

<https://postivo.pl>
e-mail: biuro@postivo.pl
tel. +48 17 717 37 60

Dokumentacja interfejsu API

wersja 1.31 [24 sierpnia 2021]



Spis treści

1. Historia zmian w dokumentacji	4
2. Wprowadzenie	4
2.1. Krok 1 - Rejestracja konta w Postivo.pl.....	5
2.2. Krok 2 - Aktywacja dostępu do konta poprzez API.....	5
3. Dostęp do API - Adres URL do Web Services (SOAP/WSDL).....	5
4. Dostępne metody.....	6
4.1. Wysłanie nowej przesyłki.....	6
4.2. Sprawdzenie statusu przesyłki	6
4.3. Sprawdzenie stanu konta Użytkownika.....	7
4.4. Sprawdzanie kosztu przesyłki.....	7
4.5. Pobieranie listy zdefiniowanych profili przesyłek.....	8
4.6. Pobieranie listy zdefiniowanych nadawców	8
4.7. Generowanie potwierdzeń nadania przesyłki.....	9
4.8. Dodawanie nowego adresu nadawcy przesyłek.....	9
4.9. Weryfikacja (aktywacja) adresu nadawcy.....	10
4.10. Usuwanie adresów nadawców.....	10
5. Opis struktur danych wykorzystanych w API	11
5.1. Struktura tns:DispatchReturnObject	11
5.2. Struktura tns:BalanceReturnObject.....	11
5.3. Struktura tns:PriceReturnObject.....	12
5.4. Struktura tns:ConfigProfilesReturnObject	12
5.5. Struktura tns:SendersReturnObject	12
5.6. Struktura tns:CertificateReturnObject.....	13
5.7. Struktura tns:AddSenderReturnObject.....	13
5.8. Struktura tns:VerifySenderReturnObject.....	14
5.9. Struktura tns:RemoveSenderReturnObject	14
5.10. Struktura tns:ArrayOfDocumentFiles	14
5.11. Struktura tns:ArrayOfRecipients	14
5.12. Struktura tns:ArrayOfOptions	15
5.13. Struktura tns:ArrayOfShipments.....	15
5.14. Struktura tns:ArrayOfShipmentsPrice	15
5.15. Struktura tns:ArrayOfSenders.....	15
5.16. Struktura tns:ArrayOfConfigProfiles.....	15
5.17. Struktura tns:DocumentFile.....	16
5.18. Struktura tns:Recipient.....	16
5.19. Struktura tns:Option.....	17
5.20. Struktura tns:ConfigProfile	17
5.21. Struktura tns:Sender	18
5.22. Struktura tns:Shipment	18
5.23. Struktura tns:ShipmentPrice.....	19
5.24. Struktura tns:SenderData	20
6. Definiowanie parametrów.....	20
6.1. Definiowanie nowych profili przesyłek.....	20
6.2. Definiowanie nadawców przesyłek	21
7. Kody błędów (rezultat) żądań.....	21
8. Kody statusów przesyłek	23
8.1. Kody statusów nie wysłanych przesyłek	23

8.2. Kody statusów prawidłowo wysłanych przesyłek	23
9. Mechanizm Callback – asynchroniczna aktualizacja statusów przesyłek	24
10. Przykładowe implementacje w różnych środowiskach programistycznych	25
11. System testowy API (sandbox)	25
12. Dodatkowe informacje i kontakt	25

1. Historia zmian w dokumentacji

Wersja	Data	Opis zmian
1.1	18 grudnia 2011 r.	W związku z rozszerzeniem zakresu usługi o przesyłki zagraniczne do struktur tns:Sender, tns:SenderData, tns:AddSenderReturnObject dodano opcjonalny parametr sender_country, natomiast do struktur tns:Shipment, tns:ShipmentPrice, tns:Recipient dodano opcjonalny parametr recipient_country. Dodatkowe kody statusów żądań (041 oraz 090). Nowy kod statusu przesyłek – 5.
1.11	20 stycznia 2012 r.	Dodatkowy kody statusu przesyłek – 200 (Status zewnętrzny)
1.12	27 kwietnia 2012 r.	Metoda removeSender, dodatkowe kody błędów: 091, 092
1.13	20 maja 2012 r.	W strukturze tns:Option dodano opcjonalny parametr custom_id
1.15	25 października 2018	Drobne zmiany edytorskie
1.20	4 września 2020	Nowe parametry mpk, order_name, wycofanie faktów z oferty
1.30	24 stycznia 2021	Dodanie integracji z biblioteką dokumentów, nowy parametr file_library_id
1.31	24 sierpnia 2021	Dodanie właściwości custom_id oraz tracking_number do struktury tns:Shipment

2. Wprowadzenie

Niniejszy dokument powstał w celu opisanie sposobu integracji platformy Postivo.pl z dowolną aplikacją lub systemem, przy wykorzystaniu interfejsu API, zbudowanego w oparciu o technologię Web Services. Obecnie istnieją dwie metody korzystania z usług Postivo.pl:

- Bezpośrednio ze strony <https://postivo.pl>
- Za pośrednictwem interfejsu API

W niniejszym dokumencie przedstawione zostały możliwości, jakie niesie za sobą wykorzystanie interfejsu API (SOAP/WSDL). W chwili obecnej poprzez API możliwa jest wysyłka przesyłek listowych i zarządzanie nimi.

Nadanie przesyłki, sprawdzenie jej statusu czy stanu konta Użytkownika za pośrednictwem API polega na wywołaniu zdalnej funkcji Web Services z jednoczesnym przekazaniem do niej pewnych parametrów.

Komunikacja z API może odbywać się zarówno na porcie 443 (z szyfrowaniem SSL) jak i na porcie 80 (w przypadku nieszyfrowanych połączeń). Zalecamy jednak wykorzystywanie wyłącznie szyfrowanego połączenia, gdzie tylko jest to możliwe. Nieszyfrowane połączenia pozostawione są jeszcze przez pewien czas jedynie dla zachowania kompatybilności ze starszymi integracjami. Wszystkie przekazywane wartości parametrów metod powinny być kodowane w standardzie UTF-8.

Prosimy o dokładne zapoznanie się z niniejszą dokumentacją przed rozpoczęciem integracji. Ułatwi to cały proces oraz pozwoli uniknąć wielu problemów. Aby rozpocząć integrację dowolnej aplikacji z platformą Postivo.pl konieczne jest założenie konta w serwisie Postivo.pl i aktywacja dostępu do konta poprzez API.

