

```
1 import 'dart:async';
2 import 'dart:convert';
3
4 import 'package:flutter/material.dart';
5 import 'package:http/http.dart' as http;
6 import 'package:transparent_image/transparent_image.dart';
7
8 import 'page.dart';
9 import 'postdetail.dart';
10
11 void main() => runApp(MyApp());
12
13 class MyApp extends StatelessWidget {
14   // This widget is the root of your application.
15   @override
16   Widget build(BuildContext context) {
17     return MaterialApp(
18       home: MyHomePage(),
19     );
20   }
21 }
22
23 class MyHomePage extends StatefulWidget {
24   @override
25   _MyHomePageState createState() => _MyHomePageState();
26 }
27
28 class DrawerItem {
29   String title;
30   IconData icon;
31
32   DrawerItem(this.title, this.icon);
33 }
34
35 class _MyHomePageState extends State<MyHomePage> {
36   // Base URL for our wordpress API
37   final String apiUrl = "https://robertrobinson.in/wp-json/wp/v2/";
38
39   // Empty list for our posts
40   List posts;
41   List pages;
42
43   // Function to fetch list of posts
44   Future<String> getPosts() async {
```

```

45     var res = await http.get(Uri.encodeFull(apiUrl + "
posts?_embed"),
46         headers: {"Accept": "application/json"});
47
48     // fill our posts list with results and update state
49     setState(() {
50         var resBody = json.decode(res.body);
51         posts = resBody;
52     });
53
54     return "Success!";
55 }
56
57 Future<String> getPages() async {
58     var pageres = await http.get(Uri.encodeFull(apiUrl + "
pages"),
59         headers: {"Accept": "application/json"});
60
61     setState(() {
62         var resPageBody = json.decode(pageres.body);
63         pages = resPageBody;
64     });
65 }
66
67 @override
68 void initState() {
69     super.initState();
70     this.getPosts();
71     this.getPages();
72 }
73
74 @override
75 Widget build(BuildContext context) {
76     if (pages == null || posts == null) {
77         return new Scaffold(
78             appBar: new AppBar(
79                 title: Text("Repute"), backgroundColor: Colors
.redAccent),
80             body: CircularProgressIndicator(),
81         );
82     } else {
83         List<Widget> drawerOptions = [];
84         for (var i = 0; i < pages.length; i++) {
85 //             var d = drawerItems[i];
86             drawerOptions.add(new ListTile(

```

```

87 //          leading: new Icon(d.icon),
88          title: new Text(
89              //d.title,
90              pages[i]["title"]["rendered"],
91              style: new TextStyle(fontSize: 18.0,
fontWeight: FontWeight.w400),
92          ),
93 //          selected: i == _selectedIndex,
94          onTap: () {
95              Navigator.push(
96                  context,
97                  new MaterialPageRoute(
98                      builder: (context) => new WebViewExample(
url: pages[i]),
99                  ),
100              );
101          },
102      ));
103  }
104
105  return Scaffold(
106      appBar:
107          AppBar(title: Text("Repute"), backgroundColor
: Colors.redAccent),
108      drawer: new Drawer(
109          child: new Column(
110              children: <Widget>[
111                  new UserAccountsDrawerHeader(
112                      decoration: BoxDecoration(color: Colors.
redAccent),
113                      accountName: new Text('Repute'),
114                      //          accountEmail: new Text('testemail@test.
com'),
115                      currentAccountPicture: new CircleAvatar(
116                          backgroundColor: Colors.white,
117                          backgroundImage: new NetworkImage(
118                              'http://reputetech.com/demo/wp-
content/uploads/2019/01/icon.png'),
119                          ),
120                      ),
121                  new Column(children: drawerOptions),
122                      //          new Column( children: <Widget>[
123                      //          ListView.builder(
124                      //              itemCount: posts == null ? 0 : posts.
