



VICTORYlink SMS Sending Web Service

Technical Documentation

Created by
Technical VAS Team

THE "NO PROBLEM" PEOPLE

Contents

Revision History3

1. Purpose4

2. Audience4

3. Introduction.....4

4. Web Service Main Web Methods:5

4.1 SendSMS5

4.1.1 Parameters of the web method:5

4.1.2 Return values of the web method:5

4.1.3 Service reference:.....6

4.1.4 Service reference:.....6

4.1.4.1 By adding Web Reference6

4.1.4.2 Through HTTP Request as the below C# code7

4.1.4.3 Through JAVA application8

4.1.4.4 Through PHP9

4.2 SendSMSWithDLR.....10

4.2.1 Parameters of the web method:10

4.2.2 Return values of the web method:10

4.2.3 Service reference:.....10

4.2.4 Service reference:.....11

4.2.4.1 By adding Web Reference11

4.2.4.2 Through HTTP Request as the below C# code12

4.2.4.3 Through JAVA application13

4.2.4.4 Through PHP14

4.3 CheckCredit15

4.3.1 Parameters of the web method:15

4.3.2 Return values of the web method:15

Revision History

Date	Version	Author	Description
06/11/2018	V1.0	Amr Farrag	Initial version

1. Purpose

- This document describes how to use Victory link SMS Sending Web Service.

2. Audience

- The intended reader of this document is anyone that wishes to use the main methods in Victory link SMS Sending Web Service.

3. Introduction

- Victory link SMS Sending Web Service is a service that is provided by Victory link to send SMSs.

4. Web Service Main Web Methods:

4.1 SendSMS

Used to send single SMS to single mobile number.

4.1.1 Parameters of the web method:

Parameter	Description	Data Type
Username	Username provided by victory link	string
Password	Password provided by victory link	string
SMSText	The message text e.g. "This is a test message"	string
SMSLang	The message language <ul style="list-style-type: none">• "E" for English• "A" for Arabic	string
SMSSender	The fake sender, it should be less than 12 characters, e.g. Test	string
SMSReceiver	The receiver mobile number, E.g. 01xxxxxxxxx	string

4.1.2 Return values of the web method:

This section is intended to describe the return values for the web method and their meanings; the web method returns an XML document as below containing an integer that represents the result:

```
<int xmlns="http://tempuri.org/">0</int>
```

Return Value	Meaning
0	The SMS is sent successfully.
-1	User is not subscribed
-5	out of credit.
-10	Queued Message, no need to send it again.
-11	Invalid language.
-12	SMS is empty.
-13	Invalid fake sender exceeded 12 chars or empty.
-25	Sending rate greater than receiving rate (only for send/receive accounts).
-100	other error

4.1.3 Service reference:

<https://smsvas.vlserv.com/KannelSending/service.asmx>

4.1.4 Service reference:

To consume the web service within your application, you can use one of the below ways:

You can use this web service and integrate it with your application. This section, you will find the code integration of this tool using:

- By adding web reference
- Through HTTP Request C# code
- Through Java Application
- Through PHP code

4.1.4.1 By adding Web Reference

Now we need to add a web reference to our application. To do this, right-click on your project and select Add Service Reference

After adding the service reference, instantiate an object of the service as the below C# code

```
KannelSendingService.Service kannelSendingServiceObj = new  
KannelSendingService.Service();  
kannelSendingServiceObj.SendSMS("username", "password", "text", "e",  
"sender",
```

4.1.4.2 Through HTTP Request as the below C# code

```
string uri =
string.Format("https://smsvas.vlserv.com/KannelSending/service.asmx/Send
SMS?username={0}&p
assword={1}&SMSText={2}&SMSLang={3}&SMSSender={4}&SMSReceiver={5}",
"username", "password", "text", "e", "sender", "01xxxxxxxxx");

getMessage(uri);
private HttpWebRequest CreateWebRequest(string endPoint)
{
    var request = (HttpWebRequest)WebRequest.Create(endPoint);
    request.Method = "Get";
    request.ContentLength = 0;
    request.ContentType = "text/xml";
    return request;
}
private string getMessage(string endPoint)
{
    HttpWebRequest request = CreateWebRequest(endPoint);
    using (var response = (HttpWebResponse)request.GetResponse())
    {
        var responseValue = string.Empty;
        if (response.StatusCode != HttpStatusCode.OK)
        {
            string message = string.Format("POST failed.
Received HTTP {0}", response.StatusCode);
            throw new ApplicationException(message);
        }
        //grab the response
        using (var responseStream = response.GetResponseStream())
        {
            using (var reader = new StreamReader(responseStream))
            {
                responseValue = reader.ReadToEnd();
            }
        }
    }
    return responseValue;
}
```

