



## Web Services API Integration Guidelines

**Web:** [www.smsconnexion.com](http://www.smsconnexion.com)

**Tel:** +91 22 42213500

**Email:** [support@celusion.com](mailto:support@celusion.com)

The information contained in this document is proprietary and copyright for the sole purpose of informing customers and partners of the above service. The service is owned by Celusion Technologies Pvt. Ltd., Mumbai, India

## Change History

<b>Document:</b>	Web Services API Integration Guidelines
<b>Version:</b>	1.0.0.0
<b>Created By:</b>	Praveen Paulose
<b>Created Date:</b>	19 Jan 2010
<b>Last Modified By:</b>	N/A
<b>Last Modified Date:</b>	N/A

<b>Date</b>	<b>Author</b>	<b>Remarks</b>
19 Jan 2010	Praveen Paulose	Initial Release

## Introduction

SMS ConneXion™, a CELUSION® enterprise offers a complete range of two-way interactive SMS solutions that allows you to rapidly deploy text messaging capability to your corporate information system.

Do you have a software system that needs to empower users with the functionality of SMS? We believe that the success of our products depend on mass adoption and have a created a vast range of software API's to simplify integration with your legacy systems. Our SOA (Service Oriented Architecture) has been implemented in various web portals, legacy software, core banking systems, SAP, Microsoft CRM and many more. We provide various modes of integration like HTTP, Web Services, XML File API and SMPP. Moreover to simplify your experience when integrating SMS ConneXion with your systems, we also provide samples in different programming languages.

This document contains detailed information on how to connect to SMS ConneXion gateways using the SOAP protocol via Web Services.

The API is platform / technology independent. You may integrate a Java, PHP, ASP, .NET or any other application using the Web Services API.

## Operations

### 1. Send SMS

In order to send SMS you need to initiate a Web Service call on the URL <http://smsc.smsconnexion.com/api/gateway.asmx>

*Note: Your URL may vary depending on the server on which your SMS ConneXion account is hosted.*

#### Method Definition:

```
int[] SendSMS(string username, string password, string[] allnumbers, string[] allsms, string[] allfrom)
```

#### Parameters:

1. **username {string}** – The account username created on the SMS ConneXion web site
2. **password {string}** – The account password for the above username on the SMS ConneXion web site
3. **allnumbers {string[]}** – An array of mobile numbers to which you want to send the SMS. The mobile number should be prefixed by the country code. E.g. **919820077777** where 91 is the country code. There should be no spaces, alphanumeric characters, zero prefixes or + sign or – sign in the mobile number.
4. **allsms {string[]}** – An array of text messages that you need to send to the corresponding mobile numbers specified in **allnumbers**.
5. **allfrom {string[]}** – An array of sender names or the source to be displayed on the mobile phone to the corresponding numbers specified in **allnumbers**. If a sender name value is specified as null or blank the API will pick up the default sender name in your profile on the SMS ConneXion web site.

#### Response:

The response is received as an array of integers. These integers are the corresponding message reference identification numbers for the messages accepted on the gateway. You may check the status of the messages by using these reference numbers later.

*Note:*

1. *The sender name should be pre-approved on the SMS ConneXion web site or else the SMS will not be sent.*
2. *The length (number of elements) of the following arrays should be the same. i.e. allnumbers, allsms & allfrom*

## **Demo Applications:**

We have created demo applications in various platforms for the convenience of software developers involved with the integration procedure.

1. Web Services API (C# Sample)
2. Web Services API (VB.NET Sample)

You may access these samples from the download section on the SMS ConneXion web site.

## **2. Check Balance**

In order to check the credit balance of a user you need to initiate a Web Service call on the URL <http://smc.smsconnexion.com/api/gateway.asmx>

*Note: Your URL may vary depending on the server on which your SMS ConneXion account is hosted.*

### **Method Definition:**

```
int GetBalance(string username, string password)
```

### **Parameters:**

1. **username {string}** – The account username created on the SMS ConneXion web site
2. **password {string}** – The account password for the above username on the SMS ConneXion web site

## Response:

The response is received an integer. The response indicates the number of available credits or unbilled credits for the specified account. If the response is a positive integer, the account is a prepaid account and the response indicates the number of available credits. If the response is a negative integer, the account is a postpaid account and the response indicates the number of unbilled credits.

### 3. Check Delivery Reports

In order to check the status of a message you need to initiate a Web Service call on the URL <http://smsc.smsconnexion.com/api/gateway.asmx>

*Note: Your URL may vary depending on the server on which your SMS ConneXion account is hosted.*

## Method Definition:

```
MessageStatus[] CheckStatus(string username, string password, int[] messageID)
```

## Parameters:

1. **username {string}** – The account username created on the SMS ConneXion web site
2. **password {string}** – The account password for the above username on the SMS ConneXion web site
3. **messageID {int[]}** – An array of message reference numbers received when calling the web service method **SendSMS**

## Response:

The response is received an array of MessageStatus objects. The members of the class MessageStatus are:

1. *Status {enum MessageStatusValue}* – Contains the status of the message. The possible values for the enumeration are Queued, Processing, Sent, Delivered and Failed.

Status	Interpretation
Queued	The SMS is queued for sending on the SMS ConneXion gateway
Processing	The SMS has been picked up for sending on the SMS ConneXion gateway.
Sent	The SMS has been sent to the mobile network.
Delivered	The SMS has been received on the recipient's mobile phone.
Failed	The SMS has not been delivered to the recipient's mobile phone.

2. *Delivered {string}* – This contains the time in string format when the delivery report for an SMS was received on the SMS ConneXion gateway.
3. *Reason {string}* – The reason for failure to deliver the message to the recipient's mobile phone.
4. *ID {int}* – The message reference number for the SMS