LOADING SCREEN STUDIO

Documentation - v1.0

Scroll down for more

Content

1. Frequently Asked Questions (FAQ)

• Does LSS support URP/HDRP rendering?

Yes. There's no difference when it comes to rendering pipelines.

• I'm stuck and need help, what can I do?

If you can't find a solution for your problem in this doc, **contact me!** I'd gladly help to solve your issue.

• What platforms can I build for?

LSS works in builds for all platforms that are listed in Unity build window.

• I'm getting errors, why and how can I fix it?

It could be about anything. If you haven't already, add your scenes to the build window (Scenes In Build). Make sure to import **TextMesh Pro** from package manager and its essentials from Window > TextMesh Pro. If you're still having the issue, contact me with some details.

• Are you going to support and add new stuff to the package?

Of course! I always reply and help to support requests, and update the package frequently.

• Can I create my own loading screen presets?

Yes, you can create new loading screens or edit the existing ones.

• Loading process goes straight to 90%, why is that?

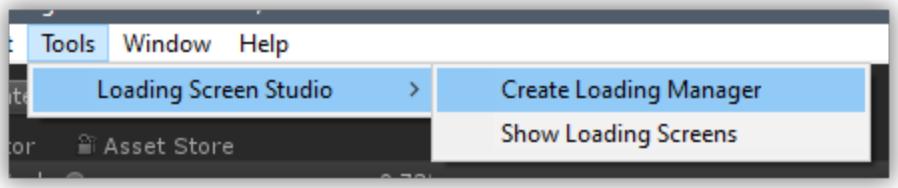
You're probably loading a lightweight scene. **0 - 90** is loading of the level, and **90 - 100** is the activation of the level. However, you can use Virtual Loading in such cases. You can scroll down for more information about virtual loading.

2. Quick Start

First of all, thanks for purchasing the package! If you need some help to get started, this is the right place.

2.1. How to use

First of all, make sure to add your scenes to the **Scenes In Build**. You'll also need a **Loading Screen Manager** in your scene, so let's create one from the top menu.



You can now select a loading screen from **Loading Screen Manager**, or tweak some settings and enable features like loading with a collider/trigger. To load a scene, you can either call the loading screen via **OnClick**, **trigger**, or within your **script**.

OnClick:

Loading Screen Manager > LoadScene() > "Your Scene Name"

Script:

```
using Michsky.LSS; // LSS namespace required

public LoadingScreenManager lsm; // Your LSM variable

void YourFunction()
{
   lsm.LoadScene("Your Scene Name");
}
```

2.2. Virtual loading

If you have a lightweight scene and want to keep loading screen for some time, you can use this option. It basically creates a 'fake' loading screen. Just enable the feature, set a time, and you're good to go.

3. Editing Loading Screen

LSS loading screens are using the native Unity UI, so you can change or add new things easily. You can also edit your screen via inspector, too. Just drag a loading screen prefab to your scene (you can see them by clicking **Tools > Loading Screen Studio > Show Loading Screens**), click on it, and start changing stuff! After making the changes, make sure to hit **'Apply'** your prefab on the top right corner, otherwise your changes won't be saved.

3.1. Layout

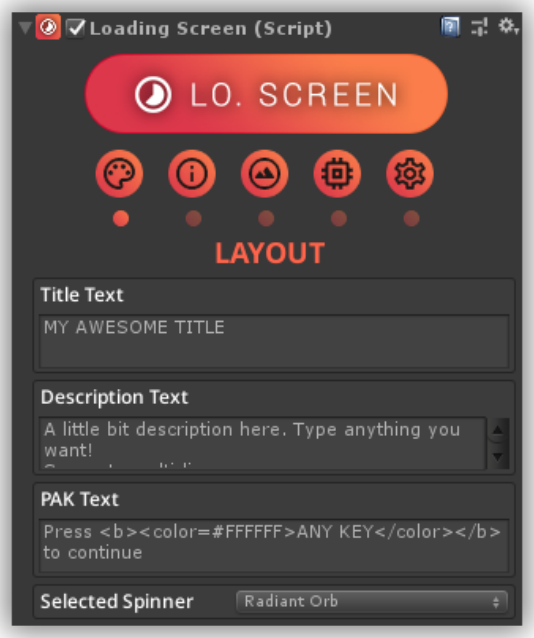
In this tab, you can change texts, fonts, colors, and spinners. By clicking 'Make It Visible', you can see the loading screen prefab in your scene view. You can later click it back to make it invisible.

3.2. Hints

If you want to give the players some tips, then enable this. You can change the timer between hints.

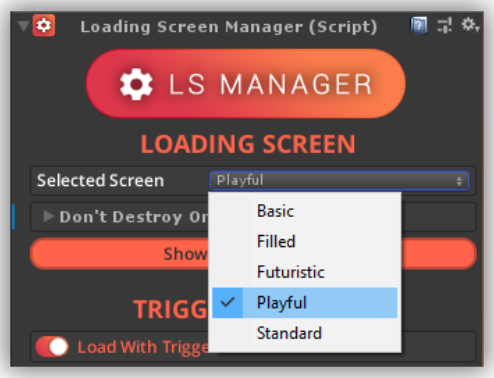
3.3. Backgrounds

You can either use a static background, or enable 'Random Background Images'. You can also change the transition speed by tweaking 'Fading Speed'.



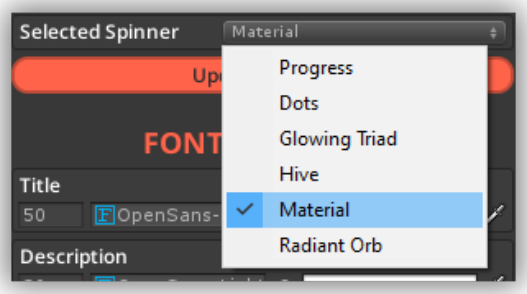
4. Creating Loading Screen

To create a new loading screen, go to **Loading Screen Studio > Resources > Loading Screens** folder via project tab, and then duplicate one of the existing loading screen. After doing that, you'll able to see a new loading screen preset in your **Loading Manager** object. You can now change anything you want, or add new stuff to your prefab by dragging the screen to your scene. After making the changes, make sure to hit **'Apply'** your prefab on the top right corner, otherwise your changes won't be saved.



5. Creating Spinner

To create a new loading spinner, drag your loading screen prefab to your scene, and then go to **Your Loading Screen > Canvas > Content > Spinner Type** via hierarchy. You'll see some spinner objects, duplicate one of them or create a new object, make sure that **'Spinner Item'** component is attached. You can add some images to Foreground or Background, depending on what you want. By creating or duplicating an object, you'll be able to see its name on the spinner list.



6. Contact & Licence

You can contact me or get the latest updates via:

Discord

E-mail

Website

YouTube

If you have any problems, questions, suggestions or feedback, please feel free to contact me.

Licence