



# Guía de Integración

API InstaPago versión 1.7

# Tabla de contenido

Histórico de cambios .....	3
Introducción.....	4
Formas de pago.....	5
Iniciar en InstaPago.....	6
Requisitos .....	6
Registro .....	7
Creación de llaves .....	7
Uso del API.....	10
Punto de Entrada del API.....	10
Crear Pago.....	10
Completar pago .....	13
Anular Pago .....	15
Consulta de Pago.....	17
Consulta de Pago por Orden de Compra .....	19
Consulta de Pagos por Lote.....	21
Consulta de Pagos por Fecha .....	22
Tarjetas de prueba .....	23
Códigos de respuesta de la llamada al API.....	24
Códigos de respuesta de las transacciones .....	25
Ejemplos de llamadas al API .....	26
Parámetros de Certificación .....	28
Recomendaciones.....	31

# Histórico de cambios

Fecha	Versión	Autor	Descripción
23/08/2013	<b>1.0</b>	John Benavides	○ Creación del Documento
28/10/2013	<b>1.1</b>	Alex Istúriz	○ Nuevo tipo del Request para Payment
07/11/2013	<b>1.2</b>	Alex Istúriz	○ Nuevo parámetro para el Request de Payment ○ Nueva URL del servicio API
20/01/2014	<b>1.3</b>	Alex Istúriz	○ La respuesta del pago incluye el voucher de la compra. Esta debe ser mostrada en el portal y correo electrónico. ○ Recordatorio de generar llave para ambiente " Al Aire
14/03/2014	<b>1.4</b>	Alex Istúriz	○ Nuevo método del servicio API para procesar preautorizaciones
16/05/2014	<b>1.5</b>	Alex Istúriz	○ <b>Modificación al método de preautorizaciones</b> ○ Nuevo método para completar preautorizaciones ○ Nuevo método para anular preautorizaciones ○ Nuevo método para consultar preautorizaciones
20/05/2014	<b>1.5.1</b>	Alex Istúriz	○ <b>Fé de errata URL de completar el pago</b> ○ Ahora todos los métodos devuelven en la respuesta el número de orden si es indicado por el comercio ○ Para el método de crear pago se debe enviar la IP del cliente de manera obligatoria
26/06/2014	<b>1.5.2</b>	César Higuera	○ <b>Nueva imagen de InstaPago en la Guía</b> ○ Correcciones de semántica
29/07/2014	<b>1.5.3</b>	Antonio Romero / Rodrigo Arellano	○ <b>Incorporación de parámetros de certificación</b> ○ <b>Incorporación de recomendaciones</b>
27/03/2015	<b>1.5.4</b>	Yelitza Cedeño / Alex Istúriz	○ Actualización de códigos de ejemplo ○ Actualización de creación de llaves ○ Actualización formato del documento
19/10/2015	<b>1.6</b>	Alex Isturiz	○ Se incluye en todas las respuestas el número de lote asociado a la transacción, el código de respuesta emitido por el banco, monto procesado y fecha de la transacción.
24/11/2017	<b>1.7</b>	Alex Isturiz	○ Nuevo método para consultar los pagos por orden de compra ○ Nuevo método para consultar los pagos por lote y por terminal ○ Nuevo método para consultar los pagos por rango de fechas

# Introducción

InstaPago es una solución tecnológica pensada para el mercado de comercio electrónico (e-Commerce) en Venezuela y Latinoamérica, con la intención de ofrecer un producto de primera categoría, que permita a las personas y empresas apalancar sus capacidades de expansión, facilitando los mecanismos de pago para sus clientes, con una integración amigable a los sistemas que actualmente utilizan.

Cualquier comercio que desee recibir pagos por internet, ya sea desde su sitio web o una aplicación móvil, puede apoyarse en InstaPago y uno de los bancos afiliados para lograr sus objetivos.

Esta Guía de Integración tiene el objeto de facilitar a los departamentos de tecnología de los comercios y a desarrolladores independientes a integrar sus soluciones web con la plataforma de pagos en línea InstaPago.

