WIGAL SOLUTIONS LTD
Behind Taxi Rank, Ritz Junction on Madina - Adenta Road
Box VV 26 Valley View, Oyibi
Ghana-West Africa
Tel 233276128036/233266000751
www.wigalsolutions.com

BULK SMS API FOR DEVELOPERS

Table of Contents

Introduction

This Document gives details on how to send messages via the WIGAL Solutions BULK SMS Gateway from your application.

You must register with WIGAL Solutions and use your credentials to send the messages through the HTTP API.

HTTP Parameters for Sending Message

Our HTTP API requires developers to send below details: to our bulk SMS API endpoint **http://isms.wigalsolutions.com/ismsweb/sendmsg/**

Parameter	Description
from	The sender ID. Max is 11 characters
to	The recipient(s). Use (,) to separate multiple recipients. The numbers must be in international format
message	The content of message to send (Can be used for 'long' messages, that is, messages longer than 160 characters for plain text)
username	Your iSMS username
password	Your iSMS Password

POSSIBLE VALUES TO BE RETURNED BY API

TYPE	Actual Message returned by API
SUCCESS	SUCCESS :: Message Sent Successfully
ERROR	ERROR :: Invalid Login credentials
ERROR	ERROR :: Destination number Invalid
ERROR	ERROR :: Invalid From. Value should not be more than 11 chars including space
ERROR	ERROR :: 1 or more Destination numbers Invalid
ERROR	ERROR :: Message Could not be sent
ERROR	ERROR :: Not Enough Balance to send {message length} message(s)

EXAMPLES:

Example 1: Sending a message to a single recipient

http://isms.wigalsolutions.com/ismsweb/sendmsg/?username=yourusername**&password**=yourpassword**&from**=WIGAL**&to=233276128036**&**message**=Hello my test message

Example 2: Sending a message to multiple recipient

http://isms.wigalsolutions.com/ismsweb/sendmsg/?username=yourusername**&password**=yourpassword**&from**=WIGAL**&to=233276128036,233266000751**&**message**=Hello my test message

HTTP Parameters for Checking Balance

Find URL for balance checking below:

http://isms.wigalsolutions.com/ismsweb/Query/

Parameter	Description
username	Your iSMS username
password	Your iSMS Password
query	The type of query you want to send to our gateway For Balance checking use '**balance'**

The API returns an integer representing your current balance

EXAMPLE:

http://isms.wigalsolutions.com/ismsweb/Query/?username=yourusername**&password=**yourpassword**&query=balance**

EXAMPLE HTTP CALLS TO THE API

PHP

```php
<?php
        $username = 'your username';
        $password ='your password';
        $message = "Demo message";
        $from = "WIGAL";//your senderid example "kwamena" max is 11 chars;
        $baseurl = "http://isms.wigalsolutions.com/ismsweb/sendmsg/";

        // All numbers must have country code. delimit them with comma(,)
        $to = '233276128036,2332545878,2332548787';
        $params = "username=".$username."&password=".$password."&from=".$from
        ."&to=".$to."&message=".$message ;
        // send the message
        $ch = curl_init();
        curl_setopt($ch,CURLOPT_URL,$baseurl);
        curl_setopt($ch,CURLOPT_RETURNTRANSFER,1);
        curl_setopt($ch,CURLOPT_POST,1);
        curl_setopt($ch,CURLOPT_POSTFIELDS,$params);
        $return=curl_exec($ch);
        curl_close($ch);
        $send = explode(" :: ",$return);
        if (stristr($send[0],"SUCCESS") != FALSE)
        {
            echo "message sent";
        }
        else
        {
            echo "message could not be sent";
        }
?>
```

Checking Balance

```php
<?php
        $username = 'your username';
        $password ='your password';
        $baseurl = "http://isms.wigalsolutions.com/ismsweb/Query/";
        $params = "username=".$username."&password=".$password."&query=balance";
        // do the check balance  call
        $balance = file($baseurl.$params);
        echo $balance;
?>
```

