Merchant Onboarding Guide

PayPal Here

This document is confidential and its distribution is restricted.

DISTRIBUTE THIS DOCUMENT ONLY UNDER PROPER NDA.

PayPal

Document Revision History

Pub. Date	Description of Changes
01/09/15	Added `logourl` field and logo guidelines.
03/03/14	Initial document release.

Table of Contents

Overview

PayPal Here lets you integrate PayPal payment processing with your merchant applications and Point of Sale systems.

A PayPal Here integration gives merchants the means to accept card *swipe* and *key entry* for debit and credit cards. Integrations also have the ability to accept PayPal payments as a payment option. By integrating PayPal Here into your applications, you give merchants a way to take advantage of the low credit-card processing rates offered by PayPal, and you open the door for them to benefit from our associated PayPal services.

PayPal Here integrations at POS applications can interact with mobile apps running the native *PayPal Here SDK* (available for iOS and Android), providing a way for you to connect mobile users to your merchant sales systems.

This guide is intended for developers who want to provide a PayPal Here onboarding flow to the merchants who use their systems. In this guide, you'll find details on how to provide a PayPal onboarding flow in your *back office* and give merchants a painless way to enable PayPal Here in your apps.

> **Note:** A merchant must have a PayPal Here-enabled account before they can accept credit card payments.

The PayPal Here onboarding flow

The *PayPal onboarding flow* provides a way for merchants to create a PayPal Business account (if they do not already have one), upgrade to a PayPal Business account (if they have a Personal or Premier account), and apply for PayPal Here approval.

The PayPal Here onboarding flow takes the merchant through the following steps:

1. The merchant is presented with a Welcome page. The Welcome page prompts them to login with an existing PayPal account or to create a new PayPal account.

   - If the merchant doesn't already have a PayPal account, the onboarding flow guides them through the steps to create a new PayPal Business account. Once they have an account, they can proceed to the next step.

   - If the merchant logs in with their PayPal account then they are upgraded to a Business account, if required and are then guided through the rest of the PayPal Here onboarding flow.

At the end Step 1, the merchant is assigned one of three states:

- `approved`
- `manual review`
- `denied`

Before a merchant is allowed to transact using PayPal Here, their account much reach a status of `approved`.

2. When the status of a merchant returns as approved, the merchant must then grant permission to the developer so the developer's application can make PayPal transaction requests on behalf of the merchant.

The partner back-office system

Partners likely have a *back office*, either browser-based or as part of their iOS/Android application, that merchants can log into to manage settings, view a history of their transactions, run reports, and so on. On your back office pages, typically in a section dedicated to payment processing, provide a button that, when clicked, initiates the PayPal Here onboarding flow for the merchant.

Once a merchant has been approved for PayPal Here, your back office should initiate the process to gain third-party permissions from the merchant so you can process PayPal requests on the merchant's behalf (the PayPal requests made through your system will act on behalf of the merchant's PayPal account).

Initiating the PayPal onboarding flow

When a merchant logs in to your back office site, display a button that initiates the PayPal Here onboarding flow. When a merchant initiates the onboarding flow, they are redirected to a PayPal web page that's displayed in a separate browser or in an embedded view. The PayPal flow leads the merchant through the steps to configure their Business account for PayPal Here processing.

The button you create must submit a form post to PayPal. When PayPal receives the submitted form, it begins the onboarding flow. We recommend you pre-fill any merchant information you might have stored in your account record for the merchant. This can reduce the data that needs to be entered by the merchant, which reduces friction for the merchant during the PayPal onboarding flow.

Form submit

PayPal supports the following parameters when you use a form to submit the request:

Parameter	Validation	Description
partnertype	SDK (Required)	Must be specified as "SDK"
partner_id	32 characters	An identifier of the partner that built the code for the button that the merchant clicked. A BN code should be obtained from your PayPal account manager.
partner_name	60 characters	Partner name to be used for co-branding and reference purposes in the PayPal onboarding flow.
returnurl	1024 characters (Required)	URL encoded value to page where you want the merchant returned to after the onboarding flow completes.
swiper	y or n	Must be "y" or "n": y = Gives the merchant the option to order a PayPal Here triangle swiper as part of the onboarding flow. n = Suppresses the option to order the PayPal Here swiper. Use this option if you issue your own swipers OR for merchants who do not intend to use the PayPal swiper. (Default)
first_name	32 characters	Merchant's first name
last_name	32 characters	Merchant's last name
email	127 characters	Merchant's email address
business_name	60 characters	Merchant's business name
business_phone	10 digits	Merchant's business phone number
business_address1	100 characters	Merchant's business address line 1
business_address2	100 characters	Merchant's business address line 2
business_city	40 characters	Merchant's business address city
business_state	2 characters	Merchant's business address state
business_zip	5 digits	Merchant's business address zip
logourl		An HTTPS URL pointing to the logo of your location.
home_address1	100 characters	Merchant's home address line 1
home_address2	100 characters	Merchant's home address line 2
home_city	40 characters	Merchant's home address city
home_state	2 characters	Merchant's home address state
home_zip	5 digits	Merchant's home address zip code
mobile_phone	10 digits	Merchant's mobile phone number

