

FIMPOSSIBLE GAMES



ICONS SCALER USER MANUAL

- 1. BASE KNOWLEDGE**
- 2. USING ICONS SCALER**

INTRODUCTION:

ICONS SCALER IS SYSTEM WHICH IS LOADING SPRITES, RESCALING THEM WITH SMART ALGORITHM AND MANAGING MEMORY. IT'S DEDICATED TO USE WITH UI IMAGES.

YOU CAN CHOOSE IMAGE FILES TO BE RESCALED AND USED AS SPRITES BY PROVIDING PATH TO STREAMING ASSETS, ONLINE ADDRESS, RESOURCES DIRECTORY PATH OR PROVIDING TEXTURE FILE FROM PROJECT'S ASSETS DIRECTORY.

SUPPORT FOR ADDRESSABLES IS ALSO PROVIDED, JUST IMPORT ADDRESSABLES SYSTEM TO YOUR PROJECT

IF YOU WILL LIKE THIS PACKAGE PLEASE VISIT PAGE ON ASSET STORE AND RATE OR WRITE REVIEW :)

ICONS SCALER

USER MANUAL

1 / 2 - BASE KNOWLEDGE:

ICONS SCALER IS LOADING TEXTURE FILES AND USING THEM TO GENERATE SCALED SPRITE FILES. FOR EXAMPLE YOU CAN USE ONE FILE IN 256X256 SIZE ICONS SCALER CAN GENERATE DIFFERENT GOOD LOOKING SPRITES IN DIFFERENT DIMENSIONS LIKE 32X32 OTHER IN 60X60 OTHER IN 160X160



FOR EACH ICON SIZE THERE IS GENERATED ONE SPRITE, THEN UI IMAGES CAN USE THIS ONE SPRITE IN A SHARED WAY.

ON THE LEFT - GENERATED SPRITE IN 80X80 FROM 512X512 SOURCE FILE
ON THE RIGHT - UI SCALED 512X512 FILE

ICONS SCALER SYSTEM IS MANAGING MEMORY IN EFFICIENT WAY. YOU CAN DEFINE HOW MUCH MEMORY CAN BE USED UNTIL NOT USED SPRITES SHOULD BE UNLOADED. KEEPING NOT USED SPRITES IN MEMORY IS IMPORTANT BECAUSE WHEN FOR EXAMPLE YOU CLOSE AND OPEN WHOLE INVENTORY WINDOW FEW TIMES IN ABOUT MINUTE, SPRITES WILL NOT NEED TO REALOAD EACH TIME BUT WILL BE VIEWED RIGHT AWAY.

WITH DEBUGGING VIEW YOU CAN EASILY CHECK WHAT IS GOING ON WITH YOUR LOADED SPRITES AND TEXTURES, HOW MANY IMAGES ARE USING CERTAIN SPRITES AND HOW MUCH MEMORY THEY TAKE.

YOU CAN DEFINE IF IMAGES WITH UNACTIVE GAME OBJECTS SHOULD BE TREATED AS NOT NEEDED THEN UNLOADED WHEN MEMORY LIMIT IS EXCEEDED AND LOADED AGAIN WHEN GAME OBJECT BECOMES ACTIVE AGAIN.

EACH TEXTURE CONTAINER HAVE OWN ID STORED AS HASH OF IT'S PATH TO TEXTURE FOR MANAGER TO QUICKLY NAVIGATE BETWEEN THEM AND TO IDENTIFY THEM WHEN NEEDED.

EACH SPRITE HAVE OWN ID FOR IT'S TEXTURE CONTAINER TO QUICKLY NAVIGATE AND IDENTIFY THEM.

TO GENERATE SPRITE ALGORITHM NEED TEXTURE2D DATA WITH READ/WRITE ENABLED, THIS OPTION IS TAKING MORE MEMORY BUT AFTER WE GET PIXEL ARRAY OF TEXTURE, SYSTEM IS UNLOADING TEXTURE WITH READ/WRITE ENABLED SO WE USE LOWEST AMOUNT OF MEMORY.

Containers		Sprites	
Loaded	cloaks.PNG 54x54 F: Bilinear ID: 120096 0.01mb 4 References Active: 4 Last Use: 14		
Loaded	gloves.PNG 54x54 F: Bilinear ID: 120096 0.01mb 3 References Active: 3 Last Use: 15		
Loaded	ingots.png 54x54 F: Bilinear ID: 120096 0.01mb 5 References Active: 5 Last Use: 16		
Loaded	book.PNG 54x54 F: Bilinear ID: 120096 0.01mb 5 References Active: 5 Last Use: 17		
Unloaded	RPG_inventory_icons/book.PNG ID: 304688 137x137 References: 1 Filter: Bilinear Active: 0		
Loaded	helmets.png 54x54 F: Bilinear ID: 120096 0.01mb 4 References Active: 4 Last Use: 18		
Loaded	f.PNG 54x54 F: Bilinear ID: 120096 0.01mb 3 References Active: 3 Last Use: 19		
Loaded	coins.png 54x54 F: Bilinear ID: 120096 0.01mb 3 References Active: 3 Last Use: 20		
Loaded	apple.png 54x54 F: Bilinear ID: 120096 0.01mb 5 References Active: 5 Last Use: 21		
Loaded	boots.png 54x54 F: Bilinear ID: 120096 0.01mb 4 References Active: 4 Last Use: 22		
Loaded	necklace.PNG 54x54 F: Bilinear ID: 120096 0.01mb 3 References Active: 3 Last Use: 23		
Unloaded	RPG_inventory_icons/necklace.PNG ID: 304688 137x137 References: 1 Filter: Bilinear Active: 0		
Loaded	shield.png 54x54 F: Bilinear ID: 120096 0.01mb 3 References Active: 3 Last Use: 24		
Unloaded	RPG_inventory_icons/shield.png ID: 304688 137x137 References: 1 Filter: Bilinear Active: 0		
Loaded	sword.png 54x54 F: Bilinear ID: 120096 0.01mb 3 References Active: 3 Last Use: 25		

