

# 2N<sup>®</sup> Indoor Touch 2.0 API



## Content:

- [1. Purpose of Document](#)
- [2. HTTP API](#)
- [3. WEB API](#)
- [4. Supplementary Information](#)

## 1. Purpose of Document

**2N® Indoor Touch 2.0 HTTP API** provides third party applications a possibility to configure and control selected system parts and the **2N® IP Mobile** calling application.

### **Caution**

- The HTTP API service is licensed under the **91378395 2N Indoor Touch HTTP API license**.

## 2. HTTP API

- 2.1 Interface
  - 2.1.1 Authentication
  - 2.1.2 Requests
  - 2.1.3 Responses to Requests
  - 2.1.4 Cautions
- 2.2 Commands
  - 2.2.1 Session Control
    - 2.2.1.1 Login
    - 2.2.1.2 Logout
  - 2.2.2 GPIO
    - 2.2.2.1 GPIO List Getting
    - 2.2.2.2 GPIO Value Setting
  - 2.2.3 2N® IP Mobile Control
    - 2.2.3.10 2N® IP Mobile Restart
    - 2.2.3.1 Call Setup
    - 2.2.3.2 Call Accept
    - 2.2.3.3 Call Termination
    - 2.2.3.4 Call State
    - 2.2.3.5 2N IP Intercom List Getting
    - 2.2.3.6 Change of Items in Added 2N IP Intercom List
    - 2.2.3.7 Door Lock Activation
    - 2.2.3.8 2N® IP Mobile Minimization
    - 2.2.3.9 2N® IP Mobile State Info
  - 2.2.4 System
    - 2.2.4.1 System Info and Status
    - 2.2.4.2 2N® Indoor Touch Restart

### 2.1 Interface

Communication is made via the HTTPS protocol, which runs on standardized port 443. Authentication requires login and password data to be transmitted in the first connection. Subsequently, the server returns the client a cookie with a session key for further requests. Having failed to send more requests within a timeout (20 minutes at present), the client will be logged out automatically. The next requests will thus return the HTTP return code 401 (Not authorized). Then the client has to log in again.

- 2.1.1 Authentication
- 2.1.2 Requests
- 2.1.3 Responses to Requests
- 2.1.4 Cautions

### 2.1.1 Authentication

**2N® Indoor Touch 2.0** supports the following two types of authentication:

- **Basic** – Basic authentication is required according to **RFC 2617**. The service requires a password, which, however, is sent in an open format.
- **Digest** – Digest authentication is required according to **RFC 2617**.

### 2.1.2 Requests

#### ⚠ Caution

- API requests are no longer supported.

### 2.1.3 Responses to Requests

A response to a request always includes JSON data in the following format:

```
{
  "data": {
    "success": true,
    "array": [
      {
        "serial_number": "54-0562-0293"
      }
    ]
  }
}
```

Make sure that the cover-all **data** container and the nested boolean **success** are always included. Success is always true if the command has been (or is expected to have been) executed successfully. Nevertheless, it cannot be guaranteed for the time being that the success=true value really means success for all commands. For example, there may be no information on successful execution of the commands that **2N® IP Mobile** resends to the 2N IP intercoms and so the value cannot be reported. Such commands include the PUT method and the commands that do not return values: calls, opening doors, etc. Thus, the user has to use another request (call/status, e.g.) to know the command's success.

The false value is returned for many reasons. At present, the HTTP API cannot distinguish errors and specific error messages will be implemented in the next versions if necessary. The short list of potential errors is as follows:

- Request entering error – invalid argument values, missing arguments, non-existent subsystem, wrong JSON data format, incorrect HTTP method, etc.
- A **2N® IP Mobile** command is entered but the application is not running. Therefore, make sure that the application has been started before sending a **2N® IP Mobile** command.

The **data** container can include more items whose names and contents depend on the request to be sent. Our example includes an array of serial numbers.

### 2.1.4 Cautions

Currently, API is used for both providing data to the web interface (REST) and user HTTP API requirements. Thus, the data that can be viewed and configured via the web can be available in API despite not being documented.

API now supports just one web interface login at a time. Hence, if a web interface is open and the user logs into another one, API will log the user out of the first one. This will become obvious when the user enters another request via the web and is forwarded to the login dialogue. This means that the web is used by another user and data might be overwritten.

HTTP API has no such limitation. This means that more users may be authenticated, but mutual data overwriting is not checked and has to be solved by the developer.

## 2.2 Commands

- [2.2.1 Session Control](#)
- [2.2.2 GPIO](#)
- [2.2.3 2N® IP Mobile Control](#)
- [2.2.4 System](#)

### 2.2.1 Session Control

- [2.2.1.1 Login](#)
- [2.2.1.2 Logout](#)

#### 2.2.1.1 Login

The login helps you get access to the selected HTTPS session using a username and password. The login is Admin by default and the password is identical with the device web interface access password.

URL:	https://<IP>/api/v1/login
HTTP method:	PUT
Request body:	{"login":"<username>","password":"<password>"}
Response code:	200 if everything is OK, otherwise 401, 405,...

 **Note**

Response code 403 in the case of default password and its necessary change.

### 2.2.1.2 Logout

The logout helps you log out the currently logged-in session.

URL:	http://<IP>/api/v1/logout
HTTP method:	PUT
Response code:	200 if everything is OK, otherwise 401, 405,...

### 2.2.2 GPIO

- [2.2.2.1 GPIO List Getting](#)
- [2.2.2.2 GPIO Value Setting](#)

#### 2.2.2.1 GPIO List Getting

The list of GPIOs stored in the array container is returned. Name {in1, in2, out1, out2, io1, io2, relay1, relay2} is the input identifier that is subsequently used for GPIO identification in the HTTP API communication too. Direction takes the values {in, out, io}.

URL:	https://<IP>/api/v1/gpio
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors

Response data:

```
{
  "data": {
    "pins": [
      {
        "id": 0,
        "name": "In1",
        "value": 1
      },
      {
        "id": 1,
        "name": "In2",
        "value": 1
      },
      {
        "id": 2,
        "name": "Out1",
        "value": -7
      },
      {
        "id": 3,
        "name": "Out2",
        "value": -7
      },
      {
        "id": 4,
        "name": "IO1",
        "value": 1
      },
      {
        "id": 5,
        "name": "IO2",
        "value": 1
      },
      {
        "id": 6,
        "name": "Relay1",
        "value": 0
      },
      {
        "id": 7,
        "name": "Relay2",
        "value": 0
      }
    ]
  }
}
```



### 2.2.2.2 GPIO Value Setting

Set the output value, switchable IO direction or relay status change.

#### Parameters

**Name** – GPIO name, refer to [2.2.2.1 Getting List of GPIO Inputs and Outputs](#)

**Value** – can be {0, 1}

**⚠ Caution**

The request body must include valid json with one GPIO at least.

URL:	https://{ip_addr}/api/v1/gpio
HTTP method:	PATCH

Request body:

```
{
  "pins": [
    {
      "name": "In1",
      "id": 0,
      "value": 1
    },
    {
      "name": "In2",
      "id": 1,
      "value": 1
    },
    {
      "name": "Out1",
      "id": 2,
      "value": -7
    },
    {
      "name": "Out2",
      "id": 3,
      "value": -7
    },
    {
      "name": "IO1",
      "id": 4,
      "value": 1
    },
    {
      "name": "IO2",
      "id": 5,
      "value": 1
    },
    {
      "name": "Relay1",
      "id": 6,
      "value": 0
    },
    {
      "name": "Relay2",
      "id": 7,
      "value": 0
    }
  ]
}
```

Response code:

Standard HTTP response.

Response data:

```
{
  "data": {
    "pins": [
      {
        "id": 0,
        "name": "In1",
        "value": 1
      },
      {
        "id": 1,
        "name": "In2",
        "value": 1
      },
      {
        "id": 2,
        "name": "Out1",
        "value": -7
      },
      {
        "id": 3,
        "name": "Out2",
        "value": -7
      },
      {
        "id": 4,
        "name": "IO1",
        "value": 1
      },
      {
        "id": 5,
        "name": "IO2",
        "value": 1
      },
      {
        "id": 6,
        "name": "Relay1",
        "value": 0
      },
      {
        "id": 7,
        "name": "Relay2",
        "value": 0
      }
    ]
  }
}
```

## 2.2.3 2N® IP Mobile Control

- [2.2.3.1 Call Setup](#)
- [2.2.3.2 Call Accept](#)
- [2.2.3.3 Call Termination](#)
- [2.2.3.4 Call State](#)
- [2.2.3.5 2N IP Intercom List Getting](#)
- [2.2.3.6 Change of Items in Added 2N IP Intercom List](#)
- [2.2.3.7 Door Lock Activation](#)
- [2.2.3.8 2N® IP Mobile Minimization](#)
- [2.2.3.9 2N® IP Mobile State Info](#)
- [2.2.3.10 2N® IP Mobile Restart](#)

### 2.2.3.1 Call Setup

Set up a call.