For example, you could submit the following form to PayPal:

```
<form method="post" name="pphsignup_form" action="https://www.paypal.com/signup-pph">
  <input type="hidden" name="partnertype" value="sdk">
  <input type="hidden" name="partner_id" value="{My-Bn-Code-Id}">
  <input type="hidden" name="partner_name" value="{SDK-Partner-Id}">
  <input type="hidden" name="swiper" value="n">
  <input type="hidden" name="first_name" value="Ryan">
  <input type="hidden" name="last_name" value="Clyde">
  <input type="hidden" name="email" value="mmerchant@example.com">
  <input type="hidden" name="business_name" value="The T-Shirt Zone">
  <input type="hidden" name="business_phone" value="4089671000">
  <input type="hidden" name="business_address1" value="123 Easy Street">
  <input type="hidden" name="business_address2" value="">
  <input type="hidden" name="business_city" value="San Jose">
  <input type="hidden" name="business_state" value="CA">
  <input type="hidden" name="business_zip" value="95131">
  <input type="submit" name="logourl" value="{Logo-Url}">
  <input type="submit" name="submit" value="Sign Up PPH">
</form>
```

About the submittal form

Your system stores certain merchant information, and you can use parts of this information to pre-populate the form. PayPal recommends you prefill as many merchant information fields as possible, including first name, last name, email, business name, business address and phone. Note that:

- You system can supply any, all, or none of the merchant's information in the submitted form.
- All of the supplied merchant information can be edited by the merchant during the PayPal Here onboarding flow.
- Any invalid information supplied by your system is silently dropped, and no errors are generated by the onboarding flow.
- Information associated with an existing PayPal account takes precedence over information submitted by your system.
- Do not populate the `home_address1` and `home_address2` fields with the any personal address information of the account owner (for example, the residential address of the account owner), nor their personal mobile phone. PayPal uses these fields for customer identity verification, and misinformation here can result in decline.

Submitting your logo

When you submit a business logo, you must follow the *PayPal Cobranding Guidelines* and consider how it will appear next to the PayPal logo. Also, please follow these logo guidelines:

Logo Property	Description
Height	66 pixels max height
Width	310 pixels max width
Background	Transparent background (the artwork is placed on a background of #F5F5F5).
Resolution	72 DPI minimum
File Format	PNG with transparency. If you are using JPG or GIF, use a #F5F5F5 background
Color Format	RGB
Policies	Only submit artwork you can legally share. The artwork must represent your company logo. Do not include additional imagery, background colors, or words. Artwork is displayed next to the PayPal logo and both logos should have equal prominence. **Note:** To promote your brand in the best light possible, PayPal will not use artwork that does not meet our guidelines or does not meet our presentation standards.

Use the following image as a guideline for your logo sizing and image placement:
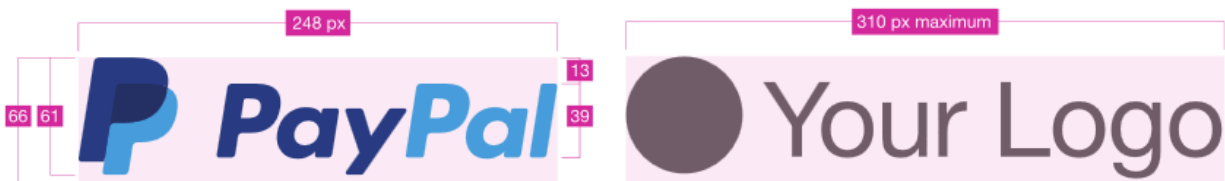
If you have questions on these guidelines, please contact your PayPal account representative.

Returning from PayPal

After completing the onboarding flow, the merchant is returned to the URL that you specified in your original redirect (the `returnurl` parameter). In its redirect back to your site, PayPal includes a code that details the status of the merchant's PayPal Here account setting.

The status is a two-level status code. The first level indicates `success`, `pending`, or `failure` and the second-level is sub code that contains a more detailed description of the status.

| Return Status	Subcode
`success`	`approved`
`pending`	`manualreview`
`failure`	Either `declined`, or `error`.

For example:

```
https://www.example.com/paypal?status=success&subcode=approved
```

What the merchant sees

PayPal presents a different onboarding flows depending on whether or not the merchant has a PayPal account.

