



Standard Checkout Integration Guide

目次

1	概要	3
1.1	実装概要	3
2	Payment Experience	3
2.1	PayPal 決済	3
2.2	ベストプラクティス	4
2.3	カート画面からの PayPal 決済	5
2.4	支払い方法画面からの PayPal 決済	6
3	PayPal アカウント登録・API 認証情報取得	6
3.1	PayPal ビジネスアカウントを作成する	6
3.2	テスト用 PayPal アカウントを作成する	6
3.3	REST API アプリの作成	7
3.4	API 認証情報を取得する	8
4	実装方法	9
4.1	JavaScript SDK	10
4.2	PayPal 決済	10
4.3	キャンセル・返金	20
4.4	処理結果	20
4.5	エラー処理	22
5	ドキュメント	22

1 概要

1.1 実装概要

実装していただく内容は以下のとおりです。

- PayPal 決済・簡易クレジットカード決済

以下の技術を使用します。

- PayPal JavaScript SDK
- Orders API v2
- Payments API v2

2 Payment Experience

2.1 PayPal 決済・簡易クレジットカード決済



PayPal を使用すると、お客様はオンラインフォームをスキップして数回クリックするだけで注文を完了させることができ、チェックアウト時のコンバージョンを高めることができます。ウェブ上では、PayPal ボタンは PayPal によってレンダリングされます。ブランドや UI ガイドラインに合わせてボタンのスタイルをカスタマイズすることができます。

PayPal 決済画面

PayPal ボタンをクリックすると以下のログイン画面がポップアップ画面で表示されます。



クレジットカード決済画面

「デビットカードまたはクレジットカード」ボタンをクリックすると以下のようにボタンの下にカード情報入力フォームが表示されます。貴社から氏名、住所を渡した場合左記左の図のように請求先住所などは非表示になります。

デビットカードまたはクレジットカード

×

カード番号

有効期限 セキュリティコード

私は法定年齢に達していることを認め、PayPalの[プライバシーステートメント](#)に同意します。

今すぐ支払う

デビットカードまたはクレジットカード

×

カード番号

有効期限 セキュリティコード

請求先住所 ● ▼

姓 名

郵便番号

都道府県 ▼

市区町村

番地

建物名・階・部屋番号

モバイル
+81

メール

請求先住所に配送

私は法定年齢に達していることを認め、PayPalの[プライバシーステートメント](#)に同意します。

今すぐ支払う

住所・氏名を送信する場合

住所・氏名を送信しない場合

2.2 ベストプラクティス

支払い画面、FAQ、お支払い方法を説明しているページでは、利用可能なすべての決済手段を明記しましょう。ペイパルの説明や、支払いができる銀行名、決済方法のバナーを掲載することで、お客様が一目で利用可能な決済手段がわかるようになります。

4



ロゴ

<https://www.paypal.com/jp/webapps/mpp/developer/logos-buttons/download>

「ペイパルについて」リンク

<https://www.paypal.com/jp/webapps/mpp/lp/about-paypal>

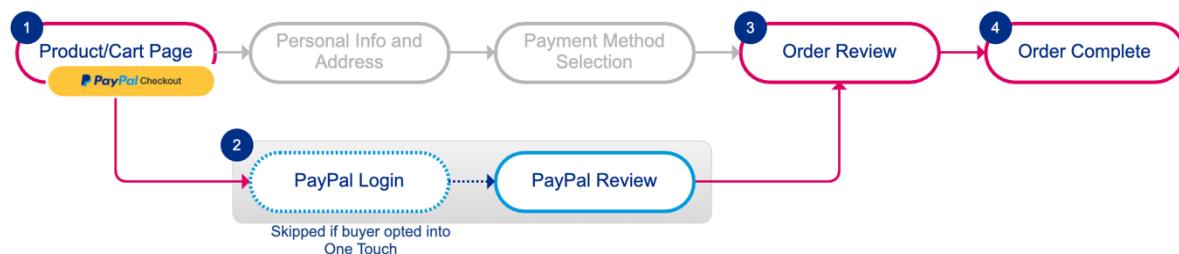
「銀行口座のご利用について」リンク

<https://www.paypal.com/jp/webapps/mpp/lp/set-up-bank>

PayPal ウォレットは、他の支払方法が利用可能な場合には、常に、他の支払方法と同等の重要性と機能を備えた状態で、他の支払方法と同等に含める必要があります。これにより、消費者は利用可能なすべての支払い方法の中から公平に選択することができます。ウェブとモバイルの統合に関する以下のベストプラクティスに従うことで、お客様に最適な決済体験を提供し、コンバージョンを向上させることができます。