length,

```

```

125 //      itemBuilder: (context, index) {
126 //
127 //          new ListTile(
128 //              title: new Text('Second Menu Item
129 //              '),
130 //              onTap: () {
131 //                  Navigator.push(context,
132 //                      new MaterialPageRoute(builder: (context) => new
133 //                          WebViewExample()),);
134 //              },
135 //          );
136 //      },
137 //      ], ),
138 //      ],
139 //  )),
140 //  body: ListView.builder(
141 //      itemCount: posts == null ? 0 : posts.length,
142 //      itemBuilder: (BuildContext context, int index)
143 //      {
144 //          return Column(
145 //              children: <Widget>[
146 //                  Card(
147 //                      child: Column(
148 //                          children: <Widget>[
149 //                              new FadeInImage.memoryNetwork(
150 //                                  placeholder: kTransparentImage,
151 //                                  image: posts[index]["
152 //                                  featured_media"] == 0
153 //                                  ? 'images/placeholder.png'
154 //                                  : posts[index]["_embedded"]["
155 //                                  wp:featuredmedia"][0]
156 //                                  ["source_url"],
157 //                              ),
158 //                              new Padding(
159 //                                  padding: EdgeInsets.all(10.0),
160 //                                  child: new ListTile(
161 //                                      title: new Padding(
162 //                                          padding: EdgeInsets.
163 //                                          symmetric(vertical: 10.0),
164 //                                          child: new Text(
165 //                                              posts[index]["title"]
166 //                                              ["rendered"])),
167 //                                      subtitle: new Text(posts[

```

```

162 index] ["excerpt"]
163                                     ["rendered"]
164                                     .replaceAll(new RegExp(r
    '<[^>]*>'), ' '),
165                                     ),
166     new ButtonTheme.bar(
167       child: new ButtonBar(
168         children: <Widget>[
169           new FlatButton(
170             child: const Text('READ
    MORE'),
171             onPressed: () {
172               Navigator.push(
173                 context,
174                 new MaterialPageRoute(
175                   builder: (context) =>
176                     new ReputePost(
    post: posts[index]),
177                 ),
178               );
179             },
180           ),
181         ],
182       ),
183     ),
184   ],
185 ),
186 ),
187 ],
188 );
189 },
190 ),
191 );
192 }
193 }
194 }
195

```

```

1 import 'dart:async';
2 import 'package:flutter/material.dart';
3 import 'package:webview_flutter/webview_flutter.dart';
4
5 void main() => runApp(MaterialApp(home: WebViewExample()))
6 ;
7 class WebViewExample extends StatelessWidget {
8   final Completer<WebViewController> _controller =
9     Completer<WebViewController>();
10  var url;
11  WebViewExample({Key key, @required var this.url}) :
12    super(key: key);
13  @override
14  Widget build(BuildContext context) {
15    return Scaffold(
16      appBar: AppBar(
17        title: new Text(url['title']['rendered']),
18        backgroundColor: Colors.redAccent,
19        // This drop down menu demonstrates that Flutter
20        widgets can be shown over the web view.
21        actions: <Widget>[
22          NavigationControls(_controller.future),
23          SampleMenu(_controller.future),
24        ],
25        body: WebView(
26          initialUrl: url['link'],
27          javascriptMode: JavascriptMode.unrestricted,
28          onWebViewCreated: (WebViewController
29            webViewController) {
30              _controller.complete(webViewController);
31            },
32          // floatingActionButton: favoriteButton(),
33        );
34    }
35
36  Widget favoriteButton() {
37    return FutureBuilder<WebViewController>(
38      future: _controller.future,
39      builder: (BuildContext context,
40        AsyncSnapshot<WebViewController> controller) {
41        if (controller.hasData) {

```

```

42         return FloatingActionButton(
43             onPressed: () async {
44                 final String url = await controller.data.