If the merchant has an existing PayPal Business account, the onboarding flow presents questions that help PayPal verify the identity of the merchant. After this verification step, the onboarding flow is complete. If the merchant has a Personal PayPal account, they are taken through a flow that upgrades their account to a Business account.
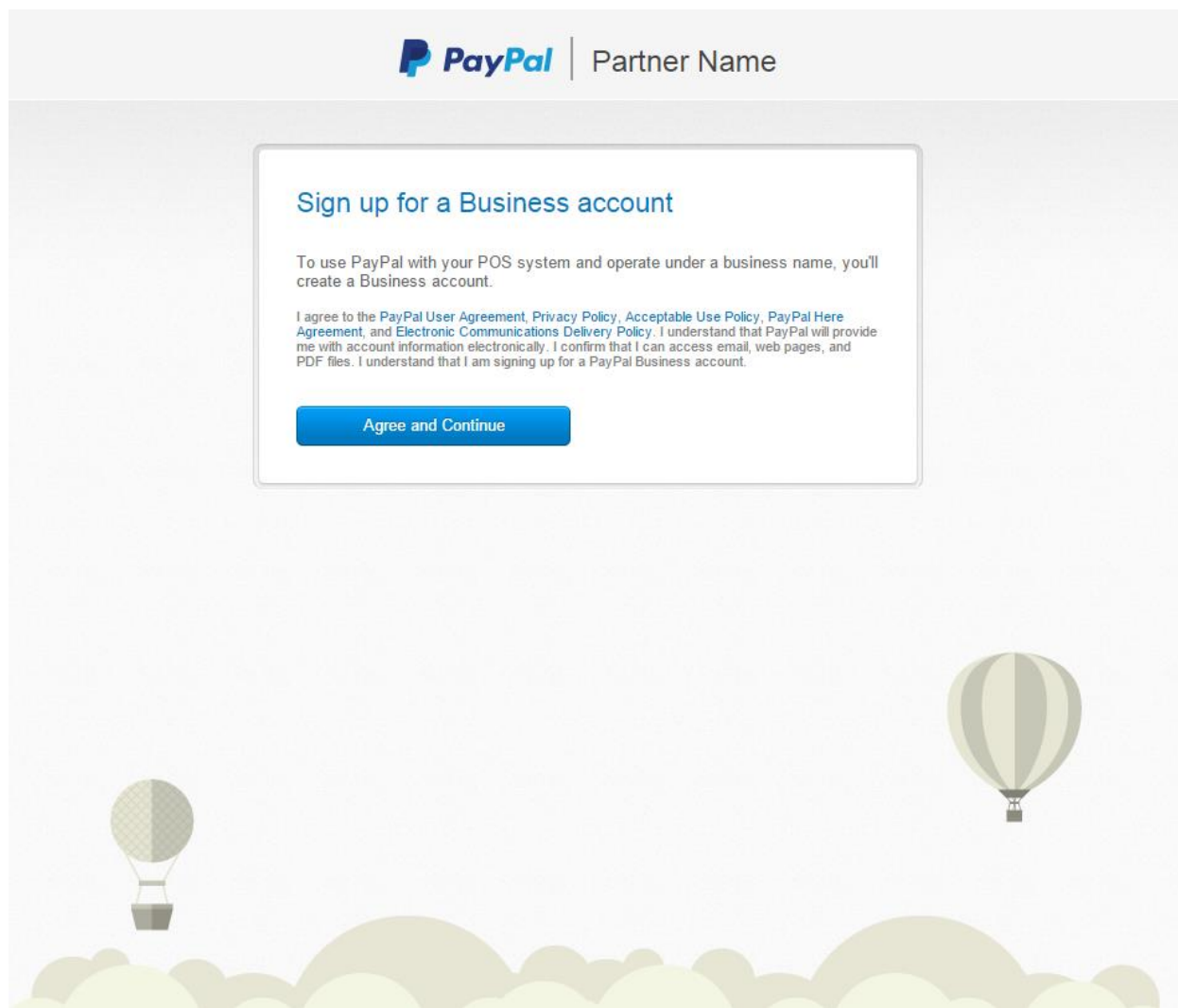
If the merchant does not have any PayPal account, they are taken through the full onboarding flow which begins with the following **Welcome to PayPal** page:
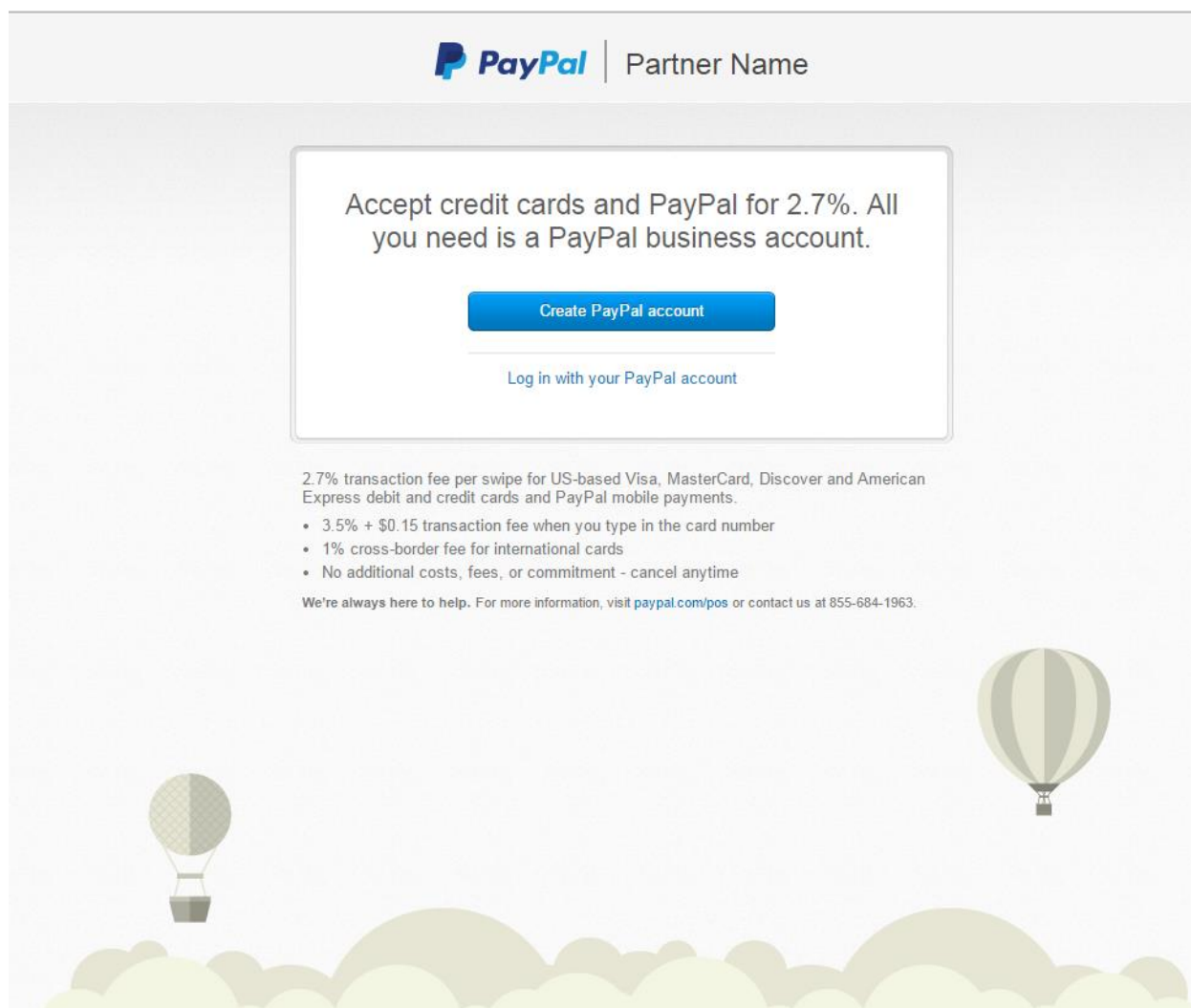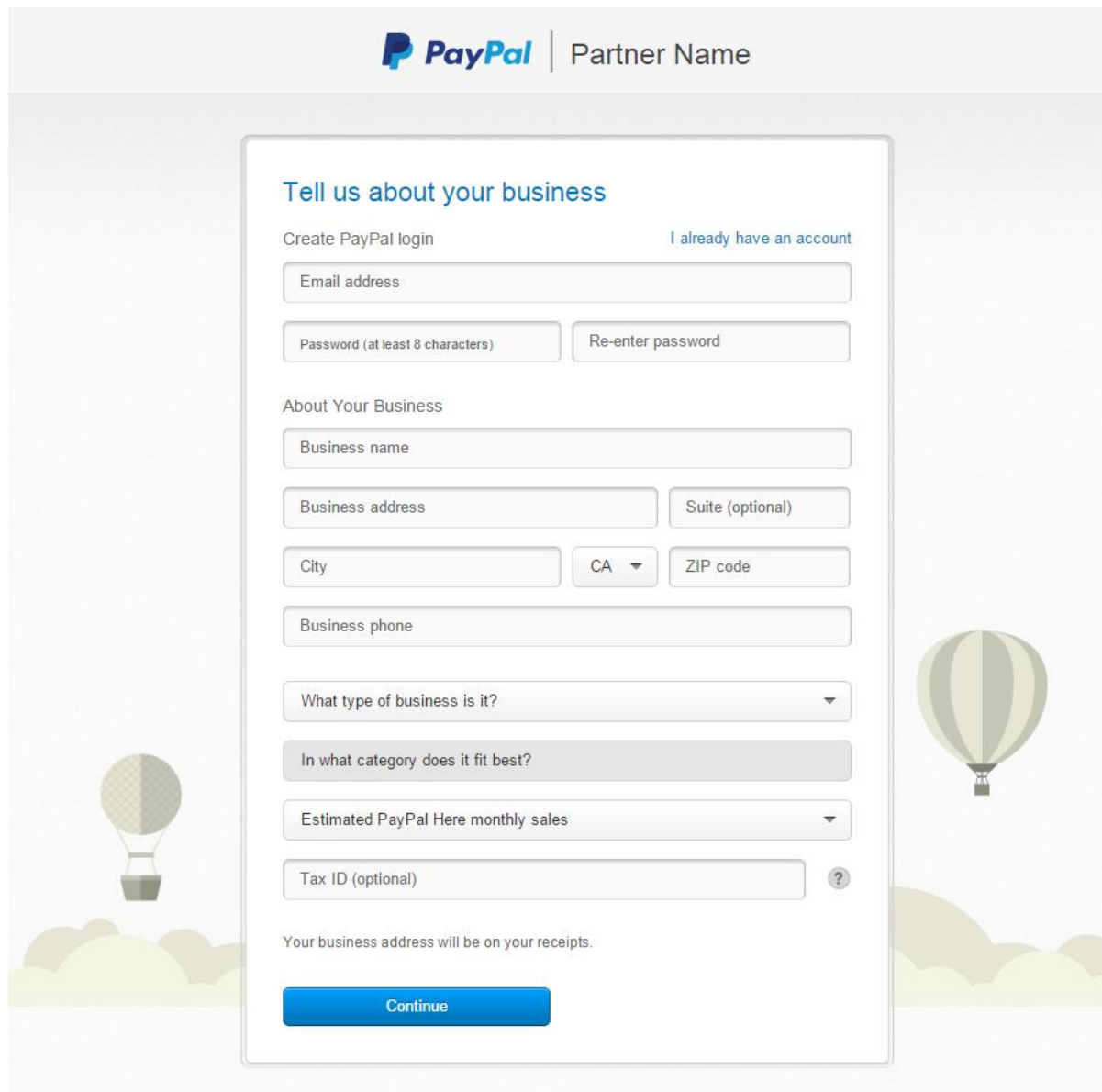
THE ONBOARDING WELCOME PAGE

