

Screen Log



Table of Contents

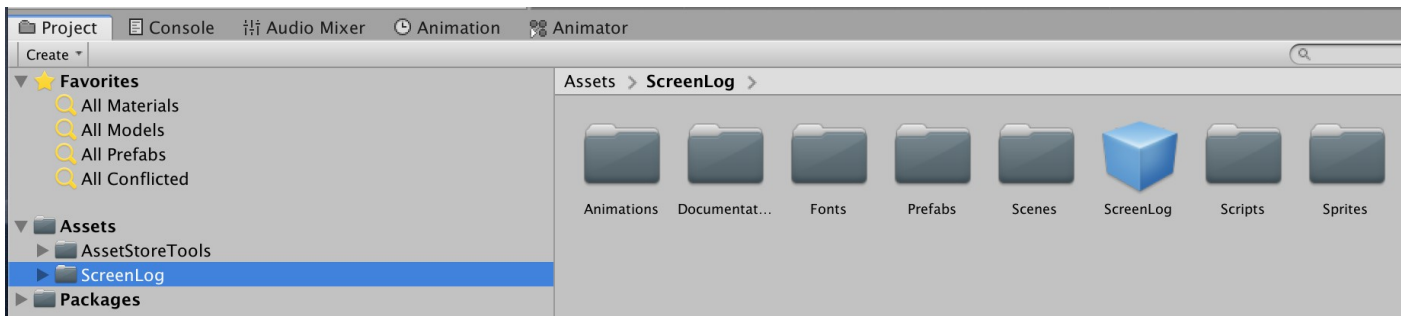
1. INTEGRATION.....	3
2. FOLDER STRUCTURE.....	3
2.1. ScreenLog.....	4
3. INTRODUCTION.....	4
4. SCREEN LOG GUI.....	5
4.1. Buttons.....	5
4.2. Settings Popup.....	5
4.3. Log.....	6
5. CODING PART.....	7
6. How to custom edit Screen Log Plugin.....	8
6.1. Graphic.....	8
6.2. Log Settings.....	8

Table of Images

Picture 1: Integration.....	3
Picture 2: Screen Log Folder Structure.....	3
Picture 3: Introduction.....	4
Picture 4: Buttons.....	5
Picture 5: Settings Popup.....	5
Picture 6: Log.....	6
Picture 7: Include namespace.....	7
Picture 8: How to send different messages.....	7
Picture 9: Customize the Screen Log.....	8

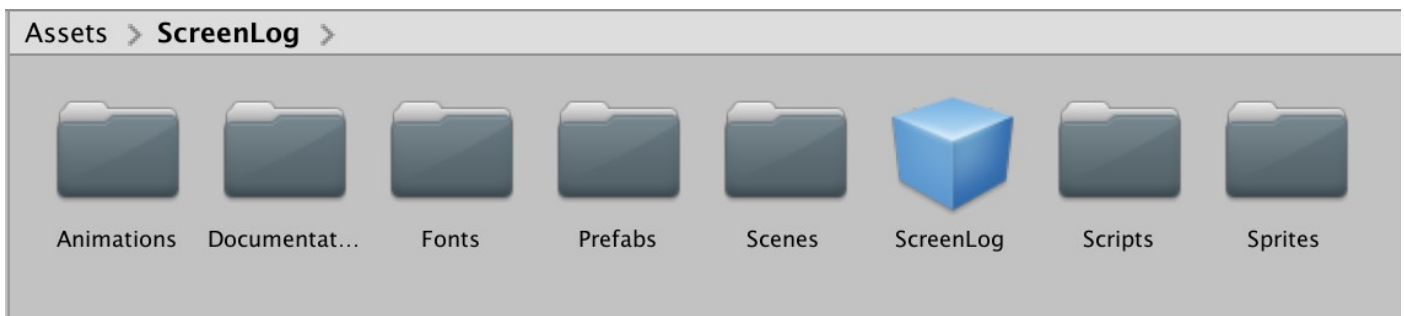
1. INTEGRATION

Download and import Screen Log Asset into your project. After that new folders appears under your Assets project folder.



