RESTful API code examples

PHP JavaC#.Net

```
<?php
// Send SMS Example
class SendSMS
{
 private $url = 'https://api.textmarketer.co.uk/services/rest/sms'; // url of the
service
 private $username = 'myAPIusername'; // CHANGE THIS!!!
 private $password = 'myAPIpassword'; // CHANGE THIS!!
 private $message_id,$credits_used;
 function __construct()
 {
 }
 public function getMessageID()
 {
  return $this->message_id;
 }
 public function getCreditsUsed()
 {
  return $this->credits_used;
 }
 // public function to commit the send
 public function send($message,$mobile,$originator)
 {
  $url_array= array('message'=>$message,'mobile_number'=>$mobile,
'originator'=>$originator,
  'username'=>$this->username, 'password'=>$this->password);
  $url_string = $data = http_build_query($url_array, '', '&');
  // we're using the curl library to make the request
  $curlHandle = curl_init();
  curl_setopt($curlHandle, CURLOPT_URL, $this->url);
  curl_setopt($curlHandle, CURLOPT_RETURNTRANSFER, true);
  curl_setopt($curlHandle, CURLOPT_POSTFIELDS, $url_string);
  curl_setopt($curlHandle, CURLOPT_POST, 1);
  $responseBody = curl_exec($curlHandle);
  $responseInfo  = curl_getinfo($curlHandle);
  curl_close($curlHandle);
  return $this->handleResponse($responseBody,$responseInfo);
 }
 private function handleResponse($body,$info)
 {
  if ($info['http_code']==200){ // successful submission
   $xml_obj = simplexml_load_string($body);
   // extract message id and credit usuage
   $this->message_id = (int) $xml_obj->message_id;
   $this->credits_used = (int) $xml_obj->credits_used;
   return true;
  }
  else{
   $this->message_id = null;
   $this->credits_used = null;
   // error handling
   return false;
```

```
  }
 }
}
/*
 * Example of use
 * Remember to change the username and password!
 */
/*
$sms = new SendSMS();
if($sms->send("hello this is a test",'07712345678',"Achme Ltd")) echo "Yay, sent!";
```

```
else echo "Boo, not sent";
*/
?>
```

```
import java.io.DataOutputStream;
import java.io.InputStream;
import java.io.UnsupportedEncodingException;
import java.net.HttpURLConnection;
import java.net.URL;
import java.net.URLEncoder;
import java.text.SimpleDateFormat;
import java.util.Date;
import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;
import org.w3c.dom.Document;
import org.w3c.dom.Node;
import org.w3c.dom.NodeList;
import com.textmarketer.Example.SMSResult.TextMarketerError;
public class Example {
 /**
  * Example code to demonstrate the use of the SendSms(...) static method.
  */
 public static void main(String[] args) {
  //Replace these parameters with your own
  SMSResult result = sendSms("Java Test", "077777777", "SenderId", "username",
"password");

  if (result.errors == null) {
   //Show the results of a successful request
   System.out.println("SMS sent OK on " + result.processedDate.toString() + ".
MessageID=" + result.messageId + ", Credits Used=" + result.creditsUsed);
  }
  else {
   //Show the results on an unsuccessful request
   System.out.println("SMS sending failed with the following errors:");
   for (TextMarketerError error : result.errors) {
    System.out.println("Code: " + error.errorCode + "  Text: " + error.errorText);
   }
  }
 }

 /**
  * This method sends an SMS request to the Text Marketer SMS web service, and
returns details of the result of the request.