If the merchant has an existing PayPal account, they can use that to log in to PayPal and continue the PayPal Here onboarding flow from there.

If the merchant does *not* have a PayPal Business account, they need to create one. The follow screen displays the **Sign up for a Business account** page, which is displayed if the merchant chooses to create a new PayPal account or if they enter an email address that is not associated with a PayPal account:

PayPal | Partner Name

Sign up for a Business account

To use PayPal with your POS system and operate under a business name, you'll create a Business account.

I agree to the PayPal User Agreement, Privacy Policy, Acceptable Use Policy, PayPal Here Agreement, and Electronic Communications Delivery Policy. I understand that PayPal will provide me with account information electronically. I confirm that I can access email, web pages, and PDF files. I understand that I am signing up for a PayPal Business account.

Agree and Continue

PAGE DISPLAYED IF THE EMAIL ADDRESS ENTERED IS NOT ASSOCIATED WITH A PAYPAL ACCOUNT

When a merchant logs in to the PayPal Here onboarding flow, they are led through a custom set of pages that includes the following two information-gathering pages:

THE FIRST INFORMATION-GATHERING PAGE.

Caution: To streamline the onboarding process, PayPal recommends you pre-fill the **Tell us about your business** form with as much merchant business information as you can.

Alternately, PayPal stresses that you *avoid pre-filling personal information* (especially mobile phone and home address information) on the **Tell us about yourself** page because of the negative consequences of any misinformation. Instead, let the merchant enter that information as they follow the onboarding process.

THE SECOND INFORMATION-GATHERING PAGE.

After clicking **Continue** on the second information gathering page, they are presented with a page that asks several questions aimed at verifying their identity. After this, PayPal assigns the merchant a `status` and a status `subcode`.

If the merchant status `subcode` equals `approved`, PayPal displays the following page and redirects the merchant back to your site when they click **Continue**:
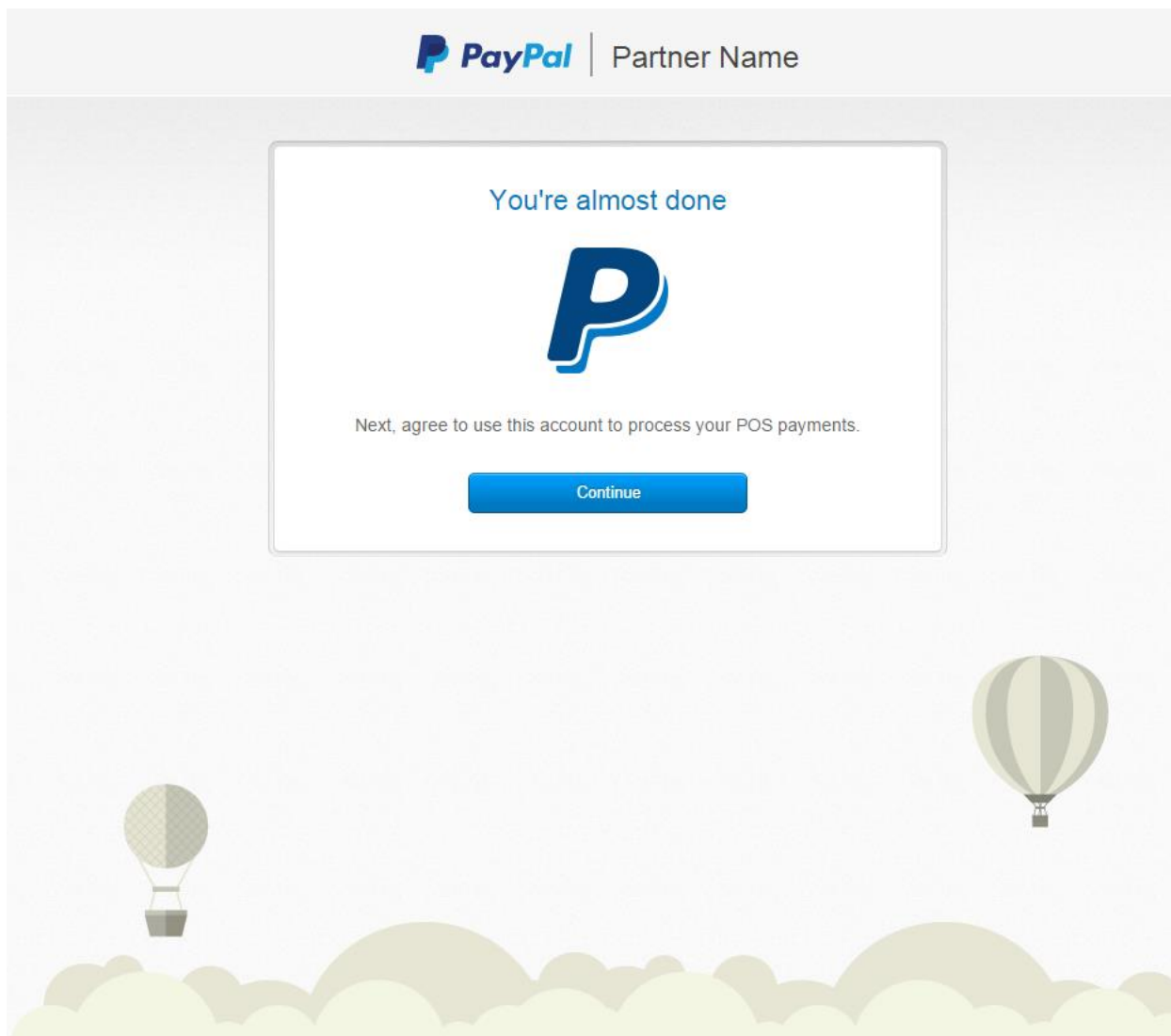
PAYPAL DISPLAYS THIS PAGE WHEN A MERCHANT IS FULLY APPROVED TO USE PAYPAL HERE