URL:	https://{ip_addr}/api/v1/hipm/call/dial?number=<sip_address>
HTTP method:	PUT
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre>{   "data": {     "state": "Ringing"   } }</pre>

### 2.2.3.2 Call Accept

Accepts currently ringing incoming call.

URL:	https://<IP>/api/v1/call/accept
HTTP method:	GET
Request body:	N/A
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	{"data":{"success":true}}

### 2.2.3.3 Call Termination

Terminates the active call.

URL:	https://{{ip_addr}}/api/v1/hipm/call/hangup
HTTP method:	PUT
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre>{   "data": {     "state": "IncomingReceived"   } }</pre>

### 2.2.3.4 Call State

Get the current state of the call processing layer as one of the strings below:

```
{Idle, IncomingReceived, OutgoingInit, OutgoingProgress, OutgoingRinging,
OutgoingEarlyMedia, Connected, StreamsRunning, Pausing, Paused, Resuming, Referred,
Error, CallEnd, PausedByRemote, UpdatedByRemote, IncomingEarlyMedia, Updating,
Released, EarlyUpdatedByRemote, EarlyUpdating}
```

Refer to <http://www.linphone.org> for more information.

URL:	https://{{ip_addr}}/api/v1/hipm/call
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre> {   "data": {     "state": "Released"   } } </pre>

### 2.2.3.5 2N IP Intercom List Getting

Get the list of the **2N IP intercoms** that are registered with **2N® IP Mobile** and can be displayed and edited in the device list.

NNURL:	https://{{ip_addr}}/api/v1/hipm/devices
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.

Response data:	<pre> {   "data": {     "devices": [       {         "id": 255,         "ip": "10.0.24.55",         "isActive": false,         "name": "2N IP Verso",         "serialNumber": "54-0776-0059"       },       {         "id": 256,         "ip": "10.0.24.135",         "isActive": false,         "name": "2N IP Verso",         "serialNumber": "54-1921-5022"       }     ]   } } </pre>
----------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 2.2.3.6 Change of Items in Added 2N IP Intercom List

Modify the list of added 2N IP intercoms.

#### Parameters

**isActive** – true or false

**id** – get the 2N IP intercom ID from the list, refer to [2.2.3.5 Getting List of Available 2N IP Intercoms](#)

URL:	https://{{ip_addr}}/api/v1/hipm/devices
HTTP method:	PATCH
Request body:	<pre> {   "devices": [     {       "isActive": true,       "id": 255     }   ] } </pre>

Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre> {   "data": {     "devices": [       {         "id": 255,         "isActive": true       }     ]   } } </pre>

### 2.2.3.7 Door Lock Activation

Send a request to activate the door lock for a selected **2N IP intercom**. Make sure that the **2N IP intercom** serial number is in the list of added devices.

URL:	https://{{ip_addr}}/api/v1/hipm/devices/open
HTTP method:	PUT
Request body	<pre> {   "switchId":1,   "device":{     "serialNumber":"54-1671-0005"   } } </pre>
Response code:	Standard HTTP response code, treating syntactic errors.



Response data:

```
{
  "data": {
    "id": 0,
    "name": "string",
    "isActive": true,
    "ip": "string",
    "serialNumber": "string"
  }
}
```

**⚠ Caution**

- The 2N IP intercom, whose door lock switch is to be activated must be active according to request [2.2.3.5 Getting List of Available 2N IP Intercoms](#). Switch the 2N IP intercom into the active state using request [2.2.3.6 Change of Items in List of Added Devices](#).

**2.2.3.8 2N® IP Mobile Minimization**

Send the **2N® IP Mobile** application to the background (displaying the **2N® Indoor Touch 2.0** launcher).

URL:	https://<IP>/api/v1/hipm/minimize
HTTP method:	PUT
Request body:	N/A
Response code:	Standard HTTP response code, treating syntactic errors.

### 2.2.3.9 2N® IP Mobile State Info

Returns the information that the application is running and capable of processing communication.

URL:	https://<IP>/api/v1/hipm
HTTP method:	GET
Request body:	N/A
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre> {   "data": {     "isRunning": true   } } </pre>

### 2.2.3.10 2N® IP Mobile Restart

Restart **2N® IP Mobile** if running and start **2N® IP Mobile** if not running.

URL:	https://<IP>/api/v1/hipm/reset
HTTP method:	PUT
Request body:	N/A
Response code:	Standard HTTP response code, treating syntactic errors.

## 2.2.4 System

- [2.2.4.1 System Info and Status](#)
- [2.2.4.2 2N® Indoor Touch Restart](#)

### 2.2.4.1 System Info and Status

As this is a WEB API function, the success value is not included in the response. Information on the system and its state is returned.

URL:	https://<IP>/api/v1/status
HTTP method:	GET
Request body:	N/A
Response code:	Standard HTTP response code, treating syntactic errors.

Response data:

```

{
  "data": {
    "device": {
      "fwVersion": "4.2.0",
      "hwVersion": 1,
      "isNfcCapable": false,
      "isWifiCapable": true,
      "lanMac": "7C:1E:B3:02:FC:12",
      "model": "2N Indoor Touch 2.0",
      "os": "Android OS 6.0.1",
      "sn": "52-2101-0129",
      "temperature": 53,
      "uptimeLife": 7898373,
      "uptimeRestart": 763111,
      "wifiMac": "7C:1E:B3:03:3F:6B"
    },
    "memory": {
      "available": 1024,
      "total": 1760
    },
    "sdCard": {
      "available": 0,
      "isInserted": false,
      "total": 0
    },
    "storage": {
      "available": 3393,
      "total": 3712
    },
    "time": {
      "date": "10/9/19",
      "time": "9:42 AM"
    }
  }
}

```

Meaning of items:

- **device**
  - **fwVersion** – device firmware version
  - **hwVersion** – device hardware version
  - **isNfcCapable** – true if NFC support is available
  - **isWifiCapable** – true if a wi-fi interface is available
  - **lanMac** – Ethernet interface MAC address
  - **model** – device model
  - **os** – operating system version
  - **sn** – device serial number
  - **temperature** – device temperature

- **upTimeLife** – gives the total device operation time
- **upTimeRestart** – gives the device operation time since the last restart
- **wifiMac** – wi-fi interface MAC address
- **memory**
  - **available** – currently available device memory size
  - **total** – total device memory size
- **sdCard**
  - **available** – currently available SD card size
  - **isInserted** – true if an SD card is inserted
  - **total** – total external SD card size
- **storage**
  - **available** – currently available device storage size
  - **total** – total device storage size
- **time**
  - **date** – current device date
  - **time** – current device time

### 2.2.4.2 2N® Indoor Touch Restart

Restart the **2N® Indoor Touch 2.0** unit.

URL:	https://<IP>/api/v1/maintenance/devicereset
HTTP method:	PUT
Request body:	N/A
Response code:	Standard HTTP response code, treating syntactic errors.

## 3. WEB API

WEB API includes a set of endpoints that are used in the **2N® Indoor Touch 2.0** web interface.

- [3.10 Maintenance](#)
- [3.11 Network](#)
- [3.1 Device info](#)
- [3.2 Application](#)
- [3.3 Audio](#)
- [3.4 Display](#)
- [3.5 IP Mobile](#)
- [3.6 Homescreen](#)
- [3.7 License](#)
- [3.8 Localsettings](#)
- [3.9 User and session control](#)

### 3.1 Device info

#### About

Get basic information on the device.

URL:	https://{{ip_addr}}/api/v1/about
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre> {   "data": {     "buildDate": "9/30/19 2:44 AM",     "fwVersion": "4.2.0",     "hwVersion": 1,     "incrementalVersion": "user.28727",     "model": "2N Indoor Touch 2.0",     "serialNumber": "52-2101-0129"   } } </pre>

#### Status

Get information on the device state.

URL:	https://{{ip_addr}}/api/v1/status
------	-----------------------------------

HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.



