HawkEye TDAService

SDK Manual

R&D Computer System Co., Ltd.

TDA_Manual_Eng_R180724.odt

Table of Contents

Computer Software Developer License Agreement i

Part List of TDA Product 1

HawkEye TDA Features 1

Application Development Steps 2

System Implementation after Application Development 3

Application Workflow 4

TDAService API 6

- serviceTA 6
- notificationTA 6
- readerTA 7
- nidNumberTA 8
- nidTextTA 8
- nidPhotoTA 10
- infoTA 10
- Data Type 13
- Return and Error Code Table 13

Meaning of TDAService Icons on Notification Bar 15

Using TDADemo 16

TDASample and Source Code for Developing 18

License file Management (DLS) 22

Card Readers Connecting Conditions of TDAService 23

Card Reader Usage 24

FAQs about HawkEye TDA 27

Computer Software Developer License Agreement

Please read the terms of this Agreement prior to entering into this Agreement by enabling this Computer Software.

After you have read the terms of this Agreement, if you don't contractually agreed, please return the Computer Software that is on the disc or any other type of media, including documents and other attachments (if applicable) to the place where you received them within 7 days (seven days) from the date of receiving the Computer Software. If you received the Computer Software by downloading or copying, you must destroy or delete the Computer Software and all the copies from the hard disk, network data storage units, data storage units in the Internet network and other storage media immediately.

Once you have installed or use this Computer Software, even just one part, you agree to be bound as a licensee to use the Computer Software pursuant to this Agreement. The following in this Agreement is called "Licensee"

R&D Computer System Co., Ltd. is located at 48/20, Ratchadapisek Soi 20, Ratchadapisek Road, Sam Sen Nok, Huaikhwang, Bangkok, Thailand, called "Licensor".

Licensor and Licensee agree to enter into an Agreement with the following terms:

1. Permission to use Computer Software.

1.1 Licensor owns copyright of Thai ID Card Development Kit for Android (TDA), HawkEye TDA, TDA SDK, TDAService, TDA Java archive files, TDASample, TDADemo, sample programs source code and APK files, other computer programs and the documents specified in this Agreement ("Computer Software").

1.2 The Licensor agrees to grant the Licensee the use of the Computer Software described in this Agreement in the following manner:

- Licensee may use this Computer Software for development of the application programs.
- For the developed application programs which any portion of this Computer Software is included, merged, linked or ran, the application programs are considered part of this Computer Software. And the Licensee must treat the application programs as well as this Computer Software.
- Licensee will only use this Computer Software and application programs for normal business operations of Licensee.
- Licensee will only use this Computer Software with a smart card reader provided or designated by Licensor.
- There will be no more than 1 person (one person) per license (one right) is using Computer Software at the same time.
- This Agreement does not grant the Licensee to obtain the source code of the Computer Software, exclude sample programs.

2. Permit Period

2.1 Licensee will use this Computer Software from the date of commencement of the Agreement until terminated or until the Licensor terminates the license.

3. The rights and obligations of the Licensee.

3.1 The Licensee has the right to use the Computer Software and documents as specified in this Agreement.

3.2 The Licensee does not have the right to make, rent, loan, or resell any Computer Software or document or any rights granted to it under this Agreement.

3.3 In case the Licensee has developed an application program, Licensee may only distribute TDAService and TDA Java archive files to third parties in the manner permitted by licensing. And the Licensee must deliver this Agreement in the form

of a paper document or computer file attached to the application program that are delivered to the other party. And anyone else who receives the application program files will be required to be a Licensee of Computer Software under this Agreement.

3.4 The Licensee has the right to make a copy of Computer Software (one set) to prevent loss during the use.

3.5 The Licensee agrees not to alter, modify, delete, destroy, damage, or obscure the name of the Computer Software, name of the copyright owner or trademark, whether intentionally or negligently.

3.6 The Licensee will handle Computer Software, documents and copies by the best method. And will not allow anyone to access, use, study, copy or do any one of the Computer Software, documents and copies.

4. Non-reverse engineering and others.

Licensee will not do or try to do, on their own or authorize, refrain from assigning or encouraging others to do any of the following: modify, adapt, damage, reverse engineer, translate, reverse translate, decompile, disassemble, system analyze, black box test, decrypt, byte code debug, binary code debug, rebuild, deconstruct, sniff, simulate, emulate and any other actions that produce the same effect, no matter what the name is called, to the components of Computer Software, firmware, protocols, user interfaces, application programming interfaces, hardware and components.

5. Return Computer Software.

In case of Agreement terminated, Licensee must:

- Returns the Computer Software, documents and all copies to the Licensor immediately.
- Destroys or deletes Computer Software and all the copies from the hard disk, network data storage units, data storage units in the Internet network and other storage media immediately.

6. Trade Secrets

Licensee acknowledges that all information, technical information, procedures and processes relating to this Computer Software are the trade secrets of the Licensor. Licensee will not disclose any of these trade secrets to third parties and will try the best to protect these trade secrets.

7. Terms of Use

- Licensee will not use Computer Software for illegal processes.
- Licensee will serves or uses this Computer Software for card holder only.
- Licensee will not read the data from Thai national ID card without permission of card holder.
- Licensee will not disclose or distribute or share Thai national ID card’s photos and text data to others without the permission of card holder.
- Any damages caused by this Computer Software or the information got from this Computer Software, Licensee must be liable for all, including the damages of civil law and others. Licensor will not be held liable to and will not accept any liability, obligation or responsibility to all damages.

