Heart Rate Monitor APIs

Candidate Version 1.0 – 19 Apr 2016

Open Mobile Alliance

OMA-TS-Heart_Rate_Monitor_APIs-V1_0-20160419-C

Use of this document is subject to all of the terms and conditions of the Use Agreement located at
http://www.openmobilealliance.org/UseAgreement.html.

Unless this document is clearly designated as an approved specification, this document is a work in process, is not an
approved Open Mobile Alliance™ specification, and is subject to revision or removal without notice.

You may use this document or any part of the document for internal or educational purposes only, provided you do not
modify, edit or take out of context the information in this document in any manner. Information contained in this document
may be used, at your sole risk, for any purposes. You may not use this document in any other manner without the prior
written permission of the Open Mobile Alliance. The Open Mobile Alliance authorizes you to copy this document, provided
that you retain all copyright and other proprietary notices contained in the original materials on any copies of the materials
and that you comply strictly with these terms. This copyright permission does not constitute an endorsement of the products
or services. The Open Mobile Alliance assumes no responsibility for errors or omissions in this document.

Each Open Mobile Alliance member has agreed to use reasonable endeavors to inform the Open Mobile Alliance in a timely
manner of Essential IPR as it becomes aware that the Essential IPR is related to the prepared or published specification.
However, the members do not have an obligation to conduct IPR searches. The declared Essential IPR is publicly available
to members and non-members of the Open Mobile Alliance and may be found on the "OMA IPR Declarations" list at
http://www.openmobilealliance.org/ipr.html. The Open Mobile Alliance has not conducted an independent IPR review of
this document and the information contained herein, and makes no representations or warranties regarding third party IPR,
including without limitation patents, copyrights or trade secret rights. This document may contain inventions for which you
must obtain licenses from third parties before making, using or selling the inventions. Defined terms above are set forth in
the schedule to the Open Mobile Alliance Application Form.

NO REPRESENTATIONS OR WARRANTIES (WHETHER EXPRESS OR IMPLIED) ARE MADE BY THE OPEN
MOBILE ALLIANCE OR ANY OPEN MOBILE ALLIANCE MEMBER OR ITS AFFILIATES REGARDING ANY OF
THE IPR'S REPRESENTED ON THE "OMA IPR DECLARATIONS" LIST, INCLUDING, BUT NOT LIMITED TO THE
ACCURACY, COMPLETENESS, VALIDITY OR RELEVANCE OF THE INFORMATION OR WHETHER OR NOT
SUCH RIGHTS ARE ESSENTIAL OR NON-ESSENTIAL.

THE OPEN MOBILE ALLIANCE IS NOT LIABLE FOR AND HEREBY DISCLAIMS ANY DIRECT, INDIRECT,
PUNITIVE, SPECIAL, INCIDENTAL, CONSEQUENTIAL, OR EXEMPLARY DAMAGES ARISING OUT OF OR IN
CONNECTION WITH THE USE OF DOCUMENTS AND THE INFORMATION CONTAINED IN THE DOCUMENTS.

Contents

Figures

Tables

1. Scope

Heart rate is one of the essential vital signs of health measurements.

The GotAPI provides a multi-purpose web-based framework to enable interwork applications and external devices such as Heart Rate Monitors. The GotAPI consists of the GotAPI Server and the Extension Plug-Ins. A smartphone application communicates with a specified Extension Plug-In through the GotAPI Server using Web technologies

In the GotAPI framework, an Extension Plug-In interacts with Heart Rate Monitors and exposes interfaces to the GotAPI Server. Thanks to Extension Plug-Ins, smartphone applications can interact with many kinds of Heart Rate Monitors using consistent APIs specified in this specification.

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

2. References

2.1 Normative References

[Bluetooth HRP]	Bluetooth SIG, Heart Rate Profile, Version 1.0. URL:https://www.bluetooth.org/docman/handlers/downloaddoc.ashx?doc_id=239865
[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
Bluetooth HRP	Bluetooth SIG, Heart Rate Profile, Version 1.0.
Bluetooth Low Energy	Wireless technology developed by the Bluetooth Special Interest Group (SIG) for short-range communication
Browser Context	Web applications executing under a Web browser as Web runtime environment.
Cardiac cycle	a complete cardiac movement, or heart beat, including systole, diastole, and the intervening pause
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].
Heart Rate Monitor	In general, an ECG device with associated lead wires and electrodes measures the electrical activity of the heart.
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].
lead	Commonly refers to two different things: It may be used to refer to the combination of an electrode and associated lead wire, used for a certain ECG recording. Alternatively, it may be used to refer to the signal obtained by tracing the voltage between two electrodes or linear combinations thereof. The latter definition is used throughout this standard
lead wire	A cable connected between an electrode and the agent device
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.
RR interval	In an electrocardiogram, the interval from the onset of one R wave to the onset of the next one, one complete cardiac cycle
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
ECG	Basic electrocardiograph
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 Heart Rate Monitor Device WebAPIs whose requirements and architecture are defined in a separate document [DWAPI-PCH]. 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. Heart Rate Monitor Plug-In is specified in this specification.

