

# GazeCaster

The [GazeCaster](#) is an optional component that will allow users to tap on any Unity UI button or [GazeResponder](#).

*Please note that any Unity UI needs to have their Canvas's "[Graphic Raycaster](#)" component deleted.*

The [GazeCaster](#) has a few options:

```
LayerMask IMask;

delegate void GazeEvent();
event GazeEvent OnGaze_Start;
event GazeEvent OnGaze_End;
event GazeEvent OnGaze_InputDown;
event GazeEvent OnGaze_InputUp;

enum GazeModeEnum
{
    tapFirst,
    tapOnly,
    gazeFirst,
    gazeOnly,
    hoverOnly,
    off
}

GazeModeEnum currMode = GazeModeEnum.tapFirst;

string tapInputIdentifier = "TapInput";
```

## **Layer Mask - (IMask)**

The layer mask is used to determine which layers should be valid for the [GazeCaster](#)'s raycast. Any object on the specified layers will check for a [GazeResponder](#) type object on raycast hit. Any valid objects will then receive a [OnGaze\\_Start GazeEvent](#) when it first receives the hit, a [OnGaze\\_End GazeEvent](#) when the raycast no longer is on the same object, a [OnGaze\\_InputDown GazeEvent](#) when input button down is received while gazing at an object, and a [OnGaze\\_InputUp GazeEvent](#) when the input button up is received.

## **Current Mode - (currMode)**

The [CurrentMode](#) specifies which behaviour should take priority when the user taps or Gaze-Clicks a button.

“[tapFirst](#)” will prioritize device finger taps over gaze cast calls. It will first see if the finger tap position hit a button, and then if it didn’t, it will cast from the [GazeCaster](#) and check for a button there.

“[tapOnly](#)” will only listen to device finger taps. It will not cast a ray from the [GazeCaster](#) to check for buttons.

“[gazeFirst](#)” will prioritize the [GazeCaster](#)’s raycast calls over the device finger taps. It will first see if the [GazeCaster](#) hit a button, and then if it didn’t, it will check from the finger tap position and check for a button there.

“[gazeOnly](#)” will only listen to [GazeCaster](#)’s raycast calls. It will not the finger tap position.

“[hoverOnly](#)” will only listen to [GazeCaster](#)’s [OnGaze\\_Start](#) and [OnGaze\\_End](#) events. It will not the finger tap position or any [OnGaze\\_InputEvent](#).

“[off](#)” will disable all input. This setting may be useful if the app needs to pause for any reason.

#### **Tap Input Identifier - (*tapInputIdentifier*)**

The [Tap Input Identifier](#) is the tag that the [GazeCaster](#) will look for when checking for valid finger tap position buttons. If the tag is left null, all objects will be considered valid recipients of a finger tap.

## **GazeResponder**

The [GazeResponder](#) is an optional base class component that provides the main functionality of the Gaze Input System. Any objects that have a [GazeResponder](#) will be valid recipients of calls from the [GazeCaster](#). Any class that you wish to receive Gaze Events from should derive from [GazeResponder](#).

```
public interface GazeResponder
{
    void OnGazeEnter();
    void OnGazeExit();
    void OnGazeTrigger();
    void OnGazeTriggerEnd();
}
```

For a basic example of this system in place, see the [BasicGazeButton](#) in the [ [Tools](#) / [Input](#) / [GazeInput](#) / [Examples](#) ] folder of the MERGE Cube SDK.