

## Bramka SMS:

- *Obsługiwanych ponad 700 sieci w ponad 200 krajach Świata*
- *SMSy z własnym polem nadawcy*
- *Raporty doręczeń*
- *Obsługa długich wiadomości SMS*
- *Obsługa SMSów binarnych (WAP PUSH, vCard itp.)*
- *Obsługa dwukierunkowych wiadomości 2Way wysyłanych z numeru dedykowanego*
- *Obsługa FAKSÓW*
- *Odbieranie wiadomości SMS*

# Dokumentacja interfejsu Webservices API

Wersja 2.1 [18 lipca 2016]

## SPIS TREŚCI:

<b>1. Historia zmian w dokumentacji.....</b>	<b>3</b>
<b>2. Wprowadzenie.....</b>	<b>3</b>
2.1. Krok 1 – Rejestracja konta w GSMService.pl.....	3
2.2. Krok 2 – Aktywacja subkonta API .....	4
<b>3. Dostęp do Webservices API.....</b>	<b>4</b>
3.1. Opis metod udostępnionych w ramach Webservices API .....	5
3.1.1. Wysyłanie wiadomości SMS (Metoda SendSMS).....	5
3.1.2. Sprawdzanie kosztu wiadomości (Metoda GetPrice).....	5
3.1.3. Sprawdzenie statusów wiadomości (Metoda GetStatus) .....	5
3.1.4. Sprawdzenie stanu konta (Metoda GetBalance) .....	6
3.1.5. Anulowanie zaplanowanej wiadomości (Metoda Cancel).....	6
3.2. Opis struktur obiektów wykorzystanych przy wywołaniu metod .....	7
3.2.1. Struktura Account.....	7
3.2.2. Struktura Message .....	7
3.2.3. Struktura SendSMSReturn .....	8
3.2.4. Struktura GetPriceReturn .....	8
3.2.5. Struktura GetStatusReturn .....	8
3.2.6. Struktura GetBalanceReturn .....	9
3.2.7. Struktura CancelReturn.....	9
<b>4. Definiowanie zawartości Pola Nadawcy.....</b>	<b>9</b>
<b>5. Rodzaje wiadomości SMS .....</b>	<b>10</b>
<b>6. Kody błędów API.....</b>	<b>10</b>
<b>7. Statusy wiadomości SMS.....</b>	<b>11</b>
<b>8. Odbieranie powiadomień o zmianie statusu wiadomości.....</b>	<b>11</b>
<b>9. Odbieranie wiadomości SMS .....</b>	<b>11</b>
<b>10. Przykłady prostej implementacji Webservices API .....</b>	<b>12</b>
<b>11. Dodatkowe informacje i kontakt.....</b>	<b>12</b>

## SPIS TABEL:

Tab. 1. Objasnienie zwracanych kodów błędów .....	10
Tab. 2. Objasnienie statusów wiadomości .....	11

## 1. Historia zmian w dokumentacji

Wersja	Data	Wprowadzone zmiany
2.0	12.01.2014	Wersja bazowa
2.1	18.07.2016	Uzupełniono dokumentację o wiadomości typu ECO oraz 2Way

## 2. Wprowadzenie

Dokument ten powstał w celu objaśnienia procesu integracji **Bramki SMS GSMService.pl** z dowolną aplikacją lub systemem poprzez interfejs Webservices API. Obecnie na platformie GSMService.pl istnieje kilka różnych możliwości wysłania wiadomości SMS:

- Bezpośrednio ze strony <http://bramka.gsmervice.pl>
- Za pośrednictwem interfejsu HTTP API
- Za pośrednictwem interfejsu Webservices API
- Za pośrednictwem usługi MAIL2SMS

W niniejszym dokumencie przedstawiono możliwości, jakie niesie za sobą wykorzystanie interfejsu Webservices API. W chwili obecnej poprzez Webservices API możliwa jest wysyłka wiadomości **SMS, Flash SMS, ECO SMS, 2Way SMS, Binarnych SMS (WAP PUSH, vCard itp.)**.

