Workplat API网页接口

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **编号** | **版本** | **修改日期** | **修改人** | **修改内容** |
| 1 | V1.0 | 2016-01-02 | 郁晓勇 | 新建文档 |
| 2 | V1.1 | 2016-04-01 | 郁晓勇 | 添加接口  1.打开个人信息界面；  2.打开通讯录选择界面；  3.网页窗口刷新事件； |
| 3 | V1.2 | 2016-05-26 | 郁晓勇 | 添加接口  1.下载文件接口； |
| 4 | V2.0 | 2017-11-09 | 郁晓勇 | 所有接口调用规范调整； |
| 5 |  |  |  |  |
| 6 |  |  |  |  |

备注：请文档修订者在每个版本中注明文档修改过的内容。

[1 API简介 4](#_Toc447294054)

[2 扫码接口 4](#_Toc447294055)

[2.1 扫码接口 4](#_Toc447294056)

[3 设备基本信息 4](#_Toc447294057)

[3.1 获取设备ID 4](#_Toc447294058)

[3.2 获取设备唯一标识 4](#_Toc447294059)

[3.3 获取手机号码 4](#_Toc447294060)

[3.4 获取sim卡序列号 5](#_Toc447294061)

[3.5 获取手机MAC地址 5](#_Toc447294062)

[3.6 获取网络状态 5](#_Toc447294063)

[4 声音提示 5](#_Toc447294064)

[4.1 声音提示接口 5](#_Toc447294065)

[5 界面操作 5](#_Toc447294066)

[5.1 打开一个应用 5](#_Toc447294067)

[5.2 打开个人信息界面 6](#_Toc447294068)

[5.3 打开通讯录选择界面 6](#_Toc447294069)

[5.4 打开一个网页窗口 6](#_Toc447294070)

[5.5 关闭一个网页窗口 7](#_Toc447294071)

[5.6 设置窗口标题 7](#_Toc447294072)

[5.7 隐藏顶部状态栏（不推荐） 7](#_Toc447294073)

[5.8 显示顶部状态栏（不推荐） 7](#_Toc447294074)

[5.9 网页窗口刷新事件 7](#_Toc447294075)

[6 图像接口 8](#_Toc447294076)

[6.1 拍照或从手机相册中选图接口 8](#_Toc447294077)

[6.2 预览图片接口 8](#_Toc447294078)

[6.3 上传图片接口 8](#_Toc447294079)

[6.4 下载图片接口 8](#_Toc447294080)

[7 文件接口 8](#_Toc447294081)

[7.1 从手机选取文件接口 8](#_Toc447294082)

[7.2 上传文件接口 8](#_Toc447294083)

[7.3 下载文件接口 8](#_Toc447294084)

[7.4 在线预览接口 8](#_Toc447294085)

[8 音频接口 8](#_Toc447294086)

[8.1 开始录音接口 8](#_Toc447294087)

[8.2 停止录音接口 8](#_Toc447294088)

[8.3 播放录音接口 8](#_Toc447294089)

[8.4 暂停播放接口 8](#_Toc447294090)

[8.5 停止播放接口 9](#_Toc447294091)

[8.6 上传语音接口 9](#_Toc447294092)

[8.7 下载语音接口 9](#_Toc447294093)

[9 地理位置接口 9](#_Toc447294094)

[9.1 获取用户地理位置 9](#_Toc447294095)

[10 图片预览 10](#_Toc447294096)

[10.1 图片预览接口 10](#_Toc447294097)

[11 分享组件 10](#_Toc447294098)

[11.1 内容分享接口 10](#_Toc447294099)

# API简介

网页SDK是移动开发平台向第三方应用开发者提供方便的网页开发工具包，为接入的网页应用提供与移动设备交互的能力。

# 初始化

workplat.config({

appKey: “E9B79472466E4596A9E4894BBBAFC8D3”

});

# 注入回调

workplat.ready(function(){

console.log(“javascript已注入完成”);

});

# 获取用户信息(workplat.user)

workplat.user.getInfo(function(user){

});

# 扫码接口(workplat.qrcode)

## 扫码接口(包括条形码和二维码)

workplat.qrcode.scan({

needResult: 0, // 默认为0，扫描结果由平台处理，1则直接返回扫描结果，

success:function(result){

console.log(result); // 当needResult 为 1 时，扫码返回的结果

},

error:function(err){

console.log(err);

