

For Android Applications

Contents ↗

- [Introduction](#)
- [Pre-requisites](#)
- [Download and install the required resources](#)
- [Running the ITL REST Server](#)
- [Sending REST Requests using API Tester](#)
- [SSP Lower-Level Logging](#)

Introduction ↗

The following describes how to run the ITL REST server using the Android_REST_API_Vx.x.x.apk and make endpoint calls to it and receive response data. There are many different Android versions available. This documentation has been compiled using Android Version 12 and version 1.3.2 of the Android REST API.

Pre-requisites ↗

- Android Device, Android instance on a Virtual Machine or Emulator
-  Minimum Android Version 10 (API 29)
- ITL cash handling device

Download and install the required resources ↗

Navigate to the following link to download the latest SDK.

[ITL SDK Package](#)

Installing the Android_REST_API_Vx.x.x.apk

Navigate to the APK download and launch the file.

Select to CONTINUE on any warning messages.

Your tablet and personal data are more vulnerable to attack by unknown apps. By installing this app, you agree that you are responsible for any damage to your tablet or loss of data that may result from its use.

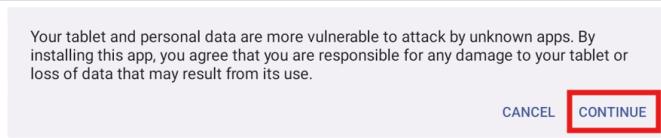
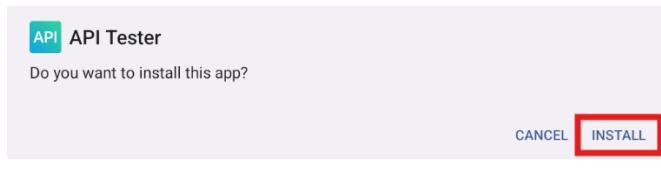
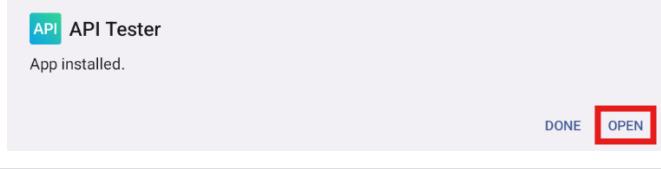
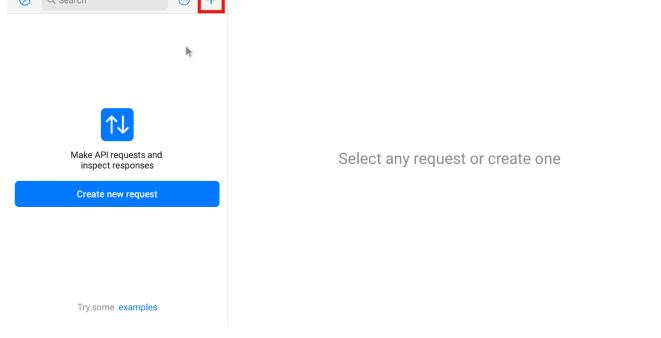
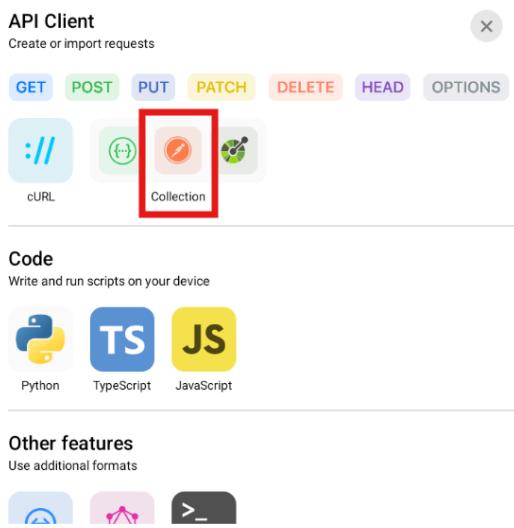
CANCEL 