# Formas de pago

Formas de pago aceptadas:



Procesamos a través de:



# Iniciar en InstaPago

Una de las ventajas principales de InstaPago, es la forma de pago mediante comunicación directa con su Payment Gateway, ya sea desde el sitio web de un comercio o una aplicación móvil. El usuario o cliente final nunca es redirigido fuera del sitio donde está comprando, esto tiene las siguientes ventajas importantes:

- Mejor experiencia de usuario
- Mayor rapidez en el proceso de compra
- Mayor control por parte del comercio dentro de sus procesos con sus clientes

## Requisitos

El Comercio debería cumplir con los siguientes requisitos para poder recibir pagos a través de InstaPago:

- Dominio propio
- Certificado SSL
- Código fuente propio o acceso al mismo
- Alojamiento en servidor propio o terceros

InstaPago es compatible con todos los lenguajes web y de plataformas móviles del mercado (PHP, Java, Asp.net C# ó VB, Ruby, Objective-C, etc). Básicamente, cualquier lenguaje que tenga implementado el protocolo HTTP.

La integración con el Payment Gateway de InstaPago se realiza mediante su API basado en métodos RESTful bajo HTTP.


# Registro

1. El primer paso **como desarrollador**, es registrarse en el portal de InstaPago, para tener acceso a la Extranet de consultas y comenzar a realizar pruebas.
2. Acceda a <https://banesco.instapago.com>
3. Haga click sobre el link “Regístrese” al final de la página.
4. Complete los datos básicos solicitados para crear su cuenta. **No importa si no es un comercio, puede crear la cuenta con sus datos personales.**

# Creación de llaves

Al registrarse en InstaPago, tendrá acceso directo e inmediato al API en modo Pruebas. Esto le permitirá desarrollar las funcionalidades de pago en línea en su sitio web ó aplicación móvil desde el momento que crea su cuenta.

Para interactuar con el API de InstaPago, usted recibirá al momento de registrarse la llave pública de conexión, que es básicamente un token que identifica a su comercio y/o usuario cuando está realizando una transacción con InstaPago. Adicionalmente deberá crear una llave privada que identificara la aplicación que interactúa con el API.

Para hacerlo, ingrese en el menú  y seleccione la opción “Configuración de API” como muestra la imagen de la siguiente página.



Ya en la sección de “Configuración del API” se muestra las llaves que tiene creadas, y permite crear nuevas. El concepto detrás de esto es que un comercio puede tener distintos puntos de venta. Por ejemplo, un distribuidor de productos electrónicos puede poseer un punto de venta en línea para detal o clientes finales, y otro para clientes Corporativos. InstaPago le permite crear una llave para cada punto de pago, y así poder



realizar consultas por separado de las transacciones recibidas por ambos puntos, y hacer conciliaciones por separado.

Para crear una llave Privada, se le asigna un nombre para identificarlo (por ejemplo, "WebSite"), y se hace click en el botón "Generar". Con esto ya tiene su llave la cual podrá utilizar para interactuar con InstaPago en modo Pruebas.

**IMPORTANTE:** Una vez completados los requisitos ante el banco para tener acceso al ambiente de producción, debe crear nuevamente las llaves para este modo.

InstaPago | Banesco  
Contigo

Ajustes de API

Inicio / Ajustes de API

Generar Nueva Llave

Descripción

Generar

Enviar llave pública

Listado de Llaves

Recuerde: Si esta en modo 'Pruebas' las llaves que verá en la lista sólo funcionarán para este modo.

Id	Descripción	Estado	
A5C61F46-9653-472D-AFC5-FF52A21BE2A3	Llave Privada	activa	Anular

# Uso del API

## Punto de Entrada del API

Todas las interacciones con InstaPago se realizan a través de la siguiente dirección URL:

<https://api.instpago.com>

## Crear Pago

Este método consta del envío de los datos de un pago con tarjeta de crédito a InstaPago para su autorización:

**URL:** <https://api.instpago.com/payment>

**Método HTTP:** POST

**Tipo de codificación:** application/x-www-form-urlencoded

**Cuerpo de la consulta (body):**

- **KeyId (Requerido):** Llave generada desde InstaPago.
- **PublicKeyId (Requerido):** Llave compartida (Enviada por correo electrónico al crear la cuenta en el portal de InstaPago)
- **Amount (Requerido):** Monto a Debitar, utilizando punto “.” como separador decimal. Por ejemplo: 200.00.
- **Description (Requerido):** Cadena de caracteres con la descripción de la operación.
- **CardHolder (Requerido):** Nombre del Tarjeta habiente.
- **CardHolderID (Requerido):** Cédula del Tarjeta habiente.
- **CardNumber (Requerido):** Numero de la tarjeta de crédito, sin espacios ni separadores.

