On Screen Log Documentation v1.0
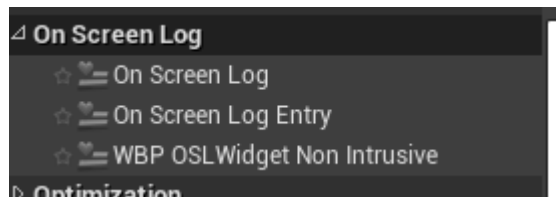
A comprehensive guide to running On Screen Logs in your Projects.
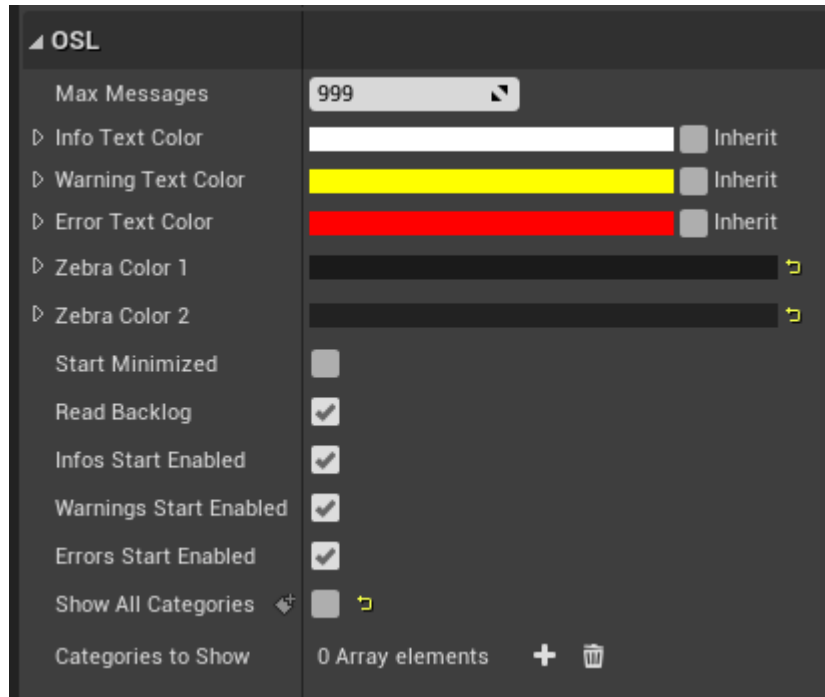
Adding a New On Screen Log

Adding a new On Screen Log is as simple as adding the On Screen Log UMG Widget to your HUD UI Blueprint.

Once you've added it, you can customize the settings to match your needs.

Max Messages	The number of messages the log can show, setting this too high may cause lag
Info Text Color	The text color of logs that have the severity of "Info"
Warning Text Color	The text color of logs that have the severity of "Warning"
Error Text Color	The text color of logs that have the severity of "Error"
Zebra Color 1	The background color to show on every second entry
Zebra Color 2	The background color to show on every other entry
Start Minimized	If true, the On Screen Log will start minimized
Read Backlog	If true, the On Screen Log will show messages from before it was added to the screen
Infos Start Enabled	If true, Info Messages will show by default
Warnings Start Enabled	If true, Warning Messages will show by default
Errors Start Enabled	If true, Error Messages will show by default

Show All Categories	If true, This On Screen Log will show messages from all categories.
Categories to Show	(Show All Categories = false only) A list of categories to show in this On Screen Log

There are also multiple functions you can call on the On Screen Log Widget.