2.1. Krok 1 - Rejestracja konta w Postivo.pl

Jeżeli nie posiadasz jeszcze konta w Postivo.pl należy je zarejestrować. W przeciwnym wypadku przejdź do kroku drugiego (Rozdział 2.2). Aby zarejestrować nowe konto w POSTIVO.PL należy:

- Otworzyć stronę <https://panel.postivo.pl/rejestracja>,
- Wypełnić formularz rejestracyjny podając wymagane dane i postępować zgodnie ze wskazówkami wyświetlanymi na ekranie.

Po prawidłowym wypełnieniu formularza rejestracyjnego na podany adres e-mail zostanie wysłana wiadomość e-mail z linkiem aktywacyjnym, który należy otworzyć w celu aktywacji konta. Od tego momentu konto będzie aktywne.

2.2. Krok 2 - Aktywacja dostępu do konta poprzez API

W celu aktywacji dostępu do konta poprzez interfejs API należy:

- Zalogować się na swoje konto na stronie <https://panel.postivo.pl>, podając login i hasło,
- Z lewego menu z sekcji „INTEGRACJE” wybrać opcję „API”, a następnie „Ustawienia API”

Na tej stronie można umożliwić lub zablokować dostęp do danego konta Użytkownika poprzez interfejs API. W celu aktywacji dostępu poprzez API proszę kliknąć na przycisk edycji przy wybranym użytkowniku konta. Następnie dwukrotnie należy wprowadzić hasło, które będzie wykorzystywane przy sprawdzaniu tożsamości podczas wywoływania zdalnych metod. Zalecamy, aby hasło do API było inne od hasła wykorzystywanego do logowania się na konto Użytkownika. W polu „Dozwolone adresy IP” opcjonalnie możesz wpisać adresy IP, z których wywoływane będą metody. Poszczególne adresy IP należy rozdzielić znakiem przecinka (.). Pozostawienie tego pola pustego oznacza, że akceptowane będą połączenia ze wszystkich adresów IP. W polu „Adres Callback” opcjonalnie można prowadzić adres URL skryptu, na który będą przekazywane aktualizacje statusów przesyłek (szczegóły w Rozdziale 9). Po wprowadzeniu wszystkich parametrów i zapisaniu ustawień dostęp do konta poprzez API jest już możliwy.

3. Dostęp do API - Adres URL do Web Services (SOAP/WSDL)

Web Service SOAP dostępny jest pod adresem:

- <https://api.postivo.pl/webservices.php> (połączenie szyfrowane - SSL)

WSDL dostępny jest pod adresem:

- <https://api.postivo.pl/webservices.php?wsdl> (połączenie szyfrowane - SSL)

4. Dostępne metody

4.1. Wysłanie nowej przesyłki

Metoda **dispatch** umożliwia zdefiniowanie nowej przesyłki z zawartymi w niej dokumentami.

Metoda	Parametr	Typ parametru	Opis
dispatch	login	xsd:string	Login konta w Postivo.pl [zdefiniowany w czasie rejestracji konta]
	api_pass	xsd:string	Hasło dostępne do API [zdefiniowane w czasie aktywacji dostępu do konta poprzez API]
	msg_type	xsd:string	Typ przesyłki [1 – przesyłka listowa]
	document_files	tns:ArrayOfDocumentFiles	Tablica składająca się ze struktur DocumentFile (zawierających zawartość plików z dokumentami, które mają zostać wysłane). Więcej w Rozdziale 5.10.
	recipients	tns:ArrayOfRecipients	Tablica składająca się ze struktur Recipient (zawierających dane odbiorców, do których przesyłka ma trafić). Więcej w Rozdziale 5.11.
	options	tns:ArrayOfOptions	Tablica składająca się ze struktur Option (zawierających informacje o dodatkowych opcjach przesyłki). Więcej w Rozdziale 5.12.

Wynik:

Struktura	Opis
tns:DispatchReturnObject	Opis struktury znajduje się w Rozdziale 5.1

4.2. Sprawdzenie statusu przesyłki

Metoda **getDispatchStatus** umożliwia sprawdzenie aktualnych statusów dotychczas wysłanych przesyłek.

Metoda	Parametr	Typ parametru	Opis
getDispatchStatus	login	xsd:string	Login konta w Postivo.pl [zdefiniowany w czasie rejestracji konta]
	api_pass	xsd:string	Hasło dostępne do API [zdefiniowane w czasie aktywacji dostępu do konta poprzez API]
	dispatch_ids	tns:ArrayOfDispatchIds	Tablica, której poszczególne elementy zawierają identyfikatory przesyłek (typu string), których status chcemy sprawdzić.

Wynik:

Struktura	Opis
tns:DispatchReturnObject	Opis struktury znajduje się w Rozdziale 5.1

4.3. Sprawdzenie stanu konta Użytkownika

 Metoda **getBalance** umożliwia sprawdzenie aktualnego stanu konta, typu konta i limitu konta Użytkownika.

Metoda	Parametr	Typ parametru	Opis
getBalance	login	xsd:string	Login konta w Postivo.pl [zdefiniowany w czasie rejestracji konta]
	api_pass	xsd:string	Hasło dostępne do API [zdefiniowane w czasie aktywacji dostępu do konta poprzez API]

Wynik:

Struktura	Opis
tns:BalanceReturnObject	Opis struktury znajduje się w Rozdziale 5.2.

4.4. Sprawdzanie kosztu przesyłki

 Metoda **getPrice** pozwala sprawdzić koszt przesyłki z wybranymi parametrami bez jej wysłania.

Metoda	Parametr	Typ parametru	Opis
getPrice	login	xsd:string	Login konta w Postivo.pl [zdefiniowany w czasie rejestracji konta]
	api_pass	xsd:string	Hasło dostępne do API [zdefiniowane w czasie aktywacji dostępu do konta poprzez API]
	msg_type	xsd:string	Typ przesyłki [1 – przesyłka listowa]
	document_files	tns:ArrayOfDocumentFiles	Tablica składająca się ze struktur DocumentFile (zawierających zawartość plików z dokumentami, które mają zostać wysłane). Więcej w Rozdziale 5.10.
	recipients	tns:ArrayOfRecipients	Tablica składająca się ze struktur Recipient (zawierających dane odbiorców, do których przesyłka ma trafić). Więcej w Rozdziale 5.11.
	config_id	xsd:int	ID zdefiniowanego na platformie Postivo.pl profilu przesyłki (jeżeli wartość parametru to 0, przyjmowana jest domyślnie konfiguracja, umożliwiającą najtańszą wysyłkę). Szczegóły w Rozdziale 6.1.

Wynik:

Struktura	Opis
tns:PriceReturnObject	Opis struktury znajduje się w Rozdziale 5.3.