4.1.4.3 Through JAVA application

```
Public class main {

    Private final String USER_AGENT = "Mozilla/5.0";
    Public static void main(String[] args) throws
    Exception { main http = new main();

    System.out.println("Testing 1 - Send Http GET
    request"); http.sendGet();

    System.out.println("Testing 2 - Send Http POST request");
    http.sendPost();
    }

    private void sendGet() throws Exception
    {
        StringBuilder address = new StringBuilder();
        address.append(("https://smsvas.vlserv.com/KannelSending/service.asm
        x/SendSMS ?"));
        address.append("username=" + "username");
        address.append("&password=" + "password");
        address.append("&SMSText=" + "text");
        address.append("&SMSLang=" + "e");
        address.append("&SMSSENDER=" + "sender");
        address.append("&SMSReceiver=" + "
        01xxxxxxxxx"); String url =
        address.toString();

        URL obj = new URL(url);
        HTTPURLConnection con = (HTTPURLConnection) obj.openConnection();

        //optional default is GET
        con.setRequestMethod("GET");

        System.out.println("\nSending 'GET' request to URL: "
        + url); BufferedReader in = new BufferedReader(new
        InputStreamReader(con.getInputStream())); String
        inputLine;

        StringBuffer response = new
        StringBuffer(); while ((inputLine =
        in.readLine()) != null) {
            response.append(inputLine);
        }
        in.close();

        //print result
        System.out.println(response.toString());
    }
}
```


4.1.4.4 Through PHP

```
<?php
    use SoapClient;
    $client = new
SoapClient("https://smsvas.vlserv.com/KannelSening/service.asmx"); $userName =
    "username";
    $Password = "password";
    $SMSText = "text";
    $SMSLang = "e";
    $SMSSender = "sender";
    $SMSReceiver = "01xxxxxxxxx";
    $result = $client->SendSMS(array(
        "Username" => $userName,
        "Password" => $Password,
        "SMSText" => $SMSText,
        "SMSLang" => $SMSLang,
        "SMSSender" => $SMSSender,
        "SMSReceiver" => $SMSReceiver));
    $response_arr = objectToArray($result);
    echo "return_code= " . str_replace(";", "",
    $response_arr);
    function objectToArray($d)
    {
        if (is_object($d))
        {$d = get_object_vars($d);}
        if (is_array($d))
        {return array_map(__FUNCTION__, $d);}
        else {return $d;}
    }
    ?>
```

Note:

- We're using **SoapClient** PHP to call the API Web Service, so you need to enable the **libxml** PHP extension.
- As its response is an object (not string) so you need to convert it to an array by using **objectToArray** function.

4.2 SendSMSWithDLR

Used to send single SMS to single mobile number and the delivery report will be reported in a separate reporting tool.

4.2.1 Parameters of the web method:

Parameter	Description	Data Type
Username	Username provided by victory link	string
Password	Password provided by victory link	string
SMSText	The message text e.g. "This is a test message"	string
SMSLang	The message language <ul style="list-style-type: none">• "E" for English• "A" for Arabic	string
SMSSender	The fake sender, it should be less than 12 characters, e.g. Test	string
SMSReceiver	The receiver mobile number, E.g. 01xxxxxxxxx	string

4.2.2 Return values of the web method:

This section is intended to describe the return values for the web method and their meanings; the web method returns an XML document as below containing an integer that represents the result:

```
<int xmlns="http://tempuri.org/">0</int>
```

Return Value	Meaning
0	The SMS is sent successfully.
-1	User is not subscribed
-5	out of credit.
-10	Queued Message, no need to send it again.
-11	Invalid language.
-12	SMS is empty.
-13	Invalid fake sender exceeded 12 chars or empty.
-25	Sending rate greater than receiving rate (only for send/receive accounts).
-100	other error

4.2.3 Service reference:

<https://smsvas.vlserv.com/KannelSending/service.asmx>

4.2.4 Service reference:

To consume the web service within your application, you can use one of the below ways:

You can use this web service and integrate it with your application. This section, you will find the code integration of this tool using:

- By adding web reference
- Through HTTP Request C# code
- Through Java Application
- Through PHP code

4.2.4.1 By adding Web Reference

Now we need to add a web reference to our application. To do this, right-click on your project and select Add Service Reference