Response data:

```

{
  "data": {
    "device": {
      "fwVersion": "4.2.0",
      "hwVersion": 1,
      "isNfcCapable": false,
      "isWifiCapable": true,
      "lanMac": "7C:1E:B3:02:FC:12",
      "model": "2N Indoor Touch 2.0",
      "os": "Android OS 6.0.1",
      "sn": "52-2101-0129",
      "temperature": 45,
      "uptimeLife": 8069000,
      "uptimeRestart": 8201,
      "wifiMac": "7C:1E:B3:03:3F:6B"
    },
    "memory": {
      "available": 1208,
      "total": 1760
    },
    "sdCard": {
      "available": 0,
      "isInserted": false,
      "total": 0
    },
    "storage": {
      "available": 3392,
      "total": 3712
    },
    "time": {
      "date": "10/11/19",
      "time": "9:09 AM"
    }
  }
}

```

## 3.2 Application

### Getting current settings

Get the [Application](#) settings.

URL:	https://{{ip_addr}}/api/v1/application
HTTP method:	GET

Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre> {   "data": {     "isInstallationAllowed": false,     "isReturnToApplicationOnCallEndAllowed": false   } } </pre>

### Section configuration

Set the values in the [Application](#) section.

URL:	https://{{ip_addr}}/api/v1/application
HTTP method:	PUT
Response code:	Standard HTTP response code, treating syntactic errors.

Request body:	<pre> {   "applicationToRunAtStart":{     "id":"","     "installDate":"","     "name":"None",     "version":""   },   "isInstallationAllowed":<b>false</b>,   "isReturnToApplicationOnCallEndAllowed":<b>false</b>,   "urlToRunAtStart":"" } </pre>
Response data:	<pre> {   "data": {     "applicationToRunAtStart": {       "id": "",       "installDate": "",       "name": "None",       "version": ""     },     "isInstallationAllowed": <b>false</b>,     "isReturnToApplicationOnCallEndAllowed": <b>false</b>,     "urlToRunAtStart": ""   } } </pre>

### List of available applications

Get a list of available applications.

URL:	https://{{ip_addr}}/api/v1/application/applications
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.



Response data:

```
{
  "data": {
    "applications": [
      {
        "id": "",
        "installDate": "",
        "name": "None",
        "version": ""
      },
      {
        "id": "com.axis.acc",
        "installDate": "1/1/09",
        "name": "Companion 3",
        "version": "3.12.15 (167)"
      },
      {
        "id": "com.axis.acs",
        "installDate": "1/1/09",
        "name": "Camera Station",
        "version": "1.13.18 (144)"
      },
      {
        "id": "com.android.music",
        "installDate": "1/1/09",
        "name": "Music",
        "version": "6.0.1 (23)"
      },
      {
        "id": "com.android.documentsui",
        "installDate": "1/1/09",
        "name": "File Manager",
        "version": "6.0.1"
      },
      {
        "id": "com.android.deskclock",
        "installDate": "1/1/09",
        "name": "Clock",
        "version": "4.1.0 (410)"
      },
      {
        "id": "com.android.gallery3d",
        "installDate": "1/1/09",
        "name": "Gallery",
        "version": "1.1.40030 (40030)"
      },
      {
        "id": "com.android.calculator2",
        "installDate": "1/1/09",
        "name": "Calculator",

```

```

        "version": "6.0.1 (23)"
      },
      {
        "id": "com.adobe.reader",
        "installDate": "1/1/09",
        "name": "Adobe Acrobat",
        "version": "19.0.0.8512 (1900008512)"
      },
      {
        "id": "com.android.browser",
        "installDate": "1/1/09",
        "name": "Browser",
        "version": "6.0.1 (23)"
      },
      {
        "id": "cz.nn.helios_mobile",
        "installDate": "1/1/09",
        "name": "2N® IP Mobile",
        "version": "4.4.6 (474)"
      }
    ]
  }
}

```

### 3.3 Audio

#### Getting current settings

Get the current [Audio](#) settings.

URL:	https://{{ip_addr}}/api/v1/audio
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.

Response data:	<pre>{   "data": {     "alarmsVolume": 86,     "musicVolume": 73,     "ringtoneVolume": 71   } }</pre>
----------------	--------------------------------------------------------------------------------------------------------

**Section setting**

URL:	https://{{ip_addr}}/api/v1/audio
HTTP method:	PUT
Response code:	Standard HTTP response code, treating syntactic errors.
Request body:	<pre>{   "alarmsVolume":5,   "musicVolume":15,   "ringtoneVolume":10 }</pre>

Response data:	<pre> {   "data": {     "alarmsVolume": 0,     "musicVolume": 13,     "ringtoneVolume": 14   } } </pre>
----------------	---------------------------------------------------------------------------------------------------------

### 3.4 Display

#### Getting current settings

Get the current [Display](#) settings.

URL:	https://{{ip_addr}}/api/v1/display
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre> {   "data": {     "brightness": 100,     "isBlueLedNotificationActive": true,     "turnOffDisplayDelay": 2   } } </pre>

#### Section setting

URL:	https://{{ip_addr}}/api/v1/display
HTTP method:	PUT
Response code:	Standard HTTP response code, treating syntactic errors.



Request body:

```
{  
  "brightness": 37,  
  "isBlueLedNotificationActive": true,  
  "turnOffDisplayDelay": 1  
}
```

Response data:	<pre> {   "data": {     "brightness": 37,     "isBlueLedNotificationActive": true,     "turnOffDisplayDelay": 1   } } </pre>
----------------	------------------------------------------------------------------------------------------------------------------------------

### 3.5 IP Mobile

#### Getting device state info

URL:	https://{{ip_addr}}/api/v1/hipmo/status
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre> {   "data": {     "isDndEnabled": false,     "isDoorButtonEnabled": false,     "isExternalRingEnabled": false,     "isMy2nProxyEnabled": false,     "isRecorderEnabled": false,     "isRunning": true,     "isSipProxyEnabled": true,     "proxyState": 1   } } </pre>

#### Application restart

URL:	https://{{ip_addr}}/api/v1/hipmo/reset
HTTP method:	PUT
Response code:	Standard HTTP response code, treating syntactic errors.

- [3.5.1 Device name](#)
- [3.5.2 General](#)
- [3.5.3 Call](#)
- [3.5.4 SIP](#)
- [3.5.5 Recorder](#)
- [3.5.6 Doorbutton](#)
- [3.5.7 Externalring](#)
- [3.5.8 DND](#)
- [3.5.9 My2N](#)
- [3.5.10 Devices](#)

### 3.5.1 Device name

#### Getting device name setting

URL:	https://{{ip_addr}}/api/v1/hipmo
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre>{   "data": {     "deviceName": "idt555"   } }</pre>

#### Device name setting

URL:	https://{{ip_addr}}/api/v1/hipmo
HTTP method:	PUT
Request body:	<pre>{   "deviceName": "&lt;string&gt;" }</pre>
Response code:	Standard HTTP response code, treating syntactic errors.

Response data:	<pre>{   "data": {     "deviceName": "idt555"   } }</pre>
----------------	-----------------------------------------------------------

### 3.5.2 General

#### Getting current settings

Get the current General settings.

URL:	https://{{ip_addr}}/api/v1/hipmo/settings/general
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre>{   "data": {     "autosnapshot": true,     "customLanguage": false,     "deactivationTimeout": 1,     "hipGroupPassword": "",     "isReceivingMulticastActive": false,     "logging": false,     "multicastDiscovery": true,     "pushToTalk": false,     "settingsPassword": ""   } }</pre>

#### Section setting

URL:	https://{{ip_addr}}/api/v1/hipmo/settings/general
HTTP method:	PUT
Response code:	Standard HTTP response code, treating syntactic errors.

Request body:	<pre> {   "hipGroupPassword": "&lt;string&gt;",   "settingsPassword": "&lt;string&gt;",   "autosnapshot": "&lt;boolean&gt;",   "pushToTalk": "&lt;boolean&gt;",   "logging": "&lt;boolean&gt;",   "multicastDiscovery": "&lt;boolean&gt;",   "customLanguage": "&lt;boolean&gt;",   "isReceivingMulticastActive": "&lt;boolean&gt;",   "deactivationTimeout": "&lt;integer&gt;" } </pre>
Response data:	<pre> {   "data": {     "autosnapshot": <b>true</b>,     "customLanguage": <b>false</b>,     "deactivationTimeout": <b>1</b>,     "hipGroupPassword": "",     "isReceivingMulticastActive": <b>false</b>,     "logging": <b>false</b>,     "multicastDiscovery": <b>true</b>,     "pushToTalk": <b>false</b>,     "settingsPassword": ""   } } </pre>

### 3.5.3 Call

#### Getting current call settings

URL:	https://{{ip_addr}}/api/v1/hipmo/settings/call
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.

