synty®

INTERFACE SCI-FI MENUS

User Guide

Welcome to the **Interface Sci-Fi Menus** pack. Found within a glowing mid-poly pyramid in orbit around a dying star, our intrepid space explorers discovered Alien artifacts such as menu screens, frames, buttons, popups and icons, as well as sprites made of unknown alloys which can be assembled into sci-fi crests like the one above.

The **Sci-Fi Menus** pack is intended to work alongside other INTERFACE packs such as **Sci-Fi Soldier HUD**.

Exploring UI for the first time? Head to the official synty studios YouTube channel for **INTERFACE** tutorials.

Any questions, please contact us via support@syntystudios.com

www.syntystore.com

1. Package Info

- This package was created using Unity Version **2022.3.0f1**
- All components, prefabs, and sprites were created with a target resolution of **4K** (3840 x 2160) in mind.
- All sprites and textures are in **.PNG** format
- Source meshes are not included
- The provided sample scripts serve only to demonstrate how you might implement art assets and UI components. They are not intended to be used in a production environment.

2. Package Contents + Structure

- UI Component prefabs, sprites, materials and animations
- Example scenes
- Gameplay Icons
- Input Icons - **Xbox**, **PlayStation**, **Switch**, **Keyboard & Mouse**, **Steam Deck**

3. Requirements

This package utilises the following packages:

- **TextMeshPro** - included with Unity Editor
- **Unity UI** (**UGUI**)
- **UI Extensions** - included with our asset pack, for more details visit: https://github.com/Unity-UI-Extensions/com.unity.uiextensions

4. Quick Start

Note: Check the Installation requirements if you are importing the package via the package manager.

Once the package is imported:

- All prefabs can be found in **Assets/Synty/InterfaceSciFiMenus/Prefabs**
- Sample scenes are available in **Assets/Synty/InterfaceSciFiMenus/Samples/Scenes**

The **00_Demo_SciFiMenus_Title** scene is an overview of the pack contents. Loading this scene and pressing Play will allow you to navigate to all the other scenes in the pack.

All scenes numbered **01-36** show each section in isolation. We recommend using these scenes to select prefabs you may wish to use in your own project.

5. Adding a Menu Screen

Here we'll go through the fundamental steps for setting up a UI Canvas and add a screen prefab to it.

You can also check out our Unity HUD Tutorial, which takes you through the basics of using Unity GUI. We're always adding more tutorials, so let us know on our Discord if there's anything you'd like covered!

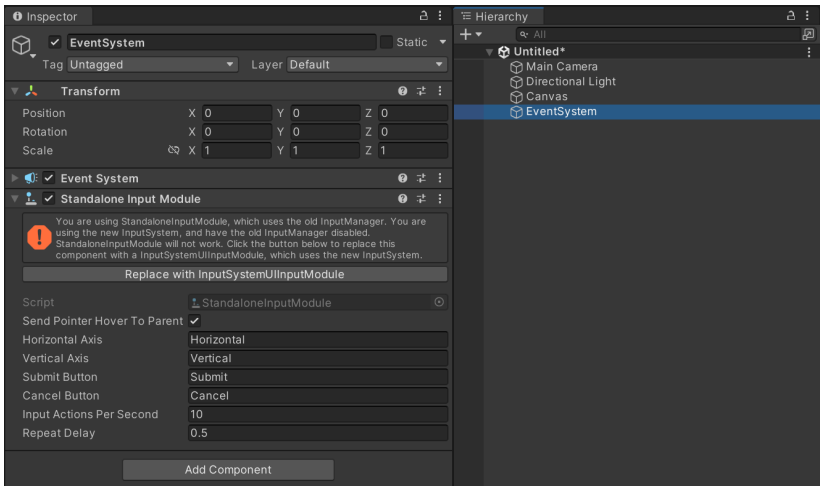
Creating a Canvas

To set up a screen, you will need to create a UI Canvas. Follow these steps:

1. Create a new scene (**File** > **New Scene**) or open an existing scene.
2. In your scene, create a Canvas object (**GameObject** > **UI** > **Canvas**). This is where all UI objects will be placed.