HawkEye TDAService SDK Manual

R&D Computer System Co., Ltd.

TDAService SDK is a software for Thai national ID card reading. It's an important part of Thai ID Card Development Kit for Android (HawkEye TDA) for TDA series product.

1. Part List of TDA Product

1.1 One smart card reader, USB or Bluetooth type depend on the model.

1.2 TDAService SDK (downloadable from Internet) that include

- Backup copy of TDAService application file (APK). For the new and stable version, please download from Google Play.
- The simple application file (APK) and source code of TDASample. This application is an easy sample for developers.
- The application file (APK) and source code of TDADemo. This application will demonstrate the API of TDAService for complex application development.
- TDA Java archive file (TDA.jar) for building the application.
- This SDK manual.

2. HawkEye TDA Features

2.1 Uses for smart card readers that support HawkEye TDA such as TDA2910R, TDA3310M2 (USB type) and TDA301BT (Bluetooth type).

2.2 Supports Java with Android Studio.

2.3 Supports Android 4.0 and higher versions.

2.4 Can read Thai national ID smart card of Thai Department of Provincial Administration as well.

3. Application Development Steps

3.1 Buy one set of HawkEye TDA product. It should be TDA301BT that can be used for developing both Bluetooth and USB readers.

- The TDA301BT is normally used wirelessly via Bluetooth. But developer can purchase an OTG cable and connect it to the existing Micro-USB cable in the TDA package. Then plugged into an Android device to develop TDA SDK in form of a USB reader.

3.2 Download the TDA SDK from company web site, in HawkEye TDA product page. Or direct download from this link: bit.ly/1XmTfEV

3.3 Download TDAService application from Google Play by searching "TDAService" or scanning this QR Code. And install into an Android smart phone.

3.4 Install TDADemo application from TDADemo.apk in SDK into smart phone. Then try each TDAService API command from TDADemo screen.

3.5 Study source code of TDASample and TDADemo.

3.6 Create a new project in Android Studio.

3.7 Import TDA Java archive file (TDA.jar) into the project.

3.8 Start application developing by calling TDA.jar with TDAService API.

3.9 Test the developed application.

3.10 Build application file (APK) and distribute to customers or upload to Google Play.

Note

Developer should has some experience of Android application development before using the HawkEye TDAService SDK.

4. System Implementation after Application Development

4.1 Let users purchase HawkEye TDA products. Choose TDA301BT for Bluetooth (wireless) readers or choose TDA2910R, TDA3310M2 if require USB type readers.

- For the special Apple Lightning reader, TDAi301U8, that can be used with both iOS and Android devices. When use with Android device, users will need to use both OTG cable and USB cable. Connect the OTG's Micro-USB connector to the Micro-USB port of Android device and connect Micro-USB connector of the USB cable into the Micro-USB port of the card reader. The reader will not work if plug the cable in the wrong direction.

4.2 For Bluetooth reader, users have to pair the reader with Android system first. Go to **Settings** menu, tap **Bluetooth** and then **Pair**.

4.3 On users phone, install TDAService from Google Play by searching "TDAService" or scanning the QR Code.

4.4 Install the developed application (Target App) from Google Play or from APK file.

4.5 Let users start using Target App.

5. Application Workflow



Target App

It is developing application for reading Thai ID card.

TDA.jar

It is TDA Java archive file of HawkEye TDAService SDK. Developers have to import this file into the project.

TDAService

It is a background service that run inside Android system. Users have to install this TDAService application on target phone.

The service program is responsible for receiving TDAService API commands from Target App and executing them, such as reads data from Thai ID card. Within the TDAService there will be a license file use for checking a registered reader.

TDA Compliant Smart Card Reader

It is a smart card reader that comes with a HawkEye TDA package. The reader has already been registered in the license file. The unregistered reader will not work with TDAService.

Thai National ID Card

It is the Thai national ID smart card.

6. TDAService API

6.1 `serviceTA(String paraString)`

Description Control, monitor the TDAService and update the license file.

Input Parameter **`String paraString`**

"0" = Stop the service.

"1" = Start the service without main application name.

"1,MainAppName" = Let the service start and assign the main application name (target application under development) for displaying in service notification. The name of main application can be up to 20 characters in length.

"2" = Let the service update TDA license file from the Internet. Will return "02" after updating successful otherwise will return an error code.

"9" = Ask for the status of the service program.

Return Type String

Return Value Status of the service program.

"00" = Service stopped

"01" = Service started and is ready.

"02" = License file updated successfully.

"09" = Service is starting, and is not ready now.

Note After the first calling **`serviceTA("1,xxxx")`** the return code may be "09" for a while. Please check again with **`serviceTA("9")`**, and wait until "01" return.

Usage Sample **`serviceTA("1,TDADemo")`**: Let the service start, and assign the main application name to "TDADemo".

6.2 `notificationTA(String paraString)`

Description Control and check the status of notification.

Input Parameter **`String paraString`**

"0" = Disable service status on notification.

"1" = Enable service status on notification.

"9" = Check the status of notification setting.

