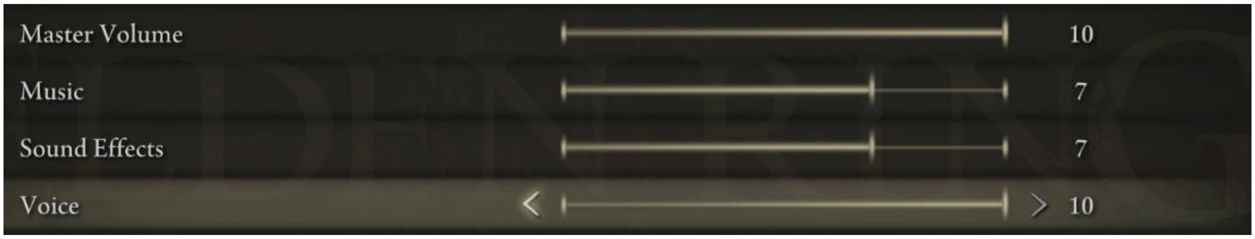
Slider

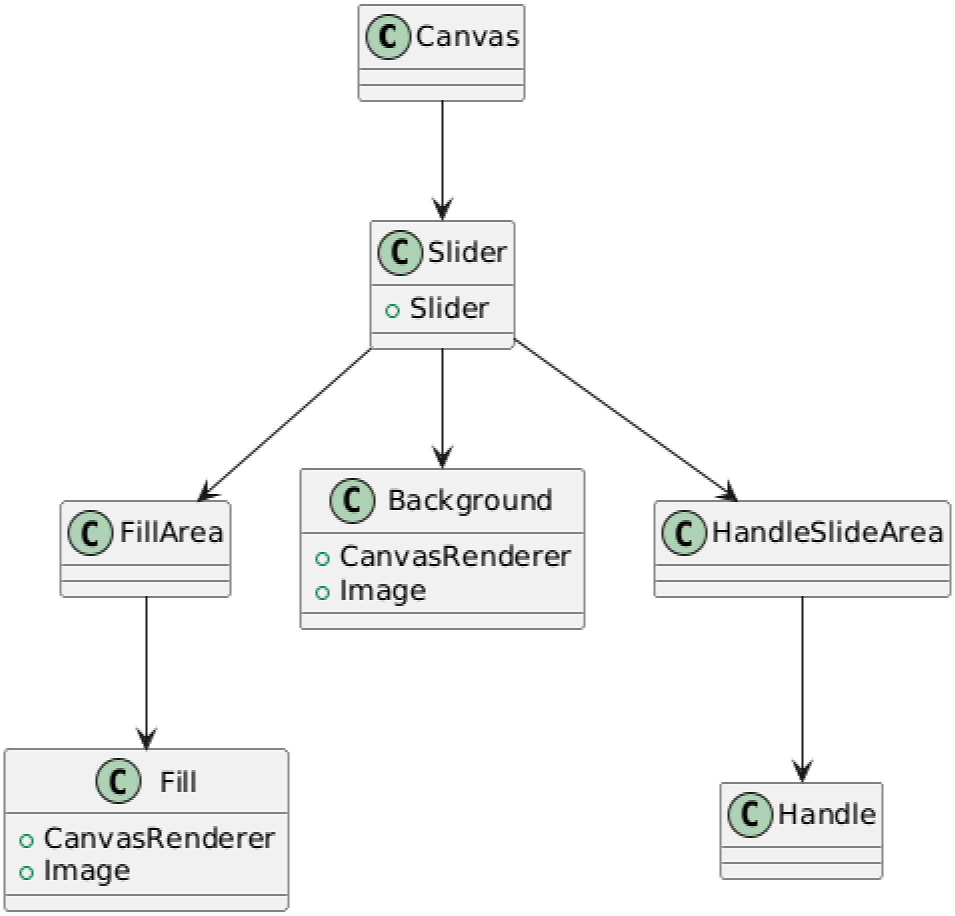
Overview of Element

The purpose of the slider is to allow a user to adjust some value. In most games, we see this primarily in the settings menu. Volume sliders, FOV sliders, there are many of these customization options that lend well to a slider. Elden Ring uses sliders to control the various audios found within the game.



Master Volume	10
Music	7
Sound Effects	7
Voice	10

Hierarchy Diagram



Hierarchy Diagram of the different GameObjects involved in a Slider, and their Components

Settings

Interactable: Allows/Disallows the user from interacting with the component

Transition: Controls how the slider reacts visually when interacted with.

Navigation: Determines how users can navigate to and from this UI element in a controller setting.

Fill Rect: Reference to the RectTransform of the Fill object. Controls the area that fills as the slider value increases.

Handle Rect: Reference to the RectTransform of the Handle object. The handle is what the user drags.