- **CVC (Requerido):** Código secreto de la Tarjeta de crédito.
- **ExpirationDate (Requerido):** Fecha de expiración de la tarjeta en el formato mostrado en la misma MM/YYYY. Por Ejemplo: 10/2014.
- **StatusId (Requerido):** Estatus en el que se creará la transacción.
  - **"1":** Retener (pre-autorización).
  - **"2":** Pagar (autorización).
- **IP (Requerido):** Dirección IP del cliente que genera la solicitud del pago.
- **OrderNumber (Opcional):** Numero de orden del pago según el comercio.
- **Address (Opcional):** Dirección asociada a la tarjeta, Utilizada por algunos bancos para mayor seguridad.
- **City (Opcional):** Ciudad asociada a la tarjeta, Utilizada por algunos bancos para mayor seguridad.
- **ZipCode (Opcional):** Código Postal asociada a la tarjeta, Utilizada por algunos bancos para mayor seguridad.
- **State (Opcional):** Estado o provincia asociada a la tarjeta, Utilizada por algunos bancos para mayor seguridad.

### Ejemplo:

```
POST /Payment HTTP/1.1
Host: https://api.instagago.com/payment
Cache-Control: no-cache
Content-Type: application/x-www-form-urlencoded
```

KeyId=00000000-0000-0000-0000-000000000000&PublicKeyId=00000000000000000000000000  
000&Amount=10.00&Description=Pago+Servicios+de+InstaPago&CardHolder=Juan+Perez&CardH  
olderId=12345678&CardNumber=4111111111111111CVC=123&ExpirationDate=10%2F2021&StatusI  
d=2&IP=127.0.0.1&OrderNumber=000000010&Address=Ave+Las+Palmas+Casa+12&City=Ciudad+Bo  
livar&ZipCode=1902&State=Bolivar



## Completar pago

Este método funciona para procesar un pago retenido o pre-autorización, para así realizar el cargo respectivo.

**URL:** <https://api.instapago.com/complete>

## Método HTTP: POST

**Tipo de codificación:** application/x-www-form-urlencoded

**Cuerpo de consulta (body):**

- **KeyId (Requerido):** Llave generada desde InstaPago.
- **PublicKeyId (Requerido):** Llave compartida enviada por correo al crear una cuenta en InstaPago.
- **Id (Requerido):** Identificador único del pago.
- **Amount (Requerido):** Monto por el cual se desea procesar el pago final.

### Ejemplo:

```
POST /Payment HTTP/1.1
Host: https://api.instapago.com/complete
Cache-Control: no-cache
Content-Type: application/x-www-form-urlencoded

KeyId=00000000-0000-0000-0000-000000000000&PublicKeyId=00000000000000000000000000000000
0000&Id=00000000-0000-0000-0000-000000000000&Amount=10.00
```

## Respuesta

○ **Tipo:** application/json

○ **Valores:**

- "success"**: Indica si fue procesado el pago (true o false),
- "message"**: Descripción de la respuesta (Máx. 200 caracteres),
- "id"**: Código del pago (Máx. 32 caracteres),
- "code"**: Código del respuesta del pago (Máx. 3 caracteres),
- "reference"**: Número de referencia del pago (Máx. 6 caracteres),
- "voucher"**: Informe de la transacción en formato HTML codificado,
- "ordernumber"**: Número de orden indicado por el comercio,
- "sequence"**: Número de identificación bancario universal (Máx. 12 caracteres),
- "approval"**: Número de aprobación bancaria (Máx. 6 caracteres),
- "lote"**: Número de lote asignado a la transacción (Máx. 3 caracteres),
- "responsecode"**: Código de respuesta de la transacción (Máx. 2 caracteres),
- "deferred"**: Indica si la transacción ha sido diferida (true o false),
- "datetime"**: Fecha de la transacción (Formato MM/dd/yyyy hh:mm:ss tt),
- "amount"**: Monto de la transacción (Formato 0.00)

## Ejemplo:

```
{
  "success": true,
  "message": "Pago Aprobado",
  "id": "00000000-0000-0000-0000-000000000000",
  "code": "201",
  "reference": "000003",
  "voucher": "<table width='100%'><tbody><tr><td><div style='border: 1px solid #222; padding: 9px; text-align: center; margin-left: auto; margin-right: auto; margin-bottom: 10px;' id='voucher'><style type='text/css'>.normal-left {font-family: Tahoma; font-size: 7pt; text-align: left;} .normal-right {font-family: Tahoma; font-size: 7pt; text-align: right;} .big-center {font-family: Tahoma; font-size: 9pt; text-align: center; font-weight: 900;} .big-center-especial {font-family: Tahoma; font-size: 9pt; text-align: center; font-weight: 900; letter-spacing: .9em;} .big-left {font-family: Tahoma; font-size: 9pt; text-align: left; font-weight: 900;} .big-right {font-family: Tahoma; font-size: 9pt; text-align: right; font-weight: 900;} .normal-center {font-family: Tahoma; font-size: 7pt; text-align: center}</div></td></tr></tbody></table>",
  "sequence": "508600010003", "approval": "210220", "lote": "122", "datetime" = "10/25/2015 04:05:11 PM", "amount", "10.00" }
```

# Anular Pago

Este método funciona para procesar una anulación de un pago, ya sea un pago o un bloqueo.