Return Type String

Return Value Status of notification setting or error code.

"00" = Notification status disabled

"01" = Notification status enabled

Note The purpose of this command is to turn on or turn off the status of TDAService on the notification bar. Developers can disable it if they wants to handle notifications by themselves or do not want any notifications on the screen. Status disabling does not affect the reading of data and the operation of other parts.

Normally the notification status enabled after the service program starts at the first time.

Usage Sample **`notificationTA("0")`**: Disable service status on notification.

6.3 `readerTA(String paraString)`

Description Find and connect a card reader and select the operating mode.

Bluetooth Auto Mode: The TDAService automatically searches for paired Bluetooth reader all the time until the Bluetooth reader found and connected successfully. If the connection is lost, TDAService will rescan and find the reader again.

Bluetooth Single Scan Mode: TDAService will search the Bluetooth reader for 10-15 seconds. If a **paired** reader is found, it will connect immediately. If not, it will return to Standby Mode when time out.

Bluetooth Standby Mode: This is normal state of TDAService, which will not find the reader. If the connection is lost while in **Standby Mode**, TDAService will not re-connect it and will not scan the reader.

Input Parameter **`String paraString`**

"0" = Go to **Bluetooth Standby Mode**. Return code is the current mode ("00", "01" or "02").

"1" = Go to **Bluetooth Single Scan Mode**. Set operation mode and then return immediately. Return code is the current mode ("00", "01" or "02"). After scanning, whether the reader is found or not, TDAService will return to **Standby Mode**. Please call **`infoTA("3")`** for checking if Bluetooth reader found and connected.

"2" = Go to **Bluetooth Auto Mode** which will automatically find and connect the Bluetooth card reader. Return immediately after mode setting, no waiting for results. Please call **`infoTA("3")`** for checking if Bluetooth reader found and connected.

"3" = Stop Bluetooth reader connection then return "03". The current operation mode will not change. The Bluetooth reader will automatic turn off after disconnected.

"9" = Check the current mode of Bluetooth operation. The return code will be "00", "01" or "02".

Return Type String

Return Value Current operation mode or error code.

"00" = Bluetooth Standby Mode

"01" = Bluetooth Single Scan Mode

"02" = Bluetooth Auto Mode

"03" = No reader connected

Note These commands are for Bluetooth readers only. The USB reader is always in automatic mode, no need to run these commands.

If the final system does not use Bluetooth readers, do not use these commands.

Users need to turn on the Bluetooth of Android devices before using these commands.

TDAService always go to **Bluetooth Standby Mode** when starting.

Usage Sample **`readerTA("1")`**: Go to Bluetooth Single Scan Mode.

6.4 `nidNumberTA(String paraString)`

Description Read Thai national ID number (13-digit) from card.

Input Parameter **`String paraString`**

"0" = Automatic select reader. Choose USB reader first then Bluetooth reader.

"1" = Read data from USB reader.

"2" = Read data from Bluetooth reader.

Return Type String

Return Value Thai national ID number or error code.

Usage Sample **`nidNumberTA("0")`**: Read Thai national ID number (13-digit) from any connected reader, automatic selection.

6.5 `nidTextTA(String paraString)`

Description Read all text data from Thai national ID card.

Input Parameter **`String paraString`**

"0" = Automatic select reader. Choose USB reader first then Bluetooth reader.

"1" = Read data from USB reader.

"2" = Read data from Bluetooth reader.

Return Type String

Return Value Text data read from Thai national ID card or error code.

Data format is:

"NID number#Thai title#Thai first name#Thai middle name#Thai last name#English title# English first name#English middle name#English last name#Number#Moo#Trok#Soi# Thanon#Tambon-Khwaeng#Amphoe-Khet#Changwat#Gender#Birth date#Issue place#Issue date#Expiry date#Issue number"

	Field	Text size	Note
1	NID number	13 characters	"1234567890123"
2	Thai title	total 100 characters maximum (# not included)	
3	Thai first name		
4	Thai middle name		
5	Thai last name		
6	English title	total 100 characters maximum (# not included)	
7	English first name		
8	English middle name		
9	English last name		
10	Number	total 150 characters maximum (# not included)	
11	Moo		
12	Trok		
13	Soi		
14	Thanon		
15	Tambon/Khwaeng		
16	Amphoe/Khet		
17	Changwat		
18	Gender	1 character	"1"=Male, "2"=Female
19	Birth date	8 characters	YYYYMMDD (Buddhist Era) DD = "00" if day is unknown. MM = "00" if month is unknown.
20	Issue place	100 characters maximum	
21	Issue date	8 characters	YYYYMMDD (Buddhist Era)

	Field	Text size	Note
22	Expiry date	8 characters	YYYYMMDD (Buddhist Era) “99999999” for life long card.
23	Issue number	14 characters	May be not available in some card versions.

Data Example “3650800011234#นาย#สมชาย##สามรักสกุล#Mr.#Somchai##Samruksakul#123/45#หมู่ที่ 12####ตำบลบึงพระ#อำเภอเมืองพิษณุโลก#จังหวัดพิษณุโลก#1#25240119#เทศบาลนครพิษณุโลก#25521026#25590118#12345678901234”

Usage Sample **`nidTextTA("1")`**: Read all Thai national ID text data from USB reader.

