Thermometer APIs

Approved Version 1.0 – 24 Jul 2018

Open Mobile Alliance

OMA-TS-Thermometer_APIs-V1_0-20180724-A

Contents

Figures

Tables

No table of figures entries found.

1. Scope

Body temperature is one of the essential vital signs of health measurements. This specification defines Thermometer Device Web base APIs, “Thermometer APIs” in short in what follows. Thermometer APIs expose data collected from external thermometers to web base applications throught the interworking of Extension Plug-In with GotAPI. Thermometer APIs expose interfaces to the applications in the GotAPI 1.1 [GotAPI1.1] framework through the GotAPI Server. With Extension Plug-Ins, smartphone applications can interact with various kinds of thermometers using the consistent APIs specified in this specification. The APIs offer the standardization of interface and data.

This is the technical specification part of the thermometer Device Web APIs whose requirements and architecture are defined in a separate document [DWAPI-PCH].

2. References

2.1 Normative References

[DWAPI-PCH] Device WebAPI-PCH
OMA-ER-Device_WebAPIs-V1_0-20160419-C URL:http://www.openmobilealliance.org/

[EventSource] "Server-Sent Events", Worldwide Web Consortium (W3C), URL:http://dev.w3.org/html5/eventsource/ (latest working draft)

[GotAPI 1.1] Generic Open Terminal API Framework (GotAPI), Candidate Version 1.1 – 15 Dec 2015 URL:http://www.openmobilealliance.org/

[HTTP/1.1] "Hypertext Transfer Protocol -- HTTP/1.1", Internet Engineering Task Force (IETF), URL:http://tools.ietf.org/search/rfc2616

[HTTP/2.0] "Hypertext Transfer Protocol version 2.0", Internet Engineering Task Force (IETF), URL:http://tools.ietf.org/search/draft-ietf-httpbis-http2-09 (latest working draft)

[JSON-RPC] "JSON-RPC 2.0 Specification", JSON-RPC Working Group, URL:http://www.jsonrpc.org/specification

[RFC2119] "Key words for use in RFCs to Indicate Requirement Levels", S. Bradner, March 1997, URL:http://www.ietf.org/rfc/rfc2119.txt

[SCRRULES] "SCR Rules and Procedures", Open Mobile Alliance™, OMA-ORG-SCR_Rules_and_Procedures, URL:http://www.openmobilealliance.org/

[WebSocket] "The WebSocket API, Worldwide Web Consortium (W3C), URL:http://dev.w3.org/html5/websockets/ (latest working draft)

2.2 Informative References

[OMADICT] "Dictionary for OMA Specifications", Version 2.9, Open Mobile Alliance™, OMA-ORG-Dictionary-V2.9, URL:http://www.openmobilealliance.org/

[OMNA] "OMA Naming Authority". Open Mobile Alliance™. URL:http://www.openmobilealliance.org/tech/omna.aspx

3. Terminology and Conventions

3.1 Conventions

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in [RFC2119].

All sections and appendixes, except “Scope” and “Introduction”, are normative, unless they are explicitly indicated to be informative.

Agent	A node that collects and transmits personal health data to an associated manager.
API Patterns	Design guidelines and requirements for definition of APIs
Browser Context	Web applications executing under a Web browser as Web runtime environment.
Datagram	An API providing access to UDP protocol based networking.
Device	A physical device implementing either an Agent or manager role.
ECMAScript	Use definition from [OMADICT].
Hybrid Native/Web App	An application designed to execute under the native OS / middleware environment of a device, and that use native APIs for the execution of web content in addition to native code.
JavaScript	Use definition from [OMADICT].
Manager	A node receiving data from one or more agent systems. Examples of managers include a cellular phone, health appliance, set top box, or computer system.
Native App	An application designed to execute under the native OS / middleware environment of a device.
Personal Health Device	A device used in personal health applications.
Socket	An API providing access to TCP protocol based networking.
Uniform Resource Identifier	Use definition from [OMADICT].
User Agent	Use definition from [OMADICT].
Web	The World Wide Web, a content and application framework based upon hypertext and related technologies, e.g. XML, JavaScript/ECMAScript, CSS, etc.
Web Application	An application designed using Web technologies (e.g. HTML, CSS, and Javascript).
Web IDL	An IDL language for Web application APIs
Web Runtime Application	A client-side Web application that is executed in Web runtime environments.
Web Runtime Environment	Client software that supports the execution of Web applications (e.g. browsers or widget engines).
WebSocket	An API providing networking services per the WebSocket standard [WebSocket].
Widget Context	Web applications installed and executing under a W3C Widget [W3C-Widgets] engine as Web runtime environment.
Widget Engine	Software which supports the execution of Web applications running outside a browser context, e.g. with the same functional capabilities as browsers but without the user interface functions provided by a browser, including window frames, menus, toolbars and scroll bars.

3.2 Abbreviations

API	Application Programming Interface
EventSource	The EventSource API (Server-Sent Events)
HTTP	HyperText Transfer Protocol
IDL	Interface Definition Language
JSON	JavaScript Object Notation
MIME	Multipurpose Internet Mail Extensions
OMA	Open Mobile Alliance
REST	REpresentational State Transfer
RPC	Remote Procedure Call
SCR	Static Conformance Requirements
TS	Technical Specification
UA	User Agent
UE	User Equipment
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
W3C	World Wide Web Consortium
WRAPI	The OMA Web Runtime API enabler
XML	eXtensible Markup Language
XSD	XML Schema Definition

4. Introduction

This is the technical specification part of the thermometer Device Web APIs whose requirements and architecture are defined in a separate document [DWAPI-PCH]. Thermometers report measurement of body temperature. They are connected to smartphones via physical media such as Bluetooth to Extension Plug-Ins that expose the thermometer's features to applications through the GotAPI 1.1 framework [GotAPI1.1]. Applications are able to use the thermometers' features through the GotAPI 1.1 framework as defined by GotAPI 1.1.

Thermometers are typically accessed by one-shot messages, where measurement data is transferred from a thermometer to an application in one transaction. Some thermometers are capable of storing data and they may transfer multiple data in a 1-shot message. The number of data stored in thermometers is typically less than 25. However, some thermometers may be able to persistently store data and may transfer a larger number of data than 25.

The description of the measurement of body temperature reported by thermometers through Plug-Ins follows the IEEE 11073 10408 Thermometer specialization specification [IEEE 11073-10408]. Nonetheless, this does not mean the thermometer itself must follow the IEEE 11073-10408 specification. The thermometer APIs specified in this document can be used for thermometers that support IEEE 11073-10408 as well as those that do not support IEEE 11073-10408. In the latter case, however, the thermometers must provide to the Plug-Ins the necessary information such that the Plug-Ins can fulfil their reporting requirements as specified in this document.

This document defines thermometer API specifications for

- Service Discovery
- One-short measuring API
- Asynchronous measuring API

The architectural aspects of these APIs are defined in the AD section of [DWAPI-PCH]. This specification must adhere to the GotAPI 1.1 specification.