4.5. Pobieranie listy zdefiniowanych profili przesyłek

 Metoda **getConfigProfiles** umożliwia pobranie aktualnej listy wszystkich zdefiniowanych profili przesyłek.

Metoda	Parametr	Typ parametru	Opis
getConfigProfiles	login	xsd:string	Login konta w Postivo.pl [zdefiniowany w czasie rejestracji konta]
	api_pass	xsd:string	Hasło dostępne do API [zdefiniowane w czasie aktywacji dostępu do konta poprzez API]
	msg_type	xsd:string	Typ przesyłki, dla których chcemy pobrać listę profili przesyłek [1 – przesyłka listowa]

Wynik:

Struktura	Opis
tns:ConfigProfilesReturnObject	Opis struktury znajduje się w Rozdziale 5.4.

4.6. Pobieranie listy zdefiniowanych nadawców

 Metoda **getSenders** umożliwia pobranie listy wszystkich nadawców, zdefiniowanych na koncie Użytkownika.

Metoda	Parametr	Typ parametru	Opis
getSenders	login	xsd:string	Login konta w Postivo.pl [zdefiniowany w czasie rejestracji konta]
	api_pass	xsd:string	Hasło dostępne do API [zdefiniowane w czasie aktywacji dostępu do konta poprzez API]

Wynik:

Struktura	Opis
tns:SendersReturnObject	Opis struktury znajduje się w Rozdziale 5.5.

4.7. Generowanie potwierdzeń nadania przesyłki

Metoda **getCertificate** umożliwia wygenerowanie potwierdzeń nadania przesyłek. Potwierdzenia nadania są dostępne po przekazaniu przesyłki do operatora pocztowego.

Metoda	Parametr	Typ parametru	Opis
getCertificate	login	xsd:string	Login konta w Postivo.pl [zdefiniowany w czasie rejestracji konta]
	api_pass	xsd:string	Hasło dostępne do API [zdefiniowane w czasie aktywacji dostępu do konta poprzez API]
	dispatch_id	xsd:string	identyfikator przesyłki, dla której wygenerowane ma zostać potwierdzenie.
	certificate_type	xsd:int	1 – w przypadku, gdy wygenerowane ma zostać potwierdzenie nadania przesyłki

Wynik:

Struktura	Opis
tns:CertificateReturnObject	Opis struktury znajduje się w Rozdziale 5.6.

4.8. Dodawanie nowego adresu nadawcy przesyłek

Metoda **addSender** umożliwia zdefiniowanie nowych adresów nadawców. Adres nadawcy nie jest umieszczany na kopercie przesyłki. Po zdefiniowaniu adresu nadawcy wysyłamy pod dany adres list z kodem weryfikacyjnym (w systemie testowym Sandbox list wysyłany jest na adres e-mail). Po otrzymaniu listu odbiorca musi zweryfikować adres przekazując do nas otrzymany kod. W praktyce należy wywołać wówczas metodę **verifySender**, opisaną w Rozdziale 4.9.

Metoda	Parametr	Typ parametru	Opis
addSender	login	xsd:string	Login konta w Postivo.pl [zdefiniowany w czasie rejestracji konta]
	api_pass	xsd:string	Hasło dostępne do API [zdefiniowane w czasie aktywacji dostępu do konta poprzez API]
	sender_data	tns:SenderData	Obiekt struktury tns:SenderData (zawierający dane nadawcy). Szczegóły znajdują się w Rozdziale 5.24
	accept_terms	xsd:boolean	Wartość true oznacza akceptację oświadczenia o następującej treści: „Zgadzam się na wysyłkę kodu weryfikacyjnego na powyższy adres (koszt zgodny z cennikiem - pobrany z konta Postivo.pl)”.

Wynik:

Struktura	Opis
tns:AddSenderReturnObject	Opis struktury znajduje się w Rozdziale 5.7.

4.9. Weryfikacja (aktywacja) adresu nadawcy

Metoda **verifySender** służy do przekazania do Postivo.pl otrzymanego listownie kodu weryfikacyjnego. Jeżeli przekazany kod będzie zgodny z kodem weryfikacyjnym znajdującym się w liście, dane nadawcy zostaną zweryfikowane i będzie je można wykorzystać w celu zlecenia przesyłek.

Metoda	Parametr	Typ parametru	Opis
verifySender	login	xsd:string	Login konta w Postivo.pl [zdefiniowany w czasie rejestracji konta]
	api_pass	xsd:string	Hasło dostępne do API [zdefiniowane w czasie aktywacji dostępu do konta poprzez API]
	sender_id	xsd:int	Identyfikator nadawcy, którego chcemy zweryfikować (wygenerowany po wywołaniu metody addSender)
	verification_code	xsd:string	Otrzymany listownie kod weryfikacyjny

Wynik:

Struktura	Opis
tns:VerifySenderReturnObject	Opis struktury znajduje się w Rozdziale 5.8.

4.10. Usuwanie adresów nadawców

Metoda **removeSender** służy do usunięcia nadawcy z listy nadawców. Usunięty może być każdy nadawca poza podstawowym (domyślnym) nadawcą

Metoda	Parametr	Typ parametru	Opis
verifySender	login	xsd:string	Login konta w Postivo.pl [zdefiniowany w czasie rejestracji konta]
	api_pass	xsd:string	Hasło dostępne do API [zdefiniowane w czasie aktywacji dostępu do konta poprzez API]
	sender_id	xsd:int	Identyfikator nadawcy, którego chcemy usunąć

Wynik:

Struktura	Opis
tns:RemoveSenderReturnObject	Opis struktury znajduje się w Rozdziale 5.9.

5. Opis struktur danych wykorzystanych w API

5.1. Struktura tns:DispatchReturnObject

Pole	Typ	Opis
result	xsd:string	Status żądania: OK – żądanie poprawne ERR – żądanie błędne
result_code	xsd:string	W przypadku błędnego żądania zawiera trzycyfrowy kod błędu, zgodny z Tabelą w Rozdziale 7. W przypadku poprawnego żądania zawiera kod 000 .
result_description	xsd:string	Tekstowy opis rezultatu żądania
shipments	tns:ArrayOfShipments	Tablica zawierająca obiekty struktury Shipment . Każdy obiekt Shipment zawiera dane o statusie pojedynczej przesyłki (wysłanej do pojedynczego odbiorcy). W przypadku błędnego żądania (result: ERR) tablica jest pusta. Szczegóły w Rozdziale 5.13.

