

Scaling UI for Different Devices

Learn how you can set up your UI to scale to different devices automatically using DPI Scaling.



When working on the UI for a project, you may already have a targeted device in mind for releasing your project. In fact, you may want to release on multiple devices or platforms. In Unreal Engine, the automatic resolution-independent UI scaling using **DPI Scaling** rules enables you to create a resolution scale for multiple devices.

DPI Scaling enables you to define a **DPI Scale Rule** and a **DPI Curve** to automatically scale your UI elements based on varying resolutions. The **DPI Scale Rule** determines which scale to apply, and the **DPI Curve** contains different resolutions and respective scale values. Using these options makes it quick and easy to translate your UI screens from devices like a phone to a tablet or a PC, with a range of device resolutions.

In this how-to, we will set up our initial aspect ratio scale, and then add another device ratio to the scale.



For more information on DPI Scale Rules and Curve, see [DPI Scaling](#).

Scaling the UI to 1

To scale the UI correctly, you need to designate an aspect ratio range for the first device and set its scale to 1.



For this how-to, we are using the **First Person Template**. However, you can use any project you wish.

1. Navigate to **Content Browser > Add New > User Interface**, create a **Widget Blueprint** called **DPI_Widget**.
2. Inside **DPI_Widget**, under the **Screen Size** drop-down menu, select a device aspect ratio.

