目 录

[混搭模式移动应用框架设计文档 1](#_Toc346808769)

[项目简介 1](#_Toc346808770)

[总体设计 2](#_Toc346808771)

[Bridge 2](#_Toc346808772)

[Native功能插件 2](#_Toc346808773)

[NEJ Native集成 2](#_Toc346808774)

[移动端Web框架 2](#_Toc346808775)

[开发工具 3](#_Toc346808776)

[第三方功能集成 3](#_Toc346808777)

[详细设计 3](#_Toc346808778)

[IOS 3](#_Toc346808779)

[Android 3](#_Toc346808780)

# 文档修订

|  |  |  |  |
| --- | --- | --- | --- |
| **版本** | **变更说明** | **作者** | **时间** |
| V0.1 | 创建文档 | 胡雪亮 | 2013.1.24 |
| V0.11 | IOS总体设计修改 | 胡雪亮 | 2013.3.25 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# 混搭模式移动应用框架设计文档

## 项目简介

混搭模式移动应用框架旨在提供一套移动端跨平台应用开发的完整解决方案，它使得开发者可以使用Web技术来构建跨平台移动应用程序，并且可以使用智能手机的核心功能（比如拍照、联系人、声音、振动等）

## 总体设计

**Native**

**Bridge**

**WebView**

NEJ Native集成

Native功能插件

移动端Web框架

开

发

工

具

第三方功能集成

### Bridge

连接Native与Web的桥梁，统一处理Web和Native之间的交互逻辑，使得Web与Native之间能够相互调用。

接收从Web端发送的调用Native功能的指令和参数，并将指令转换成调用Native功能。

发送Web调用Native功能指令，同时统一接收Native调用Web回调。

### Native功能插件

按照功能对Native API进行分类，以插件形式提供给开发者，同时提供插件开发规范，便于开发者根据需求扩展。

### NEJ Native集成

将Native功能提供的接口集成到NEJ，Web开发者通过NEJ接口统一调用Native。在NEJ中会使用HTML5规范适配，如果Webview未实现HTML5相应规范，则调用Native API。

### 移动端Web框架

主要包括移动端跨平台应用Web项目框架、移动端应用开发所需组件扩展（触摸手势、动画、模拟滚动等），这块内容主要在NEJ扩展。

### 开发工具

项目模板生成工具、项目打包和调试工具、手册与文档、App设计开发工具

### 第三方功能集成

新浪腾讯开放接口整合、移动应用统计分析、消息推送API整合等功能

## 详细设计

### IOS

#### 概述

Web调用Native时，调用指令放到Web端的一个指令队列中，同时Web通知Native，Native接收到通知之后，调用Web端接口获取指令队列内容，通过反射机制处理指令，调用相应Native API。

Native回调Web时，统一调用Web端某个接口，由该接口回调到具体的回调函数中。

#### 代码规范

Web端代码遵循NEJ规范

Native代码遵循Google Objective-C代码规范

#### 流程图

见附件

#### Web

##### NEH

提供对外API，供产品和Native调用

|  |  |  |  |
| --- | --- | --- | --- |
| **属性/方法名** | **输入** | **输出** | **备注** |
| neh.exec | 调用Native指令对象 |  | 调用Native API(Web) |
| neh.setWebViewKey | WebView标示 |  | 设置WebViewKey  (call by native) |
| neh.getCommands |  |  | 输出所有调用Native的指令(call by native) |
| neh.callback | 回调结果 |  | 统一回调函数回调函数(call by native) |

##### commandQueue

|  |  |  |  |
| --- | --- | --- | --- |
| **属性/方法名** | **输入** | **输出** | **备注** |
| add | 指令对象 |  | 添加调用Native指令 |
| output2JSON |  | JSON格式字符串 | 输出所有指令元素(JSON格式) |

##### callbackManager

|  |  |  |  |
| --- | --- | --- | --- |
| **属性/方法名** | **输入** | **输出** | **备注** |
| add | 回调对象 | 回调函数ID | 添加一个回调 |
| get | 回调函数ID |  | 获取一个回调 |
| Remove | 回调函数ID |  | 移除一个回调 |