5.2. Struktura tns:BalanceReturnObject

Pole	Typ	Opis
result	xsd:string	Status żądania: OK – żądanie poprawne ERR – żądanie błędne
result_code	xsd:string	W przypadku błędnego żądania zawiera trzycyfrowy kod błędu, zgodny z Tabelą w Rozdziale 7. W przypadku poprawnego żądania zawiera kod 000 .
result_description	xsd:string	Tekstowy opis rezultatu żądania
balance	xsd:float	Aktualny stan konta (wyrażony w zł polskich)
limit	xsd:float	W przypadku konta POST-PAID określa umowny limit maksymalnego wykorzystania konta. Dla kont PRE-PAID ma wartość 0
postpaid	xsd:int	0 – konto typu PRE-PAID 1 – konto typu POST-PAID

5.3. Struktura tns:PriceReturnObject

Pole	Typ	Opis
result	xsd:string	Status żądania: OK – żądanie poprawne ERR – żądanie błędne
result_code	xsd:string	W przypadku błędnego żądania zawiera trzycyfrowy kod błędu, zgodny z Tabelą w Rozdziale 7. W przypadku poprawnego żądania zawiera kod 000 .
result_description	xsd:string	Tekstowy opis rezultatu żądania
shipments_price	tns:ArrayOfShipmentsPrice	Tablica zawierająca obiekty struktury ShipmentPrice . Każdy obiekt ShipmentPrice zawiera dane o szczegółach i cenie pojedynczej przesyłki. W przypadku błędnego żądania (result: ERR) tablica jest pusta. Szczegóły w Rozdziale 5.14.

5.4. Struktura tns:ConfigProfilesReturnObject

Pole	Typ	Opis
result	xsd:string	Status żądania: OK – żądanie poprawne ERR – żądanie błędne
result_code	xsd:string	W przypadku błędnego żądania zawiera trzycyfrowy kod błędu, zgodny z Tabelą w Rozdziale 7. W przypadku poprawnego żądania zawiera kod 000.
result_description	xsd:string	Tekstowy opis rezultatu żądania
config_profiles	tns:ArrayOfConfigProfiles	Tablica zawierająca obiekty struktury ConfigProfile . Każdy obiekt ConfigProfile zawiera dane o pojedynczym profilu przesyłek. W przypadku błędnego żądania (result: ERR) lub braku zdefiniowanych profili przesyłek tablica jest pusta. Szczegóły w Rozdziale 0.

5.5. Struktura tns:SendersReturnObject

Pole	Typ	Opis
result	xsd:string	Status żądania: OK – żądanie poprawne ERR – żądanie błędne
result_code	xsd:string	W przypadku błędnego żądania zawiera trzycyfrowy kod błędu, zgodny z Tabelą w Rozdziale 7. W przypadku poprawnego żądania zawiera kod 000.
result_description	xsd:string	Tekstowy opis rezultatu żądania
senders	tns:ArrayOfSenders	Tablica zawierająca obiekty struktury Sender . Każdy obiekt Sender zawiera dane o pojedynczym nadawcy przesyłki. W przypadku błędnego żądania (result: ERR) tablica jest pusta. Szczegóły w Rozdziale 5.15.

5.6. Struktura tns:CertificateReturnObject

Pole	Typ	Opis
result	xsd:string	Status żądania: OK – żądanie poprawne ERR – żądanie błędne
result_code	xsd:string	W przypadku błędnego żądania zawiera trzycyfrowy kod błędu, zgodny z Tabelą w Rozdziale 7. W przypadku poprawnego żądania zawiera kod 000.
result_description	xsd:string	Tekstowy opis rezultatu żądania
cert_available	xsd:boolean	Wartość true – gdy potwierdzenie zostało wygenerowane, false – gdy dla danej przesyłki nie można wygenerować wybranego potwierdzenia.
cert_content	xsd:string	Jeżeli potwierdzenie zostało wygenerowane, pole to zawiera je w postaci pliku PDF zakodowanego w standardzie base64. Jeżeli potwierdzenie nie zostało wygenerowane, zawartość pola jest pusta.

5.7. Struktura tns:AddSenderReturnObject

Pole	Typ	Opis
result	xsd:string	Status żądania: OK – żądanie poprawne ERR – żądanie błędne
result_code	xsd:string	W przypadku błędnego żądania zawiera trzycyfrowy kod błędu, zgodny z Tabelą w Rozdziale 7. W przypadku poprawnego żądania zawiera kod 000.
result_description	xsd:string	Tekstowy opis rezultatu żądania
sender_id	xsd:int	Identyfikator dodanego nadawcy (gdy result=ERR zwracany jest pusty)
shipment_id	xsd:string	Identyfikator listu z kodem aktywacyjnym (gdy result=ERR zwracany jest pusty)
sender_name	xsd:string	Nazwa dodanego nadawcy (gdy result=ERR zwracany jest pusty)
sender_address	xsd:string	Adres dodanego nadawcy (gdy result=ERR zwracany jest pusty)
sender_home_number	xsd:string	Numer domu (budynku) dodanego nadawcy (gdy result=ERR zwracany jest pusty)
sender_flat_number	xsd:string	Numer mieszkania (lokalu) dodanego nadawcy (gdy result=ERR zwracany jest pusty)
sender_post_code	xsd:string	Kod pocztowy dodanego nadawcy (gdy result=ERR zwracany jest pusty)
sender_city	xsd:string	Miasto dodanego nadawcy (gdy result=ERR zwracany jest pusty)
sender_country	xsd:string	Parametr opcjonalny – kod kraju nadawcy zgodny z ISO 3166-1 alfa-2. Jeżeli wartość parametru nie zostanie podana, przyjmowana jest wartość PL (Polska). Listę kodów

		krajów zgodną z ISO 3166-1 alfa-2 znaleźć można na stronie https://www.iso.org/iso-3166-country-codes.html
sender_fax_number	xsd:string	DEPRECATED Parametr pozostawiony dla celów zachowania kompatybilności, zawsze zwracany jest pusty.

5.8. Struktura tns:VerifySenderReturnObject

Pole	Typ	Opis
result	xsd:string	Status żądania: OK – żądanie poprawne ERR – żądanie błędne
result_code	xsd:string	W przypadku błędnego żądania zawiera trzycyfrowy kod błędu, zgodny z Tabelą w Rozdziale 7. W przypadku poprawnego żądania zawiera kod 000.
result_description	xsd:string	Tekstowy opis rezultatu żądania

5.9. Struktura tns:RemoveSenderReturnObject

Pole	Typ	Opis
result	xsd:string	Status żądania: OK – żądanie poprawne ERR – żądanie błędne
result_code	xsd:string	W przypadku błędnego żądania zawiera trzycyfrowy kod błędu, zgodny z Tabelą w Rozdziale 7. W przypadku poprawnego żądania zawiera kod 000.
result_description	xsd:string	Tekstowy opis rezultatu żądania

5.10. Struktura tns:ArrayOfDocumentFiles

Pole	Typ	Opis
Array	tns:DocumentFile[]	Tablica zawierająca obiekty struktury DocumentFile . Każdy obiekt DocumentFile zawiera zawartość pliku z dokumentami, które mają zostać wysłane. Możesz dodać dowolną liczbę obiektów struktury DocumentFile. Szczegóły w Rozdziale 0.

