

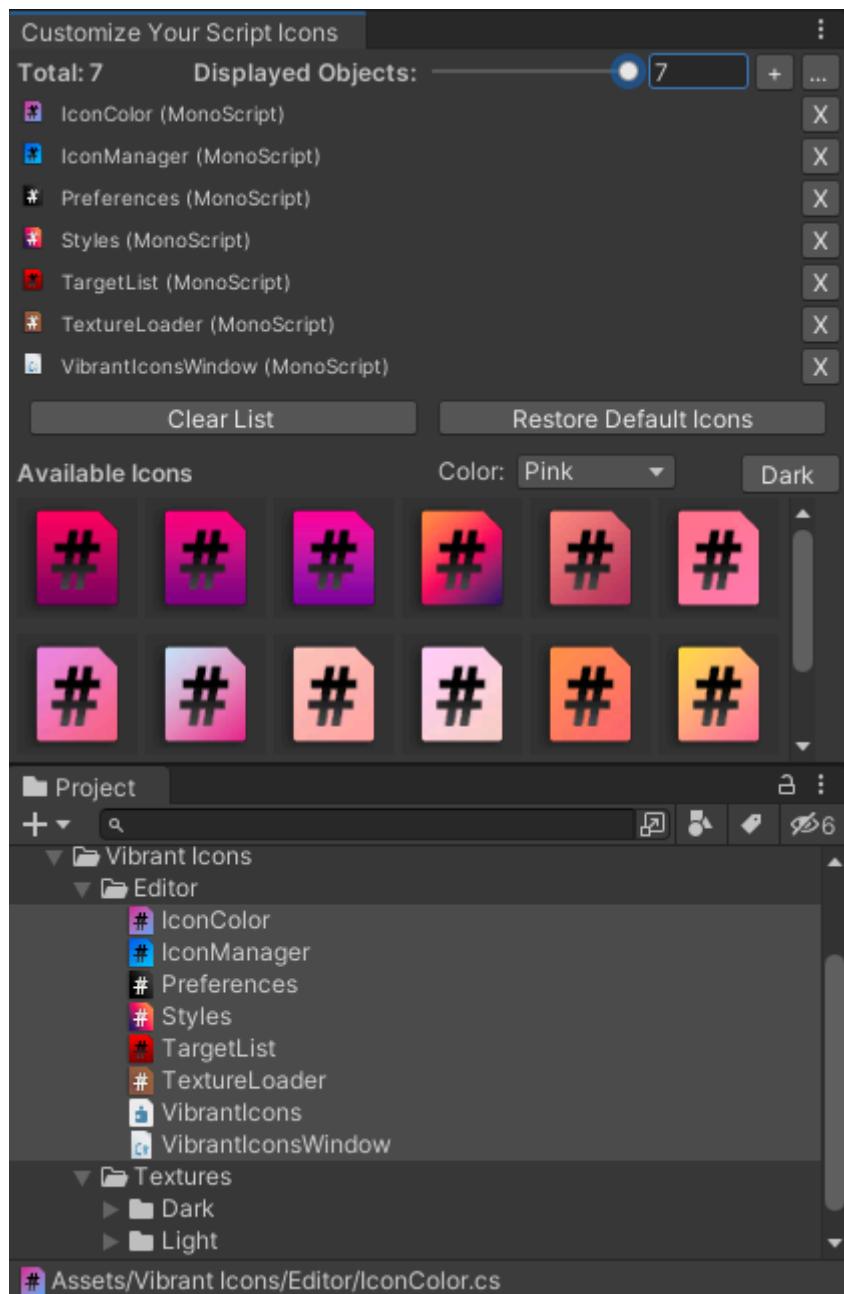
Vibrant Icons

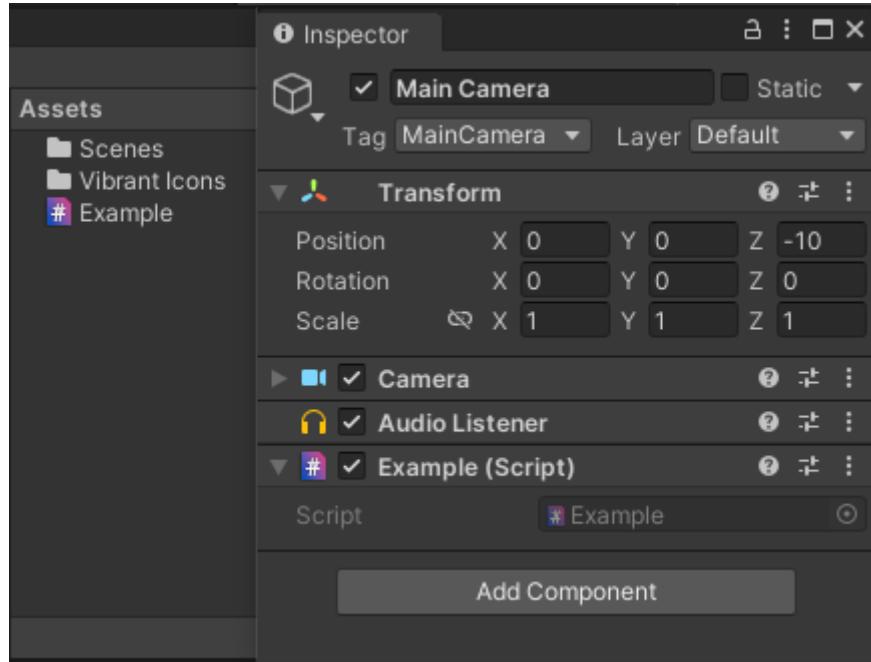
Enhance Your Experience with Custom Script Icons

Hey there! 🙋 I'm Paul, a 25-year-old student exploring the exciting world of Applied Computer Science. Recently, I've been delving into the realm of Unity to broaden my horizons. 🚀

Overview

Vibrant Icons is a Unity Editor extension that allows users to customize script icons for a more visually appealing and organized project. It provides a convenient way to assign custom icons to script files, making it easier to identify and categorize scripts within the Unity Editor.





Compatibility

- Unity Versions: The project is compatible with Unity version 2021.2.0f1 and later, and has been tested on versions 2021.2.0f1 and 2021.3.16f1.
- Platform: Developed and tested for Windows. Compatibility with macOS and Linux is expected but untested.

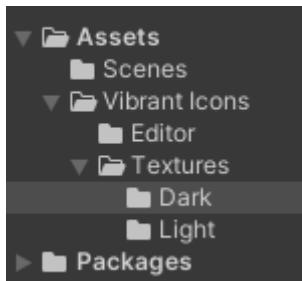
Features

- Icon Customization:**
 - ✓ Assign custom icons to script files.
 - ✓ Choose the color of displayed textures or leave it as 'All' to show all colors.
 - ✓ Choose between dark and light mode icons.
- Drag-and-Drop Functionality:**
 - ✓ Easily add script files to the customization list through drag-and-drop.
- Bulk Customization:**
 - ✓ Customize multiple script icons simultaneously.
- Restore Defaults:**
 - ✓ Restore default icons for selected script files.
- Clear List:**
 - ✓ Clear the customization list for a fresh start.
- Intuitive GUI:**
 - ✓ User-friendly interface for easy navigation and interaction.

User Guide

Installation:

1. Open your Unity project.
2. Import the purchased package into your project.
3. Ensure the presence of the "Textures" folder:
 - Make sure there is a folder named "Textures" in the same directory as the "Editor" folder (*Note: The "Editor" folder is located within the Vibrant Icons package*).



