



## REST API v.3.0

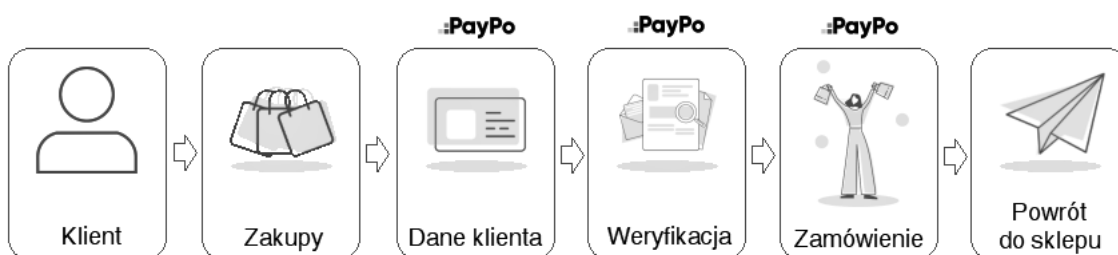
Dokument zawiera specyfikację techniczną REST API PayPo.pl w wersji 3.0.

Wersja dokumentu	Data	Wykaz zmian
3.0_r3	2020.07.16	Powiadomienia o zmianie statusu i kwoty zamówienia
3.0_r2	2020.06.01	Uwierzytelnienie komunikatów OAuth
3.0_r1	2020.05.12	Doprecyzowanie opisu atrybutów adresowych, autoryzacja HMAC
3.0_r0	2020.04.28	Definicje podstawowych endpointów, cykl życia transakcji.

# 1. Opis usługi

PayPo świadczy usługę odroczonej płatności dla klientów sklepów internetowych (Kupujących). Rejestracja nowej transakcji jest realizowana online z wykorzystaniem API, analogicznie jak w popularnych systemach płatności.

## Przebieg transakcji



- Kupujący po wybraniu produktów wybiera odroczoną płatność za zakupy z PayPo.
- Dane Kupującego są przekazywane do serwisu PayPo.
- System PayPo weryfikuje wiarygodność finansową Kupującego.
- Pozytywna weryfikacja kończy się zawarciem umowy pożyczki.
- Kupujący jest przekierowany zwrótnie na stronę sklepu.

## Dalsze kroki

- Sprzedający potwierdza realizację zamówienia w systemie PayPo.
- Przelew środków do Sprzedającego następuje po aktualizacji statusu zamówienia lub w terminie określonym w umowie.
- Kupujący opłaca koszty zakupów do PayPo w ciągu 30 dni lub rozkłada je na raty.

# 2. Integracja – informacje ogólne

API dostępne jest pod adresem: <https://api.paypo.pl/v3/>

Adres serwera testowego: <http://api.sandbox.paypo.pl/v3/>

PayPo API bazuje na architekturze REST. Komunikacja realizowana jest z wykorzystaniem protokołu HTTP (metody POST/GET/PATCH), dane wymieniane są w

formacie JSON. Podstawowym zasobem udostępnionym Sprzedającym są transakcje (/transactions).

Przykładowa rejestracja płatności:

```
curl -X POST https://api.sandbox.paypo.pl/v3/transactions \
-H "Accept: application/json" \
-H "Content-Type: application/json" \
-H "Authorization: Bearer
eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJhdWQiOiJmb28iLCJqdGkiOiJmN
jNiOGUzY2M3NGMyNWZmMjc0ODY1YjIwYTExYzJkYTM4ZjZlYWRiMWFjYmY1NWY1Yz
xMGlxNzY2MzZGZmMTEyYTF1ZjhjZDAwYzZmZiIsImh0CI6MTU5MTIxODEyNSwib
mJmIjoxNTkxMjE4MTI1LCJleHAiOiJlOTIyMTk5MjUsInN1YiI6IiIsInNjb3BlcyI
6WyJtZXJjaGFudCJdfQ.EUBTt2xcI11781msW2d2-
OIMwPuwVrHxgsmLkY2QPNGEfzQLIDBFnibSPTPQN-
V8FuidiQ7h10NCurNMLNGL79IwTLls_G1CX51QB0f1ma6wrsX7yxbKRKUY6s9kFkL
p_5YCzqveePnj2KdkG1lhI0Cq4purczMkXKbjf71PLnvCcCycTXoPqcR1222gZSVq
GTvx8J4SN2LWSIpsKWRZr7pebzQgaSeu13RApNFJ2RxV7reyRr9xRiR0Tr06nFNiTN
OB3mVQ1p0vNmn7lWxyQx7x3L1mIDJ5gh4_x9kpKYQnKdfiBdVaJJN1zpZLd0v2nc55
35xXyVaUyilCsDxA" \
-d '{
  "merchantId": "19c692be-a893-468c-a65f-b8de442e5443",
  "order": {
    "referenceId": "ord_98765/20",
    "amount": "24900",
    "billingAddress": {
      "street": "Domaniewska",
      "building": "37",
      "flat": "205",
      "zip": "02-672",
      "city": "Warszawa"
    },
    "shippingAddress": {
      "street": "Domaniewska",
      "building": "37",
      "flat": "205",
      "zip": "02-672",
      "city": "Warszawa"
    },
    "shipment": "0",
  },
  "customer": {
    "name": "Anna",
    "surname": "Nowak",
    "email": "anna.n@paypotest.pl",
    "phone": "500123456"
  },
  "configuration": {
    "returnUrl": "https://merchant.com/complete",
    "notifyUrl": "https://merchant.com/notify",
    "cancelUrl": "https://merchant.com/cancel"
  }
}'
```