Heart Rate Monitors supported by the Plug-Ins in this specification are expected to be able to report the heart rate and RR intervals, if RR is supported. The description of the measurements reported by the Heart Rate Monitor Plug-Ins follows the IEEE 11073-10406 [Basic electrocardiograph (ECG) (1- to 3-lead ECG)] specialization specification. The IEEE 11073-10406 ECG specification supports a couple of sub profiles. The Heart Rate Monitor Plug-Ins defined here only support part of the heart rate sub profile of the IEEE ECG specialization. Additional objects are defined in the IEEE 11073-10406 ECG heart rate sub profile. These additional objects are more oriented towards medical information. The Heart Rate Monitor Plug-Ins that are defined in this specification do not support the measurements of those additional objects.

There are additional features that the Heart Rate APIs defined here support beyond the heart rate sub profile of the IEEE ECG specialization, which is the calories burned measurement. Devices that support Bluetooth Low Energy Heart Rate Profile (HRP) may report calories burned measurement. This measurement is NOT supported by the IEEE 11073-10406 ECG specialization but is supported in the IEEE 11073-10441 cardiovascular specialization specification. The Heart Rate Plug-Ins that are defined in this specification will report the calories burned measurement, as the Bluetooth HRP does, if the device that is connected to the Plug-In supports calories burned measurement.

Heart Rate Monitors are typically accessed by streamed data (asynchronous messaging), where measurement data is transferred from a Heart Rate Monitor to an application asynchronously. Some Heart Rate Monitors are capable of storing data and they may transfer multiple data. The number of data stored in Heart Rate Monitors is typically less than 25. However, some Heart Rate Monitors may be able to persistently store data and may transfer a larger number of data than 25.

The description of the measurement of Heart Rate Monitors reported by Heart Rate Monitor Plug-Ins that are defined in this specification follows the IEEE 11073-10406 heart rate sub profile of the specialization specification. Nonetheless, this does not mean the Heart Rate Monitors themselves must follow the IEEE 11073-10406 heart rate sub profile. The Heart Rate Monitor APIs specified in this document can be used for Heart Rate Monitors that support IEEE 11073-10406 heart rate sub profile as well as those that do not support IEEE 11073-10406. In the latter case, however, the Heart Rate Monitors must provide the Plug-Ins with the necessary information such that the Plug-Ins can fulfill their reporting requirements as specified in this document.

It should be noted that Bluetooth Low Energy supports the Heart Rate Profile [Bluetooth HRP] which is modelled after the IEEE 11073-10406 specification. However, devices that support the Bluetooth Low Energy Heart Rate Profile are likely to be suited for casual use cases such as exercising, whereas devices supporting the IEEE standards are likely to be used in more medical care situations since the IEEE devices use electric sensor wires attached to the body whereas casual heart rate monitors use other techniques. Despite such differences, the Heart Rate Monitor APIs that are defined in this specification can be used for versatile use cases ranging from casual use cases such as exercising to medical use cases.

This document defines Heart Rate Monitor 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

Heart Rate Monitor Device WebAPIs version 1.0 includes the functionality:

- Device WebAPI specifications for DWAPI-PCH, with device classes from IEEE 11073-10406 ECG specialization based on the GotAPI 1.1 framework

- Device WebAPIs 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 Heart Rate Monitor Devices supporting IEEE 11073-10406 ECG heart rate sub profile of the IEEE ECG 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 Heart Rate Monitor APIs. Those specifications that are specific to the Heart Rate Monitor APIs are colored in green 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 the 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.

