

Sending SMS messages from your website or any computer program has never been **easier, quicker, more efficient** or more **affordable**.

Mobility Methods provides a **carrier class, global turnkey** solution that enables you to send SMS messages via our service to any capable mobile device connected to a GSM network anywhere in the world.

The Mobility Methods messaging platform is built on a highly **robust, reliable** and **scalable** software environment developed by ANAM Wireless Internet Solutions.

- ✓ **No set-up fees**
- ✓ **Free test account**
- ✓ **Online technical support via MSN/ICQ and Yahoo!**
- ✓ **Global connectivity**
- ✓ **Instant results**
- ✓ **Affordable rates**
- ✓ **All SMS Formats and features**
- ✓ **Simple Integration API**

Our **complete** SMS messaging service is **easily integrated** into your new projects as well as with any existing legacy systems via your choice of API (Application Programming Interface) and we support all popular SMS messaging features including:



- ✓ Sender ID manipulation
- ✓ Instant Message or FlashSMS
- ✓ Logo and ringtones
- ✓ Multi-part messages (Concatenation)
- ✓ Unicode and foreign characters support
- ✓ EMS (Enhanced Message Service)
- ✓ Binary SMS
- ✓ VCards and VCals

## Connection APIs

### **FTP**

SMS messages may be originated and routed through our gateway using the File Transfer Protocol. This API is specifically designed for high volume BULK SMS providers and provides an efficient method for SMS delivery. The API is intended for customers who have an existing database or application that can generate a text file of the SMS messages recipients and the associated **text only** message content. The text file is then uploaded to our FTP site as a single file upload. Once the upload is complete, our systems will process the file contents, extracting each individual row of information and constructing and submitting the SMS message to our gateway for delivery.

The uploaded file should be a plain-text file with DOS or UNIX line breaks and uploaded to our FTP server at [ftpsms.mobilitymethods.com](https://ftpsms.mobilitymethods.com) on port 4995 using your supplied username and password. On completion of the upload the uploaded file **must be renamed** in the format of **username\_sms\_date.txt**

### ***The case-sensitive structure of the file is as follows:***

- **username:** supplied username
- **password:** supplied password
- **dlr-url:** (optional) the callback url where to send the notifications. Please see the section on *Callback URL* for more details.
- **email:** (optional) email address for error and status notifications.

The SMS messages follow the header variables based on one message per line and delimited with a | (pipe symbol ASCII 124 dec) as follows.

**From | To | Message**

**Callback URL**

The callback-url sets the path for notifications. When a message is submitted to an SMSC the SMSC responds with an acknowledgement that includes a reference to the message and a result representing the status of the message as per the table below. The call-back URL is typically a script on a web server that is capable of receiving the following parameters:

- **%p** the MSISDN number of the sender of the SMS message.
- **%P** the MSISDN number of the receiver of the SMS message.
- **%c** the reference returned by the SMSC for the submitted message. This is returned in the submission notification message. It is unavailable if the message submission fails.
- **%d** the delivery notification status value. This value indicates whether the message was successfully sent or whether it failed in transit.

**For Example:**

`http://myhost.com/cgi-bin/myscript?from=%p&to=%P&ref=%c&res=%d`

SMSCCONN_SENT_SMSC	0x1000	4096	Submitted successfully to the SMSC
SMSCCONN_SENT_ME	0x1001	4097	Delivered successfully to the SMSC
SMSCCONN_FAILED_REJECTED	0x2000	8192	The SM was rejected by the SMSC.
SMSCCONN_FAILED_SMSC	0x2001	8193	Submission to the SMSC failed, a generic error
SMSCCONN_FAILED_ME	0x2002	8194	Delivery to the ME failed, this is a generic error
SMSCCONN_FAILED_INV_SOURCE	0x2003	8195	An invalid source address was specified
SMSCCONN_FAILED_INV_DEST	0x2004	8196	An invalid destination address was specified
SMSCCONN_FAILED_INV_PARAM	0x2005	8197	An invalid parameter was specified
SMSCCONN_FAILED_EXPIRED	0x2006	8198	The SM expired while attempting delivery
SMSCCONN_FAILED_DELETED	0x2007	8199	The message was deleted in the SMSC
SMSCCONN_FAILED_UNKNOWN	0x2008	8200	The message is in an unknown state
SMSCCONN_BUFFERED	0x3000	12288	The message is buffered in the SMSC