

GUI Pack2D Free

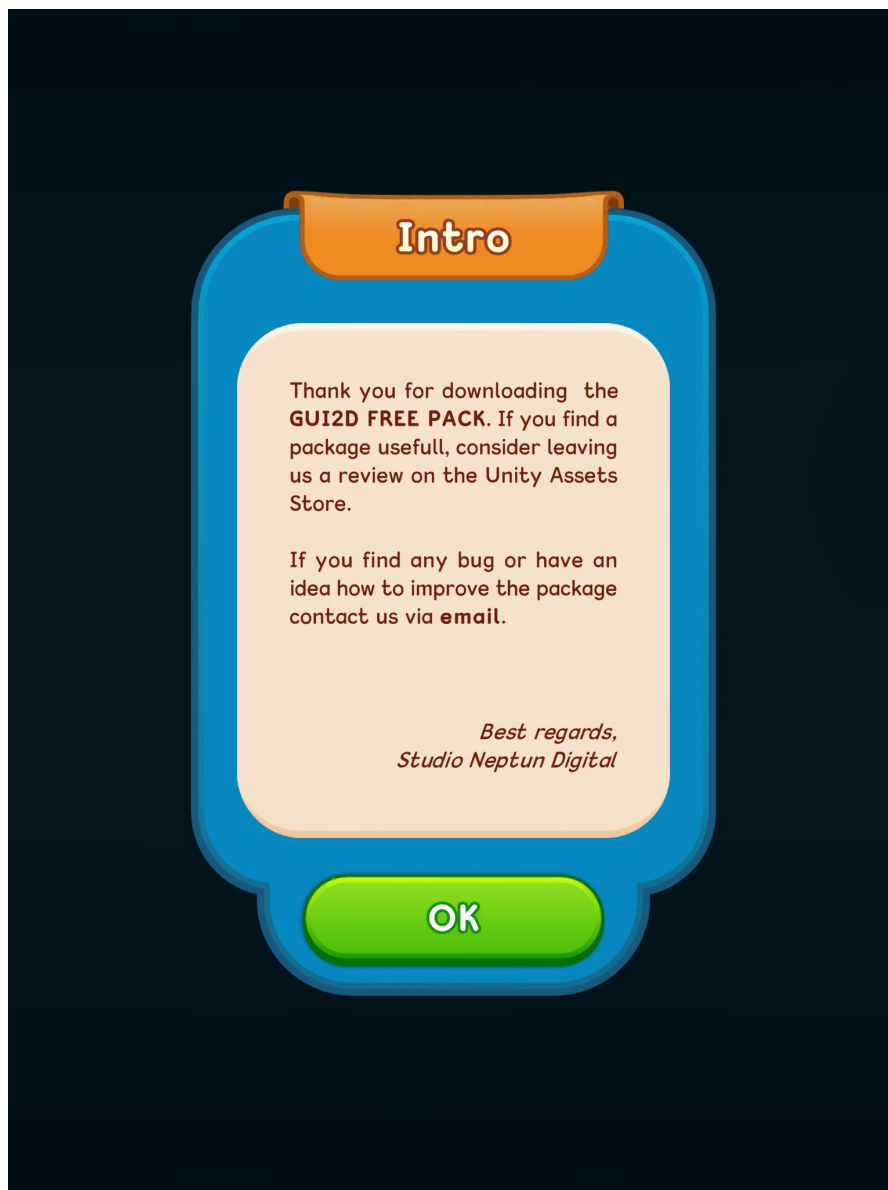


Table of Contents

1. Intro.....	3
2. Integration.....	4
3. Folder structure.....	4
3.1. GUIPack2DFree.....	5
4. Features overview.....	6
5. How to integrate and use popup system?.....	7
5.1. Step 1.....	7
5.1. Step 2.....	7
6. How to create new popup?.....	8
6.1. Step 1.....	8
6.2. Step 2.....	8
6.2. Step 3.....	8
6.2. Step 4.....	9
7. How to reskin GUIPack2DFree.....	10
7.1. Graphic.....	10
7.2. Audio.....	10
7.3. Popups.....	10
8. Important notices.....	10

Table of Images

Picture 1: Integration.....	4
Picture 2: Folder Structure.....	4
Picture 3: Integration of popup system.....	7
Picture 4: Integration of popup system.....	7
Picture 5: New popup creation.....	8
Picture 6: Popup Generator window.....	8
Picture 8:Script attachment.....	8
Pictures 7: New folders.....	8
Picture 9: Newly created popup.....	9

1. Intro

GUIPack2DFree is a clean and minimalistic 2D UI pack designed to help you build a popup system for your current or next game. This mobile-friendly pack includes a variety of components as prefabs, making it easy to create your own custom popup system.

