

## To 业务系统开发

### 设备

- 设置横竖屏
- 获取设备信息
  - 获取设备平台信息
- 获取网络信息
- 发送短信

### 文件

- 文件下载
- 打开文件

### 相机相册

- 拍照
- 相册
  - 自定义
- 九宫格图片选择
- 九宫格图片(自定义)

### 页面

- 页面涂鸦标记
- 打开新的子页面窗口

### 二维码

- 二维码扫描
- 生成二维码

### GPS位置信息

- 打开设置界面-位置信息
- 判断位置信息是否开启
- 获取gps位置

### 存储数据

- 存储数据
- 获取数据
- 删除对应key值的数据
- 保存token
- 获取token
- 清除token
- 存储用户信息
- 获取用户信息
- 清除用户信息

### U盘(OTG)

### 打印机

- 连接打印机
- 获取当前连接打印机名称
- 获取所有打印机
- 打开打印机列表页面
- 断开连接打印机
- 打印

### 消息推送(阿里云推送)

- 获取注册id
- 绑定账号
- 解绑账号
- 消息监听
- 绑定TAG
- 解绑标签
- 解绑所有标签
- 添加别名
- 解绑别名

- 删除所有别名
- 判断是否开启了通知的权限
- 请求开通通知权限
- 吐司
  - 显示自定义吐司
  - 显示默认吐司
- 应用最小化
- 监听事件
  - 监听返回键
  - 监听网络
- 应用信息
  - 获取应用名称
  - 获取应用包名
  - 获取版本号
  - APP版本升级
  - 设置开机自启动
  - 关闭开机自启动
- 键盘
  - 显示键盘
  - 隐藏键盘
  - 监听键盘弹出
  - 监听键盘隐藏
- app奔溃日志收集
- 语音播报
- 指纹
  - 指纹解锁是否可用
  - 显示指纹解锁
- MQTT
- PDA扫描
  - 开始扫描
  - 停止扫描
  - 设置接收模式
  - 设置连续扫描
  - 停止连续扫描
  - 获取扫描值
  - 关闭扫描

## 设备

### 设置横竖屏

- orientation:0横屏 1竖屏

```
NativeApi.screenOrientation(orientation)
```

### 获取设备信息

#### 获取设备平台信息

- 返回值:Android, iOS

```
NativeApi.getDevicePlatform()
```

## 获取网络信息

- 返回值:wifi 4g none

```
NativeApi.getNetwork()
```

## 发送短信

- number:电话号码
- message:内容

```
NativeApi.sendSms(number, message, successCallback, errorCallback)
```

## 文件

### 文件下载

- url为文件下载地址

```
NativeApi.downloadFile(url, successCallback, errorCallback)
```

### 打开文件

- filePath 文件路径本地或网络

```
NativeApi.openDocx(filePath, successCallback, errorCallback)
```

## 相机相册

### 拍照

- returnType: 0 返回base64 1 返回路径

```
NativeApi.takePhoto(returnType, successCallback, errorCallback)
```

### 相册

- returnType: 0 返回base64 1 返回路径

```
NativeApi.getPicture(returnType, successCallback, errorCallback)
```

## 自定义

名字	类型	默认值	描述
quality	number	50	图像的保存质量，范围0-100，100是最大值，最高的分辨率，没有任何压缩损失（请注意有关该相机的分辨率信息不可用。）
destinationType	DestinationType	FILE_URI	选择返回值的格式
sourceType	PictureSourceType	CAMERA	获取图片的来源
allowEdit	Boolean	true	允许在选择图片之前进行简单的编辑
encodingType	EncodingType	JPEG	选择图像的返回编码
targetWidth	number		宽度像素用来缩放图像。必须和targetHeight一起使用。宽高比保持不变
targetHeight	number		高度像素用来缩放图像。必须和targetWidth一起使用。宽高比保持不变
mediaType	MediaType	PICTURE	选择media类型。它只适用PictureSourceType是PHOTOLIBRARY或SAVEDPHOTOALBUM
correctOrientation	Boolean		如果是横向拍摄的照片，会自动旋转到正确的方向
saveToPhotoAlbum	Boolean		设备拍照后的图像是否保存的图片库中
popoverOptions	CameraPopoverOptions		仅ios可用，设定在ipad的popover的位置
cameraDirection	Direction	BACK	选择前置摄像头还是后置摄像头

```

var option = {
    quality: 50,
    destinationType: Camera.DestinationType.FILE_URI,
    sourceType: Camera.PictureSourceType.SAVEDPHOTOALBUM,
    encodingType: Camera.EncodingType.JPEG,
    mediaType: Camera.MediaType.PICTURE,
    allowEdit: true,
    correctOrientation: true
}
NativeApi.customPhoto(successCallback, errorCallback, option)

```

### 九宫格图片选择

- isCompress 是否压缩
- maximumImagesCount: 最多选择照片张数

```
var option = {
    isCompress:true,
    maximumImagesCount:6
}
NativeApi.getImgsPicker (imgOption, successCallback, errorCallback)
```