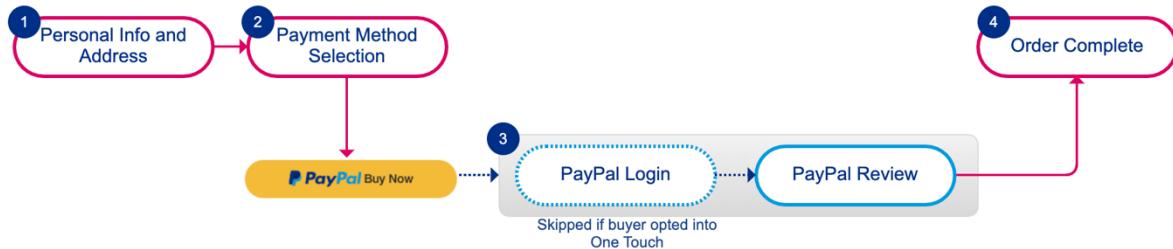
<https://developer.paypal.com/docs/checkout/best-practices/>

2.3 カート画面からの PayPal 決済



このフローは、お客様が名前、電子メール、請求先住所、その他の詳細などの余分なフィールドを入力することなく購入を完了できるようにすることで、チェックアウトを合理化します。PayPal チェックアウトボタンをクリックすると、お客様はすぐに PayPal にリダイレクトされ、トランザクションの詳細を確認することができます。お客様はその後、マーチャント名に返され、その情報は、最終確認ページを事前に入力するために PayPal によって返されます。

2.4 支払い方法画面からの PayPal 決済



このフローは、PayPal が他の支払いオプション、通常はクレジットカードやデビットカードと一緒に提示されているチェックアウトプロセスの終わりに向かって開始されます。お客様は PayPal を選択し、ログインして購入を確認するために PayPal サイトにリダイレクトされます。トータルでの PayPal 体験は、他の支払い方法と同等でなければなりません。

PayPal セラープロテクションの対象となるには、お客様から提供された配送先住所を PayPal に渡す必要があります。消費者が選択できるようにするために、PayPal の総合的な経験が他の支払い方法と同等でなければなりません。

3 PayPal アカウント登録・API 認証情報取得

3.1 PayPal ビジネスマーケットプレイスアカウントを作成する

<https://www.paypal.com/jp/webapps/mpp/merchant> で PayPal ビジネスマーケットプレイスアカウントを作成してください。

3.2 テスト用 PayPal アカウントを作成する

PayPal Sandbox 環境で取引をシミュレーションできるように、テスト用のバイヤーとマーチャントのアカウントを作成します。(デフォルトで自動作成されているアカウントを使用することもできます。)

PayPal Sandbox とは

PayPal Sandbox <https://sandbox.paypal.com> は、本番の PayPal 環境 <https://paypal.com> のテスト版です。ここでは、本番に入る前にトランザクションをテストし、実装が期待通りに動作していることを確認します。

Environment	Base URL	Test here
Sandbox	https://api-m.sandbox.paypal.com	www.sandbox.paypal.com
Live	https://api-m.paypal.com	www.paypal.com

Sandbox アカウントの作成

Sandbox のテストアカウントを作成するには

1. Dashboard にログインします。 <https://developer.paypal.com/developer/accounts/>
2. Sandbox で Accounts を選択します。
3. Create Account (アカウントの作成) を選択します。

4. Personal または Business を選択し、テストの買い手または売り手の国/地域を選択します。
5. Create を選択します。PayPal の残高、クレジットカード、銀行口座情報などが記載された Sandbox メールアカウントが生成されます。

Sandbox アカウントのデフォルト設定を変更するには、以下の手順に従います。

1. Manage Accounts で (...) を選択し、View/Edit Account を選択します。
2. Edit を選択し、変更を行います。
3. 完了したら、Save を選択します。

Sandbox にログインするには

1. <https://www.sandbox.paypal.com/> にアクセスします。
2. テスト購入者または販売者のメールアドレスとパスワードを使用してログインします。
3. 実装する際には、テストバイヤーとセラーのアカウントを交互に使用して、本番前に取引が期待通りに動作しているかどうかを確認することができます。

3.3 REST API アプリの作成

- Live 環境用の REST App を作成するためには、PayPal アカウントのメールアドレスの確認が完了している必要があります。まだ確認を行っていない場合は、PayPal アカウント作成時にメールが送信されていますのでご確認ください。以下の方法で確認メールを再送することも可能です。

メールアドレスを確認する方法

<https://www.paypal.com/jp/smarthelp/article/%E3%83%A1%E3%83%BC%E3%83%AB%E3%82%A2%E3%83%89%E3%83%AC%E3%82%B9%E3%82%92%E7%A2%BA%E8%AA%8D%E3%81%99%E3%82%8B%E6%96%B9%E6%B3%95%E3%82%92%E6%95%99%E3%81%88%E3%81%A6%E3%81%8F%E3%81%A0%E3%81%95%E3%81%84%E3%80%82-faq1485>

