



REST API SERVICE

Web Services API | Version 1.62

API Specification

The information contained within this document is the property of PageOne Communications Ltd t/a Critico and may not be copied used or disclosed in whole or in part, except with the prior written permission of Critico.

Table of Contents

REST SERVICE	1
Critico Rest API	4
Overview	4
Rest Principles	5
Rest Method Hierarchy	6
REST Methods	7
<i>Authentication</i>	7
<i>Send Message</i>	8
<i>Credits</i>	9
<i>MSISDN</i>	10
How to receive messages & delivery reports	12
<i>Push Service</i>	12
Help & Support	13
Appendix: Response and Status Codes	14
A.Send Message Result Codes	14
B.Delivery Reports Result Codes	15
C.Supported Source addresses	17

Amendment History

Date	Update
<i>August 2011</i>	<i>Initial Draft Completed</i>
<i>January 2012</i>	<i>Added section for REST Callback Servers</i>
<i>January 2013</i>	<i>Added section for Pull Service</i>
<i>January 2013</i>	<i>Added section for Push Service</i>
<i>October 2014</i>	<i>Added locate specification</i>
<i>April 2016</i>	<i>Password optional status amended Help & Support contact number amended</i>
<i>February 2017</i>	<i>Remove old Pull service</i>
<i>May 2018</i>	<i>Added reference to Destination Address Validation policy</i>
<i>Aug 2019</i>	<i>Added tx-id for client specified reference number. Location service methods removed</i>
<i>April 2020</i>	<i>Removed /Group function</i>
<i>May 2023</i>	<i>Updated senderID/alphatag rules</i>

Critico Rest API

Overview

The Critico REST interface leverages the reliability and scalability features of Oventus, Critico's multi-channel messaging platform, over a simple REST/HTTP interface. Messages are produced and consumed by sending and receiving simple HTTP messages.

Here are some of the reasons you might want to use the Critico REST Interface:

- *Usable by any programming language that has an HTTP client library.*
- *Zero client footprint.*
- *No need to download, install and configure a special library to interact with Critico.*
- *No envelope (i.e. SOAP) or feed (i.e. Atom) format requirements.*
- *Lightweight interoperability. Since interactions are RESTful the HTTP uniform interface provides all the interoperability you need to communicate between different languages, platforms.*
- *Multi-channel messaging - send messages to different device types via single API - inc. SMS, email, pager, Responder app, TTS messages.*

This REST API document relates to sending of outbound messages. If you wish to receive delivery reports and/or inbound messages Critico supports a REST Push (call-back) API that allow you to consume delivery reports and/or inbound messages via a registered client end-point.

Note:

Address Validation - any recipient numbers or addresses presented via the Critico REST interface I ('to' field) will be validated and processed in accordance with Critico's Destination Address Validation Policy. Click here: **[Address Validation Policy](#)**

SMS Message parts - for SMS messages the length and content of the message text will dictate whether the messages will be sent and charged as a multi-part message. The use of international or extended characters will reduce the number of characters per message part. See **<https://criticogroup.com/support/sms-message-formatting-charging>**

REST Principles

REST exposes resources (sources of specific information), which are referenced by global identifiers (e.g., a URI in HTTP). In order to manipulate these resources, components of the network (user agents and origin servers) communicate via a standardized interface (e.g., HTTP) and exchange representations of these resources (the actual documents conveying the information)

The Critico RESTful web service provides a simple interface over HTTP to interact with the Critico communication platform, by exposing services from a base URI to resources

Notes:

- *The base URI is rest.oventus.com/rest/*
- *The internet media type of the data supported by the web service is typically JSON or XML but can be any other valid Internet media type.*
- *The set of operations supported use HTTP methods (e.g., POST, GET, PUT or DELETE).*

The following table shows an overview of how the HTTP methods are used in the web service.

RESTful Web Service HTTP methods				
<i>Resource</i>	<i>GET</i>	<i>PUT</i>	<i>POST</i>	<i>DELETE</i>
<i>Collection URI</i>	List the URIs and perhaps other details of the collection's members.	Replace the entire collection with another collection.	Create a new entry in the collection. The new entry's URL is assigned automatically and is usually returned by the operation.	Delete the entire collection.
<i>Element URI</i>	Retrieve a representation of the addressed member of the collection, expressed in an appropriate Internet media type.	Update the addressed member of the collection.	Treat the addressed member as a collection in its own right and create a new entry in it.	Delete the addressed member of the collection.