currentUrl();
45                 Scaffold.of(context).showSnackBar(
46                     SnackBar(content: Text("Favorited $url")
47             ),
48             );
49             child: const Icon(Icons.favorite),
50         );
51     }
52     return Container();
53 });
54 }
55 }
56
57 enum MenuOptions {
58     evaluateJavascript,
59     toast,
60 }
61
62 class SampleMenu extends StatelessWidget {
63     SampleMenu(this.controller);
64     final Future<WebViewController> controller;
65
66     @override
67     Widget build(BuildContext context) {
68         return FutureBuilder<WebViewController>(
69             future: controller,
70             builder:
71                 (BuildContext context, AsyncSnapshot<
WebViewController> controller) {
72                 return PopupMenuButton<MenuOptions>(
73                     onSelect: (MenuOptions value) {
74                         switch (value) {
75                             case MenuOptions.evaluateJavascript:
76                                 _onEvaluateJavascript(controller.data,
context);
77                             break;
78                             case MenuOptions.toast:
79                                 Scaffold.of(context).showSnackBar(
80                                     SnackBar(
81                                         content: Text('You selected: $value'),
82                                     ),

```

```

83         );
84         break;
85     }
86 },
87     itemBuilder: (BuildContext context) => <
PopupMenuItem<MenuOptions>>[
88         PopupMenuItem<MenuOptions>(
89             value: MenuOptions.evaluateJavascript,
90             child: const Text('Evaluate JavaScript'),
91             enabled: controller.hasData,
92         ),
93         const PopupMenuItem<MenuOptions>(
94             value: MenuOptions.toast,
95             child: Text('Make a toast'),
96         ),
97     ],
98 );
99 },
100 );
101 }
102
103 void _onEvaluateJavascript(
104     WebViewController controller, BuildContext context)
105     async {
106     final String result = await controller
107         .evaluateJavascript("document.body.style.
108         backgroundColor = 'red'");
109     Scaffold.of(context).showSnackBar(
110         SnackBar(
111             content: Text('JavaScript evaluated, the result
112             is: $result'),
113         ),
114     );
115 }
116
117 class NavigationControls extends StatelessWidget {
118     const NavigationControls(this._webViewControllerFuture)
119     : assert(_webViewControllerFuture != null);
120
121     final Future<WebViewController>
122     _webViewControllerFuture;
123
124     @override
125     Widget build(BuildContext context) {

```



```

123     return FutureBuilder<WebViewController>(
124         future: _webViewControllerFuture,
125         builder:
126             (BuildContext context, AsyncSnapshot<
127                 WebViewController> snapshot) {
128                 final bool webViewReady =
129                     snapshot.connectionState == ConnectionState.
130                     done;
131                 final WebViewController controller = snapshot.
132                     data;
133                 return Row(
134                     children: <Widget>[
135                         //
136                         IconButton(
137                             icon: const Icon(Icons.arrow_back_ios),
138                             onPressed: !webViewReady
139                                 ? null
140                                 : () async {
141                                     if (await controller.canGoBack()) {
142                                         controller.goBack();
143                                     } else {
144                                         Scaffold.of(context).showSnackBar(
145                                             const SnackBar(content: Text("No
146                                             back history item")),
147                                         );
148                                     }
149                                     return;
150                                 },
151                         //
152                         IconButton(
153                             icon: const Icon(Icons.arrow_forward_ios
154                                 ),
155                             onPressed: !webViewReady
156                                 ? null