API 認証情報を取得する最初のステップは、REST API アプリを作成することです。アプリは、PayPal API への呼び出しを認証するために使用される保存された設定のセットです。アカウントに関連付けられた複数のアプリを持つことができます。

REST API アプリを作成するには、以下の手順に従います。

1. ダッシュボード

<https://www.paypal.com/signin?returnUri=https%3A%2F%2Fdeveloper.paypal.com%2Fdeveloper%2Fapplications> にログインします。

2. Create app を選択します。
3. App Name を入力します。
4. Create app を選択します。

App Name	Type	Actions
Create your first app to view it here.		

3.4 API 認証情報を取得する

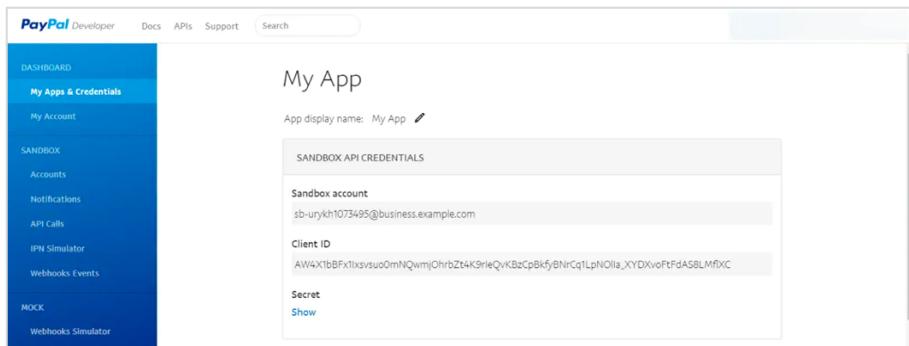
API 認証情報は、クライアント ID とシークレットで構成されており、サンドボックスとライブ環境で API 呼び出しを行い、トランザクションを処理するために必要となります。

サンドボックス API 認証情報を取得するには、以下の手順に従います。

1. ダッシュボード

<https://www.paypal.com/signin?returnUri=https%3A%2F%2Fdeveloper.paypal.com%2Fdeveloper%2Fapplications> にログインします。

2. My Apps & Credentials で、リストから REST API アプリを選択します。
3. Client ID をコピーします。
4. Secret で Show を選択して、Secret を表示します。



4 実装方法

REST API URL

PayPal REST API と通信するには、サンドボックス環境または本番環境のいずれかで、JSON 形式の HTTP リクエストを API エンドポイントに渡します。

Environment	Base URL
Sandbox	https://api-m.sandbox.paypal.com
本番環境	https://api-m.paypal.com

API べき等性

<https://developer.paypal.com/api/rest/reference/idempotency/>

サーバーがリソースに対するアクションを複数回作成したり完了したりすることなく、何度もコールを行うことができます。ネットワークタイムアウトや HTTP 5xx ステータスコードで失敗した呼び出しを、サーバが ID を保存している限り再試行することができます。べき等性を使用すると、リクエストのペイロードをレスポンスのペイロードと相關させ、重複したリクエストを排除し、失敗したリクエストやレスポンスが不明瞭なリクエストを再試行することができます。

REST API の POST 呼び出しで **Idempotency** を強制するには、**PayPal-Request-Id** リクエストヘッダーを使用します。このヘッダーには、サーバーが一定期間保存する一意のユーザー生成 ID が含まれています。

注意: すべての API がこのヘッダーをサポートしているわけではありません。お使いの API がこのヘッダーをサポートしているかどうか、およびサーバーが ID を保存する期間に関する情報については、お使いの API のリファレンスを参照してください。

たとえば、以前に指定した **PayPal-Request-Id** ヘッダーをリクエストに含めると、PayPal は、同じヘッダーを使用した以前のリクエストの最新のステータスを返します。逆に、リクエストから **PayPal-Request-Id** ヘッダーを省略すると、PayPal はリクエストを複製します。

注意: 同じ **PayPal-Request-Id** ヘッダーを持つ 2 つの API リクエストを同時に送信すると、PayPal は最初のリクエストを処理し、2 つ目のリクエストに失敗する可能性があります。

使用上の注意

- **PayPal-Request-Id** ヘッダー値は、各リクエストと API コールタイプの両方で一意でなければなりません。たとえば、**authorize payment** や **capture authorized payment** などです。

