TARGET MAKER
CURSOR CONSTRUCTION PACK
v2.0.0

Created by [Ripcord Development](#)
[info@ripcorddev.com](#)

TargetMaker is a set of scripts, models and textures that when combined can quickly create unique assets for your project. Ideal for creating dynamic looking cursors and markers like those found in Real Time Strategy games.

TargetMaker 2.0 is a much more refined/stripped down version of what it used to be. It still has all the same great features from before, only now it's much easier to implement within your project.

PRESETS
To help you get started quickly, **TargetMaker** comes with a number of pre-built cursors ready to drop into your project. Feel free to pull these presets apart to get a sense of how things can be combined to make a unique look that's right for your project.

TargetMaker is composed of three different parts, Cursors, Markers, and Effects.

Cursors	The main visual object that follows the mouse cursor. These objects can contain many layers of animated objects.
Markers	An object that is generated after a click to indicate where the click took place. The marker stays in place for a limited amount of time before disappearing.
Effects	Extra animated objects that are generated to help guide the eye to where the click is taking place. These animations play once before disappearing.

MODELS
These are the basic building blocks in **TargetMaker**. All textures in this package can be applied to any of these meshes to create a unique cursor that's right for your project.

TEXTURES
These are sorted into categories based on what they are best suited for. Try experimenting by placing them on other elements. You'd be surprised by some of the unique effects you can create.

Rings	These make up the majority of the textures in the package as they make up the bulk of the cursor. Rings are circular textures applied to flat polygons. Combined with the included scripts these rings can spin and pulse to add extra movement.
Centers	Typically used at the center of a ring, a center is another texture applied to a plane. They are meant to indicate the center of the ring, but could be used for many other creative purposes as well.
Auras	Glowing objects around the ring. These are used to add extra visibility and add some 3D depth to the cursor.
Arrows	Extra effects that are generated to help guide the eye to where the click is taking place.

All textures are supplied as high-res 1024x1024 PNG's with an alpha channel. All textures are white so that they can be easily tinted to whatever colour you want.

To get a quick preview of each texture please refer to the Preview folder in the asset package for low-resolution previews.

MATERIALS / SHADERS

All materials in this package are set up to use the included **Unlit Alpha Tint** shader. This is an unlit shader that allows you to specify a tint colour as well as a transparency level. The **Unlit Alpha Tint** shader is in the Ripcord Common Assets folder.

Note* This shader is not necessary for **TargetMaker to function.*

SCRIPTS

TargetMaker comes with two sets of scripts. The scripts intended for use with the actual package, and scripts that are necessary for the demo to run. Any script with the "**DEMO_**" tag at the front is **not** needed for **TargetMaker** to function.

Note* **TargetMaker makes use of raycasting to detect and update the position of the target object. For raycasting to work, there must be a collider in your scene. The cursor object will only move along a collider as that's what's needed for raycasting to function. Please see the included targetMaker scene for an example.*

__TargetMaker__ is set to reference the MainCamera (Camera.main). This is a camera with the "MainCamera" tag. More than likely this setup will work for your project. If you do not wish to use this setup however, the script can be easily modified to reference any camera you wish.

For additional info on all the scripts listed below, please check the comments in the actual script files. The functions and variables are thoroughly commented. If you have further questions, please don't hesitate to ask.

- **CursorManager** - This script contains all the rules for the cursor. Audio and effects related to the cursor are generated from this script. This script should be attached to the top level of your cursor object. The Cursor Manager also creates an container object called EffectsContainer which is used to store any click effects.
    - Click Audio – The audio clip that will play when the click takes place.
    - Click Marker – The gameObject that will spawn at the location the click took place.
    - Click Effect – An effect object that will spawn at the location the click took place.
    - Marker Limit – The number of click markers that be on screen at any given time.
    - DestroyOldestMarker – If this is true, the oldest marker will be destroyed when the marker limit is reached, otherwise no new markers can be created until the oldest one has timed out.
    - HideMouseCursor – Toggle whether the system cursor is visible.
    - AxisPair – The pair of axes the cursor will move along.
        - **XY** - Vertical and horizontal movement such as a puzzle game or sidescroller.
        - **YZ** - Similar to XY, but sort of difficult to manipulate. You probably won't use this. :D
        - **XZ** - Horizontal movement such as a top down RTS game or anything moving along a ground plane.
- **MarkerBehaviour** – Controls all aspects of how the Marker objects appear on screen.
    - SpinRate – How fast the object spins. Positive values for clockwise rotation, negative values for counter-clockwise rotation.

- o Initial Scale – A value for the scale of the object when it first appears in the scene. This value will be applied to the X, Y, and Z scale values of the object.
- o ScalePulseAmount – How far the object will scale above and below the initial scale value.
- o ScalePulseSpeed – How fast the object will adjust its scale.
- o MarkerLifespan – How many seconds the marker will appear on screen before it begins shrinking down to nothing.
- o ShrinkRate – The rate at which the marker shrinks down to nothing.
- **DestroyEmpty** – This script is used to remove empty container objects from the scene. When the attached object no longer has any child objects, it is destroyed.
- **Spinner** - A simple script that rotates the object on its Y axis. Positive values rotate it clockwise. Negative values rotate it counterclockwise.
- **Arrow** – This script controls the speed and lifespan of the arrow effects

GETTING STARTED

Getting **TargetMaker** up and running is a simple process:
1. Create your target prefab either using one of the included presets or make your own from the pieces provided.
   - o Creating your own target from scratch:
     - It's recommended that you start with an empty game object and add all the pieces of your cursor as child objects.
     - To make sure your target accurately follows the cursor you should place all child objects at 0,0,0 so they are centered on the transform of the parent object.
     - Convert your parent and attached child objects into your new target prefab. This prefab can be named whatever you wish.
2. Add the **CursorManager** script to your new cursor prefab.
   - o Add any audio or click effects you want (these are not required).
   - o Make sure you set the AxisPair to the pair of axes appropriate for your scene.
   - o Make sure there's a collider in your scene for the CursorManager script to raycast against.
3. That's it! Happy clicking ☺

WRAP UP

TargetMaker provides the tools to create limitless dynamic cursors that will help bring life to your project. This package contains hundreds of options which create an almost infinite number of cursor designs. The only limit is your imagination.

The code has been heavily commented and modularized to make this package as easy to understand as possible. If you have any questions or comments, please don't hesitate to reach out.

If you find this package useful, please don't forget to leave positive feedback on the Unity Asset Store. **If you have any issues, please contact me with as much information about the issue as you can and I will get back to you as soon as possible.**

Thank you!

RIPCORD //
DEVELOPMENT //

www.ripcorddev.com