After adding the service reference, instantiate an object of the service as the below C# code

```
KannelSendingService.Service kannelSendingServiceObj = new  
KannelSendingService.Service();  
kannelSendingServiceObj.SendSMSWithDLR("username", "password", "text",  
"e", "sender",
```

4.2.4.2 Through HTTP Request as the below C# code

```
string uri =
string.Format("https://smsvas.vlserv.com/KannelSending/service.asmx/
SendSMSWithDLR?username={0}&p
assword={1}&SMSText={2}&SMSLang={3}&SMSSender={4}&SMSReceiver={5}",
"username", "password", "text", "e", "sender", "01xxxxxxxxx");

getMessage(uri);
private HttpWebRequest CreateWebRequest(string endPoint)
{
    var request = (HttpWebRequest)WebRequest.Create(endPoint);
    request.Method = "Get";
    request.ContentLength = 0;
    request.ContentType = "text/xml";
    return request;
}
private string getMessage(string endPoint)
{
    HttpWebRequest request = CreateWebRequest(endPoint);
    using (var response = (HttpWebResponse)request.GetResponse())
    {
        var responseValue = string.Empty;
        if (response.StatusCode != HttpStatusCode.OK)
        {
            string message = string.Format("POST failed.
Received HTTP {0}", response.StatusCode);
            throw new ApplicationException(message);
        }
        //grab the response
        using (var responseStream = response.GetResponseStream())
        {
            using (var reader = new StreamReader(responseStream))
            {
                responseValue = reader.ReadToEnd();
            }
        }
        return responseValue;
    }
}
```

4.2.4.3 Through JAVA application

```
Public class main {

    Private final String USER_AGENT = "Mozilla/5.0";
    Public static void main(String[] args) throws
    Exception { main http = new main();

    System.out.println("Testing 1 - Send Http GET
    request"); http.sendGet();

    System.out.println("Testing 2 - Send Http POST request");
    http.sendPost();
    }

    private void sendGet() throws Exception
    {
        StringBuilder address = new StringBuilder();
        address.append\(\("https://smsvas.vlserv.com/KannelSending/service.aspx/SendSMSWithDLR ?"\)\);
        address.append("username=" + "username");
        address.append("&password=" + "password");
        address.append("&SMSText=" + "text");
        address.append("&SMSLang=" + "e");
        address.append("&SMSSENDER=" + "sender");
        address.append("&SMSReceiver=" + "
        01xxxxxxxxx"); String url =
        address.toString();

        URL obj = new URL(url);
        HTTPURLConnection con = (HTTPURLConnection) obj.openConnection();

        //optional default is GET
        con.setRequestMethod("GET");

        System.out.println("\nSending 'GET' request to URL: "
        + url); BufferedReader in = new BufferedReader(new
        InputStreamReader(con.getInputStream())); String
        inputLine;

        StringBuffer response = new
        StringBuffer(); while ((inputLine =
        in.readLine()) != null) {
            response.append(inputLine);
        }
        in.close();

        //print result
        System.out.println(response.toString());
    }
}
```

4.2.4.4 Through PHP

```
<?php
    use SoapClient;
    $client = new
SoapClient("https://smsvas.vlserv.com/KannelSending/service.asmx"); $userName =
    "username";
    $Password = "password";
    $SMSText = "text";
    $SMSLang = "e";
    $SMSSender = "sender";
    $SMSReceiver = "01xxxxxxxxx";
    $result = $client->SendSMSWithDLR(array(
        "Username" => $userName,
        "Password" => $Password,
        "SMSText" => $SMSText,
        "SMSLang" => $SMSLang,
        "SMSSender" => $SMSSender,
        "SMSReceiver" => $SMSReceiver));
    $response_arr = objectToArray($result);
    echo "return_code= " . str_replace(";", "",
    $response_arr);
    function objectToArray($d)
    {
        if (is_object($d))
        {$d = get_object_vars($d);}
        if (is_array($d))
        {return array_map(__FUNCTION__, $d);}
        else {return $d;}
    }
?>
```

Note:

- We're using **SoapClient** PHP to call the API Web Service, so you need to enable the **libxml** PHP extension.
- As its response is an object (not string) so you need to convert it to an array by using **objectToArray** function.

4.3 CheckCredit

It is used to check the account available credit in case of limited accounts. It returns the number of SMSs available to be sent.

4.3.1 Parameters of the web method:

Parameter	Description	Data Type
Username	Username provided by victory link	string
Password	Password provided by victory link	string

4.3.2 Return values of the web method:

It returns an integer that represents the number of SMSs available to be sent. It returns 0 in case of unlimited accounts.