

RAVASVend Simple AnyTime Redeem Interface Specification

JSON Request Service v1.01

S.L. Dedekind

2/18/2022

Document History

Date	Author	Comment
2022-02-18	SL Dedekind	1 st Draft
2022-03-11	SL Dedekind	Added AnyTimeVoucherLookup

Document Notes

Table of Contents

Contents

Document History.....	1
Document Notes.....	1
Table of Contents	2
Introduction.....	3
Contents	3
1. User Requirements	3
1.1 Access requirements.....	3
1.2 Daily Recon Files	4
2. AnyTimeRedeem Method	4
Example 1: Successful Request / Response	5
Example 2: Incorrect Username/Password	5
Example 3: Invalid Pin Number	6
Example 4: Duplicate MsgID	6
Example 5: Response that requires Advice	6
3. AnyTimeVoucherLookup Method	7
Example 6: Successful Request / Response for unused voucher	8
Example 7: Successful Request / Response for used voucher	8
4. Advice Method	10
Example 8: Advice for Example 5	10
Example 9: Advice for Example 1	10
Example 10: Advice for a MsgID that does not exist on the server	11

Introduction

The RAVASVend web service was created to allow Re-Sellers of VAS Products to purchase and vend these products and services from RA Cellular.

The Anytime Monetary voucher allows RAVASVend Clients to Redeem such vouchers and vend products/services to customers to the value of the redeemed voucher.

RA Cellular will thus settle the redeemed value to the Clients chosen Bank on a daily/monthly basis as is appropriate.

The Simplified service is a JSON web service that requires BASIC authentication.

The intended audience is RAVASVend clients that wish to only Redeem AnyTime Vouchers and not use the full functionality of RAVASVend (vend all other products and services like prepaid electricity / airtime etc.)

All other products & services are still available to the RAVASVend client, but these must be consumed using the full SOAP/XML service as described in the "RAVASVend Interface Specification" document.

Contents

1. User Requirements

1.1 Access requirements

The following information is required to create a TEST and then PRODUCTION account on RAVASVend for the integration to begin.

Company Name	
Company Registration Number	
Company VAT	
Company Physical Address	
Company Postal Address	
Technical Primary Contact Name	
Technical Primary Contact Tel	
Technical Primary Contact email	
Technical Secondary Contact Name	
Technical Secondary Contact Tel	
Technical Secondary Contact email	
Accounts Primary Contact Name	
Accounts Primary Contact Tel	
Accounts Primary Contact email	

After the information is supplied, a test account will be created, and the following will be supplied

- a. Account Number

- b. Authorization credentials (Username and Password)
- c. Connectivity Details (URL)

1.2 Daily Recon Files

The RAVASVend system will create a daily recon file listing all billed and redeemed transactions.

The client is required to provide an email or ftp or sftp location to where these files can be published daily.

Email address	
FTP URL	
FTP Credentials	
SFTP IP and Port	
SFTP Credentials	
SFTP Directory	

2. AnyTimeRedeem Method

The AnyTimeRedeem method is used to Redeem the value of an anytime voucher. A successful redemption will consume the voucher (i.e., it becomes “used”) and will credit the clients account on RAVASVend.

Endpoint is: /AnyTimeVoucherRedeem

(for example <https://desertretail.co.za:19002/Service1.svc/AnyTimeVoucherRedeem>)

Request Parameters are:

Name	Type (maxlength)	Description
pinNumber	String (20)	Secret PinNumber of the voucher
msgID	String(45)	Unique identifier of each request (can be UUID or incrementing int or datetime with random number) but must be unique for each request
reason	String(255)	(optional) small description of what the voucher is being used for (if available) as this information is used when querying the voucher for status.
terminalID	String(45)	Unique identifier as to who is using the voucher, this could be a MSISDN / email / Account Number / Account Code / Anything that uniquely identifies the consumer in the Clients Scope.
terminalCompanyName	String(45)	(optional) Company Name or additional identifier of the consumer
terminalOperator	String(45)	(optional) Name of the consumer or friendly identifier of the consumer

Response Parameters are:

Name	Type (maxlength)	Description
msgID	String(45)	MessageID received in request (Copied from Request)
hasFault	Boolean	True if response is an error response When missing from response the value is false
faultNumber	String(45)	If hasFault=True then each fault Type has a number starting from 0 (please see full interface spec for detail)
faultDesc	String(255)	If hasFault =True then a short description of the error (normally used

		as a header)
faultMsg	String(255)	If hasFault=True then a more detailed description of the error (normally used a error body)
faultALR	Boolean	If hasFault=True and faultALR=True then Advice is required to get the response as timeout/network failure has caused that the response could not be obtained. When missing from response the value is false;
pinNumber	String(20)	Secret PinNumber of the voucher (Copied from Request)
amount	String(20) as Decimal	If hasFault=False, The Value of the voucher
tID	String(20)	If hasFault=False, The Transaction ID of the Redemption
serial	String(20)	If hasFault=False, The Serial number of the Voucher
reference	String(20)	If hasFault=False, The Reference of the Redemption
paymentFee	String(20) as Decimal	If hasFault=False, The paymentFee (if any) for Redeeming the voucher. If a paymentFee is applicable, then the amount less the paymentFee will be credited to the clients account. It is however expected that the client will give the consumer the full amount (and not deduct the payment fee)

Http response codes will be returned as follows

Http Response Code	Reason
200	Successful response (hasFault=false)
400	Response is a Failure (hasFault=true)
401	Login Failed (hasFault=true)
500	System Exception (unable to process the request) (hasFault=true)

Example 1: Successful Request / Response

Request	Response
POST /Service1.svc/AnyTimeVoucherRedeem HTTP/1.1 Authorization: Basic VEVTVDPURVNU Content-Length: 182 Content-Type: application/json Host: desertretail.co.za:19002 { "pinNumber": "12341234556", "terminalID": "selwind@gmail.com", "msgID": "20220218131005000001", "reason": "demo Redemption using JSON service", "terminalOperator": "Selwin", }	HTTP/1.1 200 cache-control: private content-type: application/json; charset=utf-8 server: Microsoft-IIS/10.0 x-aspnet-version: 4.0.30319 x-powered-by: ASP.NET date: Fri, 18 Feb 2022 11:12:29 GMT content-length: 153 { "amount": "179.000", "msgID": "20220218131005000001", "paymentFee": "0.000", "pinNumber": "12341234556", "reference": "7836", "serial": "99991461247", "tID": "7836" }

Use case shows the credentials Username = TEST and Password = TEST being used with the Authorization Header

Example 2: Incorrect Username/Password

Request	Response
POST /Service1.svc/AnyTimeVoucherRedeem HTTP/1.1 Authorization: Basic VEVTVDPXUK9QTKc= Content-Length: 182 Content-Type: application/json Host: desertretail.co.za:19002 { "pinNumber": "12341234556", "terminalID": "selwind@gmail.com", "msgID": "20220218131005000002", }	HTTP/1.1 401 cache-control: private content-length: 145 content-type: application/json; charset=utf-8 date: Fri, 18 Feb 2022 11:14:57 GMT server: Microsoft-IIS/10.0 x-aspnet-version: 4.0.30319 x-powered-by: ASP.NET {"faultDesc": "Login Failed",

<pre> "reason":"demo Redemption using JSON service", "terminalOperator":"Selwin", } </pre>	<pre> "faultMsg":"Login Failed", "faultNumber":"1", "hasFault":true, "msgID":"20220218131005000002", "pinNumber":"12341234556"} </pre>
--	--

Example 3: Invalid Pin Number

Request	Response
<pre> POST /Service1.svc/AnyTimeVoucherRedeem HTTP/1.1 Authorization: Basic VEVTVDPURVNU Content-Length: 182 Content-Type: application/json Host: desertretail.co.za:19002 { "pinNumber":"11112222333", "terminalID":"selwind@gmail.com", "msgID":"20220218131005000003", "reason":"demo Redemption using JSON service", "terminalOperator":"Selwin", } </pre>	<pre> HTTP/1.1 400 cache-control: private content-type: application/json; charset=utf-8 server: Microsoft-IIS/10.0 x-aspnet-version: 4.0.30319 x-powered-by: ASP.NET date: Fri, 18 Feb 2022 11:20:30 GMT content-length: 233 {"faultDesc":"Empty or Invalid Voucher\Pin Number", "faultMsg":"Empty or Invalid Voucher\Pin Number supplied, Please correct and try again", "faultNumber":"82", "hasFault":true, "msgID":"20220218131005000003", "pinNumber":"11112222333"} </pre>