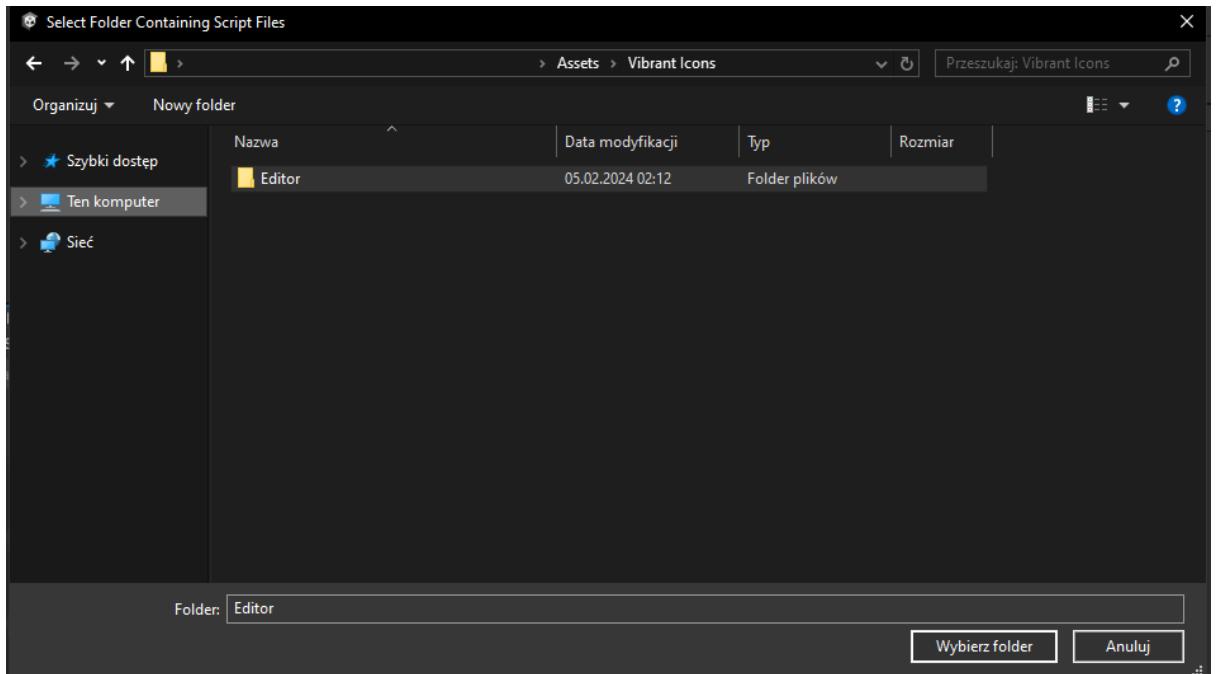
- Inside the "Textures" folder, it's recommended to maintain "Dark" and "Light" subfolders for the ability to switch between icon styles.
 - If either of these subfolders is missing, the script will automatically load all textures from the "Textures" folder.
4. Navigate to Tools > Envi > Vibrant Icons to open the customization window.



Customizing Icons:

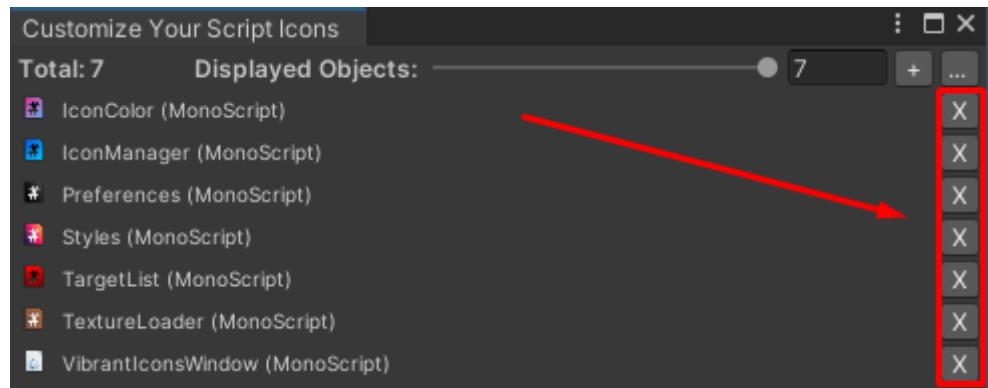
1. Adding Objects:
 - Drag script files or entire folders directly into the window to add them to the customization list.
 - Click the '+' button to select a folder containing script files for bulk customization.