**URL:** <https://api.instapago.com/payment>

**Método:** DELETE

**Tipo de codificación:** application/x-www-form-urlencoded

**Cuerpo de consulta (body):**

- **KeyId (Requerido):** Llave generada desde InstaPago.
- **PublicKeyId (Requerido):** Llave compartida Enviada por correo al crear una cuenta en InstaPago.
- **Id (Requerido):** Identificador único del pago.

**Ejemplo:**

```
DELETE /Payment HTTP/1.1
Host: https://api.instapago.com/payment
Cache-Control: no-cache
Content-Type: application/x-www-form-urlencoded
KeyId=00000000-0000-0000-0000-000000000000&PublicKeyId=00000000000000000000000000000000
000&Id=00000000000000000000000000000000
```

## Respuesta

- **Tipo:** application/json
- **Valores:**
  - "success":** Indica si fue procesado el pago (true o false),
  - "message":** Descripción de la respuesta (Máx. 200 caracteres),
  - "id":** Código del pago (Máx. 32 caracteres),
  - "code":** Código del respuesta del pago (Máx. 3 caracteres),
  - "reference":** Número de referencia del pago (Máx. 6 caracteres),
  - "voucher":** Informe de la transacción en formato HTML codificado,
  - "ordernumber":** Número de orden indicado por el comercio,
  - "sequence":** Número de identificación bancario universal (Máx. 12 caracteres),
  - "approval":** Número de aprobación bancaria (Máx. 6 caracteres),

- "lote"**: Número de lote asignado a la transacción (Máx. 3 caracteres),
- "responsecode"**: Código de respuesta de la transacción (Máx. 2 caracteres),
- "deferred"**: Indica si la transacción ha sido diferida (true o false),
- "datetime"**: Fecha de la transacción (Formato MM/dd/yyyy hh:mm:ss tt),
- "amount"**: Monto de la transacción (Formato 0.00)

### Ejemplo:

```
{
  "success": true,
  "message": "Pago Anulado",
  "id": "00000000-0000-0000-0000-000000000000",
  "code": "201",
  "reference": "000003",
  "voucher": "\ \ \ \ <table width="100%">\ \ <tbody>\ \ <tr>\ \ <td>\ \ <div style="border: 1px solid #222; padding: 9px; text-align: center; margin-left: auto; margin-right: auto; margin-bottom: 10px;" id="voucher">\ \ \ \ <style type="text/css">\ \ .normal-left {\ \ font-family: Tahoma;\ \ font-size: 7pt;\ \ text-align: left;\ \ }\ \ \ \ .normal-right {\ \ font-family: Tahoma;\ \ font-size: 7pt;\ \ text-align: right;\ \ }\ \ \ \ .big-center {\ \ font-family: Tahoma;\ \ font-size: 9pt;\ \ text-align: center;\ \ font-weight: 900;\ \ }\ \ \ \ .big-center-especial {\ \ font-family: Tahoma;\ \ font-size: 9pt;\ \ text-align: center;\ \ font-weight: 900;\ \ letter-spacing: .9em;\ \ }\ \ \ \ .big-left {\ \ font-family: Tahoma;\ \ font-size: 9pt;\ \ text-align: left;\ \ font-weight: 900;\ \ }\ \ \ \ .big-right {\ \ font-family: Tahoma;\ \ font-size: 9pt;\ \ text-align: right;\ \ font-weight: 900;\ \ }\ \ \ \ .normal-center {\ \ font-family: Tahoma;\ \ font-size: 7pt;\ \ text-align: center",
  "sequence": "508600010003", "approval": "210220", "lote": "122", "datetime" = "10/25/2015 04:05:11 PM", "amount", "10.00" }
```



## Consulta de Pago

Este método funciona para consultar un pago generado anteriormente.

