INTERFACE
SCI-FI SOLDIER HUD

User Guide

Welcome to Synty's **Interface SciFi Soldier HUD** pack! Strap on your tactical head gear and channel your inner future soldier, for in the distant future, there is only UI…

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

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

www.syntystore.com

1. Package Info

- This package was created using Unity Version **2021.1.0f1**
- 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

2. Package Contents + Structure

- UI Component prefabs, sprites and example scenes
- HUD Example scenes
- Gameplay Icons
- Input Icons - **Xbox**, **PlayStation**, **Switch**, **Keyboard & Mouse**

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/InterfaceSciFiHUD/Prefabs**
- Sample scenes are available in **Assets/Synty/InterfaceSciFiHUD/Samples/Scenes**

The **00_Demo_SciFi_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 **01-25** show each section in isolation. We recommend using these scenes to select prefabs you may wish to use in your own project.

5. Building your own HUD

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

Or, you can check out our Unity HUD Tutorial, which takes you through the basics of creating your first HUD.

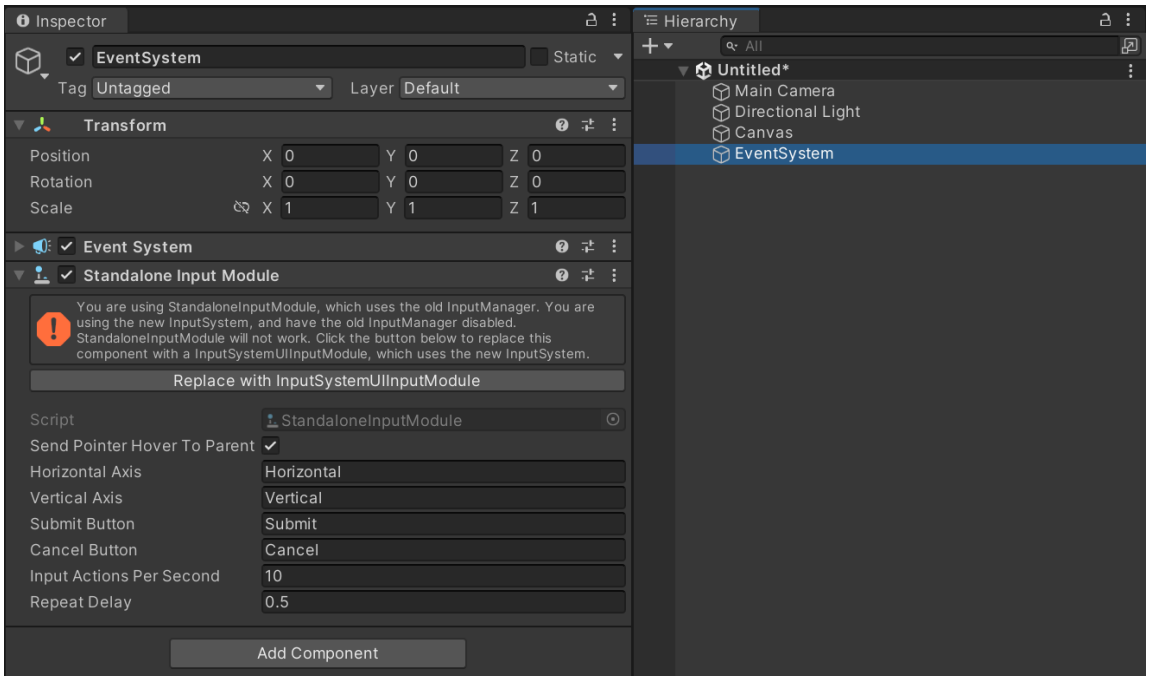
Creating a Canvas

To set up your own HUD, 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.