## 3. Tworzenie nowego zamówienia

### 3.1 Rejestracja płatności

Adres URL: <https://api.paypo.pl/v3/transactions>

W celu utworzenia nowej sesji płatności należy przekazać do PayPo podstawowe dane transakcji. Po prawidłowej weryfikacji komunikatu w odpowiedzi przekazany zostanie identyfikator transakcji oraz adres URL formularza weryfikacyjnego, pod który należy przekierować klienta sklepu.

Po weryfikacji tożsamości i potwierdzeniu wiarygodności finansowej Kupującego, system PayPo przekieruje klienta zwrótnie na stronę sklepu oraz wyśle powiadomienie o zmianie statusu zamówienia. Cykl życia transakcji został opisany w p. 4. dokumentacji.

Parametry wejściowe żądania (HTTP POST):

Nazwa parametru	Typ	Wymagany	Opis
merchantId	uuid	T	Unikalny identyfikator sprzedawcy nadany przez PayPo, dostępny z poziomu panelu Sprzedawcy
shopId	uuid	N	Identyfikator sklepu nadany przez PayPo (opcjonalnie w przypadku kiedy Sprzedawca lub Operator Płatności integruje osobne punkty sprzedaży)
order	{ }	T	Dane zamówienia:
- referenceld	string	T	Identyfikator transakcji po stronie sprzedawcy, wykorzystywany do identyfikacji przelewów
- providerId	string	N	Identyfikator transakcji w systemie Operatora Płatności (przy integracji przez Operatora)
- amount	int	T	Kwota zamówienia w groszach
- description	string	N	Opis / tytuł transakcji
- additionalInfo	{ }	N	Dodatkowy opis zamówienia, np. zawartość koszyka

- shipment	int	N	Metoda dostawy: 0 - kurier (domyślnie) 1 - dostawa do punktu (np. UPS Access point, DHL Parcel Shop) 2 - paczkomat 3 - paczka w RUCHu 4 - odbiór w sklepie (click&collect)
- billingAddress	{ }	T	Adres rozliczeniowy:
-- street	string	T	Ulica
-- building	string	T	Numer domu
-- flat	string	T	Numer lokalu
-- zip	string	T	Kod pocztowy
-- city	string	T	Miasto
-- country	string	N	Kraj (domyślnie PL)
- shippingAddress	{ }	T	Adres dostawy:
-- street	string	T	Ulica
-- building	string	T	Numer domu
-- flat	string	T	Numer lokalu
-- zip	string	T	Kod pocztowy
-- city	string	T	Miasto
-- country	string	N	Kraj (domyślnie PL)
customer	{ }	T	Dane Kupującego:
- name	string	T	Imię
- surname	string	T	Nazwisko
- email	string	T	Adres email
- phone	string	T	Numer telefonu komórkowego
configuration	{ }	T	Adresy powrotu i notyfikacji:
- returnUrl	string	T	Adres powrotu po przetworzeniu płatności
- notifyUrl	string	T	Adres, na który wysyłane będą powiadomienia
- cancelUrl	string	N	Opcjonalny adres powrotu w przypadku rezygnacji klienta z płatności PayPal

Uwierzytelnienie żądań zostało opisane szczegółowo w p. 9.1. dokumentacji. Od wersji 3.0 interfejsu domyślną metodą jest przekazywanie w nagłówku komunikatu tokenu uwierzytelniającego OAuth2.