Response data:	<pre> {   "data": {     "autoCallPickup": <b>false</b>,     "autoCallPickupDelay": 0,     "customRingtone": <b>false</b>,     "isEarlyMediaEnabled": <b>true</b>,     "ringtone": {       "name": "Flutey Phone",       "uri": "content://media/internal/audio/ media/175"     }   } } </pre>
----------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

**Section setting**

URL:	https://{{ip_addr}}/api/v1/hipmo/settings/call
HTTP method:	PUT
Response code:	Standard HTTP response code, treating syntactic errors.
Request body:	<pre> {   "ringtone": {     "uri": "&lt;string&gt;",     "name": "&lt;string&gt;"   },   "customRingtoneData": "&lt;string&gt;",   "autoCallPickup": "&lt;boolean&gt;",   "autoCallPickupDelay": "&lt;integer&gt;",   "isEarlyMediaEnabled": "&lt;boolean&gt;" } </pre>

Response data:

```

{
  "data": {
    "autoCallPickup": false,
    "autoCallPickupDelay": 0,
    "customRingtone": false,
    "isEarlyMediaEnabled": true,
    "ringtone": {
      "name": "Flutey Phone",
      "uri": "content://media/internal/audio/
media/175"
    }
  }
}

```

**Getting list of ringtones**

URL:	https://{{ip_addr}}/api/v1/hipmo/settings/call/ringtones
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.

Response  
data:

```
{
  "data": {
    "ringtones": [
      {
        "name": "None",
        "uri": "NONE"
      },
      {
        "name": "Andromeda",
        "uri": "content://media/internal/audio/media/103"
      },
      ...
      {
        "name": "Zeta",
        "uri": "content://media/internal/audio/media/206"
      }
    ]
  }
}
```



## 3.5.4 SIP

**Getting current settings**

Get the current SIP Proxy server settings.

URL:	https://{{ip_addr}}/api/v1/hipmo/settings/sip
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre> {   "data": {     "ipAddress": "",     "isEnabled": false,     "name": "IndoorTouch",     "password": "",     "port": 5060,     "protocol": 1,     "registrar": {       "expiry": 120,       "ipAddress": "",       "isEnabled": false,       "port": 5060     },     "sipPortRandomization": false,     "sipRandomPort": 5060,     "userName": ""   } } </pre>

**Section setting**

URL:	https://{{ip_addr}}/api/v1/hipmo/settings/sip
HTTP method:	PUT
Response code:	Standard HTTP response code, treating syntactic errors.

Request body:	<pre> {   "name": "&lt;string&gt;",   "userName": "&lt;string&gt;",   "password": "&lt;string&gt;",   "ipAddress": "&lt;string&gt;",   "port": "&lt;integer&gt;",   "protocol": "&lt;integer&gt;",   "isEnabled": "&lt;boolean&gt;",   "sipPortRandomization": "&lt;boolean&gt;",   "sipRandomPort": "&lt;integer&gt;",   "registrar": {     "ipAddress": "&lt;string&gt;",     "isEnabled": "&lt;boolean&gt;",     "expiry": "&lt;integer&gt;",     "port": "&lt;integer&gt;"   } } </pre>
Response data:	<pre> {   "data": {     "ipAddress": "",     "isEnabled": <b>false</b>,     "name": "IndoorTouch",     "password": "",     "port": 5060,     "protocol": 1,     "registrar": {       "expiry": 120,       "ipAddress": "",       "isEnabled": <b>false</b>,       "port": 5060     },     "sipPortRandomization": <b>false</b>,     "sipRandomPort": 5060,     "userName": ""   } } </pre>

### 3.5.5 Recorder

#### Getting current settings

Get the current answering machine settings.

URL:	<a href="https://{{ip_addr}}/api/v1/hipmo/settings/recorder">https://{{ip_addr}}/api/v1/hipmo/settings/recorder</a>
------	---------------------------------------------------------------------------------------------------------------------

HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre> {   "data": {     "activationTimeout": 0,     "announcementType": 0,     "isEnabled": false   } } </pre>

### Section setting

URL:	https://{{ip_addr}}/api/v1/hipmo/settings/recorder
HTTP method:	PUT
Response code:	Standard HTTP response code, treating syntactic errors.
Request body:	<pre> {   "activationTimeout":0,   "announcementType":0,   "isEnabled":true } </pre>
Response data:	<pre> {   "data": {     "activationTimeout": 0,     "announcementType": 0,     "isEnabled": false   } } </pre>

### 3.5.6 Doorbutton

#### Getting current settings

Get the current doorbell button settings.

URL:	https://{{ip_addr}}/api/v1/hipmo/settings/doorbutton
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre> {   "data": {     "bypassDnd": false,     "invertInput": true,     "isEnabled": false,     "maxRingingTime": 30,     "minRingingTime": 3,     "usedInput": 0   } } </pre>

**Section setting**

URL:	https://{{ip_addr}}/api/v1/hipmo/settings/doorbutton
HTTP method:	PUT
Response code:	Standard HTTP response code, treating syntactic errors.
Request body:	<pre> {   "isEnabled": "&lt;boolean&gt;",   "invertInput": "&lt;boolean&gt;",   "bypassDnd": "&lt;boolean&gt;",   "usedInput": "&lt;integer&gt;",   "minRingingTime": "&lt;integer&gt;",   "maxRingingTime": "&lt;integer&gt;" } </pre>

Response data:	<pre> {   "data": {     "bypassDnd": false,     "invertInput": true,     "isEnabled": false,     "maxRingingTime": 30,     "minRingingTime": 3,     "usedInput": 0   } } </pre>
----------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 3.5.7 Externalring

#### Getting current settings

Get the current external ringing notification settings.

URL:	https://{{ip_addr}}/api/v1/hipmo/settings/externalring
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre> {   "data": {     "bypassDnd": false,     "invertOutput": true,     "isEnabled": false,     "maxRingingTime": 30,     "minRingingTime": 3,     "usedOutput": 0   } } </pre>

#### Section setting

URL:	https://{{ip_addr}}/api/v1/hipmo/settings/externalring
HTTP method:	PUT
Response code:	Standard HTTP response code, treating syntactic errors.

Request body:	<pre> {   "isEnabled": "&lt;boolean&gt;",   "invertOutput": "&lt;boolean&gt;",   "bypassDnd": "&lt;boolean&gt;",   "usedOutput": "&lt;integer&gt;",   "minRingTime": "&lt;integer&gt;",   "maxRingTime": "&lt;integer&gt;" } </pre>
Response data:	<pre> {   "data": {     "bypassDnd": <b>false</b>,     "invertOutput": <b>true</b>,     "isEnabled": <b>false</b>,     "maxRingTime": <b>30</b>,     "minRingTime": <b>3</b>,     "usedOutput": <b>0</b>   } } </pre>

### 3.5.8 DND

#### Getting current settings

Get the current DND settings.

URL:	https://{{ip_addr}}/api/v1/hipmo/settings/dnd
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.

Response data:

```

{
  "data": {
    "dnd": {
      "isEnabled": false,
      "time": 0
    },
    "dndNight": {
      "days": [
        5,
        1,
        4,
        2,
        3,
        0,
        6
      ],
      "endTime": "00:00",
      "isEnabled": false,
      "startTime": "00:00"
    }
  }
}

```

**Section setting**

URL:	https://{{ip_addr}}/api/v1/hipmo/settings/dnd
HTTP method:	PUT
Response code:	Standard HTTP response code, treating syntactic errors.

Request body:	<pre>{   "dnd": {     "isEnabled": "&lt;boolean&gt;",     "time": "&lt;integer&gt;"   },   "dndNight": {     "isEnabled": "&lt;boolean&gt;",     "startTime": "&lt;string&gt;",     "endTime": "&lt;string&gt;",     "days": [       "&lt;integer&gt;",       "&lt;integer&gt;"     ]   } }</pre>
Response data:	<pre>{   "data": {     "dnd": {       "isEnabled": <b>false</b>,       "time": 0     },     "dndNight": {       "days": [         5,         1,         4,         2,         3,         0,         6       ],       "endTime": "00:00",       "isEnabled": <b>false</b>,       "startTime": "00:00"     }   } }</pre>

### 3.5.9 My2N

#### Getting current settings

Get the current My2N settings.



