Generate`).

Known Issues

If you are using an optional Pusher package, after installation on 2021.x.x versions you might get an error saying:

`Assembly 'Packages/com.pusher.pusherwebsocketunity/Packages/PusherClient.2.1.0/lib/net472/PusherClient.dll' will not be loaded due to errors: PusherClient references strong named Newtonsoft.Json Assembly references: 12.0.0.0 Found in project: 13.0.0.0.`

To resolve the issue go to `Edit > Project Settings > Player > Other Settings > Configuration > Assembly Version Validation` and disable Version Validation.