5.11. Struktura tns:ArrayOfRecipients

Pole	Typ	Opis
Array	tns:Recipient []	Tablica zawierająca obiekty struktury Recipient . Każdy obiekt Recipient zawiera dane pojedynczego odbiorcy przesyłki, ID grupy odbiorców lub ID pojedynczego odbiorcy z książki adresowej. Możesz dodać dowolną liczbę obiektów struktury Recipient. Szczegóły w Rozdziale 5.18.

5.12. Struktura tns:ArrayOfOptions

Pole	Typ	Opis
Array	tns:Option []	Tablica zawierająca obiekty struktury Option . Każdy obiekt Option zawiera nazwę oraz wartość parametru konfiguracji przesyłki. Szczegóły w Rozdziale 5.19.

5.13. Struktura tns:ArrayOfShipments

Pole	Typ	Opis
Array	tns:ArrayOfShipments[]	Tablica zawierająca obiekty struktury Shipment . Każdy obiekt Shipment zawiera dane o statusie pojedynczej przesyłki (wysłanej do pojedynczego odbiorcy). W przypadku błędnego żądania (result: ERR) tablica jest pusta. Szczegóły w Rozdziale 5.22.

5.14. Struktura tns:ArrayOfShipmentsPrice

Pole	Typ	Opis
Array	tns:ArrayOfShipmentsPrice []	Tablica zawierająca obiekty struktury ShipmentPrice . Każdy obiekt ShipmentPrice zawiera dane o szczegółach i cenie pojedynczej przesyłki. W przypadku błędnego żądania (result: ERR) tablica jest pusta. Szczegóły w Rozdziale 5.23.

5.15. Struktura tns:ArrayOfSenders

Pole	Typ	Opis
Array	tns:ArrayOfSenders[]	Tablica zawierająca obiekty struktury Sender . Każdy obiekt Sender zawiera dane o pojedynczym nadawcy przesyłek. W przypadku błędnego żądania (result: ERR) tablica jest pusta. Szczegóły w Rozdziale 5.21.

5.16. Struktura tns:ArrayOfConfigProfiles

Pole	Typ	Opis
Array	tns:ArrayOfConfigProfiles[]	Tablica zawierająca obiekty struktury ConfigProfile . Każdy obiekt ConfigProfile zawiera dane o pojedynczym profilu przesyłki. W przypadku błędnego żądania (result: ERR) lub braku zdefiniowanych profili przesyłek tablica jest pusta. Szczegóły w Rozdziale 5.20.

5.17. Struktura tns:DocumentFile

Pole	Typ	Opis
file_stream	xsd:string	Zawartość pojedynczego pliku (stream) w formacie PDF zakodowana w standardzie base64. Jeżeli podano również parametr file_library_id, wartość zostanie zignorowana.
file_name	xsd:string	Nazwa pliku (dla celów identyfikacji, może być dowolna). Jeżeli podano również parametr file_library_id, wartość zostanie zignorowana.
file_library_id	xsd:int	ID pliku z dokumentem znajdującego się w bibliotece plików na koncie Użytkownika, który ma zostać dodany do przesyłki. Parametr ten ma właściwość nadrzędną - jeżeli został ustawiony to ignorowane są parametry file_stream oraz file_name.

5.18. Struktura tns:Recipient

Pole	Typ	Opis
source	xsd:string	Źródło danych odbiorcy. Dostępne wartości: - inline – dane odbiorcy wprowadzone są w polach tej struktury (wartość domyślna) - group – dane odbiorców pobierane są z grupy odbiorców (wymagany parametr id) - book – dane odbiorcy pobierane są z książki adresowej (wymagany parametr id)
id	xsd:int	Gdy parametr source= group , parametr id zawiera id grupy odbiorców Gdy parametr source= book , parametr id zawiera id kontaktu w książce adresowej Gdy parametr source= inline , parametr id jest ignorowany
recipient_name	xsd:string	Zawiera nazwę odbiorcy (parametr ignorowany gdy source=group lub source=book)
recipient_address_prefix	xsd:string	Parametr opcjonalny. Zawiera prefiks adresu odbiorcy, np. „ul.”, „al.”, „pl.” (parametr ignorowany gdy source=group lub source=book)
recipient_address	xsd:string	Zawiera adres (ulicę) odbiorcy (parametr ignorowany gdy source=group lub source=book)
recipient_home_number	xsd:string	Zawiera numer domu (budynku) odbiorcy (parametr ignorowany gdy source=group lub source=book)
recipient_flat_number	xsd:string	Zawiera numer mieszkania (lokalu) odbiorcy (parametr ignorowany gdy source=group lub source=book)
recipient_post_code	xsd:string	Zawiera kod pocztowy odbiorcy (parametr ignorowany gdy source=group lub book)
recipient_city	xsd:string	Zawiera miasto odbiorcy (parametr ignorowany gdy source=group lub source=book)
recipient_country	xsd:string	Parametr opcjonalny – kod kraju odbiorcy zgodny z ISO 3166-1 alfa-2. Jeżeli wartość parametru nie zostanie podana, przyjmowana jest wartość PL (Polska). Listę kodów krajów zgodną z ISO 3166-1 alfa-2 znaleźć można na stronie https://www.iso.org/iso-3166-country-codes.html
recipient_fax_number	xsd:string	Zawiera numer telefonu odbiorcy w formacie międzynarodowym (48 na początku dla Polski) - parametr opcjonalny, (parametr ignorowany gdy source=group lub source=book) [uwaga: nazwa parametru zachowana została dla kompatybilności]

