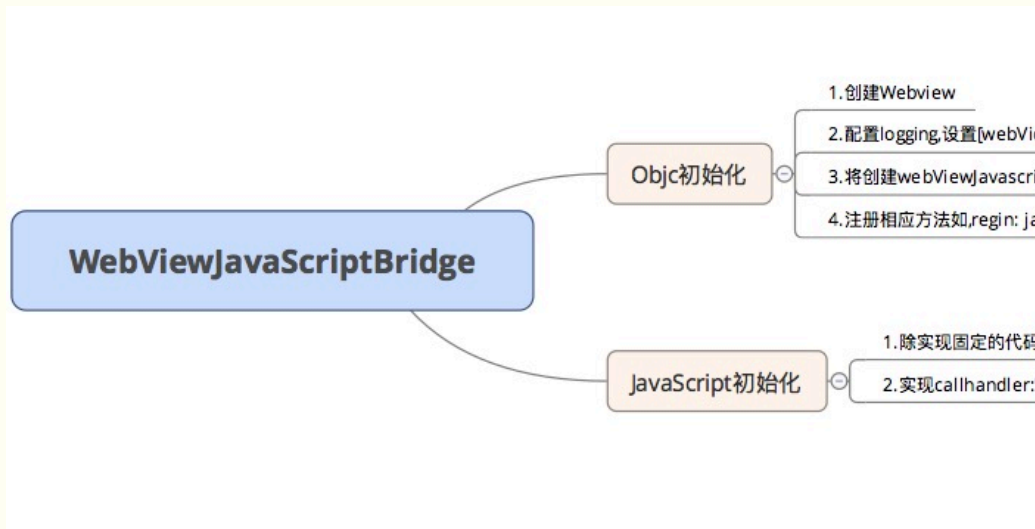


iOS与Html5和JS之间的交互---学习笔记五

首先采用的框架是WebViewJavaScriptBridge，采用这套框架可以方便的使iOS与JS交互

一. 流程图（主要介绍思路）



二.iOS端如何使用

```
首先导入#import "WebViewJavaScriptBridge.h"文件

第一步：开启日志

// 开启日志，方便调试
[WebViewJavaScriptBridge enableLogging];

第二步：给Objc与JS建立桥梁

// 给哪个webview建立JS与Objc的沟通桥梁
self.bridge = [WebViewJavaScriptBridgebridgeForWebView:webView];

// 设置代理，如果不需要实现，可以不设置
[self.bridgesetWebViewDelegate:self];

第三步：注册HandleName，用于给JS端调用iOS端

// JS主动调用Objc的方法
// 这是JS会调用getIdFromObjc方法，这是OC注册给JS调用的
// JS需要回调，当然JS也可以传参数过来。data就是JS所传的参数，不一定需要传
// OC端通过responseCallback回调JS端，JS就可以得到所需要的数据
[self.bridgeregisterHandler:@"getIdFromObjc"handler:^(id data, WVJBResponseCallback
responseCallback) {
    NSLog(@"js call getIdFromObjc, data from js is %@", data);
    if (responseCallback) {
        // 反馈给JS
        responseCallback(@{"userId": @"123456"});
    }
}];
```

昵称: TheYouth

园龄: 1年6个月

粉丝: 0

关注: 0

+加关注

2017年3月						
日	一	二	三	四	五	六
26	27	28	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	1
2	3	4	5	6	7	8

搜索

常用链接

我的随笔
我的评论
我的参与
最新评论
我的标签
更多链接

我的标签

iOS基础(39)
网络哪些事(18)
swift 语言(16)
UIKit(15)
学习资料(14)
iOS高级(10)
ScrollView哪些事(9)
App上线审核哪些事和证书(7)
代码编写与管理工具哪些事(6)
C语言(5)
更多

随笔档案

2017年3月 (25)
2017年2月 (18)
2017年1月 (6)
2016年12月 (1)
2016年11月 (7)
2016年10月 (1)
2016年9月 (5)
2016年8月 (5)
2016年6月 (3)
2016年5月 (5)
2016年4月 (23)

```

    }
    }];

[self.bridgeregisterHandler:@"getBlogNameFromObjC"handler:^(id data, WVJBResponseCallback
responseCallback) {
    NSLog(@"js call getBlogNameFromObjC, data from js is %@", data);
    if (responseCallback) {
        // 反馈给JS
        responseCallback(@{@"blogName": @"星仔"});
    }
}
}];

```

第四步：直接调用JS端注册的HandleName

```

[self.bridgecallHandler:@"getUserInfos"data:@{@"name": @"ddsds"}responseCallback:^(id
responseData) {
    NSLog(@"from js: %@", responseData);
}];

