

听力宝 – iOS听力测试SDK集成文档

1:SDK介绍

iOS听力测试SDK是一个动态framework，TLBHearingTestSDK.framework，支持iOS8以上，提供3种测试逻辑：自动测试／手动测试／标准测试，目前支持iphone一代&二代耳机，并可获得测试结果。

2:获取Appkey

集成SDK之前，您首先需要通过商务或者客服获取Appkey，需要提供app包名、名称、平台、类型等信息

特别提醒： 建议提供真实信息，避免虚假信息带来的使用问题

3:下载导入SDK文件集成 在工程目录结构中，添加听力测试框架，在选项 TARGETS-->Build Phases-->Link Binary With Libraries--> Add Other，找到下载的TLBHearingTestSDK.framework文件并选择确认；

4:SDK功能集成

4.1 SDK 初始化 在工程的AppDelegate.m 文件中如下函数，并添加相关代码如下：

```
- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions {  
    // Override point for customization after application launch.  
  
    [TLBHearingTest registerAppKey:@"8a2b0565571d2024015727edcf730000"];  
  
    return YES;  
}
```

4.2 iOS9中ATS配置 由于iOS9引入了App Transport Security (ATS)特性，要求App访问的网络使用HTTPS协议，如果不做特殊设置，http请求会失败，所以需要开发者在工程中增加设置以便可以发送http请求，在info plist中增加字段：

```
<key>NSAppTransportSecurity</key> <dict>  
<key>NSAllowsArbitraryLoads</key><true/> </dict>
```

配置完后如图所示：



4.3初始化测试类

```
-(__nullable instancetype )initWithTestType:(TLBHearingTestType)type;
```

目前的测试类型如下：

```
typedef enum
{
    AutoTest, //自动测试
    ManualTest, //手动测试
    StandardTest //标准测试
}TLBHearingTestType;
```

4.4设置耳机参数

```
-(void) setEarPlugModel:(nonnull NSString*) model completionHandler: (nonnull void (^)( NSError * __nullable error))completionHandler; //设置插入的耳机类型,必须设置
```

目前支持：一代耳机，传入参数@"0"；二代耳机，传入参数!"1"

特别提醒：该接口是个异步接口，设置之后会检测较准信息，completionHandler回调后才结束

4.5设置噪音参数

```
-(void) setNosiseValue:(nonnull NSString*) dbValue; //设置当前的环境噪声大小,没有测试的话可设置为0
```

环境噪声影响测试结果，设置当前的环境噪声大小,没有测试的话可设置为0

4.6设置代理

测试过程中的回调信息

```
//=====
@protocol TLBHearingTestDelegate <NSObject>

@required
@optional
//测试完成
-(void) hearingTest:(nonnull TLBHearingTest*)test completeWithLeftResult:(nonnull NSDictionary*)
    result right:(nonnull NSDictionary*)right;
//单耳完成
-(void) hearingTest:(nonnull TLBHearingTest*)test oneEarCompleteWithResult:(nonnull NSDictionary*)
    result isLeft: (BOOL)isLeft;
//将要测试某hz的某db
-(void) hearingTest:(nonnull TLBHearingTest*)test willTestAtHz:(nonnull NSString*) hz db:(nonnull
    NSString*)db volume:(float)volume isLeft: (BOOL)isLeft;
//将要选中某hz的某db为阈值点
-(void) hearingTest:(nonnull TLBHearingTest*)test willSelectedAtHz:(nonnull NSString*) hz db:
    (nonnull NSString*)db isLeft: (BOOL)isLeft;
@end
```

4.7开始测试

```
- (void)startTest:(BOOL) isleft;//单耳开始测试
```

4.8暂停当前测试

```
- (void)pauseTest;//暂停当前测试
```

4.9继续测试

```
- (void)continueTest;//继续测试
```

4.10获取测试左耳或右耳

```
- (BOOL)isTestLeft;////测试左耳或右耳
```

4.11获取测试进度

```
-(float)getTestProgress;//获取测试进度
```

4.12测听者听到了声音

```
- (void)listened;//听到了声音
```

4.13用户手动测试时调用

```
//=====
-(void) playAtHz:(nonnull NSString*)hz db:(nonnull NSString*)db;//用户手动测试
//=====
```

4.14获取测试结果：结果为一个字典，key为hz，value是db值，例如@"1000hz":@"45"

```
-(nullable NSDictionary*)getTestResult:(BOOL)isleft;
```

4.15依据测试结果计算听力损失db,需传入测试者的年龄和性别gender，男性为@"1"，女性为@"0"

```
+(NSInteger)computeEarTestResult:(nonnull NSDictionary*)dataDictory age:(NSInteger) age gender:(nonnull NSString*)gender;//依据测试结果计算听力损失db,gender，男性为@"1"
```

4.16依据听力损失获取听力结果描述，包括正常、轻度听力损失、中度听力损失、中重度听力损失、重度听力损失、极重度听力损失

```
+(nullable NSString *)computeResultString:(NSInteger)result;//依据听力损失获取听力结果描述
```