5.19. Struktura tns:Option

Pole	Typ	Opis
name	xsd:string	<p>Nazwa parametru. Dozwolone parametry (opcjonalnie):</p> <ul style="list-style-type: none"> - config_id – ID zdefiniowanego profilu przesyłki (zobacz Rozdział 6.1) - sender_id – ID zdefiniowanego nadawcy przesyłki (zobacz Rozdział 6.2) - dispatch_date – Data nadania wysyłki w odpowiednim formacie - callback_url – Adres URL skryptu, na który będą przekazywane zaktualizowane statusy przesyłek – szczegóły w Rozdziale 9. - custom_id – Opcjonalny identyfikator przesyłki podany przez Użytkownika - mpk – Miejsce powstawania kosztu - order_name – opcjonalna nazwa zlecenia w ramach którego wysyłana jest przesyłka
value	xsd:string	<p>Dla parametru:</p> <ul style="list-style-type: none"> - name=config_id parametr value powinien zawierać ID zdefiniowanego na platformie Postivo.pl profilu przesyłki (jeżeli nie zostanie podany, przyjmowana jest domyślnie konfiguracja, umożliwiająca najtańszą wysyłkę). Szczegóły w Rozdziale 6.1. - name=sender_id parametr value powinien zawierać ID zdefiniowanego w systemie nadawcy przesyłki (domyślnie wybierany jest domyślny nadawca zdefiniowany na koncie). Szczegóły w Rozdziale 6.2. - name=dispatch_date parametr value powinien zawierać datę, kiedy przesyłka ma zostać nadana. Format dla przesyłek listowych: RRRR-MM-DD, gdzie RRRR – rok, MM –miesiąc, DD – dzień, GG – godzina, MM – minuta. Jeżeli parametr dispatch_date nie zostanie podany, wysyłka nastąpi w pierwszym możliwym terminie. - name=callback_url parametr value powinien zawierać adres URL skryptu w formacie np. http://strona.pl/skrypt.php, na który będą przekazywane zaktualizowane statusy przesyłek (zobacz Rozdział 9). Jest to parametr opcjonalny. - name=custom_id parametr value powinien zawierać dowolny identyfikator przesyłki określony przez Użytkownika (max 50 znaków). Jest to parametr opcjonalny - name=mpk – parametr value powinien zawierać nazwę miejsca powstawania kosztu (opcjonalnie). Podanie parametru spowoduje, że na fakturze POST-PAID wyszczególnione będą koszty w rozbiciu na poszczególne MPK - name=order_name – parametr value powinien zawierać nazwę zlecenia w ramach którego wysyłana jest przesyłka (opcjonalnie)

5.20. Struktura tns:ConfigProfile

Pole	Typ	Opis
config_id	xsd:int	Identyfikator profilu przesyłki
config_name	xsd:string	Nazwa profilu przesyłki
paper_type	xsd:string	Rodzaj papieru, na którym drukowane będą dokumenty
paper_print_type	xsd:string	Typ wydruku dokumentów (kolorowy lub skala szarości)
envelope_type	xsd:string	Rodzaj koperty
envelope_template	xsd:string	Nazwa szablonu koperty

envelope_print_type	xsd:string	Kolor wydruku koperty
carrier	xsd:string	Operator pocztowy, który dostarczy przesyłkę do odbiorcy
dispatch_type	xsd:string	Typ listu (jeżeli dotyczy), którym wysłana zostanie przesyłka do odbiorcy

5.21. Struktura tns:Sender

Pole	Typ	Opis
sender_id	xsd:string	Identyfikator nadawcy
sender_name	xsd:string	Nazwa nadawcy (firma lub imię i nazwisko)
sender_address	xsd:string	Adres nadawcy przesyłki
sender_home_number	xsd:string	Numer domu (budynku) nadawcy przesyłki
sender_flat_number	xsd:string	Numer mieszkania (lokalu) nadawcy przesyłki
sender_post_code	xsd:string	Kod pocztowy nadawcy przesyłki
sender_city	xsd:string	Miasto nadawcy przesyłki
sender_country	xsd:string	Parametr opcjonalny – kod kraju nadawcy zgodny z ISO 3166-1 alfa-2. Jeżeli wartość parametru nie zostanie podana, przyjmowana jest wartość PL (Polska). Listę kodów krajów zgodną z ISO 3166-1 alfa-2 znaleźć można na stronie https://www.iso.org/iso-3166-country-codes.html
sender_fax_number	xsd:string	Parametr pozostawiony dla celów zachowania kompatybilności, zawsze zwracany jest pusty.
default	xsd:string	1 – w przypadku, gdy nadawca jest domyślnym na koncie, 0 – w pozostałych przypadkach
active	xsd:string	1 – w przypadku, gdy nadawca został zweryfikowany 0 – w przypadku, gdy nadawca nie został jeszcze zweryfikowany

5.22. Struktura tns:Shipment

Pole	Typ	Opis
id	xsd:string	Numer identyfikacyjny przypisany do danej przesyłki
custom_id	xsd:string	Identyfikator przesyłki podany przez Użytkownika
tracking_number	xsd:string	Numer przesyłki u danego operatora pocztowego (np. numer listu poleconego itp.)
recipient_name	xsd:string	Nazwa odbiorcy przesyłki
recipient_address	xsd:string	Adres odbiorcy przesyłki

recipient_home_number	xsd:string	Numer domu (budynku) odbiorcy przesyłki
recipient_flat_number	xsd:string	Numer mieszkania (lokalu) odbiorcy przesyłki
recipient_post_code	xsd:string	Kod pocztowy odbiorcy przesyłki
recipient_city	xsd:string	Miasto odbiorcy przesyłki
recipient_country	xsd:string	Parametr opcjonalny – kod kraju odbiorcy zgodny z ISO 3166-1 alfa-2. Jeżeli wartość parametru nie zostanie podana, przyjmowana jest wartość PL (Polska). Listę kodów krajów zgodną z ISO 3166-1 alfa-2 znaleźć można na stronie https://www.iso.org/iso-3166-country-codes.html
recipient_fax_number	xsd:string	Numer telefonu odbiorcy przesyłki [uwaga: nazwa parametru zachowana została z konieczności zachowania kompatybilności]
dispatch_carrier	xsd:string	Operator pocztowy, który dostarczy przesyłkę do odbiorcy
dispatch_type	xsd:string	Typ listu (jeżeli dotyczy), którym wysłana zostanie przesyłka do odbiorcy
page_number	xsd:int	Ilość stron dokumentów zawartych w przesyłce
price	xsd:float	Całkowity koszt netto przesyłki (wyrażony w zł polskich)
status	xsd:string	Status przesyłki: OK – jeżeli przesyłka została przyjęta do wysyłki ERR – jeżeli przesyłka nie została przyjęta do wysyłki
status_code	xsd:string	Kod statusu przesyłki, zgodnie z Tabelą w Rozdziale 8.
status_description	xsd:string	Tekstowy opis statusu przesyłki