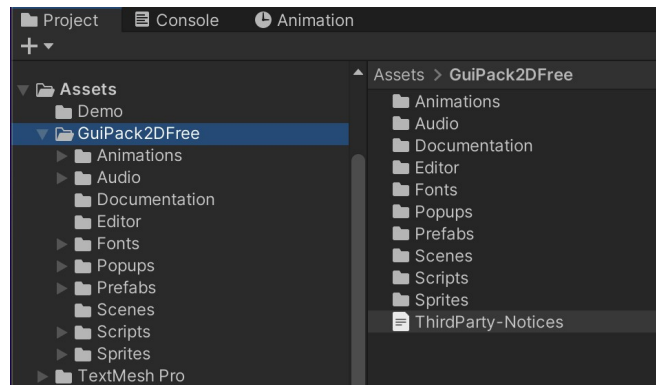
In the test scene, you can explore 5 different popup examples, demonstrating how effortlessly they can be created using this pack.

It provides a flexible yet user-friendly foundation for designing the great user popup system for your games.

2. Integration

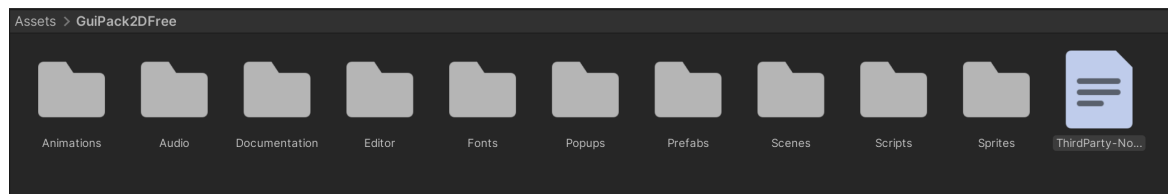
Download and import the GUIPack2D Free pack into your project. Once imported, a new folder will appear under your Assets directory.

The UI elements in this pack use TextMesh Pro Asset, so that asset is essential. If it is not already included in your project, you will need to add it manually.



Picture 1: Integration

3. Folder structure



Picture 2: Folder Structure

3.1. *GUIPack2DFree*

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

- **Animations:** animations which are used in project
- **Audio:** sound file included in project
- **Documentation:** documentation about the project
- **Fonts:** fonts and fonts assets used in the project
- **Popups:** prefabs, sprites and scripts of all available popups
- **Prefabs:** prefabs used in project
- **Scenes:** scenes used in project
- **Scripts:** script files
- **Sprites:** sprite files used in project

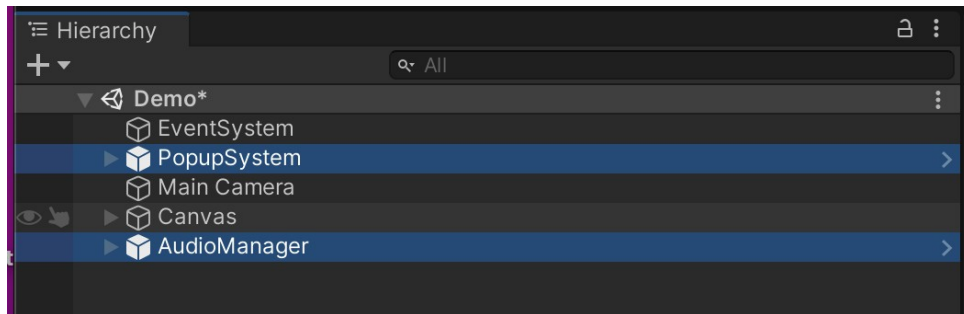
4. Features overview

- **Simple and Minimalistic**
- **Easy generation of new popups** – create new popup with just a few click!
- **Visual effects:** animations, sprites in the game
- Full C# source code included
- **User customizable**
- **Works in Unity Personal and Unity Professional**
- Simple and clean code
- **Works on all platforms:** PC, MacOS, iOS, Android, WebGL
- **Optimized for mobile and tablet devices**