157                                 : () async {
158                                     if (await controller.canGoForward()) {
159                                         controller.goForward();
160                                     } else {
161                                         Scaffold.of(context).showSnackBar(
162                                             const SnackBar(
163                                                 content: Text("No forward
164                                             history item")),
165                                         );
166                                     }
167                                     return;
168                                 },
169                     ],
170                 ),
171             ),
172         ),
173     ),
174     ),
175     ),
176     ),
177     ),
178     ),
179     ),
180     ),
181     ),
182     ),
183     ),
184     ),
185     ),
186     ),
187     ),
188     ),
189     ),
190     ),
191     ),
192     ),
193     ),
194     ),
195     ),
196     ),
197     ),
198     ),
199     ),
200     ),
201     ),
202     ),
203     ),
204     ),
205     ),
206     ),
207     ),
208     ),
209     ),
210     ),
211     ),
212     ),
213     ),
214     ),
215     ),
216     ),
217     ),
218     ),
219     ),
220     ),
221     ),
222     ),
223     ),
224     ),
225     ),
226     ),
227     ),
228     ),
229     ),
230     ),
231     ),
232     ),
233     ),
234     ),
235     ),
236     ),
237     ),
238     ),
239     ),
240     ),
241     ),
242     ),
243     ),
244     ),
245     ),
246     ),
247     ),
248     ),
249     ),
250     ),
251     ),
252     ),
253     ),
254     ),
255     ),
256     ),
257     ),
258     ),
259     ),
260     ),
261     ),
262     ),
263     ),
264     ),
265     ),
266     ),
267     ),
268     ),
269     ),
270     ),
271     ),
272     ),
273     ),
274     ),
275     ),
276     ),
277     ),
278     ),
279     ),
280     ),
281     ),
282     ),
283     ),
284     ),
285     ),
286     ),
287     ),
288     ),
289     ),
290     ),
291     ),
292     ),
293     ),
294     ),
295     ),
296     ),
297     ),
298     ),
299     ),
300     ),
301     ),
302     ),
303     ),
304     ),
305     ),
306     ),
307     ),
308     ),
309     ),
310     ),
311     ),
312     ),
313     ),
314     ),
315     ),
316     ),
317     ),
318     ),
319     ),
320     ),
321     ),
322     ),
323     ),
324     ),
325     ),
326     ),
327     ),
328     ),
329     ),
330     ),
331     ),
332     ),
333     ),
334     ),
335     ),
336     ),
337     ),
338     ),
339     ),
340     ),
341     ),
342     ),
343     ),
344     ),
345     ),
346     ),
347     ),
348     ),
349     ),
350     ),
351     ),
352     ),
353     ),
354     ),
355     ),
356     ),
357     ),
358     ),
359     ),
360     ),
361     ),
362     ),
363     ),
364     ),
365     ),
366     ),
367     ),
368     ),
369     ),
370     ),
371     ),
372     ),
373     ),
374     ),
375     ),
376     ),
377     ),
378     ),
379     ),
380     ),
381     ),
382     ),
383     ),
384     ),
385     ),
386     ),
387     ),
388     ),
389     ),
390     ),
391     ),
392     ),
393     ),
394     ),
395     ),
396     ),
397     ),
398     ),
399     ),
400     ),
401     ),
402     ),
403     ),
404     ),
405     ),
406     ),
407     ),
408     ),
409     ),
410     ),
411     ),
412     ),
413     ),
414     ),
415     ),
416     ),
417     ),
418     ),
419     ),
420     ),
421     ),
422     ),
423     ),
424     ),
425     ),
426     ),
427     ),
428     ),
429     ),
430     ),
431     ),
432     ),
433     ),
434     ),
435     ),
436     ),
437     ),
438     ),
439     ),
440     ),
441     ),
442     ),
443     ),
444     ),
445     ),
446     ),
447     ),
448     ),
449     ),
450     ),
451     ),
452     ),
453     ),
454     ),
455     ),
456     ),
457     ),
458     ),
459     ),
460     ),
461     ),
462     ),
463     ),
464     ),
465     ),
466     ),
467     ),
468     ),
469     ),
470     ),
471     ),
472     ),
473     ),
474     ),
475     ),
476     ),
477     ),
478     ),
479     ),
480     ),
481     ),
482     ),
483     ),
484     ),
485     ),
486     ),
487     ),
488     ),
489     ),
490     ),
491     ),
492     ),
493     ),
494     ),
495     ),
496     ),
497     ),