4.1 Version 1.0

Thermometer Device WebAPIs version 1.0 includes the functionality:

- Device Web API specifications for DWAPI-PCH, with device classes from IEEE 11073-10408 Thermometer specialization based on the GotAPI 1.1 framework
- Device Web APIs for Service Discovery, One-shot measuring and asynchronous measuring
- Requirements and architecture documents [DWAPI-PCH]

5. Technical Specifications

This specification must adhere to the GotAPI 1.1 specification. This document specifies certain aspect of GotAPI 1.1 as the basis and introduces new elements, that are necessary for thermometers supporting IEEE 11073-10408 Thermometer specialization.

In order to increase readability, the specification described below uses the same tables as defined in GotAPI 1.1., describing the necessary features including those of the general procedures of any GotAPI 1.1 uses as well as those specific to the thermometer APIs. Those specifications that are specific to the thermometer APIs are colored in green in the following speicificaitons in the following tables, in order to increase readability, to make identiy distinction easily. Those rows that are not colored in green are merely copies from GotAPI 1.1 specification [GotAPI 1.1]

5.1 The Service Discovery on the GotAPI-4 Interface

Service Discovery API enables applications to discover available services as define in the Section 7.2.1[DWAPI-PCH]. Service Discovery API specification adheres to that of GotAPI 1.1.

Here is the Service Discovery based on what is defined in GotAPI 1.1. After the application obtains authorization for access to GotAPI-based APIs using the GotAPI-2 Interface, the application sends the Service Discovery request to the GotAPI Server. Then the GotAPI Server sends the Service Discovery request to all of the installed Extension Plug-Ins. The message flow of the Service Discovery is shown in Fig. 1.

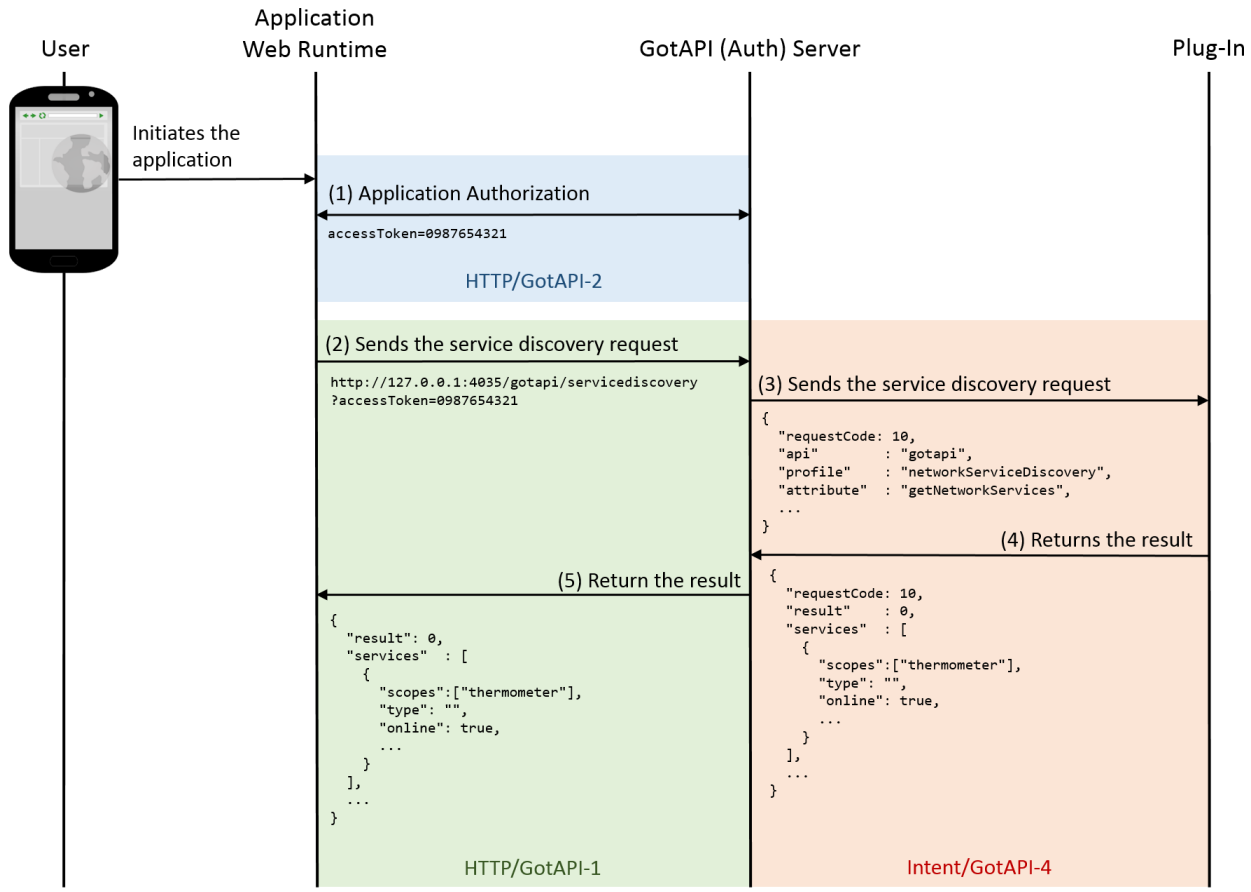

Figure 1: Message flow of the Service Discovery

The specific data in the message flows labelled (4) in the figure above are defined as follows. The other message flows SHALL be consistent to what are defined in the GotAPI 1.1 specification:

When the GotAPI Server receives the request of the Service Discovery API from an application, the GotAPI Server sends the Plug-In discovery request to the installed Plug-Ins as defined in the GotAPI specification. When the Thermometer Plug-In

receives the Plug-In discovery request from the GotAPI Server, the Thermometer Plug-In SHALL return the message as follows:

Definition of the data object for the Plug-In discovery response

Name	Sub name	Type	Definition of value	Mandatory/Optional
requestCode		int	The request code coming from the GotAPI Server.	Mandatory
result		int	If success, the value is 0, otherwise an integer other than 0, which indicates an error code. This specification doesn't define error codes.	Mandatory
services		Array		Mandatory
	serviceId	String	The service identifier. The id could be "com.example.plugin".	Mandatory
	name	String	The name of the targeted thermometer.	Mandatory
	manufacturer	String	The manufacturer of the targeted thermometer.	Optional
	version	String	The version of the targeted thermometer.	Optional
	type	String	This value represents the type of the network used to connect to the thermometer. The value must be any one of "WiFi", "BLE", "NFC", "Bluetooth" or "USB".	Optional
	online	Boolean	If the service is available, this value SHALL be true. Otherwise (e.g. the Thermometer Plug-In has not yet detect any thermometer or the Plug-In is not allowed to access to any devices), this value SHALL be false.	Mandatory
	scopes	Array	This value SHALL be an array including a string "thermometer" as an array element (["thermometer", ...]).	Mandatory

Editor's note: The value "USB" was added to the list of the allowed values for the "type" property in the data object. The value "USB" is not specified in the GotAPI 1.1 specification. Is the value "USB" necessary? If necessary, we may need to revise GotAPI 1.1 to support USB.