6.6 `nidPhotoTA(String paraString)`

Description Read portrait photo from Thai national ID card.

Input Parameter **`String paraString`**

“0” = Automatic select reader. Choose USB reader first then Bluetooth reader.

“1” = Read data from USB reader.

“2” = Read data from Bluetooth reader.

Return Type Byte [ ]

Return Value Photo data or error code

If the first byte value is 0xFF (hex), it is the photo data.

If the first byte value is 0x45 (hex), it indicates an error code.

A readable image is a byte array of 5120 bytes in JPEG format. The image size is 297x355 pixels or 148x178 pixels. The first 2 bytes of data are 0xFF, 0xD8.

The error code is 2 bytes. The first byte is 0x45 (the ASCII code of the letter “E”). The second byte is the value of the ErrorCode. The ErrorCode is an 8-bit signed byte. For example, 0x45 0xFF is Error -1 and 0x45 0xFE is Error -2.

Usage Sample **`nidPhotoTA("2")`**: Read portrait photo from Bluetooth reader.

6.7 `infoTA(String paraString)`

Description Get information of TDA, split into multiple sub-commands.

`infoTA("0")` Get TDA information.

Return Type String

Return Value TDA information. Each field is separated by a # character in this way:

“Info of TDAService#Info of TDA.jar#Copyright Information”

	Field	Size	Example
1	Info of TDAService	Not defined	“TDAService 1.0.00S”
2	Info of TDA.jar	Not defined	“TDA.jar 1.0.00J”
3	Copyright Information	Not defined	“Copyright R&D Computer System Co., Ltd.”

Data Example “TDAService 1.0.00S#TDA.jar 1.0.00J#Copyright R&D Computer System Co., Ltd”

`infoTA("1")` Get main application name.

Return Type String

Return Value Main application name that defined by **`serviceTA("1,MainAppName")`**

	Field	Size	Example
1	Main application name	20 characters maximum	“TDADemo”

Data Example “TDADemo”

Note Will return “” if didn’t define the main application name in **`serviceTA("1")`**

`infoTA("2")` Get information of connected readers.

Return Type String

Return Value Information of connected readers or error code.

Connected readers information. Each field is separated by a # character in this way:

“Info of USB reader#Info of Bluetooth reader”

	Field	Size	Example
1	USB reader model and ID	Not defined	“TDA2910R - 12345678901234”
2	Bluetooth reader model and ID	Not defined	“TDA301BT - FT_123456789012”

Data Example “TDA2910R - 12345678901234#TDA301BT – FT_123456789012”

(When connecting to both USB and Bluetooth readers)

“TDA2910R – 12345678901234#”

(In case USB reader is connected only)

“#”

(In case no readers are connected, it will return a single # character)

`infoTA("3")` Get readers status.

Return Type String

Return Value Readers status or error code.

The status of the card reader is two digits number. The left digit is the status of the USB reader. The right is the status of the Bluetooth reader.

"0" = Reader not found

"1" = Found reader, no card is inserted in the reader.

"2" = Found reader, and found a card inserted in the reader.

"3" = Found reader but it is an unregistered reader.

Data Example "20"

(Found a USB reader with a card inserted ("2") but no Bluetooth reader found ("0").

"12"

(Found a USB reader without a card ("1") and found a Bluetooth reader with a card inserted ("2").

`infoTA("4")` Get license file information.

Return Type String

Return Value License file information or error code.

License file information includes the license file name and the file date (YYYY-MM-DD format), with the # character separated by this.

"License Filename#Date of License File"

	Field	Size	Example
1	License Filename	Not defined	"/storage/TDAService/rdnidlib.dls"
2	Date of License File	10 characters	"2016-01-21"

Error code:

"-2" Invalid license file.

"-12" License file not found or error.

Data Example "/storage/TDAService/rdnidlib.dls#2016-01-21"

`infoTA("5")` Get TDAService folder name.

Return Type String

Return Value TDAService folder name or error code.

	Field	Size	Example
1	TDAService Folder Name	Not defined	“/storage/TDAService”

Data Example “/storage/TDAService”

6.8 Data Type

Type	Description
`String`	String Unicode UTF8
`Byte[]`	Byte Array
`Byte`	8-bit signed two’s complement integer, value from -128 to +127

6.9 Return and Error Code Table

Return Code (String)	Return Code (Byte)	Name	Description
“OK”	4F 4B	TA_SUCCESS	All done, no error.
“-1”	45 FF	TA_INTERNAL_ERROR	Found internal error.
“-2”	45 FE	TA_INVALID_LICENSE	Invalid license.
“-3”	45 FD	TA_READER_NOT_FOUND	Smart card reader not found.
“-4”	45 FC	TA_CONNECTION_ERROR	Connection error.
“-5”	45 FB	TA_GET_PHOTO_ERROR	Can’t get photo from card.
“-6”	45 FA	TA_GET_TEXT_ERROR	Can’t get text data from card.
“-7”	45 F9	TA_INVALID_CARD	Invalid card or not Thai ID card.
“-8”	45 F8	TA_UNKNOWN_CARD_VERSION	Unknown Thai ID card version.
“-9”	45 F7	TA_DISCONNECTION_ERROR	Can’t disconnect.
“-10”	45 F6	TA_INIT_ERROR	Initialize error or TDAService didn’t open yet.
“-11”	45 F5	TA_READER_NOT_SUPPORTED	Doesn’t support this reader.
“-12”	45 F4	TA_LICENSE_FILE_ERROR	License file error.
“-13”	45 F3	TA_PARAMETERS_ERROR	Parameters error.
“-14”	45 F2	TA_SERVICE_ERROR	TDAService not found or TDAService error.