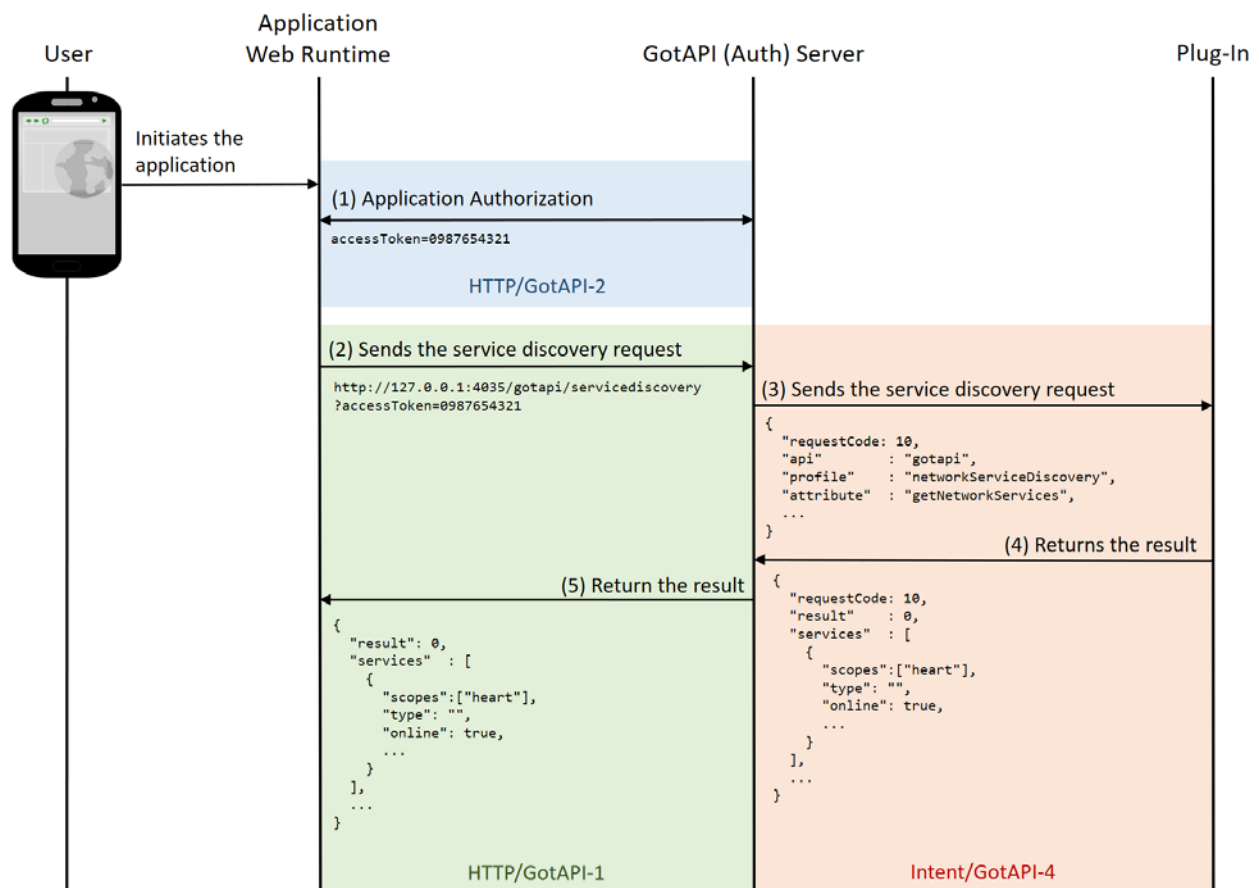

Figure 1: Message flow of the Service Discovery

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 Plug-In receives the Plug-In discovery request from the GotAPI Server, the 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 device.	Mandatory
	manufacturer	String	The manufacturer of the targeted device.	Optional
	version	String	The version of the targeted device.	Optional
	type	String	This value represents the type of the network used to connect to the device. 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 Plug-In has not yet detected any devices 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 "bca" as an array element (["heart", ...]).	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 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 Heart Rate Monitor",`

`    "manufacturer": "ABC Health Care Inc.",`

`    "version": "3.0",`

`    "type": "Bluetooth",`

`    "online": true,`

`    "scopes": ["heart"]`

`  },`

`  ...`

`]` |
| | *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 targeted devices 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.
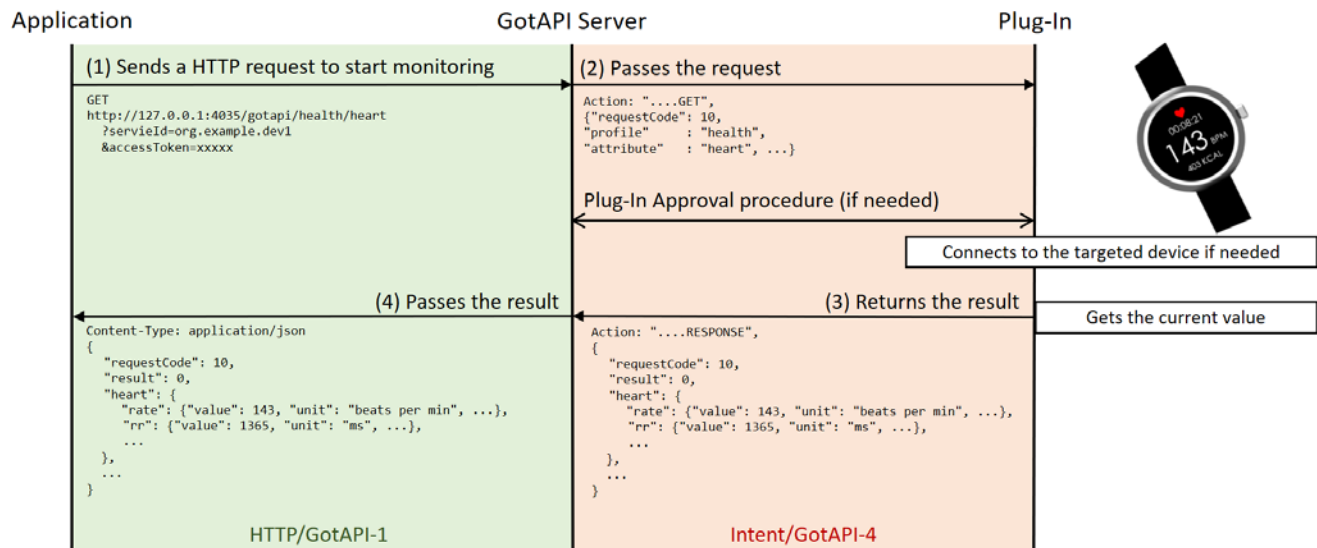
Figure 2: Message flow of the One-shot measuring API