URL:	https://{{ip_addr}}/api/v1/hipmo/settings/my2n
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre> {   "data": {     "deviceId": "",     "isEnabled": false,     "password": ""   } } </pre>

**Section setting**

URL:	https://{{ip_addr}}/api/v1/hipmo/settings/my2n
HTTP method:	PUT
Response code:	Standard HTTP response code, treating syntactic errors.
Request body:	<pre> {   "isEnabled": "&lt;boolean&gt;",   "deviceId": "&lt;string&gt;",   "password": "&lt;string&gt;" } </pre>
Response data:	<pre> {   "data": {     "deviceId": "",     "isEnabled": false,     "password": ""   } } </pre>

## 3.5.10 Devices

**Getting list of all devices**

URL:	https://{{ip_addr}}/api/v1/hipmo/devices
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre> {   "data": {     "actual": [],     "lan": [       {         "id": 148,         "ip": "10.27.6.213",         "isActive": false,         "name": "2NIndoorCompact-5223420153"       },       ...       {         "id": 242,         "ip": "10.27.21.10",         "isActive": false,         "name": "2N IP Force"       }     ],     "sip": []   } } </pre>

**Getting list of active devices**

URL:	https://{{ip_addr}}/api/v1/hipmo/devices/actual
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.

Response data:	<pre>{   "data": {     "devices": []   } }</pre>
----------------	--------------------------------------------------

- [3.5.10.1 LAN](#)
- [3.5.10.2 SIP](#)

### 3.5.10.1 LAN

#### Getting list of LAN devices

URL:	https://{{ip_addr}}/api/v1/hipmo/devices/lan
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre>{   "data": {     "devices": [       {         "id": 148,         "ip": "10.27.6.213",         "isActive": false,         "name": "2NIndoorCompact-5223420153"       },       ...       {         "id": 242,         "ip": "10.27.21.10",         "isActive": false,         "name": "2N IP Force"       }     ]   } }</pre>

**Section setting**

URL:	https://{{ip_addr}}/api/v1/hipmo/devices/lan/{{id}}
HTTP method:	PUT
Path variable:	id – device ID
Response code:	Standard HTTP response code, treating syntactic errors.
Request body:	<pre> {   "id": "&lt;integer&gt;",   "name": "&lt;string&gt;",   "isActive": "&lt;boolean&gt;",   "ip": "&lt;string&gt;" } </pre>
Response data:	<pre> {   "data": {     "devices": [       {         "id": 148,         "ip": "10.27.6.213",         "isActive": false,         "name": "2NIndoorCompact-5223420153"       },       ...       {         "id": 242,         "ip": "10.27.21.10",         "isActive": false,         "name": "2N IP Force"       }     ]   } } </pre>

## 3.5.10.2 SIP

**Getting list of SIP devices**

URL:	https://{{ip_addr}}/api/v1/hipmo/devices/sip
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre> {   "data": {     "devices": []   } } </pre>

**Create SIP device**

URL:	https://{{ip_addr}}/api/v1/hipmo/devices/sip
HTTP method:	PUT
Response code:	Standard HTTP response code, treating syntactic errors.
Request body:	<pre> {   "id": "&lt;integer&gt;",   "name": "&lt;string&gt;",   "isActive": "&lt;boolean&gt;",   "isMy2n": "&lt;boolean&gt;",   "sipLine": "&lt;string&gt;",   "dtmf1": "&lt;string&gt;",   "dtmf2": "&lt;string&gt;",   "dtmf3": "&lt;string&gt;",   "dtmf4": "&lt;string&gt;" } </pre>

Response data:	<pre> {   "data":{     "devices":[       "device":{         "id":"&lt;integer&gt;",         "name":"&lt;string&gt;",         "isActive":"&lt;boolean&gt;",         "isMy2n":"&lt;boolean&gt;",         "sipLine":"&lt;string&gt;",         "dtmf1":"&lt;string&gt;",         "dtmf2":"&lt;string&gt;",         "dtmf3":"&lt;string&gt;",         "dtmf4":"&lt;string&gt;"       }     ]   } } </pre>
----------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

**SIP device update**

URL:	https://{{ip_addr}}/api/v1/hipmo/devices/sip/{{id}}
HTTP method:	PUT
Path variable:	id – device ID
Request body:	<pre> {   "id": "&lt;integer&gt;",   "name": "&lt;string&gt;",   "isActive": "&lt;boolean&gt;",   "isMy2n": "&lt;boolean&gt;",   "sipLine": "&lt;string&gt;",   "dtmf1": "&lt;string&gt;",   "dtmf2": "&lt;string&gt;",   "dtmf3": "&lt;string&gt;",   "dtmf4": "&lt;string&gt;" } </pre>
Response code:	Standard HTTP response code, treating syntactic errors.

Response data:	<pre> {   "data":{     "devices":[       "device":      {         "id":"&lt;integer&gt;",         "name":"&lt;string&gt;",         "isActive":"&lt;boolean&gt;",         "isMy2n":"&lt;boolean&gt;",         "sipLine":"&lt;string&gt;",         "dtmf1":"&lt;string&gt;",         "dtmf2":"&lt;string&gt;",         "dtmf3":"&lt;string&gt;",         "dtmf4":"&lt;string&gt;"       }     ]   } } </pre>
----------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

**Delete device**

URL:	https://{{ip_addr}}/api/v1//hipmo/devices/sip/{{id}}
HTTP method:	DELETE
Path variable:	id – device ID
Response code:	Standard HTTP response code, treating syntactic errors.
Request body:	<pre> {   "id": "&lt;integer&gt;",   "name": "&lt;string&gt;",   "isActive": "&lt;boolean&gt;",   "isMy2n": "&lt;boolean&gt;",   "sipLine": "&lt;string&gt;",   "dtmf1": "&lt;string&gt;",   "dtmf2": "&lt;string&gt;",   "dtmf3": "&lt;string&gt;",   "dtmf4": "&lt;string&gt;" } </pre>

**3.6 Homescreen****Getting current settings**

Get the current [Homepage](#) settings.

URL:	https://{{ip_addr}}/api/v1/homescreen
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.



Response data:

```
{
  "data": {
    "buttons": [
      {
        "application": {
          "id": "com.android.documentsui",
          "name": "File Manager"
        },
        "icon": 6,
        "id": 1,
        "label": "File Manager",
        "position": 1
      },
      {
        "application": {
          "id": "com.android.browser",
          "name": "Browser"
        },
        "icon": 3,
        "id": 2,
        "label": "Browser",
        "position": 2
      },
      {
        "application": {
          "id": "cz.nn.helios_mobile",
          "name": "2N® IP Mobile"
        },
        "icon": 8,
        "id": 3,
        "label": "Intercoms",
        "position": 3
      },
      {
        "application": {
          "id": "com.android.gallery3d",
          "name": "Gallery"
        },
        "icon": 7,
        "id": 4,
        "label": "Gallery",
        "position": 4
      },
      {
        "application": {
          "id": "com.axis.acs",
          "name": "Camera Station"
        },
        "icon": 0,

```

```

        "id": 5,
        "position": 5
      }
    ],
    "isClockOnlyModeActive": false
  }
}

```

**Section setting**

URL:	https://{ip_addr}/api/v1/homescreen
HTTP method:	PUT
Request body:	<pre> {   "isClockOnlyModeActive": "&lt;boolean&gt;" } </pre>
Response code:	Standard HTTP response code, treating syntactic errors.

Response data:	<pre>{   "data": {     "isClockOnlyModeActive": <b>false</b>   } }</pre>
----------------	--------------------------------------------------------------------------

**Get a list of applications available for the Homepage.**

URL:	https://{ip_addr}/api/v1/homescreen/applications
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.

