

weex debug 断点调试使用文档

- 一、准备工作
- 二、插件改造
- 三、启动调试
- 四、视频操作

一、准备工作

美居site8.23以上版本（移动开发平台下载）

PC和手机网络必须在同一个局域网并且可联通

Mac M1芯片的电脑暂时不能调试

step1 安装

```
# Windowsnpmnpm install -g npm@^8
npm config set puppeteer-download-host "https://cdn.npmmirror.com/binaries"

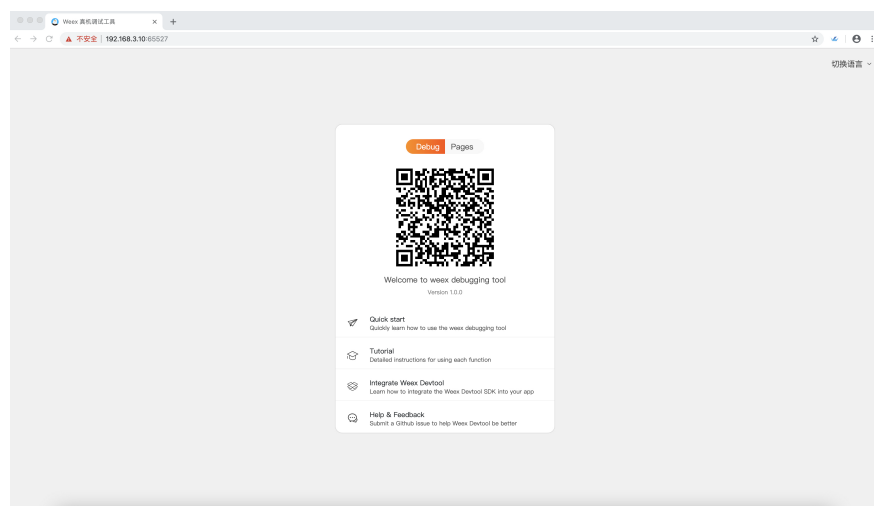
npm install -g dolphin-weex-debug
```

step2 运行

```
# Windows
dolphinDebug.cmd

# MAC
dolphinDebug
```

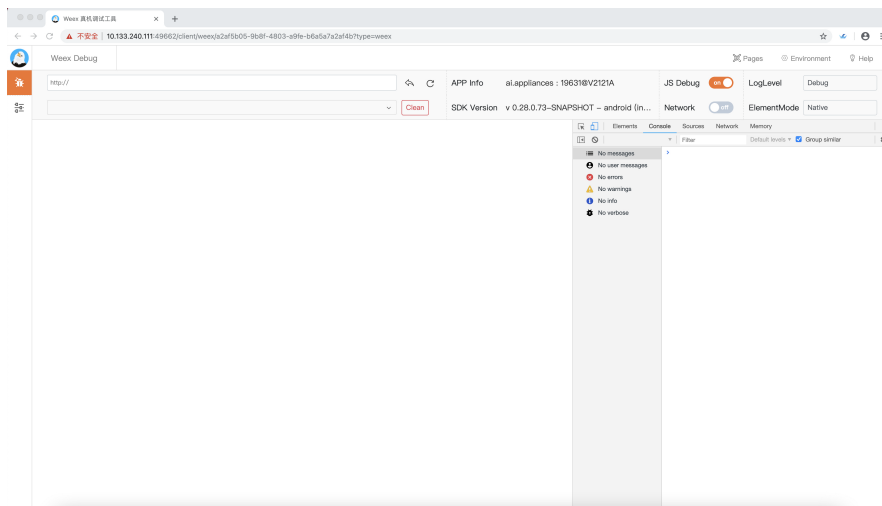
运行命令后出现下面界面



step3 用美居APP(version>=8.23 sit) 扫码

安卓：美居 APP-》我的-》设置-》关于美居-》长按美居 logo，下面出现一个“扫一扫”

iOS： 美居 APP-》左下角浮动的小黑球》混合扫描



二、插件改造

插件有新模板和旧模板区别，下面部分为新模板改造

2.1 \${project}/package.json

- devDependencies 新增"@dolphinweex/weex-loader": "^0.2.6"
- (可选) dependencies 如果依赖了dolphin-native-bridge, dolphin-weex-ui, 必须升级 "dolphin-native-bridge": "^0.4.0"、"dolphin-weex-ui": "^1.4.5"，升级完后需要重新npm install

2.2 \${project}/configs/webpack.common.conf

```
const weexConfig = getBaseConfig();
weexConfig.entry = weexEntry;
weexConfig.output.filename = '[name].js';
weexConfig.module.rules[1].use.push(
  {
    loader: '@dolphinweex/weex-loader', // 'weex-loader' '@dolphinweex/weex-loader'
  }
);
```

2.3 \${project}/configs/webpack.devdev.conf

```

/**
 * Webpack configuration for weex.
 */
const weexConfig = webpackMerge(commonConfig[1], {
  devtool: 'inline-source-map', // inline-source-map
  plugins: [
    /**
     * Plugin: webpack.DefinePlugin
     * Description: The DefinePlugin allows you to create global constants which can be configured at compile
    time.
     *
     * See: https://webpack.js.org/plugins/define-plugin/
     */
    new webpack.DefinePlugin({
      "process.env": {
        NODE_ENV: config.dev.env
      }
    })
  ],
  watch: true
});

```

2.4 启动项目

npm run start 或者 npm run serve



Use Weex playground app to scan it.

device logs moreMenuItems weex

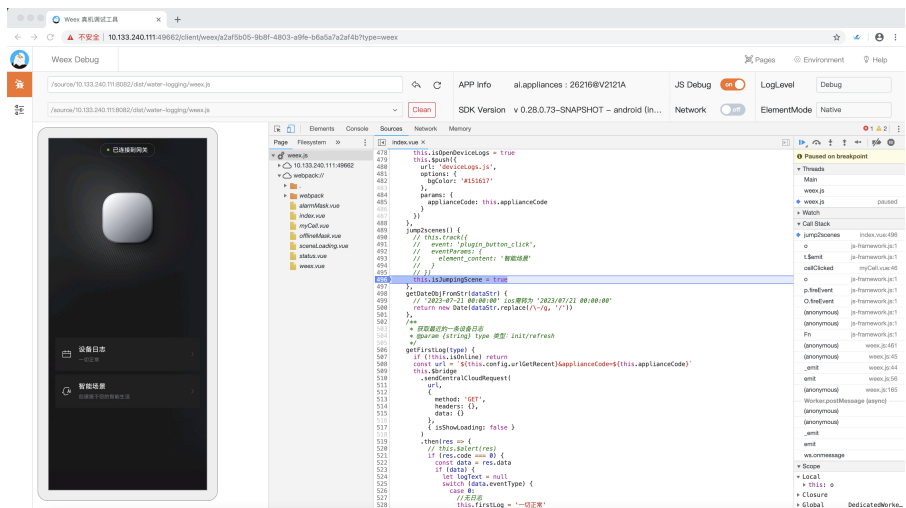
三、启动调试

3.1 上面准备好的美居APP再次扫码

打断点的方式

- 代码里面写入debugger
- 在source 标签中找到\${source}.js/webpack:// 下面找到对应的源码文件打断点

接下来的调试就基本和 chrome 中调试一致了



四、视频操作

Your browser does not support the HTML5 video element