   This will also create an **EventSystem** object - allowing inputs to be sent to all interface objects.

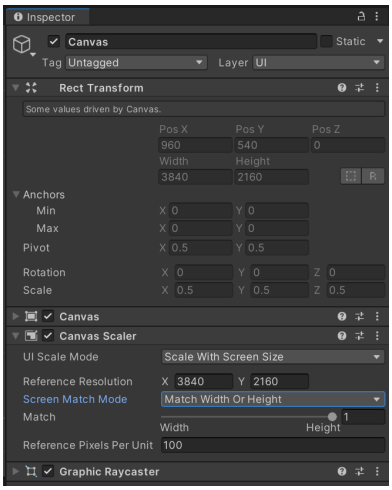
Note: If you are using Unity's Input System package, you will be prompted with the below message on your EventSystem object. Click the **Replace with InputSystemUIInputModule** button before continuing.



3. On the Canvas object, apply these settings on the **Canvas Scaler** component in the Inspector window:

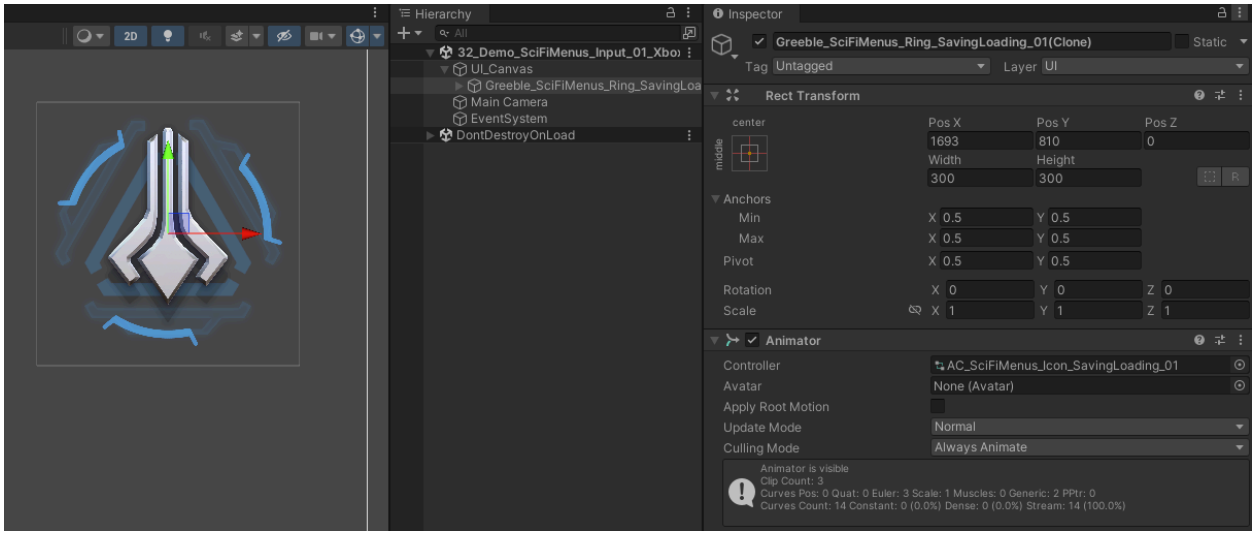
   - Set **UI Scale Mode** to **Scale with Screen Size**
   - Set **Screen Match Mode** to **Match Width Or Height**
   - Set **Match** to **1** (Height)
   - Set **Reference Resolution** to **X: 3840 Y: 2160**

   Note: We have created our prefabs with a 4K screen size as reference - this setting will scale our prefabs accordingly.



Adding UI components

1. In the Project tab (**Window** > **General** > **Project**), navigate to our Greebles folder: **Assets/Synty/InterfaceSciFiMenus/Prefabs/Greebles/**
2. Locate the **Greeble_SciFiMenus_Ring_SavingLoading_01** prefab.
3. Drag the object into the **Canvas** object in the Hierarchy tab.