REST Method Hierarchy

The following structure illustrates the URI hierarchy, http verbs and methods currently supported

- */rest*
 - */**{version_number}*
 - */**{username}*
 - *GET (Authenticate User)*
 - */credits*
 - *GET (Retrieve Account Credits)*
 - */msisdn*
 - *GET (List Msisdn, ShortCodes and Alpha Tags)*
 - */message*
 - *POST (Send a message)*
 - */received*
 - *GET (Fetch newly received messages [max 20])*
 - */reports*
 - *GET (Fetch newly received delivery reports[max 20])*

REST Methods

Base URL of all requests

https://rest.omentus.com/rest/{version_number}/{username}/

Where **{version_number}** = v1

Note:

- The path *{username}* is unique to each account and will be provided by Critico and is a requirement of all requests.
- Authentication failures for all rest methods result in a 401 response and the cancellation of the request

GET	Authentication	{BaseURI}
------------	-----------------------	-----------

The Critico REST service supports both stateless and 'state full' interaction, in a 'state full' scenario authentication is required at the start of a session and after any authorisation failures. Alternatively a stateless session can be used where authorisation will occur on each request.

Request

Field Name	Definition
name	The username of the account
password	The password of the account

Request Headers

Header Name	Header Value
Accept	application/xml application/json

Response

Field Name	Definition
status:Description	Describes the result of the request
status	Authentication code, 200 for all authenticated requests

Exception

ID	Definition
401	Authorisation Required

Example

Request	<code>https://rest.omentus.com/rest/v1/{username}?password=<password></code>
Response (JSON)	<code>{"accountID":"<unique_account_id>","username":"<username>","status":{"@description":"User Authenticated","\$":"200"}}</code>
Failed Authentication Response (HTTP)	401 Authorisation Required User is not authorized to access this service.

POST

Send Message

{BaseURI}/message

The send message method provides the ability to publish/send a message. The response will contain a status description expressing the acceptance or rejection of the message for delivery and a 'transaction id' which will be required for matching message reports/delivery receipts.

Request

Field Name	Definition	Optional
Name	The username of the account	
password	The password of the account	
to (List<String>)	List of intended recipients (max. 100 per request)*	
from	The msisdn/callsign/alphanum of the sender	
deliverytime	Schedules in GMT when the message is to be sent. Format: yyyy-MM-dd HH:mm:ss	Y
message	The message text **	
tx_id	client transaction ref. (inc. as <i>clientTransactionID</i> in delivery reports - see REST PUSH)	Y

*Note, any recipient numbers or addresses presented within the 'to' field will be validated and processed in accordance with Critico's Destination Address Validation Policy as published: <https://criticogroup.com/AddressValidationPolicy>

** For SMS messages the length and content of the message text will dictate whether the messages will be sent and charged as a multi-part message. See: <https://criticogroup.com/support/sms-message-formatting-charging>

Request Headers

Header Name	Header Value
Content-Type	application/x-www-form-urlencoded
Accept	application/xml application/json

Response

Field Name	Definition
transactionID	Critico transactionID assigned to the message
status:Description	Describes the result of the request
status	201 for all accepted messages

Exception

ID	Definition
401	Authorisation Required
500	Service Exception

Example

Request	https://rest.ovens.com/rest/v1/{username}/message?password=<password>
Request Body	to=<toAddress>&message=<messageText>
Response (JSON)	{"@transactionID":"14024254","status":{"@description":"Accepted","\$":"201"}}

GET	Credit	{BaseURI}/credits
------------	---------------	-------------------

The credits service returns the number of remaining credits assigned to an account.

Request

Field Name	Definition
name	The username of the account
password	The password of the account

Request Headers

Header Name	Header Value
Accept	application/xml application/json

Response

Field Name	Definition
""	Number of credits remaining

Exception

ID	Definition
401	Authorisation Required

Example

Request	https://rest.oventus.com/rest/v1/{username}/credits?password=<password>
Response (JSON)	"<number of credits>"
Failed Authentication Response (HTTP)	401 Authorisation Required User is not authorized to access this service.

GET

MSISDN

{BaseURI}/msisdn

The msisdn service is used to return the source addresses which are assigned to the account.