### 3.2 Odpowiedź serwera

W odpowiedzi na wywołanie metody /transactions PayPo zwraca status operacji oraz URL formularza, pod który należy przekierować Kupującego.

Przykładowa odpowiedź serwera:

```
HTTP 201 Created
{
  "transactionId": "5909da74-af95-41e9-b8e2-12e61c3c6f27",
  "redirectUrl": "https://process.sandbox.paypo.pl/5909da74-af95-41e9-b8e2-12e61c3c6f27"
}
```

Komunikaty błędów zostały opisane w p. 10. dokumentacji.

### 3.3 Weryfikacja danych Kupującego i powrót do sklepu

Po przekierowaniu do PayPo następuje uzupełnienie i weryfikacja danych Kupującego. Po przeprosowaniu płatności Klient jest przekierowywany zwrótnie na adres returnUrl. W parametrach GET adresu powrotnego przekazywany jest status płatności, w celu wyświetlenia odpowiedniego komunikatu w przeglądarce.

[https://merchant.com/thankyou?status=\[OK|ERR\]](https://merchant.com/thankyou?status=[OK|ERR])

Jeżeli w komunikacie inicjującym transakcję został dodatkowo przesłany atrybut returnUrl, rezygnacja w trakcie procesu weryfikacji skutkuje przekierowaniem na podany adres (np. ekran wyboru alternatywnej metody płatności przy checkoucie).

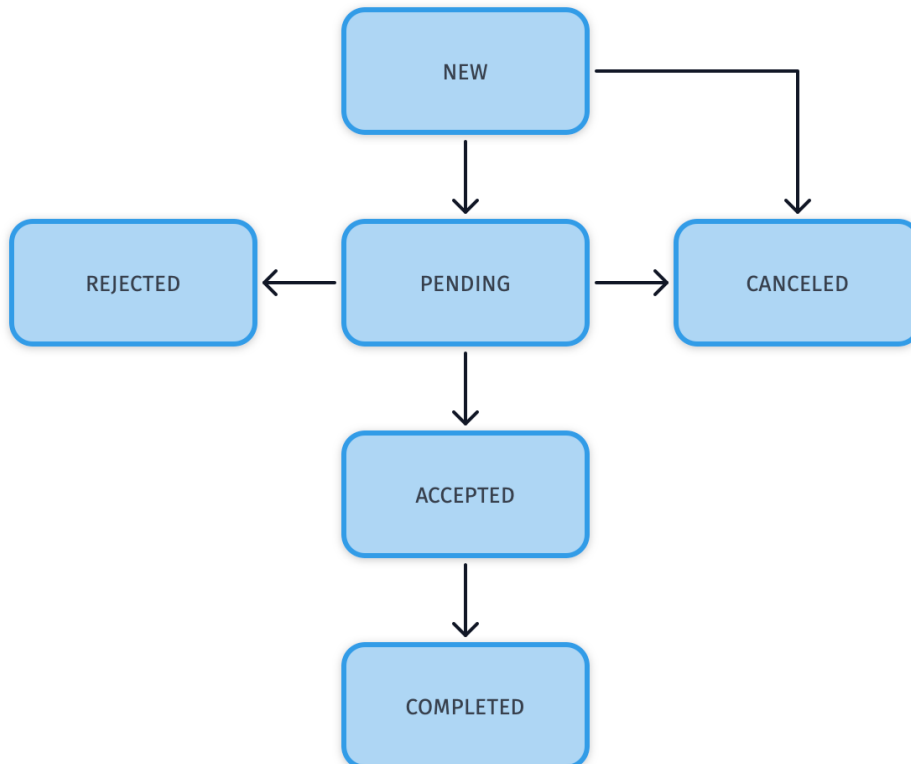
## 4. Statusy i cykl życia transakcji

Statusy płatności:

Status	Opis
NEW	Nowa transakcja
PENDING	Płatność w trakcie procesowania, oczekuje na potwierdzenie tożsamości Kupującego lub potwierdzenie dostępności środków
ACCEPTED	Płatność zaakceptowana

COMPLETED	Transakcja potwierdzona przez Sprzedawcę, płatność gotowa do rozliczenia
REJECTED	Odmowa realizacji płatności
CANCELED	Transakcja anulowana
ERROR	Błąd przetwarzania

Diagram przejść:



## 5. Powiadomienia

Po zmianie statusu płatności szczegółowa informacja wysyłana jest asynchronicznie na adres `notifyUrl`. System Sprzedającego w odpowiedzi na powiadomienie powinien zwrócić HTTP status 200.