**URL:** <https://api.instapago.com/payment>

**Método:** GET

**Tipo de codificación:** application/x-www-form-urlencoded

**Variables de la cadena de consulta (querystring):**

- **KeyId (Requerido):** Llave generada desde InstaPago.
- **PublicKeyId (Requerido):** Llave compartida Enviada por correo al crear una cuenta en InstaPago.
- **Id (Requerido):** Identificador único del pago.

**Ejemplo:**

```
GET /Payment HTTP/1.1
Host: https://api.instapago.com/payment
Cache-Control: no-cache
Content-Type: application/x-www-form-urlencoded

KeyId=00000000-0000-0000-0000-000000000000&PublicKeyId=00000000000000000000000000000000
000&Id=00000000000000000000000000000000
```

## Respuesta

- **Tipo:** application/json
- **Valores:**
  - "**success**": Indica si fue procesado el pago (true o false),
  - "**message**": Descripción de la respuesta (Máx. 200 caracteres),
  - "**id**": Código del pago (Máx. 32 caracteres),
  - "**code**": Código del respuesta del pago (Máx. 3 caracteres),
  - "**reference**": Número de referencia del pago (Máx. 6 caracteres),
  - "**voucher**": Informe de la transacción en formato HTML codificado,
  - "**ordernumber**": Número de orden indicado por el comercio,
  - "**sequence**": Número de identificación bancario universal (Máx. 12 caracteres),
  - "**approval**": Número de aprobación bancaria (Máx. 6 caracteres),

- "lote"**: Número de lote asignado a la transacción (Máx. 3 caracteres),
- "responsecode"**: Código de respuesta de la transacción (Máx. 2 caracteres),
- "deferred"**: Indica si la transacción ha sido diferida (true o false),
- "datetime"**: Fecha de la transacción (Formato MM/dd/yyyy hh:mm:ss tt),
- "amount"**: Monto de la transacción (Formato 0.00)

### Ejemplo:

```
{
  "success": true,
  "message": "Pago Aprobado",
  "id": "00000000-0000-0000-0000-000000000000",
  "code": "201",
  "reference": "000003",
  "voucher": "\ \ \ \ <table width="100%">\ \ <tbody>\ \ <tr>\ \ <td>\ \ <div style="border: 1px solid #222; padding: 9px; text-align: center; margin-left: auto; margin-right: auto; margin-bottom: 10px;" id="voucher">\ \ \ \ <style type="text/css">\ \ .normal-left {\ \ font-family: Tahoma;\ \ font-size: 7pt;\ \ text-align: left;\ \ }\ \ \ \ .normal-right {\ \ font-family: Tahoma;\ \ font-size: 7pt;\ \ text-align: right;\ \ }\ \ \ \ .big-center {\ \ font-family: Tahoma;\ \ font-size: 9pt;\ \ text-align: center;\ \ font-weight: 900;\ \ }\ \ \ \ .big-center-especial {\ \ font-family: Tahoma;\ \ font-size: 9pt;\ \ text-align: center;\ \ font-weight: 900;\ \ letter-spacing: .9em;\ \ }\ \ \ \ .big-left {\ \ font-family: Tahoma;\ \ font-size: 9pt;\ \ text-align: left;\ \ font-weight: 900;\ \ }\ \ \ \ .big-right {\ \ font-family: Tahoma;\ \ font-size: 9pt;\ \ text-align: right;\ \ font-weight: 900;\ \ }\ \ \ \ .normal-center {\ \ font-family: Tahoma;\ \ font-size: 7pt;\ \ text-align: center",
  "sequence": "508600010003",
  "approval": "210220"
}
```

# Consulta de Pago por Orden de Compra

Este método funciona para consultar los pagos asociados (aprobados y/o rechazados) a través del campo de OrdenNumber enviado al momento de crearlos.

**URL:** <https://api.instepago.com/query>

**Método:** GET

**Tipo de codificación:** application/x-www-form-urlencoded

**Variables de la cadena de consulta (querystring):**

- **KeyId (Requerido):** Llave generada desde InstaPago.
- **PublicKeyId (Requerido):** Llave compartida Enviada por correo al crear una cuenta en InstaPago.
- **OrderNumber (Requerido):** Numero de orden del pago según el comercio.

