

Body fat scale SDK instructions

Version	Update Time	Author	Update information
1.0	2017/2/15	Suzy	Init version
1.0.1	2017/7/31	Suzy	Fix some data conversion bugs
1.0.2	2017/9/11	Suzy	The type in the instruction is set according to the type of device actually connected.
1.0.3	2017/11/16	Suzy	Code optimization
1.0.4	2018/7/12	Suzy	Compatible with broadcast scales
1.0.5	2018/1/15	Suzy	Compatible with new protocols
1.0.6	2018/2/25	Suzy	Function optimization
1.0.7	2018/2/28	Suzy	Package decimal point class
1.0.8	2018/3/14	Suzy	Fix bugs in body age errors
1.0.9	2018/4/9	Suzy	Compatible with BM09 protocol
1.1.1	2018/5/10	Suzy	Compatible with kg/lb indexing protocol
1.1.2	2018/5/11	Suzy	Solve the bug that the st unit data is inconsistent with the scale display
1.1.3	2018/6/22	Suzy	Solve bugs that are sometimes not available for decimal point
1.1.4	2018/7/7	Suzy	Compatible with BM15 protocol
1.1.5	2018/7/20	Suzy	Compatible algorithm ID protocol

目录

Body fat scale SDK instructions-----1

A. Precautions : ----- 3

B. Version update information : ----- 4

C. Conditions of Use : ----- 5

D. Start using : ----- 5

E. Discovery device:----- 5

F. Device Type: ----- 6

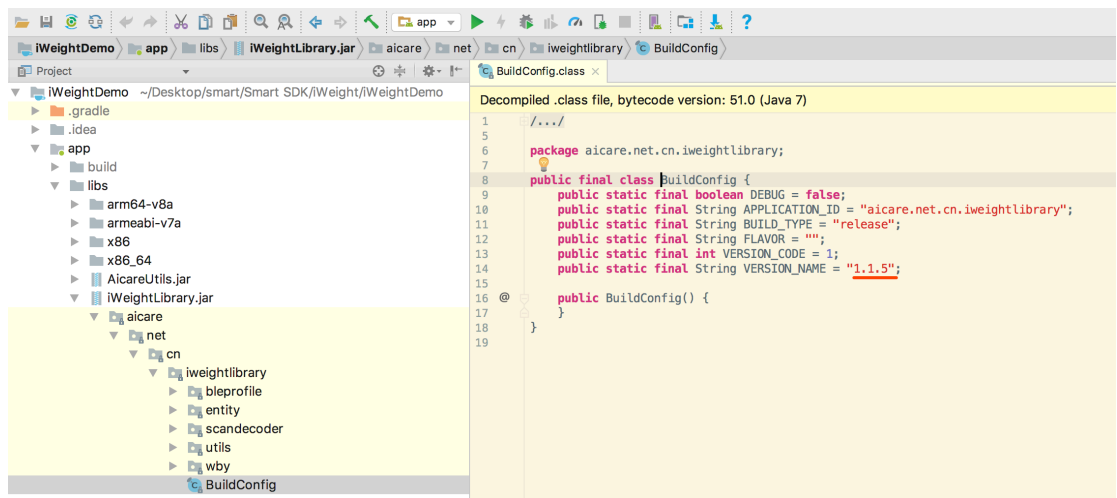
G. Flow Description:----- 7

H. Communicate with the device:----- 7

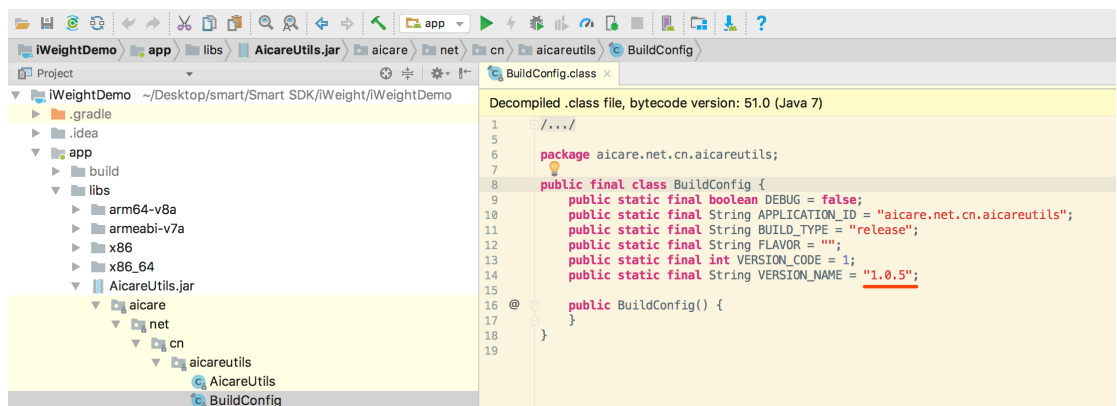
I. Object description (aicare.net.cn.iweightlibrary.entity) : ----- 9