Parametry notyfikacji (HTTP POST):

Nazwa parametru	Typ	Opis
<code>merchantId</code>	uuid	Identyfikator sprzedawcy
<code>shopId</code>	uuid	Identyfikator POS (jeśli był podany przy rejestracji)

referenceId	string	Identyfikator transakcji po stronie sprzedawcy
transactionId	uuid	Identyfikator transakcji PayPo
transactionStatus	string	Status zamówienia, zgodnie z tabelą w p. 4
amount	int	Aktualna kwota transakcji w groszach
lastUpdate	timestamp	Data ostatniej aktualizacji
message	string	Opis statusu operacji

Notyfikacje zawierają w nagłówkach dodatkowo typ dokumentu oraz skrót uwierzytelniający. Algorytm wyliczania i weryfikacji skrótu został opisany w p.9.2.

Przykładowa notyfikacja:

```
POST /notifyUrl HTTP/1.1
Content-Type: application/json
X-PayPo-Signature: dfcfcfd0cbed7bbbd11b7d23f3a2e7a7458ecf90
{
  "merchantId": "19c692be-a893-468c-a65f-b8de442e5443",
  "referenceId": "ord_987654",
  "transactionId": "00102030",
  "transactionStatus": "PENDING",
  "amount": "24900",
  "lastUpdate": "2020-03-05T10:54:02",
  "message": "Payment request awaiting confirmation",
}
```

W przypadku braku odpowiedzi lub odebrania statusu innego niż HTTP 200 system PayPo podejmie próbę ponowienia notyfikacji zgodnie z poniższym schematem (konfiguracja ponowień może wyglądać inaczej w środowiskach nieprodukcyjnych).

Pierwsza godzina po poprawnie zweryfikowanej transakcji	Co 10 minut
Kolejne 5h	Co 20 minut
Kolejne 18h	Co 60 minut

## 6. Potwierdzenie transakcji

Po pozytywnej weryfikacji płatności, przestaniu notyfikacji ze statusem ACCEPTED i przekierowaniu Kupującego na stronę sklepu, system PayPo oczekuje na potwierdzenie realizacji zamówienia przez Sprzedawcę. Niepotwierdzone zamówienie będzie aktywne przez 72h, po czym zostanie automatycznie anulowane.



W celu potwierdzenia zamówienia należy ustawić status COMPLETED. Istnieje możliwość automatycznego potwierdzania każdego zamówienia z zaakceptowaną płatnością, konfiguracja jest przygotowywana indywidualnie na prośbę Sprzedawcy.

Adres URL: [https://api.paypo.pl/v3/transactions/\\${UUID}](https://api.paypo.pl/v3/transactions/${UUID})

Parametry wejściowe (HTTP PATCH):

Nazwa parametru	Wartość	Opis
status	COMPLETED	Potwierdzenie realizacji zamówienia

Prawidłowe przyjęcie żądania zostanie potwierdzone odpowiedzią ze statusem 200. Po aktualizacji statusu płatności odpowiednia notyfikacja zostanie przesłana na adres `notifyUrl` przekazany przy rejestracji płatności.

Przykładowa odpowiedź:

```
HTTP 200 OK
{
  "status": "200",
  "message": "Transaction updated successfully",
}
```

## 7. Weryfikacja statusu płatności

Status zamówienia można odczytać w dowolnym momencie wywołując metodę HTTP GET. Żądanie powinno być uwierzytelnione tokenem OAuth.