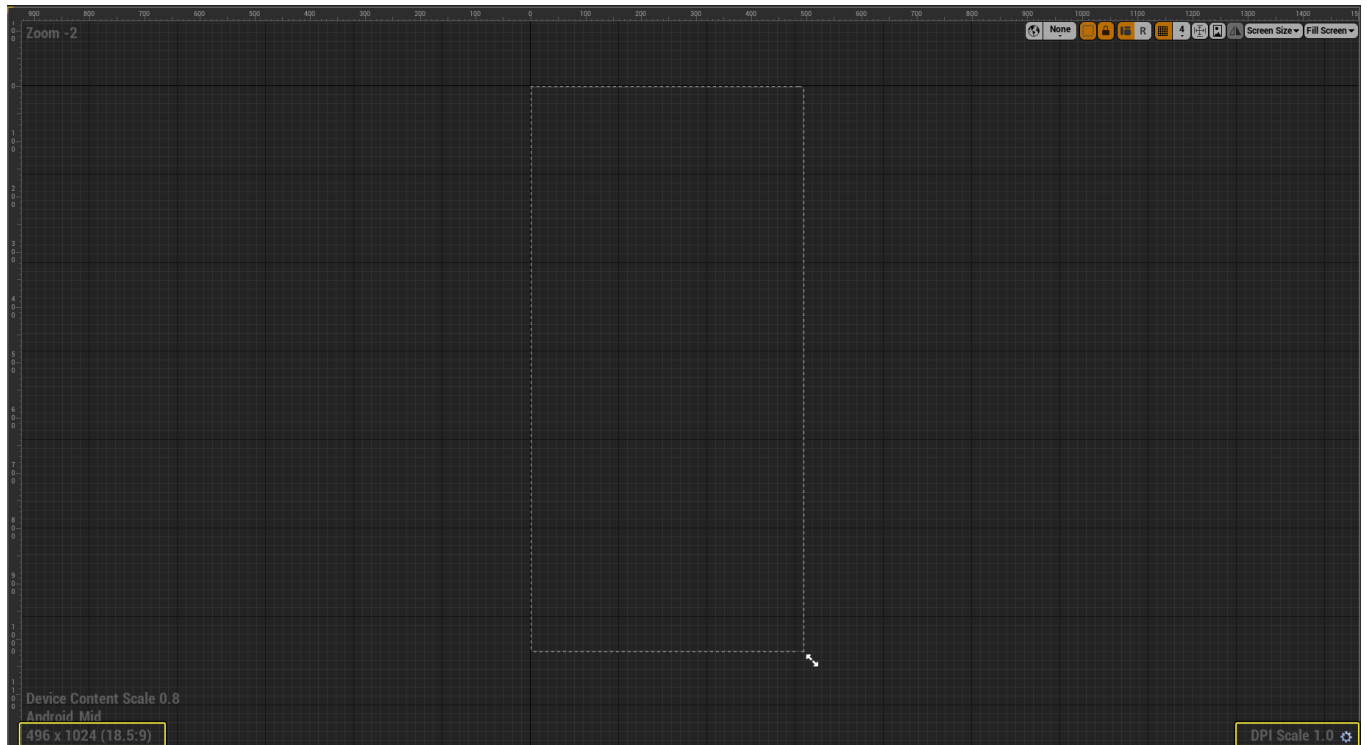


For this example, we are going to use an Andriod phone as our first device for this guide.



The Screen Size options automatically update in UE4 based on approved and supported devices at the time of each version release.

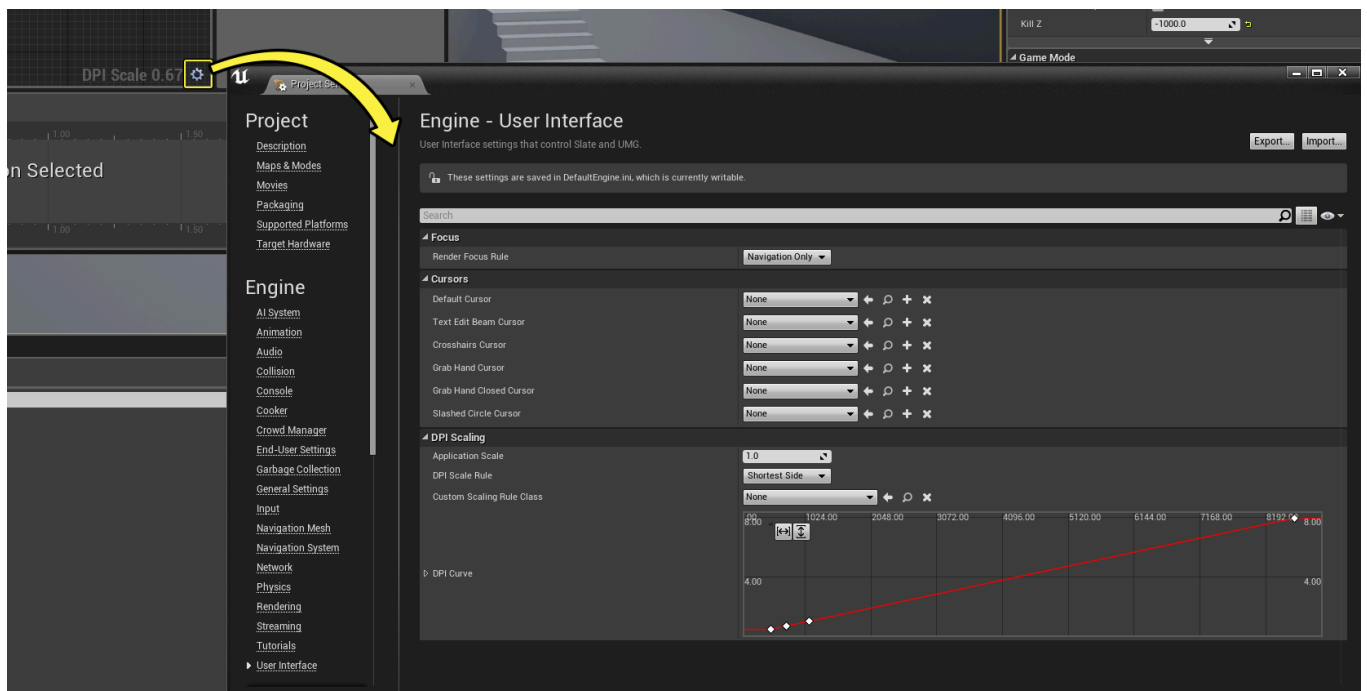
3. Take note of the Resolution aspect ratio in the lower left corner, and the **DPI Scale** in the lower-right corner.



Click for Full view.