5. How to integrate and use popup system?

5.1. Step 1.

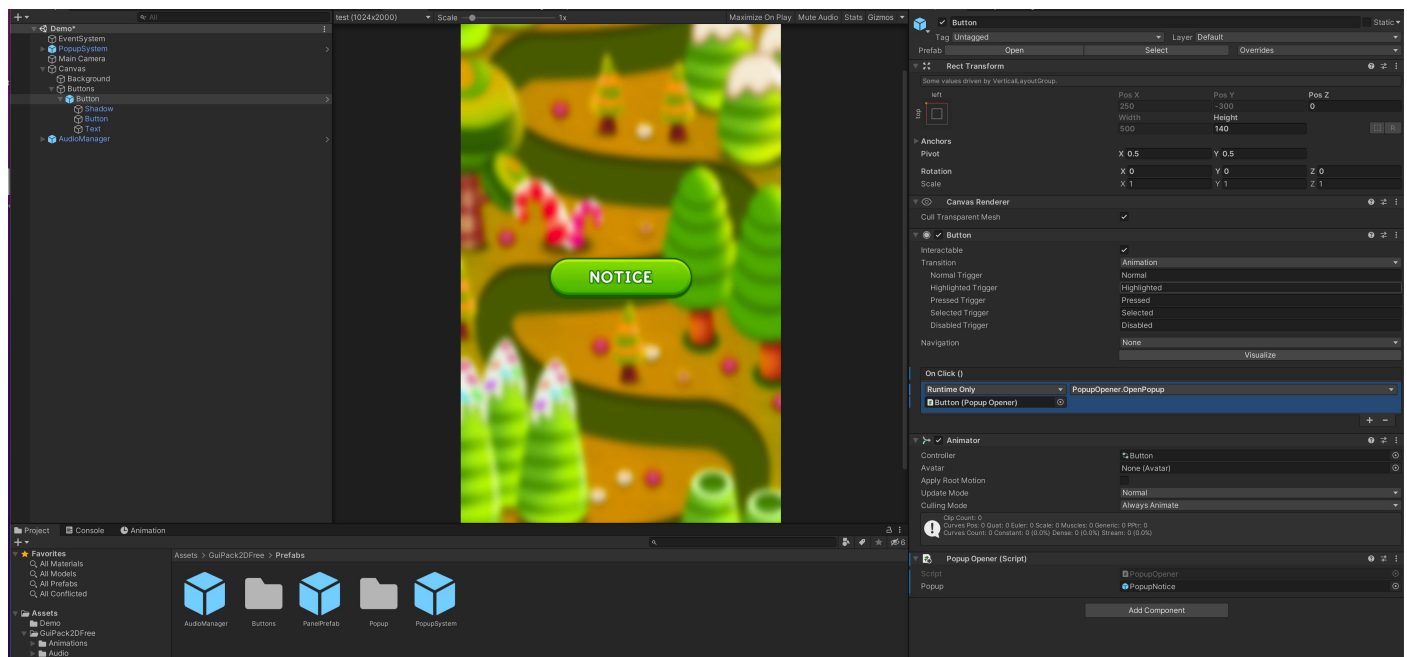
Drag and drop the PopupSystem and AudioManager prefabs into your scene. If your project has multiple scenes, you only need to add AudioManager to the first scene. However, PopupSystem must be added to every scene where it is required.



Picture 3: Integration of popup system

5.1. Step 2.

Create a new button object in the scene and attach the PopupOpener script to it. In the PopupOpener script, drag and drop or select the prefab of the popup you want to open. Lastly, don't forget to add an OnClick event to the button and assign it to OpenPopup.

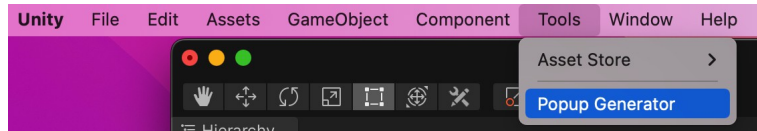


Picture 4: Integration of popup system

6. How to create new popup?

6.1. Step 1.

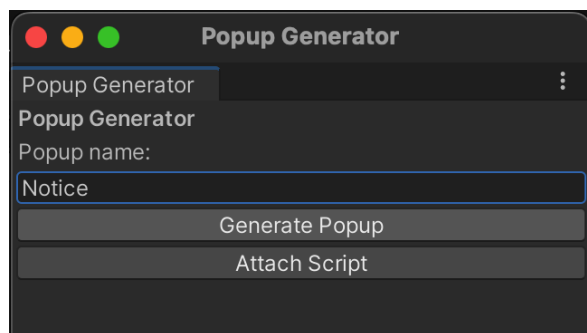
In the main menu, navigate to Tools and select Popup Generator.