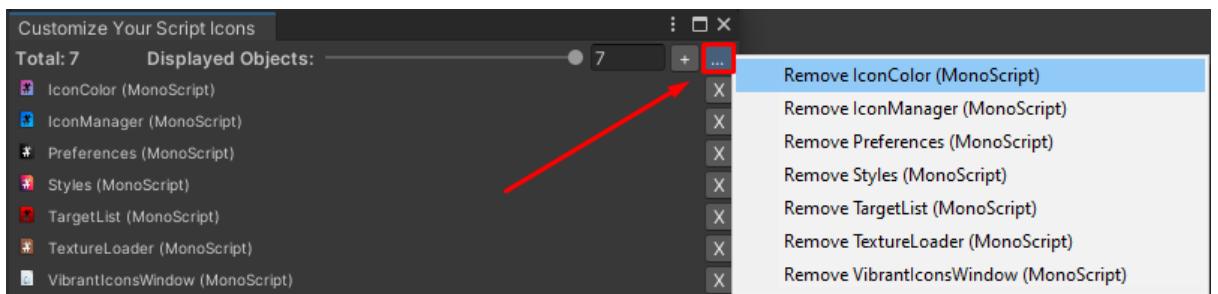


2. Removing Objects:

- To remove an object from the list, click the rightmost button 'X' next to the object's name.

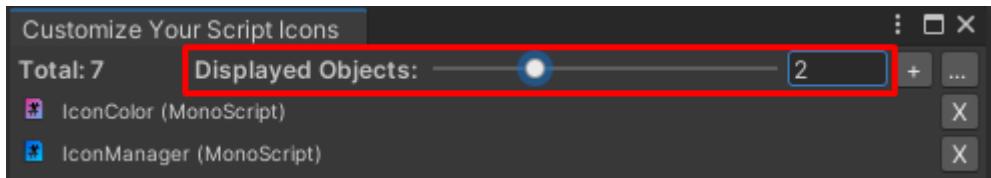


- Alternatively, expand the full list of objects using the '...' button and choose the desired item to remove from the context menu.

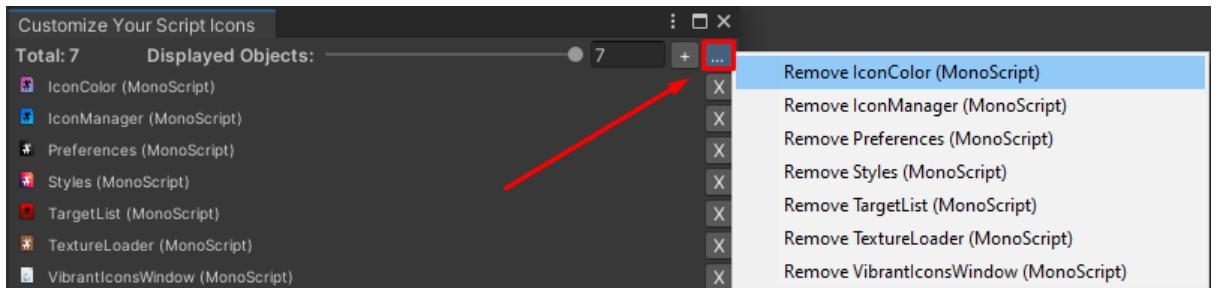


3. Adjusting Displayed Items:

- Use the slider to control the number of items displayed in the window.