The Thermometer Plug-In MAY append additional data in the data object as needed.

This data object is sent to the Plug-Ins in an OS specific mechanism, .e.g., Intents for Android.

Requirements for OS-specific response channel and data container

OS	Description
Android	The GotAPI Server must use Explicit Intents for the response. The data object must be mapped to the Extra directly.

Example of the data object of the Android Explicit Intents

Name		Example of value	Note
Action		"org.deviceconnect.action.RESPONSE"	This value is defined by the GotAPI Server application.
Component		"org.deviceconnect"	This value is the package name of the GotAPI Server application.
Extra			

	requestCode	1	
	result	0	
	services	*[Array Object]*	This value is an example. Note that this is "not" a JSON string. This value must be an Array object whose content is the same as the following JSON example: [{ "id": "org.example.plugin.12345", "name": "Coolest Thermometer", "manufacturer": "ABC Health Care Inc.", "version": "3.0", **"type": "Bluetooth",** **"online": true,** **"scopes": ["thermometer"]** }, ...]
	config	*"additional parameters"*	*This name-value pair is an additional data which is not defined by this specification.*

5.2 One-shot measuring API

One-shot API enables applications to receive measured data from thermometers by one HTTP request/response transaction as define in the Section 7.2.2 [DWAPI-PCH]. One-shot measuring API specification adheres to that of GotAPI 1.1.

As defined by GotAPI 1.1, after the application obtains authorization to access GotAPI-based APIs using the GotAPI-2 Interface and completes the Service Discovery, the application can use the service (so called "One-shot measuring API") provided by the Plug-In through the GotAPI Server.

The One-shot measuring API offers a measurement result reported by the targeted device in response to a request. The message flow of this API is as shown blow.

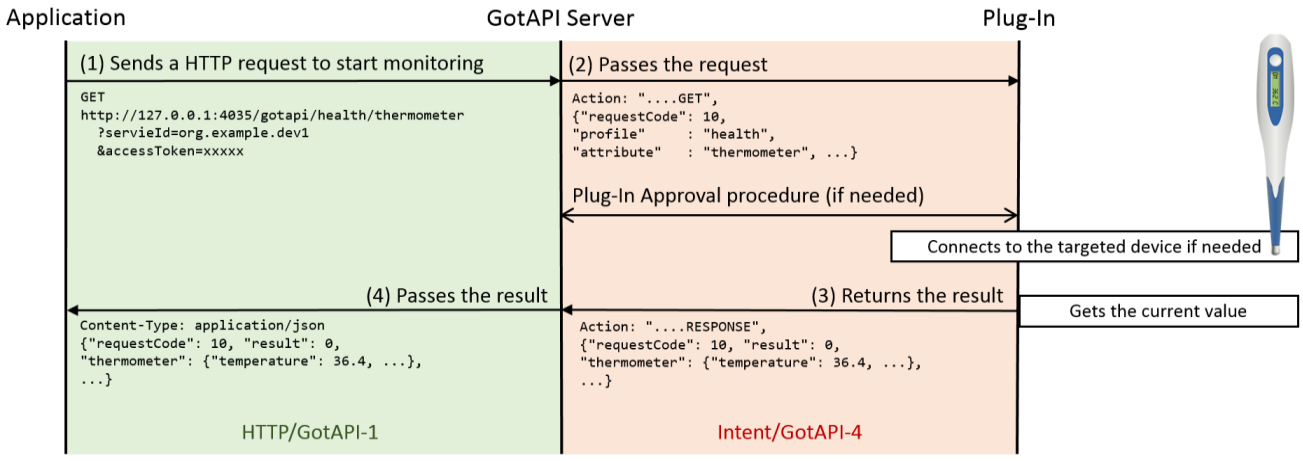

Figure 2: Message flow of the One-shot measuring API

This section defines the data object for all the message flows described in the figure above.

5.2.1 Request for one-shot measuring on the GotAPI-1 Interface

When the application uses the API in order to receive asynchronous messages, it sends a request to the GotAPI Server on the GotAPI-1 Interface as follows:

Definition of the HTTP request

	Definitions
Method	HTTP PUT
Request URL	**http://127.0.0.1:4035/gotapi/health/thermometer** **https://127.0.0.1:4036/gotapi/health/thermometer**

Definition of the request parameters

Parameter name	Definition of value	Mandatory/Optional
serviceId	The identifier of the targeted service. This value is available from the Service Discovery API on the GotAPI-1 Interface.	Mandatory
accessToken	The access token obtained from the GotAPI Auth Server through the GotAPI-2 Interface.	Mandatory
nonce	A nonce generated by the application, which is described in the section "7.3.3.3 HMAC server authentication using trusted Application ID for the Server spoofing attack" in the GotAPI specification.	Optional

Example of the request URL

```
http://127.0.0.1:4035/gotapi/health/thermometer?serviceId=abcdefg123&accessToken=0987654321&nonce=93b3a219347
```

5.2.2 Request for one-shot measuring on the GotAPI-4 Interface

When an application sends a request to the GotAPI Server on the GotAPI-1 Interface, the GotAPI Server passes the request to the Plug-In on the GotAPI-4 Interface. The request includes the data object as follows:

Definition of the data object for request

Name		Type	Definition of value	Mandatory/Optional
method		String	This value SHALL be "GET".	Mandatory if the OS is not Android. Otherwise, optional. If the OS is Android, the "Action" value SHALL include this information as described below.
receiver		String	The address of the GotAPI Server application used by Plug-Ins. Generally, it is the application ID recognized by the OS, such as a package name.	Mandatory
requestCode		int	A request code identifying the request. This value could be any number but must MUST be an integer greater than 0, and unique for each open request, to ensure responses can be correlated.	Mandatory
serviceId		String	The identifier of the targeted Service. This value is provided by the application over the GotAPI-1 Interface.	Mandatory
api		String	The value must be "gotapi".	Mandatory
profile		**String**	**The value must be "health".**	**Mandatory**

attribute		**String**	**The value must be "thermometer"**	**Mandatory**
clientId		String	The identifier of the application, which is generated by the Plug-In when the Plug-In Approval procedure defined in the GotAPI specification.	Mandatory
accessToken		String	The access token for the application, which is generated by the Plug-In when the Plug-In Approval procedure defined in the GotAPI specification.	Mandatory

This data object is sent to the Plug-Ins in an OS specific mechanism, .e.g., Intents for Android.

Requirements for OS-specific request channel and data container

OS	Description
Android	The GotAPI Server must use Explicit Intents for the request. The data object must be mapped to the Extra directly.