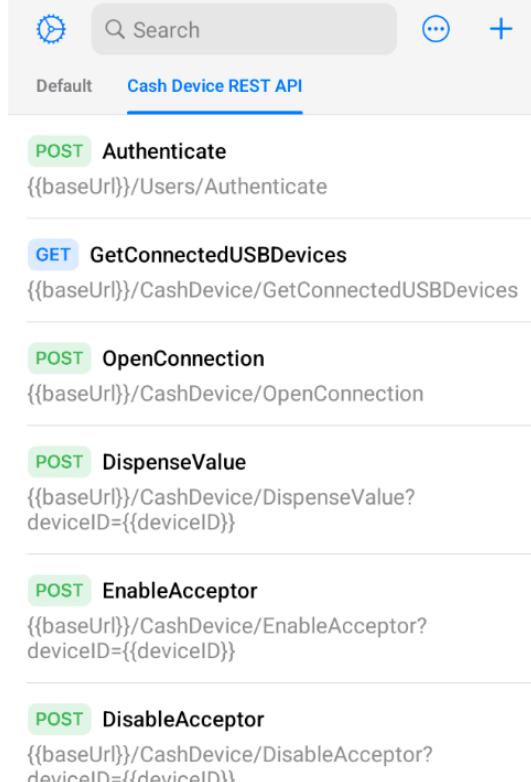
Select INSTALL to begin the process.

 **Android REST API**

Do you want to install this app?

CANCEL 

Select DONE once the install completes.	
Download and install an API Tester like the following example:	API Tester
Installing the API Tester.	
Navigate to the APK download and launch the file.	
If sideloading the APK, select to CONTINUE on any warning messages.	
Select INSTALL to begin the process.	
Select OPEN to launch the application.	
Import the Collection to the API Tester.	
Select the + icon.	
Select Collection.	

Select File.	<p>Import collection via link or file</p> <p>Link</p> <p>File</p> <p>Cancel</p>
Navigate to the saved collection file Android-API-Collection.json and select the file.	<p>Wait until the import completes.</p> <p style="text-align: center;">Importing...</p> <p style="text-align: center;">It can take a few minutes</p>
Once the import has completed you will find the collection list on the left of the API Tester app.	 <p>The screenshot shows the API Tester application interface. At the top, there's a search bar with a magnifying glass icon and a placeholder 'Search'. Below it, a navigation bar has tabs for 'Default' and 'Cash Device REST API', with 'Cash Device REST API' being the active tab. The main area displays a list of API endpoints:</p> <ul style="list-style-type: none"> POST Authenticate {{baseUrl}}/Users/Authenticate GET GetConnectedUSBDevices {{baseUrl}}/CashDevice/GetConnectedUSBDevices POST OpenConnection {{baseUrl}}/CashDevice/OpenConnection POST DispenseValue {{baseUrl}}/CashDevice/DispenseValue? deviceID={{deviceID}} POST EnableAcceptor {{baseUrl}}/CashDevice/EnableAcceptor? deviceID={{deviceID}} POST DisableAcceptor {{baseUrl}}/CashDevice/DisableAcceptor? deviceID={{deviceID}}

Running the ITL REST Server ↗

Launch the Android REST API application	
The server will launch and begin listening on http://127.0.0.1:5000/api/CashDevice	 <p>Android REST API</p> <p>V1.3.2</p> <p>Server listening on http://127.0.0.1:5000/api/CashDevice</p>

REST requests are case sensitive. Ensure upper case letters are used where advised.

The Android REST API application must remain active at all times. Closing the application will stop the API from communicating between your software and the cash device. A fix has been implemented from version 1.3.2 to ensure the server remains active until explicitly closed, however some Android versions can pause or stop apps running in the background e.g. switching between the server app and the API Testing app. Ensure your Android version can allow apps to remain active in the background.

Sending REST Requests using API Tester ↗

Set the baseURL in the Variables table.

To access the Variables table, select the 3 dots contained in the circle to the right of the search bar.

Cash Device REST API

- POST** OpenConnection
{{baseUrl}}/CashDevice/OpenConnection
- POST** DisconnectDevice
{{baseUrl}}/CashDevice/DisconnectDevice?
deviceID={{deviceID}}
- POST** StopDevice
{{baseUrl}}/CashDevice/StopDevice?
deviceID={{deviceID}}
- POST** DisableAcceptor
{{baseUrl}}/CashDevice/DisableAcceptor?
deviceID={{deviceID}}
- POST** EnableAcceptor
{{baseUrl}}/CashDevice/EnableAcceptor?
deviceID={{deviceID}}
- GET** GetConnectedUSBDevices
{{baseUrl}}/CashDevice/GetConnectedUSBDevices

Select Variables.

Variables

Ensure the correct baseURL is set in the variables table.