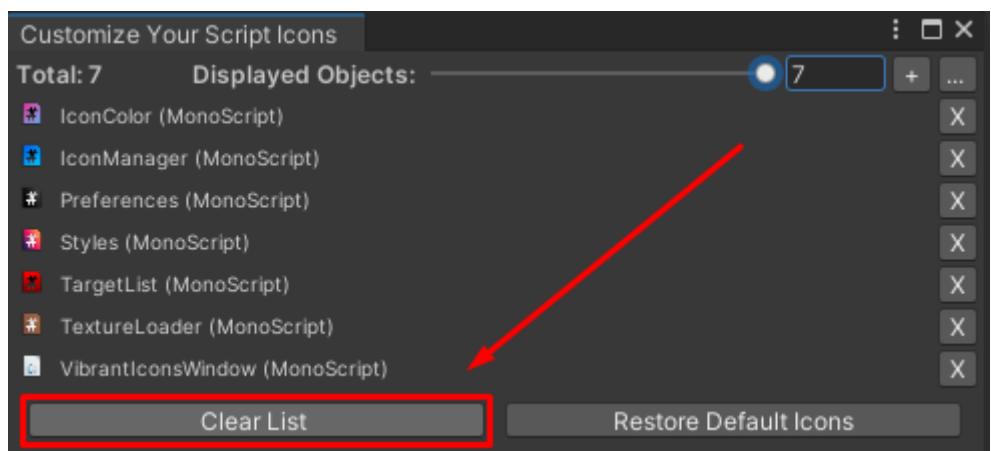


- Click '...' to expand the full list, where you can also remove individual items from the context menu.



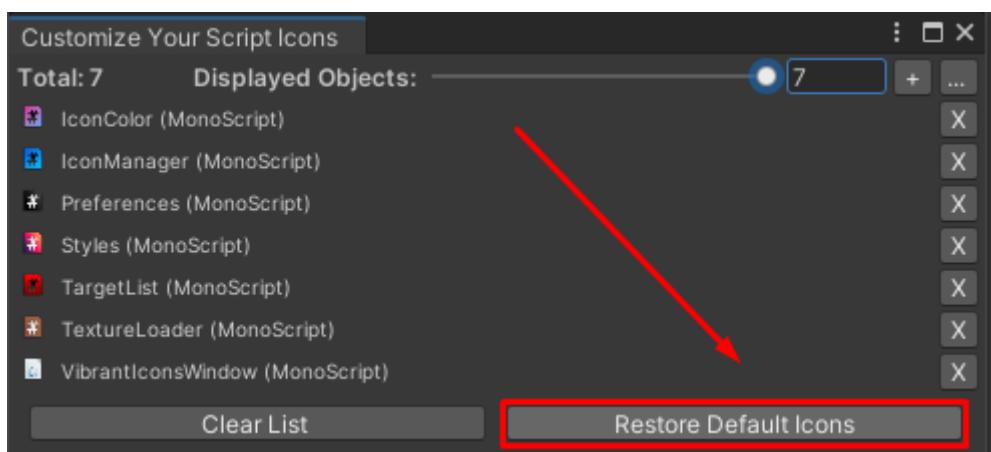
4. Clear List:

- Click the 'Clear List' button to remove all script files from the customization list.



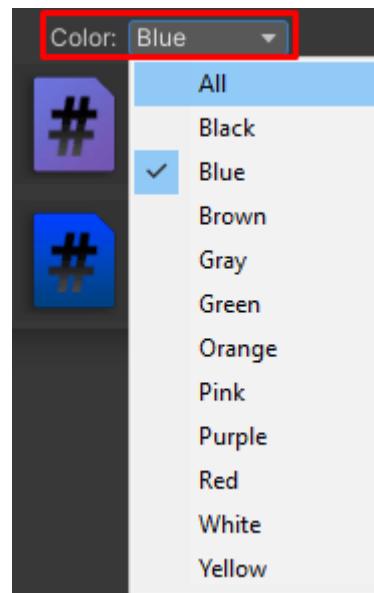
5. Restore Default Icons:

- Click 'Restore Default Icons' to revert selected script files to their default icons.



6. Select Textures Color:

- Use the dropdown menu to choose the color of displayed textures.