Response data:

```
{
  "data": {
    "applications": [
      {
        "id": "com.axis.acc",
        "installDate": "1/1/09",
        "name": "Companion 3",
        "version": "3.12.15 (167)"
      },
      {
        "id": "com.axis.acs",
        "installDate": "1/1/09",
        "name": "Camera Station",
        "version": "1.13.18 (144)"
      },
      {
        "id": "com.android.music",
        "installDate": "1/1/09",
        "name": "Music",
        "version": "6.0.1 (23)"
      },
      {
        "id": "com.android.documentsui",
        "installDate": "1/1/09",
        "name": "File Manager",
        "version": "6.0.1"
      },
      {
        "id": "com.android.deskclock",
        "installDate": "1/1/09",
        "name": "Clock",
        "version": "4.1.0 (410)"
      },
      {
        "id": "com.android.gallery3d",
        "installDate": "1/1/09",
        "name": "Gallery",
        "version": "1.1.40030 (40030)"
      },
      {
        "id": "com.android.calculator2",
        "installDate": "1/1/09",
        "name": "Calculator",
        "version": "6.0.1 (23)"
      },
      {
        "id": "com.adobe.reader",
        "installDate": "1/1/09",
        "name": "Adobe Acrobat",
```

```

        "version": "19.0.0.8512 (1900008512)"
    },
    {
        "id": "com.android.browser",
        "installDate": "1/1/09",
        "name": "Browser",
        "version": "6.0.1 (23)"
    },
    {
        "id": "cz.nn.helios_mobile",
        "installDate": "1/1/09",
        "name": "2N® IP Mobile",
        "version": "4.4.6 (474)"
    }
]
}

```

### Restore the Homepage default setting

URL:	https://{{ip_addr}}/api/v1/homescreen/layoutreset
HTTP method:	PUT
Response code:	Standard HTTP response code, treating syntactic errors.

Response data:

```
{
  "data": {
    "buttons": [
      {
        "application": {
          "id": "com.android.documentsui",
          "name": "File Manager"
        },
        "icon": 6,
        "id": 1,
        "label": "File Manager",
        "position": 1
      },
      {
        "application": {
          "id": "com.android.browser",
          "name": "Browser"
        },
        "icon": 3,
        "id": 2,
        "label": "Browser",
        "position": 2
      },
      {
        "application": {
          "id": "cz.nn.helios_mobile",
          "name": "2N® IP Mobile"
        },
        "icon": 8,
        "id": 3,
        "label": "Intercoms",
        "position": 3
      },
      {
        "application": {
          "id": "com.android.gallery3d",
          "name": "Gallery"
        },
        "icon": 7,
        "id": 4,
        "label": "Gallery",
        "position": 4
      }
    ],
    "isClockOnlyModeActive": false
  }
}
```

- [3.6.1 Buttons](#)

### 3.6.1 Buttons

#### Getting button list

Get the current list of available buttons (programmable positions for application start).

URL:	https://{{ip_addr}}/api/v1/homescreen/buttons
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.

Response data:

```
{
  "data": {
    "buttons": [
      {
        "application": {
          "id": "com.android.documentsui",
          "name": "File Manager"
        },
        "icon": 6,
        "id": 1,
        "label": "File Manager",
        "position": 1
      },
      {
        "application": {
          "id": "com.android.browser",
          "name": "Browser"
        },
        "icon": 3,
        "id": 2,
        "label": "Browser",
        "position": 2
      },
      {
        "application": {
          "id": "cz.nn.helios_mobile",
          "name": "2N® IP Mobile"
        },
        "icon": 8,
        "id": 3,
        "label": "Intercoms",
        "position": 3
      },
      {
        "application": {
          "id": "com.android.gallery3d",
          "name": "Gallery"
        },
        "icon": 7,
        "id": 4,
        "label": "Gallery",
        "position": 4
      }
    ]
  }
}
```

**Create a new button**



URL:	https://{{ip_addr}}/api/v1/homescreen/buttons
HTTP method:	POST

Request body:	<pre> {   "id": "&lt;integer&gt;",   "position": "&lt;integer&gt;",   "label": "&lt;string&gt;",   "icon": "&lt;integer&gt;",   "application": {     "id": "&lt;string&gt;",     "name": "&lt;string&gt;",     "version": "&lt;string&gt;",     "installDate": "&lt;string&gt;"   } } </pre>
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre> {   "data": {     "application": {       "id": "com.android.gallery3d",       "installDate": "1/1/09",       "name": "Gallery",       "version": "1.1.40030 (40030)"     },     "icon": 7,     "id": 6,     "label": "test",     "position": 5   } } </pre>

- [3.6.1.1 Button](#)

### 3.6.1.1 Button

#### Get the selected button setting

URL:	<code>https://{{ip_addr}}/api/v1/homescreen/buttons/{{id}}</code>
HTTP method:	GET
Path variable:	id – button ID, refer to request <a href="#">3.6.1 Buttons</a>

Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre> {   "data": {     "application": {       "id": "com.android.documentsui",       "name": "File Manager"     },     "icon": 6,     "id": 1,     "label": "File Manager",     "position": 1   } } </pre>

**Set the selected button**

URL:	https://{{ip_addr}}/api/v1/homescreen/buttons/{{id}}
HTTP method:	PUT
Path variable:	id – button ID, refer to request <a href="#">3.6.1 Buttons</a>
Request body:	<pre> {   "id": "&lt;integer&gt;",   "position": "&lt;integer&gt;",   "label": "&lt;string&gt;",   "icon": "&lt;integer&gt;",   "application": {     "id": "&lt;string&gt;",     "name": "&lt;string&gt;",     "version": "&lt;string&gt;",     "installDate": "&lt;string&gt;"   } } </pre>
Response code:	Standard HTTP response code, treating syntactic errors.

Response data:	<pre> {   "data": {     "application": {       "id": "com.android.gallery3d",       "installDate": "1/1/09",       "name": "Gallery",       "version": "1.1.40030 (40030)"     },     "icon": 7,     "id": 4,     "label": "test",     "position": 5   } } </pre>
----------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

**Remove button**

URL:	https://{ip_addr}/api/v1/homescreen/buttons/{id}
HTTP method:	DELETE
Path variable:	id – button ID, refer to request <a href="#">3.6.1 Buttons</a>
Request body:	<pre> {   "id": "&lt;integer&gt;",   "position": "&lt;integer&gt;",   "label": "&lt;string&gt;",   "icon": "&lt;integer&gt;",   "application": {     "id": "&lt;string&gt;",     "name": "&lt;string&gt;",     "version": "&lt;string&gt;",     "installDate": "&lt;string&gt;"   } } </pre>
Response code:	Standard HTTP response code, treating syntactic errors.

### 3.7 License

**Getting current settings**

Get the current license settings.

URL:	https://{{ip_addr}}/api/v1/license
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre> {   "data": {     "features": {       "isAppInstalationAllowed": true,       "isHttpApiAllowed": true,       "isStockLauncherAllowed": false     },     "key": "EJF3A-7SGBQ-UTGYC-VWV38-MKPGG",     "remainingHours": 0,     "state": 2   } } </pre>

**Section setting**

URL:	https://{{ip_addr}}/api/v1/license
HTTP method:	POST
Response code:	Standard HTTP response code, treating syntactic errors.

Request body:	<pre> {   "key": "&lt;string&gt;",   "remainingHours": "&lt;integer&gt;",   "state": "&lt;integer&gt;",   "features": {     "isHttpApiAllowed": "&lt;boolean&gt;",     "isStockLauncherAllowed": "&lt;boolean&gt;",     "isAppInstalationAllowed": "&lt;boolean&gt;"   } } </pre>
Response data:	<pre> {   "data": {     "features": {       "isAppInstalationAllowed": true,       "isHttpApiAllowed": true,       "isStockLauncherAllowed": false     },     "key": "EJF3A-7SGBQ-UTGYC-VWV38-MKPGG",     "remainingHours": 0,     "state": 2   } } </pre>

### 3.8 Localsettings

#### Getting current settings

Get the current [Local settings](#) values.

URL:	https://{{ip_addr}}/api/v1/localsettings
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.

Response data:

```

{
  "data": {
    "language": {
      "id": "en-US",
      "name": "English (United States)"
    },
    "time": {
      "dateTime": "10/11/19 8:46 AM",
      "is24hourFormatUsed": false,
      "isNetworkTimeUsed": true,
      "isoDateTime": "2019-10-11T08:46:27+0000",
      "timeZone": {
        "id": 27,
        "name": "GMT +00:00 Coordinated
universal time (GMT)"
      }
    },
    "weather": {
      "location": "Prague,CZ",
      "source": "www.openweathermap.org",
      "unit": 0,
      "updatePeriod": 0
    }
  }
}

```

**Section setting**

URL:	https://{{ip_addr}}/api/v1/localsettings
HTTP method:	PUT
Response code:	Standard HTTP response code, treating syntactic errors.