}

});

## 二维码识别接口(暂缓)

用于长按识别图片中的二维码

workplat.qrcode.recognize({

url:”http://xxx.png”,

success:function(result){

console.log(result);

},

error:function(err){

console.log(err);

}

});

# 设备基本信息(workplat.device)

## 读取设备信息

#获取设备信息接口

workplat.device.getInfo({

success: function(deviceInfo){

console.log(deviceInfo);

},

error: function(err){

console.log(err);

}

});

deviceInfo.uuid #返回设备的唯一标识

deviceInfo.type #返回设备的类型

deviceInfo.name #返回设备名称

deviceInfo.version #返回设备版本号

deviceInfo.mac #获取设备mac

## 开启设备震动

#使设备震动固定时长

workplat.device.vibrate(millisecond);

millisecond：震动时长，单位：毫秒

## 取消设备震动

#使设备震动停止震动

workplat.device.cancelVibrate();

# 设备网络信息(workplat.network)

## 获取WIFI接入状态

workplat.network.getWifiStatus({

success:function(data){

console.log(data.status);

console.log(data.ssid);

console.log(data.bssid);

},

error:function(err){

console.log(err);

}

});

data.status //wifi的可用状态，0：不可用，1：可用

data.ssid //wifi的名称

data.bssid //wifi的mac地址，不是本机的mac地址

## 获取网络接入类型

workplat.network.getNetworkType({

success:function(type){

console.log(type); //返回网络类型2g，3g，4g，wifi，none

},

error:function(err){

console.log(err);

}

});

success:调用正确情况下的回调函数；

error:调用失败情况下的回调函数；

# webview窗口操作(workplat.window)

## 打开网页窗口

通过此接口可以打开一个带webview的原生界面，并通过一些初始参数可以对窗口进行个性化配置。

workplat.window.open ({

[url:"http://www.baidu.com](url:%22http://www.baidu.com)",

title:"标题",

background:”#000000”,

close:function(data){

//close为回调函数，当被打开的窗口关闭时，会调用这个函数

//data为被关闭窗口回传的数据，json格式

}

})；

url: 打开窗口使用的http url地址,不能为空。

title: 打开窗口使用的标题，可以为空，当标题为空时，默认显示html的标题。

background: 背景颜色。

close: 窗口关闭时的回调函数

## 关闭网页窗口

通过此接口可以关闭当前打开的网页窗口（关闭当前窗口后，默认返回之前的页面）。

workplat.window.close(data);

data：回传给上一个页面的数据，json对象格式（非json字符串），具体返回数据自定义

## 设置标题栏文字

workplat.window.setTitle("标题");

## 设置标题栏颜色

workplat.window.setBackground(“#000000”);

## 设置按钮菜单

workplat.window.setMenu({

color:”#FF0000”,

items:[

{ id:”1”, iconId:”xxx”, icon:”http://xxx.png”, text:”按钮3” },

{ id:”2”, iconId:”yyy”, icon:”http://yyy.png”, text:”按钮4” }

],

success:function(id){

console.log(id);

},

error:function(err){

console.log(err);

}

});

color:菜单文字颜色

id: 菜单选项唯一标识

iconId: 菜单选项内置图标

icon: 菜单选项图标路径，通过网络下载。

text: 菜单选项名称。

success: 当点击菜单选项时调用的函数。

error: 当发生错误时调用的函数

## 设置标题栏状态

workplat.window.setNaviBarStatus(status);

status 0:不显示，1：显示

# 原生界面接口（workplat.ui）

## 打开原生页面

workplat.ui.open({

url:”ui://profile”,

params: {id:11},

success:function(data){

console.log(data);//返回值true或false

},

error:function(err){

console.log(err);

}

});

url: 需要打开的原生界面地址

1、个人资料页面

ui://profile

id:用户ID

2、消息聊天页面

ui://chat

id:聊天对象ID

type:聊天对象类型，0：用户，1：应用，2：组

3、打开应用页面

ui://application

id:应用ID

success: 打开原生界面成功调用

error: 打开原生界面失败调用

# 图片接口(workplat.image)

## Input[“file”]原生支持

支持通过input[“file”]上传图片。

## 图片上传接口

workplat.image.upload({

type: 0,

multiple: true,

max: 5,

compression: true,

quality: 60,

resize: 50,

success:function(resIds){

console.log(resIds);