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 one-shot measuring 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/heart https://127.0.0.1:4036/gotapi/health/heart

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/heart?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 "heart"	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	heart	
	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
heart					Mandatory
	device		Object		Mandatory
		productName	String	The product name of the targeted device. If the Plug-In cannot obtain this information from the targeted device, it SHALL create a name for the device 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 device. If the Plug-In cannot obtain this information from the targeted device, this value SHALL be an empty string.	Mandatory
		modelNumber	String	The model number of the targeted device. If the Plug-In cannot obtain this information from the targeted device, this value SHALL be an empty string.	Mandatory

		firmwareRevision	String	The firmware revision of the targeted device. If the Plug-In cannot obtain this information from the targeted device, this value SHALL be an empty string.	Mandatory
		serialNumber	String	The serial number of the targeted device. If the Plug-In cannot obtain this information from the targeted device, this value SHALL be an empty string.	Mandatory
		softwareRevision	String	The software revision of the targeted device. If the Plug-In cannot obtain this information from the targeted device, this value SHALL be an empty string.	Mandatory
		hardwareRevision	String	The hardware revision of the targeted device. If the Plug-In cannot obtain this information from the targeted device, this value SHALL be an empty string.	Mandatory
		partNumber	String	The part number of the targeted device. If the Plug-In cannot obtain this information from the targeted device, this value SHALL be an empty string.	Mandatory
		protocolRevision	String	The protocol revision of the targeted device. If the Plug-In cannot obtain this information from the targeted device, this value SHALL be an empty string.	Mandatory
		systemId	String	The system id of the targeted device. 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 device, this value SHALL be "0000000000000000" (a string of 16 '0' characters).	Mandatory

		batteryLevel	Float	The battery level of the targeted device. This value must be a float number in a range from 0.0 to 1.0. The value 0.0 represents that the targeted device is completely out of charge. The value 1.0 represents that the targeted device is fully charged. Even if the targeted device 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 device, this value SHALL be -1.0.	Mandatory
	rate				Mandatory
		value	Float	This value represents the heart rate measured by the targeted device.	Mandatory
		mderFloat	String	This value represents the heart rate measured by the targetd device, which is a hexadecimal string of an MDER FLOAT, such as "0000002D", which means 75 bpm if the value of "unit" is "beats per min".	Mandatory
		type	String	This value represents the TYPE attribute as a human readable string and as its 32-bit MDC code such as "Heart rate". If the Plug-In can't obtain the type, this value SHALL be an empty string.	Mandatory
		typeCode	String	This value represents the TYPE attribute, which is expressed by a code such as "147842" (This code means "Heart rate"). If the Plug-In can't obtain the type, this value SHALL be an empty string.	Mandatory
		unit	String	This value represents the unit of the reported heart rate, which is expressed by a human readable string such as "beats per min".	Mandatory
		unitCode	String	This value represents the unit of the reported heart rate, which is expressed by a code such as "264864" (This code means "beats per min").	Mandatory
		timeStamp	int	This value represents the measurement time when the mesurement was done. If the measurement time is reported from the targeted device, 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
	rr				Mandatory if the device reports RR interval. Otherwise, this SHALL NOT exist.
		value	Float	This value represents the RR interval measured by the targeted device.	Mandatory
		mderFloat	String	This value represents the RR interval measured by the targetd device, which is a hexadecimal string of an MDER FLOAT, such as "00000555", which means 1365 ms if the value of "unit" is "ms".	Mandatory
		type	String	This value represents the TYPE attribute as a human readable string and as its 32-bit MDC code such as "RR interval". If the Plug-In can't obtain the type, this value SHALL be an empty string.	Mandatory
		typeCode	String	This value represents the TYPE attribute, which is expressed by a code such as "147240" (This code means "RR interval"). If the Plug-In can't obtain the type, this value SHALL be an empty string.	Mandatory
		unit	String	This value represents the unit of the reported RR interval, which is expressed by a human readable string such as "ms".	Mandatory
		unitCode	String	This value represents the unit of the reported RR interval, which is expressed by a code such as "264338" (This code means "ms").	Mandatory
		timeStamp	int	This value represents the measurement time when the mesurement was done. If the measurement time is reported from the targeted device, 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