- PayPal は、元のリクエストのステータスではなく、現在の時点でのリクエストのステータスを提供します。

Rate Limit

PayPal は、加盟店をサポートするために、サイトの可用性とセキュリティに主眼を置いています。

レートリミットポリシーは公表していませんが、乱用と思われるトラフィックを確認した場合、一時的に API 使用制限を行うことがあります。API 使用制限は、その行為が PayPal、加盟店、またはお客様にとって問題がないと確信できるまで行います。

最大限の保護とサイトの安定性を確保するために、トラフィックの急増と減少を常に評価し、ポリシーを調整しています。お客様やお客様の顧客が HTTP 429 Unprocessable Entity - RATE_LIMIT_REACHED というステータスコードを受け取った場合、リクエストが多すぎて異常なトラフィックを示している可能性があるため、サイトの安定性を確保するために API 使用制限を行います。

<https://developer.paypal.com/api/rest/reference/rate-limiting/>

Sandbox では、1 分以内に同一 IP アドレスから 50 以上のリクエスト（すべてのエンドポイントの累積ヒット数）を受信すると、次の 5 分間その IP をブロックします。

IP アドレスが一時的にブロックされるのを防ぐには、1 分間のリクエスト数を 50 以下に制限してください。

<https://www.paypal.com/us/smarthelp/article/why-did-i-get-http-error-code-429-ts2115>

4.1 JavaScript SDK

JavaScript SDK は、あなたのページに PayPal がサポートしている支払い方法を表示し、バイヤーにパーソナライズされた合理的なチェックアウト体験を提供します。

JavaScript SDK スクリプトにクエリとスクリプトのパラメータを渡すことで、実装をカスタマイズできます。これらのパラメータは、PayPal が最適な決済方法とボタンを決定して、バイヤーに表示するのに役立ちます。

<https://developer.paypal.com/sdk/js/configuration/>

<https://developer.paypal.com/sdk/js/reference/>

<https://developer.paypal.com/sdk/js/performance/>

4.2 PayPal 決済

概要

1. 以下のサンプルコードをコピーして、チェックアウトページに貼り付けてください。
2. コード中の YOUR_SB_CLIENT_ID を、使用する Client ID に置き換えてください。
3. 必要に応じて JavaScript SDK に渡すパラメータを明示的に指定して、デフォルトの値を上書きしてください。各パラメータのデフォルト値については以下をご参照ください。
<https://developer.paypal.com/docs/business/javascript-sdk/javascript-sdk-configuration/>
4. 決済情報を送信するためのサーバ側コードを実装してください。createOrder と onApprove のメソッド内からサーバ側へリクエストしてください。サーバ側の実装は以降の説明をご参照ください。

4.2.1 Access Token を作成する

<https://developer.paypal.com/api/rest/#link-getanaccesstoken>

サンプルリクエスト

```
curl -v https://api-m.sandbox.paypal.com/v1/oauth2/token \
-H "Accept: application/json" \
-H "Accept-Language: en_US" \
-u "<client_id>:<secret>" \
-d "grant_type=client_credentials"
```

サンプルレスポンス

```
{
  "access_token": "A21AAGk9mYWDNvy-OzYmU985kgwscf16by3R_97_RF-
t90y_0TFTFQlhSndxdUcpfuY9dnMOpbwuNkmOZrsfnImVSIPIOGraQ",
  "app_id": "APP-804485P543195W28T",
  "expires_in": 32400,
  "nonce": "2019-12-04T09:29:10Z01fPaJg9ztZoKhcsglG1cjIWPxTy_HCIHPIDtPKvfDs",
  "scope": "https://uri.paypal.com/services/invoicing https://uri.paypal.com/services/disputes/read-buyer
https://uri.paypal.com/services/payments realtimepayment
https://uri.paypal.com/services/disputes/update-seller
https://uri.paypal.com/services/payments/payment/authcapture openid
https://uri.paypal.com/services/disputes/read-seller https://uri.paypal.com/services/payments/refund
https://api.paypal.com/v1/vault/credit-card https://api.paypal.com/v1/payments/*
https://uri.paypal.com/payments/payouts https://api.paypal.com/v1/vault/credit-card/*
https://uri.paypal.com/services/subscriptions https://uri.paypal.com/services/applications/webhooks",
  "token_type": "Bearer"
}
```

4.2.2 PayPal 決済ボタンを追加する

<https://developer.paypal.com/docs/checkout/standard/>

PayPal JavaScript SDK のデフォルトの intent は、指定しない場合 intent=capture であり、トランザクションのオーソリ(authorize)と売上(capture)の両方を直ちに行うことです。オーソリと売上を別々のアクションに分割するには、PayPal JavaScript SDK のスクリプトタグに &intent=authorize を追加します。

ボタンのカスタマイズは、下記を参照してください。レイアウト等については別途ご相談させていただきます。

<https://developer.paypal.com/docs/checkout/standard/customize/buttons-style-guide/>

