

基于ionic.io平台的ionic消息推送功能实现

1 [One Time]安装ionic

- 1 安装node.js <https://nodejs.org/en/> 安装node.js的主要作用就是它提供的npm
- 2 安装cordova和ionic `npm install -g cordova ionic`

2 新建ionic项目

- 1 新建项目，例如testApp `ionic start testApp tabs` 这个tabs只是ionic提供的一个模板
- 2 [One Time]注册一个ionic平台的账号
- 3 切换到testApp目录 `cd testApp`
- 4 添加推送插件phonegap-plugin-push `ionic plugin add phonegap-plugin-push`
- 5 连接到ionic io平台 `ionic io init`
- 6 设置开发模式为false `ionic config set dev_push false`
- 7 在app.js中添加一些代码，获取device token，只有app在设备上运行时才能获取到device token，获取到值后，想办法保存下来（例如放在localStorage中，然后想办法输出到页面），发送消息时候需要用到。

```
.run(function($ionicPlatform) {
  $ionicPlatform.ready(function() {
    var push = new Ionic.Push({
      "debug": true
    });
    push.register(function(token) {
      console.log("Device token:"+token.token);
      window.localStorage['device_token'] = token.token;
    });
  });
})
```

3 iOS平台
<http://docs.ionic.io/docs/push-ios-setup>

- 为项目添加iOS平台 `ionic platform add ios` 这个命令会在项目的platforms目录下边添加一个ios目录
- [One Time]在Apple开发者中心添加一个Certificate Certificates, Identifiers&Profiles>iOS Apps>Certificates>">iOS Development>Create a Certificate Signing Request>Upload CSR file>download
- [One Time]在Apple开发者中心添加一个Device Certificates, Identifiers&Profiles>iOS Apps>Devices>">Name&UDID <http://jingyan.baidu.com/article/0eb457e506c8d503f1a905b0.html>
- 在Apple开发者中心添加一个App ID Certificates, Identifiers&Profiles>iOS Apps>Identifiers>">Name&Bundle ID, check Push Notifications
- 在Apple开发者中心配置刚刚生成的App ID Edit>为刚刚的新建的App ID生成一个Push SSL Certificate>upload CSR file>download cer file>打开之后导出.p12文件
- 在Apple开发者中心创建一个Provision File [指定设备可以用来测试指定程序] Certificates, Identifiers&Profiles>iOS Apps>Provisioning Profiles>">iOS App Development>App ID>iOS Development Certificate>Device>name>download
- 配置项目中的config.xml文件 把<widget id="..."中id的值换成刚刚App ID中设置的bundle id
- 连接ionic.io平台 `ionic push --ios-dev-cert` 把刚刚的.p12文件拖入terminal然后回车
- 建立ios项目 `ionic build ios`
- 用Xcode打开/platforms/ios/中的xcodeproj项目文件
- [One Time]用Apple 开发者账号登陆Xcode Xcode>Preferences>Accounts
- [?]在Xcode中修改testApp项目的配置文件，把Apple ID设置为刚刚登陆的账号。
- 连接上开发设备(刚在Apple开发者中心添加的Device) 在Window/Devices中配置好Provision File
- 在运行的地方选中开发设备
- 点击运行，将App安装到设备
- 记下device token

4 Android平台
<http://docs.ionic.io/docs/push-android-setup>

- 1 为项目添加Android平台 `ionic platform add android`
- 2 [One Time]安装JDK(OS自带JDK无法启动Android Studio) <http://www.oracle.com/technetwork/java/javase/downloads/index.html>
- 3 [One Time]安装Android Studio <https://developer.android.com/sdk/index.html>
- 4 登陆谷歌开发者控制台，如果界面和链接教程中的不一样，就点右上角的竖着的三个"."，把界面调整为旧版
- 5 在谷歌开发者控制台新建一个项目/专案 需要记住它的project number
- 6 在谷歌开发者控制台启动GCM服务
- 7 在谷歌开发者控制台新建一个API Key 需要记住这个api key
- 8 发送API Key到ionic.io `ionic push --google-api-key your-google-api-key`
- 9 连接ionic.io `ionic config set gcm_key <your-gcm-project-number>`
- 10 插上安卓设备，开启开发者模式，允许USB调试
- 11 在安卓设备上运行 `ionic run --device android`
- 12 记下device token

5 推送消息

- 1 登录 <https://apps.ionic.io/>
- 2 找到并点击刚刚新建的testApp
- 3 找到 "Want to send a one-time notification?" 并点击 "one-time"
- 4 输入消息title, message以及设备token(前边获得的device token)
- 5 点击 "Send"，即可在设备上边收到推送的消息。
- 6 在send旁边一个button，可以显示对应的curl命令