		energy				Mandatory if the device reports expended energy. Otherwise, this SHALL NOT exist.
			value	Float	This value represents the expended energy measured by the targeted device.	Mandatory
			mderFloat	String	This value represents the expended energy measured by the targetd device, which is a hexadecimal string of an MDER FLOAT, such as " 00000005", which means 5 Calories per time if the value of "unit" is "Calories per time".	Mandatory
			type	String	This value represents the TYPE attribute as a human readable string and as its 32-bit MDC code such as "energy expended". If the Plug-In can't obtain the type, this value SHALL be an empty string.	Mandatory
			typeCode	String	This value represents the TYPE attribute, which is expressed by a code such as "119" (This code means "energy expended"). If the Plug-In can't obtain the type, this value SHALL be an empty string.	Mandatory
			unit	String	This value represents the unit of the reported expended energy, which is expressed by a human readable string such as "Calories".	Mandatory
			unitCode	String	This value represents the unit of the reported expended energy, which is expressed by a code such as "6784" (This code means "Calories").	Mandatory
			timeStamp	int	This value represents the measurement time when the mesurement was done. If the measurement time is reported from the targeted device, 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	
	heart				
		device			
			productName	ABC Heart Rate Monitor 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	
		rate			
			value	75	
			mderFloat	0000002D	
			type	Heart rate	
			typeCode	147842	
			unit	beats per min	
			unitCode	264864	
			timeStamp	1431856940275	

			timeStampString	20150517100220.000-0000	
		rr			
			value	1365	
			mderFloat	00000555	
			type	RR interval	
			typeCode	147240	
			unit	ms	
			unitCode	264338	
			timeStamp	1431856940275	
			timeStampString	20150517100220.000-0000	
		energy			
			value	5	
			mderFloat	00000005	
			type	energy expended	
			typeCode	119	
			unit	Calories	
			unitCode	6784	
			timeStamp	1431856940275	
			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 ("heart", "{\"deviceProductName\":\"ABC Heart Rate Monitor 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
heart				Object		Mandatory
	device			Object		Mandatory
		productName		String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		manufacturerName		String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		modelNumber		String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		firmwareRevision		String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		serialNumber		String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		softwareRevision		String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		hardwareRevision		String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		partNumber		String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		protocolRevision		String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		systemId		String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		batteryLevel		Number	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
	rate			Object		Mandatory
		value		Float	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		mderFloat		String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		type		String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		typeCode		String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		unit		String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		unitCode		String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		timeStamp		int	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		timeStampString		String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory

rr			Object		Mandatory if the device reports RR interval. Otherwise, this SHALL NOT exist.
		value	Float	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		mderFloat	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		type	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		typeCode	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		unit	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		unitCode	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		timeStamp	int	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		timeStampString	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
	energy		Object		Mandatory if the device reports expended energy. Otherwise, this SHALL NOT exist.
		value	Float	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		mderFloat	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		type	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		typeCode	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		unit	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		unitCode	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		timeStamp	int	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		timeStampString	String	This value SHALL be the same as what 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,
  "heart"              : {
    "device": {
      "productName"      : "ABC Heart Rate Monitor 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
    },
    "rate": {
      "value"          : 75,
      "mderFloat"      : "0000002D",
      "type"           : "Heart rate",
      "typeCode"       : "147842",
      "unit"           : "beats per min",
      "unitCode"       : "264864",
      "timeStamp"      : 1431856940275,
      "timeStampString" : "20150517100220.000-0000"
    }
    "rr": {
      "value"          : 1365
      "mderFloat"      : "00000555",
      "type"           : "RR interval",
      "typeCode"       : "147240",
      "unit"           : "ms",
      "unitCode"       : "264338",
      "timeStamp"      : 1431856940275,
      "timeStampString" : "20150517100220.000-0000"
    },
    "energy": {
      "value"          : 5,
```

