**Android调试1：(推荐)**

**首先有几个先决条件：**

1. Android手机要Android 4.4以上版本（手机上查看关于设备的信息）；Android手机上安装Chrome for android 32 以及之后的版本（即手机上的chrome app）；
2. APP的Android 里针对 Webview 做以下设置：

if(Build.VERSION.SDK\_INT >= Build.VERSION\_CODES.KITKAT) { WebView.setWebContentsDebuggingEnabled(true); }

这个确认过我们的应用已经增加过

1. pc上同样需要chrome最低需要 32 版本（我记得以前看见过需要pc上的版本比安卓上的高，但没证实过）；windows 用户需要安装iTools(目的是为了装Android 驱动)；且PC需要翻墙（需要goagent之类的翻墙工具（试过没有装的话打不开inspect页面，可能需要加载一些文件））

**接着就是步骤**：

1. 开启 Android 的 USB 调试功能。
2. 用 USB 先连接到你的电脑。
3. 在自己的电脑的chrome上打入地址chrome://inspect/#devices，勾选Discover USB devices
4. 在手机的Chrome 上打开你要调试的网页，就可以在自己的电脑的chrome（chrome://inspect/#devices）上远程调式；在自己手机中打开app应用浏览到相应的页面，同样可以看见pc的检查设备页面内相应的页面也可以inspect远程调试了

**Android调试2：**

（不需要dev环境，不需要配置ip，设置任何东西,甚至可以用线上的生产包）

用grunt打包好以后，获取diyfhx或者diyfhxsdp文件，把获得的文件直接拷贝到

data>data>ctrip.android.view>app\_ctripwebapp下替换原来的文(需要装rootExplore：勾选挂载后是否可以直接在电脑上覆盖文件)

文件夹能访问的是data>data的文件夹

**IOS调试：**

**方法1：**

安装最新ios测试包

1. 要进行远程调试，首先要打开开启 iPhone/iPad 上的 Safari 的远程调试功能，“通过 设置 > Safari > 高级”开启：Web检查器
2. 在mac上的safari中需要展示开发标签栏（左上角点击safari，选择高级，然后给底部“在菜单栏中显示开发菜单”打钩）
3. 这时候在手机ios的safari浏览器打开对应的文件，就可以在mac的开发>陈佳杰的iphone看到对应问错误
4. 如果想要调试app包里面的错误，还是直接用代理比较方便，用fat395环境，用fiddler把对应文件替换成开发文件，然后再app的wifi中设置代理机器为自己的电脑，秒改，马上找到问题

**方法2：**

安装最新ios测试包

在h5设置中绑定“自己的ip：5389/webapp/diyfhx“就可以在ios上看到本地的dp代码效果了（注意是压缩文件还是其他文件）

**android和ios通用调试**

**（可以修改任何包，生产包和测试包都可以）：（不推荐）**

1. 代码提交到git
2. 测试那边编译一下
3. 编译好以后，从git上对于的发布地址获取最新的sdp和sp的压缩文件包，获得对应的diyfhx和diyfhxsdp文件夹
4. **打开itools，在手机中选择应用，点击文件共享，删除/Documents/webapp\_work路径下的**diyfhx和diyfhxsdp文件夹，然后再倒入第一步获取的2个diyfhx和diyfhxsdp文件夹

公用方法2：（推荐）

1. 自己开发机器上打开fiddler（这时候，本地的开发环境5389端口必须关闭，所以一般用测试环境的代码），让本地的某个开发文件的代码代替测试环境的某个文件。（这个就可以用本地开发文件给“线上或者测试环境”调试错误，开发功能等）
2. 给子机的手机wifi设置代理，代理的ip为开发机电脑ip，端口为公司允许的5389端口（端口因不同公司而不同，有的公司压根就没有禁用端口）
3. 这样自己手机的浏览器或者app发送请求，会首先代理到本地的开发机器，开发机发送请求的时候，又会被fiddler捕获，让后代理成fiddler中设置的文件