#### Native

##### NEHUIWebView

:UIView

|  |  |  |  |
| --- | --- | --- | --- |
| **属性/方法名** | **输入** | **输出** | **备注** |
| loadRequest |  |  | 加载请求 |
| stringByEvaluatingJavascriptFromString |  |  | 执行脚本 |
| stopLoading |  |  | 停止加载 |
| goBack |  |  | 后退 |
| goForwad |  |  | 前进 |
| isLoading |  |  |  |
| canGoBack |  |  |  |
| canGoForward |  |  |  |
| reload |  |  |  |

<WebView Delegate>

|  |  |  |  |
| --- | --- | --- | --- |
| **属性/方法名** | **输入** | **输出** | **备注** |
| shouldStartLoadWithRequest |  |  | Webview请求拦截 |
|  |  |  |  |
|  |  |  |  |

##### NEHHost（Webview Host）

|  |  |  |  |
| --- | --- | --- | --- |
| **属性/方法名** | **输入** | **输出** | **备注** |
| webView |  |  | webView属性 |
| initWithWebView |  |  | 使用WebView初始化 |
| evalJs |  |  | 执行Js |
| getCommandFromJs |  |  | 获取Web调用Native指令 |
| callbackJsWithCallbackId /result/keepCallback |  |  | 回调Web |

##### NEHHostManager（Webiew Host管理）

|  |  |  |  |
| --- | --- | --- | --- |
| **属性/方法名** | **输入** | **输出** | **备注** |
| addHostForKey |  |  | 添加Host |
| removeHostForKey |  |  | 移除Host |
| getHostForKey |  |  | 获取Host |

##### NEHURLProtocol（拦截协议）

:NSURLProtocol

|  |  |  |  |
| --- | --- | --- | --- |
| **属性/方法名** | **输入** | **输出** | **备注** |
| canInitWithRequest |  |  | 拦截指令 |
|  |  |  |  |

##### NEHCommand（指令对象）

:NSObject

|  |  |  |  |
| --- | --- | --- | --- |
| **属性/方法名** | **输入** | **输出** | **备注** |
| Host |  |  | webView Host |
| className |  |  | 类名 |
| methodName |  |  | 方法名 |
| callbackId |  |  | 回调函数id |
| Arguments |  |  | 调用参数 |
| initWithJSON host |  |  | 使用JSON初始化 |
| execute |  |  | 执行指令 |

##### NEHLog（日志对象）

:NEHModule

|  |  |  |  |
| --- | --- | --- | --- |
| **属性/方法名** | **输入** | **输出** | **备注** |
| write |  |  | 写日志 |

##### NEHDevice（设备对象）

:NEHModule

|  |  |  |  |
| --- | --- | --- | --- |
| **属性/方法名** | **输入** | **输出** | **备注** |
| getDeviceInfo |  |  | 获得设备属性 |

##### NEHModule（插件基类）

:NSObject

|  |  |  |  |
| --- | --- | --- | --- |
| **属性/方法名** | **输入** | **输出** | **备注** |
| Host |  |  | Host对象 |
| initWithHost | Host |  | 使用host初始化 |
| successWithCallbackId |  |  | 成功返回 |

##### NEHArgument（插件函数调用参数）

:NSObject

|  |  |  |  |
| --- | --- | --- | --- |
| **属性/方法名** | **输入** | **输出** | **备注** |
| callbackId |  |  | 回调id |
| Arguments |  |  | 方法调用参数 |
| initWithCallbackId/  arguments: |  |  | 初始化 |

##### NEHResult（插件函数调用结果）

:NSObject

|  |  |  |  |
| --- | --- | --- | --- |
| **属性/方法名** | **输入** | **输出** | **备注** |
| Code |  |  | 状态码 |
| Message |  |  |  |
| Data |  |  |  |
| initWithCode/ message/data |  |  |  |
| initWithSuccessData |  |  |  |
| initWithErrorMessage |  |  |  |
| resultToString |  |  |  |

### Android

#### 跨平台应用Web项目框架

#### 动画

#### 模拟滚动解决方案