PayPal 決済ボタン+クレジットカード決済ボタン サンプルフォーム

```
<!DOCTYPE html>
<html>
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1"> <!-- Ensures optimal rendering
on mobile devices. -->
  <meta http-equiv="X-UA-Compatible" content="IE=edge" /> <!-- Optimal Internet Explorer compatibility -
->
</head>

<body>
```

```

<script src="https://www.paypal.com/sdk/js?client-
id=<client_id>&components=buttons&currency=JPY&locale=ja_JP&intent=capture">
</script>

<div id="paypal-button-container"></div>

<!-- Add the checkout buttons, set up the order and approve the order --&gt;
&lt;script&gt;
paypal.Buttons({
  createOrder: function() {
    return fetch('/my-server/create-paypal-transaction', {
      method: 'post',
      headers: {
        'content-type': 'application/json'
      }
    }).then(function(res) {
      return res.json();
    }).then(function(data) {
      return data.orderID; // Use the same key name for order ID on the client and server
    });
  },
  onApprove: function(data) {
    return fetch('/my-server/capture-paypal-transaction', {
      headers: {
        'content-type': 'application/json'
      },
      body: JSON.stringify({
        orderID: data.orderID
      })
    }).then(function(res) {
      return res.json();
    }).then(function(details) {
      alert('Transaction funds captured from ' + details.payer_given_name);
    })
  }
}).render('#paypal-button-container'); // Display payment options on your web page
&lt;/script&gt;
&lt;/body&gt;
&lt;/html&gt;
</pre>

```

4.2.3 決済を作成する

サンプル HTML フォームの /my-server/create-paypal-transaction で下記 /v2/checkout/orders を貴社サーバからコールし決済を作成します。

https://developer.paypal.com/api/orders/v2/#orders_create

Intent に CAPTURE または AUTHORIZE を設定します。JavaScript のクエリパラメータで指定した intent と一致する必要があります。intent パラメータを Capture にすると決済が確定されます。Authorize の場合には、与信のみを取得して決済の確定は行われません。

以下のパラメータを追加してください。PayPal セラープロテクションの対象となるには、お客様から提供された配送先住所を PayPal に渡す必要があります。ギフト配送など購入者とは異なる配送先に

商品を配送する場合は、`purchase_units.shipping` には購入者の住所ではなく実際の配送先の住所を指定してください。

- `purchase_units.shipping`: 貴社サイト上で指定した配送先住所
- `payment_source.payment_source.experience_context.shipping_preference: "SET_PROVIDED_ADDRESS"`
- `invoice_id`: 貴社で発番した一意の注文 ID

サンプルリクエスト

```
curl -v -X POST https://api-m.sandbox.paypal.com/v2/checkout/orders \
-H "Content-Type: application/json" \
-H "Authorization: Bearer <access token>" \
-d '{
  "intent": "CAPTURE",
  "payment_source": {
    "paypal": {
      "experience_context": {
        "payment_method_preference": "IMMEDIATE_PAYMENT_REQUIRED",
        "brand_name": "Your Site Name",
        "locale": "ja-JP",
        "shipping_preference": "SET_PROVIDED_ADDRESS",
        "user_action": "PAY_NOW",
        "return_url": "https://example.com/returnUrl",
        "cancel_url": "https://example.com/cancelUrl"
      }
    }
  },
  "purchase_units": [
    {
      "amount": {
        "currency_code": "JPY",
        "value": "1000",
        "breakdown": {
          "item_total": {
            "currency_code": "JPY",
            "value": "1000"
          }
        }
      },
      "items": [
        {
          "name": "Sample Goods",
          "unit_amount": {
            "currency_code": "JPY",
            "value": "1000"
          },
          "quantity": "1",
          "category": "PHYSICAL_GOODS"
        }
      ],
      "shipping": {
        "name": {
          "full_name": "太郎 青山"
        },
        "address": {
          "line1": "123 Main St",
          "line2": "Apt 456",
          "city": "Anytown",
          "state": "CA",
          "zip": "90210",
          "country_code": "US"
        }
      }
    }
  ]
}'
```

```

        "address_line_1": "青山 1-2-3",
        "address_line_2": "ペイパルビル 1001",
        "admin_area_2": "港区",
        "admin_area_1": "東京都",
        "postal_code": "123-4567",
        "country_code": "JP"
    }
},
"invoice_id": "your-own-uniqueid-for-an-order",
"custom_id": "貴社指定のカスタムパラメータ値"
}
]
}'