Picture 1: Integration

2. FOLDER STRUCTURE



Picture 2: Screen Log Folder Structure

2.1. ScreenLog

This is the main project folder. It includes the following subfolders:

- **Animations:** animations which are used in the project
- **Documentation:** documentation about the project
- **Fonts:** main project font
- **Prefabs:** object prefabs used in the project
- **Scenes:** test scene
- **Scripts:** script files
- **Sprites:** sprite files used in the project

3. INTRODUCTION

Screen Log is a tool to help you track down bugs and tweaks your projects while running on the target devices (mobile or desktop). Checking the log messages allows you to diagnose the problems, or to make adjustments easier.



Picture 3: Introduction

4. SCREEN LOG GUI

4.1. Buttons

Screen Log has 3 buttons on the bottom of the screen for the portrait mode or on the left part of screen for the landscape mode. First button Show or Hide Settings popup. Second button Show or Hide log window, and the third button enable or disable logging. Every button has different color animation when enabled or disabled.



Picture 4: Buttons

4.2. Settings Popup

With the dropdown 'Position on the screen' you can change position of the log on the screen.

To change the size of the log adjust slider 'Size (% of screen)'.

With slider 'Number of lines' you can set maximum number of lines in the log. *Notice: when you adjust this slider the log will be cleared and counter reseted.* When the log reach the max number of lines, then automatically remove first line and add the new one.

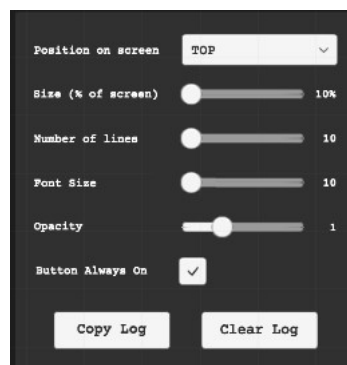
Slider 'Font size' change the size of the font used in the log.

Slider 'Opacity' change the opacity of log and button's background.

With Check 'Button Always On' you can set that the Log Screen button will be always on the screen (swipe won't hide them.).

Button 'Copy Log' copies the log content into the clipboard.


Button 'Clear Log' clears the log and reset the content.



Picture 5: Settings Popup

4.3. Log

Here are showed messages which are sent to the log. The latest message is always at the bottom. Every different type of message has own color. Default colors are white for debug message, yellow for the warning message and red for the error message.

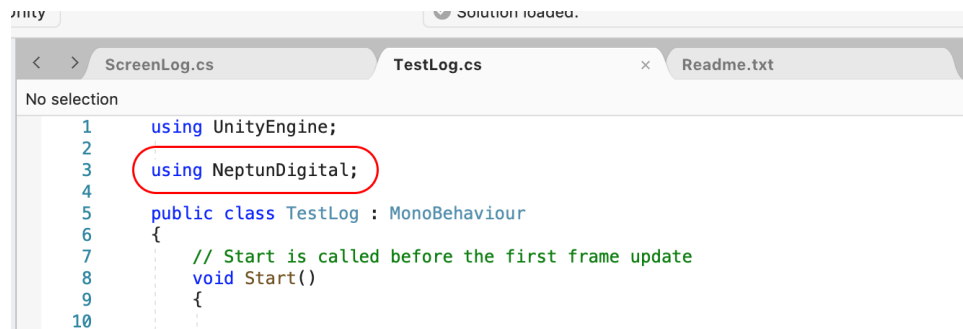


```
1. Warning: Buttons not always on the screen.  
2. Debug: Previous values loaded.  
3. Debug: Log position changed.  
4. Debug: Log init.  
5. Debug: Log position changed.  
6. Debug: Log position changed.  
7. Debug: Changing log size...  
8. Debug: Log position changed.  
9. Debug: Changing log size...  
10. Debug: Log position changed.  
11. Debug: Changing log size...  
12. Debug: Log position changed.  
13. Debug: Changing log size...  
14. Debug: Log position changed.  
15. Debug: Changing log size...  
16. Debug: Log position changed.
```

Picture 6: Log

5. CODING PART

In top of you class add line “using NeptunDigital;”, to include the plugin namespace.