Example 4: Duplicate MsgID

Request	Response
<pre> POST /Service1.svc/AnyTimeVoucherRedeem HTTP/1.1 Authorization: Basic VEVTVDPURVNU Content-Length: 182 Content-Type: application/json Host: desertretail.co.za:19002 { "pinNumber":"11112222333", "terminalID":"selwind@gmail.com", "msgID":"20220218131005000003", "reason":"demo Redemption using JSON service", "terminalOperator":"Selwin", } </pre>	<pre> HTTP/1.1 400 cache-control: private content-type: application/json; charset=utf-8 server: Microsoft-IIS/10.0 x-aspnet-version: 4.0.30319 x-powered-by: ASP.NET date: Fri, 18 Feb 2022 11:21:27 GMT content-length: 173 {"faultDesc":"Duplicate MsgID", "faultMsg":"Duplicate Msg ID Exists on the Server", "faultNumber":"7", "hasFault":true, "msgID":"20220218131005000003", "pinNumber":"11112222333"} </pre>

When the server receives a message with the same MsgID as a previous request, the response will reflect the error.

Example 5: Response that requires Advice

Request	Response
<pre> POST /Service1.svc/AnyTimeVoucherRedeem HTTP/1.1 Authorization: Basic VEVTVDPURVNU Content-Length: 182 Content-Type: application/json Host: desertretail.co.za:19002 { "pinNumber":"33332222333", "terminalID":"selwind@gmail.com", "msgID":"20220218131005000005", "reason":"demo Redemption using JSON service", "terminalOperator":"Selwin", } </pre>	<pre> HTTP/1.1 400 cache-control: private content-type: application/json; charset=utf-8 server: Microsoft-IIS/10.0 x-aspnet-version: 4.0.30319 x-powered-by: ASP.NET date: Fri, 18 Feb 2022 11:22:56 GMT content-length: 203 {"faultALR":true, "faultDesc":"No Response From Utility", "faultMsg":"The Utility did not respond to the request", "faultNumber":"9", "hasFault":true, "msgID":"20220218131005000005", "pinNumber":"33332222333"} </pre>

An advice request should be submitted to obtain the result of the request, as it is not known if the redemption was successful or not.

3. AnyTimeVoucherLookup Method

The AnyTimeVoucherLookup method is an optional method that can be used to obtain the status of an anytime voucher (used / not used) prior to attempting to redeem.

Endpoint is: /AnyTimeVoucherLookup

(for example <https://desertretail.co.za:19002/Service1.svc/AnyTimeVoucherLookup>)

Request Parameters are:

Name	Type (maxlength)	Description
pinNumber	String (20)	(optional) Secret PinNumber of the voucher Either Serial or PinNumber must be specified, if both are specified then only pinNumber shall be used
msgID	String(45)	Unique identifier of each request (can be UUID or incrementing int or datetime with random number) but must be unique for each request
serial	String (20)	(optional) Serial number of the voucher Either Serial or PinNumber must be specified, if both are specified then only pinNumber shall be used
terminalID	String(45)	Unique identifier as to who is using the voucher, this could be a MSISDN / email / Account Number / Account Code / Anything that uniquely identifies the consumer in the Clients Scope.

Response Parameters are:

Name	Type (maxlength)	Description
msgID	String(45)	MessageID received in request (Copied from Request)
hasFault	Boolean	True if response is an error response When missing from response the value is false
faultNumber	String(45)	If hasFault=True then each fault Type has a number starting from 0 (please see full interface spec for detail)
faultDesc	String(255)	If hasFault =True then a short description of the error (normally used as a header)
faultMsg	String(255)	If hasFault=True then a more detailed description of the error (normally used a error body)
faultALR	Boolean	If hasFault=True and faultALR=True then Advice is required to get the response as timeout/network failure has caused that the response could not be obtained. When missing from response the value is false;
pinNumber	String(20)	If hasFault=False, Secret PinNumber of the voucher (if Serial was used to query then only first 4 digits of pin will be returned)
amount	String(20) as Decimal	If hasFault=False, The Value of the voucher
serial	String(20)	If hasFault=False, The Serial number of the Voucher
reserved	Boolean	If hasFault=False, If the voucher has been reserved. A voucher that is reserved but not yet used is currently in process
dateReserved	String(20)	If reserved=true then the date that the voucher was reserved in the format yyyy-MM-dd HH:mm:ss
used	Boolean	If hasFault=False, if the voucher has been used.
dateUsed	String(20)	If used=true then the date the voucher was used in the format yyyy-