C#

```
using System.IO;
using System.Net;
using System.Security.Cryptography.X509Certificates;
using System.Net.Security;
using System;
using System.Text;


class iSMSAPI
{
    /// <summary>
    /// Used to send message
    /// </summary>
    /// <param name="from">The sender ID. Max is 11 characters</param>
    /// <param name="to">The recipient(s). Use (,) to separate multiple recipients</param>
    /// <param name="message">The content of message to send </param>
    /// <param name="username">Your iSMS username</param>
    /// <param name="password">Your iSMS Password</param>
    /// <returns>Status of message submission to the SMS gateway</returns>
    public bool sendMessage(string from, string to, string message, string username, string password)
    {

        bool sucess = false;

        //use comma (,) to separate multiple recipients. remember that the numbers has to be in international format
        //example to = 23327612806,2335487899,233244587896,233266000751,233277014525
        WebRequest request;
        HttpWebResponse response;
        string responseMess = "";
        string apiparams = "username={0}&password={1}&from={2}&to={3}&message={4}";
        string URL = "http://isms.wigalsolutions.com/ismsweb/sendmsg/";
        apiparams = string.Format(apiparams, username, password, from, to, message);
        try
        {

            request = HttpWebRequest.Create(URL);
            request.Method = "POST";
            request.ContentType = "application/x-www-form-urlencoded";
            var data = Encoding.ASCII.GetBytes(apiparams);

            request.ContentLength = data.Length;
            Stream dataStream = request.GetRequestStream();
            dataStream.Write(data, 0, data.Length);
            dataStream.Close();
            request.Timeout = 5000;

            System.Net.ServicePointManager.ServerCertificateValidationCallback =
                delegate(object Certsender, X509Certificate certificate, X509Chain chain,
                SslPolicyErrors sslPolicyErrors) { return true; };

            response = (HttpWebResponse)request.GetResponse();
            StreamReader reader = new StreamReader(response.GetResponseStream());
            responseMess = reader.ReadToEnd();
            response.Close();
            if (responseMess.StartsWith("SUCCESS"))

                sucess = true;
            else
                sucess = false;

        }
        catch (Exception ex) { sucess = false; }


        return sucess;
    }
```

JAVA

```java
import java.io.BufferedReader;
import java.io.DataOutputStream;
import java.io.InputStreamReader;
import java.net.HttpURLConnection;
import java.net.URL;
import java.net.URLEncoder;

/**

* An Example Class to use for the submission using HTTP API You can perform
* your own validations into this Class For username, password,destination,
* source, type and message
**/
public class ISMS_HTTP {

// Username that is to be used for submission

String username;

// password that is to be used along with username

String password;

// Message content that is to be transmitted

String message;

/**

* What type of the message that is to be sent
* <ul>
* <li>text:means plain text</li>
* <li>flash:means flash</li>
* </ul>
*/
String type;

/**

* Destinations to which message is to be sent. For submitting more than one
* destination at once, destinations should be comma separated Like
* 233276128036,233266000751
*/
String to;


// ISMS_HTTP Sender ID to be used for submitting the message. Like WiGAL

String from;

// To what URL you need to connect to for submission
//This URL will change to isms.wigalsolutions.com/ismsweb/sendmsg soon!
final String server= "http://isms.wigalsolutions.com/ismsweb/sendmsg/";



public ISMS_HTTP(String username, String password,String message, String type, String destination, String source)
{

    this.username = username;

    this.password = password;

    this.message = message;

    this.type = type;
```

```
    this.to = destination;

    this.from = source;


}

    private boolean submitMessage()
    {
        boolean sucess = false;

       try
       {
           // Url that will be called to submit the message
           URL sendUrl = new URL(this.server);
           HttpURLConnection httpConnection = (HttpURLConnection) sendUrl .openConnection();

           // This method sets the method type to POST so that messages
           // will be sent as a POST request

           httpConnection.setRequestMethod("POST");

           // This method is set as true when we intend to send
           // input to the server

           httpConnection.setDoInput(true);

           // This method implies that we intend to receive data from server.

           httpConnection.setDoOutput(true);

           // Implies do not use cached data


           httpConnection.setUseCaches(false);

           // Data that will be sent over the stream to the server.

           DataOutputStream dataStreamToServer = new DataOutputStream( httpConnection.getOutputStream());
           dataStreamToServer.writeBytes("username=" + URLEncoder.encode(this.username, "UTF-8") +
                   "&password=" + URLEncoder.encode(this.password, "UTF-8") +
                   "&type=" + URLEncoder.encode(this.type, "UTF-8") +
                   "&from=" + URLEncoder.encode(this.from, "UTF-8") +
                   "&to=" + URLEncoder.encode(this.to, "UTF-8") +
                   "&message=" + URLEncoder.encode(this.message, "UTF-8"));
           dataStreamToServer.flush();
           dataStreamToServer.close();
           // Here take the output value of the server.

           BufferedReader dataStreamFromUrl = new BufferedReader(new InputStreamReader(httpConnection.getInputStream()));
           String dataFromUrl = "", dataBuffer = "";

           // Writing information from the stream to the buffer
           while ((dataBuffer = dataStreamFromUrl.readLine()) != null)
           {
               dataFromUrl += dataBuffer;
           }

           /**

           * Now dataFromUrl variable contains the Response received from the
           * server so we can parse the response and process it accordingly.
           */
           dataStreamFromUrl.close();
          //  System.out.println("Response: " + dataFromUrl);
```

```
            if(dataFromUrl.startsWith("SUCCESS"))
            {
                sucess = true;
            }

        } catch (Exception ex) {
            sucess = false;
            ex.printStackTrace();
        }

        return sucess;
    }
    public static void main(String[] args)
    {
        try
        {
            //Below is an example call to send a single SMS
            ISMS_HTTP s = new ISMS_HTTP("yourusername","yourpassword","Testing the Java API","text","233276128036,233241458112","WIGAL");
            s.submitMessage();

        } catch(Exception ex){ /*Handel your exception here*/ }

    }
}
```