**Ejemplo:**

```
GET /Payment HTTP/1.1
Host: https://api.instepago.com/payment
Cache-Control: no-cache
Content-Type: application/x-www-form-urlencoded

KeyId=00000000-0000-0000-0000-000000000000&PublicKeyId=00000000000000000000000000000000
000&OrderNumber=00000000000000000000000000000000
```

## Respuesta

- **Tipo:** application/json, arreglo [ ]
- **Valores (Lista):**
  - "success":** Indica si fue procesado el pago (true o false),
  - "message":** Descripción de la respuesta (Máx. 200 caracteres),
  - "id":** Código del pago (Máx. 32 caracteres),
  - "code":** Código del respuesta del pago (Máx. 3 caracteres),
  - "reference":** Número de referencia del pago (Máx. 6 caracteres),
  - "voucher":** Informe de la transacción en formato HTML codificado,
  - "ordernumber":** Número de orden indicado por el comercio,
  - "sequence":** Número de identificación bancario universal (Máx. 12 caracteres),

- "approval"**: Número de aprobación bancaria (Máx. 6 caracteres),
- "lote"**: Número de lote asignado a la transacción (Máx. 3 caracteres),
- "responsecode"**: Código de respuesta de la transacción (Máx. 2 caracteres),
- "deferred"**: Indica si la transacción ha sido diferida (true o false),
- "datetime"**: Fecha de la transacción (Formato MM/dd/yyyy hh:mm:ss tt),
- "amount"**: Monto de la transacción (Formato 0.00)

### Ejemplo:

```
[
  {
    "success": true,
    "message": "Pago Aprobado",
    "id": "00000000-0000-0000-0000-000000000000",
    "code": "201",
    "reference": "000003",
    "voucher": "\ \ \ \ <table width="100%">\ \ <tbody>\ \ <tr>\ \ <td>\ \ <div style="border: 1px solid #222; padding: 9px; text-align: center; margin-left: auto; margin-right: auto; margin-bottom: 10px;" id="voucher">\ \ \ \ <style type="text/css">\ \ .normal-left {\ \ font-family: Tahoma;\ \ font-size: 7pt;\ \ text-align: left;\ \ }\ \ \ \ .normal-right {\ \ font-family: Tahoma;\ \ font-size: 7pt;\ \ text-align: right;\ \ }\ \ \ \ .big-center {\ \ font-family: Tahoma;\ \ font-size: 9pt;\ \ text-align: center;\ \ font-weight: 900;\ \ }\ \ \ \ .big-center-especial {\ \ font-family: Tahoma;\ \ font-size: 9pt;\ \ text-align: center;\ \ font-weight: 900;\ \ letter-spacing: .9em;\ \ }\ \ \ \ .big-left {\ \ font-family: Tahoma;\ \ font-size: 9pt;\ \ text-align: left;\ \ font-weight: 900;\ \ }\ \ \ \ .big-right {\ \ font-family: Tahoma;\ \ font-size: 9pt;\ \ text-align: right;\ \ font-weight: 900;\ \ }\ \ \ \ .normal-center {\ \ font-family: Tahoma;\ \ font-size: 7pt;\ \ text-align: center",
    "sequence": "508600010003",
    "approval": "210220"
  },
  {
    "success": false,
    "message": "Pago Rechazado",
    "id": "00000000-0000-0000-0000-000000000000",
    "code": "403",
    "reference": "000002",
    "voucher": "",
    "sequence": "508600010002",
    "approval": "000000"
  }
]
```

## Consulta de Pagos por Lote

Este método funciona para consultar los pagos asociados (aprobados y/o rechazados) a un lote y terminal en particular.

**URL:** <https://api.instepago.com/reports>

**Método:** GET

**Tipo de codificación:** application/x-www-form-urlencoded

**Variables de la cadena de consulta (querystring):**

- **KeyId (Requerido):** Llave generada desde InstaPago.
- **PublicKeyId (Requerido):** Llave compartida Enviada por correo al crear una cuenta en InstaPago.
- **Terminal (Requerido):** Numero de orden del pago según el comercio.
- **Lote (Requerido):** Numero de orden del pago según el comercio.

**Ejemplo:**

```
GET /Payment HTTP/1.1
Host: https://api.instepago.com/payment
Cache-Control: no-cache
Content-Type: application/x-www-form-urlencoded

KeyId=00000000-0000-0000-0000-000000000000&PublicKeyId=0000000000000000000000000000
000&Terminal=000000&Lote=000
```

## Respuesta

- **Tipo:** text/csv (Archivo de texto separado por punto y coma)
- **Contenido:**

```
Referencia_Instepago;Fecha;Cliente;Cedula;Descripcion;Monto;Estatus;Anulada;Cerrada;Tipo;Tarjeta;Lote
;Aprobacion;Referencia;Secuencia;Terminal;Orden;Llave;Banco;Tasa;ISLR;Comision;TotalNeto
0169e180-77e5-406f-a3f4-ba4d5399bc03;10/3/2017 14:54;Mariano Caicedo;5611712;Test
;450;Completado;NO;NO;MasterCard;525739*****6278;20;6485;7;7,27606E+11;20800641;;WEB;Otros
Bancos;5;20,09;22,5;407,41
```

## Consulta de Pagos por Fecha

Este método funciona para consultar los pagos (aprobados y/o rechazados) dado un rango de fechas y un terminal.