Wysyłanie wiadomości SMS, sprawdzenie jej kosztu, statusu czy stanu konta poprzez Webservices API polega na wywołaniu zdalnej funkcji Webservices z jednoczesnym przekazaniem do niej odpowiednich parametrów.

Komunikacja z Webservices API może odbywać się zarówno na porcie **80** (w przypadku nieszyfrowanych połączeń) lub na porcie **443** (z szyfrowaniem **SSL**).

**Bardzo prosimy dokładnie zapoznać się z niniejszą dokumentacją przed rozpoczęciem integracji. Ułatwi to cały proces oraz pozwoli uniknąć wielu problemów.**

Aby rozpocząć integrację dowolnej aplikacji z niniejszą Bramką SMS konieczne jest założenie konta w serwisie Bramki SMS GSMService.pl i aktywacja dostępu do konta API (zwanego w dalszej części tej dokumentacji subkontem API).

Poniżej prezentujemy krok po kroku jak aktywować dostęp do interfejsu Webservices API.

### 2.1. Krok 1 – Rejestracja konta w GSMService.pl

Jeżeli nie posiadasz jeszcze konta w GSMService.pl należy je zarejestrować. W przeciwnym wypadku możesz przejść do kroku 2. Aby zarejestrować nowe konto w GSMService.pl należy:

- Wejść na stronę <http://bramka.gsmervice.pl/rejestracja-konta>,
- Wypełnić formularz rejestracyjny.

Po pomyślnej rejestracji na podany adres e-mail zostanie wysłana wiadomość z linkiem aktywacyjnym. Wystarczy kliknąć na link znajdujący się w wiadomości, a konto zostanie aktywowane. Następnie zaloguj się na swoje konto przy użyciu wybranego loginu i hasła. W dalszej kolejności należy wybrać typ konta (firmowe lub prywatne) i zaakceptować Regulamin Bramki SMS. Kolejnym etapem jest wprowadzenie danych właściciela konta oraz wybranie sposobu dostarczania **Faktur VAT**, jeżeli będziesz chciał je otrzymywać po każdym zasileniu konta. Opcjonalnie możesz również wybrać, jakiego rodzaju powiadomienia chcesz otrzymywać na e-mail. Po zakończeniu tych czynności zostaniesz przeniesiony do Bramki SMS.

## 2.2. Krok 2 – Aktywacja subkonta API

Po zalogowaniu się na <http://bramka.gsm-service.pl> należy:

- Wybrać z lewego MENU, z sekcji „KANAŁY DOSTĘPU” opcję „Interfejs API (integracja) -> Aktywacja i konfiguracja”,
- Wypełnić formularz podając:
  - Dowolnie wybrany login oraz hasło dla subkonta API (wpisywany login i hasło posłużą do autentykacji użytkownika przy wywoływaniu wszystkich funkcji Webservices API, dlatego też prosimy o staranne dobranie i bezpieczne przechowywanie loginu oraz hasła, aby nikt niepowołany nie uzyskał dostępu do Państwa subkonta API).
  - standard API, poprzez który nastąpi komunikacja z Bramką SMS (w tym przypadku Webservices API).
  - Opcjonalnie listę adresów IP, z których możliwy będzie dostęp do API. Poszczególne adresy powinny być oddzielone przecinkiem.
  - Adres callback, na który przekazywane będą raporty doręczeń wiadomości (więcej w rozdziale 8)

Formularz aktywacji subkonta API:

Login:  Login oraz hasło służą do uzyskania dostępu do API, dlatego nie przekazuj tych danych osobom niepowołanym.

Hasło:  W polu "Login" dozwolone są znaki a-z, A-Z, 0-9 oraz \_ (max 16 znaków).

Powtórz hasło:

Standard API:  HTTP API  SOAP API Wybierz standard API, który chcesz aktywować.