Example of the data object of the Android Explicit Intents

Name		Example of value	Note
Action		org.deviceconnect.action.**GET**	This value is defined by the GotAPI Server application. But the last part SHALL be "**GET**".
Component		org.example.plugin	This value is the package name of the Plug-In application.
Extra			
	receiver	org.deviceconnect	
	requestCode	10	
	servcieId	dev1.example.org	
	api	gotapi	
	profile	**health**	
	attribute	**thermometer**	
	clientId	1234567890	
	accessToken	0987654321	

5.2.3 Response for one-shot measuring on the GotAPI-4 Interface

When the Plug-In receives the request, it SHALL respond to the GotAPI Server as follows:

Definition of the data object for the response

Name			Type	Definition of value	Mandatory/Optional
method			String	This value SHALL be "RESPONSE".	Mandatory if the OS is not Android. Otherwise, optional. If the OS is Android, the "Action" value SHALL include this information as described below.

requestCode			int	The request code coming from the GotAPI Server.	Mandatory
result			int	If success, the value is 0, otherwise an integer greater than 0, which indicates an error code. This specification doesn't define error codes.	Mandatory
thermometer					
	device		Object		Mandatory
		productName	String	The product name of the targeted thermometer. If the Plug-In cannot obtain this information from the targeted thermometer, it SHALL create a name for the thermometer using an arbitrary algorithm. The algorithm is up to the Plug-In implementation, and this specification does not define any algorithms.	Mandatory
		manufacturerName	String	The manufacturer name of the targeted thermometer. If the Plug-In cannot obtain this information from the targeted thermometer, this value SHALL be an empty string.	Mandatory
		modelNumber	String	The model number of the targeted thermometer. If the Plug-In cannot obtain this information from the targeted thermometer, this value SHALL be an empty string.	Mandatory
		firmwareRevision	String	The firmware revision of the targeted thermometer. If the Plug-In cannot obtain this information from the targeted thermometer, this value SHALL be an empty string.	Mandatory
		serialNumber	String	The serial number of the targeted thermometer. If the Plug-In cannot obtain this information from the targeted thermometer, this value SHALL be an empty string.	Mandatory

		softwareRevision	String	The software revision of the targeted thermometer. If the Plug-In cannot obtain this information from the targeted thermometer, this value SHALL be an empty string.	Mandatory
		hardwareRevision	String	The hardware revision of the targeted thermometer. If the Plug-In cannot obtain this information from the targeted thermometer, this value SHALL be an empty string.	Mandatory
		partNumber	String	The part number of the targeted thermometer. If the Plug-In cannot obtain this information from the targeted thermometer, this value SHALL be an empty string.	Mandatory
		protocolRevision	String	The protocol revision of the targeted thermometer. If the Plug-In cannot obtain this information from the targeted thermometer, this value SHALL be an empty string.	Mandatory
		systemId	String	The system id of the targeted thermometer. This value SHALL be a 16-character HEX string without a '0x' prefix (e.g. "ABCDEF0123456789"). If the Plug-In cannot obtain this information from the targeted thermometer, this value SHALL be "0000000000000000" (a string of 16 '0' characters).	Mandatory

		batteryLevel	Float	The battery level of the targeted thermometer. This value must be a float number in a range from 0.0 to 1.0. The value 0.0 represents that the targeted thermometer is completely out of charge. The value 1.0 represents that the targeted thermometer is fully charged. Even if the targeted thermometer reports this value in percent in a range from 1 to 100, the Plug-In SHALL convert it to a float number in a range from 0.0 to 1.0. If the Plug-In can't obtain battery level from the targeted thermometer, this value SHALL be -1.0.	Mandatory
	temperature		Object		Mandatory
		value	Float	This value represents the temperature measured by the targeted thermometer.	Mandatory
		mderFloat	String	This value represents the temperature measured by the targetd thermometer, which is a hexadecimal string of a MDER FLOAT, such as "FFFFC8E".	Mandatory
		type	String	This value represents the body part where the Thermometer is measuring, which is expressed by a human readable string such as "Oral body temperature". If the Plug-In can't obtein the type, this value SHALL be an empty string.	Mandatory
		typeCode	String	This value represents the body part where the Thermometer is measuring, which is expressed by a code such as "188424" (This code means "Oral body temperature"). If the Plug-In can't obtein the type, this value SHALL be an empty string.	Mandatory
		unit	String	This value represents the unit of the reported temperature, which is expressed by a human readable string such as "deg C".	Mandatory

		unitCode	String	This value represents the unit of the reported temperature, which is expressed by a code such as "268192" (This code means "deg C").	Mandatory
		timeStamp	int	This value represents the measurement time when the temperature was measured. If the measurement time is reported from the targeted thermometer, the Plug-In SHALL convert it to a unix time stamp in millisecond. Otherwise, the Plug-In set this value to the unix time when the Plug-In receives the measurement value from the Plug-In based on the clock of the underlying operating system.	Mandatory
		timeStampString	String	This value represents the same time stamp as "timeStamp". The format is "YYYYMMDDHHMMSS.sss+/-HHMM", such as "20150504135813.220-0400"	Mandatory

The Plug-In MAY append additional data in the data object as needed.

This data object is sent to the GotAPI Server in an OS specific mechanism, .e.g., Intents for Android.

Requirements for OS-specific response channel and data container

OS	Description
Android	The GotAPI Server must use Explicit Intents for the request. The data object must be mapped to the Extra directly.

Example of the data object of the Android Intents

Name				Example of value	Note
Action				org.deviceconnect.action.**RESPONSE**	This value is defined by the GotAPI Server application. But the last part SHALL be "**RESPONSE**".
Component				org.deviceconnect	This value is the package name of the GotAPI Server application.
Extra					
	requestCode			10	
	result			0	
	thermometer				

		device			
			productName	ABC Thermo Pro	
			manufacturerName	ABC Inc.	
			modelNumber	TP-001	
			firmwareRevision	rev.1.001.003	
			serialNumber	01234-5678-9ABCD-EF01	
			softwareRevision	rev.2.000.000	
			hardwareRevision	rev.1.0	
			partNumber	002	
			protocolRevision	rev.3.1	
			systemId	ABCDEF0123456789	
			batteryLevel	0.5	
		temperature			
			value	36.0	
			mderFloat	FFFFC8E	
			type	Oral body temperature	
			typeCode	188424	
			unit	deg C	
			unitCode	268192	
			timeStamp	1431856940275	The sample value in the left cell represents "Sun, 17 May 2015 10:02:20 GMT".
			timeStampString	20150517100220.000-0000	

Editor's note:

The extra data of Android is just a key-value structure. How should such structured data above be expressed? JSON string?

```
intent.putExtra ("thermometer", "{\"deviceProductName\":\"ABC Thermo Pro\", ...}");
```

5.2.4 Response for one-shot measuring on the GotAPI-1 Interface

When GotAPI Server receives the response from the Plug-In, the GotAPI Server passes it to the application as follows:

Definition of the HTTP response

	Definitions
MIME-Type	application/json

HTTP status	200 OK

Definition of the data object for the response

Name			Type	Definition of value	Mandatory/Optional
product			String	The name of the GotAPI Server (e.g. "ABConnect")	Mandatory
version			String	The version of the GotAPI Server (e.g. "1.0").	Mandatory
result			Number	If success, the value is 0, otherwise an integer greater than 0, which indicates an error code. This specification doesn't define error codes.	Mandatory
thermometer					
	device		Object		Mandatory
		productName	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		manufacturerName	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		modelNumber	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		firmwareRevision	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		serialNumber	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		softwareRevision	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		hardwareRevision	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		partNumber	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		protocolRevision	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		systemId	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		batteryLevel	Number	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
	temperature		Object		Mandatory

		value	Number	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		mderFloat	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		type	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		typeCode	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		unit	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		unitCode	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		timeStamp	Number	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		timeStampString	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
hmac			String	An HMAC generated for the counter measure against the GotAPI Server spoofing attack. If the application includes a key for HMAC calculation in the API request, the GotAPI Server adds this value in the API response. Evaluating whether the HMAC is identical to the result of calculation of HMAC from the key, the application can ensure that the response is genuine.	Mandatory if the application provide a key to the GotAPI Server

The GotAPI Server SHALL serialize the data structure above as a JSON formatted stream (i.e. JSON string).

Example of the response

```
{
  "product"     : "ABCConnect",
  "version"     : "1.0",
  "requestCode" : 10,
  "result"      : 0,
  "thermometer" : {
    "device"     : {
      "productName"      : "ABC Thermo Pro",
      "manufacturerName" : "ABC Inc.",
      "modelNumber"      : "TP-001",
      "firmwareRevision" : "rev.1.001.003",
      "serialNumber"     : "01234-5678-9ABCD-EF01",
      "softwareRevision" : "rev.2.000.000",
      "hardwareRevision" : "rev.1.0",
```

```
      "partNumber"       : "002",
      "protocolRevision" : "rev.3.1",
      "systemId"         : "ABCDEF0123456789",
      "batteryLevel"     : 0.5
    },
    "temperature": {
      "value"            : 36.4,
      "mderFloat"        : "FFFFC8E",
      "type"             : "Oral body temperature",
      "typeCode"         : "188424",
      "unit"             : "deg C",
      "unitCode"         : "268192",
      "timeStamp"        : 1431856940275,
      "timeStampString"  : "20150517100220.000-0000"
    }
  },
  "hmac"        : "0123456789"
}
```


5.3 Asynchronous messaging API

Asynchronous messaging API enables applications to receive measured data from thermometers asynchronously using WebSocket as define in the Section 7.2.3 [DWAPI-PCH]. Asynchronous messaging API specification adheres to that of GotAPI 1.1.

As defined by GotAPI 1.1, after the application obtains authorization to access GotAPI-based APIs using the GotAPI-2 Interface and completes the Service Discovery, the application can use the service (so called "Asynchronous messaging API") provided by the Plug-In through the GotAPI Server.

The Asynchronous messaging API offers a series of measurement values reported by the targeted device to an application in real time as the measurement values become available. The timing when and the reasons why such measurement values become available is determined by the Plug-Ins and connected devices, and is out of the scope of this specification.

This API uses WebSocket protocol to handle asynchronous event messages. The message flow of this API is shown blow:

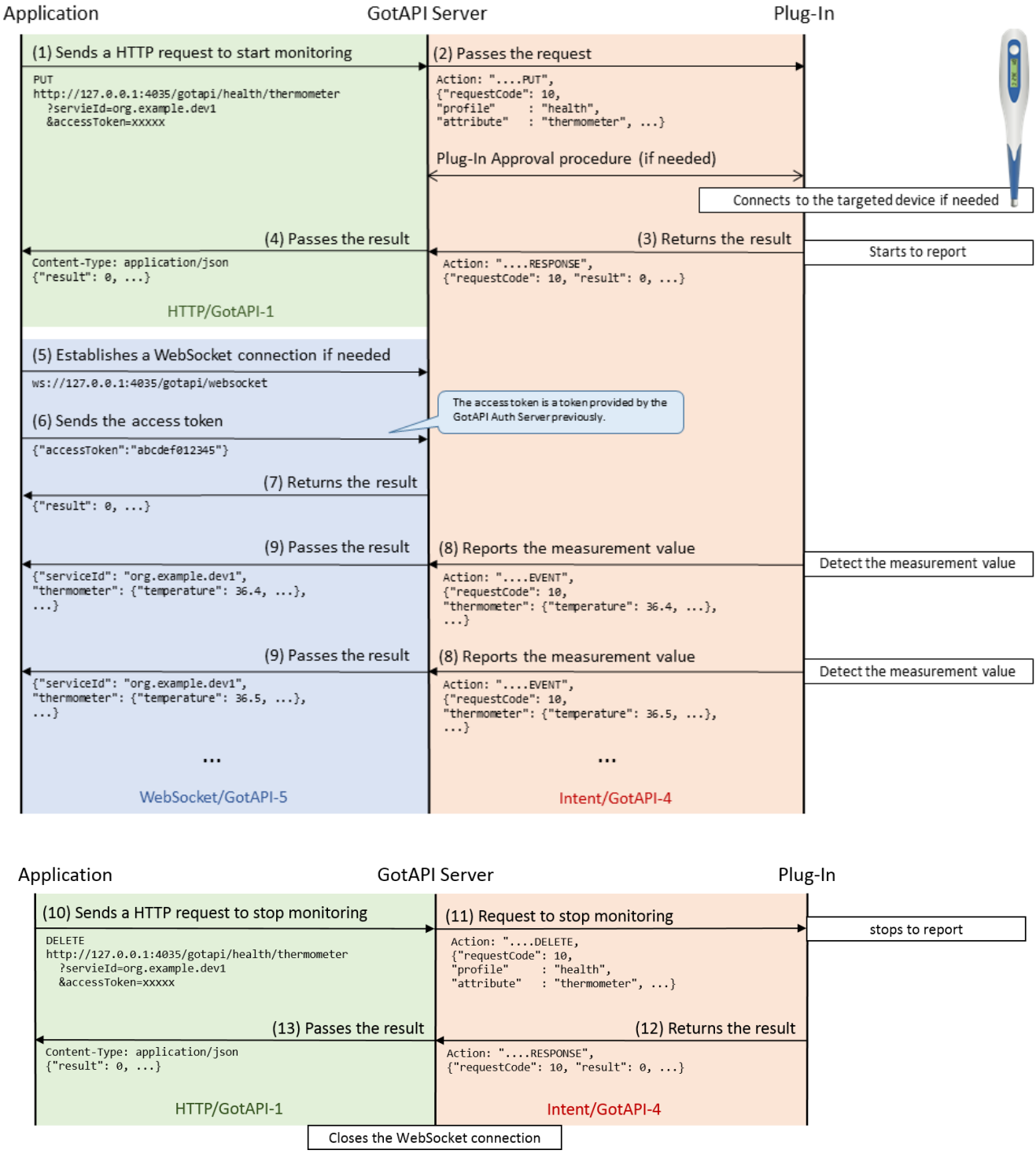

Figure 3: Message Flow of the Asynchronous messaging API

This section defines the data object for the message flows labelled from (1) to (4) and from (8) to (13) described in the figure above.