Set Severity Enabled	Toggles whether or not the severity will show
Set Category Enabled	Toggles whether or not the category will show (overridden by bShowAllCategories)
Set Show All Categories	Toggles whether or not this logger should show every category of message
Clear Log	Clears this logger and removes all messages
Minimize	Minimizes this logger so its hidden on the screen
Maximize	Opens this logger so its visible on the screen
Is Minimized	returns true if this logger is minimized
Is Maximized	returns true if this logger is maximized
Is Message Visible	returns true if the message in this Loggers' log is visible (not being filtered out)
Is Severity Enabled	returns true if this logger is showing messages with the given severity
Is Category Enabled	returns true if this logger is showing messages with the given category
Is Showing All Categories	returns true if this logger is showing messages from every category
Get Severity Color	returns the text color for the given severity

Enabling Logs in Shipping Builds

By default, the plugin will disable itself if Shipping is Enabled.
To enable logs in shipping builds, ue4 requires you to first build from their source version of the engine. You can find this on their Github.
Once you have the Source Version, you can modify your {Game}.Target.cs file to enable logs in shipping.

```
// enable logs and debugging for Shipping builds
if (Configuration == UnrealTargetConfiguration.Shipping)
{
    BuildEnvironment = TargetBuildEnvironment.Unique;
    bUseChecksInShipping = true;
    bUseLoggingInShipping = true;
}
```

Once you have the source build working - you can simply go to the On Screen Logging settings and turn off "Disable in Shipping".
(Edit > Project Settings > On Screen Logging)
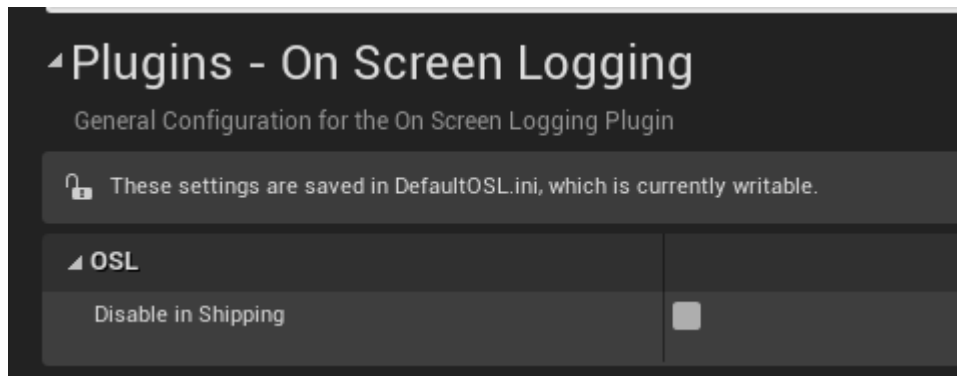
The logs will now work in shipping builds. Remember to turn this off before you release it!

Installing To Project

To install the plugin to a project (and not the entire engine) - you must first have it installed to an Unreal Engine 4 version. After this you can copy it out of the engine folder and install it directly onto a project.

First off - navigate to the On Screen Log Plugin directory in your UE4 engine folder.
This is Typically found in C:\Program Files\Epic Games\UE_4.27\Engine\Plugins\Marketplace\OnScreenLog

Copy this entire directory.
Next go to your Project directory and add a "Plugins" folder

Paste the Plugin into this folder (It should end up as Project > Plugins > On Screen Log)

You can now open your project and enable your plugin as normal.
Remember to uninstall it from the engine as well!

Troubleshooting
If you run into any problems with the installation - try these steps first:
1. Download and install Dot Net 3.1
   (https://dotnet.microsoft.com/en-us/download/dotnet/3.1)
2. Install the latest Visual Studio to Unreal (you can do this quickly by creating a new C++ project in UE4 - it will prompt you to install Visual Studio).

If you are still having problems - send an email to the store support link and attach your editor logs (Logs are found in Project Name > Saved > Logs)

Installing To Unreal Engine 5

To install the plugin to Unreal Engine 5 - you must first have it installed to an Unreal Engine 4 version. After this you can copy it out of the engine folder and install it directly onto a UE5 project.