		MM-dd HH:mm:ss
usedByReason	String(255)	If used=true, then an explanation as to what the voucher was used for (copied from the reason in the Redeem Request)

Http response codes will be returned as follows

Http Response Code	Reason
200	Successful response (hasFault=false)
400	Response is a Failure (hasFault=true)
401	Login Failed (hasFault=true)
500	System Exception (unable to process the request) (hasFault=true)

Example 6: Successful Request / Response for unused voucher

Request	Response
POST /Service1.svc/AnyTimeVoucherLookup HTTP/1.1 Authorization: Basic VEVTVDPURVNU Content-Length: 102 Content-Type: application/json Host: desertretail.co.za:19002 <pre>{ "pinNumber": "55668899001", "terminalID": "selwind@gmail.com", "msgID": "20220311123825000001", }</pre>	HTTP/1.1 200 cache-control: private content-type: application/json; charset=utf-8 server: Microsoft-IIS/10.0 x-aspnet-version: 4.0.30319 x-powered-by: ASP.NET date: Fri, 11 Mar 2022 10:41:51 GMT content-length: 119 <pre>{ "amount": "5.000", "hasFault": false, "msgID": "20220311123825000001", "reserved": false, "serial": "99995093062", "used": false }</pre>
POST /Service1.svc/AnyTimeVoucherLookup HTTP/1.1 Authorization: Basic VEVTVDPURVNU Content-Length: 99 Content-Type: application/json Host: desertretail.co.za:19002 <pre>{ "serial": "99995093062", "terminalID": "selwind@gmail.com", "msgID": "20220311123825000002", }</pre>	HTTP/1.1 200 cache-control: private content-type: application/json; charset=utf-8 server: Microsoft-IIS/10.0 x-aspnet-version: 4.0.30319 x-powered-by: ASP.NET date: Fri, 11 Mar 2022 10:42:40 GMT content-length: 135 <pre>{ "amount": "0.000", "hasFault": false, "msgID": "20220311123825000002", "pinNumber": "0", "reserved": false, "serial": "99995093062", "used": false }</pre>

Use case shows the credentials Username = TEST and Password = TEST being used with the Authorization Header

Example 7: Successful Request / Response for used voucher

Request	Response
POST /Service1.svc/AnyTimeVoucherLookup HTTP/1.1 Authorization: Basic VEVTVDPURVNU Content-Length: 102 Content-Type: application/json Host: desertretail.co.za:19002 <pre>{ "pinNumber": "12348888881", "terminalID": "selwind@gmail.com", "msgID": "20220311123826000001", }</pre>	HTTP/1.1 200 cache-control: private content-type: application/json; charset=utf-8 server: Microsoft-IIS/10.0 x-aspnet-version: 4.0.30319 x-powered-by: ASP.NET date: Fri, 11 Mar 2022 10:43:52 GMT content-length: 282 <pre>{ "amount": "1.000", "dateReserved": "2022-03-01 12:56:24", "dateUsed": "2022-03-01 12:56:25", "hasFault": false, "msgID": "20220311123826000001", "pinNumber": "12348888881", "reserved": true, "serial": "99998300679", "used": true, }</pre>

	"usedByReason": "V_RECHARGE_AIRTIME + TELKOM (R50.00) for 0813236726"} <td></td>	
--	--	--

Use case shows the credentials Username = TEST and Password = TEST being used with the Authorization Header

On the test server any PinNumber that is not of length 11 or starts with "1111" will return a "Voucher not found" error

Any pin number that is of length 11 and starts with "1234" will return a "used=true" voucher

Any pin number that is of length 11 and does not start with "1234" will return an "unused" voucher

4. Advice Method

The Advice method is used to obtain the result of a previously sent request.