A. Precautions :

1. The latest version of iWeightLibrary.jar is 1.1.5. (2018/8/16)



2. The latest version of AicareUtils.jar is 1.0.5. (2018/8/16)



3. If targetSdkVersion>22 needs to dynamically obtain location permissions (some phones need to turn on GPS).
4. Users/units can be synchronized only after receiving STATE_INDICATION_SUCCESS.
5. Please synchronize the user after receiving STATE_INDICATION_SUCCESS, otherwise the body fat data will not be obtained.
6. The unit only supports 4 kinds (kg, lb, st, jin) at most. Please refer to the scale for specific units.
7. The weight data returned by the scale end is not converted. To convert, use the `AicareBleConfig.getWeight(double weight, byte unit, DecimalFormat decimalInfo)` method.

B. Version update information :

Vesion1.0:

1. Init version.

Version1.0.1:

1. Increase the function of receiving the disconnect command to disconnect (fit BM05);
2. Fixed a bug in some data conversion errors.

Version1.0.2:

1. The type in the instruction is set according to the type of device actually connected.

Version1.0.3:

1. Data transfer mode from interface to broadcast;
2. Modify the enum of AicareBleConfig;
3. Modify HashMap->SparseArray.

Version1.0.4:

1. Compatible with broadcast scales.

Version1.0.5:

1. Compatible with new protocols.

Version1.0.6:

1. Connection status added: connection and connection timeout (timeout defaults to 10s);
2. Scan is modified to rescan every 1 minute.

Version1.0.7:

1. Package decimal point class.

Version1.0.8:

1. Fix bugs in body age errors.

Version1.0.9:

1. Compatible with BM09 protocol.

Version1.1.1:

1. Compatible with kg/lb indexing protocol.

Version1.1.2:

1. Solve the bug that the st unit data is inconsistent with the scale display.

Version1.1.3:

1. Solve bugs that are sometimes not available for decimal point.

Version1.1.4:

1. Compatible with BM15 protocol.

Version1.1.5:

1. Compatible algorithm ID protocol.

C. Conditions of Use :

1. The lowest version of android4.3(API 18).
2. The Bluetooth version used by the device requires 4.0 and above.

D. Start using :

1. Set minSdkVersion: minSdkVersion needs to be greater than or equal to 18.
2. Add related permissions:

```
<uses-permission android:name="android.permission.BLUETOOTH"/>
<uses-permission android:name="android.permission.BLUETOOTH_ADMIN"/>
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
```

3. Registration service:

```
<service android:name="aicare.net.cn.iweightlibrary.wby.WBYService"/>
```

4. The Activity that needs to get the device data inherits the BleProfileServiceReadyActivity and overrides the related methods.

E. Discovery device:

1. Determine if the device supports BLE:

Call ensureBLESupport() in BleProfileServiceReadyActivity. If true, the device supports BLE.

2. Bluetooth status:

The Bluetooth switch will trigger `bluetoothStateChanged(int state)` in `BleProfileServiceReadyActivity`.

- i. Bluetooth is turned on: `BluetoothAdapter.STATE_ON`
- ii. Bluetooth is off: `BluetoothAdapter.STATE_OFF`
- iii. Bluetooth is turning on: `BluetoothAdapter.STATE_TURNING_ON`
- iv. Bluetooth is turning off: `BluetoothAdapter.STATE_TURNING_OFF`

3. Start scanning:

If the device supports BLE, call `startScan()` in `BleProfileServiceReadyActivity` to start scanning.

4. Stop scanning:

You can stop the scan by calling `stopScan()` in `BleProfileServiceReadyActivity` when you don't need to scan.

F. Device Type:

The device is mainly divided into the following two categories. The device type can be obtained according to `BroadData.getDeviceType()`.