5.3.1 Request for asynchronous messaging on the GotAPI-1 Interface

When the application uses the API in order to receive asynchronous messages, it sends a request to the GotAPI Server on the GotAPI-1 Interface as follows:

Definition of the HTTP request

	Definitions
Method	HTTP PUT
Request URL	**http://127.0.0.1:4035/gotapi/health/thermometer** **https://127.0.0.1:4036/gotapi/health/thermometer**

Definition of the request parameters

Parameter name	Definition of value	Mandatory/Optional
serviceId	The identifier of the targeted service. This value is available from the Service Discovery API on the GotAPI-1 Interface.	Mandatory
accessToken	The access token obtained from the GotAPI Auth Server through the GotAPI-2 Interface.	Mandatory
nonce	A nonce generated by the application, which is described in the section "7.3.3.3 HMAC server authentication using trusted Application ID for the Server spoofing attack" in the GotAPI specification.	Optional

Example of the request URL

```
http://127.0.0.1:4035/gotapi/health/thermometer?serviceId=abcdefg123&accessToken=0987654321&nonce=93b3a219347
```

5.3.2 Request for asynchronous messaging on the GotAPI-4 Interface

When an application sends a request to the GotAPI Server on the GotAPI-1 Interface, the GotAPI Server passes the request to the Plug-In on the GotAPI-4 Interface. The request includes the data object as follows:

Definition of the data object for request

Name		Type	Definition of value	Mandatory/Optional
method		String	This value SHALL be "PUT".	Mandatory if the OS is not Android. Otherwise, optional. If the OS is Android, the "Action" value SHALL include this information as described below.
receiver		String	The address of the GotAPI Server application used by Plug-Ins. Generally, it is the application ID recognized by the OS, such as a package name.	Mandatory
requestCode		int	A request code identifying the request. This value could be any number but must MUST be an integer greater than 0, and unique for each open request, to ensure responses can be correlated.	Mandatory
serviceId		String	The identifier of the targeted Service. This value is provided by the application over the GotAPI-1 Interface.	Mandatory
api		String	The value must be "gotapi".	Mandatory
profile		**String**	**The value must be "health".**	**Mandatory**

attribute		**String**	**The value must be "thermometer"**	**Mandatory**
clientId		String	The identifier of the application, which is generated by the Plug-In when the Plug-In Approval procedure defined in the GotAPI specification.	Mandatory
accessToken		String	The access token for the application, which is generated by the Plug-In when the Plug-In Approval procedure defined in the GotAPI specification.	Mandatory

This data object is sent to the Plug-Ins in an OS specific mechanism, .e.g., Intents for Android.

Requirements for OS-specific request channel and data container

OS	Description
Android	The GotAPI Server must use Explicit Intents for the request. The data object must be mapped to the Extra directly.

Example of the data object of the Android Explicit Intents

Name		Example of value	Note
Action		org.deviceconnect.action.**PUT**	This value is defined by the GotAPI Server application. But the last part SHALL be "**PUT**".
Component		org.example.plugin	This value is the package name of the Plug-In application.
Extra			
	receiver	org.deviceconnect	
	requestCode	10	
	servcieId	dev1.example.org	
	api	gotapi	
	profile	**health**	
	attribute	**thermometer**	
	clientId	1234567890	
	accessToken	0987654321	

5.3.3 Response for asynchronous messaging on the GotAPI-4 Interface

When the Plug-In receives the request, it SHALL respond to the GotAPI Server as follows:

Definition of the data object for the response

Name			Type	Definition of value	Mandatory/Optional
method			String	This value SHALL be "RESPONSE".	Mandatory if the OS is not Android. Otherwise, optional. If the OS is Android, the "Action" value SHALL include this information as described below.

requestCode			Number	The request code coming from the GotAPI Server.	Mandatory
result			Number	If success, the value is 0, otherwise an integer greater than 0, which indicates an error code. This specification doesn't define error codes.	Mandatory
thermometer			Object		Mandatory
	device		Object		Mandatory
		productName	String	The product name of the targeted thermometer. If the Plug-In cannot obtain this information from the targeted thermometer, it SHALL create a name for the thermometer using an arbitrary algorithm. The algorithm is up to the Plug-In implementation, and this specification does not define any algorithms.	Mandatory
		manufacturerName	String	The manufacturer name of the targeted thermometer. If the Plug-In cannot obtain this information from the targeted thermometer, this value SHALL be an empty string.	Mandatory
		modelNumber	String	The model number of the targeted thermometer. If the Plug-In cannot obtain this information from the targeted thermometer, this value SHALL be an empty string.	Mandatory
		firmwareRevision	String	The firmware revision of the targeted thermometer. If the Plug-In cannot obtain this information from the targeted thermometer, this value SHALL be an empty string.	Mandatory
		serialNumber	String	The serial number of the targeted thermometer. If the Plug-In cannot obtain this information from the targeted thermometer, this value SHALL be an empty string.	Mandatory
		softwareRevision	String	The software revision of the targeted thermometer. If the Plug-In cannot obtain this information from the targeted thermometer, this value SHALL be an empty string.	Mandatory

		hardwareRevision	String	The hardware revision of the targeted thermometer. If the Plug-In cannot obtain this information from the targeted thermometer, this value SHALL be an empty string.	Mandatory
		partNumber	String	The part number of the targeted thermometer. If the Plug-In cannot obtain this information from the targeted thermometer, this value SHALL be an empty string.	Mandatory
		protocolRevision	String	The protocol revision of the targeted thermometer. If the Plug-In cannot obtain this information from the targeted thermometer, this value SHALL be an empty string.	Mandatory
		systemId	String	The system id of the targeted thermometer. This value SHALL be a 16-character HEX string without a '0x' prefix (e.g. "ABCDEF0123456789"). If the Plug-In cannot obtain this information from the targeted thermometer, this value SHALL be "0000000000000000" (a string of 16 '0' characters).	Mandatory

The Plug-In MAY append additional data in the data object as needed.

This data object is sent to the GotAPI Server in an OS specific mechanism, .e.g., Intents for Android.

Requirements for OS-specific response channel and data container

OS	Description
Android	The GotAPI Server must use Explicit Intents for the request. The data object must be mapped to the Extra directly.