Adres URL: [https://api.paypo.pl/v3/transactions/\\${UUID}](https://api.paypo.pl/v3/transactions/${UUID})

Przykładowa odpowiedź:

```
HTTP 200 OK
{
  "merchantId": "19c692be-a893-468c-a65f-b8de442e5443",
  "referenceId": "ord_987654",
  "transactionId": "cd975bc6-a755-4141-b7a0-d7e8f7a308ef",
  "transactionStatus": "COMPLETED",
  "amount": "24900",
  "settlementStatus": "PAID",
  "lastUpdate": "2020-03-05T10:54:02",
}
```

## 8. Modyfikacja atrybutów transakcji

Sprzedający może modyfikować status transakcji oraz kwotę zamówienia. Dozwolone zmiany statusu to anulowanie (CANCELED) lub potwierdzenie (COMPLETED, opisane wyżej). Do modyfikacji kwoty zamówienia służy dedykowana metoda /refunds.

### 8.1 Anulowanie płatności

Adres URL: [https://api.paypo.pl/v3/transactions/\\${UUID}](https://api.paypo.pl/v3/transactions/${UUID})

Parametry wejściowe (HTTP PATCH):

Nazwa parametru	Wartość	Opis
status	CANCELED	Anulowanie zamówienia

Anulowanie płatności może zostać zlecone przez Sprzedającego w przypadku braku możliwości realizacji zamówienia. Wywołanie operacji anulowania potwierdzonej (status COMPLETED) transakcji zostanie odrzucone z błędem 409 Conflict.

Przykładowa odpowiedź:

```
HTTP 200 OK
{
  "status": "200",
  "message": "Transaction updated successfully",
}
```

Prawidłowe przetworzenie zmiany statusu płatności zostanie dodatkowo potwierdzone w notyfikacji przesłanej na adres notifyUrl. Ewentualne pobrane środki zostaną zwrócone na konto Kupującego.

### 8.2 Zwroty

Zwrot części lub całości środków za zamówienie jest możliwy dla transakcji, które zostały zaakceptowane (status ACCEPTED lub COMPLETED). Ewentualne pobrane środki zostaną zwrócone na konto Kupującego.

Adres URL: [https://api.paypo.pl/v3/transactions/\\${UUID}/refunds](https://api.paypo.pl/v3/transactions/${UUID}/refunds)

Parametry wejściowe (HTTP POST):

Nazwa parametru	Typ	Opis
amount	int	Kwota zwrotu w groszach

Ze względu na specyfikę usługi odroczonej płatności (kredyt konsumencki) nie jest możliwe podwyższenie wartości zamówienia.

Możliwa jest realizacja kilku następujących po sobie zwrotów częściowych pod warunkiem, że suma przesłanych kwot zwrotu nie przekroczy wartości zamówienia. W przypadku zwrotu całości zakupu atrybut amount powinien zawierać aktualną wartość zamówienia.

Przykładowe żądanie dla pełnego zwrotu:

```
curl -X POST https://api.paypo.pl/v3/transactions/cd975bc6-
a755-4141-b7a0-d7e8f7a308ef/refunds \
-H "Content-Type: application/json" \
-H "Authorization: Bearer fLEsy5wk9jQKK0NjbF57sb0wa4BdCPu1" \
-d '{"amount": "24900"}'
```

Przykładowa odpowiedź:

```
HTTP 200 OK
{
  "status": „200”,
  "message": "Refund request accepted",
}
```

Analogicznie jak w przypadku zmiany statusu, prawidłowe przeprocesowanie zwrotu zostanie potwierdzone w notyfikacji ze zaktualizowaną kwotą zamówienia przesłanej na adres `notifyUrl`.

## 9. Uwierzytelnienie

PayPo API wymaga uwierzytelnienia komunikacji dla wszystkich metod HTTP modyfikujących transakcje płatności - POST oraz PATCH oraz żądań HTTP GET.

Wartości atrybutów wymaganych do podpisywania wiadomości: identyfikatora sprzedawcy `merchantId`, unikalnego klucza `merchantApiKey` oraz `Oauth ClientId` i `ClientSecret` są dostępne są z poziomu panelu transakcyjnego PayPo.

### 9.1 OAuth2

Domyślną i rekomendowaną metodą uwierzytelnienia żądań z systemu Sprzedawcy jest OAuth2. PayPo implementuje metodę zgodnie ze specyfikacją protokołu <https://oauth.net/2/>.

W celu pobrania tokenu uwierzytelniającego należy wywołać metodę `/oauth/tokens`. Zwrócony ciąg jest zakodowany zgodnie ze standardem JWT (<https://tools.ietf.org/html/rfc7519>). Domyślnie token ważny jest przez 30 minut.