Request body:

```
{
  "language": {
    "id": "<string>",
    "name": "<string>"
  },
  "time": {
    "timeZone": {
      "id": "<integer>",
      "name": "<string>"
    },
    "isNetworkTimeUsed": "<boolean>",
    "dateTime": "<string>",
    "isoDateTime": "<string>",
    "is24hourFormatUsed": "<boolean>"
  },
  "weather": {
    "location": "<string>",
    "source": "<string>",
    "unit": "<integer>",
    "updatePeriod": "<integer>"
  }
}
```



Response data:

```

{
  "data": {
    "language": {
      "id": "en-US",
      "name": "English (United States)"
    },
    "time": {
      "dateTime": "10/15/19 11:54 AM",
      "is24hourFormatUsed": false,
      "isNetworkTimeUsed": true,
      "isoDateTime": "2019-10-15T11:54:54+0000",
      "timeZone": {
        "id": 27,
        "name": "GMT +00:00 Coordinated
universal time (GMT)"
      }
    },
    "weather": {
      "location": "Prague,CZ",
      "source": "www.openweathermap.org",
      "unit": 0,
      "updatePeriod": 0
    }
  }
}

```

**Getting list of available languages**

URL:	https://{ip_addr}/api/v1/localsettings/languages
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.

Response data:

```

{
  "data": {
    "languages": [
      {
        "id": "af-ZA",
        "name": "Afrikaans"
      },
      {
        "id": "az-AZ",
        "name": "Azərbaycan"
      },
      ...
      {
        "id": "ja-JP",
        "name": "日本語"
      }
    ]
  }
}

```

**Getting list of available time zones**

URL:	https://{{ip_addr}}/api/v1/localsettings/timezones
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.

Response data:

```

{
  "data": {
    "timezones": [
      {
        "id": 0,
        "name": "GMT -11:00 Midway Island
(Pacific/Midway)"
      },
      {
        "id": 1,
        "name": "GMT -10:00 Hawaii (Pacific/
Honolulu)"
      },
      ...
      {
        "id": 84,
        "name": "GMT +13:00 Tonga (Pacific/
Tongatapu)"
      }
    ]
  }
}

```

### 3.9 User and session control

#### User login

URL:	https://{ip_addr}/api/v1/login
HTTP method:	PUT
Response code:	Standard HTTP response code, treating syntactic errors.
Request body:	<pre> {   "login": "admin",   "password": "2n" } </pre>

#### User logout

URL:	https://{ip_addr}/api/v1/logout
------	---------------------------------

HTTP method:	PUT
Response code:	Standard HTTP response code, treating syntactic errors.

**User settings**

URL:	https://{{ip_addr}}/api/v1/user
HTTP method:	PUT
Request body:	<pre>{   "oldPassword": "2n",   "newPassword": "NewPass123" }</pre>
Response code:	Standard HTTP response code, treating syntactic errors.

Request data:	<pre>{   "data": {     "newPassword": "NewPass123",     "oldPassword": "2n"   } }</pre>
---------------	-----------------------------------------------------------------------------------------

### 3.10 Maintenance

#### Getting current settings

Get the current [System management](#) settings.

URL:	https://{{ip_addr}}/api/v1/maintenance
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre>{   "data": {     "isRemoteEnabled": true,     "isWelcomeScreenEnabled": false   } }</pre>

#### Section setting

URL:	https://{{ip_addr}}/api/v1/maintenance
HTTP method:	PUT
Request body:	<pre>{   "isWelcomeScreenEnabled": "&lt;boolean&gt;",   "isRemoteEnabled": "&lt;boolean&gt;" }</pre>
Response code:	Standard HTTP response code, treating syntactic errors.

Response data:	<pre> {   "data": {     "isRemoteEnabled": true,     "isWelcomeScreenEnabled": false   } } </pre>
----------------	---------------------------------------------------------------------------------------------------

**Factory default**

URL:	https://{{ip_addr}}/api/v1/maintenance/factoryreset
HTTP method:	PUT
Response code:	Standard HTTP response code, treating syntactic errors.

**Device restart**

URL:	https://{{ip_addr}}/api/v1/maintenance/devicereset
HTTP method:	PUT
Response code:	Standard HTTP response code, treating syntactic errors.

- [3.10.1 Config](#)
- [3.10.2 Upgrade](#)
- [3.10.3 Logs](#)
- [3.10.4 Security](#)

**3.10.1 Config****Download current configuration**

Download the current device configuration file.

URL:	https://{{ip_addr}}/api/v1/maintenance/config
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.



## Response data:

```

{
  "application": {
    "applicationToRunAtStart": {
      "id": "",
      "installDate": "",
      "name": "None",
      "version": ""
    },
    "isInstallationAllowed": false,
    "isReturnToApplicationOnCallEndAllowed": false,
    "urlToRunAtStart": ""
  },
  "audio": {
    "alarmsVolume": 0,
    "musicVolume": 13,
    "ringtoneVolume": 14
  },
  "display": {
    "brightness": 37,
    "isBlueLedNotificationActive": true,
    "turnOffDisplayDelay": 1
  },
  "fwVersion": "4.2.0",
  "hipmo": {
    "call": {
      "autoCallPickup": false,
      "autoCallPickupDelay": 0,
      "ringtoneUri": "content://media/internal/audio/
media/175"
    },
    "dndMode": {
      "dnd": {
        "isEnabled": false,
        "time": 0
      },
      "dndNight": {
        "days": [
          5,
          1,
          4,
          2,
          3,
          0,
          6
        ],
        "endTime": "00:00",
        "isEnabled": false,
        "startTime": "00:00"
      }
    }
  }
}

```



```
},
"doorButton": {
  "bypassDnd": false,
  "invertInput": true,
  "isEnabled": false,
  "maxRingingTime": 30,
  "minRingingTime": 3,
  "usedInput": 0
},
"externalRing": {
  "bypassDnd": false,
  "invertOutput": true,
  "isEnabled": false,
  "maxRingingTime": 30,
  "minRingingTime": 3,
  "usedOutput": 0
},
"general": {
  "autosnapshot": true,
  "customLanguage": false,
  "deactivationTimeout": 1,
  "hipGroupPassword": "",
  "isReceivingMulticastActive": false,
  "logging": false,
  "multicastDiscovery": true,
  "pushToTalk": false,
  "settingsPassword": ""
},
"my2n": {
  "deviceId": "",
  "isEnabled": false,
  "password": ""
},
"recorder": {
  "activationTimeout": 0,
  "announcementType": 0,
  "isEnabled": false
},
"sipProxy": {
  "ipAddress": "",
  "isEnabled": false,
  "name": "IndoorTouch",
  "password": "",
  "port": 5060,
  "protocol": 1,
  "registrar": {
    "expiry": 120,
    "ipAddress": "",
    "isEnabled": false,
    "port": 5060
  },
  "sipPortRandomization": false,
```

```

        "sipRandomPort": 5060,
        "userName": "111"
    }
},
"homescreen": {
    "buttons": [
        {
            "application": {
                "id": "com.android.documentsui",
                "name": "File Manager"
            },
            "icon": 6,
            "id": 1,
            "label": "File Manager",
            "position": 1
        },
        {
            "application": {
                "id": "com.android.browser",
                "name": "Browser"
            },
            "icon": 3,
            "id": 2,
            "label": "Browser",
            "position": 2
        },
        {
            "application": {
                "id": "cz.nn.helios_mobile",
                "name": "2N® IP Mobile"
            },
            "icon": 8,
            "id": 3,
            "label": "Intercoms",
            "position": 3
        },
        {
            "application": {
                "id": "com.android.gallery3d",
                "name": "Gallery"
            },
            "icon": 7,
            "id": 6,
            "label": "test",
            "position": 5
        }
    ],
    "isClockOnlyModeActive": false
},
"licence": {
    "key": "EJF3A-7SGBQ-UTGYC-VWV38-MKPGG"
},

```

```

"localsettings": {
  "languageId": "en-US",
  "time": {
    "dateTime": "10/15/19 12:47 PM",
    "is24hourFormatUsed": false,
    "isNetworkTimeUsed": true,
    "isoDateTime": "2019-10-15T12:47:50+0000",
    "timeZoneId": 27
  },
  "weather": {
    "location": "Prague,CZ",
    "unit": 0,
    "updatePeriod": 0
  }
},
"maintenance": {
  "isRemoteEnabled": true,
  "isWelcomeScreenEnabled": false
},
"network": {
  "dns": "10.0.100.101",
  "gateway": "10.27.0.1",
  "ipAddress": "10.27.5.173",
  "isDhcpEnabled": true,
  "isEthernetEnabled": true,
  "mask": "255.255.0.0"
}
}

```

### Upload configuration backup

URL:	https://{{ip_addr}}/api/v1/maintenance/config
HTTP method:	POST
Request body:	form-data: <ul style="list-style-type: none"> <li>• file=&lt;binary&gt;</li> <li>• isNetworkSettingsEnabled=&lt;boolean&gt;</li> </ul>
Response code:	Standard HTTP response code, treating syntactic errors.

