

EEGSmart Umind Sdk (Android) 使用文档

导入jar包

1.将eegsmartumind.jar拷贝到你的工程libs目录下。

2.在build.gradle中添加依赖:

```
compile files('libs/eegsmartumind.jar')
```

3.在你的工程的AndroidManifest.xml文件中添加权限:

```
<uses-permission android:name="android.permission.BLUETOOTH"/>
```

```
<uses-permission android:name="android.permission.BLUETOOTH_ADMIN"/>
```

4.注册服务:

```
<service android:name="com.eegsmart.mylibrary.server.BluetoothUtilService">
</service>
```

完成上面步骤,就可以使用jar包中的方法、接口了。

初始化

在要使用蓝牙搜索的Activity中实现IBluetoothUtilsListener接口,在onCreate方法里面进行相关初始化。

```
//混合方式启动服务
intent btUtilService = new Intent(MainActivity.this,BluetoothUtilService.class);
startService(btUtilService);
bindService(intent, mBTServiceConnection, BIND_AUTO_CREATE);
//初始化蓝牙工具类
mBluetoothUtils = BluetoothUtils.getInstance();
mBluetoothUtils.init(MainActivity.this, this);
```

其中mBTServiceConnection 为:

```
private ServiceConnection mBTServiceConnection = new ServiceConnection() {
    @Override
    public void onServiceConnected(ComponentName componentName, IBinder service) {
        // 服务绑定成功,获取到BluetoothUtilService
        mBTServiceUtil = ((BluetoothUtilService.BTBinder) service).getService();// 注意括号
        mBTServiceUtil.setHandler(UIHandler);//设置接收Umind连接数据的handler
        //这里的BluetoothUtilService最好能够在Application (定义get和set BTServiceUtil () 方法) 中
        //保存下来,方便其他Activity页面使用
        (Your Application).getInstance().setBTServiceUtil(mBTServiceUtil);
    }

    @Override
    public void onServiceDisconnected(ComponentName componentName) {
        // 绑定失败
        mBTServiceUtil = null;
    }
};
```

其中UIHandler为UI线程,用于接收Umind发送过来的数据。

搜索蓝牙

调用searchBluetooth()方法开始搜索:

```
mBluetoothUtils.searchBluetooth();
```

开始搜索时会调用IBluetoothUtilsListener接口里面的onSearchStart()方法:

```
public void onSearchStart() {
    //开始搜索蓝牙设备
}
```

搜索完成时会调用IBluetoothUtilsListener接口里面的onSearchStart()方法:

```
public void onSearchFinish(ArrayList<BluetoothDevice> devicesList) {
    //蓝牙设备搜索完成,devicesList设备list
}
```

蓝牙连接

在设备列表的点击事件中调用下面代码，实现与设备与umind的连接。

```
BluetoothDevice device = mBluetoothAdapter.getRemoteDevice(bluetoothDevice.getAddress());
mBTServiceUtil.connect(device);
```

UIHandler 接收Umind数据

```
private Handler UIHandler = new Handler() {
    @Override
    public void handleMessage(Message msg) {
        switch (msg.what) {
            case BTConstants.MESSAGE_STATE_CHANGE:
                switch (msg.arg1) {
                    case BTConstants.BLUETOOTH_STATE_CONNECTING:
                        //正在连接umind
                        break;
                    case BTConstants.BLUETOOTH_STATE_CONNECTED:
                        //umind连接成功
                        break;
                }
            case BTConstants.MESSAGE_TOAST:
                //umind连接失败的信息msg.getData().getString(BTConstants.KEY_TOAST)
                break;
            case BTConstants.MESSAGE_UPDATE_EEG_SIGNAL_QUALITY:
                //信号质量noise值msg.obj 0 - 200 0代表质量好
                break;
            case BTConstants.MESSAGE_EEG_ATTENTION:
                //专注度 ATTENTION msg.obj 0-100
                break;
            case BTConstants.MESSAGE_EEG_MEDITATION:
                //放松度 MEDITATION msg.obj 0-100
                break;
            case BTConstants.MESSAGE_ASCII_EEG_POWER:
                //EEG_POWER 包含脑波的八个纬度 msg.obj为：分隔的String
                //八个纬度跟String字符串的顺序对应为：
                //delta, theta, low_alpha, height_alpha,
                //low_belta, high_belat, low_gamma, heigh_gamma
                break;
            case BTConstants.MESSAGE_PIECE_ORIGIN_INT_RAW_DATA:
                //raw_data int[]数组 ints = (int[]) msg.obj;
                break;
            default:
                break;
        }
    }
};
```

当想在其他Activity接收到Umind数据时，可在需要的的Activity中定义一个UIHandler，然后将该UIhandler传递给mBTServiceUtil即可。

```
// 获取BluetoothUtilService
mBTServiceUtil = (Your Application).getInstance().getBTServiceUtil();
mBTServiceUtil.setHandler(UIHandler);//设置接收Umind连接数据的handler
```

Umind数据传输

脑电数据的传输，通过蓝牙将数据发送出去，按 10 包为一个整体，90ms的发送间隔。第 1 包数据包含信号质量、脑电波波段的电流、专注度 冥想度及原始波值。

信号质量noise: 0-200 noise值越小，信号强度越高，0表示信号强度高。

专注度attention: 0-100 值越高表示越专注。

放松度meditation: 0-100 值越高表示越放松。

原始脑波值：最原始的脑波数据。

Umind数据传输开关

打开传输数据(连接成功之后才能开启)：

```
mBTServiceUtil.sendBeginDataTransferCommand();
```

关闭传输数据：

```
mBTServiceUtil.sendStopCommand();
```

停止服务与解除广播

在onDestroy()里面添加以下代码:

```
protected void onDestroy() {  
    super.onDestroy();  
    mBluetoothUtils.unregisterReceiver();  
    if (btUtilService != null) {  
        stopService(btUtilService);  
    }  
}
```