```
      "mderFloat"          : "00000005",
      "type"               : "energy expended",
      "typeCode"           : "119",
      "unit"               : "Calories",
      "unitCode"           : "6784",
      "timeStamp"          : 1431856940275,
      "timeStampString"    : "20150517100220.000-0000"
    }
  }
  "hmac"                   : "0123456789"
 }
```

5.3   Asynchronous messaging API

Asynchronous messaging API enables applications to receive measured data from the targeted device 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:

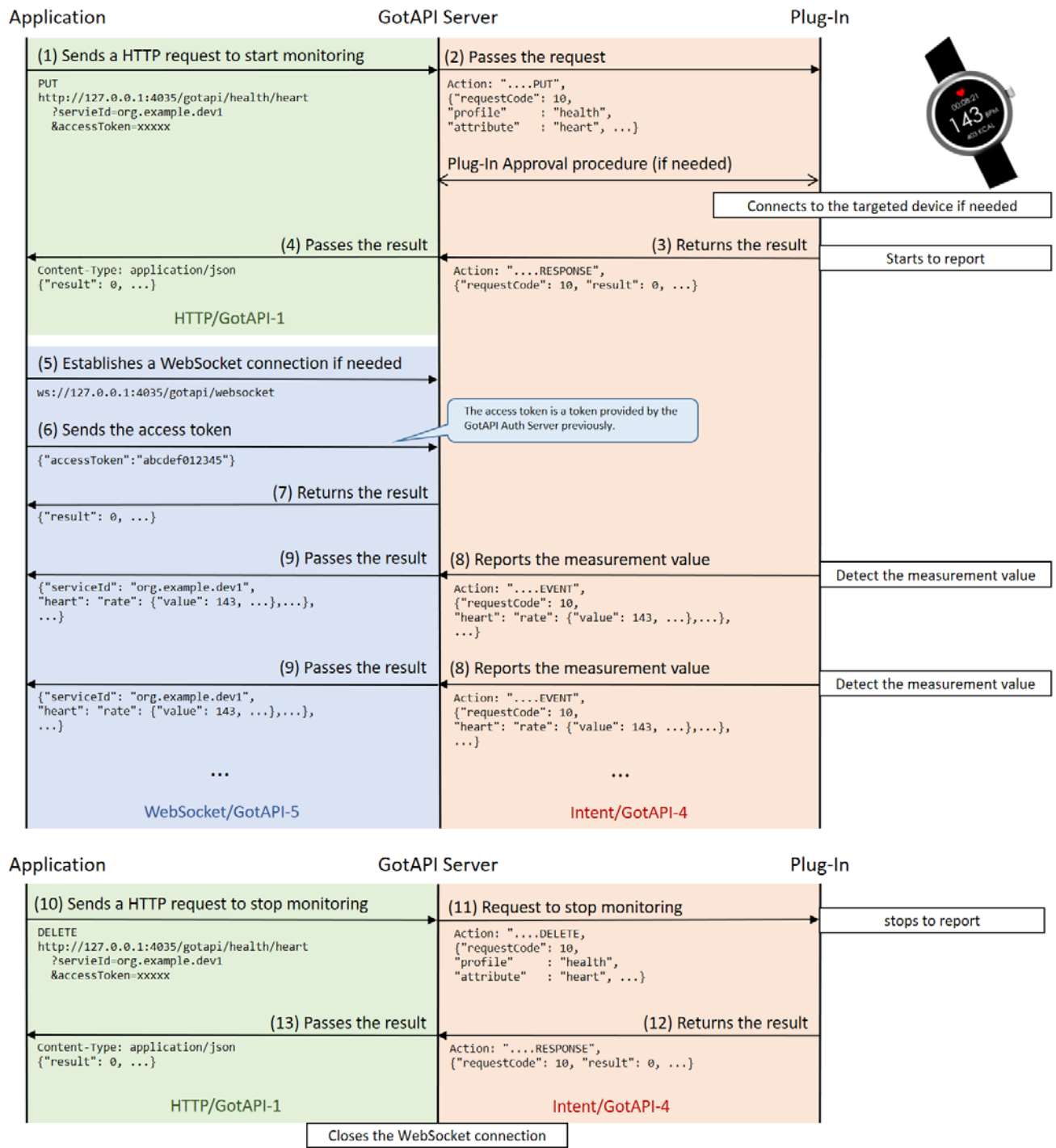**Figure 3: Message Flow of the Asynchronous messaging API**

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/heart https://127.0.0.1:4036/gotapi/health/heart

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/heart?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 "heart"	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	heart	
	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
heart						Mandatory
	device			Object		Mandatory
		productName		String	The product name of the targeted device. If the Plug-In cannot obtain this information from the targeted device, it SHALL create a name for the device 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 device. If the Plug-In cannot obtain this information from the targeted device, this value SHALL be an empty string.	Mandatory
		modelNumber		String	The model number of the targeted device. If the Plug-In cannot obtain this information from the targeted device, this value SHALL be an empty string.	Mandatory
		firmwareRevision		String	The firmware revision of the targeted device. If the Plug-In cannot obtain this information from the targeted device, this value SHALL be an empty string.	Mandatory
		serialNumber		String	The serial number of the targeted device. If the Plug-In cannot obtain this information from the targeted device, this value SHALL be an empty string.	Mandatory

		softwareRevision	String	The software revision of the targeted device. If the Plug-In cannot obtain this information from the targeted device, this value SHALL be an empty string.	Mandatory
		hardwareRevision	String	The hardware revision of the targeted device. If the Plug-In cannot obtain this information from the targeted device, this value SHALL be an empty string.	Mandatory
		partNumber	String	The part number of the targeted device. If the Plug-In cannot obtain this information from the targeted device, this value SHALL be an empty string.	Mandatory
		protocolRevision	String	The protocol revision of the targeted device. If the Plug-In cannot obtain this information from the targeted device, this value SHALL be an empty string.	Mandatory
		systemId	String	The system id of the targeted device. 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 device, 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	

	heart					
		device				
			productName	ABC Heart Rate Monitor 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 ("heart", "{\"deviceProductName\":\"ABC Heart Rate Monitor 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
heart				Object		Mandatory
	device			Object		Mandatory
		productName		String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory

		manufacturerName	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		modelNumber	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		firmwareRevision	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		serialNumber	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		softwareRevision	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		hardwareRevision	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		partNumber	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		protocolRevision	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		systemId	String	This value SHALL be the same as what 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,
  "heart" : {
    "device": {
      "productName"     : "ABC Heart Rate Monitor 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"
    }
  },
  "hmac"        : "0123456789"
}
```