Example of the data object of the Android Intents

Name				Example of value	Note
Action				org.deviceconnect.action.**RESPONSE**	This value is defined by the GotAPI Server application. But the last part SHALL be "**RESPONSE**".
Component				org.deviceconnect	This value is the package name of the GotAPI Server application.
Extra					
	requestCode			10	
	result			0	

	thermometer				
		device			
			productName	ABC Thermo Pro	
			manufacturerName	ABC Inc.	
			modelNumber	TP-001	
			firmwareRevision	rev.1.001.003	
			serialNumber	01234-5678-9ABCD-EF01	
			softwareRevision	rev.2.000.000	
			hardwareRevision	rev.1.0	
			partNumber	002	
			protocolRevision	rev.3.1	
			systemId	ABCDEF0123456789	

Editor's note:
The extra data of Android is just a key-value structure. How should such structured data above be expressed? JSON string?
`intent.putExtra ("thermometer", "{\"deviceProductName\":\"ABC Thermo Pro\", ...}");`

5.3.4 Response for asynchronous messaging on the GotAPI-1 Interface

When GotAPI Server receives the response from the Plug-In, the GotAPI Server passes it to the application as follows:

Definition of the HTTP response

	Definitions
MIME-Type	application/json
HTTP status	200 OK

Definition of the data object for the response

Name			Type	Definition of value	Mandatory/Optional
product			String	The name of the GotAPI Server (e.g. "ABConnect")	Mandatory
version			String	The version of the GotAPI Server (e.g. "1.0").	Mandatory
result			Number	If success, the value is 0, otherwise an integer greater than 0, which indicates an error code. This specification doesn't define error codes.	Mandatory
thermometer			Object		Mandatory
	device		Object		Mandatory
		productName	String	This value SHALL be the same as that which the GotAPI Server received	Mandatory

				from the Plug-In.	
		manufacturerName	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		modelNumber	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		firmwareRevision	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		serialNumber	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		softwareRevision	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		hardwareRevision	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		partNumber	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		protocolRevision	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		systemId	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
hmac			String	An HMAC generated for the counter measure against the GotAPI Server spoofing attack. If the application includes a key for HMAC calculation in the API request, the GotAPI Server adds this value in the API response. Evaluating whether the HMAC is identical to the result of calculation of HMAC from the key, the application can ensure that the response is genuine.	Mandatory if the application provide a key to the GotAPI Server

The GotAPI Server SHALL serialize the data structure above as a JSON formatted stream (i.e. JSON string).

5.3.5 Asynchronous message from the Plug-In to the GotAPI Server on the GotAPI-4 Interafce

The Plug-In sends an asynchronous message as follows:

Definition of the data object for request

Name	Sub name		Type	Definition of value	Mandatory/Optional
method			String	This value SHALL be "EVENT".	Mandatory if the OS is not Android. Otherwise, optional. If the OS is Android, the "Action" value SHALL include this information as described below.
requestCode			int	The request code coming from the GotAPI Server.	Mandatory
result			Number	If success, the value is 0, otherwise an integer greater than 0, which indicates an error code. This specification doesn't define error codes.	Mandatory
thermometer			Object		Mandatory
	device		Object		Mandatory
		batteryLevel	Float	The battery level of the targeted thermometer. This value must be a float number in a range from 0.0 to 1.0. The value 0.0 represents that the targeted thermometer is completely out of charge. The value 1.0 represents that the targeted thermometer is fully charged. Even if the targeted thermometer reports this value in percent in a range from 1 to 100, the Plug-In SHALL convert it to a float number in a range from 0.0 to 1.0. If the Plug-In can't obtain battery level from the targeted thermometer, this value SHALL be -1.0.	Mandatory
	temperature		Object		Mandatory
		value	Float	This value represents the temperature measured by the targeted thermometer.	Mandatory
		mderFloat	String	This value represents the temperature measured by the targetd thermometer, which is a hexadecimal string of a MDER	Mandatory

				FLOAT, such as "FFFFC8E".	
		type	String	This value represents the body part where the Thermometer is measuring, which is expressed by a human readable string such as "Oral body temperature". If the Plug-In can't obtein the type, this value SHALL be an empty string.	Mandatory
		typeCode	String	This value represents the body part where the Thermometer is measuring, which is expressed by a code such as "188424" (This code means "Oral body temperature"). If the Plug-In can't obtein the type, this value SHALL be an empty string.	Mandatory
		unit	String	This value represents the unit of the reported temperature, which is expressed by a human readable string such as "deg C".	Mandatory
		unitCode	String	This value represents the unit of the reported temperature, which is expressed by a code such as "268192" (This code means "deg C").	Mandatory
		timeStamp	int	This value represents the measurement time when the temperature was measured. If the measurement time is reported from the targeted thermometer, the Plug-In SHALL convert it to a unix time stamp in millisecond. Otherwise, the Plug-In set this value to the unix time when the Plug-In receives the measurement value from the Plug-In based on the clock of the underlying operating system.	Mandatory
		timeStampString	String	This value represents the same time stamp as "timeStamp". The format is "YYYYMMDDHHMMSS.sss+/-HHMM", such as "20150504135813.220-0400"	Mandatory

The Plug-In MAY append additional data in the data object as needed.

This data object is sent to the Plug-Ins in an OS specific mechanism, .e.g., Intents for Android.

Requirements for OS-specific request channel and data container

OS	Description
Android	The GotAPI Server must use Explicit Intents for the request. The data object must be mapped to the Extra directly.

Example of the data object of the Android Explicit Intents

Name	Extra key name	Example of value	Note

Action				org.deviceconnect.action.**EVENT**	This value is defined by the GotAPI Server application. But the last part SHALL be "**EVENT**".
Component				org.example.plugin	This value is the package name of the Plug-In application.
Extra					
	requestCode			10	
	result			0	
	thermometer				
		device			
			batteryLevel	0.5	
		temperature			
			value	36.0	
			mderFloat	FFFFC8E	
			type	Oral body temperature	
			typeCode	188424	
			unit	deg C	
			unitCode	268192	
			timeStamp	1431856940275	The sample value in the left cell represents "Sun, 17 May 2015 10:02:20 GMT".
			timeStampString	20150517100220.000-0000	

Editor's note:

The extra data of Android is just a key-value structure. How should such structured data above be expressed? JSON string?

```
intent.putExtra ("thermometer", "{\"deviceProductName\":\"ABC Thermo Pro\", ...}");
```

5.3.6 Asynchronous message from the GotAPI Server to the application on the GotAPI-5 Interface

When the GotAPI Server receives an asynchronous message from the Plug-In, the GotAPI Server passes it to the application on the GotAPI-5 Interface. The format of the data is a JSON string as follows:

Definition of the data object

Name	Sub name		Type	Definition of value	Mandatory/Optional

serviceId			String	The identifier of the targeted Service. This value is provided by the application when the application send the originated API request on the GotAPI-1 Interface.	Mandatory
thermometer					Mandatory
	device		Object		Mandatory
		batteryLevel	Number	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
	temperature		Object		Mandatory
		value	Number	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		mderFloat	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		type	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		typeCode	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		unit	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		unitCode	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		timeStamp	Number	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
		timeStampString	String	This value SHALL be the same as that which the GotAPI Server received from the Plug-In.	Mandatory
hmac			String	An HMAC generated for the counter measure against the GotAPI Server spoofing attack. If the application includes a key for HMAC calculation in the API request, the GotAPI Server adds this value in the API response. Evaluating whether the HMAC is identical to the result of calculation of HMAC from the key, the application can ensure	Mandatory if the application provide a key to the GotAPI Server

				that the response is genuine.	

