

个推 SDK 文档

getui.com

October 2016

目录

快速入门	2
第一步：创建个推开发者账号	2
第二步：获取访问凭证	3
第三步：安装 Demo 应用到手机	4
第四步：安装个推 C/C++ SDK	5
第五步：使用 C/C++ SDK 发送消息	5

快速入门

第一步：创建个推开发者账号

访问个推开发者平台 <http://dev.getui.com>，注册个推开发者帐号，如下图所示：

请登录

我要注册

 请输入用户名

 请输入登录密码

☐ 记住我

其他账号登录: **CSDN**

登 录

[忘记密码？](#)

* 用户名：

* 帐户类型：☐ 个人 ☒ 公司

* 公司全称：

* 联系人：

* 电子邮箱：

* 手机号：

* 密码：

* 确认密码：

* 验证码： 

推码/优惠码：

第二步：获取访问凭证

创建账号成功并登录后，可以在应用列表中看到自动生成的【个推 App Demo】，然后点击【创建推送】，如下图所示：



然后点击左侧菜单栏【配置管理】--->【应用配置】，就能看到关于此 app 的配置信息，这些配置信息就是第五步使用 SDK 发送消息需要的 AppID, AppKey, MasterSecret，如下图所示：



第三步：安装 Demo 应用到手机

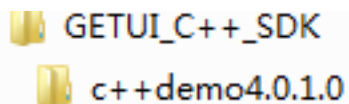
在应用【个推 App Demo】点击【下载应用】，下载后安装到手机上，用来接受推送消息：



第四步：安装个推 C/C++ SDK

下载 demo 程序

下载个推 C/C++ Demo，下载地址为：http://www.getui.com/download/docs/server/GETUI_C++_SDK.zip，目录结构如下图所示：



第五步：使用 C/C++ SDK 发送消息

安装调试好 C++ 环境后，下面是以 app 推送接口代码为例来发送消息：

```
#include "IGtPush.h"
#include <iostream>
#include <stdio.h>
#include <stdlib.h>
```

```
#include <string.h>
#include <string>
#ifdef WIN32
#include <windows.h>
#endif

using namespace std;
static void printResult(IPushResult &result);
static void TransmissionTemplateDemo(TransmissionTemplate* templ)

// http的域名
//static char *host ="http://sdk.open.api.igexin.com/apiex.htm";

//https的域名
static char *host ="https://api.getui.com/apiex.htm";

//定义常量, appId、appKey、masterSecret 采用本文档 “第二步 获取访问凭证” 中获得的应用配置
static char *appId = "";
static char *appKey = "";
static char *masterSecret = "";

void toapptest();

void toapptest() {

    // 定义"AppMessage"类型消息对象, 设置消息内容模板、发送的目标App列表、
    // 是否支持离线发送、以及离线消息有效期(单位毫秒)
    Message msg = {0};
    msg.isOffline = 0;//是否离线下发
    msg.offlineExpireTime = 1000*3600*2;//离线下发有效期 毫秒
    msg.pushNetWorkType = 0;//0不限 1wifi 2:4G/3G/2G

    AppMessage appMsg = {0};
    appMsg.msg = msg;
    appMsg.speed = 10000;//定速
    char* appList[] = {appId};
    appMsg.appIdList = appList;
    appMsg.appIdListSize = 1;

    IPushResult result = {0};
    //透传模板, 设置透传内容和透传形式
    TransmissionTemplate tpl= {0};
    TransmissionTemplateDemo(& tpl);
    result = pushMessageToApp(appKey, &appMsg, &tpl, Transmission);
    //打印结果
```

```
    printResult(result);
}

/* 透传消息模板 */
void TransmissionTemplateDemo(TransmissionTemplate* templ)
{
    templ->t.appId = appId;
    templ->t.appKey = appKey;
    //应用启动类型，1：强制应用启动 2：等待应用启动
    templ->transmissionType = 1;
    //透传内容
    templ->transmissionContent = "您输入的透传内容";

    //templ->t.duration_start="2015-07-10 18:00:00";
    //templ->t.duration_end="2015-07-10 19:00:00";
}

static void printResult(IPushResult &result) {
    cout << "print result:-----" << endl;
    for (int i = 0; i < result.size; i++) {
        cout << result.entry[i].key << ": " << result.entry[i].value << endl;
    }
    cout << "print end:-----" << endl;
}

int main() {
    // 注意：接口传入字符必须为UTF-8编码，因ASCII字符UTF-8编码后与原先一样，
    // 所以可以不编码，但中文等非ASCII字符必须编码
    // 如果返回的类似错误"post http data failed, code=6"
    // 错误码可百度CURL返回的错误码是什么意思，如http:
    // www.cnblogs.com/wainiwann/p/3492939.html
    // "编码"两个字为固定写法，不需要做转换
    Result r = pushInit(host, appKey, masterSecret, "编码");
    if(r!=SUCCESS) {
        printf("pushInit for app failed: ret=%d\n", r);
        return -1;
    }
    toapptest();

    return 0;
}
```

调试运行。如果收到下面一样的推送，恭喜你 ~ 推送成功了。