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			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
heart			Object		Mandatory
	device		Object		Mandatory
		batteryLevel	Float	The battery level of the targeted deivce. This value must be a float number in a range from 0.0 to 1.0. The value 0.0 represents that the targeted deivce is completely out of charge. The value 1.0 represents that the targeted deivce is fully charged. Even if the targeted deivce 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 deivce, this value SHALL be -1.0.	Mandatory
	rate				Mandatory
		value	Float	This value represents the heart rate measured by the targeted device.	Mandatory
		mderFloat	String	This value represents the heart rate measured by the targetd device, which is a hexadecimal string of an MDER FLOAT, such as "0000002D", which means 75 bpm if the value of "unit" is "beats per min".	Mandatory

		type	String	This value represents the TYPE attribute as a human readable string and as its 32-bit MDC code such as "Heart rate". If the Plug-In can't obtain the type, this value SHALL be an empty string.	Mandatory
		typeCode	String	This value represents the TYPE attribute, which is expressed by a code such as "147842" (This code means "Heart rate"). If the Plug-In can't obtain the type, this value SHALL be an empty string.	Mandatory
		unit	String	This value represents the unit of the reported heart rate, which is expressed by a human readable string such as "beats per min".	Mandatory
		unitCode	String	This value represents the unit of the reported heart rate, which is expressed by a code such as "264864" (This code means "beats per min").	Mandatory
		timeStamp	int	This value represents the measurement time when the mesurement was done. If the measurement time is reported from the targeted device, 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
	rr				Mandatory if the device reports RR interval. Otherwise, this SHALL NOT exist.
		value	Float	This value represents the RR interval measured by the targeted device.	Mandatory
		mderFloat	String	This value represents the RR interval measured by the targetd device, which is a hexadecimal string of an MDER FLOAT, such as "00000555", which means 1365 ms if the value of "unit" is "ms".	Mandatory
		type	String	This value represents the TYPE attribute as a human readable string and as its 32-bit MDC code such as "RR interval". If the Plug-In can't obtain the type, this value SHALL be an empty string.	Mandatory
		typeCode	String	This value represents the TYPE attribute, which is expressed by a code such as "147240" (This code means "RR interval"). If the Plug-In can't obtain the type, this value SHALL be an empty string.	Mandatory