},

error:function(err){

console.log(err);

}

});

type 0:拍照+选图，1：只拍照，2：只选图，默认为0

multiple 是否支持多选，默认false

compression 是否压缩，默认为true

quality 图片压缩质量，默认50

resize 图片缩放比例，默认50

success 上传成功调用函数

error 上传失败调用函数

## 预览图片接口

workplat.image.preview({

resIds: [“111”,”222”,”333”],

urls:[“http://xxx.png”,”http://yyy.png”,”http://zzz.png”],

success: function(data){

console.log(data);

},

error:function(err){

console.log(err);

}

});

## 图片预览接口

通过给img设置自定义属性实现图片预览。

view: true 图片是否支持预览

viewpath: “http://xxx.png” 图片预览路径

<img view="true" viewpath=”http://xxx.png” src=”http://xxx.png”/>

# 文件接口(workplat.file)

## 文件预览接口

workplat.file.preview({

resId:”111”,

url:”http://xxx/xx.doc”,

success:function(data){

console.log(data);

},

error:function(err){

console.log(err);

}

});

resId: 平台文件资源ID；

url: 文件资源链接地址，可以是平台地址，也可以是外部文件链接地址；

resId与url参数二选一。

# 音频接口(workplat.audio)

## 开始录音接口

音频格式为amr格式，支持最长为60秒的音频录制。

workplat.audio.startRecord({

success: function(){

},

error: function(err){

console.log(err);

}

});

## 停止录音接口

workplat.audio.stopRecord({

success: function(data){

console.log(data.resId); //音频资源ID

console.log(data.duration);//音频时长（秒）

},

error: function(err){

console.log(err);

}

});

## 录音结束事件

workplat.audio.onRecordEnd({

success: function(data){

console.log(data.resId); //音频资源ID

console.log(data.duration);//音频时长（秒）

},

error: function(err){

console.log(err);

}

});

## 播放录音接口

workplat.audio.play({

resId: ”xxx”,

success: function(data){

},

error: function(err){

console.log(err);

}

});

## 停止播放接口

workplat.audio.stop({

resId: “xxx”,

success: function(data){

console.log(data.resId);

},

error: function(err){

console.log(err);

}

});

## 播放停止事件

workplat.audio.onPlayEnd({

success: function(data){

console.log(data.resId);

},

error: function(err){

console.log(err);

}

});

# 视频接口(workplat.video)

## 开始录制视频接口

视频格式为MP4格式，大小为10M以内，支持最长为30秒的视频录制。

workplat.video.startRecord({

success: function(data){

console.log(data.resId);//视频资源ID

console.log(data.coverId);//视频封面ID

console.log(data.duration);//视频时长，单位：秒

},

error: function(err){

console.log(err);

}

});

## 开始播放视频接口

workplat.video.play({

resId: ”xxx”,

success: function(data){

},

error: function(err){

console.log(err);

}

});

## 播放视频停止事件

workplat.video.onPlayEnd({

success: function(data){

console.log(data.resId);

},

error: function(err){

console.log(err);

}

});

# 地理位置接口(workplat.location)

## 获取用户地理位置

workplat.location.getPosition({

success:function(location){

var lng = location.lng //经度

var lat = location.lat //纬度

},

error:function(msg){

var msg = msg;//获取失败信息

}

});

//获取用户当前的地理位置信息

其中success、error为回调函数

loc格式如下：

{

lng:120.591261 //经度信息

lat:31.309939 //纬度信息

}

# 分享组件(workplat.share)

## 内容分享接口

* **接口示例**

workplat.share.shareMessage({

title: ”分享标题”,

content: ”分享内容”,

url: ”http://www.baidu.com/”, //分享链接地址

image: ”http://apps.bdimg.com/icon.png”, //分享图片链接地址

success:function(data){

},

error:function(msg){

var msg = msg;//获取失败信息

}

});

* **参数说明**

| 参数 | 参数类型 | 说明 |
| --- | --- | --- |
| title | String | 分享标题 |
| content | String | 分享内容 |
| image | String | 分享的图片 |
| url | String | 分享链接地址 |