498     ),
499     ),
500     ),
501     ),
502     ),
503     ),
504     ),
505     ),
506     ),
507     ),
508     ),
509     ),
510     ),
511     ),
512     ),
513     ),
514     ),
515     ),
516     ),
517     ),
518     ),
519     ),
520     ),
521     ),
522     ),
523     ),
524     ),
525     ),
526     ),
527     ),
528     ),
529     ),
530     ),
531     ),
532     ),
533     ),
534     ),
535     ),
536     ),
537     ),
538     ),
539     ),
540     ),
541     ),
542     ),
543     ),
544     ),
545     ),
546     ),
547     ),
548     ),
549     ),
550     ),
551     ),
552     ),
553     ),
554     ),
555     ),
556     ),
557     ),
558     ),
559     ),
560     ),
561     ),
562     ),
563     ),
564     ),
565     ),
566     ),
567     ),
568     ),
569     ),
570     ),
571     ),
572     ),
573     ),
574     ),
575     ),
576     ),
577     ),
578     ),
579     ),
580     ),
581     ),
582     ),
583     ),
584     ),
585     ),
586     ),
587     ),
588     ),
589     ),
590     ),
591     ),
592     ),
593     ),
594     ),
595     ),
596     ),
597     ),
598     ),
599     ),
600     ),
601     ),
602     ),
603     ),
604     ),
605     ),
606     ),
607     ),
608     ),
609     ),
610     ),
611     ),
612     ),
613     ),
614     ),
615     ),
616     ),
617     ),
618     ),
619     ),
620     ),
621     ),
622     ),
623     ),
624     ),
625     ),
626     ),
627     ),
628     ),
629     ),
630     ),
631     ),
632     ),
633     ),
634     ),
635     ),
636     ),
637     ),
638     ),
639     ),
640     ),
641     ),
642     ),
643     ),
644     ),
645     ),
646     ),
647     ),
648     ),
649     ),
650     ),
651     ),
652     ),
653     ),
654     ),
655     ),
656     ),
657     ),
658     ),
659     ),
660     ),
661     ),
662     ),
663     ),
664     ),
665     ),
666     ),
667     ),
668     ),
669     ),
670     ),
671     ),
672     ),
673     ),
674     ),
675     ),
676     ),
677     ),
678     ),
679     ),
680     ),
681     ),
682     ),
683     ),
684     ),
685     ),
686     ),
687     ),
688     ),
689     ),
690     ),
691     ),
692     ),
693     ),
694     ),
695     ),
696     ),
697     ),
698     ),
699     ),
700     ),
701     ),
702     ),
703     ),
704     ),
705     ),
706     ),
707     ),
708     ),
709     ),
710     ),
711     ),
712     ),
713     ),
714     ),
715     ),
716     ),
717     ),
718     ),
719     ),
720     ),
721     ),
722     ),
723     ),
724     ),
725     ),
726     ),
727     ),
728     ),
729     ),
730     ),
731     ),
732     ),
733     ),
734     ),
735     ),
736     ),
737     ),
738     ),
739     ),
740     ),
741     ),
742     ),
743     ),
744     ),
745     ),
746     ),
747     ),
748     ),
749     ),
750     ),
751     ),
752     ),
753     ),
754     ),
755     ),
756     ),
757     ),
758     ),
759     ),
760     ),
761     ),
762     ),
763     ),
764     ),
765     ),
766     ),
767     ),
768     ),
769     ),
770     ),
771     ),
772     ),
773     ),
774     ),
775     ),
776     ),
777     ),
778     ),
779     ),
780     ),
781     ),
782     ),
783     ),
784     ),
785     ),
786     ),
787     ),
788     ),
789     ),
790     ),
791     ),
792     ),
793     ),
794     ),
795     ),
796     ),
797     ),
798     ),
799     ),
800     ),
801     ),
802     ),
803     ),
804     ),
805     ),
806     ),
807     ),
808     ),
809     ),
810     ),
811     ),
812     ),
813     ),
814     ),
815     ),
816     ),
817     ),
818     ),
819     ),
820     ),
821     ),
822     ),
823     ),
824     ),
825     ),
826     ),
827     ),
828     ),
829     ),
830     ),
831     ),
832     ),
833     ),
834     ),
835     ),
836     ),
837     ),
838     ),
839     ),
840     ),
841     ),
842     ),
843     ),
844     ),
845     ),
846     ),
847     ),
848     ),
849     ),
850     ),
851     ),
852     ),
853     ),
854     ),
855     ),
856     ),
857     ),
858     ),
859     ),
860     ),
861     ),
862     ),
863     ),
864     ),
865     ),
866     ),
867     ),
868     ),
869     ),
870     ),
871     ),
872     ),
873     ),
874     ),
875     ),
876     ),
877     ),
878     ),
879     ),
880     ),
881     ),
882     ),
883     ),
884     ),
885     ),
886     ),
887     ),
888     ),
889     ),
890     ),
891     ),
892     ),
893     ),
894     ),
895     ),
896     ),