Request

Field Name	Definition
name	The username of the account
password	The password of the account

Request Headers

Header Name	Header Value
Accept	application/xml application/json

Response

Field Name	Definition
Msisdn	Msisdn/Alphatag/shortcode address
Keyword	Account based keywords associated with the account ('*' denotes allows all)

Exception

ID	Definition
401	Authorisation Required
500	Service Exception

Example

Request	<code>https://rest.ovens.com/rest/v1/{username}/msisdn?password=<password></code>
Response (JSON)	<code>{"Msisdn":[{"msisdn":"44700000001","keyword":"*"}, {"msisdn":"AlphaTag","keyword":"*"}]}</code>

How to receive inbound messages & delivery reports

REST Push (Call-back) Service

Critico's REST Push Service provides users with the ability to receive inbound messages/replies, delivery reports and Responder device messages as soon as they are available to a registered push receiver end-point. Once the implementation is complete, according to the specification stated in this document, the user can assess and process the information.

In order to utilise this service the user needs to obtain a Critico account and have at least one push receiver registered with Critico.

See Critico REST Push Services API User Guide for detailed information available at <https://criticogroup.com/resources/developer-apis/>

Help & Support

For more information and support, please contact customer support:

Tel: 0333 200 5033
email: customersupport@criticogroup.com
website: criticogroup.com

Appendix: Response and Status Codes

A. Send Message Result Codes

The following list of status/error messages are generated by the Critico Oventus gateway during a validation phase before we accept the message.

<i>Code</i>	<i>Description</i>	
201	<i>Accepted</i>	<i>Message request has been successfully received by Critico</i>
400	<i>No Access to this Service</i>	<i>You have not signed up to the service you are trying to use</i>
402	<i>Service currently unavailable</i>	<i>Service connection is currently unavailable</i>
552	<i>Inactive Subscriber</i>	<i>The device number is inactive</i>
553	<i>Max Limit Reached</i>	<i>The set amount of messages for this service has been reached (Generally applies to our trial customers)</i>
554	<i>Please Contact Service Provider</i>	<i>Operator network failure</i>
555	<i>System Error</i>	<i>Service connection is currently unavailable</i>
556	<i>Unknown</i>	<i>Final status is unknown</i>
558	<i>Failure to Replace</i>	<i>MT network unable to replace the SMS on the MT customer's handset</i>

B. Delivery Reports Result Codes

These are message statuses that are generated after the Critico gateway has accepted the message for delivery.

Code	Description	In Detail
200	Sent	Message has been delivered to handset
201	Accepted	Message has been received and is being sent to handset
202	Group Accepted	Group message has been received and sent to the group device number
214	TTS Accepted	Message has been accepted as type TTS
203	Valid Login	Login Successful
400	No Access to this Service	You have not signed up to the service you are trying to use
401	Message Not Sent	Rejected by Remote Provider
402	Service currently unavailable	Service connection is currently unavailable
403	Could not send e-mail	Unable to convert message / device into e-mail
404	Unable to deliver	Unspecified protocol error on the MT handset
501	Bad Character	Content related error
502	Invalid Number	The device number you are sending to is incorrect
503	Unknown Called Address	The device number you are sending to does not exist
504	Unknown Pager	The device number you are sending to does not exist
505	Protocol Error	Permanent operator error
507	Insufficient Credits	Your account does not presently possess enough message credits to send your request
551	Failed Security	MT handset is listed as an Illegal device on the MSC.
552	Inactive Subscriber	The device number is inactive
553	Max Limit Reached	The set amount of messages for this service has been reached (Generally applies to our trial customers)

554	Please Contact Service Provider	Operator network failure
555	Systems Error	Service connection is currently unavailable
556	Unknown	Final status is unknown
557	Expired by Provider	Unable to deliver within the allocated time frame
558	Failed to replace	MT network unable to replace the SMS on the MT customer's handset
559	Insufficient Funds	Credit related - message has not been retried by operator

C. Supported Source (Alphatag) addresses

When sending a message, the source address (alphatag) specified should confirm to the following standard.

1. *The maximum length is 11 characters*
2. *The supported characters are given in table below*
3. *Alphatags must be pre-registered against your account or risk being blocked by certain mobile networks*

Supported sender ID / Alphatag characters are shown below

Hex	Dec	Chr	ISO/IEC 10646-1:2000 Character Name
20	32		SPACE
26	38	&	AMPERSAND
2D	45	-	HYPHEN-MINUS
2E	46	.	FULL STOP
2F	47	/	SOLIDUS
30	48	0-9	ALL DIGITS FROM 0 -9
41-5A 61-7A	65-90 97-122	A-Z and a-z	LATIN CAPITAL LETTER A TO Z AND LATIN SMALL LETTER a-z
5F	95	_	UNDERSCORE