  * @param message The body of the text message to send
  * @param mobileNumber The phone number to send the text message to
  * @param originator Text or a phone number, to indicate who the sender is to the
recipient
  * @param username The username of your Text Marketer account
  * @param password The password of your Text Marketer account
  * @return An object representing the result of the call to the web service
  */
 public static SMSResult sendSms(String message, String mobileNumber, String
originator, String username, String password) {
  // Create a string representing the data elements to post, then convert it to a
byte array
```

```
  StringBuilder postData = new StringBuilder();
  try {
   postData.append("message=").append(URLEncoder.encode(message, "UTF-8"));
   postData.append("&mobile_number=").append(URLEncoder.encode(mobileNumber,
"UTF-8"));
   postData.append("&originator=").append(URLEncoder.encode(originator, "UTF-8"));
   postData.append("&username=").append(URLEncoder.encode(username, "UTF-8"));
   postData.append("&password=").append(URLEncoder.encode(password, "UTF-8"));
  }
  catch (UnsupportedEncodingException e) {
  }
  String postDataStr = postData.toString();
  URL url;
  HttpURLConnection connection = null;
  try {
   // Create connection
   url = new URL("https://api.textmarketer.co.uk/services/rest/sms");
   connection = (HttpURLConnection) url.openConnection();
   connection.setRequestMethod("POST");
   connection.setRequestProperty("Content-Type",
"application/x-www-form-urlencoded");
   connection.setRequestProperty("Content-Length", "" +
Integer.toString(postDataStr.getBytes().length));
   connection.setUseCaches(false);
   connection.setDoInput(true);
   connection.setDoOutput(true);
   // Send request
   DataOutputStream wr = new DataOutputStream(connection.getOutputStream());
   wr.writeBytes(postDataStr);
   wr.flush();
   wr.close();
   // Get Response
   InputStream inputStreams = connection.getInputStream();
   // Convert the response data into an XML document and parse the expected
   // elements to get the individual response fields
   DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
   DocumentBuilder builder = factory.newDocumentBuilder();
   Document document = builder.parse(inputStreams);
   String date =
document.getElementsByTagName("response").item(0).getAttributes().getNamedItem("proc
essed_date").getNodeValue();
   String reFromattedDate = date.substring(0, 22) + date.substring(23);
   SimpleDateFormat format = new SimpleDateFormat("yyyy-MM-dd'T'HH:mm:ssZ");
   Date processedDate = format.parse(reFromattedDate);
   String messageId =
document.getElementsByTagName("message_id").item(0).getFirstChild().getNodeValue();
   int creditsUsed =
Integer.parseInt(document.getElementsByTagName("credits_used").item(0).getFirstChild
().getNodeValue());
   SMSResult result = new SMSResult(processedDate, messageId, creditsUsed, null);
   return result;
  }
  catch (Exception e) {
   try {
    InputStream errorStream = connection.getErrorStream();
    // Convert the response data into an XML document and parse the expected
    // error elements to get the individual response fields
    DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
    DocumentBuilder builder = factory.newDocumentBuilder();
```

```
    Document document = builder.parse(errorStream);
    String date =
document.getElementsByTagName("response").item(0).getAttributes().getNamedItem("proc
essed_date").getNodeValue();
    String reFromattedDate = date.substring(0, 22) + date.substring(23);
    SimpleDateFormat format = new SimpleDateFormat("yyyy-MM-dd'T'HH:mm:ssZ");
    Date processedDate = format.parse(reFromattedDate);
    NodeList errorNodeList = document.getElementsByTagName("error");
    TextMarketerError[] errors = new TextMarketerError[errorNodeList.getLength()];
    for (int i = 0; i < errorNodeList.getLength(); i++) {
     Node errorNode = errorNodeList.item(i);
     String errorCode =
errorNode.getAttributes().getNamedItem("code").getNodeValue();
     String errorText = errorNode.getFirstChild().getNodeValue();
     TextMarketerError error = new TextMarketerError(Integer.parseInt(errorCode),
errorText);
     errors[i] = error;
    }
    SMSResult result = new SMSResult(processedDate, null, 0, errors);
    return result;
   }
   catch (Exception e2) {
   }
  }
  finally {
   if (connection != null) {
    connection.disconnect();
   }
  }
  return null;
 }
 /**
  * Encapsulates the results of a call to the Text Marketer web service.
  */
 public static class SMSResult {
  public Date processedDate;
  public String messageId;
  public int creditsUsed;
  public TextMarketerError[] errors;
  public SMSResult(Date processedDate, String messageId, int creditsUsed,
TextMarketerError[] errors) {
   this.processedDate = processedDate;
   this.messageId = messageId;
   this.creditsUsed = creditsUsed;
   this.errors = errors;
  }
  public static class TextMarketerError {
   public int errorCode;
   public String errorText;
   public TextMarketerError(int errorCode, String errorText) {
    this.errorCode = errorCode;
    this.errorText = errorText;
   }
```

```
  }
 }
}
```

```
using System;
using System.Text;
using System.Net;
using System.IO;
using System.Xml;
using System.Collections;
namespace TextMarketer {
 class Example {
  /// <summary>
  /// Example code to demonstrate the use of the SendSms(...) static method.
  /// </summary>
  /// <param name="args"></param>
  static void Main(string[] args) {
   //Declare the OUT parameters that are used to return the results of the SendSms
request
   DateTime processedDate;
   string messageId;
   int creditsUsed;
   TextMarketerError[] errors;
   //Replace these parameters with your own
   bool success = SendSms("This is my message", "018118055", "me", "myUserName",
"myPassword", out processedDate, out messageId, out creditsUsed, out errors);
   if (success) {
    //Show the results of a successful request
    Console.WriteLine("SMS sent OK on " + processedDate.ToString() + ". MessageID="
+ messageId + ", Credits Used=" + creditsUsed);
   }
   else {
    //Show the results on an unsuccessful request
    Console.WriteLine("SMS sending failed with the following errors:");
    foreach (TextMarketerError error in errors) {
     Console.WriteLine("Code: " + error.errorCode + "  Text: " + error.errorText);
    }
   }
  }
  /// <summary>
  /// This method sends an SMS request to the Text Marketer SMS web service, and
returns details of the result of the request.