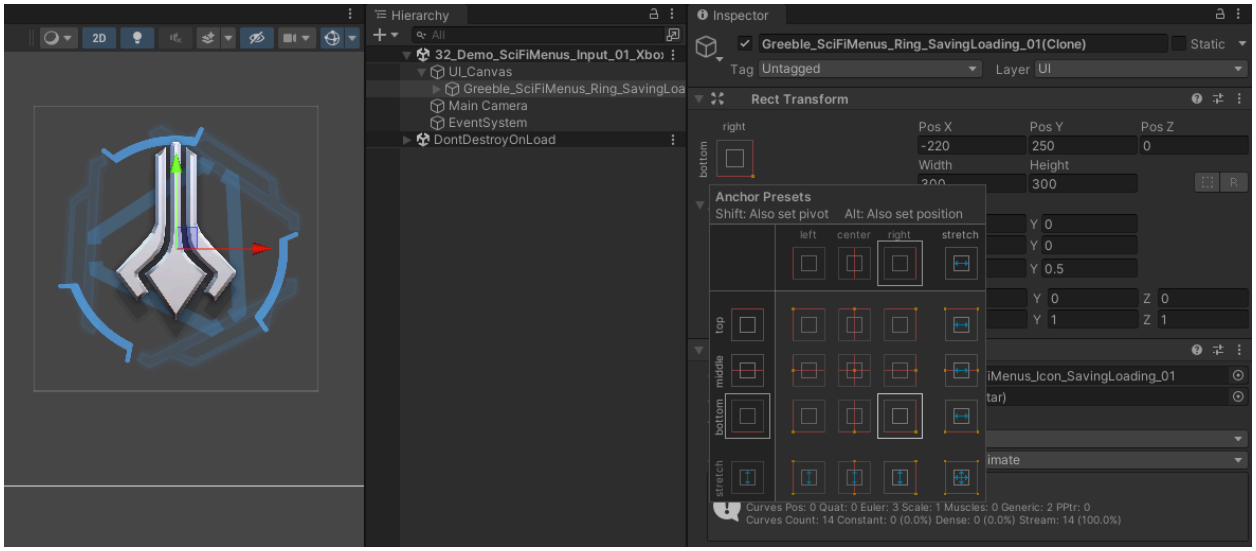


Positioning / Anchoring UI components

Note: When working in the Scene tab, our UI Canvas will reflect the aspect ratio of the Game tab. Therefore, before continuing in this section, you may wish to set your Game view's aspect ratio to **16:9**.

To anchor the health bar to bottom-right:

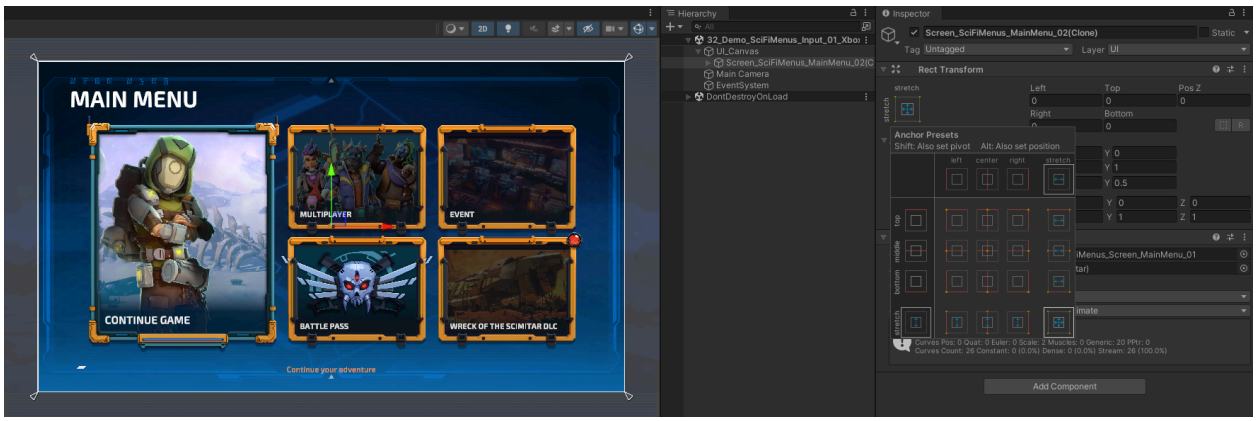
1. In the Hierarchy, find our prefab instance.
2. Position it at the bottom-right of the canvas.
3. In the Inspector, click on the anchor icon on the RectTransform component.
4. Select the inner bottom-right position:



Now, as your game window resizes, the prefab will always be positioned relative to the bottom-right corner of the screen.

When adding full-screen prefabs such as **Screen_SciFiMenus_MainMenu_02**:

- Set the anchor icon to **Stretch X and Y** (bottom-right)
- Check that your **Left**, **Right**, **Top** and **Bottom** values are all set to **0**

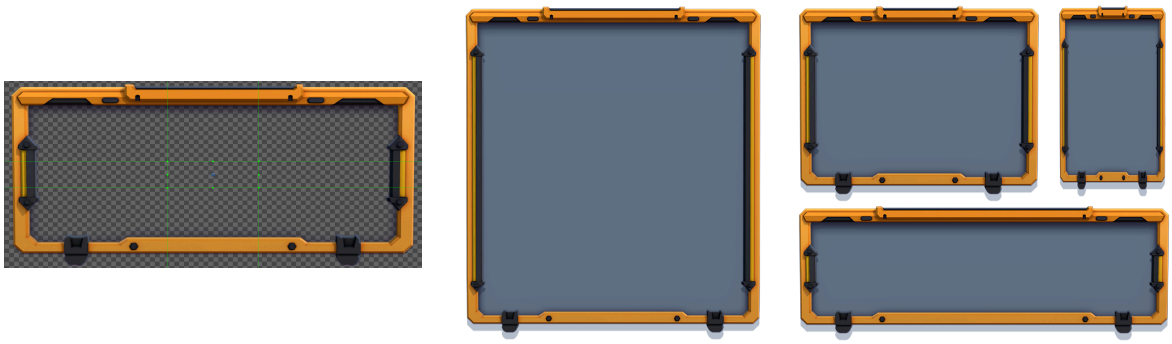


Congratulations - you are on your way towards creating a fantastic UI for your game!

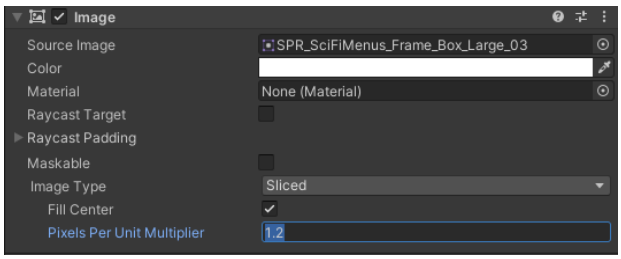
6. Naming conventions

AC_	Animator Controller
ANIM_	Animation
FX	Visual Effect
Greeble	A cosmetic detail sprite, used to add visual interest to a component
HUD	Heads-Up Display
ICON_	Icon
MAT_	Material
SPR_	Sprite
UI	User Interface