Dozwolone IP:\*  Adresy IP rozdzielone przecinkiem, z których możliwy jest dostęp do API.

Callback:\*  Adres, pod który przesyłane będą raporty doręczeń. Szczegóły w dokumentacji API.

\* pole opcjonalne

Utwórz konto    Resetuj

Po poprawnym wypełnieniu formularza i wciśnięciu przycisku „Utwórz konto”, subkonto API zostanie założone i aktywowane.

Posiadając aktywne subkonto API można rozpocząć proces integracji. Aby wysyłać wiadomości SMS, konieczne jest zasilenie konta dowolną kwotą (minimalnie 10 zł brutto).

W celu zasilenia konta, będąc zalogowanym do Bramki SMS, należy w lewym MENU wybrać z sekcji „Rozliczenia” opcję „Doładuj konto”, a następnie postępować zgodnie ze wskazówkami.

## 3. Dostęp do Webservices API

Udostępniamy możliwość integracji dowolnej aplikacji za pośrednictwem Webservices API.

WebService SOAP dostępny jest pod adresami:

- <http://api.gsm-service.pl/soap/v2/gateway.php>
- <https://api.gsm-service.pl/soap/v2/gateway.php> (połączenie szyfrowane - SSL)

WSDL dostępny jest pod adresami:

- <http://api.gsm-service.pl/soap/v2/gateway.php?wsdl>
- <https://api.gsm-service.pl/soap/v2/gateway.php?wsdl> (połączenie szyfrowane - SSL)

### 3.1. Opis metod udostępnionych w ramach Webservices API

#### 3.1.1. Wysyłanie wiadomości SMS (Metoda SendSMS)

Metoda **SendSMS** umożliwia wysłanie pojedynczej lub masowej wiadomości SMS. Metoda przyjmuje następujące parametry:

Metoda	Parametr	Typ parametru	Opis
<b>SendSMS</b>	account	Account	Struktura zawierające dane konieczne do uwierzytelnienia zapytania. Więcej w rozdziale 3.2.1.
	messages	Array of Message	Tablica składająca się ze struktur Message, zawierających informacje o poszczególnych wiadomościach SMS wysyłanych w ramach jednego wywołania tej metody. Szczegóły w rozdziale 3.2.2.

Wynik:

Struktura	Opis
<b>Array of SendSMSReturn</b>	Tablica składająca się ze struktur SendSMSReturn. Opis struktury znajduje się w Rozdziale 3.2.3.

#### 3.1.2. Sprawdzanie kosztu wiadomości (Metoda GetPrice)

Metoda **GetPrice** umożliwia sprawdzenie kosztu wiadomości przed ich wysłaniem. Metoda przyjmuje następujące parametry:

Metoda	Parametr	Typ parametru	Opis
<b>GetPrice</b>	account	Account	Struktura zawierające dane konieczne do uwierzytelnienia zapytania. Więcej w rozdziale 3.2.1
	messages	Array of Message	Tablica składająca się ze struktur Message, zawierających informacje o poszczególnych wiadomościach SMS których koszt jest sprawdzany w ramach jednego wywołania tej metody. Szczegóły w rozdziale 3.2.2.

Wynik:

Struktura	Opis
<b>Array of GetPriceReturn</b>	Tablica składająca się ze struktur GetPriceReturn. Opis struktury znajduje się w Rozdziale 3.2.4.

#### 3.1.3. Sprawdzenie statusów wiadomości (Metoda GetStatus)

Metoda **GetStatus** umożliwia sprawdzenie aktualnego statusu wysłanych wiadomości SMS. Statusy wiadomości zmieniają się np. w momencie doręczenia wiadomości do odbiorcy, itp. Alternatywnie można wykorzystać mechanizm Callback w celu odbierania automatycznych powiadomień o zmianie statusów. Szczegóły tego rozwiązania znajdują się w Rozdziale 8.

