

The Integration Walkthrough

The Integration Walkthrough (IWT) is our integration-certification process, wherein your Integration Engineer (IE) will review your integration, ensuring that it works well for both buyers and sellers and adheres to our best practices. This document outlines the integration materials required to complete the IWT and the requirements that will be used to assess your integration.

Once the materials have been submitted and the review is complete, we will provide you with feedback, and once any required changes have been implemented, the IWT will be complete. After the IWT is complete, your IE will provision your live account.

Materials Required

To certify your integration, your IE needs to review both the API calls that your integration makes as well as the buyer- and seller-facing user flows offered.

API Samples

Submit plaintext samples for each type of API call that your integration makes, including the body and headers for both the request and response. An example format is below.

```
Create Partner Referral
Endpoint: POST /v2/customer/partner-referrals
Headers sent:
{
  "Authorization": "Bearer ABC...",
  ...
}
Body sent:
{
  "operations": [...],
  ...
}
```

Recordings

Submit a collection of videos, through attachment or accessible link, that demonstrates the functionality of your integration, including

- a seller onboarding successfully onto your platform
- a seller onboarding unsuccessfully, e.g., because they are not ready to transact
- a buyer making a successful purchase using each payment method
- a buyer attempting an unsuccessful purchase using each payment method, e.g., because their payment method was declined


Questionnaire

Please answer all questions below, including screenshots where appropriate.



- Are all PayPal logos displayed taken from official sources?
- Does your integration re-use access tokens until they expire?
- Does your integration handle the case that a refund is requested with insufficient seller balance?
- Are all PayPal JavaScript files loaded dynamically from the official URL rather than saved locally?
- Is the partner BN code being included in the `data-partner-attribution-id` attribute of the JS SDK's script tag?


Best Practices


Section	Subsec.	Best Practice	Description	Docs
Pre-Onboarding	Admin Panel	PayPal is presented as the first payment processor available to sellers.		
Post-Onboarding	Admin Panel	PayPal Checkout is defaulted on for cart pages for onboarded sellers.		
Post-Onboarding	Admin Panel	PayPal Checkout is defaulted on for product pages for onboarded sellers.		
Checkout	JS SDK	The page type passed into the JS SDK's <code>data-page-type</code> attribute.		
Checkout	Prefill	For return buyers who used PayPal for their previous transaction, PayPal is preselected as their payment method during checkout.		
Checkout	Prefill	The buyer's email address or phone number is passed into "create order" API calls for PayPal orders.		
Checkout	Presentment	Payment options are presented equally without a default option selected.		
Checkout	Presentment	PayPal Checkout buttons are featured above other payment options.		
Checkout	Messaging	A "We accept PayPal and Venmo" banner is added to the header of the page.		


Onboarding Requirements

Section	Subsec.	Requirement	Description	Docs
Pre-Onboarding	Experience	Onboarding sellers are directed to PayPal without any deviations.	The onboarding flow should be conducted in a mini-browser or through a full-page redirect.	
Pre-Onboarding	Experience	The onboarding flow is initiated clearly through a sign-up link or button.		
Pre-Onboarding	Partner Referrals	A return URL is provided for onboarding to allow merchants to return to the partner's site.	The return URL should be provided through the <code>partner_config_override.return_url</code> parameter of the "create partner referral" API call.	Partner Referral API
Pre-Onboarding	Partner Referrals	The features included in "create partner referral" API calls match the Solution Design.		
Pre-Onboarding	ACDC	Sellers from ACDC-ineligible countries are onboarded only for PayPal checkout.		



Section	Subsec.	Requirement	Description	Docs
Pre-Onboarding	Apple Pay	<p>Apple Pay-enabled "create partner referral" API calls include</p> <ul style="list-style-type: none"> "PAYMENT_METHODS" in the <code>products</code> array and "APPLE_PAY" in the <code>capabilities</code> array. 		
Pre-Onboarding	Google Pay	<p>Google Pay-enabled "create partner referral" API calls include</p> <ul style="list-style-type: none"> "PAYMENT_METHODS" in the <code>products</code> array and "GOOGLE_PAY" in the <code>capabilities</code> array. 		
Post-Onboarding	Admin Panel	PayPal Checkout is defaulted on for payment pages for onboarded sellers.		
Post-Onboarding	Experience	<p>Sellers are shown their onboarding status, including</p> <ul style="list-style-type: none"> their PayPal account ID and the scopes granted to the partner. 		

Section	Subsec.	Requirement	Description	Docs
Post-Onboarding	Experience	<p>Sellers are notified that they are unable to transact if</p> <ul style="list-style-type: none"> • their primary email address is unconfirmed, • their PayPal account is unable to receive payments, or • permissions were not successfully granted to the partner account. 	<p>Sellers must be able to view their onboarding status. If a seller is unable to transact through PayPal for any of the listed reasons, PayPal checkout should not be rendered for that seller.</p> <p>To verify a seller's onboarding status, make a "show seller status" API call, ensuring that, in the response,</p> <ul style="list-style-type: none"> • the <code>payments_receivable</code> flag is <code>true</code>, • the <code>primary_email_confirmed</code> flag is <code>true</code>, and • the enclosed <code>scopes</code> array contains the permissions requested. 	
		<p>Sellers are able to disconnect and reconnect PayPal accounts within the partner's platform.</p>	<p>A "Disconnect PayPal"-like button is available and presents a confirmation message like "Disconnecting your PayPal account will prevent you from offering PayPal services and products on your website. Do you wish to continue?"</p>	
Post-Onboarding	Partner Referrals	<p>The partner is able to request the onboarding status of a seller with the seller's payer ID.</p>		
Post-Onboarding	Refunds	<p>The platform gracefully handles the error returned in the case that a refund is attempted without sufficient seller balance.</p>		