```

サンプルレスポンス

```

{
  "id": "3W9401969M757672V",
  "status": "PAYER_ACTION_REQUIRED",
  "payment_source": {
    "paypal": {}
  },
  "links": [
    {
      "href": "https://api.sandbox.paypal.com/v2/checkout/orders/3W9401969M757672V",
      "rel": "self",
      "method": "GET"
    },
    {
      "href": "https://www.sandbox.paypal.com/checkoutnow?token=3W9401969M757672V",
      "rel": "payer-action",
      "method": "GET"
    }
  ]
}

```

この決済作成処理後に PayPal のログイン/新規アカウント作成画面が表示され、新規ユーザーはクレジットカードまたは銀行口座の登録～支払いの同意を行います。

※ 即時決済の場合は以下の「決済を実行する」を行う

4.2.4 決済を実行する

サンプル HTML フォームの /my-server/capture-paypal-transaction で下記
 /v2/checkout/orders/{id}/capture を貴社サーバからコールし決済を実行します。

https://developer.paypal.com/api/orders/v2/#orders_capture

以下のヘッダを追加してください。

"PayPal-Request-Id: <Unique-Session-Identifier>"

サンプルリクエスト

```
curl -v -X POST https://api-m.sandbox.paypal.com/v2/checkout/orders/3W9401969M757672V/capture \
```

```
-H "Content-Type: application/json" \
-H "Authorization: Bearer Access-Token" \
-H "PayPal-Request-Id: 7b92603e-77ed-4896-8e78-5dea2050476a"
```

サンプルレスポンス

```
{
  "id": "3W9401969M757672V",
  "status": "COMPLETED",
  "payment_source": {
    "paypal": {
      "email_address": "buyer@example.com",
      "account_id": "ZJEHNEECZ6DKS",
      "name": {
        "given_name": "太郎",
        "surname": "青山"
      },
      "address": {
        "country_code": "JP"
      }
    }
  },
  "purchase_units": [
    {
      "reference_id": "default",
      "shipping": {
        "name": {
          "full_name": "太郎 青山"
        },
        "address": {
          "address_line_1": "青山 1-2-3",
          "address_line_2": "ペイパルビル 1001",
          "admin_area_2": "港区",
          "admin_area_1": "東京都",
          "postal_code": "123-4567",
          "country_code": "JP"
        }
      }
    },
    "payments": {
      "captures": [
        {
          "id": "OBW08839JT755205T",
          "status": "COMPLETED",
          "amount": {
            "currency_code": "JPY",
            "value": "1000"
          },
          "final_capture": true,
          "seller_protection": {
            "status": "ELIGIBLE",
            "dispute_categories": [
              "ITEM_NOT RECEIVED",
              "UNAUTHORIZED TRANSACTION"
            ]
          },
          "seller_receivable_breakdown": {
            "item": {
              "label": "Item A"
            },
            "tax": {
              "label": "Tax"
            }
          }
        }
      ]
    }
  ]
}
```

```

    "gross_amount": {
      "currency_code": "JPY",
      "value": "1000"
    },
    "paypal_fee": {
      "currency_code": "JPY",
      "value": "76"
    },
    "net_amount": {
      "currency_code": "JPY",
      "value": "924"
    }
  },
  "invoice_id": "your-own-uniqueid-for-an-order",
  "links": [
    {
      "href": "https://api.sandbox.paypal.com/v2/payments/captures/0BW08839JT755205T",
      "rel": "self",
      "method": "GET"
    },
    {
      "href": "https://api.sandbox.paypal.com/v2/payments/captures/0BW08839JT755205T/refund",
      "rel": "refund",
      "method": "POST"
    },
    {
      "href": "https://api.sandbox.paypal.com/v2/checkout/orders/3W9401969M757672V",
      "rel": "up",
      "method": "GET"
    }
  ],
  "create_time": "2021-01-21T07:21:27Z",
  "update_time": "2021-01-21T07:21:27Z"
}
],
{
  "payer": {
    "name": {
      "given_name": "太郎",
      "surname": "青山"
    },
    "email_address": "buyer@example.com",
    "payer_id": "ZJEHNEECZ6DKS",
    "address": {
      "country_code": "JP"
    }
  },
  "links": [
    {
      "href": "https://api.sandbox.paypal.com/v2/checkout/orders/3W9401969M757672V",
      "rel": "self",
      "method": "GET"
    }
  ]
}

```

```
]  
}
```

※ オーソリを使用する場合は以下の「オーソリを実行する」「決済を実行する」を行う

4.2.5 オーソリを実行する

サンプル HTML フォームの /my-server/authorize-paypal-transaction で下記
/v2/checkout/orders/{id}/authorize を貴社サーバからコールし決済を実行します。

https://developer.paypal.com/api/orders/v2/#orders_authorize

以下のヘッダを追加してください。

"PayPal-Request-Id: <Unique-Session-Identifier>"