If the merchant is entered into a *manual review* process, PayPal displays the following screen:
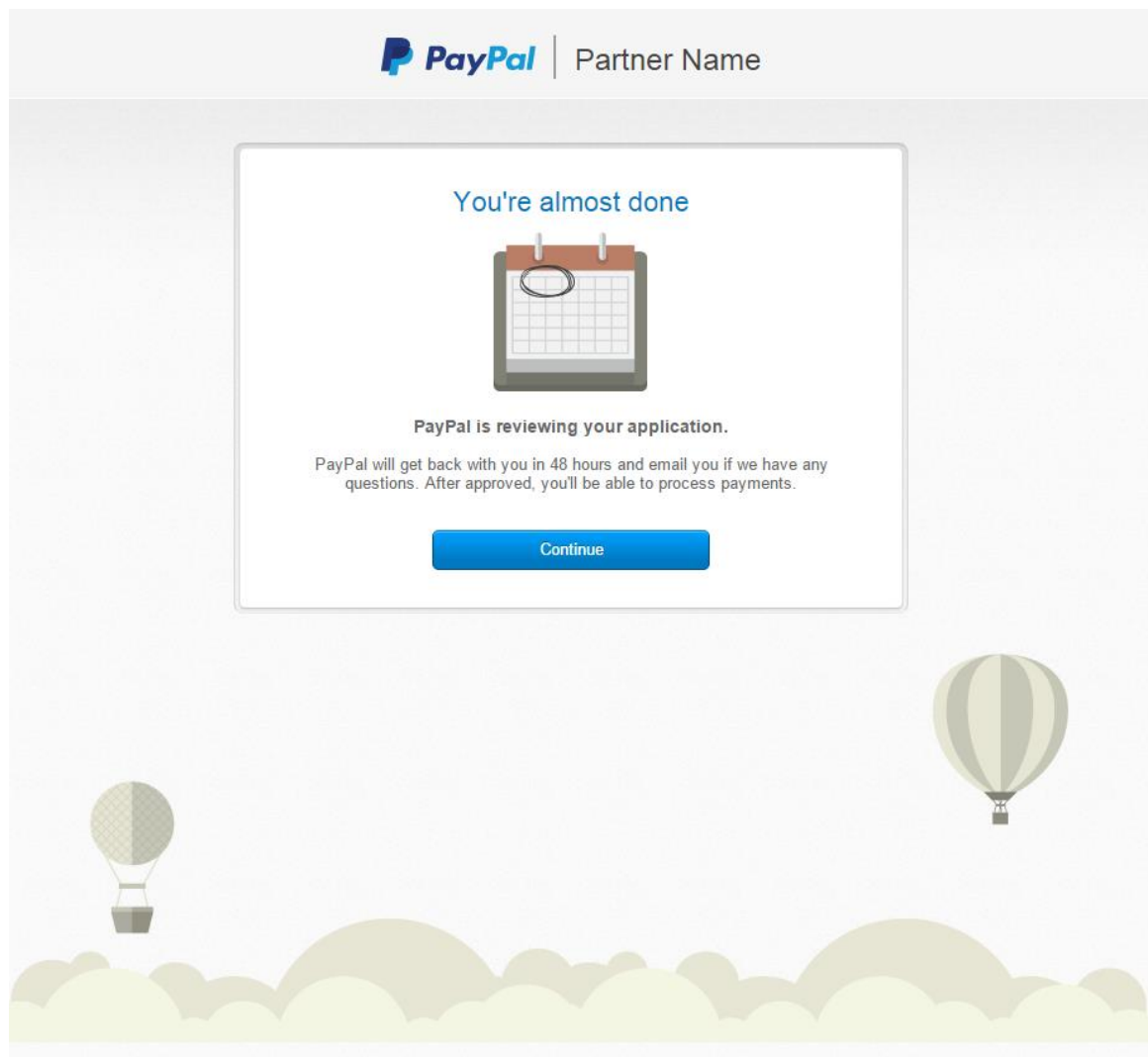
And last, the following screen displays if the merchant is *declined*:
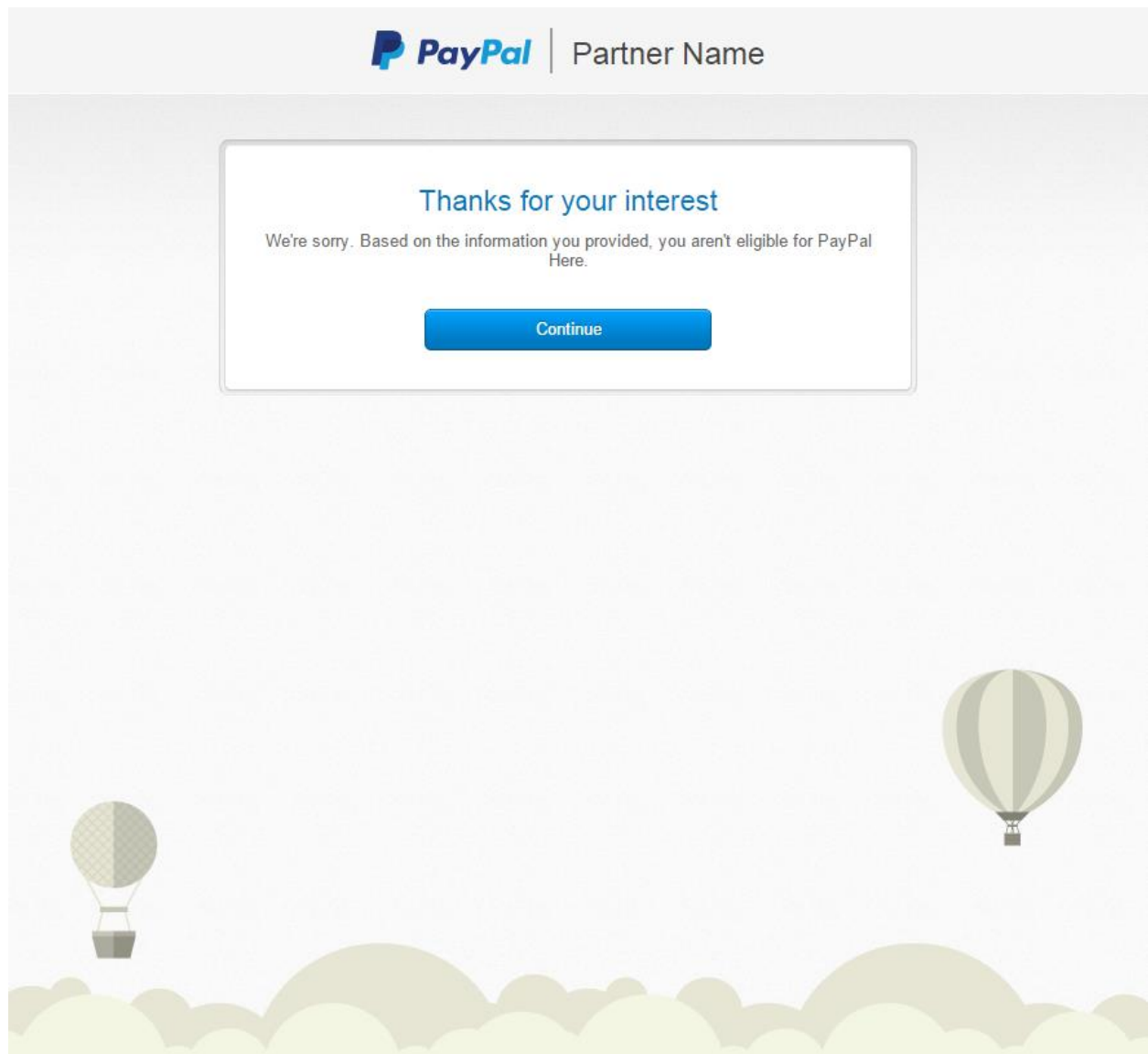
Handling the returning merchant