1. Broadcast scale (weight data is obtained by parsing Bluetooth broadcast data, and Bluetooth broadcast data can be obtained according to `BroadData.getSpecificData()`):
 - 1.1 Weight broadcast scale (`deviceType=0`), weight broadcast scale with temperature (`deviceType=1`):

Call `AicarConfig.getWeightData(byte[] b)` to get the `WeightData`.
 - 1.2 BM09 broadcast scale (`deviceType=9`)

Call `AicareConfig.getBm09Data(String address, byte[] specialData)` to get the `BM09Data`.
 - 1.3 BM15 broadcast scale (`deviceType=15`)

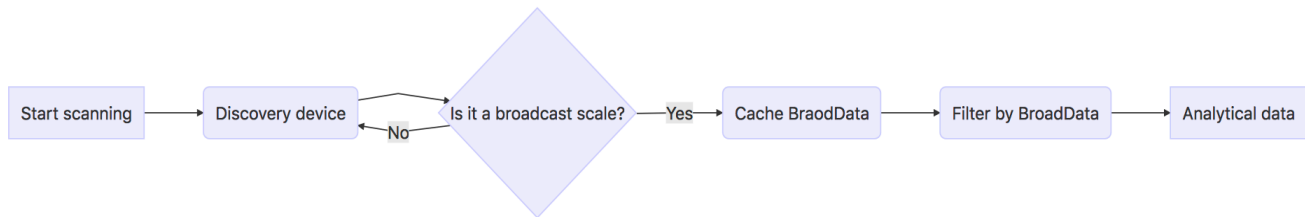
Call `AicareConfig.getBm15Data(String address, byte[] specialData)` to get the `BM15Data`.
2. Connection scale:
 - 2.1 Calculate body fat data on the scale

The body fat data is returned by `BleProfileServiceReadyActivity.onGetFatDat(boolean isHistory, BodyFatData bodyFataData)` on the premise of successful synchronization of the user and impedance measurement.
 - 2.2 Scale-end return algorithm ID (calculates body fat data from app)

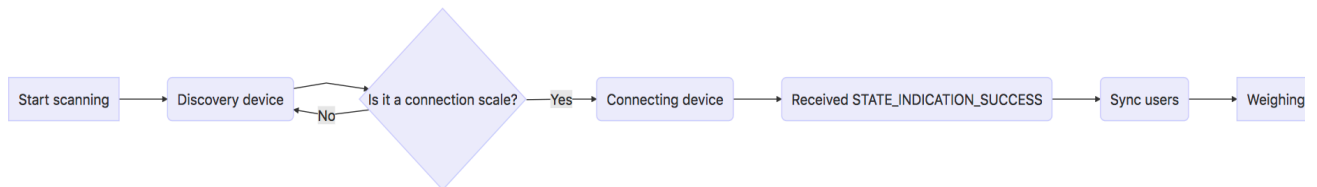
On the premise that the impedance measurement is successful, the algorithm sequence information is returned by `BleProfileServiceReadyActivity.onGetAlgorithmInfo(AlgorithmInfo algorithmInfo)`.

G. Flow Description:

1. Broadcast scale:



2. Connection scale:



H. Communicate with the device:

1. Connect the device:

- a) When scanned to the device, `getAicareDevice(BroadData broadData)` in `BleProfileServiceReadyActivity` is triggered. (If you have multiple devices, you need to handle it yourself)
- b) After scanning to the device that needs to be connected, call `startConnect (String address)` in `BleProfileServiceReadyActivity` to connect to the device.

2. Connection Status (`#onStateChanged(String deviceAddress, int state)`)

- a) Connection succeeded: `BleProfileService.STATE_CONNECTED`.
- b) Disconnect: `BleProfileService.STATE_DISCONNECTED`.
- c) Discovery service: `BleProfileService.STATE_SERVICES_DISCOVERED`.
- d) Indication success: `BleProfileService.STATE_INDICATION_SUCCESS`.
- e) Connection timed out: `BleProfileService.STATE_TIME_OUT`.
- f) Connecting: `BleProfileService.STATE_CONNECTING`.

3. Connection error (`#onError(String msg, int errorCode)`)
4. The app sends data to the device:
 - a) After successfully connecting to the device, you can get the bridge binder that communicates with the device through the `onServiceBinded(E binder)` method in `BleProfileServiceReadyActivity`. (See Demo for details)
 - b) After getting `WBYService.WBYBinder`, you can call the corresponding method in `WBYBinder` to communicate with the device.

