

CERTIFIED DEVELOPERS ARGENTINA

Certificación para Checkout de Mercado Pago

¡Bienvenido al examen de certificación del Checkout de Mercado Pago!

Queremos que formes parte de nuestra comunidad de desarrolladores certificados y para ello te invitamos a que realices este examen.

Para lograr obtener esta certificación será necesario que desarrolles un ejercicio práctico y funcional implementando un Checkout de Mercado Pago que cumpla con los estándares de calidad de nuestro equipo de integraciones.

Desde Mercado Pago validaremos tus respuestas para determinar si se cumplió con lo solicitado en el ejercicio. Te recomendamos leer este documento en su totalidad antes de comenzar.

Conocimientos técnicos necesarios:

- Front End: HTML, JavaScript
- Backend: PHP, .NET, Java o Node JS (Con saber alguna de estas tecnologías es suficiente).
- Servidores: Configuraciones básicas
- API REST
- GitHub

Requerimientos técnicos

- Necesitarás subir el ejercicio a un servidor como Heroku o algún servidor que permita disponibilizar estos archivos públicamente en internet. El desarrollo tendrá una parte de front y otra parte de back.

Documentación

POST - API Preferencia:

https://www.mercadopago.com.ar/developers/es/reference/preferences/_checkout_preferences/post/

GET - API Payments:

https://www.mercadopago.com.ar/developers/es/reference/payments/_payments_id/get/

SDK's Backend: PHP, .NET, Java o Node JS

<https://github.com/mercadopago>

Integración Web Checkout:

<https://www.mercadopago.com.ar/developers/es/guides/payments/web-payment-checkout/introduction/>

Notificaciones de Pago:

<https://www.mercadopago.com.ar/developers/es/guides/notifications/webhooks/>

¿Cómo agregar el x-integrator-id al momento de generar una preferencia?

https://www.mercadopago.com.ar/developers/es/guides/payments/web-payment-checkout/configurations/#bookmark_obt%C3%A9n_informaci%C3%B3n_sobre_tu_negocio

Código de seguridad para el comportamiento de clientes:

https://www.mercadopago.com.ar/developers/es/guides/manage-account/payment-rejections/#bookmark_recomendaciones_para_mejorar_tu_aprobaci%C3%B3n

Si programas en PHP o NodeJS, debes utilizar la tienda *dummy* creada por el equipo de Mercado Pago para la realización de este examen. De lo contrario, puedes generar tu propia tienda para hacer las pruebas de este examen.

Tiendas para realizar el examen :

<https://github.com/agustinfrancesconi/mp-ecommerce-php> (php)

<https://github.com/hdlopez/mp-ecommerce-nodejs> (nodejs)

Ejercicio:

Descripción

La empresa “**Tienda e-commerce**” (Argentina) que vende celulares está terminando su sitio de eCommerce. Te han contratado para que implementes el Checkout de Mercado Pago como solución de pago sobre el proyecto que actualmente tiene la empresa.

Requerimiento:

Medios de Pago

El cliente quiere que los pagos con tarjeta de crédito se puedan pagar permitiendo como máximo **6 cuotas** (mensualidades). A su vez, no quiere permitir pagos con tarjetas American Express (**amex**) ni tampoco con medios de pago del tipo cajero automático (**atm**).

Información del pagador

Datos del pagador:

- a) Nombre y Apellido: Lalo Landa
- b) Email: *El email del test-user pagador entregado en este documento.*
- c) Código de área: 11
- d) Teléfono: 22223333

Datos de la dirección del pagador:

- a) Nombre de la calle: False
- b) Número de la casa: 123
- c) Código postal: 1111

Producto

A partir del producto seleccionado, se deberá enviar el mismo como un ítem de la preferencia de pago:

- a) ID: **1234**
- b) Nombre del Producto: ***Nombre del producto seleccionado del carrito del ejercicio.***
- c) Descripción del Producto: **"Dispositivo móvil de Tienda e-commerce"**
- d) URL Imagen: ***Foto del producto seleccionado. (url válida)***
- e) Cantidad: **1**
- f) Precio: ***Precio del producto seleccionado.***
- g) Número de orden del pedido (external_reference): ***Deberás indicar aquí tu Correo electrónico, el mismo que usarás para rellenar el formulario del examen (el primero).***

Páginas de retorno (back_url)

Una vez que el pagador termine el flujo de Mercado Pago, se deberá retornar al sitio web del cliente y manejar una pantalla diferente para cada estado de pago:

- a) El pago haya sido "rechazado" o no haya finalizado (failure)
- b) El usuario haya decidido pagar con un medio de pago offline (ticket) y el pago quede en un estado "pending" o "in_process" (pending)
- c) El pago haya sido exitoso. En la pantalla se deberá mostrar la información proveniente de los parámetros que enviamos en el Query String como: payment_method_id que se usó para pagar, el valor del campo external_reference y el ID de pago (payment_id o collection_id) de Mercado Pago.
 - i) Se deberá configurar para que desde el checkout de Mercado Pago el retorno en caso de pago aprobado sea automático (auto_return).