1 `http://127.0.0.1:5000/api`

variableKey	value
baseUrl	http://127.0.0.1:5000/api
token	value
deviceID	value

Select Done to update the table and return to the request list.	<p>Cash Device REST API</p> <p>Use double curly braces in URL, query and body params, headers and auth to reference variables: {{variableKey}}.</p> <p>baseUrl token deviceID</p> <p>Add Variable</p> <p>The response body contains a large Bearer Token:</p> <pre>{"token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1bmJxdWVfbmZSI6ImFkbWluiwibmJmljoxNzM3NjM10DA4LCJleHAiOjE3MzgyNDA2MDgsImhdC16MTczNzYzNTgwOCwiaXNzljoiSU50T1ZBVEIWRs1URUNITk9MT0dZlwiYXVkljoiSU50T1ZBVEIWRs1URUNITk9MT0dZlIn0.Mepb-70wYWjWRhVoDINPHaBRzx_wHL2kpZv-oNgbie"}</pre> <p>Copy Share Select all</p>
Request a Bearer Token for authorisation of upcoming REST requests using the Authenticate request.	<p>Select the Authenticate request and select the play button from the top right to send the request.</p>
Copy the Bearer Token to the Variables table for future requests.	<p>Highlight the Bearer Token and select Copy.</p>
Paste the Bearer Token into the token value field of the variables table.	<p>Cash Device REST API</p> <p>Use double curly braces in URL, query and body params, headers and auth to reference variables: {{variableKey}}.</p> <p>baseUrl token deviceID</p> <p>Add Variable</p>
Select Done to update the table and return to the request list.	<p>Cash Device REST API</p> <p>Use double curly braces in URL, query and body params, headers and auth to reference variables: {{variableKey}}.</p> <p>baseUrl token deviceID</p> <p>Add Variable</p>
Send the GetConnectedUSBDevices request to obtain the correct Port.	<p>Select Done to update the table and return to the request list.</p> <p>Cash Device REST API</p> <p>Use double curly braces in URL, query and body params, headers and auth to reference variables: {{variableKey}}.</p> <p>baseUrl token deviceID</p> <p>Add Variable</p> <p>The response body contains a list of connected USB devices:</p> <pre>[{"Port": 0, "DeviceName": "NV200 Spectral", "VendorId": 6428, "ProductId": 16644}]</pre>
Navigate to the Post Data for the OpenConnection request.	<p>Cash Device REST API</p> <p>Use double curly braces in URL, query and body params, headers and auth to reference variables: {{variableKey}}.</p> <p>baseUrl token deviceID</p> <p>Add Variable</p> <p>No response yet</p> <p>Make a request to see its response</p>

Update the Post Data to the correct Port returned for the [GetConnectedUSBDevices](#) request.

```

1  "ComPort": 0,
2
3  "SetFilepath": {
4    "LogFilePath": "/storage/emulated/0/Documents/ITL_SSP_logs/SSP_Log.log",
5    "SetInhibits": [
6      {
7        "Denomination": "500 GBP",
8        "Inhibit": true
9      }
10    ],
11    "SetRoutes": [
12      {
13        "Denomination": "500 GBP",
14        "Route": 2
15      },
16      {
17        "Denomination": "1000 GBP",
18        "Route": 3
19      },
20      {
21        "Denomination": "2000 GBP",
22        "Route": 4
23      }
24    ],
25    "SetCashBoxPayoutLimit": [
26      20,
27      20,
28      20,
29      0
30    ],
31    "EnableAcceptor": true,
32  }

```

Send the [OpenConnection](#) request once the Post Data has been updated. Please refer to the [OpenConnection](#) details for further information on the full Post Data body.

Update the deviceID in the variables table.

The OpenConnection request will return the correct deviceID to use on subsequent requests.

POST

OpenConnection

Params Settings

Query

No query params yet

Add Param Edit

Body

URLEncoded FormData Raw Binary

Content Type application/json >

Post Data

Headers

Content-Type application/json

200 7.33s 2.1kb

Save the deviceID to the variables table.

Cash Device REST API Done

Use double curly braces in URL, query and body params, headers and auth to reference variables: {{variableKey}}.

baseURL http://127.0.0.1:5000/api

token eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9eyJsbWfmdWVfbmFZSI6ImFkbWluZWJmIjoxNzNzE5MjczLCJleHAiOjE3Mzg2MjQwNzcsImhhdC16MTc2NzcxOTI3NywiXNzIj

deviceID NV4000-0

Add Variable

Your validator will be enabled and ready to accept currency if the below parameters of the [OpenConnection](#) request are set to true.

```

1   "EnableAcceptor": true,
2   "EnablePayout": true

```

SSP Lower-Level Logging ↴

Available from version 1.3.2

Android will store the Lower level SSP logs from version 1.3.2.

By default, logs are stored at the following path:

```
/storage/emulated/0/Documents/ITL_SSP_logs/{default_log_name}.log
```

If you wish to use a custom log location, simply specify it in the JSON request using the following format:

```
"LogFilepath": "{directory}/{log_file_name}.log"
```

To access the default logs on your Android device, navigate to:

Files -> Internal Storage -> Documents