Example of the JSON string

```
{
  "serviceId" : 0,
  "thermometer": {
    "device"      : {
      "batteryLevel"    : 0.5
    }
    "temperature" : {
      "value":          : 36.4,
      "mderFloat"       : "FFFFC8E",
      "type"            : "Oral body temperature",
      "typeCode"        : "188424",
      "unit"            : "deg C",
      "unitCode"        : "268192",
      "timeStamp"       : 1431856940275,
      "timeStampString" : "20150517100220.000-0000"
    }
  }
}
```

5.3.7 Stop request from the application to the GotAPI Server on the GotAPI-1 Interface

When the application wants to stop receiving asynchronous messages, it sends a request to the GotAPI Server on the GotAPI-1 Interface as follows:

Definition of the HTTP request

	Definitions
Method	HTTP DELETE
Request URL	**http://127.0.0.1:4035/gotapi/health/thermometer** **https://127.0.0.1:4036/gotapi/health/thermometer**

Definition of the request parameters

Parameter name	Definition of value	Mandatory/Optional
serviceId	The identifier of the targeted service. This value is available from the Service Discovery API on the GotAPI-1 Interface.	Mandatory
accessToken	The access token obtained from the GotAPI Auth Server through the GotAPI-2 Interface.	Mandatory
nonce	A nonce generated by the application, which is described in the section "7.3.3.3 HMAC server authentication using trusted Application ID for the Server spoofing attack" in the GotAPI specification.	Optional

Example of the request URL

```
http://127.0.0.1:4035/gotapi/health/thermometer?serviceId=abcdefg123&accessToken=0987654321&nonce=93b3a219347
```

5.3.8 Stop request from the GotAPI Server to the Plug-In on the GotAPI-4 Interface

When the GotAPI Server receives a stop request from the application on the GotAPI-1 Interface, the GotAPI Server sends a stop request to the Plug-in on the GotAPI-4 Interface. The request includes the data object as follows:

Definition of the data object for request

Name		Type	Definition of value	Mandatory/Optional
method		String	This value SHALL be "DELETE".	Mandatory if the OS is not Android. Otherwise, optional. If the OS is Android, the "Action" value SHALL include this information as described below.
receiver		String	The address of the GotAPI Server application used by Plug-Ins. Generally, it is the application ID recognized by the OS, such as a package name.	Mandatory
requestCode		int	A request code identifying the request. This value could be any number but must MUST be an integer greater than 0, and unique for each open request, to ensure responses can be correlated.	Mandatory
serviceId		String	The identifier of the targeted Service. This value is provided by the application over the GotAPI-1 Interface.	Mandatory
api		String	The value must be "gotapi".	Mandatory
profile		**String**	**The value must be "health".**	**Mandatory**
attribute		**String**	**The value must be "thermometer"**	**Mandatory**
clientId		String	The identifier of the application, which is generated by the Plug-In when the Plug-In Approval procedure defined in the GotAPI specification.	Mandatory
accessToken		String	The access token for the application, which is generated by the Plug-In when the Plug-In Approval procedure defined in the GotAPI specification.	Mandatory

This data object is sent to the Plug-Ins in an OS specific mechanism, .e.g., Intents for Android.

Requirements for OS-specific request channel and data container

OS	Description
Android	The GotAPI Server must use Explicit Intents for the request. The data object must be mapped to the Extra directly.

Example of the data object of the Android Explicit Intents

Name		Example of value	Note
Action		org.deviceconnect.action.**DELETE**	This value is defined by the GotAPI Server application. But the last part SHALL be "**DELETE**".
Component		org.example.plugin	This value is the package name of the

			Plug-In application.
Extra			
	receiver	org.deviceconnect	
	requestCode	10	
	servcieId	dev1.example.org	
	api	gotapi	
	profile	health	
	attribute	thermometer	
	clientId	1234567890	
	accessToken	0987654321	

5.3.9 Stop response from the Plug-In to the GotAPI Server on the GotAPI-4 Interface

When the Plug-In receives the stop request, it SHALL respond as follows:

Definition of the data object for the response

Name		Type	Definition of value	Mandatory/Optional
method		String	This value SHALL be "RESPONSE".	Mandatory if the OS is not Android. Otherwise, optional. If the OS is Android, the "Action" value SHALL include this information as described below.
requestCode		Number	The request code coming from the GotAPI Server.	Mandatory
result		Number	If success, the value is 0, otherwise an integer greater than 0, which indicates an error code. This specification doesn't define error codes.	Mandatory

The Plug-In MAY append additional data in the data object as needed.

This data object is sent to the GotAPI Server in an OS specific mechanism, .e.g., Intents for Android.

Requirements for OS-specific response channel and data container

OS	Description
Android	The GotAPI Server must use Explicit Intents for the request. The data object must be mapped to the Extra directly.

Example of the data object of the Android Intents

Name	Sub name	Example of value	Note
Action		org.deviceconnect.action.**RESPONSE**	This value is defined by the GotAPI Server application. But the last part SHALL be "**RESPONSE**".
Component		org.deviceconnect	This value is the package name of the GotAPI Server application.

Extra			
	requestCode	10	
	result	0	

5.3.10 Stop response from the GotAPI Server to the application on the GotAPI-1 Interface

When the GotAPI Server receives the stop response, the GotAPI Server passes the response to the application follows:

Definition of the HTTP response

	Definitions
MIME-Type	application/json
HTTP status	200 OK

Definition of the data object for the response

Name		Type	Definition of value	Mandatory/Optional
product		String	The name of the GotAPI Server (e.g. "ABConnect")	Mandatory
version		String	The version of the GotAPI Server (e.g. "1.0").	Mandatory
result		Number	If success, the value is 0, otherwise an integer greater than 0, which indicates an error code. This specification doesn't define error codes.	Mandatory
hmac		String	An HMAC generated for the counter measure against the GotAPI Server spoofing attack. If the application includes a key for HMAC calculation in the API request, the GotAPI Server adds this value in the API response. Evaluating whether the HMAC is identical to the result of calculation of HMAC from the key, the application can ensure that the response is genuine.	Mandatory if the application provide a key to the GotAPI Server

The GotAPI Server SHALL serialize the data structure above as a JSON formatted stream (i.e. JSON string), then send it to the originating application on the GotAPI-5 (WebSocket connection).

Example of the response

```
{
  "product": "ABCConnect",
  "version": "1.0",
  "result" : 0,
  "hmac"   : "0123456789"
}
```

Appendix A. Change History (Informative)

A.1 Approved Version History

Reference	Date	Description
OMA-TS-Thermometer_APIs-V1_0-20180724-A	24 Jul 2018	Status changed to Approved by CD Doc Ref # OMA-CD-2018-0005-INP_DWAPI_V1_0_ERP_for_final_Approval