Metoda przyjmuje następujące parametry:

Metoda	Parametr	Typ parametru	Opis
	account	Account	Struktura zawierające dane konieczne do uwierzytelnienia

<b>GetStatus</b>			zapytania. Więcej w rozdziale 3.2.1
	ids	Array of xsd:string	Tablica składająca się z identyfikatorów wiadomości otrzymanych po wysłaniu wiadomości, których statusy mają zostać wyświetlone.

Wynik:

Struktura	Opis
<b>Array of GetStatusReturn</b>	Tablica składająca się ze struktur GetStatusReturn. Opis struktury znajduje się w Rozdziale 3.2.5.

### 3.1.4. Sprawdzenie stanu konta (Metoda GetBalance)

Metoda **GetBalance** umożliwia sprawdzenie aktualnego stanu konta w Bramce SMS. Metoda przyjmuje następujące parametry:

Metoda	Parametr	Typ parametru	Opis
<b>GetBalance</b>	account	Account	Struktura zawierające dane konieczne do uwierzytelnienia zapytania. Więcej w rozdziale 3.2.1

Wynik:

Struktura	Opis
<b>GetBalanceReturn</b>	Opis struktury znajduje się w Rozdziale 3.2.6

### 3.1.5. Anulowanie zaplanowanej wiadomości (Metoda Cancel)

Metoda **Cancel** pozwala anulować wiadomość, która została zaplanowana do wysyłki z przyszłą datą. Anulować można jedynie te wiadomości, które nie zostały jeszcze wysłane.

Metoda	Parametr	Typ parametru	Opis
<b>Cancel</b>	account	Account	Struktura zawierające dane konieczne do uwierzytelnienia zapytania. Więcej w rozdziale 3.2.1
	ids	Array of xsd:string	Tablica składająca się z identyfikatorów wiadomości otrzymanych po wysłaniu wiadomości, które powinny zostać anulowane.

Wynik:

Struktura	Opis
<b>Array of CancelReturn</b>	Tablica składająca się ze struktur CancelReturn. Opis struktury znajduje się w Rozdziale 0.

## 3.2. Opis struktur obiektów wykorzystanych przy wywołaniu metod

### 3.2.1. Struktura Account

Pole	Typ	Wymagalność	Wartość domyślna	Opis
login	xsd:string	wymagane	brak	Login subkonta API w Bramce SMS [zdefiniowany w czasie aktywacji API]
pass	xsd:string	wymagane	brak	Hasło do subkonta API w Bramce SMS [zdefiniowane w czasie aktywacji API]

### 3.2.2. Struktura Message

Pole	Typ	Wymagalność	Wartość domyślna	Opis
recipients	Array of xsd:string	wymagane	brak	Tablica składająca się z numerów telefonów, do których wiadomość ma zostać wysłana. UWAGA! Numery telefonu należy koniecznie podawać w postaci międzynarodowej z kodem kraju na początku (bez znaku +) np. 48601444555  <i>Opcjonalnie po każdym numerze odbiorcy można wstawić znak   a następnie podać własny identyfikator wiadomości, który będzie identyfikował ją w naszym systemie.</i>
message	xsd:string	zależnie	brak	Treść wiadomości, która ma zostać wysłana. Uwaga! Niektóre znaki (nie UNICODE) liczone są podwójnie, np. znaki: [ ] ~ ^ { } \   €. Treść powinna być zgodna z kodowaniem <b>UTF-8</b> . W przypadku wysyłania wiadomości SMS Binarnych, parametr ten powinien zawierać odpowiednią wiadomość binarną.
url	xsd:string	zależnie	brak	Adres URL dla wiadomości WAP PUSH
udh	xsd:string	zależnie	brak	Nagłówek UDH dla wiadomości Binarnej SMS
sender	xsd:string	opcjonalnie	skonfigurowany domyślny nadawca w ramach konta	Pole nadawcy (nadpis), z puli zdefiniowanych nadawców w ramach danego konta lub dzierżawiony numer dedykowany. Szczegóły w rozdziale 4.
msgType	xsd:int	opcjonalnie	1	Identyfikator rodzaju wiadomości SMS zgodnie z tabelą w Rozdziale 5.
unicode	xsd:boolean	opcjonalnie	false	Jeżeli wiadomość powinna zawierać znaki w kodowaniu unicode (np. polskie znaki diakrytyczne), które mają być doręczone do odbiorcy, wartość powinna być równa 1.
timestamp	xsd:int	opcjonalnie	aktualny timestamp	Czas doręczenia wiadomości do odbiorcy wyrażony w postaci timestamp (epoka Unixa)
sandbox	xsd:boolean	opcjonalnie	false	W przypadku prowadzenia testów z API, można ustawić wartość tego parametru na true, a wiadomości w rzeczywistości nie będą doręczane i konto nie będzie obciążane kosztami.