5.23. Struktura tns:ShipmentPrice

Pole	Typ	Opis
recipient_name	xsd:string	Nazwa odbiorcy przesyłki
recipient_address	xsd:string	Adres odbiorcy przesyłki
recipient_home_number	xsd:string	Numer domu (budynku) odbiorcy przesyłki
recipient_flat_number	xsd:string	Numer mieszkania (lokalu) odbiorcy przesyłki
recipient_post_code	xsd:string	Kod pocztowy odbiorcy przesyłki
recipient_city	xsd:string	Miasto odbiorcy przesyłki
recipient_country	xsd:string	Parametr opcjonalny – kod kraju odbiorcy zgodny z ISO 3166-1 alfa-2. Jeżeli wartość parametru nie zostanie podana, przyjmowana jest wartość PL (Polska). Listę kodów krajów zgodną z ISO 3166-1 alfa-2 znaleźć można na stronie https://www.iso.org/iso-3166-country-codes.html

recipient_fax_number	xsd:string	Numer telefonu odbiorcy przesyłki [uwaga: nazwa parametru zachowana została z konieczności zachowania kompatybilności]
dispatch_carrier	xsd:string	Operator pocztowy, który dostarczy przesyłkę do odbiorcy
dispatch_type	xsd:string	Typ listu (jeżeli dotyczy), którym wysłana zostanie przesyłka do odbiorcy
page_number	xsd:int	Ilość stron dokumentów zawartych w przesyłce
price	xsd:float	Całkowity koszt netto przesyłki (wyrażony w zł polskich)

5.24. Struktura tns:SenderData

Pole	Typ	Opis
sender_name	xsd:string	Nazwa dodanego nadawcy
sender_address_prefix	xsd:string	Parametr opcjonalny. Zawiera prefiks adresu odbiorcy, np. „ul.”, „al.”, „pl.”
sender_address	xsd:string	Adres dodanego nadawcy
sender_home_number	xsd:string	Numer domu (budynku) dodanego nadawcy
sender_flat_number	xsd:string	Parametr opcjonalny. Numer mieszkania (lokalu) dodanego nadawcy
sender_post_code	xsd:string	Kod pocztowy dodanego nadawcy
sender_city	xsd:string	Miasto dodanego nadawcy
sender_country	xsd:string	Parametr opcjonalny – kod kraju nadawcy zgodny z ISO 3166-1 alfa-2. Jeżeli wartość parametru nie zostanie podana, przyjmowana jest wartość PL (Polska). Listę kodów krajów zgodną z ISO 3166-1 alfa-2 znaleźć można na stronie https://www.iso.org/iso-3166-country-codes.html
sender_fax_number	xsd:string	Parametr opcjonalny. Numer telefonu dodanego nadawcy [uwaga: nazwa parametru zachowana została z konieczności zachowania kompatybilności]

6. Definiowanie parametrów

6.1. Definiowanie nowych profili przesyłek

Korzystając z API istnieje również możliwość wskazania dodatkowych parametrów przesyłki, takich jak rodzaj papieru i kolor wydruku, rodzaj i szablon koperty, typ przesyłki (polecona czy zwykła itp.), operator pocztowy, który doręczy przesyłkę do odbiorców itd. W tym celu należy utworzyć profil z wybranymi parametrami przesyłki. Zaloguj się na swoje konto w serwisie <https://postivo.pl>, a następnie:

- Wybierz z lewego menu z sekcji „DODATKI” opcję „Personalizacja”, a następnie „Profile przesyłek”
- Kliknij na przycisk „Utwórz nowy profil przesyłek”

- Ustaw wszystkie parametry przesyłki zgodnie z Twoim wyborem i nadaj profilowi dowolną nazwę.
- Kliknij na przycisk „Dodaj profil przesyłek”.

Profil z wybranymi parametrami przesyłki zostanie zapisany pod wskazaną nazwą oraz uzyska unikalny identyfikator. W strukturze Option możesz wskazać ID profilu przesyłki (name=config_id), którą chcesz wykorzystać w celu wysłania danej przesyłki. ID profilu przesyłki uzyskasz wybierając z lewego menu z sekcji „DODATKI” opcję „Personalizacja”, a następnie „Profile przesyłek”. Zostanie wyświetlona tabela z wszystkimi dotychczas zapisanymi profilami przesyłek. Przy każdej pozycji znajduje się ID danego profilu.

6.2. Definiowanie nadawców przesyłek

Wysyłając przesyłkę należy określić, kto będzie jej zleceniodawcą. Nadawcy przesyłek muszą zostać uprzednio zdefiniowani. Opis w jaki sposób dodać i aktywować nadawcę znajduje się w instrukcji dostępnej na stronie <https://postivo.pl/pomoc-pytania-i-odpowiedzi-faq>. Nadawcę można również zdefiniować przy użyciu interfejsu API – szczegóły w Rozdziale 4.8 oraz 4.9.

W strukturze Option możesz wskazać ID nadawcy, którego chcesz wykorzystać w wysyłanej przesyłce (name=sender_id). ID nadawcy uzyskasz wybierając z lewego menu z sekcji „DODATKI” opcję „Personalizacja”, a następnie „Nadawcy przesyłek”. Zostanie wyświetlona tabela z wszystkimi zdefiniowanymi dotychczas nadawcami. Przy każdej pozycji znajduje się ID danego nadawcy, który możesz wykorzystać w momencie wywołania odpowiedniej metody.

7. Kody błędów (rezultat) żądań

Kod błędu	Opis błędu
000	Żądanie poprawne, brak błędów
001	Wprowadzone hasło jest nieprawidłowe lub konto o podanym loginie nie istnieje
002	Konto o podanym loginie nie zostało jeszcze aktywowane
003	Konto podanym loginie nie zostało jeszcze zweryfikowane przez administratora
004	Dostęp przez API do konta o podanym loginie nie został aktywowany
005	Dostęp przez API do konta o podanym loginie nie jest dozwolony z tego adresu IP
006	Wystąpił nieznaną błąd podczas sprawdzania tożsamości
010	Typ przesyłki jest nieprawidłowy
020	Brak wartości parametru document_files
021	Przekazano plik nieznanego typu
022	Nie można otworzyć przekazanego pliku

023	Przekazany plik jest uszkodzony
024	Przekazany plik ma zablokowaną opcję wydruku
025	Przekazany plik jest zabezpieczony przed przetwarzaniem
026	Konwersja przekazanego pliku nie powiodła się
027	Wystąpił błąd podczas przetwarzania przekazanych plików
028	Wystąpił nieznan problem z dostępem do zawartości przekazanego pliku
029	Błąd otwarcia pliku z biblioteki
030	Brak odbiorców przesyłki
031	Niepoprawna nazwa odbiorcy
032	Niepoprawny adres odbiorcy
033	Niepoprawny numer domu odbiorcy
034	Niepoprawny kod pocztowy odbiorcy
035	Niepoprawne miasto odbiorcy
036	Niepoprawny numer telefonu odbiorcy
037	<i>DEPRECATED</i>
038	Wskazana grupa odbiorców nie zawiera żadnych adresów odbiorców
039	<i>DEPRECATED</i>
040	Brak wskazanego odbiorcy w książce adresowej
041	Nieobsługiwany kraj odbiorcy
050	Brak zdefiniowanego profilu przesyłek o takim ID
051	Brak dostępnego nadawcy o takim ID
052	Data realizacji zlecenia jest niewłaściwa
053	Podany adres Callback URL jest nieprawidłowy
060	Wystąpił nieznan błąd
070	Brak wartości parametru dispatch_ids