WHEN LOADING TEXTURE WHICH DON'T HAVE READ/WRITE ENABLED (ONLY WHEN LOADING FROM RESOURCES OR ASSETS FOLDER TEXTURE FILE) WE RENDERING READABLE TEXTURE WITH 'RENDER TEXTURE' GETTING PIXELS ARRAY THEN UNLOADING RENDERED TEXTURE.

ICONS SCALER

USER MANUAL

2 / 2 - USING ICONS SCALER:

TO USE ICONS SCALER, ADD **FIcons_Manager** COMPONENT TO NEW EMPTY GAME OBJECT ON THE SCENE.

TO GENERATE SPRITE IN NEW SIZE YOU CAN USE **FIcons_Loader** COMPONENT, BUT IT'S RECOMMENDED TO LOAD EVERYTHING FROM CODE WITH METHOD LIKE:

```
using UnityEngine;
using UnityEngine.UI;
using FIMSpace.FIcons;

public class TestLoading : MonoBehaviour
{
    public Image myUIImage;

    void Start ()
    {
        int imageSize = (int)myUIImage.rectTransform.sizeDelta.x; // Size of UI image
        FIcons_Manager.Get.LoadSpriteManaged("FIconsScalerExample/Mask 1.png", myUIImage, imageSize);
    }
}
```

USING ICONS SCALER NAMESPACE

IMAGE TO WHICH WILL BE ASSIGNED NEW SPRITE

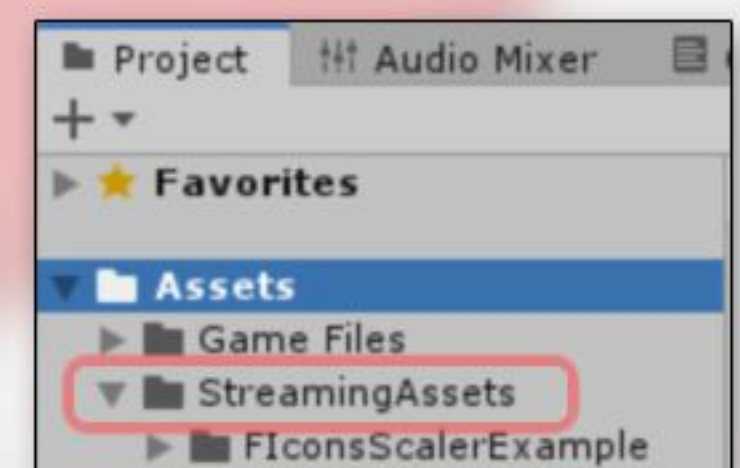
PATH TO STREAMING ASSETS It also can be any number like 60

REQUESTING IMAGE TO BE LOADED

ICONS MANAGER WILL TAKE CARE OF THE REST.
TO UNLOAD IMAGE YOU DESTROY GAME OBJECT WITH IT, OR DISABLE IT IF YOU DEFINED TO UNLOAD DEACTIVATED OBJECTS IN ICONS MANAGER INSPECTOR WINDOW.
(IMAGES WILL BE UNLOADED WHEN MEMORY USE LIMIT EXCEEDS)

TO BE ABLE TO LOAD FROM STREAMING ASSETS, YOU NEED TO HAVE 'STREAMINGASSETS' DIRECTORY INSIDE YOUR ASSETS FOLDER, AFTER MAKING BUILD UNITY IS AUTOMATICALLY MOVING FILES FROM THIS DIRECTORY TO BUILD DIRECTORY.

WITH LOADSPRITEMANAGED() METHOD YOU CAN DEFINE FROM WHERE YOU WANT TO LOAD TEXTURE. CHOOSE IMAGE TO WHICH SPRITE SHOULD BE ASSIGNED AFTER LOADING SUCCESS. DEFINE DESIRED WIDTH AND HEIGHT OF SPRITE. DECIDE IF USE OR NOT ANIMATION WHEN SPRITE IS BEING GENERATED.



STREAMING ASSETS FOLDER IN ASSETS DIRECTORY OF THE PROJECT

FILTER OPTION IS UNITY'S WAY TO VIEW IMAGES WITH **POINT** YOU GET SHARPEST LOOK BUT WHEN YOU MOVE IMAGE IN VIEW SLOWLY, MOTION WILL BE LESS SMOOTH THAN WHEN YOU USE **BILINEAR** OR **TRILINEAR**, ALSO THIS TWO OPTIONS ARE NICE WHEN YOU NEED TO SCALE SPRITE A BIT AFTER RESCALING WITH **ICONS SCALER**.
SET NATIVE SIZE WILL SET IMAGE COMPONENT SIZE TO BE THE SAME AS GENERATED SPRITE.