Direction: Sets which way the slider fills.

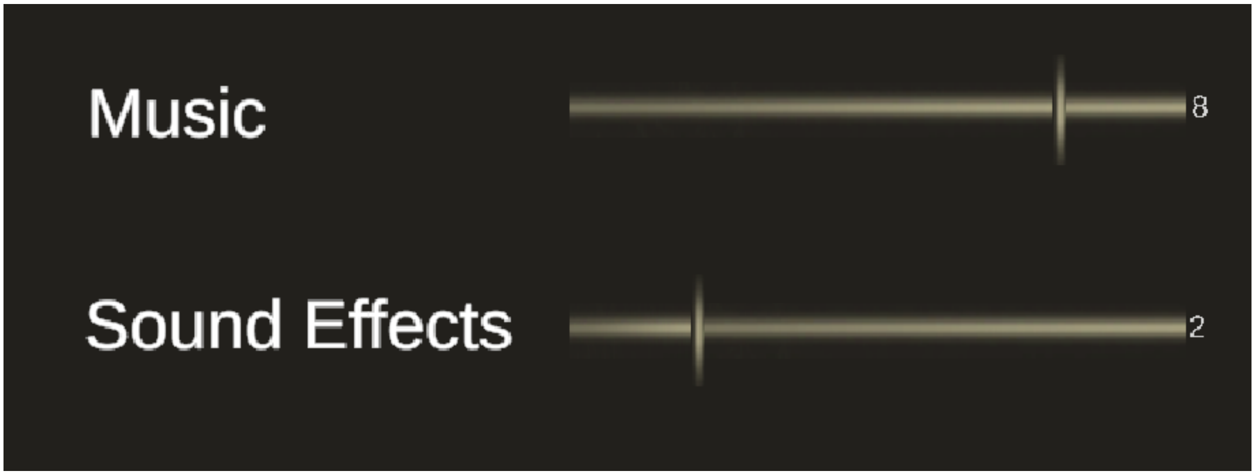
Min Value/Max Value: The minimum/maximum value the slider can represent (default: 1).

Whole Numbers: When checked, the slider snaps to whole number values.

Value: The current value of the slider.

On Value Changed: An event triggered when the slider value changes. Can be connected to functions in scripts similar to buttons.

Demo



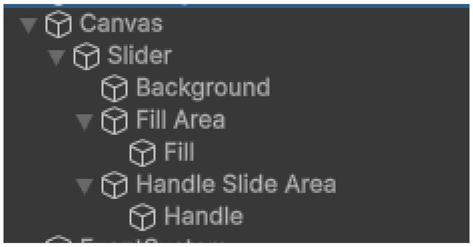
This demo implements two sliders, both sliders have the whole number setting enabled. There is a listener attached to both sliders that displays the value of the slider and is triggered whenever the value of the slider is changed.

Reference Sheet

Overview:

The Slider UI element allows users to select a numeric value from a defined range by dragging a handle along a track. In games, sliders are commonly found in the settings menu for setting things like volume control, field of view adjustment, graphics quality, and other numerical parameters that have minimum and maximum bounds.

Hierarchy:

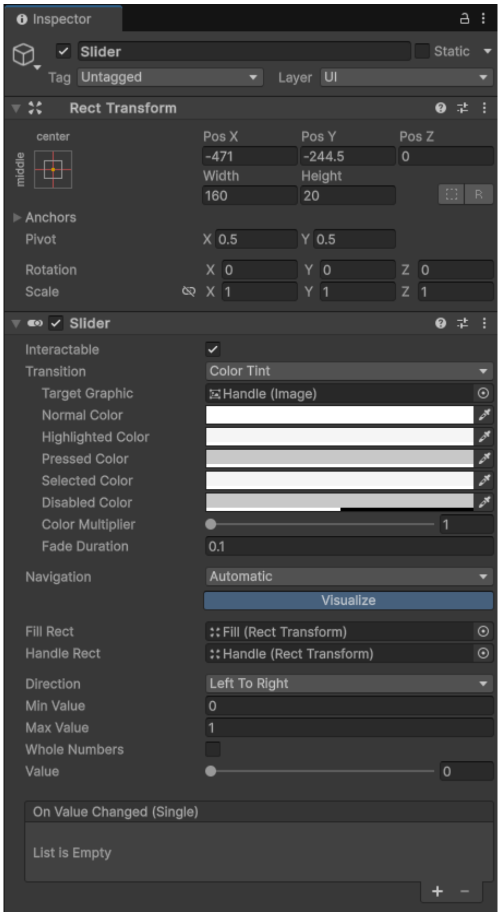


- Canvas
  - Slider
    - Background
    - Fill Area
      - Fill
    - Handle Slide Area
      - Handle