First off - navigate to the On Screen Log Plugin directory in your UE4 engine folder.
This is Typically found in C:\Program Files\Epic Games\UE_4.27\Engine\Plugins\Marketplace\OnScreenLog

Copy this entire directory.
Next go to your UE5 Project directory and add a "Plugins" folder

Paste the Plugin into this folder (It should end up as Project > Plugins > On Screen Log)

Next, open the folder that you just pasted and open the OnScreenLog.uplugin file with a text editor. You will most likely have to right click->open with… and then select "notepad".
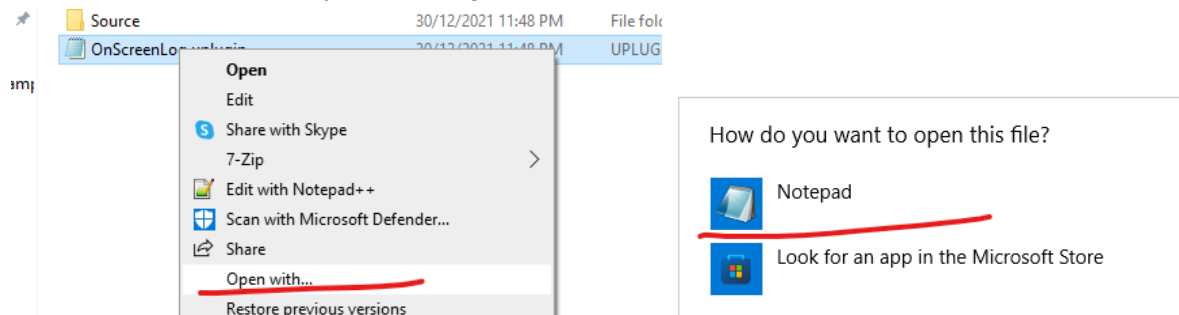
Finally, where the file says "Engine Version", replace this to be "5.0.0"

```
File   Edit   Format   View   Help
{
        "FileVersion": 3,
        "Version": 1,
        "VersionName": "1.0",
        "FriendlyName": "On Screen Log",
        "Description": "A simple plugin that can draw the UE4 Output Log to the screen for debugging purposes",
        "Category": "Other",
        "CreatedBy": "JustinKiesskalt",
        "CreatedByURL": "",
        "DocsURL": "https://docs.google.com/document/d/1dyPsD-q8QH9FBxaNYSOFEEZTUNWquV1WwALvsLmvSU8/edit?usp=sharing",
        "MarketplaceURL": "com.epicgames.launcher://ue/marketplace/product/96d86c7b87c04643a6ebf2f1508fb0d0",
        "SupportURL": "",
        "EngineVersion": "5.0.0",
        "CanContainContent": true,
        "Installed": true,
        "Modules": [
                {
                        "Name": "OnScreenLog",
                        "Type": "Runtime",
                        "LoadingPhase": "Default",
                        "WhitelistPlatforms": [
                                "Win64",
                                "Mac"
                        ]
                }
        ]
}
```

You can now open your project and enable your plugin as normal.

Troubleshooting
If you run into any problems with the installation - try these steps first:
3. Download and install Dot Net 3.1
    (https://dotnet.microsoft.com/en-us/download/dotnet/3.1)
4. Install the latest Visual Studio to Unreal (you can do this quickly by creating a new C++ project in UE4 - it will prompt you to install Visual Studio).

If you are still having problems - send an email to the store support link and attach your editor logs (Logs are found in Project Name > Saved > Logs)

Miscellaneous Functions

On Screen Logs offers some extra functionality to enable logging in blueprints as well as handling the history of logs. Below is a list of all functions offered by the Plugin.

OSL Clear Backlog	Clears the backlog, meaning Loggers with "Read From Backlog" Enabled will not read logs from before this point.
OSL Log Info	Logs a new Info Message to the "LogDefault" Category
OSL Log Warning	Logs a new Warning Message to the "LogDefault" Category
OSL Log Error	Logs a new Error Message to the "LogDefault" Category
OSL Get Num Messages	Returns the number of messages that are stored in the backlog.