```



三.JS端如何使用



```

<script>
    window.onerror = function(err) {
        log('window.onerror: ' + err)
    }

    /*这段代码是固定的，必须要放到js中*/
    function setupWebViewJavascriptBridge(callback) {
        if (window.WebViewJavascriptBridge) { return callback(WebViewJavascriptBridge); }
        if (window.WVJBCallbacks) { return window.WVJBCallbacks.push(callback); }
        window.WVJBCallbacks = [callback];
        var WVJBIframe = document.createElement('iframe');
        WVJBIframe.style.display = 'none';
        WVJBIframe.src = 'wvjscheme://__BRIDGE_LOADED__';
        document.documentElement.appendChild(WVJBIframe);
        setTimeout(function() { document.documentElement.removeChild(WVJBIframe) }, 0)
    }

    /*与OC交互的所有JS方法都要放在此处注册，才能调用通过JS调用OC或者让OC调用这里的JS*/
    setupWebViewJavascriptBridge(function(bridge) {
        var uniqueId = 1
        function log(message, data) {
            var log = document.getElementById('log')
            var el = document.createElement('div')
            el.className = 'logLine'
            el.innerHTML = uniqueId++ + ' ' + message + ' :<br/>' + JSON.stringify(data)
            if (log.children.length) {
                log.insertBefore(el, log.children[0])
            } else {
                log.appendChild(el)
            }
        }
    })
    /* Initialize your app here */

    /*我们在这注册一个js调用OC的方法，不带参数，且不用ObjC端反馈结果给JS：打开本demo对应的博文*/
    bridge.registerHandler('openWebviewBridgeArticle', function() {
        log("openWebviewBridgeArticle was called with by ObjC")
    })
    /*JS给ObjC提供公开的API，在ObjC端可以手动调用JS的这个API。接收ObjC传过来的参数，且可以回调ObjC*/-
    bridge.registerHandler('token', function(data, responseCallback) {
        log("G星爷: ", data)
        responseCallback({这里给我返回拼接后的地址})
    })

    /*JS给ObjC提供公开的API，ObjC端通过注册，就可以在JS端调用此API时，得到回调。ObjC端可以在处理完成后，反馈给JS，这样写就是在载入页面完成时就先调用*/
    bridge.callHandler('getUserIdFromObjC', function(responseData) {

```

2016年3月 (8)
 2016年2月 (5)
 2016年1月 (9)
 2015年12月 (8)
 2015年11月 (20)
 2015年10月 (8)
 2015年9月 (1)

☕ 阅读排行榜

1. iOS与Html5和JS之间的交互---学习笔记五(1554)
2. iOS 使用证书时遇到的错误一(1430)
3. iOS基础 --- UILabel的使用及对齐方式(1198)
4. 审核被拒(后台定位,autio,voip,发表朋友圈)(898)
5. XCodeo如何去除多余的模拟器---学习笔记七(536)

☕ 推荐排行榜

1. iOS与Html5和JS之间的交互---学习笔记五(1)


```

        log("JS call ObjC's getUserIdFromObjC function, and js received response:",
responseData)
    })

    document.getElementById('blogIds').onclick = function (e) {
        log('js call objc: getBlogNameFromObjC')
        bridge.callHandler('ww', {'blogdURL': 'weidsfdl'}, function(response) {
            log('JS got response', response)
        })
    }
})

</script>

```



四.注意事项

1.当出现这样的提示的话，说明JS那边已经实现了名字为'ww'的方法，但是iOS这边没有注册，

```

2016-09-08 20:38:20.676 WebViewBridgeDemo[1508:259517] WVJBNoHandlerException, No handler for mess
from JS: {
  callbackId = "cb_2_1473338300672";
  data = {
    blogURL = weidsfdl;
  };
  handlerName = ww;
}

```

具体实现是：

```

[self.bridgeregisterHandler:@"ww"handler:^(id data, WVJBResponseCallback responseCallback) {
    NSLog(@"js call getBlogNameFromObjC, data from js is %@", data);
    if (responseCallback) {
        // 反馈给JS
        responseCallback(@{"blogName": @"星仔"});
    }
}];

```

2. 当在push或者说页面消失的时候，将bridge对象置为空。

五.防止WebView长按实现的方法如下：



```

- (void)webViewDidFinishLoad:(UIWebView *)webView{
    // 禁用 页面元素选择
    [webView
 stringByEvaluatingJavaScriptFromString:@"document.documentElement.style.webkitUserSelect=
'none';"];

    // 禁用 长按弹出ActionSheet
    [webView
 stringByEvaluatingJavaScriptFromString:@"document.documentElement.style.webkitTouchCallou
t='none';"];
}

```



将来的自己,会感谢现在不放弃的自己!

标签： [iOS高级](#)， [第三方及项目框架及结构](#)

好文要顶

关注我

收藏该文



TheYouth

[关注 - 0](#)

[粉丝 - 0](#)

[+加关注](#)

1

推荐

0


反对

« 上一篇： [UINavigationController设置\(一\)](#)

» 下一篇: [iOS状态栏---学习笔记六](#)

posted @ 2016-09-08 20:56 TheYouth 阅读(1553) 评论(0) 编辑 收藏

[刷新评论](#) [刷新页面](#) [返回顶部](#)

 注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问网站首页](#)。

【推荐】50万行VC++源码：大型组态工控、电力仿真CAD与GIS源码库

【免费】自开发零实施的H3 BPM免费下载

【推荐】Google+GitHub联手打造前端工程师课程



最新IT新闻：

- 豆瓣宣布“豆瓣时间”销售额过百万元 付费用户过万
- 2017谷歌云大会，一口气发布100+消息
- NASA太远 阿里的技术实力需要跨过阿尔法狗的尸体才能证明
- 修单车师傅再就业 这些共享单车“好可怜”
- 美国人愿意尝试实验室培育的肉

» 更多新闻...



最新知识库文章：

- 为什么我要写自己的框架？
- 垃圾回收原来是这么回事
- 「代码家」的学习过程和学习经验分享
- 写给未来的程序媛
- 高质量的工程代码为什么难写

» 更多知识库文章...