Return Code (String)	Return Code (Byte)	Name	Description
"-15"	45 F1	TA_INTERNET_ERROR	Internet connection error.
"-16"	45 F0	TA_CARD_NOT_FOUND	Card not found.
"-17"	45 EF	TA_BLUETOOTH_DISABLED	Bluetooth is turned off.
"-18"	45 EE	TA_LICENSE_UPDATE_ERROR	Can't update license file.

7. Meaning of TDAService Icons on Notification Bar

- Character S: Indicates that the TDAService is running. Wait for connection to card reader. (S = Standby)
- USB symbol: Displays the USB reader status:



  - White: USB reader connected and ready to use.
  - Gray: Connected to a USB reader, but is not a registered reader which TDA can't use it. If ask TDAService with **`infoTA("3")`** command, "3x" will return.
- Bluetooth Symbol: Displays the connection to the Bluetooth reader.



  - Flashing: Finding (scanning) for a Bluetooth reader.
  - Blue or White: Bluetooth reader connected and ready to use.
  - Gray: Connected to a Bluetooth reader, but is not a registered reader which TDA can't use it. If ask TDAService with **`infoTA("3")`** command, "x3" will return.
- Numbers 0 to 100: Show progress of Thai ID card reading in percentage (%).

8. Using TDADemo

- TDADemo is a sample application for TDAService. It's a tool for learning and testing of TDAService API.
- There is TDADemo.apk file in SDK package. Developers can install and run it immediately from Android phone.
- Source code of TDADemo is also provided in SDK package.
- TDADemo using steps:
  - Check that the TDAService application is installed on the Android device. If not, please installed it from Google Play or from the development kit's APK files. If possible, should install from Google Play, as this will always be the latest TDAService version.
  - Launch TDADemo from Android phone.
  - Tap the Parameter box (located at the upper right corner of the screen) to configure the desired number. Tap the command button to try each command. Then look at the output of the left side window of screen.
    - For example, to send a command **`serviceTA("1")`**, tap the Parameter box and enter the number 1, then tap the **serviceTA** button.
  - The value returned by the application, such as "01" or error code, such as "-12", can be seen in the description of each command and the **Return and Error Code Table**.
  - Please try these steps:
    - **`serviceTA("1")`** (Look at the Icon on the notification bar)
    - **`notificationTA("0")`** (Look at the Icon on the notification bar)
    - **`notificationTA("1")`** (Look at the Icon on the notification bar)
    - Plug in the USB reader cable or turn on the Bluetooth reader that already paired with Android phone.
    - **`readerTA("2")`** (Wait until the Icon on the notification stops flashing. Use this command when trying with a Bluetooth reader only. If the project use USB reader only, this command is no need.)
    - **`infoTA("3")`** (See the result.)
    - Insert the Thai ID card into the reader.
    - **`infoTA("3")`** (See the result again.)
    - **`nidNumberTA("0")`** (Read ID number from card.)
    - **`nidTextTA("0")`** (Read Thai ID text data from card.)

- **`nidPhotoTA("0")`** (Read portrait photo from Thai national ID card.)
- **`infoTA("0")`** (Get TDA information.)
- **`infoTA("1")`** (Get main application name.)
- **`infoTA("2")`** (Get information of connected readers.)
- **`infoTA("3")`** (Get readers status.)
- **`infoTA("4")`** (Get license file information.)
- **`infoTA("5")`** (Get TDAService folder name.)
- **`serviceTA("0")`** (Stop the service.)

9. TDASample and Source Code for Developing

TDASample is a simple application with source code for developers to use as a template. There are a source code and APK file in TDA SDK. The developer can try in Android Studio immediately.