サンプルリクエスト

```
curl -v -X POST https://api-m.sandbox.paypal.com/v2/checkout/orders/3W9401969M757672V/authorize \  
-H "Content-Type: application/json" \  
-H "Authorization: Bearer Access-Token" \  
-H "PayPal-Request-Id: 7b92603e-77ed-4896-8e78-5dea2050476a"
```

サンプルレスポンス

```
{  
    "id": "3W9401969M757672V",  
    "status": "COMPLETED",  
    "payment_source": {  
        "paypal": {  
            "email_address": "buyer@example.com",  
            "account_id": "ZJEHNEECZ6DKS",  
            "name": {  
                "given_name": "太郎",  
                "surname": "青山"  
            },  
            "address": {  
                "country_code": "JP"  
            }  
        }  
    },  
    "purchase_units": [  
        {  
            "reference_id": "default",  
            "shipping": {  
                "name": {  
                    "full_name": "太郎 青山"  
                },  
                "address": {  
                    "address_line_1": "青山 1-2-3",  
                    "address_line_2": "ペイパルビル 1001",  
                    "admin_area_2": "港区",  
                    "admin_area_1": "東京都",  
                    "postal_code": "123-4567",  
                    "country_code": "JP"  
                }  
            }  
        },  
    ]  
}
```

```

"payments": [
    "authorizations": [
        {
            "status": "CREATED",
            "id": "1T3898K57614G8S79",
            "amount": {
                "currency_code": "JPY",
                "value": "1000"
            },
            "invoice_id": "your-own-uniqueid-for-an-order",
            "seller_protection": {
                "status": "ELIGIBLE",
                "dispute_categories": [
                    "ITEM_NOT RECEIVED",
                    "UNAUTHORIZED TRANSACTION"
                ]
            },
            "expiration_time": "2021-02-20T05:38:07Z",
            "links": [
                {
                    "href": "https://api.sandbox.paypal.com/v2/payments/authorizations/1T3898K57614G8S79",
                    "rel": "self",
                    "method": "GET"
                },
                {
                    "href": "https://api.sandbox.paypal.com/v2/payments/authorizations/1T3898K57614G8S79/capture",
                    "rel": "capture",
                    "method": "POST"
                },
                {
                    "href": "https://api.sandbox.paypal.com/v2/payments/authorizations/1T3898K57614G8S79/void",
                    "rel": "void",
                    "method": "POST"
                },
                {
                    "href": "https://api.sandbox.paypal.com/v2/payments/authorizations/1T3898K57614G8S79/reauthorize",
                    "rel": "reauthorize",
                    "method": "POST"
                },
                {
                    "href": "https://api.sandbox.paypal.com/v2/checkout/orders/32G78336EM1179410",
                    "rel": "up",
                    "method": "GET"
                }
            ],
            "create_time": "2021-01-22T05:38:07Z",
            "update_time": "2021-01-22T05:38:07Z"
        }
    ]
},
"payer": {

```

```

{
  "name": {
    "given_name": "太郎",
    "surname": "青山"
  },
  "email_address": "buyer@example.com",
  "payer_id": "ZJEHNEECZ6DKS",
  "address": {
    "country_code": "JP"
  },
  "links": [
    {
      "href": "https://api.sandbox.paypal.com/v2/checkout/orders/3W9401969M757672V",
      "rel": "self",
      "method": "GET"
    }
  ]
}

```

※ オーソリを使用する場合は上の「オーソリを実行する」、下の「決済を実行する」を行う

4.2.6 決済を実行する

下記 /v2/payments/authorizations/{authorization_id}/capture を貴社サーバからコールし決済を実行します。

https://developer.paypal.com/api/payments/v2/#authorizations_capture

以下のヘッダを追加してください。

"PayPal-Request-Id: <Unique-Session-Identifier>"

また、以下のように金額を指定してオーソリ金額の一部金額を売上処理することが可能です。後で残りも売上処理する場合は final_capture に false を指定します。

```
{
  "amount": {
    "value": "700",
    "currency_code": "JPY"
  },
  "invoice_id": "your-own-uniqueid-for-an-order",
  "final_capture": true
}
```

サンプルリクエスト

```
curl -v -X POST 'https://api-sandbox.paypal.com/v2/payments/authorizations/1T3898K57614G8S79/capture' \
-H 'Authorization: Bearer <Access-Token>' \
-H 'Accept: application/json' \
-H 'Content-Type: application/json' \
-H 'PayPal-Request-Id: 1cc9e87b-0410-4999-9fef-76d732a94bc9'
```