When the merchant completes the PayPal Here onboarding flow, PayPal returns them to your site with the status of their PayPal Here application.

After the merchant has been approved to use PayPal Here, you must gain permission from the merchant before you can begin processing PayPal Here requests on their behalf from your apps.

To provide the best experience for the merchant, code your `returnurl` page so that it scans and records the account status returned by PayPal. If `subcode` equals `approved` or `manualreview`, it is best to instantly redirect the merchant to the PayPal URI that is used to establish your *third-party permissions*. The following section provides details on these permissions.

Obtaining third-party permissions

There are three entities that partake in a PayPal Here transaction:

- The *customer* who is purchasing the goods or services
- The *merchant* who is selling the goods or services
- *Your application*, which handles the interfaces to PayPal

In this setup, your application is handling transactions on behalf of the merchants (the transaction funds are going into the merchant's PayPal account), and the merchants must grant permission to you permission so your app can act on behalf of the merchant.

Merchant permissions are returned to you from the PayPal onboarding process in the form of an *authorization code*.

The permissions flow

To obtain the merchant permissions for your PayPal Here app, direct the merchant to the following PayPal URL (note this example uses the Sandbox version of the endpoint):
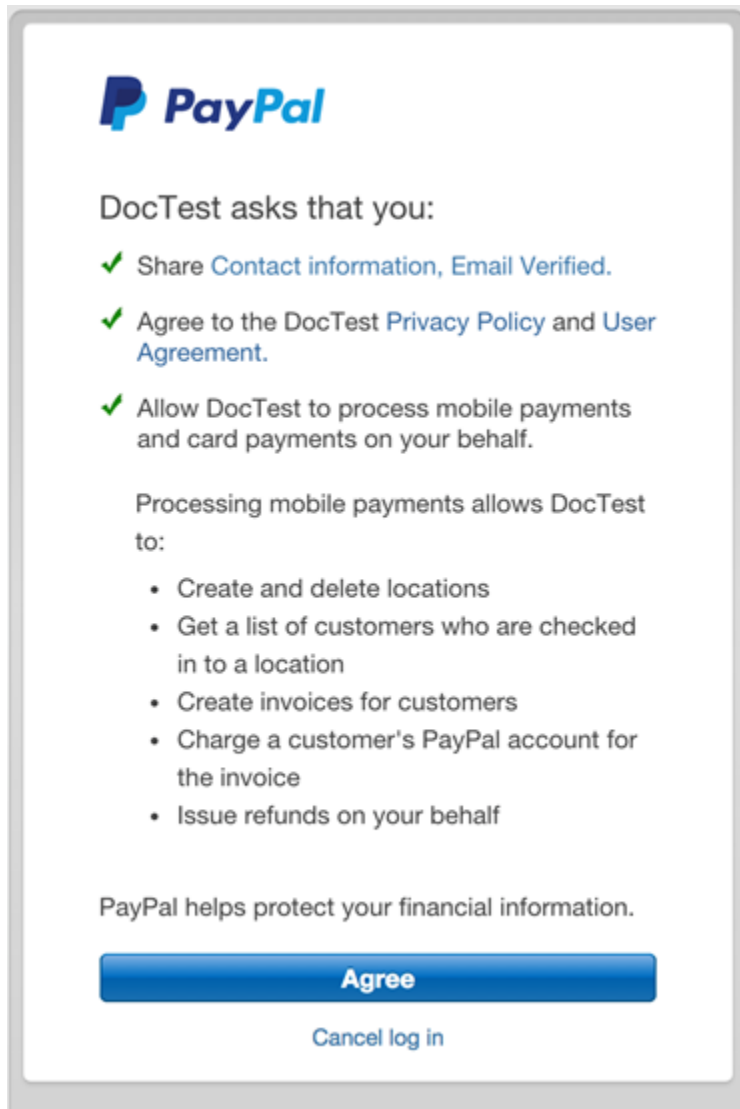
```
https://www.sandbox.paypal.com/webapps/auth/protocol/openidconnect/v1/authorize?
  scope=https://uri.paypal.com/services/paypalhere
  &response_type=code
  &redirect_uri={redirectUri}
  &client_id={clientId}
```

Supply all the following string parameters in the URL:

- **Scope** – For managing payments and using the PayPal Here functionality, use "https://uri.paypal.com/services/paypalhere".For additional permissions, please refer to the Log In with PayPal documentation.