Since the slider is a UI component, it is a child of the canvas. The slider itself contains three children of its own: Background, Fill Area, and Handle Slide Area. Those control the size/color of the slider background, handle, and "fill" (when sliding to the right, everything to the left will have the properties defined in fill, and when sliding left, everything to the right will have the properties of background)

Slider Settings:

- Interactable: Allows/Disallows the user from interacting with the component
- Transition: Controls how the slider reacts visually when interacted with.
- Navigation: Determines how users can navigate to and from this UI element in a controller setting.
- Fill Rect: Reference to the RectTransform of the Fill object. Controls the area that fills as the slider value increases.
- Handle Rect: Reference to the RectTransform of the Handle object. The handle is what the user drags.
- Direction: Sets which way the slider fills.
- Min Value/Max Value: The minimum/maximum value the slider can represent (default: 1).
- Whole Numbers: When checked, the slider snaps to whole number values.
- Value: The current value of the slider.
- On Value Changed: An event triggered when the slider value changes. Can be connected to functions in scripts similar to buttons.



☑ Slider □ Static
Tag Untagged Layer UI

Rect Transform
center middle
Pos X -471 Pos Y -244.5 Pos Z 0
Width 160 Height 20

Anchors
Pivot X 0.5 Y 0.5
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1

☑ Slider
Interactable ☑
Transition Color Tint
Target Graphic Handle (Image)
Normal Color
Highlighted Color
Pressed Color
Selected Color
Disabled Color
Color Multiplier 1
Fade Duration 0.1
Navigation Automatic

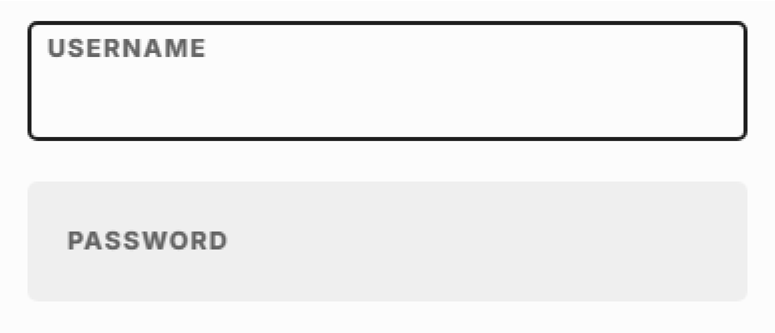
Fill Rect Fill (Rect Transform)
Handle Rect Handle (Rect Transform)
Direction Left To Right
Min Value 0
Max Value 1
Whole Numbers □
Value 0

On Value Changed (Single)
List is Empty

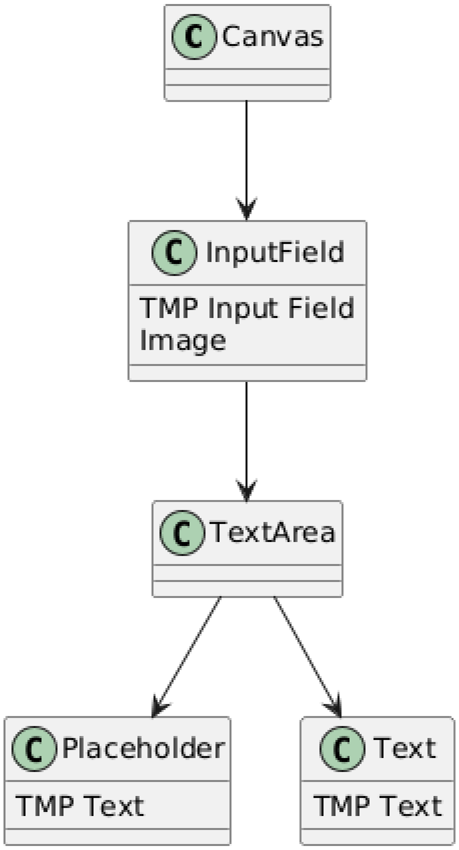
Input Field

Overview of Element

The Input Field is a way for the user to input text into the game. Typically, you would see these in login screens, but you would also see them in games when searching for an item or giving something a name. The example below is the login screen for the Riot Client.



Hierarchy Diagram



Hierarchy Diagram of InputField

Settings

InputField->Image controls how the input field is rendered.

- Source Image: Defines the sprite used for the InputField's background.
- Color: Sets the tint color of the Image.
- Raycast Target: Determines if the Image component blocks raycasts.
- Type: Controls how the sprite is rendered (Simple, Sliced, Tiled, or Filled).

InputField->TMP Input Field controls the text properties within the input field.