### 3.10.2 Upgrade

#### Upgrade state info

URL:	https://{{ip_addr}}/api/v1/maintenance/upgrade
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre> {   "data": {     "fwVersion": "4.2.0",     "isDowngrade": false,     "status": 0,     "wipeData": false   } } </pre>

**Upload firmware**

URL:	https://{{ip_addr}}/api/v1/maintenance/upgrade/upload
HTTP method:	PUT
Request body:	form-data: <ul style="list-style-type: none"> <li>• file=&lt;binary&gt;</li> </ul>
Response code:	Standard HTTP response code, treating syntactic errors.

Response data:	<pre> {   "data": {     "fwVersion": "4.2.0",     "isDowngrade": false,     "status": 1,     "wipeData": false   } } </pre>
----------------	-----------------------------------------------------------------------------------------------------------------------------

**Start upgrade**

URL:	https://{{ip_addr}}/api/v1/maintenance/upgrade/start
HTTP method:	PUT
Request body:	<pre> {"wipeData": true} </pre>
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre> {   "data": {     "fwVersion": "4.2.0",     "isDowngrade": false,     "status": 2,     "wipeData": true   } } </pre>

**Cancel upgrade**

URL:	https://{{ip_addr}}/api/v1/maintenance/upgrade/cancel
HTTP method:	PUT
Response code:	Standard HTTP response code, treating syntactic errors.

Response data:	<pre> {   "data": {     "fwVersion": "4.2.0",     "isDowngrade": false,     "status": 0,     "wipeData": true   } } </pre>
----------------	----------------------------------------------------------------------------------------------------------------------------

### 3.10.3 Logs

#### Getting basic log archives

URL:	https://{{ip_addr}}/api/v1/maintenance/logs/simple
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	Log archive file

#### Getting all device log archives

URL:	https://{{ip_addr}}/api/v1//maintenance/logs/full
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	Log archive file

### 3.10.4 Security

#### Getting access setting info

Getting information on [System settings](#) access settings.

URL:	https://{{ip_addr}}/api/v1/maintenance/security
------	-------------------------------------------------

HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre>{   "data": {     "useWebPassword": true   } }</pre>

### Access setting

Set different access passwords for the web interface and system settings.

URL:	https://{{ip_addr}}/api/v1/maintenance/security
HTTP method:	PUT
Request body:	<pre>{   "useWebPassword": true }</pre>
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre>{   "data": {     "useWebPassword": true   } }</pre>

### Getting System settings access settings

URL:	https://{{ip_addr}}/api/v1/maintenance/security/settings/access
HTTP method:	GET

Response code:	Standard HTTP response code, treating syntactic errors.
----------------	---------------------------------------------------------



Response data:	<pre>{   "data": {     "accessSettingsPasswordType": 1   } }</pre>
----------------	--------------------------------------------------------------------

**Set the System settings access**

URL:	https://{ip_addr}/api/v1/maintenance/security/settings/access
HTTP method:	PUT
Request body:	<pre>{   "accessSettingsPasswordType": "&lt;integer&gt;",   "password": "&lt;string&gt;" }</pre>
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre>{   "data": {     "accessSettingsPasswordType": 1,     "password": "test"   } }</pre>

**Set the administrator access password**

URL:	https://{ip_addr}/api/v1/maintenance/security/settings/admin
HTTP method:	PUT

Request body:	<pre>{   "password": "&lt;string&gt;" }</pre> <div style="border: 1px solid orange; padding: 5px; margin-top: 10px;"> <p><b>⚠ Caution</b></p> <p>Make sure that the password meets the minimum password setting requirements, refer to <a href="#">Security</a>.</p> </div>
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre>{   "data": {     "password": "Test12345"   } }</pre>

- [3.10.4.1 Lockscreen](#)

### 3.10.4.1 Lockscreen

#### Getting current settings

Getting information on screen lock settings

URL:	<code>https://{{ip_addr}}/api/v1/maintenance/security/lockscreen</code>
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre>{   "data": {     "isEnabled": <b>false</b>   } }</pre>

**Change of current settings****⚠ Caution**

You can only turn the lock screen off/on using this endpoint. The lock screen cannot be configured via a web interface.

URL:	https://{ip_addr}/api/v1/maintenance/security/lockscreen
HTTP method:	PUT
Request body:	<pre>{   "isEnabled": "&lt;boolean&gt;" }</pre>
Response code:	Standard HTTP response code, treating syntactic errors.
Response data:	<pre>{   "data": {     "isEnabled": <b>false</b>   } }</pre>

**3.11 Network****Getting current settings**

Get the current [Network](#) settings.

URL:	https://{ip_addr}/api/v1/network
HTTP method:	GET
Response code:	Standard HTTP response code, treating syntactic errors.

Response data:	<pre> {   "data": {     "dns": "8.8.8.8",     "gateway": "192.168.1.1",     "ipAddress": "192.168.1.173",     "isDhcpEnabled": true,     "isEthernetEnabled": true,     "mask": "255.255.255.0"   } } </pre>
----------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

**Section setting**

URL:	https://{{ip_addr}}/api/v1/network
HTTP method:	PUT
Response code:	Standard HTTP response code, treating syntactic errors.
Request body:	<pre> {   "isDhcpEnabled": "&lt;boolean&gt;",   "isEthernetEnabled": "&lt;boolean&gt;",   "ipAddress": "&lt;string&gt;",   "dns": "&lt;string&gt;",   "gateway": "&lt;string&gt;",   "mask": "&lt;string&gt;" } </pre>
Response data:	<pre> {   "data": {     "dns": "8.8.8.8",     "gateway": "192.168.1.1",     "ipAddress": "192.168.1.173",     "isDhcpEnabled": true,     "isEthernetEnabled": true,     "mask": "255.255.255.0"   } } </pre>

## 4. Supplementary Information

- [4.1 Directives, Laws and Regulations – General Instructions and Cautions](#)

### 4.1 Directives, Laws and Regulations – General Instructions and Cautions

Please read this User Manual carefully before using the product. Follow all instructions and recommendations included herein.

Any use of the product that is in contradiction with the instructions provided herein may result in malfunction, damage or destruction of the product.

The manufacturer shall not be liable and responsible for any damage incurred as a result of a use of the product other than that included herein, namely undue application and disobedience of the recommendations and warnings in contradiction herewith.

Any use or connection of the product other than those included herein shall be considered undue and the manufacturer shall not be liable for any consequences arisen as a result of such misconduct.

Moreover, the manufacturer shall not be liable for any damage or destruction of the product incurred as a result of misplacement, incompetent installation and/or undue operation and use of the product in contradiction herewith.

The manufacturer assumes no responsibility for any malfunction, damage or destruction of the product caused by incompetent replacement of parts or due to the use of reproduction parts or components.

The manufacturer shall not be liable and responsible for any loss or damage incurred as a result of a natural disaster or any other unfavourable natural condition.

The manufacturer shall not be held liable for any damage of the product arising during the shipping thereof.

The manufacturer shall not make any warrant with regard to data loss or damage.

The manufacturer shall not be liable and responsible for any direct or indirect damage incurred as a result of a use of the product in contradiction herewith or a failure of the product due to a use in contradiction herewith.

All applicable legal regulations concerning the product installation and use as well as provisions of technical standards on electric installations have to be obeyed. The manufacturer shall not be liable and responsible for damage or destruction of the product or damage incurred by the consumer in case the product is used and handled contrary to the said regulations and provisions.

The consumer shall, at its own expense, obtain software protection of the product. The manufacturer shall not be held liable and responsible for any damage incurred as a result of the use of deficient or substandard security software.

The consumer shall, without delay, change the access password for the product after installation. The manufacturer shall not be held liable or responsible for any damage incurred by the consumer in connection with the use of the original password.

The manufacturer also assumes no responsibility for additional costs incurred by the consumer as a result of making calls using a line with an increased tariff.

### Electric Waste and Used Battery Pack Handling



Do not place used electric devices and battery packs into municipal waste containers. An undue disposal thereof might impair the environment!

Deliver your expired electric appliances and battery packs removed from them to dedicated dumpsites or containers or give them back to the dealer or manufacturer for environmental-friendly disposal. The dealer or manufacturer shall take the product back free of charge and without requiring another purchase. Make sure that the devices to be disposed of are complete.

Do not throw battery packs into fire. Battery packs may not be taken into parts or short-circuited either.