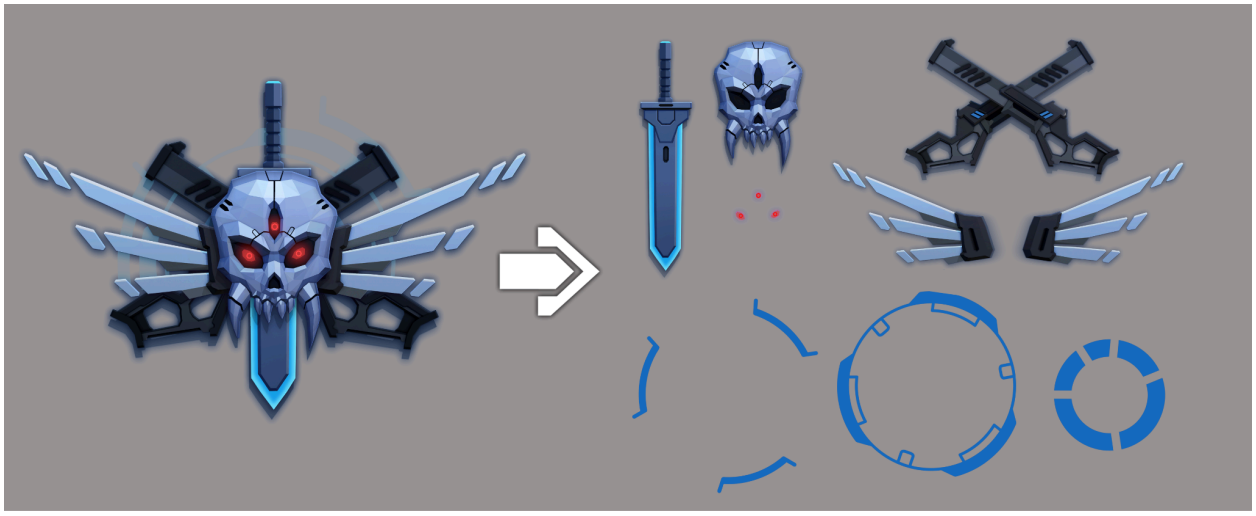
7. Sprites and 9-slicing



- Where possible, sprites have been **9-sliced** via Unity's **Sprite Editor**. This allows for components to be resized.
- The components make use of Unity's **Pixels Per Unit Multiplier** function to change the width of boxes:

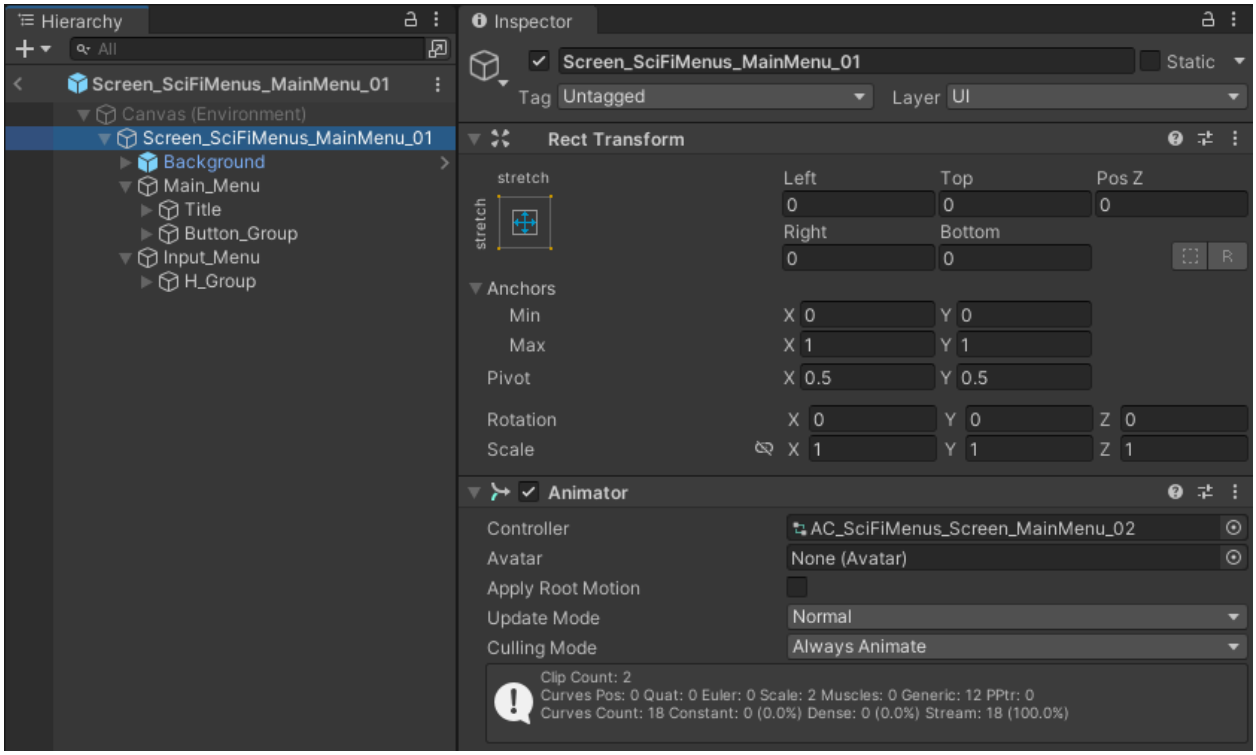


8. Modular Sprites (Greebles and Crests)



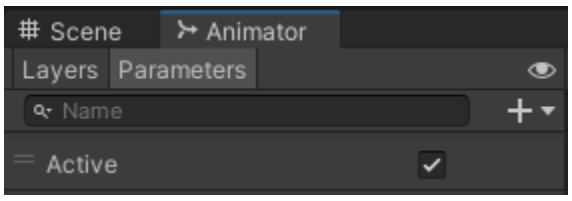
Sprites have been created to allow for as much modularity as possible. Sprites intended as cosmetic have the word ‘greeble’ in their filename.

9. Animated Prefabs



- Many prefabs feature animations
- **Animator Controllers** generally sit on the top layer of prefabs.
- Where possible, Animator Controllers are set up with a boolean parameter named **Active**

You can preview these animations while playing the build in the Game window, by navigating to the Animator Controller in the Hierarchy window, then toggling parameters in the Animator window:



Note: With our Animator selected, you can browse the animations and their keyframes in the Animation tab (**Window** > **Animation** > **Animation**).

10. Input Icons

The package contains various input icons for:

- **Xbox**
- **PlayStation**
- **Switch**
- **Keyboard & Mouse**
- **Steam Deck**

Xbox / **PlayStation** / **Switch** button glyphs are baked into the icons.

Keyboard glyphs are NOT baked - these are set up as prefabs which resize based on their content text. We feel this is the optimal way to present them, as it allows for localised keyboard inputs.

11. Fonts

This pack contains and uses variations of the following fonts:

- **Bungee**, designed by **David Jonathan Ross**
- **Exo 2**, designed by **Natanael Gama**
- **Gugi**, designed by **TAE System & Typefaces Co.**
- **Quantico**, designed by **MADType**
- **WDXL Lubrifont SC Regular**, designed by **NightFurySL2001**
- **Zen Dots Regular**, designed by **Yoshimichi Ohira**

All fonts are sourced from Google Fonts and are licensed under the **SIL Open Font License, Version 1.1.**

This license is included in this package, and is also available with a FAQ at:
http://scripts.sil.org/OFL

12. Unity UI Extensions

Several pack assets make use of the **UIParticleSystem** script from the Unity UI Extensions project, to allow particles to display in the UI. As such, we've included the **Unity UI Extensions** package within our pack:

Repository link: **https://github.com/Unity-UI-Extensions/com.unity.uiextensions**

Unity UI Extensions License (BSD3)

Copyright (c) 2019

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

13. Acknowledgement of Copyrights

- **Xbox™** and the **Xbox™** logos are a trademark of **Microsoft® Corporation**
- **Nintendo Switch™**, **Switch 2™**, **Joy-Con™** and their logos are trademarks of **Nintendo of America Inc.**
- **PlayStation™** is a registered trademark of **Sony Group Corporation**
- **Steam™** and **Steam Deck™** are trademarks and/or registered trademarks of **Valve Corporation**

14. Terms of Use

For all the terms and conditions of your licence, please review the EULA you agreed to when purchasing this asset. Each asset store has its own EULA. It is important to understand where you purchased this asset from and what EULA you are bound by.

For convenience here are links to EULAs for common stores Synty Assets are purchased from.
Synty Store: https://syntystore.com/pages/licences-overview
Unity Asset Store: https://unity.com/legal/as-terms
Unreal Marketplace: https://www.unrealengine.com/en-US/eula/content
FAB: https://www.fab.com/eula