7. Dark / Light Mode:

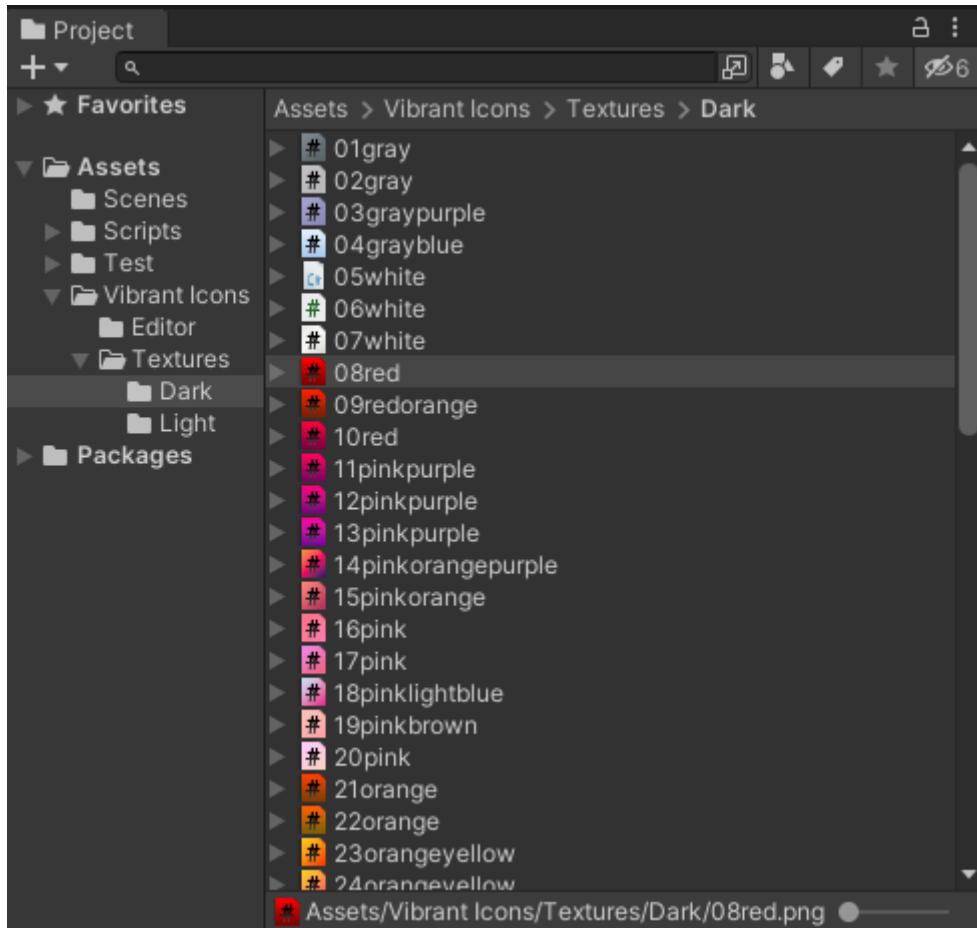
- Switch between dark and light mode icons using the corresponding button.



Modification:

The project is designed for easy modification:

- **Icon Graphics:** Add your custom icons to the "Dark" and "Light" folders in the "Vibrant Icons/Textures" directory. Make sure to include the full name of the color(s) in the filename of each icon. For example, if you have a texture named "MyIcon.png" and you want it associated with the color red, rename it to "Red_MyIcon.png".



- **GUI Layout:** For modifying the GUI layout and styling, you can make adjustments directly in the VibrantIconsWindow.cs script. Customize the layout to suit your preferences and requirements.
- **Folder and Subfolder Names:** You have the flexibility to customize the names of the "Textures", "Dark", and "Light" folders according to your project's needs. These folder names can be modified within the VibrantIconsWindow.cs script to match your desired naming conventions.

Note: Feel free to relocate the entire package folder to different directories within your project. The script automatically adapts to changes in the project's location, ensuring seamless integration regardless of folder structure adjustments.

Support

For any questions, assistance, or issues, feel free to contact me at
paweltrojanski@gmail.com. I'm here to help! 😊