897     ),
898     ),
899     ),
900     ),
901     ),
902     ),
903     ),
904     ),
905     ),
906     ),
907     ),
908     ),
909     ),
910     ),
911     ),
912     ),
913     ),
914     ),
915     ),
916     ),
917     ),
918     ),
919     ),
920     ),
921     ),
922     ),
923     ),
924     ),
925     ),
926     ),
927     ),
928     ),
929     ),
930     ),
931     ),
932     ),
933     ),
934     ),
935     ),
936     ),
937     ),
938     ),
939     ),
940     ),
941     ),
942     ),
943     ),
944     ),
945     ),
946     ),
947     ),
948     ),
949     ),
950     ),
951     ),
952     ),
953     ),
954     ),
955     ),
956     ),
957     ),
958     ),
959     ),
960     ),
961     ),
962     ),
963     ),
964     ),
965     ),
966     ),
967     ),
968     ),
969     ),
970     ),
971     ),
972     ),
973     ),
974     ),
975     ),
976     ),
977     ),
978     ),
979     ),
980     ),
981     ),
982     ),
983     ),
984     ),
985     ),
986     ),
987     ),
988     ),
989     ),
990     ),
991     ),
992     ),
993     ),
994     ),
995     ),
996     ),
997     ),
998     ),
999     ),

```

```
162 //      ),
163      IconButton(
164          icon: const Icon(Icons.replay),
165          onPressed: !webViewReady
166              ? null
167              : () {
168                  controller.reload();
169              },
170      ),
171  ],
172  );
173  },
174  );
175  }
176 }
```

```

1 import 'package:flutter/material.dart';
2 import 'package:transparent_image/transparent_image.dart';
3 //import 'package:webview_flutter/webview_flutter.dart';
4 //import 'package:flutter_html_view/flutter_html_view.dart
  ';
5 import 'package:flutter_html/flutter_html.dart';
6
7 class ReputePost extends StatelessWidget {
8
9   var post;
10   ReputePost({Key key, @required var this.post}) : super(
    key: key);
11
12   @override
13   Widget build(BuildContext context) {
14     return new Scaffold(
15       appBar: new AppBar(
16         title: new Text(post['title']['rendered']),
17         backgroundColor: Colors.redAccent,
18       ),
19       body: new Padding(
20         padding: EdgeInsets.all(16.0),
21         child: new ListView(
22           children: <Widget>[
23             new FadeInImage.memoryNetwork(
24               placeholder: kTransparentImage,
25               image: post["featured_media"] == 0
26                 ? 'images/placeholder.png'
27                 : post["_embedded"]["wp:featuredmedia"][
28                   0]["source_url"],
29             ),
30             // new Text(post['content']['rendered'].
31             // replaceAll(new RegExp(r'<[^>]*>'), ''))
32             new Html(data: post['content']['rendered'])
33             // onLaunchFail: (url) { // optional, type
34             // Function
35             // print("launch $url failed");
36             // })
37           ],
38         ),
39       );
40     }
41   }

```

```

1 import 'package:flutter/material.dart';
2 import 'package:transparent_image/transparent_image.dart';
3 //import 'package:webview_flutter/webview_flutter.dart';
4 //import 'package:flutter_html_view/flutter_html_view.dart
  ';
5 import 'package:flutter_html/flutter_html.dart';
6
7 class ReputePage extends StatelessWidget {
8
9   var page;
10   ReputePage({Key key, @required var this.page}) : super(
    key: key);
11
12   @override
13   Widget build(BuildContext context) {
14     return new Scaffold(
15       appBar: new AppBar(
16         title: new Text(page['title']['rendered']),
17         backgroundColor: Colors.redAccent,
18       ),
19       body: new Padding(
20         padding: EdgeInsets.all(16.0),
21         child: new ListView(
22           children: <Widget>[
23             new FadeInImage.memoryNetwork(
24               placeholder: kTransparentImage,
25               image: page["featured_media"] == 0
26                 ? 'images/placeholder.png'
27                 : page["_embedded"]["wp:featuredmedia"][
28                   0]["source_url"],
29             ),
30             // new Text(page['content']['rendered'].
31             // replaceAll(new RegExp(r'<[^>]*>'), ''))
32             new Html(data: page['content']['rendered'])
33             // onLaunchFail: (url) { // optional, type
34             // Function
35             // print("launch $url failed");
36             // })
37           ],
38         ),
39       );

```