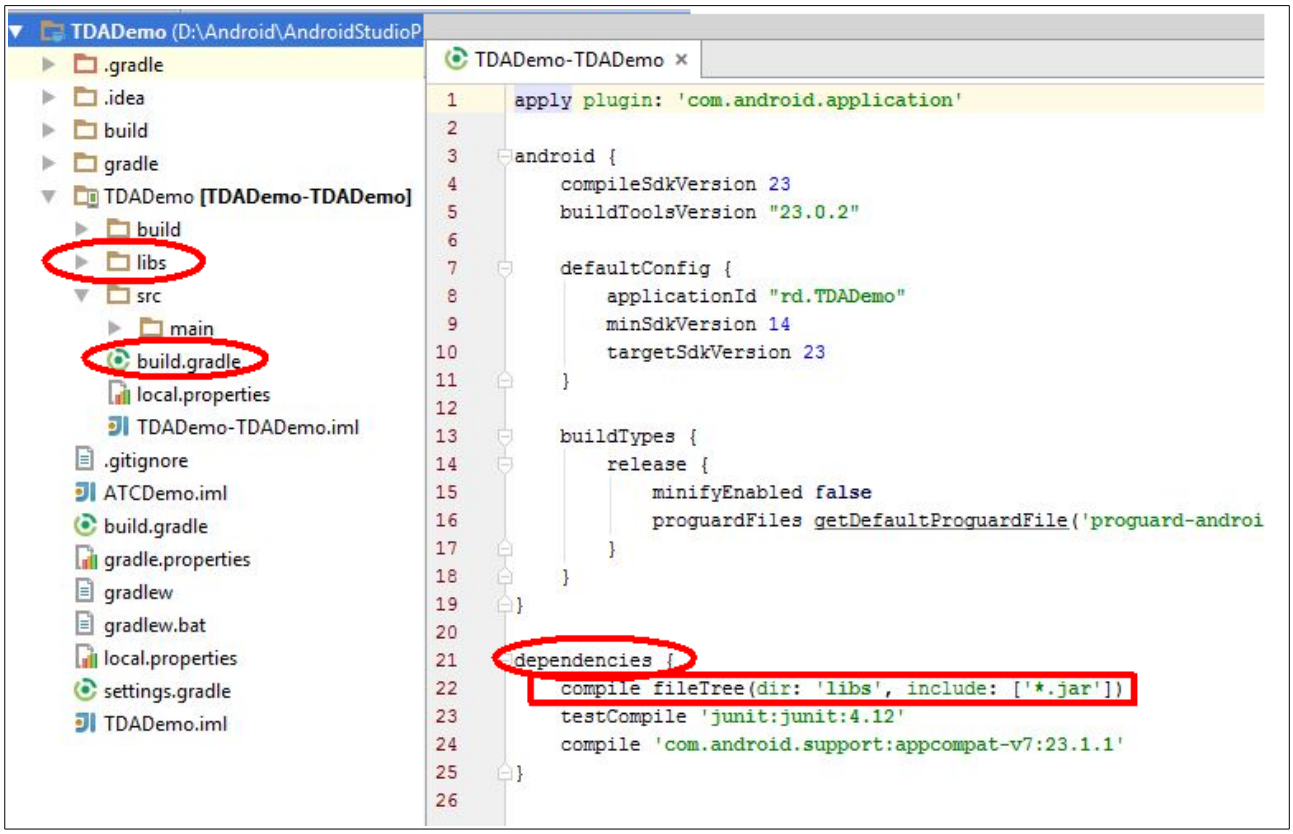
Examples of development steps and programming instructions can be found in the TDASample source code. Developers can try by these steps:

9.1 New a project

9.1.1 Create a new project in Android Studio.

9.1.2 Import TDA.jar into the project.

- Create libs folder.
- Place TDA.jar into libs folder.
- Open the file build.gradle then add these text:
  `compile filetree(dir: 'libs', include: ['*.jar'])`
  under **`dependencies`**



9.2 Application Development Guideline

9.2.1 Start the application

- Check that the TDAService is installed on Android device or not.
- Use **`serviceTA("0")`** command to close the TDAService that may remain running.

- Use **`serviceTA("1,xxxx")`** to start TDAService. Set "xxxx" as the name of the main application. This name will show on notification bar.
- Check **`serviceTA("9")`** until "01" is returned to ensure that TDAService is ready to service. It takes about 1 to 5 seconds depending on the speed of the Android device.
- Check whether the license file is present by using **`infoTA("4")`** command. If the return value is an error code such as "-2" or "-12", the license has a problem. Please update the license file with **`serviceTA("2")`** command. The Internet connection must be enabled. Alternatively, the license file (rdnidlib.dls) can be manual placed in TDAService folder. You can query the folder name from **`infoTA("5")`** command.
- Call **`notificationTA("0")`** if you don't want to display TDAService Icon on the notification bar.

9.2.2 While connecting with the readers

- Let user to insert a USB reader or turn on the Bluetooth reader.
  - It is recommended that users should pair Bluetooth reader with Android device before using application.
- Case 1: The application doesn't use the Bluetooth reader (i.e. use the USB reader only), skip below Bluetooth scanning commands and go to the Thai ID card data reading step.
- Case 2: This application uses the Bluetooth reader and want to scan Bluetooth reader automatically:
  - Turn on Bluetooth signal in Android device or notify user to turn on the Bluetooth signal.
  - Use **`readerTA("2")`** command to enter the **Bluetooth Auto Mode**. Check the return value is "02" to ensure **Bluetooth Auto Mode**.
- Case 3: This application uses the Bluetooth reader and want to scan Bluetooth reader manually
  - Turn on Bluetooth signal in Android device or notify user to turn on the Bluetooth signal.
  - Use **`readerTA("1")`** command to enter the **Bluetooth Single Scan Mode**.
  - Loop with **`readerTA("9")`** command and exit when Bluetooth reader scanning is finished.

- If the value is “00”, the scanning is finished, then check the reader connection results.
- If value is “01”, it is still under scanning, try **`readerTA("9")`** again.

▪ Check the Bluetooth reader connection results by using **`infoTA("3")`** command. The return value is a two digit number, look at the last number (the number on the right).

- If the value is “x0”, the Bluetooth reader is not found or can’t be connected. Then the application should prompt the user and repeat the scanning (how many times depends on developer needs).
- If the value is “x1” or “x2”, the Bluetooth reader connection is successful and ready to use.
- If the value is “x3”, the connection is successful. But it is a reader that has not been registered with the license file (which will not work). Please try to update the license file to the latest version by using **`serviceTA("2")`** command. Then call **`infoTA("3")`** again. If the return value is still “x3”, then you must inform the user that this reader is not support and please replace with a new registered reader.

