

# WebView

This extension allows **Android** and **iOS** developers to make use of the WebView to display site URLs directly from inside their game.

## Functions

---

The following functions are provided to manipulate the WebView:

- [WebView\\_Create](#)
- [WebView\\_Destroy](#)
- [WebView\\_Button\\_Add](#)
- [WebView\\_Button\\_Destroy](#)
- [WebView\\_Button\\_SetAlpha](#)

# WebView\_Create

Creates a WebView with assigned url.

## Syntax:

```
WebView_Create(url);
```

Argument	Type	Description
url	string	The URL that the web view will open

## Returns:

N/A

## Triggers:

Asynchronous Social Event

async_load Contents		
Key	Type	Description
type	string	The string "onCloseWindow"

## Example:

```
WebView_Create("https://www.yoyogames.com/")
```

The code sample above will create a new webview (with the specified url). This view can be destroyed at any time using the function **WebView\_Destroy** triggering a callback that can be called inside an **Async Social** event.

```
if(async_load["type"] == "WebView")
switch(async_load["event"])
{
    case "onCloseWindow":
        show_debug_message("[WebView] This webview is was closed!")
}
```

```
}  
    break
```

The code above matches the response against the correct event **type** and logs it to the console.

# WebView\_Destroy

Destroys a previously created WebView.

**NOTE** If there is no created WebView nothing will happen.

## Syntax:

```
WebVi ew_Destroy();
```

## Returns:

N/A

## Example:

```
WebVi ew_Create();  
  
// Later in the code  
  
WebVi ew_Button_Destroy();  
WebVi ew_Destroy();
```

The code above provides a sample of a WebView being created (using [WebView\\_Create](#), there's a full sample on that function's reference) and later in the code being destroyed.

# WebView\_Button\_Add

Attaches a button to the WebView. When the button is pressed an **Async Social** event is triggered.

**NOTE** If this function is called before the WebView is created, the called is ignored.

## Syntax:

```
WebVi ew_Button_Add(path);
```

Argument	Type	Description
path	string	The path to the image to be used as button.

## Returns:

N/A

## Triggers:

Asynchronous Social Event

async_load Contents		
Key	Type	Description
type	string	The string <code>"onButtonPressed"</code>

## Example:

```
WebVi ew_Create("https: //www. yoyogames. com/")  
WebVi ew_Button_Add("WebVi ew/i mg_cl ose. png")
```

The code sample above creates a WebView and attaches a button to it, after this we can catch the click of the button inside an **Async Social** event.

```
if(async_load["type"] == "WebView")
switch(async_load["event"])
{
    case "onButtonPressed":
        WebView_Button_Destroy()
        WebView_Destroy()
        break
}
```

The code above matches the response against the correct event **type** and logs the success of the task.

# WebView\_Button\_Destroy

Destroys the button from the WebView.

**WARNING** Always make sure to destroy the button before destroying the WebView.

## Syntax:

```
WebVi ew_Button_Destroy();
```

## Returns:

N/A

## Example:

```
WebVi ew_Create();  
  
// Later in the code  
  
WebVi ew_Button_Destroy();  
WebVi ew_Destroy();
```

The code above provides a sample of a WebView being created (using [WebView\\_Create](#), there's a full sample on that function's reference) and later in the code being destroyed (using the function [WebView\\_Destroy](#)). Note that the button should always be destroyed before the WebView does.

# WebView\_Button\_SetAlpha

Sets alpha value of the button attached to the WebView. This should be a value ranging from 0 to 1, where 0 means hidden and 1 means fully visible.

## Syntax:

```
WebView_Button_SetAlpha(alpha);
```

Argument	Type	Description
alpha	real	Value from 0-1

## Returns:

N/A

## Example:

```
WebView_Create("https://www.yoyogames.com/");  
WebView_Button_Add("WebView/img_close.png");  
WebView_Button_SetAlpha(0.5);
```

The code above will create a WebView attach it a button and make sure that the button is semi-transparent.