Section	Subsec.	Requirement	Description	Docs
Post-Onboarding	Refunds	Sellers may issue refunds through the partner's platform.	Sellers can use the Platform to view completed orders, issue refunds, or redirect the seller to their PayPal account.	
Post-Onboarding	Pay Later	Sellers are informed about Pay Later offers and may disable Pay Later.	Pay Later offers should be enabled by default. When disabled, the JS SDK is updated with <code>disable_funding=paylater</code> .	JS SDK
Post-Onboarding	ACDC	<p>Sellers are notified if their ACDC-vetting application</p> <ul style="list-style-type: none"> • requires more information from the seller, • is still in review, or • has been denied. 		

Payments Requirements

Section	Subsec.	Requirement	Description	Docs
Integration Method	JS SDK	Errors thrown by the PayPal JS SDK are caught and handled.		
Integration Method	JS SDK	The JS SDK's script tag includes the partner's BN code in the script tag's <code>data-partner-attribution-id</code> attribute.		
Integration Method	JS SDK	<p>The PayPal JS SDK is configured using the relevant query parameters, including</p> <ul style="list-style-type: none"> the partner's client ID, the seller's payer ID, commit, currency, and intent. 		
Integration Method	JS SDK	The PayPal JS SDK is loaded from the official URL, not saved locally.		
Integration Method	REST API	The partner's BN code is included in the <code>PayPal-Partner-Attribution-Id</code> header in all API calls.		
Integration Method	REST API	PayPal-generated access tokens are re-used until expiration.		



Section	Subsec.	Requirement	Description	Docs
Checkout	Experience	Buyers are not required to input information that is available through PayPal APIs.		
Checkout	REST API	Each order includes item-level detail for each purchase unit.	Item-level detail must be provided through the each purchase unit's <code>items</code> array.	
Checkout	REST API	Each "create order" request specifies a seller using their PayPal account's ID.	The seller may be specified either through each purchase unit's <code>payee</code> parameter or through the <code>PayPal-Auth-Assertion</code> header.	
Checkout	Thank You	<p>Buyers are redirected to a "thank you" page after successful checkout that displays</p> <ul style="list-style-type: none"> the payment source used, the buyer's PayPal email address (if used), the buyer's shipping address (if used), and the buyer's billing address (if used). 		



PayPal Checkout Requirements

Section	Subsec.	Requirement	Description	Docs
Checkout	App Switch	The <code>app_switch_preference</code> parameter is passed as <code>true</code> in the "create order" API request's <code>experience_context</code> object.		Integrate app-switch server-side
Checkout	App Switch	The <code>appSwitchWhenAvailable</code> parameter is passed as <code>true</code> in the JS SDK's <code>paypal.Buttons()</code> setup call.		Integrate app-switch client-side
Checkout	Experience	All buyer-present PayPal Checkout transactions use one-time payments rather than vaulting.		
PayPal	Experience	Buyers are presented with Pay Now Experience when going through PayPal Checkout.		
PayPal	Experience	Buyers are returned to the seller's site after cancelling the PayPal checkout.		
PayPal	Experience	The seller's name appears correctly in the "Cancel and return to {seller-name}" link presented at the bottom of PayPal checkout.		
PayPal	Experience	Buyers can complete checkout within two steps after PayPal checkout.		



Section	Subsec.	Requirement	Description	Docs
PayPal	Experience	Buyers are directed to PayPal without any deviations.		
PayPal	Experience	<p>The complete PayPal experience is in parity with other payments methods across the integration, including</p> <ul style="list-style-type: none"> • PayPal logos and buttons are presented with equal prominence and close proximity to other payment methods; • PayPal logos are taken from official sources; • "PayPal" is capitalized correctly; and • no additional surcharge or fee is added to PayPal transactions. 		
PayPal	Order API	Orders are not created until the buyer has clicked on the PayPal button.		
PayPal	Order API	Orders are updated using a PATCH request if the buyer changes the purchase.		
PayPal	Messaging	Button messaging is shown with the PayPal button.	Integrate messaging directly with your buttons to promote Pay Later offers and other PayPal value propositions to your customers.	
Venmo	Presentment	The Venmo button is rendered for qualifying buyers.		



Section	Subsec.	Requirement	Description	Docs
Venmo	Thank You	The "thank you" page displays Venmo as the payment source if the buyer paid with Venmo.		



Expanded Checkout Requirements

Section	Subsec.	Requirement	Description	Docs
ACDC	Presentment	Card fields are presented during checkout.		
Apple Pay	Presentment	Apple Pay buttons are presented on all product and cart pages.		
Apple Pay	Thank You	The "thank you" page displays Apple Pay as the payment source for Apple Pay-funded transactions.		
Google Pay	Presentment	Google Pay buttons are presented on all cart, product, and checkout pages.		
Google Pay	Thank You	The "thank you" page displays Google Pay as the payment source for Google Pay-funded transactions.		



