

# DEVELOPER PARTNER PROGRAM

## Examen de certificación del Checkout Pro de Mercado Pago

Bem-vindo ao exame de certificação do Checkout PRO Mercado Pago!

Queremos que você faça parte de nossa comunidade de desenvolvedores certificados e, para isso, convidamos a fazer este exame.

Para obter esta certificação, será necessário que você desenvolva um exercício prático e funcional de implementação de um Checkout PRO Mercado Pago que atenda aos padrões de qualidade de nossa equipe de integração.

Validaremos suas respostas para determinar se a solicitação no exercício foi atendida. Recomendamos que leia este documento por completo antes de começar.

### Conhecimento técnico necessário:

- Front End: HTML, JavaScript
- Backend: PHP, .NET, Java o Node JS (Conhecer qualquer uma dessas tecnologias é o suficiente).
- Servidores: Configurações básicas
- API REST
- GitHub

### Requerimentos técnicos

- Você precisará subir o exercício em um servidor como o Heroku ou em um servidor que permita que esses arquivos sejam disponibilizados publicamente na Internet. O desenvolvimento terá uma parte frontend e uma parte backend.

- Se você programa em [PHP](#) ou [NodeJS](#), é possível usar a loja fictícia criada pela equipe do Mercado Pago para realizar este exame. Caso contrário, você pode criar sua própria loja para fazer os testes deste exame.

## Exercício:

### Descrição

A empresa “Tienda e-commerce” que vende celulares está finalizando seu site de eCommerce. Você foi contratado para implantar o [Checkout PRO](#) Mercado Pago como solução de pagamento do projeto que a empresa possui atualmente.

### Requerimento:

#### Meios de pagamento

O cliente deseja que os pagamentos com cartão de crédito sejam pagos, permitindo no máximo 6x (parcelas). Ao mesmo tempo, não pretende permitir pagamentos com cartões American Express (amex).

#### Informação do pagador

Detalhes do pagador:

- Nome e sobrenome: **Lalo Landa**
- Email: ***O e-mail do usuário de teste pagante entregue neste documento.***
- Código de área: **55**
- Telefone: **98529-8743**

Detalhes do endereço do pagador:

- Nome da rua: **Insurgentes Sur**
- Número da casa: **1602**
- Código postal: **78134-190**

## Producto

Do produto selecionado, deve ser enviado como item da preferência de pagamento:

- a) ID: **1234**
- b) Nome do Produto: **Nome do produto selecionado no carrinho de venda.**
- c) Descrição do produto: **“Celular de Tienda e-commerce”**
- d) URL Imagen: **Foto do produto selecionado. (url válida)**
- e) Quantidade: **1**
- f) Preço: **Preço do produto selecionado.**
- g) Número de ordem do pedido (external\_reference): **Deve indicar aqui o seu Email, o mesmo que utilizará para preencher o formulário do exame. (o primeiro).**

## Páginas de retorno (back\_url)

Assim que o pagador conclui o fluxo do Mercado Pago, ele deve retornar ao site do cliente e lidar com uma tela diferente para cada status de pagamento:

- a) O pagamento foi "rejeitado" ou não foi concluído (falha)
- b) O usuário decidiu pagar com um método de pagamento offline (boleto) e o pagamento permanece em um status “pendente” ou “in\_process” (pendente)
- c) O pagamento foi realizado com sucesso. A tela deve mostrar as informações dos parâmetros que enviamos na Query String como: payment\_method\_id que foi usado para pagar, o valor do campo external\_reference e o ID do pagamento (payment\_id ou collection\_id) do Mercado Pago.
  - i) Deve ser configurado para que a partir do checkout Mercado Pago o retorno em caso de pagamento aprovado seja automático (auto\_return).

### Nota:

Estes são os parâmetros que enviamos na QueryString quando redirecionamos para o back\_url estabelecido:  
`https://www.tusitio.com/success.php?collection_id=[PAYMENT_ID]&collection_status=approved&external_reference=[EXTERNAL_REFERENCE]&payment_type=credit_card&preference_id=[PREFERENCE_ID]&site_id=[SITE_ID]&processing_mode=aggregator&merchant_account_id=null`

## Checkout PRO Mercado Pago

O cliente solicitou que o Checkout PRO Mercado Pago fosse aberto na mesma janela em que a loja está exposta. O botão que inicia o fluxo de checkout deve dizer: "Pague a compra"

Ao gerar a preferência de pagamento, você deve:

- Enviar as informações do item que está sendo adquirido.
- Enviar os detalhes do pagador.
- Enviar o URL onde as notificações de pagamento serão recebidas.
- Enviar o número do pedido (`external_reference`)
- Limitar a quantidade de parcelas conforme solicitado.
- Não oferecer o meio de pagamento solicitado.

## Integrator ID

Deve ser incluído o [x-integrator-id](#), que é um código único para identificar os desenvolvedores ou agências que realizaram a integração (para este exercício você deve usar o exemplo **integrator\_id** mencionado neste documento, caso seja aprovado no teste, iremos gerar seu próprio integrador-id).

## Notificações Webhook

Você deve gerar um endpoint que seja capaz de receber nossas notificações de Webhook de acordo com a [documentação](#) e no campo `notification_url` da preferência deve apontar para ele. No formulário de exame, pediremos que você cole o JSON que recebeu da transação de pagamento bem-sucedida.

## Código de segurança

É necessário que inclua em cada um dos seus arquivos o [javascript de segurança](#) que indicamos na documentação que encontra neste documento, que serve para detectar comportamentos anormais dos compradores no site. Este deve constar no fluxo completo de compras e o parâmetro "view" deve ser enviado corretamente, atribuindo o valor correspondente à página do site que está sendo visitada ("home", "search", "item" ou vazio).

## Avaliação

Você precisará carregar sua loja no Heroku ou em um servidor que seja publicamente visível na Internet. A loja não deve ter nenhum tipo de autenticação para acesso dos usuários.

Os principais pontos a avaliar são:

- 1) A completude da preferência de pagamento gerada para o início do fluxo de pagamento (incluindo o `integrator_id`) será revisada de acordo com as informações fornecidas neste exercício.
- 2) O tratamento correto do `back_url` para os diferentes status de pagamento será revisado.
- 3) Você precisará desenvolver um receptor de notificação e especificá-lo em `notification_url`. O tratamento correto dessas notificações de pagamento será analisado. Será necessário que você nos envie o JSON que recebe na URL de notificação, será verificado se corresponde à preferência gerada.
- 4) Será verificado se o script de segurança para o comportamento do usuário foi incluído corretamente nas páginas do seu projeto.
- 5) Será verificado se a página de sucesso `back_urls` mostra os campos obrigatórios indicados no exercício.

### Notas

Para gerar a preferência, eles devem utilizar as credenciais produtivas do usuário-teste do Mercado Pago que indicamos na página abaixo.

Para a simulação de pagamento, você deve redirecionar o usuário para o produtivo "init\_point" que você obterá de nossa API de preferências (não use "sandbox\_init\_point").

## Informações adicionais

Integrator-id

id: **dev\_24c65fb163bf11ea96500242ac130004**

### Test User (Vendedor)

Access-Token	<b>APP_USR-334491433003961-030821-12d7475807d694b645722c1946 d5ce5a-725736327</b>
--------------	---

Para testar os fluxos, eles devem usar um usuário teste que será o pagador ou comprador e poderá usar os [cartões de teste](#) do Mercado Pago.

### Test User (Pagador)

Email	<b>test_user_92801501@testuser.com</b>
-------	--