

——微信与移动Web开发之

第10讲 JS-SDK

本次课程目录

- 一 . JS-SDK基础
- 二 . JS-SDK基本使用
- 三 . JS-SDK基本使用 (续)

课程概要

- JS-SDK的配置与签名验证。
- JS-SDK的接口开发。

第一节

- 一 . JS-SDK基础
- 二 . JS-SDK基本使用
- 三 . JS-SDK基本使用 (续)

JS-SDK介绍

- 微信JS-SDK是微信公众平台面向网页开发者提供的基于微信内的网页开发工具包。
- 通过使用微信JS-SDK，网页开发者可借助微信高效地使用拍照、选图、语音、位置等手机系统的能力，同时可以直接使用微信分享、扫一扫、卡券、支付等微信特有的能力，为微信用户提供更优质的网页体验。

JS-SDK使用步骤

步骤一：绑定域名

先登录微信公众平台进入“公众号设置”的“功能设置”里填写“JS接口安全域名”。

备注：登录后可在“开发者中心”查看对应的接口权限。

步骤二：引入JS文件

在需要调用JS接口的页面引入如下JS文件，（支持https）：<http://res.wx.qq.com/open/js/jweixin-1.2.0.js>（备注：支持使用 AMD/CMD 标准模块加载方法加载）

步骤三：通过config接口注入权限验证配置

所有需要使用JS-SDK的页面必须先注入配置信息，否则将无法调用（同一个url仅需调用一次，对于变化url的SPA的web app可在每次url变化时进行调用，目前Android微信客户端不支持pushState的H5新特性，所以使用pushState来实现web app的页面会导致签名失败，此问题会在Android6.2中修复）。

步骤四：通过ready接口处理成功验证

步骤五：通过error接口处理失败验证

配置JS-SDK

```
wx.config({
```

 debug: true, // 开启调试模式,调用的所有api的返回值会在客户端alert出来, 若要查看传入的参数, 可以在pc端打开, 参数信息会通过log打出, 仅在pc端时才会打印。

```
    appId: "", // 必填, 公众号的唯一标识
```

```
    timestamp: "", // 必填, 生成签名的时间戳
```

```
    nonceStr: "", // 必填, 生成签名的随机串
```

```
    signature: "", // 必填, 签名, 见附录1
```

```
    jsApiList: [] // 必填, 需要使用的JS接口列表, 所有JS接口列表见附录2
```

```
});
```

JS-SDK的jsapi_ticket

- jsapi_ticket：调用JS-SDK接口的凭证，需要在生成配置的签名时提供，首先要获取access_token，然后用access_token获取jsapi_ticket。
- 正确调用返回的数据：

```
{
  "errcode":0,
  "errmsg":"ok",
  "ticket":"bxLdikRXVbTPdHSM05e5u5sUoXNKd8-
41ZO3MhKoyN5OfkWITDGgnr2fwJ0m9E8NYzWKVZvdVtaUgWvsdshFKA",
  "expires_in":7200
}
```
- expires_in表示有效时间为7200秒，类似于access_token，需要定期刷新。

签名的参数

- 生成配置签名需要的参数：

timestamp：时间戳

jsapi_ticket：接口调用票据

noncestr：随机字符串

url：使用JS-SDK的链接

- 正确配置JS-SDK的安全域名：

微信测试号中配置“**JS接口安全域名**”，注意填写一级域名即可。

调用链接：<http://www.example.com/test/test.php>

配置域名：example.com

签名的步骤

- 生成签名的步骤：
 1. 获取jsapi_ticket并生成timestamp与noncestr。
 2. 对所有待签名参数按照字段名的ASCII 码从小到大排序（字典序）。
 3. 对参数名与参数值使用url键值对的格式（key1=value1&key2=value2...）进行字符串拼接。

注意：这里所有的参数名都是小写字符，并且字段名和字段值都采用原始值，不进行URL 转义。
 4. 对拼接的字符串进行sha1加密签名：sha1(string)。

第二讲

一 . WeUI基础与使用

二 . JS-SDK基本使用

三 . JS-SDK基本使用 (续)

