



BLE SDK GUIDELINE

(Android)



i-Neighbour



i-TimeTec



TimeTec Parking



TimeTec VMS



TimeTec Access

Project Owner: TimeTec Cloud Sdn Bhd
Prepared By: Jack Tan Chee Jhen & Lee Boon Kong
Document Version Date: 23 August 2019

Copyright Notice

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without written permission from Timetec Cloud Sdn Bhd. Every precaution has been made to supply complete and accurate information. Information in this document is subject to change without prior notice.

Disclaimer

No person should rely on the contents of this publication without first obtaining advice from a qualified professional person. The company expressly disclaims all and any liability and responsibility to any terminal or user of this book, in respect of anything, and of the consequences of anything, done by any such person in reliance, whether wholly or partially, upon the whole or any part of the contents of this book.

TimeTec Cloud Sdn Bhd

TimeTec BLE Development kit

What is included?

The development kit contains instructions for BLE-16 and BLE-2, including optimized C++ codes and encryptions, precompiled and it's ready to be used in your projects in a single .arr format Android library.

Minimum required SDK: Android API Level 21

TimetecBleSdk

Public Static Constants

String	BLE_SERVICE_UUID - UUID for The BLE service
String	BLE_CONTROL_CHARACTERISTIC_UUID - UUID for BLE control characteristic
String	BLE_READ_LOG_CHARACTERISTIC_UUID - UUID to read BLE logs
String	OPERATION_DELETE - Delete operation String for unique ID and Card ID
String	OPERATION_ADD - Add operation String for unique ID and card ID

Public Constructor

TimetecBleUtils (Context context)
- Requires Context argument

Public Methods

boolean	setMacAddress (String macAddress); - Pass in macAddress to initialize the class. - This must be called before the data is sent to BLE devices. - Accepts Raw Mac Address, eg. "00:00:00:00:00:00" - Returns true if successfully set, and false if otherwise. eg: setMacAddress ("34:55:32:8A:1D:37")
byte[]	getParamBle2 (int channel); - Get the BLE-2 instructions that will trigger either channel "1", "2", or "3" - Send the return byte array to your BLE Destination to trigger it. - Use this if you intend to send data to a BLE-2 - The returned byte array is ready to be written into Bluetooth Characteristics
byte[]	getParamBle16 (int channel); - Same as getParamBle2 but with up to 16 channels("1" to "16") - Use this if you intend to send data to a BLE-16 - The returned byte array is encrypted - The returned byte array is ready to be written into Bluetooth Characteristics

byte[]	getParamKeyfold (String id, int operation) - Get parameters to write keyfold id into BLE - id must be between 000000 and FFFFFFFF - operation can be either OPERATION_ADD or OPERATION_DELETE eg getParamKeyfold("4443540",TimetecBleSdk.OPERATION_ADD)
byte[]	getParamKeyfoldClear() - Get parameter to clear keyfold data in BLE
byte[]	getParamCard(String id, int operation) - Get parameters to write card id into BLE - operation can be either OPERATION_ADD or OPERATION_DELETE eg getParamCard("4443540",TimetecBleSdk.OPERATION_ADD)
byte[]	getParamCardClear() - Get parameters to clear card data in BLE

TimetecBleSdkAuth

- To verify license

Public Static Constants

- None

Public Constructor

TimetecBleUtils (Context context)

Public Methods

void	verifyLicense(String licenseld) - Call this method to verify license and corresponding MAC Addresses - This method needs to be called at least one time when the app is installed, or before attempting to use TimetecBleSdkUtils
void	verifyLicense(String licenseld, TimetecBleSdkEventListener listener) - Pass in TimetecBleSdkEventListener for callbacks during getMacAddress

Public Static Methods

void	verifyLicense(Context context, String licenseld) - Static method to verify license and corresponding MAC Addresses. - This method needs to be called at least one time when the app is installed, or before attempting to use TimetecBleSdkUtils - Works similar to verifyLicense (String licenseld) , but does not require you to construct a new object.
void	verifyLicense(Context context, String licenseld, TimetecBleSdkEventListener listener) - Pass in TimetecBleSdkEventListener for callbacks during getMacAddress - Works similar to verifyLicense (String licenseld, TimetecBleSdkEventListener listener) , but does not require you to construct a new object.

Interface TimetecBleSdkEventListener

void	onGetMacAddress() - Method is called before online verification starts
void	onGetMacAddressSuccess() - Method is called after verification process successfully executed
void	onGetMacAddressFailed() - Method is called whenever verification process failed at any point

TimetecBleKeyfold

Public Static Constants

String	KEYFOLD_SETTING_SERVICE_UUID - UUID for The Keyfold service
String	KEYFOLD_SETTING_CHARACTERISTIC_UUID - UUID for keyfold characteristic

Public Constructor

TimetecBleKeyfold()

Public Methods

byte[]	getSetFloorParam (int buttonNo, int floorNo) - Set floor number to a button - buttonNo must be a int between 1 to 4 - Position of buttons on the keyfold are as follow: 3 4 1 2 - floorNo must be a int of 1 to 16 Eg. setting button 3 with floor 14, getSetFloorParam(3, 14)
--------	--

User Guide and Instructions

Installing the package into your Android Studio Project:

1. Click File -> New -> Import Module, select the TimetecBleSdk.aar package to store the file and click Finish.
2. In the build.gradle file at app level, include the following:

```
dependencies {  
    implementation project(":TimetecBleSdk")  
}
```

3. In settings.gradle file, add the following:

```
include ':app', ':TimetecBleSdk'
```

(you can have more than one "include ':app'")

4. Gradle Sync and you'll have the BLE SDK, feel free to use TimetecBleSdk classes in your project.

License Authentication

License Authentication and Authorization is required to connect your TimeTec BLE products with this Android-Java SDK. This process needs to be done when the application is installed in the phone. Below are the recommended methods for License Authentication:

1. Install Mobile Application and run it on the first launch.

This is the most efficient method, however if you have any additional devices added afterwards, the user will not be updated with the latest license.

2. Authenticate occasionally

Occasionally updates the user with the latest license. Not the most efficient, but will keep your users up to date if you plan to add additional BLE products in the future.

The authentication also requires network connection. It'll take a few seconds to authenticate your products with your SDK.

To authenticate, just create an object instance of TimetecBleSdkAuth class, and provide your license ID as a argument to getAddress() method. Refer to the documentation for more information on the class and method.

Connecting your BLE through Android

Depending on the API levels you are targeting (minimum SDK required is level 21 or above), you can use the methods provided by the official Android Development Kit to connect to a Bluetooth Low Energy. You will need to connect the Bluetooth Low Energy as a GATT.

Refer to <https://developer.android.com/guide/topics/connectivity/bluetooth-le> for information on how to connect to a BLE device.

You need to connect to the Service and Characteristic of the BLE provided by the SDK, then write to the Characteristics with the values, and state the BLE Relay levels that you would like to trigger. Refer to the `TimetecBleSdk` class documentation for more information.

Steps to use this library:

- 1) Call `getMacAddress()` at `TimetecBleSdkAuth` to verify your license.
- 2) Connect to a BLE device.
- 3) Call `setMacAddress()` before attempting any kind of operations.
- 4) If `setMacAddress()` is successful, proceed by searching for Services, Characteristics and write your data to characteristics.
- 5) Follow the same process if you are connecting to a keyfold controller: use `TimetecBleKeyfold` class, `setMacAddress()` is not required.