Dynamic Split Screen

Dynamic Split Screen provides a quick way to prototype split screen project without caring about rendering.

All rendering is done in SplitScreenManager with use of 4 cameras (1 Background, 2 player cameras, and Main camera which renders overlay and UI).

Supported modes are 2D and 3D top-down view (with camera looking from an angle).

Quick start

Before doing anything else, please read these simple instructions to set things up properly.

All you need to do to setup the dynamic split screen is to:

1. Import package, make sure to select all

2. Create 2 sorting layers, "Background" and "Foreground", in this order, for 2D scene to render properly, this is to make sure background of the level is rendered below players who are placed on "Foreground" layer with player UI

3. Open sample scenes and test things out by yourself

Custom setup

1. Create new scene or use existing one

2. Top menu -> DynamicSplitScreen -> Click Setup2D / Setup3D

3. In your player script or any other script, register players by calling `SplitScreenManager.Instance.RegisterPlayer(transform);`

4. In case you are using 2D camera, make sure your players are placed on "Foreground" sorting layer and whatever static background sprites you have in the scene are on "Background" sorting layer

LWRP / URP specific setup

Before custom setup you need to follow these steps for LWRP setup.

1. Create pipeline asset
2. Enable depth texture in pipeline asset
3. Upgrade materials.
4. Create ForwardRendererData (For LWRP, in case of URP it's automatically created with pipeline asset)
5. Uncomment `#define LWRP` in **SplitScreenManager**.cs and **SpltiScreenRendererPass** . Make sure editor recompiles code.
6. Open your custom **ForwardRendererData** and **assign SplitScreenRendererPass** to Renderer Features list.
7. Next up, find **SplitScreenCamera3D** in your project and set Camera **RendererType** to **Custom** and **assign** your new **RendererData**. Also make sure you assign RendererData to SplitScreenManager component.

FAQ

How to create Custom Split Line?

- Check out `DynamicSplitScreen/Prefabs/SplitLine` contents to see how you should make your own split line. For split background, full red color (255,0,0) defines player 1 screen.
- Use higher resolution (4096x4096 or higher) square texture with vertical split line, since it's going to be scaled by SplitLine.cs to more than the size of the screen.
- And most importantly, assign new background and line textures to `SplitLine` script attached to `SplitScreenUI`.

How to modify camera effects and transitions?

- In `DynamicSplitScreen/Prefabs/CameraControls` you can find test `CameraControlTrigger 2D` or `CameraControlTrigger 3D`, drag it into the scene and add CameraControlTrigger component, there you can enable/disable fade and drag CameraControls into `onTriggerEnterControls` or `onTriggerExitControls` that you want.
- If you want to add different controls, then right click in project window `Create -> Camera Control -> Choose whichever you want`, modify it, and drag to CameraControlTrigger.
- If you want to add completely new controls, in `DynamicSplitScreen/Code/CameraControls` you can see scripts that are generating scriptable objects for use in CameraControlTrigger, create your own by deriving a class from CameraControlBase.

For more in depth guides go to https://marekkost.com