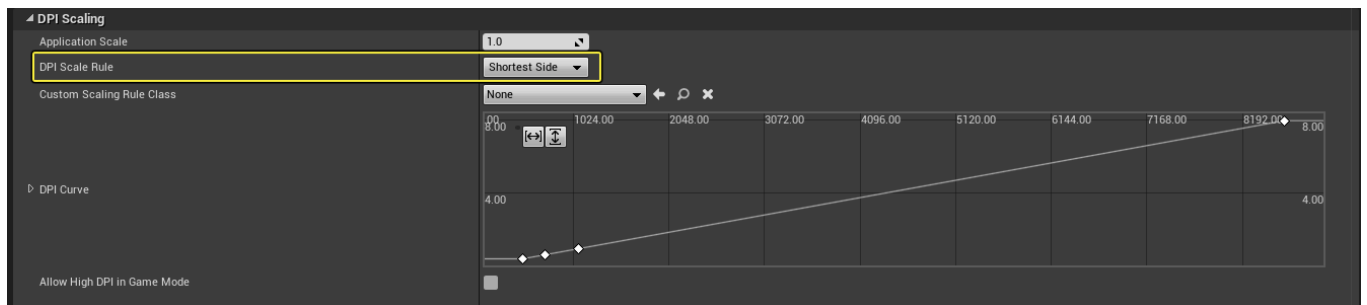
Typically, it is best to work on your UI elements in a scale of 1.0 and use the DPI Scaling Rule to scale up or down your UI elements.

4. Click the **Gear** icon in the lower right corner to open up the **User Interface Settings** window.

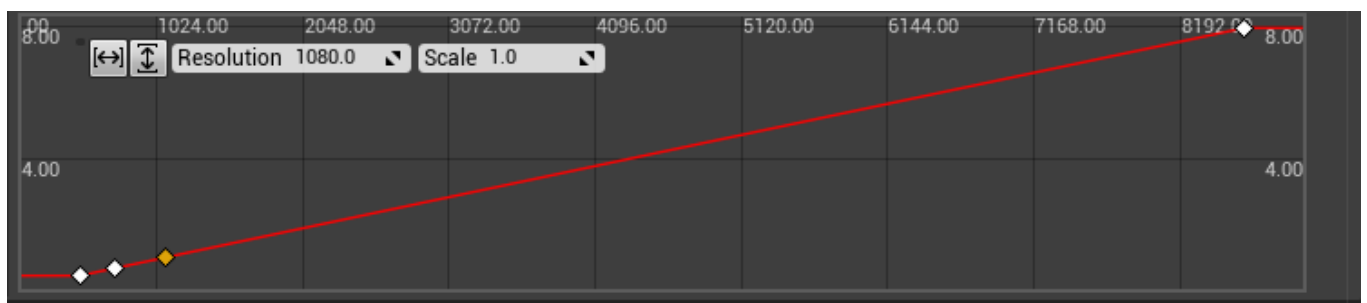


You can also access the User Interface Settings from the [Project Settings](#) of your project.

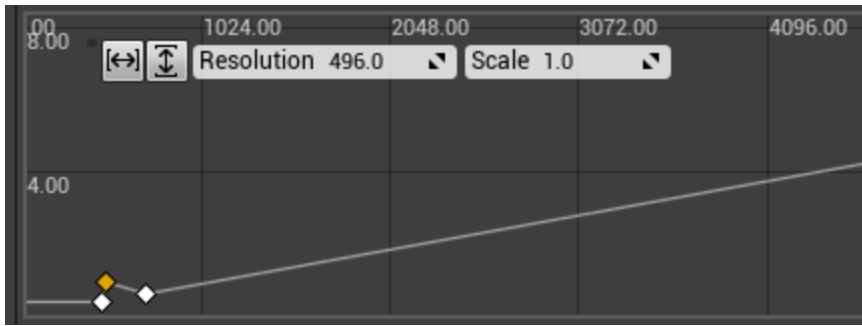
- Under **DPI Scaling**, choose the **DPI Scale Rule** you prefer to use. In this example, we are using the **Shortest Side** of the viewport.



- On the **DPI Curve**, find the key on the curve which reflects a **Scale** value of 1.0.