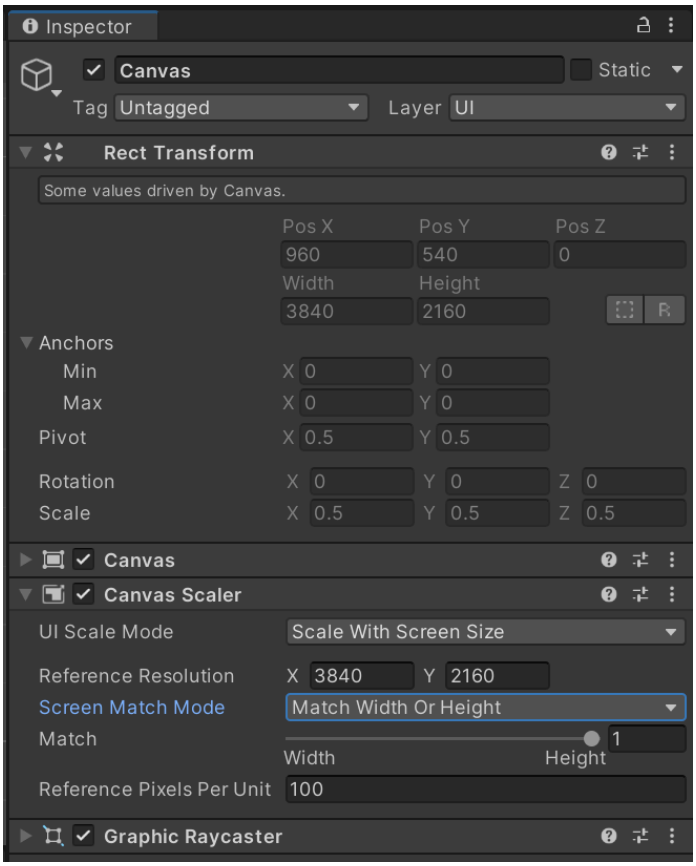
   Note: If you are using Unity's Input System package, you will be prompted with the below message on your EventSystem object. Click the **Replace with InputSystemUIInputModule** button before continuing.



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