Przykładowe żądanie:

```
curl ,https://api.sandbox.paypo.pl/v3/oauth/tokens' \  
-H 'Accept: application/json' \  
-H 'Content-Type: application/x-www-form-urlencoded' \  
--data-raw \  
'grant_type=client_credentials&client_id=S{ClientId} \  
&client_secret=${ClientSecret}'
```

Wyznaczony skrót należy przekazać w nagłówku żądania:

```
POST /v3/transactions  
Authorization: Bearer  
eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJhdWQiOiJmb28iLCJqdGkiOiI3ZmFkNmM5MTA1NWYxNzNiNDY3MjFlNW00TAYzAxYjVhZjlnN2I2YjA2ZGI0TdjYTM5YmU5ZjgzZDBhNzVmOTdlMzZkNWxNzRmOGEwYSIsImVhZCI6MTU5MTI1NjU0MywibmJmIjoxNTkxMjU2NTQzLCJleHAiOiJE10TEyNTgzNDIsInN1YiI6IiIsInNjb3BlcyI6WyJtZXJjaGFudCJdfQ.Polvo57N7bsJ_iJrKUD8Y_05scCnYMQ3JRJv7fBe5Rs2bd2rGNC0U76L9IE6cch05xC0IoVkiNB6CrrLt7MmakBUopP4s5aVHY6WW9ZH7D5JvsnaCSPz9M9MwCQwsBrg6vLW_VHaKltlpLZsTIktEBBihFeWHBGo89y_5CsZ070Ys4lSBjhmKE3TXZqbSI93kAN63Bv-y9BhnBomtWSc1CV_UMrELhSm5Fys2Pb_51oB8CESdHcg4Hoja8y6nfHp6u4zizttPEg7BEmq6ZHAR-9Mmt0JAVPv9IK0g5UpL_xArtI2I07VEaDslva92WhfKW2G4PH03iUAtFsGWBaDWg
```

## 9.2 HMAC

Schemat uwierzytelnienia ze skrótem wiadomości przekazywanym w nagłówku jest wykorzystywany w komunikacji zwrotnej. Do wyliczenia wartości skrótu wykorzystywane są poniższe dane połączone znakiem „+”:

- metoda HTTP: POST
- URL dla notyfikacji, np. `https://merchant.com/notifyUrl/{referenceId}`
- treść komunikatu w formacie JSON

```
$json = json_encode($payload, JSON_UNESCAPED_SLASHES |  
JSON_UNESCAPED_UNICODE);  
$data = $method."+".$endpoint."+".$json; // "POST+notifyUrl+  
{payload}"  
$hash = base64_encode(hash_hmac('sha256', $data,  
$merchant_api_key, true));
```

Wyznaczony skrót zostanie przekazany w nagłówku notyfikacji:

```
POST /notifyUrl  
X-PayPo-Signature: fLEsy5wk9jQKK0Njbf57sbA0wa+BdCPulGa5TJIfpI4=
```

Weryfikacja skrótu po stronie systemu Sprzedawcy nie jest wymagana, ale zalecana ze względów bezpieczeństwa.

## 10. Statusy operacji

### Statusy HTTP

W odpowiedzi na każde zapytanie zwracany jest status HTTP informujący o wyniku przetwarzania żądania.

**200 OK** – prawidłowe przyjęcie komunikatu

**201 Created** – prawidłowe utworzenie zasobu

**400 Bad Request** – niepoprawne dane lub format komunikatu

**401 Unauthorized** – błąd uwierzytelnienia, np nieważny token

**403 Forbidden** – brak uprawnień do zasobu

**404 Not Found** – URL lub zasób nie został odnaleziony

**409 Conflict** – konflikt identyfikacji zasobu (przykładowo numer zamówienia PayPal i numer zamówienia Sprzedającego identyfikują różne obiekty)

**503 Service Unavailable** – usługa tymczasowo niedostępna

Wywołania API zwracają obiekt JSON zarówno w przypadku sukcesu jak i dla błędów przetwarzania. Zwracany kod odpowiedzi HTTP jest podstawowym wyznacznikiem statusu operacji. Dodatkowo w przypadku niektórych błędów może zostać zwrócona szczegółowa informacja zwrotna.

Przykładowa odpowiedź w przypadku błędu:

```
HTTP 400
{
  "code": "400",
  "message": "Bad request",
  "errors": {
    { "path": „billingAddress.zip”, „message”: "Missing
mandatory parameter" },
  }
}
```