• To disconnect from a Bluetooth reader or switch to another reader. Disconnect the old connection by using the **`readerTA("3")`** command. The Bluetooth reader will automatically shut down.

9.2.3 Thai ID card data reading

• When the user wants to read data from Thai ID card, please use the command **`nidNumberTA("0")`**, **`nidTextTA("0")`** or **`nidPhotoTA("0")`** as needed.

▪ Parameter “0” is read from any reader automatically. Use parameter “1” for reading from USB reader or use the parameter “2” for reading from Bluetooth reader.

▪ After reading command, if the returned value is “-2” indicates that TDAService does not recognize the reader. Please update the license file with the command. **`serviceTA("2")`** (the Internet must be enabled). After updating, if it still return “-2” indicates that the reader is not from TDA product, please notify the user to replace the reader.

- If you want to check before reading what kind of reader is connected and the card is present or not, use the command **`InfoTA("3")`**.

9.2.4 When exit the application

- Always call **`serviceTA("0")`** to stop TDAService.
- Then close the application.

10. License file Management (DLS)

License File is the most important for using HawkEye TDAService. This file is a collection of registered and licensed readers for TDA.

- HawkEye TDA license filename is rdnidlib.dls.
- Can get license folder name, license filename and license date from About screen of TDAService. Or using **`infoTA("4")`** command.
- All registered readers S/N will be saved inside this license file.
- Only registered readers can be use with TDAService.
- The new license file will bigger than the old one.
- For the first run of developed application, developer should download the new license file from the Internet with **`serviceTA("2")`** command.
- In case of application software found unregistered reader from **`infoTA("3")`** command, please try to update the new license file from Internet with **`serviceTA("2")`** command or ask user to replace with a new registered reader.
- How to get the new license file:
  - Method 1: Use **`serviceTA("2")`** commands inside the application (enable the Internet first).
  - Method 2: Let users to update by themselves. By pressing the Update License button on the About screen of the TDAService application, users can access this screen in two ways, launch TDAService from Android or scroll down the notification bar and tap on the line of the TDAService application.
  - Method 3: Let TDAService update automatically when it finds an unregistered reader. However, this method may not work if the Internet is not enabled.
- Developers should be programmed to load and update license files in a variety of ways for ease of application using.
  - After installation, we should update the new license file from Internet immediately.
  - Should update the new license file automatically when get return code of "-2" or "-12" or "-11". And in case of getting "3x" or "x3" return from **`infoTA("3")`** command.
  - Always put an "Update License" command button on an application screen. Then users can solve the problem by updating license file manually by themselves.
- Updating license file requires Internet connection. Please ask user to make Internet connection when found "-15" or "-18" return code from **`serviceTA("2")`** command.

11. Card Readers Connecting Conditions of TDAService

- TDAService can support only one USB card reader at one time. TDAService may malfunction if it found more than one USB card readers in the system.
- TDAService can only use one Bluetooth reader at a time.
- TDAService can supports one USB reader and one Bluetooth reader at the same time.

12. Card Reader Usage (End Users should be notified.)

12.1 USB Card Reader Usage

12.1.1 Selecting Application for TDA USB Reader

When the USB reader is plugged into the port of your phone or tablet. Android system will detect the USB device and will ask the user what application to use with this USB device.

- Please always select the TDAService and then tap **JUST ONCE**. Should not tap **ALWAYS** or **Remember this choice**.
  - If **ALWAYS** or **Remember this choice** is selected, the next time you plug in the reader, Android system will not let user select other application. It will use the selected application forever.
- If the wrong application is selected and **JUST ONCE** is tapped, please unplug the reader and plug it again.
- If the wrong application is selected and **ALWAYS** is tapped, users have to reset the selection. Go to **Settings** > **Application Manager** or **App Info** > select the wrong selected application name > **Open by default** or **Launch by default** and tap **CLEAR DEFAULTS.** Then unplug the reader and plug it again.

12.1.2 Using USB reader with USB Type C Android device

Please use an adapter to convert Micro-USB connector of reader to USB Type C, then plug it into USB Type C port of phone or tablet.

12.2 Bluetooth Card Reader Usage

12.2.1 Bluetooth device pairing (for TDA301BT)

Normally, every Bluetooth device to use on a mobile phone needs to be “Paired” before. The pairing between the Bluetooth card reader and the mobile phone can do as follows:

- Install the batteries into reader’s battery slot.
- Push the button on the front of the Bluetooth reader. Notice there is a flashing blue light.
- On the phone or tablet, go to Android **Settings** menu.
- Tap **Bluetooth** > **Scan** then look at the bottom of the screen, try scrolling to the bottom.
- Wait for a while, the card reader name start with “FT_”, like “FT__8CDE1234ABCD”, will shown on screen. Tap that reader name.
- If it can not find a reader, tap **Scan** again.

12.2.2 The Bluetooth reader should be paired with the phone before launching the application for the first time. If an application is launched before, then it will not connect to the reader.