  /// </summary>
  /// <param name="message">The body of the text message to send</param>
  /// <param name="mobileNumber">The phone number to send the text message
to</param>
  /// <param name="originator">Text or a phone number, to indicate who the sender is
to the recipient</param>
  /// <param name="username">The username of your Text Marketer account</param>
  /// <param name="password">The password of your Text Marketer account</param>
  /// <param name="processedDate">Returns the date and time at which the server
processed your request</param>
  /// <param name="messageId">Returns a unique Id by which the server identifies the
message</param>
  /// <param name="creditsUsed">Returns the number of Text Marketer credits consumed
by sending the text message</param>
```

```
  /// <param name="errors"></param>Returns an array of TextMarketerError objects
identifying an errors in your request.
  /// <returns>True if the request was successful, false if the server could not
process your request (see 'errors' out-parameter for details)</returns>
  public static bool SendSms(string message, string mobileNumber, string originator,
string username, string password, out DateTime processedDate, out string messageId,
out int creditsUsed, out TextMarketerError[] errors) {
   //Create a string representing the data elements to post
   StringBuilder postData = new StringBuilder();
   postData.Append("message=").Append(message);
   postData.Append("&mobile_number=").Append(mobileNumber);
   postData.Append("&originator=").Append(originator);
   postData.Append("&username=").Append(username);
   postData.Append("&password=").Append(password);
   //Encode the data string into ASCII bytes
   ASCIIEncoding encoding = new ASCIIEncoding();
   byte[] data = encoding.GetBytes(postData.ToString());
   //Create the web request to the Text Marketer web service and attach the data
   HttpWebRequest webRequest =
(HttpWebRequest)WebRequest.Create("https://api.textmarketer.co.uk/services/rest/sms"
);
   webRequest.Method = "POST";
   webRequest.ContentType = "application/x-www-form-urlencoded";
   webRequest.ContentLength = data.Length;
   Stream newStream = webRequest.GetRequestStream();
   newStream.Write(data, 0, data.Length);
   newStream.Close();
   string resultString;
   try {
    //Action the post to the server and get the response data stream
    HttpWebResponse webResponse = (HttpWebResponse)webRequest.GetResponse();
    using (StreamReader streamReader = new
StreamReader(webResponse.GetResponseStream())) {
     resultString = streamReader.ReadToEnd();
     streamReader.Close();
    }
    //Convert the response data into an XML document and parse the expected elements
to get the individual response fields
    XmlDocument xmlDocument = new XmlDocument();
    xmlDocument.LoadXml(resultString);
    processedDate =
DateTime.Parse(xmlDocument.GetElementsByTagName("response")[0].Attributes.GetNamedIt
em("processed_date").Value);
    messageId = xmlDocument.GetElementsByTagName("message_id").Item(0).InnerText;
    creditsUsed =
Int32.Parse(xmlDocument.GetElementsByTagName("credits_used").Item(0).InnerText);
    errors = null;
    return true;
   }
   catch (WebException e) {
    //If we get here, the service returned a server error, so read the server
response from the exception
    using (StreamReader streamReader = new
StreamReader(e.Response.GetResponseStream())) {
     resultString = streamReader.ReadToEnd();
     streamReader.Close();
    }
    //Convert the response data into an XML document and parse the expected error
elements to get the individual response fields
```

```
    XmlDocument xmlDocument = new XmlDocument();
    xmlDocument.LoadXml(resultString);
    processedDate =
DateTime.Parse(xmlDocument.GetElementsByTagName("response")[0].Attributes.GetNamedIt
em("processed_date").Value);
    XmlNodeList errorsNodeList = xmlDocument.GetElementsByTagName("errors");
    XmlNodeList errorNodeList = errorsNodeList.Item(0).ChildNodes;
    errors = new TextMarketerError[errorNodeList.Count];
    IEnumerator errorNodeListEnum = errorNodeList.GetEnumerator();
    int i = 0;
    while (errorNodeListEnum.MoveNext()) {
     XmlNode errorNode = (XmlNode)errorNodeListEnum.Current;
     string errorCode = errorNode.Attributes.GetNamedItem("code").Value;
     string errorText = errorNode.InnerText;
     TextMarketerError error = new TextMarketerError(Int32.Parse(errorCode),
errorText);
     errors[i++] = error;
    }
    messageId = null;
    creditsUsed = 0;
    return false;
   }
  }
  /// <summary>
  /// Represents an individual error response element from the Text Marketer SMS
service
  /// </summary>
  public class TextMarketerError {
   public int errorCode;
   public String errorText;
   public TextMarketerError(int errorCode, string errorText) {
    this.errorCode = errorCode;
    this.errorText = errorText;
   }
```

```
    }
  }
}
```

You must know!
Our example code is an illustration of how you might integrate with our systems and is not certified for production environments. You are responsible for testing and QA.

- Next: Fully-validated XML and DTDs
- Home: Developers Documentation