

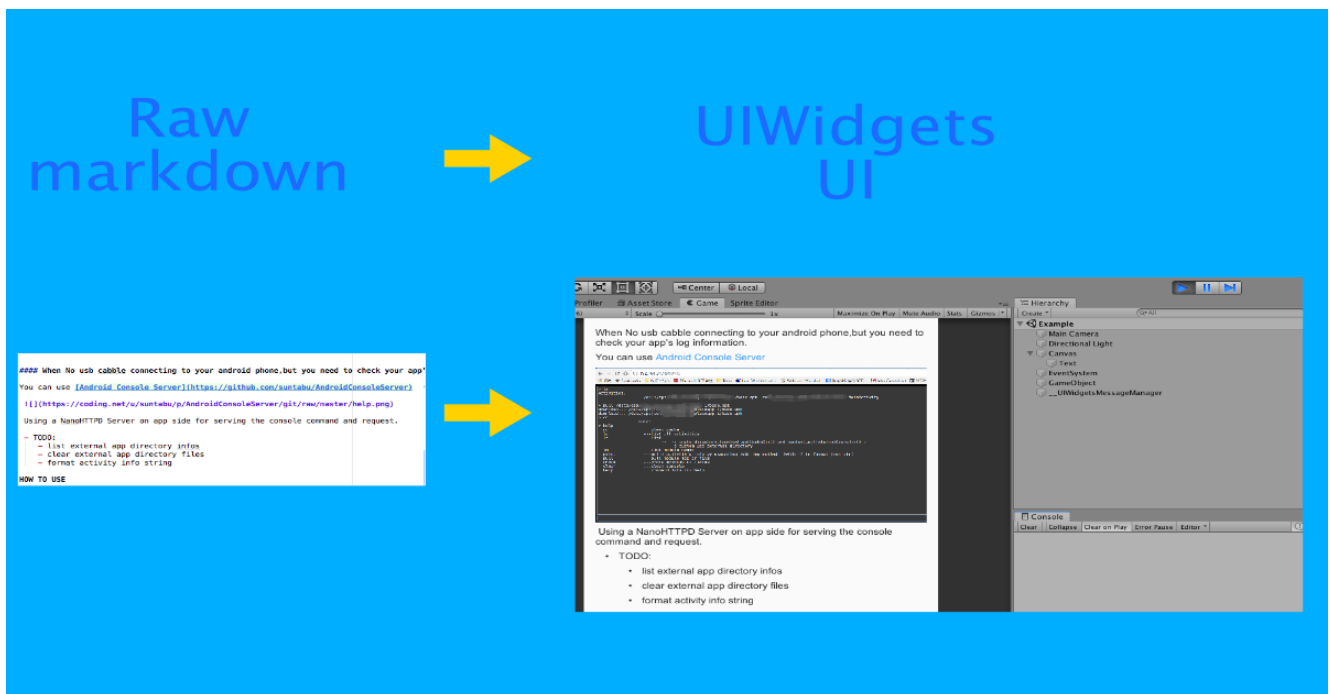
New unity plugin has been released: <https://youtu.be/5Q6qpgveYgA>

What is UIWidgets?

UIWidget is a Unity Package which helps developers to create, debug and deploy efficient, cross-platform Apps. Details could be found at their [github page](#)

What does this plugin do?

Just parse and render raw markdown strings into UIWidgets elements.



How to use

- 1. You must learn how to use UIWidgets first
- 2. Import UIWidgets into your Unity project according to [Requirements](#)
- 3. Then import this plugin into your Unity project
- 4. Embed your markdown element into UIWidgets UI codes just like below:

```
protected override Widget createWidget()
{
    return new MaterialApp(
        title: "Markdown Demo",
        showPerformanceOverlay: false,
        home: new Scaffold(
            body: new Markdown(null, markdownData1,
                syntaxHighlighter:
```

```

        new DartSyntaxHighlighter(SyntaxHighlighterStyle.lightTheme:
        onTapLink:
        url => { Application.OpenURL(url); })
    )
);
}

```

- 6. Adjust your code for better performance.
- 7. Enjoy!

Tips

- To use UIWidgets, Unity version must be larger than 2018.3
- If the markdown string is too long, it would take much more time to parse and render, so `async` operation is needed and a [Dispatcher](#) script is provided to handle this situation

```

buildThread = new Thread(() =>
{
    try
    {
        Stopwatch sw = Stopwatch.StartNew();
        if (nodes == null)
        {
            nodes = document.parseLines(lines.ToList());
        }

        if (widget.onParsed != null)
        {
            widget.onParsed(widget.id, nodes);
        }

        var elements = builder.build(nodes);

        // make the callback running on the unity main thread
        Dispatcher.Invoke(() =>
        {
            updateState(elements);
        });

        sw.Stop();
    }
}

```

```
        Debug.Log(sw.ElapsedMilliseconds / 1000f);  
    }  
    catch (ThreadAbortException e)  
    {  
        Debug.Log(e.Message);  
    }  
});
```

- Also could use [in-app webview](#) to handle URL click event to replace `Application.OpenURL` for viewing web page in your unity application.