		unit	String	This value represents the unit of the reported RR interval, which is expressed by a human readable string such as "ms".	Mandatory
		unitCode	String	This value represents the unit of the reported RR interval, which is expressed by a code such as "264338" (This code means "ms").	Mandatory
		timeStamp	int	This value represents the measurement time when the mesurement was done. If the measurement time is reported from the targeted device, 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
	energy				Mandatory if the device reports expended energy. Otherwise, this SHALL NOT exist.
		value	Float	This value represents the expended energy measured by the targeted device.	Mandatory
		mderFloat	String	This value represents the expended energy measured by the targetd device, which is a hexadecimal string of an MDER FLOAT, such as "00000005", which means 5 Calories if the value of "unit" is "Calories".	Mandatory
		type	String	This value represents the TYPE attribute as a human readable string and as its 32-bit MDC code such as "energy expended". If the Plug-In can't obtain the type, this value SHALL be an empty string.	Mandatory
		typeCode	String	This value represents the TYPE attribute, which is expressed by a code such as "119" (This code means "energy expended"). If the Plug-In can't obtain the type, this value SHALL be an empty string.	Mandatory
		unit	String	This value represents the unit of the reported expended energy, which is expressed by a human readable string such as "Calories".	Mandatory
		unitCode	String	This value represents the unit of the reported expended energy, which is expressed by a code such as "6784" (This code means "Calories").	Mandatory

| | | timeStamp | int | This value represents the measurement time when the mesurement was done. If the measurement time is reported from the targeted device, 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	
	heart			
		device		
			deviceBatteryLevel	0.5
		rate		
			value	75
			mderFloat	0000002D
			type	Heart rate
			typeCode	147842
			unit	beats per min
			unitCode	264864
			timeStamp	1431856940275

			timeStampString	20150517100220.000-0000
		rr		
			value	1365
			mderFloat	00000555
			type	RR interval
			typeCode	147240
			unit	ms
			unitCode	264338
			timeStamp	1431856940275
			timeStampString	20150517100220.000-0000
		energy		
			value	5
			mderFloat	00000005
			type	energy expended
			typeCode	119
			unit	Calories
			unitCode	6784
			timeStamp	1431856940275
			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 ("heart", "{\"deviceProductName\":\"ABC Heart Rate Mointor 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
heart			Object		Mandatory
	device		Object		Mandatory

		batteryLevel	Number	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
	rate				Mandatory
		value	Float	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		mderFloat	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		type	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		typeCode	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		unit	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		unitCode	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		timeStamp	int	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		timeStampString	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
	rr		Object		Mandatory if the device reports RR interval. Otherwise, this SHALL NOT exist.
		value	Float	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		mderFloat	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		type	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		typeCode	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		unit	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		unitCode	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		timeStamp	int	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		timeStampString	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
	energy		Object		Mandatory if the device reports expended energy. Otherwise, this SHALL NOT exist.
		value	Float	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory

		mderFloat	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		type	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		typeCode	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		unit	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		unitCode	String	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		timeStamp	int	This value SHALL be the same as what the GotAPI Server received from the Plug-In.	Mandatory
		timeStampString	String	This value SHALL be the same as what 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

Example of the JSON string

```
{
  "serviceId" : 0,
  "heart"     : {
    "device": {
      "batteryLevel"    : 0.5
    },
    "rate": {
      "value"           : 75,
      "mderFloat"       : "0000002D",
      "type"            : "Heart rate",
      "typeCode"        : "147842",
      "unit"            : "beats per min",
      "unitCode"        : "264864",
      "timeStamp"       : 1431856940275,
      "timeStampString" : "20150517100220.000-0000"
    }
    "rr": {
      "value"           : 1365
      "mderFloat"       : "00000555",
      "type"            : "RR interval",
      "typeCode"        : "147240",
      "unit"            : "ms",
      "unitCode"        : "264338",
      "timeStamp"       : 1431856940275,
      "timeStampString" : "20150517100220.000-0000"
    },
```

```
    "energy": {
      "value"           : 5,
      "mderFloat"       : "00000005",
      "type"            : "energy expended",
      "typeCode"        : "119",
      "unit"            : "Calories",
      "unitCode"        : "6784",
      "timeStamp"       : 1431856940275,
      "timeStampString" : "20150517100220.000-0000"
    }
  },
  "hmac"                : "0123456789"
}
```

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/heart https://127.0.0.1:4036/gotapi/health/heart

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/heart?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 "heart"	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	heart	
	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
n/a	n/a	No prior version

A.2   Draft/Candidate Version 1.0 History

Document Identifier	Date	Sections	Description
Draft Versions OMA-TS- Heart_Rate_Monitor_APIs- V1_0	04 Nov 2015	All	Initial baseline document.
	05 Nov 2015	Various	Updates for agreed CR: OMA-CD-DWAPI-2015-0042- CR_Heart_Rate_Monitor_APIs_TS_Baseline
	03 Jan 2016	5.2.3 5.2.4 5.3.5 5.3.6	OMA-CD-DWAPI-2015-0047- CR_Codes_for_energy_expended_specified_in_Heart_Rate_Monitor_APIs_TS
Candidate Version OMA-TS- Heart_Rate_Monitor_APIs- V1_0	19 Apr 2016	n/a	Status changed to Candidate by TP TP Ref # OMA-TP-2016-0057- INP_DWAPI_V1_0_ERP_for_Candidate_approval