## 3.2.3. Struktura SendSMSReturn

Pole	Typ	Wymagalność	Opis
status	xsd:string	wymagane	Status żądania: OK – żądanie prawidłowe ERR – żądanie błędne (szczegółowy kod statusu błędu znajduje się w polu errCode)
errCode	xsd:string	opcjonalnie	kod statusu błędu zgodnie z tabelą w rozdziale 6
description	xsd:string	wymagane	Opis tekstowy statusu żądania
recipient	xsd:string	opcjonalnie	numer telefonu odbiorcy wiadomości
msgId	xsd:int	opcjonalnie	Identyfikator wysłanej wiadomości
parts	xsd:int	opcjonalnie	Liczba części SMS z której składa się wiadomość
price	xsd:double	opcjonalnie	Całkowity koszt wiadomości netto

## 3.2.4. Struktura GetPriceReturn

Pole	Typ	Wymagalność	Opis
status	xsd:string	wymagane	Status żądania: OK – żądanie prawidłowe ERR – żądanie błędne (szczegółowy kod statusu błędu znajduje się w polu errCode)
errCode	xsd:string	opcjonalnie	kod statusu błędu zgodnie z tabelą w rozdziale 6
description	xsd:string	wymagane	Opis tekstowy statusu żądania
recipient	xsd:string	opcjonalnie	numer telefonu odbiorcy wiadomości
parts	xsd:int	opcjonalnie	Liczba części SMS z której składa się wiadomość
price	xsd:double	opcjonalnie	Całkowity koszt wiadomości netto

## 3.2.5. Struktura GetStatusReturn

Pole	Typ	Wymagalność	Opis
status	xsd:string	wymagane	Status żądania: OK – żądanie prawidłowe ERR – żądanie błędne (szczegółowy kod statusu błędu znajduje się w polu errCode)
errCode	xsd:string	opcjonalnie	kod statusu błędu zgodnie z tabelą w rozdziale 6
description	xsd:string	wymagane	Opis tekstowy statusu żądania
msgId	xsd:int	opcjonalnie	Identyfikator wiadomości
recipient	xsd:string	opcjonalnie	Numer telefonu odbiorcy wiadomości
origDate	xsd:string	opcjonalnie	Data i godzina wysłania wiadomości (format: RRRR-MM-DD GG:MM:SS)
doneDate	xsd:string	opcjonalnie	Data i godzina ostatniej zmiany statusu wiadomości (format: RRRR-MM-DD GG:MM:SS)
msgStatusCode	xsd:string	opcjonalnie	Kod statusu wiadomości zgodny z tabelą w rozdziale 0.
msgStatusDescription	xsd:string	opcjonalnie	Tekstowy opis statusu wiadomości
msgType	xsd:int	opcjonalnie	Rodzaj wiadomości SMS zgodnie z tabelą w Rozdziale 5
parts	xsd:int	opcjonalnie	Liczba części SMS z której składa się wiadomość
price	xsd:double	opcjonalnie	Całkowity koszt wiadomości netto
extId	xsd:string	opcjonalnie	Identyfikator wiadomości nadany przez Użytkownika
unicode	xsd:boolean	opcjonalnie	true - jeżeli wiadomość została wysłana z kodowaniem UNICODE (zawierała np. polskie znaki diakrytyczne), false w przeciwnym wypadku.