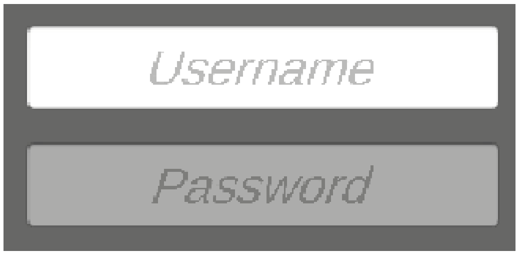
- Text: The current text content inside the input field.
- Font Asset: The font of the text.
- Character Limit: Maximum number of characters allowed.
- Line Type: Specifies how text is displayed:
    - Single Line: Text remains on a single line.
    - Multi-Line Submit: Allows multiple lines but submits on "Enter."
    - Multi-Line Newline: Allows multiple lines with Enter creating a new line.
- Content Type: Selects what is expected as an input (Integer, Decimal, Password, Name, Email, Standard, and more).
- OnFocus - Select All: Toggles if all text is selected when the input field is selected
- ReadOnly: Toggles if the input field is read only.

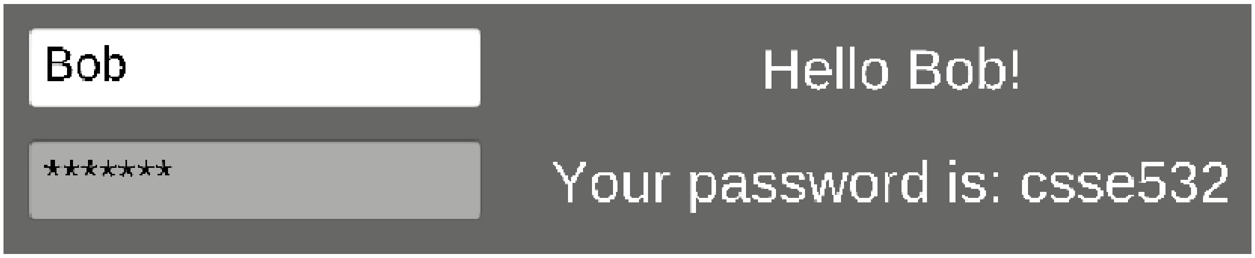
Text Area determines where the actual text box is located with the transform component.

Placeholder is the text displayed when the input field is not selected. Text is the user input that will be displayed. Both have the TMP Text component with the following settings.

- Text: The actual string displayed in the UI.
- Auto Size: Dynamically adjusts font size within a specified range to fit content.
- Font Asset: The font of the text.
- Overflow Mode: Determines how text behaves when it exceeds the available space.
- Word Wrapping: Enables text to break at spaces to fit within the container.

Demo




Bob

****

Hello Bob!

Your password is: csse532

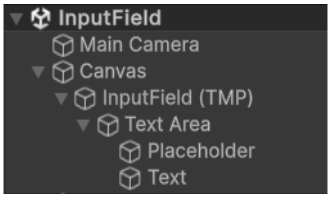
The demo is comprised of two input field elements. The top picture shows the placeholder text for both input fields. The bottom picture shows the user input. The Password input field has the Content Type set to password, which is why the input is asterisks. Additionally, the user input is displayed next to the input fields to show that the user input can be easily obtained and used through listeners on the input field.

Reference Sheet

Overview:

The Input Field is a way for the user to input text into the game. Typically, you would see these in login screens, but you would also see them in games when searching for an item or giving something a name.

Hierarchy:



```
▼ InputField
    Main Camera
  ▼ Canvas
    ▼ InputField (TMP)
      ▼ Text Area
          Placeholder
          Text
```

The InputField is a UI element, so it is a child of Canvas. InputField controls how the element is rendered in the scene and also more general properties of how the Input Field acts, such as character limit or font. InputField has a child Text Area which is the location of the actual text box. Text Area has two children: Placeholder and Text. Placeholder is the text that is displayed before the user has input anything. Text is the user input. Both Placeholder and Text contain the properties of how the text is rendered.

InputField Settings:

Image controls how the input field is rendered.

- Source Image: Defines the sprite used for the InputField's background.
- Color: Sets the tint color of the Image.
- Raycast Target: Determines if the Image component blocks raycasts.
- Type: Controls how the sprite is rendered (Simple, Sliced, Tiled, or Filled).

TMP Input Field controls the text properties within the input field.

- Text: The current text content inside the input field.
- Font Asset: The font of the text.
- Character Limit: Maximum number of characters allowed.
- Line Type: Specifies how text is displayed:
    - Single Line: Text remains on a single line.
    - Multi-Line Submit: Allows multiple lines but submits on "Enter."
    - Multi-Line Newline: Allows multiple lines with Enter creating a new line.
- Content Type: Selects what is expected as an input (Integer, Decimal, Password, Name, Email, Standard, and more).
- OnFocus - Select All: Toggles if all text is selected when the input field is selected
- ReadOnly: Toggles if the input field is read only.