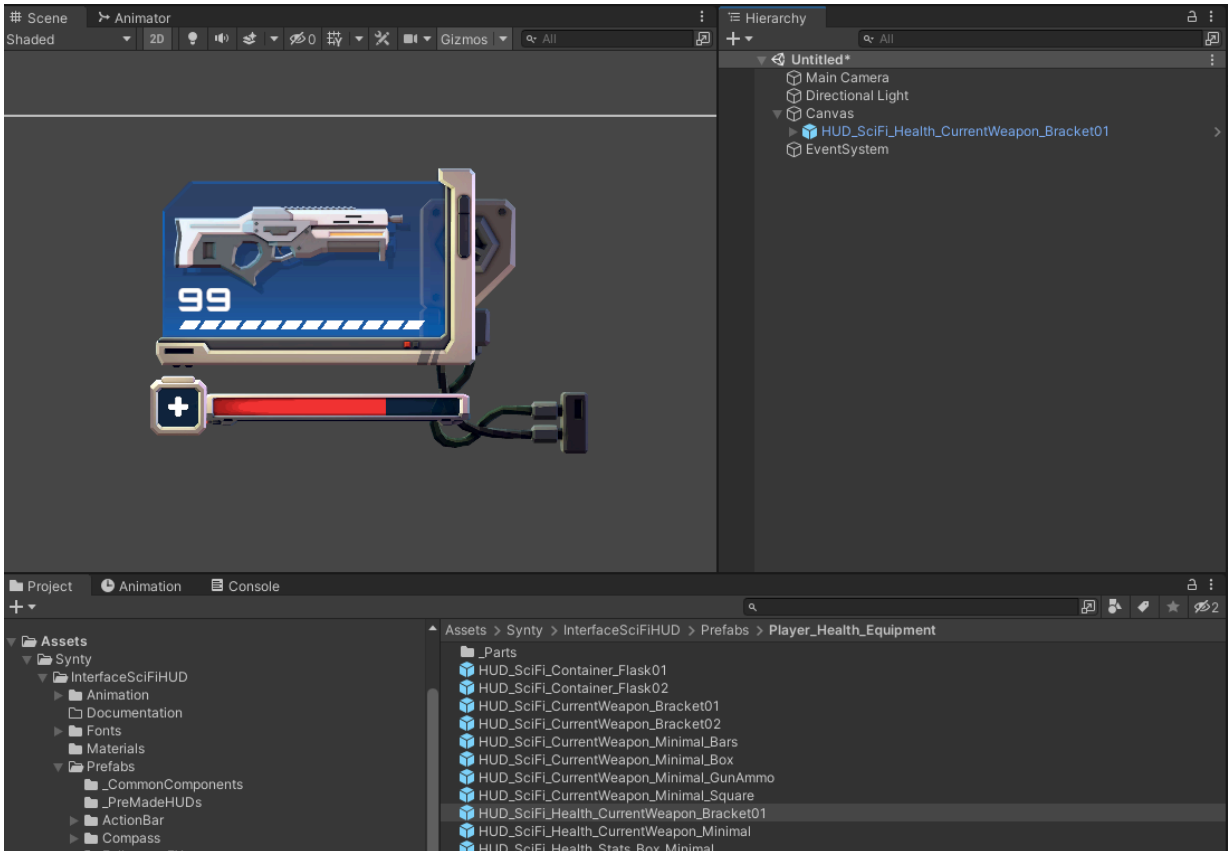
   - Set **UI Scale Mode** to **Scale with Screen Size**
   - Set **Screen Match Mode** to **Match Width Or Height**
   - Set **Match** to **1** (Height)
   - Set **Reference Resolution** to **X: 3840 Y: 2160**