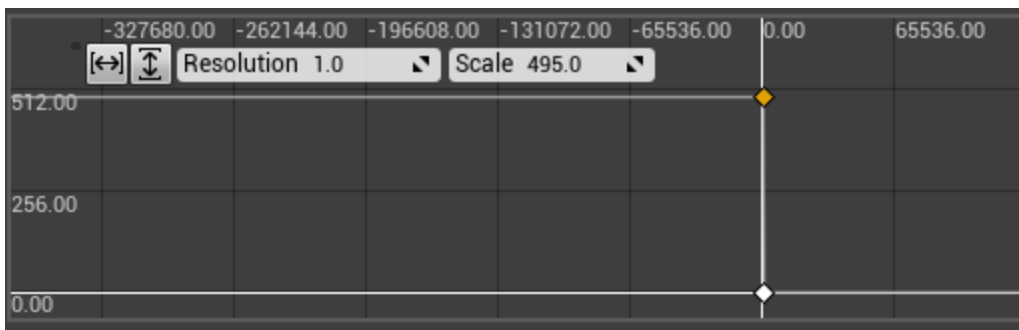


- Set the **Resolution** based on the Scale Rule you designated. This example uses the smallest side, so the resolution is reset from 1080 to 496.



This is the base key, from which all other keys change in relation to. If the resolution value is slightly off, such as 496.000061, this is because we have not set a resolution range yet.

8. Select another key on the graph and set the **Resolution** to 1, and the scale to **495**.



In order to make sure the UI elements are set correctly between different resolutions, we need this second key to set a designated resolution range for the UI to render correctly.



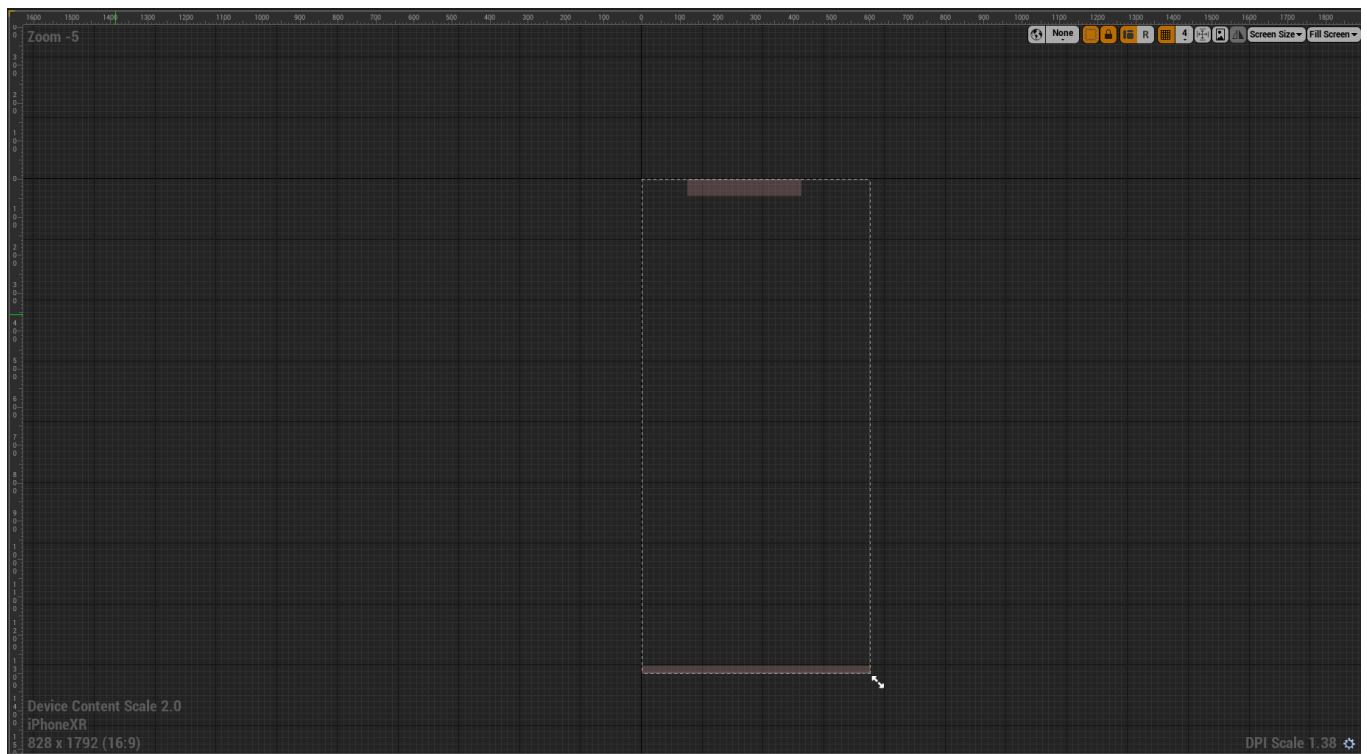
You can zoom in and out of the graph to get a detailed or high-level view of the range.