## 3.2.6. Struktura GetBalanceReturn

Pole	Typ		Opis
status	xsd:string	wymagane	Status żądania: OK – żądanie prawidłowe ERR – żądanie błędne (szczegółowy kod statusu błędu znajduje się w polu errorCode)
errorCode	xsd:string	opcjonalnie	kod statusu błędu zgodnie z tabelą w rozdziale 6
description	xsd:string	wymagane	Opis tekstowy statusu żądania
balance	xsd:double	opcjonalnie	Stan konta netto
currency	xsd:string	opcjonalnie	Waluta konta

## 3.2.7. Struktura CancelReturn

Pole	Typ	Wymagalność	Opis
status	xsd:string	wymagane	Status żądania: OK – żądanie prawidłowe ERR – żądanie błędne (szczegółowy kod statusu błędu znajduje się w polu errorCode)
errorCode	xsd:string	opcjonalnie	kod statusu błędu zgodnie z tabelą w rozdziale 6
description	xsd:string	wymagane	Opis tekstowy statusu żądania
msgId	xsd:int	opcjonalnie	Identyfikator wiadomości, która powinna zostać anulowana
recipient	xsd:string	opcjonalnie	Numer telefonu odbiorcy wiadomości, która powinna zostać anulowana

## 4. Definiowanie zawartości Pola Nadawcy

Pole nadawcy jest to numer lub nazwa nadawcy SMSa, która wyświetla się na telefonie odbiorcy w polu „od”. Z uwagi na konieczność zachowania wysokich standardów bezpieczeństwa oraz konieczność ograniczenia potencjalnych nadużyć niezbędne jest uprzednie zdefiniowanie oraz zweryfikowanie zawartości pola nadawcy, które można będzie wykorzystać podczas wysyłania wiadomości SMS.

W tym celu należy zalogować się na konto bramki w serwisie <http://bramka.gsm-service.pl>, a następnie wybrać z lewego menu z sekcji „Obsługa konta” opcję „Edycja nadawców wiadomości”.

W formularzu można wpisać zarówno **numer telefonu**, jak i **tekst alfanumeryczny**, który chcesz wykorzystać w polu nadawcy. Tekst alfanumeryczny może zawierać maksymalnie 11 znaków, w tym małe i duże litery od A do Z, cyfry od 1 do 9 oraz kropkę i znak spacji. Pozostałe znaki nie są akceptowane.

Nadawca alfanumeryczny po zweryfikowaniu przez operatora zostanie przypisany do Twojego konta i będziesz mógł go używać **w polu nadawcy** podczas wysyłania wiadomości SMS. W przypadku, gdy wprowadzisz do formularza numer telefonu, otrzymasz na ekranie wskazówki, w jaki sposób go

zweryfikować. Wiadomości z nadawcą alfanumerycznym mogą być wysyłane do wszystkich sieci, natomiast z numerem telefonu w polu nadawcy wyłącznie do niektórych sieci zagranicznych.

Każdy użytkownik Bramki SMS może zdefiniować **dowolną ilość** pól nadawcy. W przypadku, gdy wysyłając wiadomość SMS użyjesz niezdefiniowane uprzednio pole nadawcy, wiadomość zostanie wysłana z polem nadawcy ustawionym domyślnie na Twoim koncie.