Picture 5: New popup creation

6.2. Step 2.

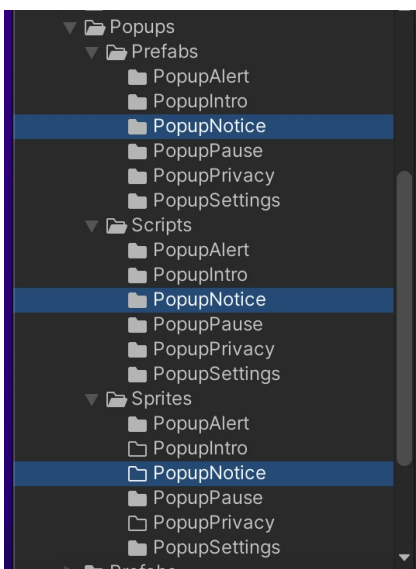
After clicking Popup Generator, a window will appear. Enter the desired popup name in the text input field - this name will be used for generating the popup. Click Generate Popup, and the plugin will create the necessary folders, duplicate the base popup prefab, and generate a base script for the new popup. Finally, click Attach Script to assign the newly created script to the new popup prefab.



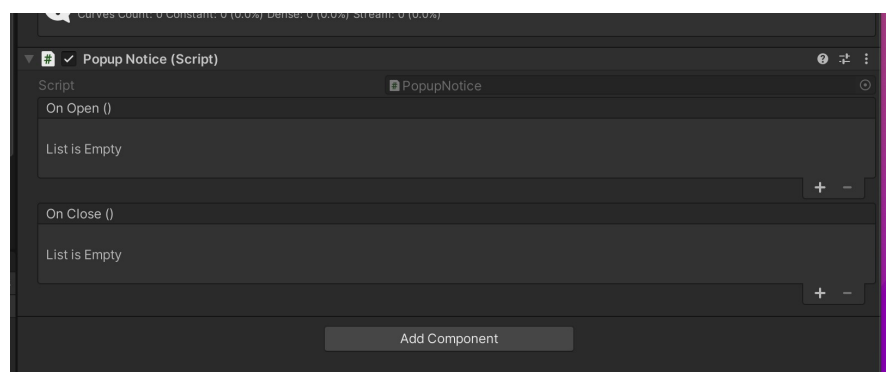
Picture 6: Popup Generator window

6.2. Step 3.

Under folder Popups under each of folder Prefabs, Scripts and Sprites are created new folders. There script will put newly created prefab and script for new popup.



Pictures 7: New folders



Picture 8: Script attachment

6.2. Step 4.

Once you've created the popup and attached the script to its prefab, you can start customizing your new popup.



Picture 9: Newly created popup

7. How to reskin GUIPack2DFree

7.1. Graphic

Almost all graphic elements are located in the Assets/GuiPack2Free/Sprites folder. To use your own images, replace the existing image files while keeping the same name, size, and PNG/PSB format. Otherwise, you may need to adjust prefabs and object positions manually.

Note: Some UI elements use sliced images, so be careful when making changes.

7.2. Audio

Audio files are located in the Assets/GuiPack2Free/Sounds folder. To use your own music and sound effects, replace the existing files in this folder while keeping the same names and WAV format.

7.3. Popups

Popups files like prefabs, scripts, graphic are stored under the folders Popups. Each popup has own folders for stored graphic, prefabs or scripts.

8. Important notices

Please note that this is a graphical UI pack containing UI components, scripts, images, effects, and animations to serve as a starting point for creating a popup system in your games or projects. Game-specific features (such as custom button interactions, custom logic, etc...) will require additional programming on your end.

The pack uses Unity's UI system, so it is highly recommended that you are familiar with it to make the most of this asset.

The UI elements in this pack use TextMesh Pro Asset, so that asset is essential.