Nota:

Estos son los parámetros que enviamos en la QueryString cuando redirigimos a las back_url establecidas:

```
https://www.tusitio.com/success.php?collection_id=[PAYMENT_ID]&collection_status=approved&external_ref  
erence=[EXTERNAL_REFERENCE]&payment_type=credit_card&preference_id=[PREFERENCE_ID]&site_id  
=[SITE_ID]&processing_mode=aggregator&merchant_account_id=null
```

Checkout Mercado Pago

El cliente solicitó que el checkout de Mercado Pago se abra en la misma ventan en la que se muestra la tienda (redirect). El botón que inicia el flujo del checkout debe decir: “Pagar la compra”

Al generar la preferencia de pago se debe:

- Enviar la información del ítem que se está comprando.
- Enviar los datos del pagador.
- Enviar la URL donde se van a recibir las notificaciones de pago.
- Enviar el número de orden (`external_reference`)
- Limitar la cantidad de cuotas según lo solicitado.
- No ofrecer los medios de pago según lo solicitados.

Integrator ID

Se debe incluir el `x-integrator-id` que es un código único para identificar a los desarrolladores o agencias que realizaron la integración (para este ejercicio debes utilizar el **integrator_id** de ejemplo que se menciona en este documento, en caso de aprobar la prueba te generaremos tu propio integrator-id).

Notificaciones Webhook

Deberás generar un endpoint que sea capaz de recibir nuestras notificaciones Webhook según la [documentación](#) y en el campo `notification_url` de la preferencia deberá apuntar a éste. En el formulario del examen solicitaremos nos pegues el JSON que recibiste de tu transacción de pago exitoso.

Código de seguridad

Es necesario que incluyas en cada uno de tus archivos el [javascript de seguridad](#) que indicamos en la documentación que puedes encontrar en este documento, que sirve para detectar comportamientos inusuales de los compradores en el sitio. Éste deberá aparecer en el flujo completo de la compra y se deberá enviar correctamente el parámetro “view” asignando el valor correspondiente a la página del sitio que se está visitando (“home”, “search”, “item” o vacío).

Evaluación

Será necesario que subas tu tienda a Heroku o algún servidor visible desde internet públicamente. La tienda no deberá contar con ningún tipo de autenticación para que los usuarios puedan acceder.

Los principales puntos a evaluar son:

- 1) Se revisará la completitud de la preferencia de pago generada para el inicio del flujo de pago (incluyendo el `integrator_id`) según la información facilitada en este ejercicio.
- 2) Se revisará el correcto manejo de las `back_url` para los diferentes estados de pagos.
- 3) Deberás desarrollar un receptor de notificaciones y especificarlo en `notifications_url`. Se revisará el correcto manejo de estas notificaciones de pago. Será necesario que nos envíes el JSON que recibas en el URL de notificaciones, se revisará que coincida con la preferencia generada.
- 4) Se verificará que se haya incluido correctamente el script de seguridad para el comportamiento del usuario en las páginas de tu ejercicio.
- 5) Se revisará que en la página de éxito de `back_urls` muestre los campos requeridos indicados en el ejercicio.

Notas

Para la generación de la preferencia deberán utilizar las credenciales productivas del test-user de Mercado Pago que indicamos en la siguiente página más abajo.

Para la simulación del pago, deberás redireccionar al usuario al “**init_point**” productivo que obtendrás de nuestra API de preferencias (no utilizar “`sandbox_init_point`”).

Información Adicional

Integrator-id

id: dev_24c65fb163bf11ea96500242ac130004

Test User (Vendedor)

collector_id	469485398
Public-Key	APP_USR-7eb0138a-189f-4bec-87d1-c0504ead5626
Access-Token	APP_USR-6317427424180639-042414-47e969706991d3a442922b0702a0da44-469485398

Para probar los flujos, deberán usar un test-user que será el pagador o comprador y podrán utilizar las tarjetas de prueba de Mercado Pago:

Test User (Comprador o Pagador)

ID	471923173
Email	test_user_63274575@testuser.com
Password	qatest2417

Tarjetas para hacer las simulaciones de pago en testeo

<https://www.mercadopago.com.ar/developers/es/guides/payments/web-payment-checkout/test-integration/>