071	Nieprawidłowy ID przesyłki
072	Brak przesyłki z podanym ID
080	Niepoprawna nazwa nadawcy
081	Niepoprawny adres nadawcy
082	Niepoprawny numer domu nadawcy
083	Niepoprawny kod pocztowy nadawcy
084	Niepoprawne miasto nadawcy
085	<i>DEPRECATED</i>
086	Brak zgody na wysyłkę listu z kodem weryfikacyjnym na wprowadzony adres
087	Brak wystarczających środków na koncie, aby wysłać list z kodem weryfikacyjnym
088	Wprowadzony nadawca już istnieje.
089	Wprowadzony kod weryfikacyjny dla danego nadawcy jest nieprawidłowy
090	Kraj nadawcy nie jest obsługiwany
091	Wskazany nadawca nie istnieje
092	Nie można usunąć domyślnego nadawcy

8. Kody statusów przesyłek

8.1. Kody statusów nieprzyjętych przesyłek

Kod błędu	Opis błędu
-3	Przesyłka nie została wysłana. Brak środków na koncie
-2	Przesyłka nie została wysłana. Nieudana personalizacja
-1	Przesyłka nie została wysłana. Nieudany zapis do bazy

8.2. Kody statusów prawidłowo wysłanych przesyłek

Kod statusu	Opis statusu
1	Przesyłka przyjęta do realizacji
2	Przesyłka w trakcie konfekcjonowania

3	Przesyłka przekazana operatorowi pocztowemu
4	Przesyłka doręczona do odbiorcy
5	Przesyłka zwrócona do Postivo.pl
50	<i>DEPRECATED</i>
51	<i>DEPRECATED</i>
52	<i>DEPRECATED</i>
53	<i>DEPRECATED</i>
54	<i>DEPRECATED</i>
55	<i>DEPRECATED</i>
56	<i>DEPRECATED</i>
100	Przesyłka anulowana
200	Status zewnętrzny – status pochodzący od operatora pocztowego. Treść statusu znajduje się w polu status_description.

9. Mechanizm Callback – asynchroniczna aktualizacja statusów przesyłek

Udostępniamy asynchroniczny mechanizm umożliwiający przekazanie na dowolny adres URL informacji o każdej zmianie statusu przesyłki wysłanej za pośrednictwem API. **Globalny adres Callback** można skonfigurować na stronie konfiguracji dostępu do konta poprzez API (szczegóły w rozdziale 2.2). Alternatywnie podczas wysyłania przesyłki metodą dispatch można również podać **lokalny adres Callback** (jako element callback_url parametru options – zobacz Rozdział 5.19).

Różnica między globalnym a lokalnym adresem callback polega na tym, że jeżeli w metodzie dispatch podano parametr callback_url to statusy aktualizacyjne tej przesyłki będą wysyłane wyłącznie na ten adres. Jeżeli nie podano tego parametru, wówczas statusy będą przekazywane na globalny adres callback (oczywiście jeżeli został on zdefiniowany).

Po każdej zmianie statusu przesyłki wysłanej za pośrednictwem API system platformy Postivo.pl wywoła właściwy adres Callback przekazując do niego metodą GET następujące parametry:

- **dispatch_id** - identyfikator przesyłki, której status został zmieniony
- **status_code** – Aktualny status przesyłki zgodny z Tabelą w Rozdziale 8.2
- **date** - Data zmiany statusu w formacie RRRR-MM-DD GG:MM:SS

Jeżeli z jakiegóż przyczyny podany adres URL Callback nie jest w danym momencie osiągalny, wówczas system ponowi jego wywołanie kolejno po 5, 15, 60 minutach i po 6 godzinach.

Przykład:

Podczas wysyłania przesyłki metodą dispatch nie został podany parametr callback_url. Jednak przy konfiguracji dostępu do konta poprzez API ustawiony został globalny adres Callback: <http://jakas-domena.pl/skrypt.php>. W momencie zmiany statusu przesyłki nr T00001224 zostanie wywołany następujący adres: http://jakas-domena.pl/skrypt.php?dispatch_id=T00001224&status_code=3&date=2011-01-21+13%3A22%3A14.

10. Przykładowe implementacje w różnych środowiskach programistycznych

Przygotowaliśmy gotowe fragmenty kodu w różnych środowiskach programistycznych, obrazujące sposób integracji z platformą Postivo.pl. Udostępniamy gotowy kod między innymi dla: C#, JAVA, Perl, PHP4, PHP5, Python 3. Dodatkowe informacje oraz gotowe do pobrania kody zostały umieszczone na stronach serwisu pod adresem <https://postivo.pl/przykladowe-zastosowania-api/#przyklady-implementacji>.

11. System testowy API (sandbox)

Udostępniamy platformę testową umożliwiającą sprawdzenie poprawności integracji własnego systemu z naszą platformą bez konieczności ponoszenia jakichkolwiek kosztów. Przesyłki nadane za pośrednictwem systemu testowego w rzeczywistości nie zostaną wysłane do odbiorców. System testowy ma pełną funkcjonalność systemu produkcyjnego.

Web Service SOAP systemu testowego (sandbox) dostępny jest pod adresami:

- <https://api.postivo.pl/webservices-sandbox.php> (połączenie szyfrowane - SSL)

WSDL systemu testowego (sandbox) dostępny jest pod adresami:

- <https://api.postivo.pl/webservices-sandbox.php?wsdl> (połączenie szyfrowane - SSL)

Po wysłaniu przesyłki do systemu testowego jej status jest zmieniany na „Przesyłka przekazana operatorowi pocztowemu” (kod statusu: 3).

System testowy (sandbox) nie wymaga założenia oddzielnego konta, a panel użytkownika znajduje się pod adresem: <https://sandbox.postivo.pl>.

12. Dodatkowe informacje i kontakt

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

POSTIVO.PL Spółka z ograniczoną odpowiedzialnością Spółka komandytowa

ul. Litewska 4B/16

35-302 Rzeszów

tel. +48 17 717 37 60 (od pn-pt w godz. 8-16),

e-mail: biuro@postivo.pl