- **response_type** – Must be set to `code`.

- **redirect_uri** – This is a page you host where you present an appropriate page to the user handle the PayPal response. The merchant is redirected to this page after they grant permissions for your application. Use this page to collect the *authorization code* that PayPal generates when the merchant grants the permissions.

- **client_id** – This is the unique Client ID value that PayPal assigned to your application as an API credential when you created it on the PayPal Developer website. Be sure to use the Client ID from the environment you are addressing (Sandbox or live).

> **Note:** The realm/domain of the `redirect_uri` must be the same as the
> realm/domain that you provided on the Developer website when you created
> your application with PayPal and it must be an SSL-enabled domain (for example,
> `https://example.com`).

If you redirect the merchant to PayPal immediately after they finish the onboarding flow,
they will not need to sign back in to PayPal. Instead, they will be prompted to grant your
application the permissions it needs to make calls on their behalf. The user consent page is
shown below with a fictitious business name of *DocTest*:

THE PAYPAL THIRD-PARTY CONSENT PAGE.

When the merchant clicks **Agree**, they are redirected to the `returnURL` you specified in the
redirect, and PayPal appends an *authorization code* to the URL in the form of a `code`
parameter.

Obtaining your Refresh Token and Access Tokens

The authorization code (returned as `code`) is valid for only three minutes. You must use this code to generate the *refresh token* for the merchant's application. In addition to generating the refresh token, the following call also generates an initial *access token*, which you can use in your requests to the PayPal Here APIs:

```
curl https://www.sandbox.paypal.com/webapps/auth/protocol/openidconnect/v1/tokenservice  \
  -u "<Client-Id>:<Secret>"  \
  -d "grant_type=authorization_code
   &code=<Authorization-Code>"
```

Provide the following field values in this request:

- **–u** – The Client ID and Secret API credentials, separated by a colon (:). Like this: `{clientId}:{Secret}`

- **grant_type** – The `grant_type` to get your refresh token must be `authorization_code`.
- **code** – Supply the authorization code generated by PayPal and returned when the merchant grants permission to your app.

Your refresh token POST request returns the following JSON object:

```
{
    "token_type": "Bearer",
    "expires_in": "28800",
    "refresh_token": "<Refresh-Token-Value>"
    "access_token": "<Access-Token-Value>"
}
```

The response fields are as follows:

Field	Type	Description
token_type	String	The type of token issued, which in this case is `Bearer`.
refresh_token	String	The refresh token.
access_token	String	A current access token.
expires_in	Integer	The number of seconds until the access token expires. (default = 28800)

> **Important:** PayPal Here application tokens become inactive if the merchant revokes their third-party permissions from your application via the PayPal website.

About refresh tokens

Refresh tokens are a maximum of 1024 characters in length and do not expire. Like the OAuth Secret assigned to your application, store your application's refresh token only in a secure, persistent data store.

About access tokens

Access tokens are a maximum of 1024 bytes in length and are valid for 8 hours (28800 seconds). The `access_token` value should be used as a runtime variable. Store the value in a short-term program cache (or equivalent) only if necessary.

Authentication token policies

It is important that you adhere to the following policies to ensure the merchant is protected on payments that use PayPal Here requests:

- The application *Secret* must be stored in a secure server facility; the application must not store these values locally.
- You must not store any `access_token` in your services (for example, in a secure data center). You must cache this value in your application to make an API call each time you need to make a PayPal Here request.
- Do not store the `refresh_token` locally on the app. The value of the `refresh_token` should be stored in a secure location in your data center.
- You can use the access token for all PayPal Here operations, including accessing user profiles, getting the status of the payments, and so on.

Onboarding best practices

PayPal is interested in making the PayPal Here sign-up process as streamlined as possible. With this in mind, we recommend that you use the following *best practices* to simplify the merchant sign-up experience:

- Because the PayPal Here SDK supports only US merchants, you should make the PayPal Here onboarding process available to only your merchants who are based in the US.
- If the merchant has not set up any credit card processing relationships or messaging or promotions on their main site, you should prompt them to do so with PayPal.
- Dedicate a page in your back-office application to credit card processing and feature PayPal at the top of the page.
- If the merchant has applied to PayPal for credit card processing, your back-office page should indicate the status of their account (for example. Active, Under Review, or Declined) as provided by the response from the PayPal Here onboarding flow.

- If the user has not applied for PayPal for credit card processing, your back office should prompt them to (1) create a PayPal Here account, and (2) link their PayPal Here account to your system by having them grant the appropriate permissions.
- All merchants using PayPal Here for card processing should also accept PayPal mobile payments by maintaining a valid PayPal location. Do not permit a merchant to activate card processing without an active location. Therefore you should "nest" a toggle to enable card payments underneath the option to enable Mobile Payments and manage the location details, which should happen automatically after the merchant completes account linking.
- Include a way for merchants to remove PayPal linking. Your back office system should also include a message that advises the merchant to also disable the link on the PayPal side. Note that if the merchant removes PayPal linking, you must delete their associated refresh tokens and access tokens.
- Include a way for merchants to discontinue using PayPal for credit card processing, but keep PayPal for check-in payment processing.
- Please use the following verbiage when referring to PayPal products (these terms may be subject to change):

  - *PayPal Mobile Payments*
  - *PayPal Card Processing*
  - *PayPal Here Account* (this terminology is required for merchants who accept swiped credit-card payments)