### 九宫格图片(自定义)

配置参数	参数含义
maximumImagesCount	多选限制数量，默认为9
width	设置输出图片的width，默认为自动
height	设置输出图片的height，默认为自动
quality	图片质量 默认80
enablePickOriginal	允许选择原图 默认true

```
var option = {
    maximumImagesCount : 9,
    width : 1920,
    height : 1440,
    quality : 100
}
NativeApi.getCustomImgsPicker(option, successCallback, errorCallback)
```

### 页面

#### 页面涂鸦标记

- filePath:图片路径

```
NativeApi.sign (filePath, successCallback, errorCallback)
```

#### 打开新的子页面窗口

- 仅打开新的页面，无法使用硬件功能

```
NativeApi.openNewUrl(url)
```

- 打开页面，可以使用硬件功能

```
NativeApi.openUrl(url)
```

## 二维码

### 二维码扫描

```
NativeApi.scan(successCallback, errorCallback)
```

### 生成二维码

```
NativeApi.encode(content, successCallback, errorCallback)
```

## GPS位置信息

### 打开设置界面-位置信息

```
NativeApi.locationSetting()
```

### 判断位置信息是否开启

```
NativeApi.isLocationAvailable()
```

### 获取gps位置

```
NativeApi.getPosition(successCallback, errorCallback)
```

- 示例:

```
NativeApi.getPosition((data) => {
  if (data.latitude == undefined) {
    alert("请检查位置信息是否开启");
    return;
  }
  alert("Latitude:" + data.latitude + ";" + data.longitude);
  window.location.href = 'gps.html?lat=' + data.latitude + '&lon=' +
data.longitude;
}, (err) => {
  alert(err)
});
```

## 存储数据

### 存储数据

- 键值对, value为字符串

```
NativeApi.setStorage(key, value, successCallback, errorCallback)
```

### 获取数据

- 根据key值获取

```
NativeApi.getStorage(key, successCallback, errorCallback)
```

## 删除对应key值的数据

```
NativeApi.removeStorage(key, successCallback, errorCallback)
```

## 保存token

```
NativeApi.setToken(token, successCallback, errorCallback)
```

## 获取token

```
NativeApi.getToken(successCallback, errorCallback)
```

## 清除token

```
NativeApi.clearToken(successCallback, errorCallback)
```

## 存储用户信息

```
NativeApi.setUserInfo(userInfo, successCallback, errorCallback)
```

## 获取用户信息

```
NativeApi.getUserInfo(successCallback, errorCallback)
```

## 清除用户信息

```
NativeApi.clearUserInfo(successCallback, errorCallback)
```

## U盘(OTG)

- modeType: 文件处理模式: 1压缩 2压缩和上传
- data: 上传时需要携带的参数 (modeType为2时需要提供)
- url: 上传的服务器地址 (modeType为2时需要提供)
- path: 文件路径集合, 为空时为不指定路径, 不为空时为直接选中指定路径的文件
- fileRule: 文件夹名称集合, 为空时不指定文件夹, 所有文件夹都可以通过长按选中。不为空时选择的文件夹名字需在集合中, 集合外的文件夹长按时选择不了, 只有在集合中的文件夹名称长按时才会被选择。(仅check文件夹)

```
var option={
  modeType:2,
  data:{ "user":"001","password":"pass"},
  path:["A1533/Backup/DATA/disk1/0000/X.NC","A1533/DiagChecker/Setting"],
  fileRule:[ "0003","0005"],
  url:"https://build.ymsl.com.cn/Epwechat/Upload/Upload/uploadZipFile",
}
```

```
NativeApi.otg(option,successCallback, errorCallback)
```

# 打印机

## 连接打印机

- 默认是精臣B50W型号打印机
- option: 打印机名称

```
var option = "B50W-90600508";
NativeApi.connectPrinter(option, successCallback, errorCallback)
```

## 获取当前连接打印机名称

```
NativeApi.getConnectPrinter(successCallback, errorCallback)
```

## 获取所有打印机

```
NativeApi.getAllPrinter(successCallback, errorCallback)
```

## 打开打印机列表页面

```
NativeApi.printerList(successCallback, errorCallback)
```

## 断开连接打印机

```
NativeApi.disconnectPrinter(successCallback, errorCallback)
```

## 打印

- width: 打印页面宽度 默认50
- height: 打印页面高度 默认30,
- type: 1: 文本 2: qrcode 3: barcode 4: 图片 5: 直线 6: 圆 7: 矩形,
- text: 打印内容
- left: 左上角水平位置 (单位毫米)
- top: 左上角垂直位置 (单位毫米)
- height: 绘制的文本框垂直高度 (单位毫米) ,
- width: 绘制的文本框水平宽度 (单位毫米) ,
- size: 文字大小 (单位毫米) , 字号和毫米的转换公式为: 字号 = 毫米 \* 72 / 25.4 比方说 9 号字为 3.175 毫米, 12 号字为 4.233 毫米
- horAlign: 水平对齐方式, 默认 0:left, 1:center
- verAlign: 垂直对齐方式, 默认 0:top, 1:center

```
var option = {
    height: 30,
    width: 50,
    content: [{
        type: 1,
        text: "YMCN    IT资产管理卡",
        left: 0,
        top: 2,
        height: 4,
        width: 50,
        size: 3,
        horAlign: 1,
    }]
}
```



```
}  
NativeApi.print(option, successCallback, errorCallback)
```