サンプルレスポンス

```
{
```

```
"id": "0VG7SW2984469872W",
"status": "COMPLETED",
"links": [
  {
    "href": "https://api.sandbox.paypal.com/v2/payments/captures/0VG7SW2984469872W",
    "rel": "self",
    "method": "GET"
  },
  {
    "href": "https://api.sandbox.paypal.com/v2/payments/captures/0VG7SW2984469872W/refund",
    "rel": "refund",
    "method": "POST"
  },
  {
    "href": "https://api.sandbox.paypal.com/v2/payments/authorizations/1T3898K57614G8S79",
    "rel": "up",
    "method": "GET"
  }
]
```

4.3 キャンセル・返金

4.3.1 決済のキャンセル・返金

下記 /v2/payments/captures/{capture_id}/refund を貴社サーバからコールし決済のキャンセル、返金を実行します。

https://developer.paypal.com/api/payments/v2/#captures_refund

PayPal アカウントの残高から返金されますので、実行するには十分な残高が必要です。返金可能な期間は取引日から 180 日間です。

金額を指定し一部金額を返金することができます。

4.3.2 オーソリのキャンセル・返金

下記 /v2/payments/authorizations/{authorization_id}/void を貴社サーバからコールしオーソリのキャンセルを実行します。

https://developer.paypal.com/api/payments/v2/#authorizations_void

4.4 処理結果

4.4.1 HTTP ステータスコード

<https://developer.paypal.com/api/rest/responses/>

PayPal API コールは、HTTP ステータスコードを返します。

リクエストが成功した場合、PayPal は HTTP 2XX のステータスコードを返します。

リクエストが失敗した場合、PayPal は HTTP 4XX または 5XX のステータスコードを返します。

200 系: 正常終了
400 系: 決済失敗
500 系: 内部エラーのためリトライ (※タイムアウトの場合もリトライ)

注意:

HTTP 5xx またはネットワークタイムアウトの場合でも、PayPal トランザクションが作成された可能性があります。トランザクションの結果を受けるには、以前と同じ PayPal-Request-Id HTTP ヘッダーを使用して、同じリクエストコールを少なくとも一回は繰り返すことをお勧めします。これを参照してください。

API idempotency
<https://developer.paypal.com/api/rest/reference/idempotency/>

4.4.2 取引ステータス

Capture の結果は、purchase_units.paymentscaptures.status を確認してください。

COMPLETED - 売上実行済み
DECLINED - 売上却下
PARTIALLY_REFUNDED - 売上金額の一部金額が返金済み
PENDING - 売上の保留。詳細については、status.details を参照してください。
REFUNDED - 返金済み

Authorization の結果は、purchase_units.payments.authorizations.status を確認してください。

CREATED - オーソリ実行済み。売上は未実行。
CAPTURED - オーソリに対する売上が実行済み
DENIED - オーソリ却下
EXPIRED - オーソリ失効
PARTIALLY_CAPTURED - オーソリに対し、一部金額の売上が実行済み
VOIDED - オーソリ取り消し済み
PENDING - 保留。主なケースはリスク判断により一旦保留にするケースです。詳細については、status.details を参照してください。

なお、レスポンスの status の値は以下のとおりです。

CREATED - 注文が作成済み
SAVED - (今回は使用しません)
APPROVED - ユーザが PayPal 決済画面で支払いを承認済み
VOIDED - 取り消し済み
COMPLETED - オーソリまたは売上が実行済

4.4.3 Paypal-Debug-Id

レスポンスヘッダの Paypal-Debug-Id は必ず保存してください。API コールに問題などが発生した場合の調査に必要です。

4.5 エラー処理

4.5.1 擬似的にエラーのレスポンスを返す方法

エラーをシミュレートするには、Orders v2 および Payments v2 REST API 呼び出しで PayPal-Mock-Response リクエストヘッダーを使用します。

```
-H "PayPal-Mock-Response: {"mock_application_codes" : "INSTRUMENT_DECLINED"}"
```

<https://developer.paypal.com/tools/sandbox/negative-testing/request-headers/>

HTTP ステータスコード

<https://developer.paypal.com/api/rest/responses/>

Orders v2 API エラーメッセージ

<https://developer.paypal.com/api/rest/reference/orders/v2/errors/>

Payments v2 API エラーメッセージ

<https://developer.paypal.com/api/payments/v2/#errors>

5 ドキュメント

PayPal デベロッパーサイト

<https://developer.paypal.com/home/>

Sandbox (テスト環境)

<https://developer.paypal.com/tools/sandbox/>

※ Developer account は日本ではご利用になられません。通常のビジネスアカウントを作成ください。

概要

<https://developer.paypal.com/api/rest/>

Orders API

<https://developer.paypal.com/api/orders/v2/>

Payments API

<https://developer.paypal.com/api/payments/v2/>

API レスポンス

<https://developer.paypal.com/api/rest/responses/>