接口配置列表

- 请确保要调用的接口在配置列表中，否则无法调用。

ready接口的回调函数

- config信息验证后会执行ready方法，所有接口调用都必须在config接口获得结果之后，config是一个客户端的异步操作，所以如果需要在页面加载时就调用相关接口，则须把相关接口放在ready函数中调用来确保正确执行。
- 对于用户触发时才调用的接口，则可以直接调用，不需要放在ready函数中。
- 以on开头的接口都是要放到ready函数中配置的。

error接口的回调函数

- config信息验证失败会执行error函数，如签名过期导致验证失败。
- 具体错误信息可以打开config的debug模式查看，也可以在返回的res参数中查看，对于SPA可以在这里更新签名。

分享到朋友圈接口

- 接口参数:

```
wx.onMenuShareTimeline({  
  title: "", // 分享标题  
  link: "", // 分享链接, 该链接域名或路径必须与当前页面对应的公众号JS安全域名一致  
  imgUrl: "", // 分享图标  
  success: function () {  
    // 用户确认分享后执行的回调函数  
  },  
  cancel: function () {  
    // 用户取消分享后执行的回调函数  
  }  
});
```
- 调用方式:
在ready函数中配置, 用户分享页面到朋友圈会自动调用此接口。

分享给朋友接口

- 接口参数:
 wx.onMenuShareAppMessage({
 title: "", // 分享标题
 desc: "", // 分享描述
 link: "", // 分享链接, 该链接域名或路径必须与当前页面对应的公众号JS安全域名一致
 imgUrl: "", // 分享图标
 type: "", // 分享类型,music、video或link, 不填默认为link
 dataUrl: "", // 如果type是music或video, 则要提供数据链接, 默认为空
 success: function () {
 // 用户确认分享后执行的回调函数
 },
 cancel: function () {
 // 用户取消分享后执行的回调函数
 }
 });
- 调用方式:
 在ready函数中配置, 用户分享页面到朋友圈会自动调用此接口。

关闭页面接口

- 调用接口:
`wx.closeWindow();`

第三讲

一 . WeUI基础与使用

二 . JS-SDK基本使用

三 . JS-SDK基本使用 (续)

拍照或从手机相册中选图接口

- 接口参数：

```
wx.chooseImage({  
  count: 1, // 默认9  
  sizeType: ['original', 'compressed'], // 可以指定是原图还是压缩图，默认二者都有  
  sourceType: ['album', 'camera'], // 可以指定来源是相册还是相机，默认二者都有  
  success: function (res) {  
    // 返回选定照片的本地ID列表，localId可以作为img标签的src属性显示图片  
    var localIds = res.localIds;  
    //上传图像，把调用接口的函数写在这里传递res.localIds到上传图像接口  
  }  
});
```

- 调用方式：

通过用户点击按钮触发此操作。

上传图像接口

- 接口参数：

```
wx.uploadImage({  
  localId: '', // 需要上传的图片的本地ID，由chooseImage接口获得  
  isShowProgressTips: 1, // 默认为1，显示进度提示  
  success: function (res) {  
    var serverId = res.serverId; // 返回图片的服务器端ID  
  }  
});
```

- 调用方式：

可以通过某一事件进行触发，也可以通过在选择图像接口的回调中进行上传。

录音接口

- 接口调用:
`wx.startRecord();`
- 执行此函数会直接进行录音，注意系统设置权限是否允许录音。

停止录音接口

- 接口参数：

```
wx.stopRecord({  
  success: function (res) {  
    var localId = res.localId;  
  }  
});
```

- 回调函数返回值是作为播放录音接口的参数使用的，所以如果不是在回调函数里面调用播放接口，要把返回值保存，方便播放接口调用。

自动停止录音接口

- 接口参数：

```
wx.onVoiceRecordEnd({  
  // 录音时间超过一分钟没有停止的时候会执行 complete 回调  
  complete: function (res) {  
    var localId = res.localId;  
  }  
});
```

- 调用方式：
在ready接口中配置。

播放录音接口

- 调用接口播放录音：
 wx.playVoice({
 localId: " // 需要播放的音频的本地ID，由stopRecord接口获得
 });

语音播放完毕接口

- 接口参数：

```
wx.onVoicePlayEnd({  
  success: function (res) {  
    var localId = res.localId; // 返回音频的本地ID  
  }  
});
```

- 调用方式：

ready接口中进行配置。

感谢聆听！

THANK YOU FOR YOUR ATTENTION