`WBYBinder` includes the following methods:

- 1) Sync user: `syncUser(User user);`
- 2) Sync unit: `syncUnit(byte unit);`

5. The device sends data to the app:
 - a) After get `STATE_INDICATION_SUCCESS`, the data sent by the device to the app will trigger the following corresponding methods:
 - i. Change and stable weight data and temperature (only AC03 supports temperature):
`onGetWeightData(WeightData weightData);`
 - ii. Status information returned by the device:
`onGetSettingStatus(@AicareBleConfig.SettingStatus int status);`
 - iii. Information such as the Bluetooth version, date, time, and impedance returned by the device:
`onGetResult(int index, String result);`
 - iv. The body fat data returned by the device (only need to get the data whose `isHistory` is false):
`onGetFatData(boolean isHistory, BodyFatData bodyFatData);`
 - v. The number of decimal point returned by the device:
`onGetDecimalInfo(DecimalInfo decimalInfo);`
 - vi. Algorithm ID information returned by the device:
`onGetAlgorithmInfo(AlgorithmInfo algorithmInfo);`

6. Disconnect:

You can disconnect by calling `disconnect()` in `WBYService.WBYBinder`.

I. Object description

(aicare.net.cn.iweightlibrary.entity) :

1. AlgorithmInfo(Algorithm ID information)

Type	Param Name	Description
double	weight	Weight
int	algorithmId	Algorithm ID
int	adc	Impedance value
DecimalInfo	decimalInfo	Decimal point

2. BM09Data

Type	Param Name	Description
int	agreementType	Agreement type
int	unitType	Unit type
DecimalInfo	decimalInfp	Decimal point
double	weight	Weight
int	adc	Impedance value
double	temp	Temperature
int	algorithmId	Algorithm ID
int	did	(Currently useless)
String	bleVersion	Bluetooth version
int	bleType	Device type(0x09)
String	address	Device address
long	timeMillis	Measurement timestamp
boolean	isStable	Is the data stable?

3. BM15Data

Type	Param Name	Description
String	version	Bluetooth version
int	agreementType	Agreement type
int	unitType	Unit type
double	weight	Weight
int	adc	Impedance value
double	temp	Temperature (if temp=6553.5, the scale does not support temperature)
int	algorithmId	Algorithm ID
int	did	(Currently useless)
int	bleType	Device type(0x15)
String	address	Device address

4. BodyFatData(Body fat data)

Type	Param Name	Description
String	date	Measurement date
String	time	Measurement time
double	weight	Weight
double	bmi	Body mass index
double	bfr	Body fat rate
double	sfr	Subcutaneous fat rate
int	uvi	Visceral fat
double	rom	Rate of muscle
double	bmr	Basal metabolic rate
double	bm	Bone Mass
double	vwc	Moisture rate
double	bodyAge	Physical bodyAge
double	pp	Protein percentage

int	number	ID
int	sex	Sex(1, male; 2, female)
int	age	Age
int	height	Height
int	adc	Impedance value

5. BroadData

Type	Param Name	Description
String	name	Device name
String	address	Device address
boolean	isBright	Is the scale bright?
int	rss	Signal value
byte[]	specificData	Broadcast data
int	deviceType	Device type

6. DecimalInfo(Decimal point information)

Type	Param Name	Description
int	sourceDecimal	Source data decimal point
int	kgDecimal	kg decimal point
int	lbDecimal	lb decimal point
int	stDecimal	st decimal point
int	kgGraduation	kg graduation
int	lbGraduation	lb graduation

7. User(User information)

Type	Param Name	Description
int	id	ID
int	sex	Sex(1, male; 2, female)

int	age	Age
int	height	Height
int	weight	Weight
int	adc	Impedance value (deprecated)

8. WeightData

Type	Param Name	Description
int	cmdType	Command type (1, change; 2, stable; 3, impedance measurement)
double	weight	Weight
double	temp	Temperature (if the temperature is Double.MAX_VALUE, the scale does not support temperature)
DecimalInfo	decimalInfo	Decimal point information
int	adc	Impedance value
int	algorithmType	Algorithm ID
int	unitType	Unit type
int	deviceType	Device type