The Advice scenario should be initiated if

1. Response is not received (due to timeout / network error) and it cannot be certain if request was received by RAVASVend
2. Response is received that hasFault = true and faultALR=true (for either an Advice Request or AnyTimeVoucherRedeem)

Endpoint is: /Advice

(for example <https://desertretail.co.za:19002/Service1.svc/Advice>)

Request Parameters are:

Name	Type (maxlength)	Description
msgID	String(45)	Unique identifier of each request (can be UUID or incrementing int or datetime with random number) but must be unique for each request
adviceMsgID	String(45)	Unique identifier of the request to which response is required.
terminalID	String(45)	Unique identifier as to who I using the voucher, this could be a MSISDN / email / Account Number / Account Code / Anything that uniquely identifies the consumer in the Clients Scope.

Response Parameters are the same as AnyTimeVoucherRedeem request.

Example 8: Advice for Example 5

Request	Response
POST /Service1.svc/Advice HTTP/1.1 Authorization: Basic VEVTVDPURVNU Content-Length: 113 Content-Type: application/json Host: desertretail.co.za:19002 { "terminalID":"selwind@gmail.com", "msgID":"20220218131005000006", "adviceMsgID":"20220218131005000005", }	HTTP/1.1 400 cache-control: private content-type: application/json; charset=utf-8 server: Microsoft-IIS/10.0 x-aspnet-version: 4.0.30319 x-powered-by: ASP.NET date: Fri, 18 Feb 2022 11:28:41 GMT content-length: 161 { "faultDesc":"Error Using AnyTime Voucher, Please try again!", "faultMsg":"Voucher is reserved", "faultNumber":"91", "hasFault":true, "msgID":"20220218131005000005"} }

Notice response has hasFault=true but faultALR is missing (thus false) so the response is final and no further Advice requests are required.

Example 9: Advice for Example 1

Assume that response for Example 1 was not received, Advice request would thus be sent to get the response

Request	Response
POST /Service1.svc/Advice HTTP/1.1 Authorization: Basic VEVTVDPURVNU Content-Length: 113 Content-Type: application/json Host: desertretail.co.za:19002 {	HTTP/1.1 200 cache-control: private content-type: application/json; charset=utf-8 server: Microsoft-IIS/10.0 x-aspnet-version: 4.0.30319 x-powered-by: ASP.NET

<pre>"terminalID":"selwind@gmail.com", "msgID":"20220218131005000007", "adviceMsgID":"20220218131005000001", }</pre>	<pre>date: Fri, 18 Feb 2022 11:33:56 GMT content-length: 153 {"amount":"179.000", "msgID":"20220218131005000001", "paymentFee":"0.000", "pinNumber":"12341234556", "reference":"7836", "serial":"99991461247", "tID":"7836"}</pre>
--	--

Response is exactly the same as received for example 1.

Example 10: Advice for a MsgID that does not exist on the server

Assume a request was sent to the server, but RAVASVend did not receive it so it has no data for the supplied adviceMsgID

Request	Response
<pre>POST /Service1.svc/Advice HTTP/1.1 Authorization: Basic VEVTVDPURVNU Content-Length: 112 Content-Type: application/json Host: desertretail.co.za:19002 { "terminalID":"selwind@gmail.com", "msgID":"20220218131005000008", "adviceMsgID":"2022021813100500XXXX", }</pre>	<pre>HTTP/1.1 400 cache-control: private content-type: application/json; charset=utf-8 server: Microsoft-IIS/10.0 x-aspnet-version: 4.0.30319 x-powered-by: ASP.NET date: Fri, 18 Feb 2022 11:35:42 GMT content-length: 189 {"faultDesc":"Request Not Received", "faultMsg":"The Request was never received by the server. Normal Vending can continue", "faultNumber":"38", "hasFault":true, "msgID":"20220218131005000008"}</pre>

The Test server has the following static use cases that can be used

Test Case #	Pin Number	Description (Expected Response)
1	Length !=11 or starts with 1111	Pin Not found (Does not exist)
2	Pin Starts with 2222 & (length == 11)	No response from Utility (mustALR=true) this Advice is required but advice will return a successful response (Voucher was redeemed)
3	Pin Starts with 3333 & (length=11)	No response from Utility (mustALR=true) thus Advice is required but advice will return a response that failed
4	Any Other pin that has length=11 and does not start with above test cases	Successful response