## 5. Rodzaje wiadomości SMS

ID rodzaju wiadomości	Rodzaj wiadomości
1	SMS Tradycyjny
2	Flash SMS
3	SMS ECO
4	2Way SMS (komunikacja dwukierunkowa – możliwość wysyłania SMS z uniwersalnego numeru Bramki SMS lub numeru dedykowanego oraz otrzymania na niego odpowiedzi)

## 6. Kody błędów API

Opisane w niniejszej dokumentacji metody w przypadku, gdy status ma wartość ERR zwracają w polu errorCode trzycyfrowe kody błędu. W **Tab. 1** znajduje się ich spis wraz z niezbędnymi objaśnieniami.

**Tab. 1. Objasnienie zwracanych kodów błędów**

Kod błędu	Opis błędu
001	Konto Użytkownika nie istnieje
002	Dostęp do tego typu interfejsu API jest zablokowany w konfiguracji API
003	Dostęp do API z tego adresu IP jest zablokowany w konfiguracji API
004	Na Twoim koncie nie ma wystarczających środków, aby wysłać tą wiadomość. Doładuj konto i spróbuj ponownie
110	Nie wprowadzono treści wiadomości
111	Wprowadzona treść wiadomości zawiera niedozwolone znaki.
112	Nieobsługiwany format kodowania treści wiadomości (encoding)
113	Wiadomość WAP PUSH nie zawiera prawidłowego adresu URL
114	Wiadomość FLASH nie może zawierać znaków UNICODE
115	Wprowadzony nagłówek UDH jest nieprawidłowy.
116	Numer odbiorcy wiadomości jest nieprawidłowy
117	Zaplanowana data wysyłki wiadomości nie może być dalsza niż 14 dni
118	Podano nieprawidłowy typ wiadomości
119	Parametry UDH oraz URL nie mogą być przekazywane jednocześnie
120	Numer, pod który próbowano wysłać wiadomość jest obecnie nieobsługiwany
121	SMSy ECO oraz 2Way można wysłać wyłącznie na standardowe numery polskich odbiorców
200	Wystąpił nieznaną błąd
201	Brak wiadomości z podanym id
202	Nie można anulować wskazanej wiadomości

## 7. Statusy wiadomości SMS

Tab. 2. Objasnienie statusów wiadomości

Kod statusu	Opis statusu
QUEUED	Wiadomość oczekuje na wysłanie
SCHEDULED	Wiadomość oczekuje na wysłanie o określonej porze
SENT	Wiadomość została wysłana
DELIVERED	Wiadomość została doręczona do odbiorcy
REJECTED	Wiadomość została odrzucona
CANCELLED	Wiadomość została anulowana
UNDELIVERED	Wiadomość nie została doręczona do odbiorcy
EXPIRED	Wiadomość wygasła
UNKNOWN	Status wiadomości nie jest znany

## 8. Odbieranie powiadomień o zmianie statusu wiadomości

Istnieje możliwość przekazywania statusów wysłanych wiadomości SMS na podany adres Callback tuż po każdej zmianie statusu danej wiadomości. Po zmianie statusu wiadomości wywołany jest uprzednio zdefiniowany na Państwa koncie adres Callback i przekazywane są metodą GET następujące parametry:

- msg\_id – identyfikator wiadomości, której dotyczy raport (uzyskany po wysłaniu wiadomości)
- status – aktualny status wiadomości zgodnie z tabelą w Rozdziale 0
- status\_date – data i godzina zmiany statusu w formacie RRRR-MM-DD GG:MM:SS (zakodowana funkcją urlencode)
- recipient – numer telefonu odbiorcy wiadomości

Jeżeli z jakiegokolwiek powodu adres Callback jest w danej chwili niedostępny, wówczas system ponowi próby przekazania raportu ze statusem wiadomości po 5, 15 minutach, 1 godzinie i 6 godzinach.

