MUI + 个推实现消息推送

步城曦整理提供

素材来源:[www.getui.com](http://www.getui.com" \o "个推官网)

整理提供消息推送实现接口和基本实现思路

T:

本文档仅适用于安卓

小米手机未通过测试

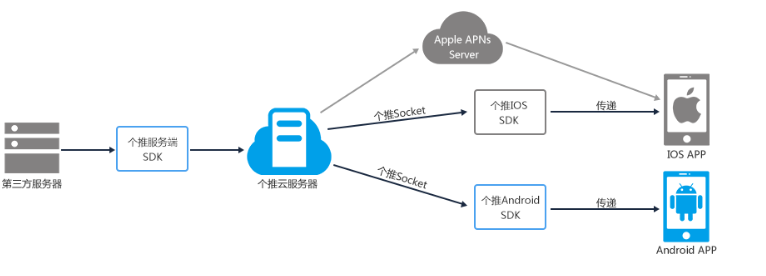
适用于包括但不限于VIVO/OPPO/华为/三星

Iphone端请参见:<http://docs.getui.com/getui/start/ios/>

1.名词解释:

* + 1. clientId:用户标识,用于识别客户端身份,单用户唯一,简称Cid;
    2. AppId:是与个推服务器通信的标识之一,每个app唯一;
    3. AppKey:是与个推服务器通信的标识之一;
    4. MasterSecret:API鉴权码,调用推送时使用;
    5. 标签:用户的一种属性,在进行消息推送的时候,一个标签可当一用户组进行识别推送

1. 整体架构



1. 实现简单消息推送
   1. 创建个推开发者账号,请访问:[https://dev.getui.com/dev/#/login](https://dev.getui.com/dev/" \l "/login)
   2. 下载Demo应用
      1. 创建账号并登陆个推开发者中心,可在列表看到自动生成的[个推APP Demo],点击[下载应用],并安装到手机;
   3. 推送测试
      1. 打开下载的[个推测试]应用,点击获取Cid,如果Cid!=null即成功
      2. 点击通知栏测试,在个推开发者中心点击创建推送,如果手机收到通知即通过简单消息推送
2. 利用服务端进行推送测试[测试过程需联网]
   1. 安装个推SDK
3. Maven依赖(springboot项目可能引发jar文件冲突,建议下载SDK)

<dependency>

<groupId>com.gexin.platform</groupId>

<artifactId>gexin-rp-sdk-http</artifactId>

<version>4.1.0.0</version>

</dependency>

增加一个repository到pom.xml (springboot项目可能引发jar文件冲突,建议下载SDK)

<repositories>

<repository>

<id>getui-nexus</id>

<url>http://mvn.gt.igexin.com/nexus/content/repositories/releases/</url>

</repository>

</repositories>

②下载SDK手动导入jar文件,下载地址:

<http://www.getui.com/download/docs/getui/server/GETUI_JAVA_SDK_4.1.0.0.zip>

5.1服务器代码测试(STS测试通过,运行环境:windows7)

* 1. GitHub下载地址
     1. <https://github.com/buchengxi/getuiTestDemo>
     2. 内含:API基类,单用户/多用户/指定用户/群组/标签组推送
     3. Iphone端测试请参见:<http://docs.getui.com/getui/start/ios/>

1. Hbuilder + 个推
   1. 打开工具选择项目的mainsoft.json文件
   2. 点击SDK配置,拉到plus.share-分享
   3. 勾选个推根据个推开发者测试Demo依次填写AppId,AppKey等
   4. 完成配置
2. Hbuilder + 个推指导
   1. 首先我们要用到mui的push (参见MUI技术文档push)
      1. 手机的应用推送权限记得不要关闭,省点模式,游戏模式尽量关闭
   2. 了解push
      1. 用户在触发事件的时候在回调中获取PushMessage对象,获取其title和content属性的值;
      2. 在用户发送消息(推送)的时候都会进入系统的消息中心,但是要求符合格式,不符合格式将不会触发事件;
   3. 配置平台信息
      1. 监听收到的推送消息事件和点击事件
         1. Receive事件监听:客户端在操作,监听到了会触发
         2. Click事件监听:用户点击触发
         3. 代码如下:

document.addEventListener( "plusready", function(){

message = document.getElementById("message");

// 监听点击消息事件

plus.push.addEventListener( "click", function( msg ) {

// 判断是从本地创建还是离线推送的消息

switch( msg.payload ) {

case "LocalMSG":

outSet( "点击本地创建消息启动：" );

break;

default:

outSet( "点击离线推送消息启动：");

break;

}

// 提示点击的内容

plus.ui.alert( msg.content );

// 处理其它数据

logoutPushMsg( msg );

}, false );

// 监听在线消息事件

plus.push.addEventListener( "receive", function( msg ) {

if ( msg.aps ) { // Apple APNS message

outSet( "接收到在线APNS消息：" );

} else {

outSet( "接收到在线透传消息：" );

}

logoutPushMsg( msg );

}, false );

}, false );

* + - 1. 获取客户端推送标识

function getPushInfo(){

var info = plus.push.getClientInfo();

outSet( "获取客户端推送标识信息：" );

outLine( "token: "+info.token );

outLine( "clientid: "+info.clientid );

outLine( "appid: "+info.appid );

outLine( "appkey: "+info.appkey );

}

更多相关push请参见MUI技术文档push块

2019-01-06 00:26