Picture 7: Include namespace

On the picture 6 you can see, how can you send different messages to the Screen Log.

```
//send the debug message  
ScreenLog.Instance.SendEvent(TextType.Debug, "Debug message.");  
  
//send the warning message  
ScreenLog.Instance.SendEvent(TextType.Warning, "Warning message.");  
  
//send the error message  
ScreenLog.Instance.SendEvent(TextType.Error, "Error message.");
```

Picture 8: How to send different messages

6. How to custom edit Screen Log Plugin

We do not recommended this for new Unity users.

6.1. Graphic

All graphic elements are located in the folder **Assets/ScreenLog/Sprites**. To use your own images replace image files and do not forget to keep the same name, size and format (else you will need to change ui object, positions, etc...)

6.2. Log Settings

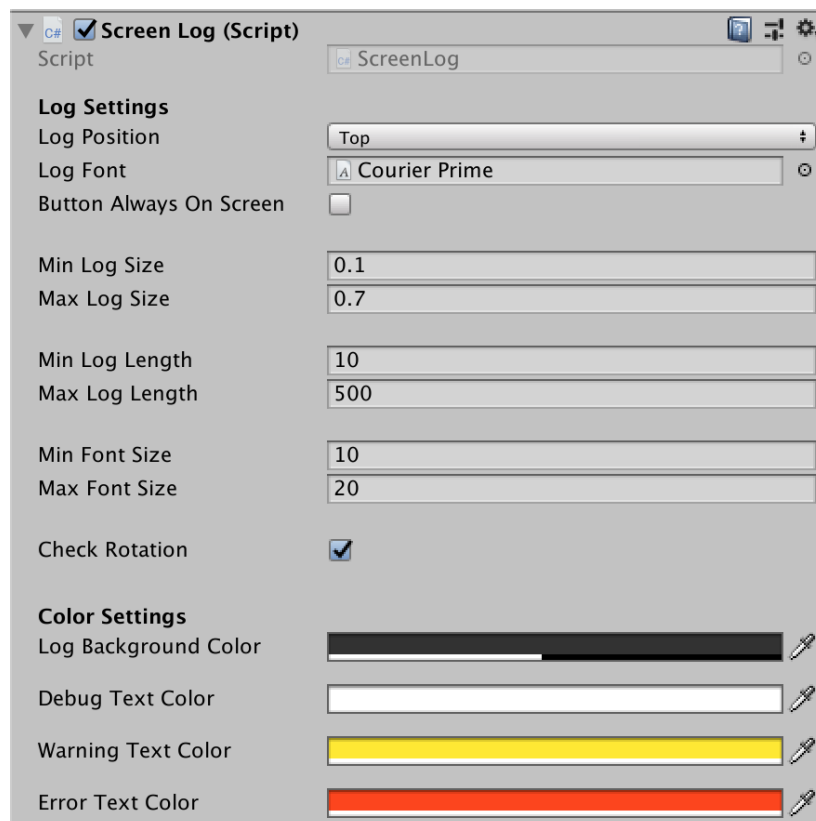
To change the plugin font drop your own font to Log Font setting. Code will automatically change all fonts on the Text components (visible on the running device or the editor).

With Min and Max Log Size you can set minimum in maximum size of Screen Log. You can change float Range [0,1]. 0 mean 0% of screen, 1 mean 100% of screen.

With Min and Max Log Length you can set minimum and maximum of log length (lines in the log). Keep in mind that every line is gameobject with the Text component. So set Max Log Length to very high value can lead to crashes or other issues.

With Min and Max Font Size you can set minimum and maximum size of font used in the log.

Under the Color settings you can change the color of log background and Colors of every type of messages.



Picture 9: Customize the Screen Log