   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 Prefabs folder: **Assets/Synty/InterfaceSciFiHUD/Prefabs/Player_Health_Equipment/**
2. Locate the **HUD_SciFi_Health_CurrentWeapon_Bracket01** prefab.
3. Drag the object into the **Canvas** object in the Hierarchy tab.

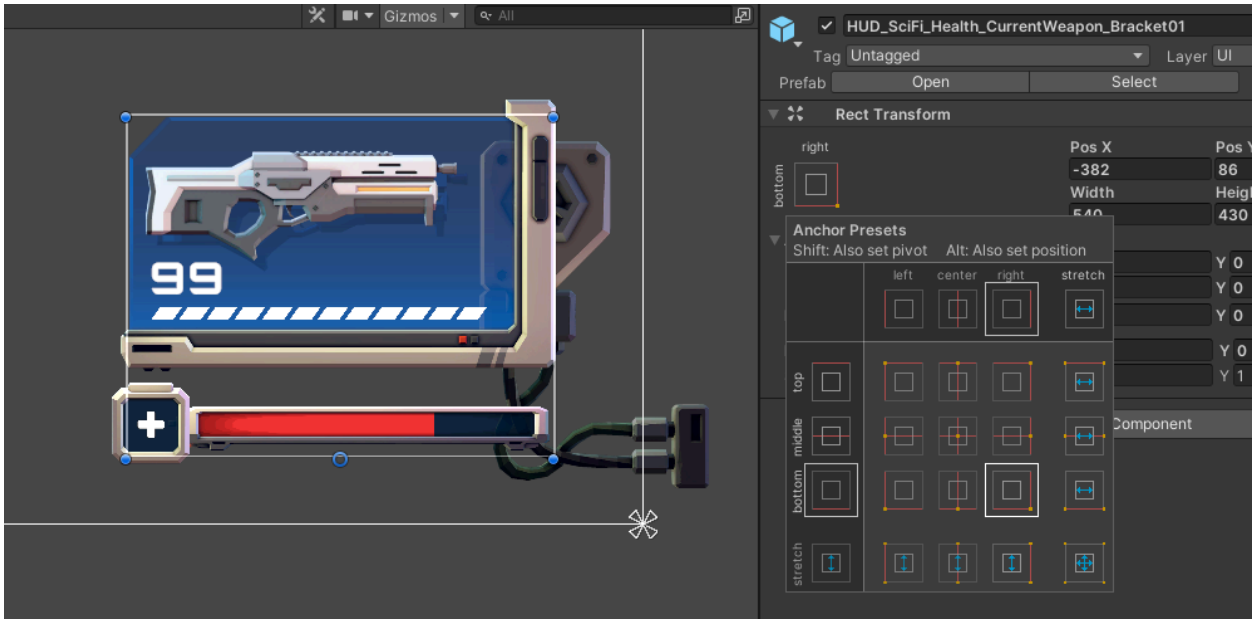


Positioning / Anchoring UI components

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 the bottom-right:

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



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

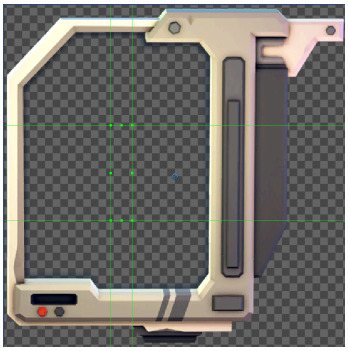
Congratulations - you are on your way to creating a fantastic HUD for your game!

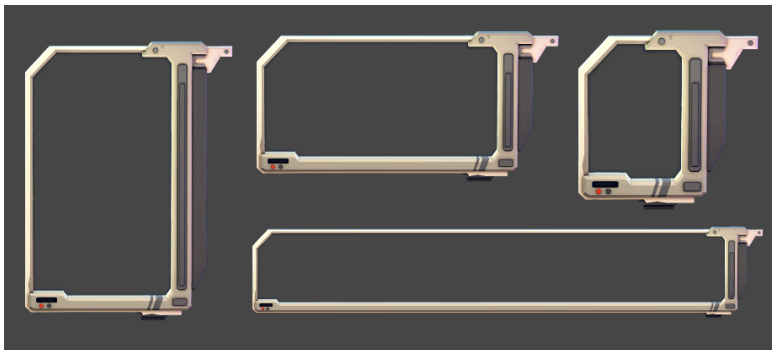
If you would like to try a fully-featured HUD with more screen elements, check out the **Prefabs/_PreMadeHUDs** folder, or take a look at our **Samples/Scenes** folder for some example arrangements.

6. Naming conventions

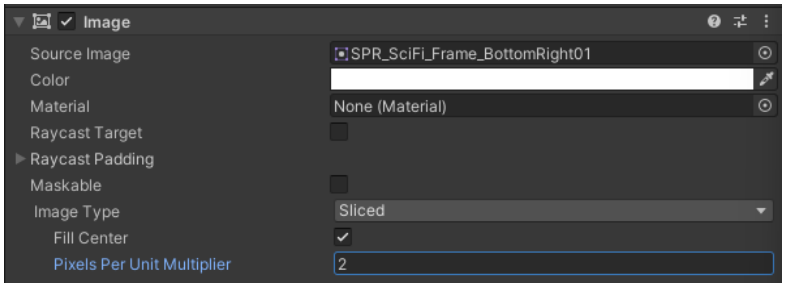
AC_	Animator Controller
ANIM_	Animation
FX	Visual Effect
Greeble	A cosmetic detail sprite, used to add visual interest to a component
HUD	Heads-Up Display
ICON_	Icon
MAT_	Material
SPR_	Sprite
UI	User Interface

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:



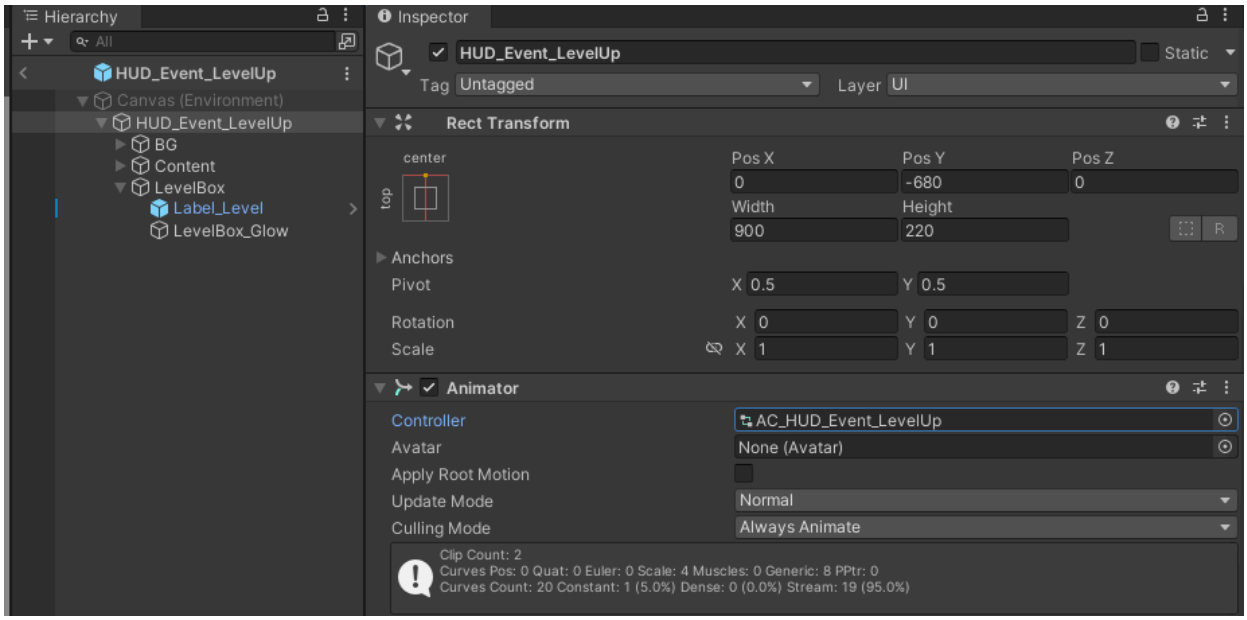
8. Modular Sprites (Greebles)

The HUD sprites have been created to allow for as much modularity as possible. Don't like those cables hanging off the edge of the components? Just delete 'em.

Sprites intended as cosmetic have the word 'greeble' in their filename.

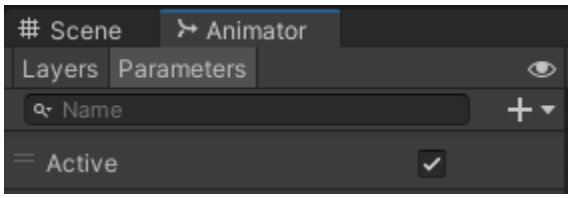
The HUD sprites are generally **NOT** created at power-of-two resolution. They are intended to be atlased once you select the sprites that suit the needs of your project.

9. Animated Prefabs



- Many of the HUD 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. 2D POLYGON Asset Icons

This pack contains rendered 2D icons of Synty POLYGON assets. These assets and their meshes are not included in this pack.

Please use syntysearch.com and search for the icon filenames for more information about the 3D assets and POLYGON packs.

11. Input Icons

The package contains various input icons for:

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

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.

12. Fonts

This pack contains and uses variations of the **Exo 2.0** and **Orbitron** fonts:

- **Exo2.0-Regular**
- **Exo2.0-Semibold**
- **Orbitron-ExtraBold**

Exo 2.0 was created by **Natanael Gama** and is sourced from Google Fonts. It is licensed under the **SIL Open Font License, Version 1.1**

Orbitron was created by **Matt McInerney** and is sourced from Google Fonts. It is 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

13. Unity UI Extensions

A couple of the 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.

14. Acknowledgement of Copyrights

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

15. 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