Adres Callback można skonfigurować na stronie aktywacji API – szczegóły w Rozdziale **Błąd! Nie można odnaleźć źródła odwołania.**

### Przykład:

Została wysłana wiadomość SMS pod numer 48509876543. Po jej doręczeniu do odbiorcy zostanie wywołany następujący adres: [http://jakas-domena.pl/skrypt.php?msg\\_id=455&status=DELIVERED&date=2010-04-21+13%3A22%3A14&recipient=48601234567](http://jakas-domena.pl/skrypt.php?msg_id=455&status=DELIVERED&date=2010-04-21+13%3A22%3A14&recipient=48601234567)

Przykładowy skrypt PHP odbierający raport z adresu Callback umieszczony został w przykładach implementacji na stronie <http://bramka.gsm-service.pl/przyklady-implementacji-api>.

## 9. Odbieranie wiadomości SMS

Udostępniamy usługę umożliwiającą odbieranie:

- odpowiedzi na wysłane z Bramki SMS wiadomości 2Way SMS,
- wiadomości SMS wysłane na przydzielony Państwu numer dedykowany,
- wiadomości SMS wysłane na przydzielony Państwu numer z dedykowanym prefiksem.

Szczegółowy opis poszczególnych wariantów usługi zawarty został na stronie <http://bramka.gsmervice.pl/odbieranie-wiadomosci-sms>. Odebrane wiadomości mogą być przekazywane na podany adres **Callback**. Po odebraniu wiadomości SMS kierowanej do Państwa nasza platforma wywoła adres Callback przekazując do niego metodą GET następujące parametry:

- **msg\_id** - Unikalny identyfikator odebranej wiadomości
- **sender** - Numer telefonu nadawcy wiadomości
- **recipient** - Numer telefonu odbiorcy wiadomości
- **text** - Treść odebranej wiadomości SMS (zakodowanej funkcją urlencode)
- **date** - Data odbioru wiadomości SMS w formacie RRRR-MM-DD GG:MM:SS (zakodowana funkcją urlencode)

Jeżeli z jakiegokolwiek powodu adres Callback jest w danej chwili niedostępny, wówczas system ponowi próby przekazania wiadomości odpowiednio po 5, 15 minutach, 1 godzinie i 6 godzinach.

Umożliwi to Państwu odebranie wiadomości SMS bezpośrednio w Państwa serwisach, systemach i aplikacjach.

#### **Przykład:**

Użytkownik usługi odbioru wiadomości SMS opartej na numerze dedykowanym wskazał adres <http://jakas-domena.pl/skrypt.php> jako adres **Callback**. W momencie aktywacji usługi został przydzielony numer dedykowany +48601234567. W chwili odbierania przez Bramkę wiadomości SMS o treści „To jest wiadomosc testowa” wysłanej spod numeru 48509876543 zostanie wywołany następujący adres: [http://jakas-domena.pl/skrypt.php?msg\\_id=455&sender=48509876543&recipient=48601234567&text=To+jest+wiadomosc+testowa&date=2010-04-21+13%3A22%3A14](http://jakas-domena.pl/skrypt.php?msg_id=455&sender=48509876543&recipient=48601234567&text=To+jest+wiadomosc+testowa&date=2010-04-21+13%3A22%3A14).

Przykładowy skrypt PHP odbierający wiadomość SMS z adresu Callback umieszczony został w przykładach implementacji na stronie <http://bramka.gsmervice.pl/przyklady-implementacji-api>.

## **10. Przykłady prostej implementacji Webservices API**

Przykłady implementacji w różnych językach programowania dostępne są do pobrania na stronie <http://bramka.gsmervice.pl/przyklady-implementacji-api>

## **11. Dodatkowe informacje i kontakt**

Odpowiedzi na wszelkie pytania oraz dodatkowe informacje uzyskać można kontaktując się z nami:

### **INTERMEDIA**

www: <http://www.gsmervice.pl>  
e-mail: [bramka@gsmervice.pl](mailto:bramka@gsmervice.pl)



Aktualne informacje dotyczące interfejsów API zawsze znaleźć można pod adresem: <http://bramka.gsmervice.pl/dokumentacja-api>.