9. Return to your DPI_Widget Blueprint and, if it wasn't before, the DPI Scale for the device is now 1.0.

Adding New Device Ratios

With the initial aspect ratio set, you can now add more aspect ratios for different devices.

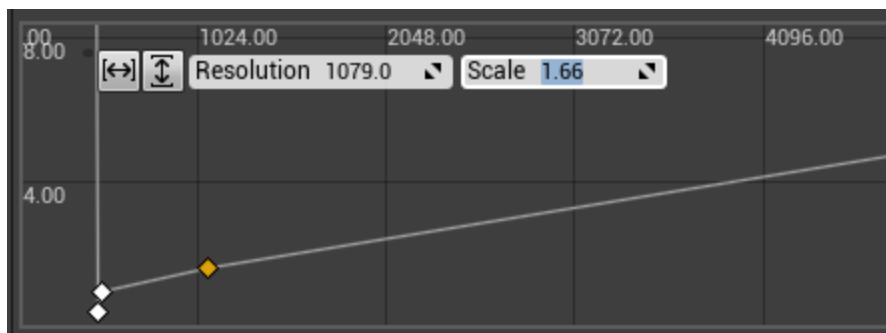
1. Select the **Screen Size** drop-down and select a different aspect ratio, such as a tablet or different phone brand.



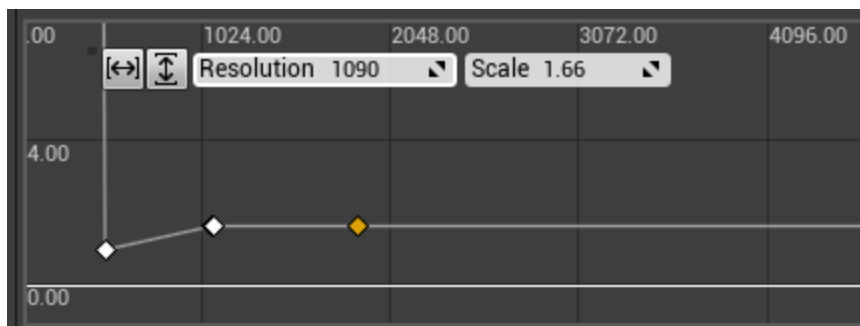
Click for Full view.

In this example, we will use an iPhone.

2. In the User Interface Settings for the **DPI Curve**, hold **Shift** and **Left-click** and create two new keys.
3. In the first key, set the Resolution to **1079** and the Scale to **1.66**.



4. In the second key, set the Resolution to **1090** and the Scale to **1.66**.



To determine the new scale for these keys, we find the difference between the DPI Scale Rule of the first and new devices. In this example, we divide the new device's smallest side by the first device's smallest side, so we divide $828/496$ for a new scale of 1.66.



When adding more devices, keep the same resolution for the new nodes of **1079** and **1090** as these are standard resolutions. The **Scale** is the variable that should change with each new device.

Your DPI Widget now includes an Android phone and an iPhone. Repeat this procedure to add new devices to your widget scaling.



If you are placing widgets and they are shifting off-screen, you may need to anchor your widgets to locations in the viewport. See [Anchors](#) for more information about anchoring widgets.