12.2.3 Definition of LED lights of TDA301BT Bluetooth reader

- Blue: Blink 1 time: Wait for connection.
- Blue: Blink 2 times: Successfully connected. Can read the data.
- Blue: Blink 3 times: Self-checking after power on and wait for the connection.
- White: On or flash when the card is reading.
- Yellow: On: Battery is weak. The reader should be charged.
- Orange: On when charging the battery. And it will be turned off when the battery is full.
  - If the charger is plugged in but the orange light does not light up, indicating that the battery is full or almost full. No need to charge again. That is, if the charger is plugged in then the orange light is not lit, it does not mean that the reader is malfunction. But that means charging is no need and the reader is still working well.

12.2.4 Power control for Bluetooth Reader

- Power on by pressing the front button, notice the blue light flashing.
- Power off by pressing the front button 3 seconds, all LED light will turn off.
- The reader will automatically shut down to save energy when
  - Disconnected from the phone, turn off the Bluetooth signal of the phone or move the reader far away from the phone.
  - Didn't use the reader more than 3 minutes.

12.2.5 Do not pair one Bluetooth reader with many Android devices, as it will be confusing.

12.2.6 One Android phone should be paired with only one Bluetooth reader. Always unpair the unused reader in Android phone.

12.2.7 Bluetooth reader shutting down or Bluetooth connection dropped problems.

- It can be caused by the Android device that turn off the Bluetooth itself when in standby mode or power save mode or security mode or when the screen is closed, depending on user's Android device. Correct the problems by finding the way to set the Bluetooth don't turn off when entering the various modes above, or to disable the above modes.
- The Bluetooth reader sometimes may be disconnected by interference or moving the reader far away from the phone.
- When the Bluetooth connection is off, the reader will shut itself down. Therefore, users will have to press the Bluetooth reader button to continue to use it.

13. FAQs about HawkEye TDA

Question

What is HawkEye TDA Software Development Kit?

Answer

HawkEye Thai ID Card Development Kit for Android (TDA) is a hardware and software solutions for developing applications on Android to read Thai national ID card. The kit includes a smart card reader (USB or Bluetooth), built sample applications and source code.

Question

What are the different of TDA readers such as TDA2910R, TDA3310M2 and TDA301BT from TRA readers and standard smart card readers?

Answer

TDA is an SDK suite for Android application development. Suitable for programmers or developers.
TRA is a Thai ID card reading solution for Android phone or tablet. The package includes both a reader and application software. It is suitable for general end users, no programmers required.
The standard card readers, such as the bR301, CLOUD2910R and SCR3310M2, are just smart card readers without application software. They are suitable for organizations that already have Thai ID card applications software.

Question

Can I buy only the standard smart card reader such as bR301, CLOUD2910R or SCR3310M2 (hardware only products) then use it with the TDA SDK?

Answer

The HawkEye TDA software development kit is only compatible with registered readers in the TDA series product. All TDA readers are already registered for using with the TDA before selling.
For unregistered readers such as standard readers (hardware only) and TRA readers can not use with TDA software.

Question

What are the application development steps for Thai ID card reading?

Answer

Do these steps:

- Buy one set of HawkEye TDA. It should be TDA301BT that can be used for developing both Bluetooth and USB readers.
- Study source code of TDASample, TDADemo and description of API from HawkEye TDAService SDK Manual.
- Develop a new project in Android Studio and test with the reader that come from TDA package.
- When the application is complete. Purchase additional TDA products for all users.
- Install application into the Android device and give it with TDA reader to users.

Question

Can we use the TDA developed application with other brand readers or hardware only reader?

Answer

No, because the TDA SDK is only compatible with the card reader that comes with the TDA package. Other card readers are not compatible.

Question

If the TDA card reader is broken, can I transfer the license right to other unregistered reader?

Answer

If the card reader is broken or damaged, the license is considered to be terminated.

However, all TDA readers have the warranty. If the reader is damaged during the warranty period, the company will repair the reader to normal state for customer, free of charge.

Question

Can TDA software get online data from Department of Provincial Administration - DOPA's server?

Answer

No, TDA will read the data from the chip in Thai ID card only. It works offline.

Question

What are the TDA SDK terms of use?

Answer

Conditions are:

- Do not use TDA SDK for illegal processes.
- Do serve or use this TDA SDK for card holder only.

- Do not read the data from Thai national ID card without permission of card holder.
- Do not disclose or distribute or share Thai national ID card's photos and text data to others without the permission of card holder.
- Any damages caused by this TDA SDK or the information got from this TDA SDK, you must be liable for all, including the damages of civil law and others. R&D Computer System Co., Ltd. will not be held liable to and will not accept any liability, obligation or responsibility to all damages.

Question

Where can I download TDAService?

Answer

Developers can download TDAService from Google Play or scan this QR Code.

Question

Where can I download the TDA SDK?

Answer

Developers can download the SDK software from the company website. Visit the HawkEye TDA product page (such as TDA301BT) and click on the link at the bottom on of that page. Or go directly to the page by this link: bit.ly/1XmTfEV

บริษัท อาร์แอนด์ดี คอมพิวเตอร์ ซิสเท็ม จำกัด
R&D Computer System Co., Ltd

02-694-3133
sales@rd-comp.com
http://www.rd-comp.com

TDA_Manual_Eng_R180724.odt