## 消息推送(阿里云推送)

### 获取注册id

- 获取阿里云推送注册id判断是否注册成功

```
NativeApi.getRegisterId(successCallback, errorCallback)
```

### 绑定账号

- 绑定后可以按账号推送通知

```
NativeApi.bindAccount(successCallback, errorCallback, account)
```

### 解绑账号

```
NativeApi.unbindAccount (successCallback, errorCallback)
```

### 消息监听

```
NativeApi.messageListener (successCallback, errorCallback)
```

### 绑定TAG

- 阿里云推送绑定标签

```
NativeApi.bindTags(successCallback, errorCallback, target, tags, alias)
```

### 解绑标签

```
NativeApi.unbindTags(successCallback, errorCallback, target, tags, alias)
```

### 解绑所有标签

```
NativeApi.listTags(successCallback, errorCallback)
```

### 添加别名

```
NativeApi.addAlias(successCallback, errorCallback, alias)
```

### 解绑别名

```
NativeApi.removeAlias (successCallback, errorCallback, alias)
```

## 删除所有别名

```
NativeApi.listAliases(successCallback, errorCallback)
```

## 判断是否开启了通知的权限

```
NativeApi.isEnableNotification(successCallback, errorCallback)
```

## 请求开通通知权限

```
NativeApi.requireNotifyPermission (successCallback, errorCallback)
```

## 吐司

### 显示自定义吐司

- option设置吐司的位置、显示时长，颜色等样式

```
var option = {
    message: message,
    duration: "short",
    position: "bottom",
    styling: {
        opacity: 0.75,
        backgroundColor: '#333333',
        textColor: '#efefef',
        textSize: 14,
    }
}
NativeApi.showToastCustom(option,successCallback,errorCallback)
```

### 显示默认吐司

- message:显示消息内容

```
NativeApi.showToast(message)
```

## 应用最小化

```
NativeApi.exitApp()
```

## 监听事件

### 监听返回键

```
NativeApi.listenBackbutton(listener)
```

### 监听网络

- 是否连接wifi或移动网络 state:'online'和 'offline'两种状态

```
NativeApi.listenNetworkState(state, callback)
```

## 应用信息

### 获取应用名称

```
NativeApi.getAppName(callback)
```

### 获取应用包名

```
NativeApi.getPackageName(callback)
```

### 获取版本号

```
NativeApi.getVersionNumber(callback)
```

### APP版本升级

- checkOnly: false 有新版本直接升级，无新版本不好弹出已是最新版本的提示

```
var option = {  
    checkOnly: false  
}  
NativeApi.checkUpdate(option, successCallback, errorCallback)
```

### 设置开机自启动

```
NativeApi.enableAutoStart()
```

### 关闭开机自启动

```
NativeApi.disableAutoStart()
```

## 键盘

### 显示键盘

```
NativeApi.showKeyboard()
```

### 隐藏键盘

```
NativeApi.hideKeyboard()
```

### 监听键盘弹出

```
NativeApi.listenKeyboardShow(listener)
```

### 监听键盘隐藏

```
NativeApi.listenKeyboardHide(listener)
```

## app崩溃日志收集

- appName:app名称

```
NativeApi.initBugly(appName, successCallback, errorCallback)
```

## 语音播报

```
var option = {  
    text: 'hello, world!',  
    locale: 'en-GB',  
    rate: 0.75  
}  
NativeApi.speak (option, successCallback, errorCallback)
```

## 指纹

### 指纹解锁是否可用

```
NativeApi.fingerprintIsAvailable(successCallback, errorCallback)
```

### 显示指纹解锁

```
var option ={  
    title:"指纹验证进入",  
    disableBackup:true,  
    description: "请触摸指纹验证"  
}  
NativeApi.showFingerprint(option, successCallback, errorCallback)
```

## MQTT

## PDA扫描

### 开始扫描

```
NativeApi.pdaStartDecode(successCallback, errorCallback)
```

### 停止扫描

```
NativeApi.pdaStopDecode(successCallback, errorCallback)
```

### 设置接收模式

- 设置接收模式0 广播 1 input

```
var option = 1  
NativeApi.pdaSetOutScanMode(option,successCallback, errorCallback)
```

### 设置连续扫描

```
NativeApi.pdaStartContinueMode(successCallback, errorCallback)
```

### 停止连续扫描

```
NativeApi.pdaStopContinueMode(successCallback, errorCallback)
```

### 获取扫描值

```
NativeApi.pdaScanResult(successCallback, errorCallback)
```

### 关闭扫描

```
NativeApi.pdaCloseScanner(successCallback, errorCallback)
```