**URL:** <https://api.instapago.com/reports>

### Método: POST

**Tipo de codificación:** application/x-www-form-urlencoded

**Cuerpo de la consulta (body):**

- **KeyId (Requerido):** Llave generada desde InstaPago.
- **PublicKeyId (Requerido):** Llave compartida Enviada por correo al crear una cuenta en InstaPago.
- **Terminal (Requerido):** Numero de terminal usado para el pago según el comercio.
- **StartDate (Requerido):** Fecha inicial de la consulta.
- **EndDate (Requerido):** Fecha final de la consulta.

### Ejemplo:

[illegible]

## Respuesta

- **Tipo:** text/csv (Archivo de texto separado por punto y coma)
- **Contenido:**

```
Referencia_Instapago;Fecha;Cliente;Cedula;Descripcion;Monto;Estatus;Anulada;Cerrada;Tipo;Tarjeta;Lote
;Aprobacion;Referencia;Secuencia;Terminal;Orden;Llave;Banco;Tasa;ISLR;Comision;TotalNeto
0169e180-77e5-406f-a3f4-ba4d5399bc03;10/3/2017 14:54;Mariano Caicedo;5611712;Test
;450;Completado;NO;NO;MasterCard;525739*****6278;20;6485;7;7,27606E+11;20800641;;WEB;Otros
Bancos;5;20,09;22,5;407,41
```

## Tarjetas de prueba

Para realizar las pruebas, se provee de los siguientes datos para comprobar la integración:

- **Tarjetas aprobadas:** Pueden indicar cualquier valor para Cédula o RIF, Fecha de Vencimiento y CVC:
  - **Visa:** 4111 1111 1111 1111
  - **MasterCard:** 5105 1051 0510 5100
  - **Sambil:** 8244 0011 0011 0011
  - **Rattan:** 8244 0211 0011 0011
  - **Locatel:** 8244 0411 0011 0011
- **Tarjeta rechazada:** Cualquier número de tarjeta de crédito Visa, MasterCard, American Express, Sambil, Rattan y Locatel válidas serán “Rechazadas”. **Esto no genera ninguna transacción con el Banco.**

# Códigos de respuesta de la llamada al API

Para todas las transacciones realizadas bajo el API, los códigos HTTP de respuestas corresponden a los siguientes estados:

- **201:** Pago procesado, se detalla la transacción en la respuesta.
- **400:** Error al validar los datos enviados (Adicionalmente se devuelve una cadena de caracteres con la descripción del error).
- **401:** Error de autenticación, ha ocurrido un error con las llaves utilizadas.
- **403:** Pago Rechazado por el banco.
- **500:** Ha Ocurrido un error interno dentro del servidor.
- **503:** Ha Ocurrido un error al procesar los parámetros de entrada. Revise los datos enviados y vuelva a intentarlo.

Si recibe un código de respuesta diferente a los antes descritos deben ser tomados como errores de protocolo HTTP.



## Códigos de respuesta de las transacciones

En todas las respuestas de las transacciones se añade un valor de respuesta “**responsecode**”, que corresponde a la respuesta que arroja el banco al momento de procesar la transacción. Sus valores posibles son los siguientes:

Código	Motivo				
00	APROBADO	41	TARJETA EXTRAVIADA	79	FECHA INVALIDA
01	LLAME AL EMISOR	43	TARJETA ROBADA	80	SIS. NO DISPONIBLE
02	CEDULA INVALIDA	51	FONDO INSUFICIENTE	81	ERROR DE CLAVE
03	COMERCIO INVALIDO	52	NO ES CTA. CTE.	82	TARJETA INVALIDA
04	RETENGA Y LLAME	53	NO ES CTA. AHORRO	83	COD SEG INVALIDO
05	TRANS. RECHAZADA	54	TARJETA VENCIDA	84	AUTORIZ. INVALIDA
06	ERROR	55	CLAVE INVALIDA	85	NO RECHAZADA
07	RETENGA Y LLAME	57	TRANS.NO PERMITIDA	86	CLAVE NO VERIF.
12	TRANS. INVALIDA	58	TRANS,NO PERMITIDA	87	CONCIL.DETENIDA
13	MONTO INVALIDO	61	EXC.MONTO RETIROS	88	TOTALES NO DISP.
14	TARJETA INVALIDA	62	TARJ. RESTRINGIDA	89	INF. NO AUTORIZADA
15	EMISOR INVALIDO	63	FALLA SEGURIDAD	90	SERVICIO INACTIVO
19	REINTENTE TRANS.	65	EXCEDE RETIROS	91	EMISOR INACTIVO
21	LLAME AL EMISOR	68	SIN RESP. DEL HOST	92	SERVICIO NO DISP.
25	REG. NO LOCALIZADO	71	EXCEPCION INVAL.	93	TRANS. INCOMPLETA
28	ARCHIVO. NO DISP.	72	CONTRACARGO INVAL.	94	TRANS. DUPLICADA
30	ERROR EN FORMATO	75	EXC. CLAVE INVAL.	95	ENVIANDO DETALLE
39	NO ES CTA. CREDITO	76	CUENTA INVALIDA	96	SISTEMA INACTIVO
40	FUNC. NO SOPORTADA	77	CUENTA INVALIDA	97	SIN RESP. DEL HOST
		78	CUENTA INVALIDA	99	RECHAZADA

## Ejemplos de llamadas al API

A continuación se incluyen algunos ejemplos de cómo realizar las llamadas al API para autorizaciones desde distintos lenguajes de programación. Cabe destacar que **todas las llamadas que se hagan al API de InstaPago deben hacerse desde el servidor donde se encuentra alojado el sitio o aplicación del comercio y no directamente desde el dispositivo del cliente.**

### PHP

```
<?php

$url = 'https://api.instapago.com/payment';
$fields = array("KeyId" => " ", "PublicKeyId" => " ", "Amount" => "1", "Description" => "
", "CardHolder" => " ", "CardHolderId" => " ", "CardNumber" => " ", "CVC" => " ",
"ExpirationDate" => " ", "StatusId" => " ", "Address" => " ", "City" => " ", "ZipCode" => "
", "State" => " " );

$ch = curl_init();

curl_setopt($ch, CURLOPT_URL, $url );
curl_setopt($ch, CURLOPT_POST, 1);
curl_setopt($ch, CURLOPT_POSTFIELDS, http_build_query($fields));
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);

$server_output = curl_exec ($ch);
curl_close ($ch);
echo $server_output ;

?>
```

## C#

```

WebRequest __webrequest = WebRequest.Create("https://api.instapago.com/payment") as
HttpWebRequest;

string postData =
String.Format("KeyId={Account_Key}&&Amount={Monto}&Description=={Texto-
descripcion}&CardHolder={tarjeta-habiente}& CardHolderId={cedula-RIF-tarjeta-
habiente}&CardNumber={numero-tarjeta}&CVC={codigo-seguridad-tarjeta}&ExpirationDate={
fecha-expiracion(mm/yyyy)}&StatusId= {status(1=retenido,2=pagado)",

        formData.Payments.First().Cardholder,
        formData.Payments.First().CardNumber,
        formData.Payments.First().CardNumberId,
        formData.Payments.First().SecurityCode,
        formData.Payments.First().ExpMonth.ToString("00"),
        formData.Payments.First().ExpYear.ToString()
    );
byte[] __postDataStream = Encoding.UTF8.GetBytes(postData);

__webrequest.Method = "POST";
__webrequest.ContentType = "application/x-www-form-urlencoded";
__webrequest.ContentLength = __postDataStream.Length;
Stream __requestStream = __webrequest.GetRequestStream();
__requestStream.Write(__postDataStream, 0, __postDataStream.Length);
__requestStream.Close();

// response
WebResponse __webresponse = __webrequest.GetResponse();
Stream dataStream = __webresponse.GetResponseStream();
StreamReader reader = new StreamReader(dataStream);
string responseFromServer = reader.ReadToEnd();

JavaScriptSerializer __serializer = new JavaScriptSerializer();
Models.InstapagoRS __instaRs =
__serializer.Deserialize(responseFromServer, typeof(Models.InstapagoRS)) as
Models.InstapagoRS;

if (__instaRs.success == true)
    String __codigoAprobacion = __instaRs.id.ToString();
else
    String __errorMessage = "Lo sentimos, no hemos podido procesar su tarjeta de crédito. El
mensaje del banco fue: " + __instaRs.message;

```

# Parámetros de Certificación

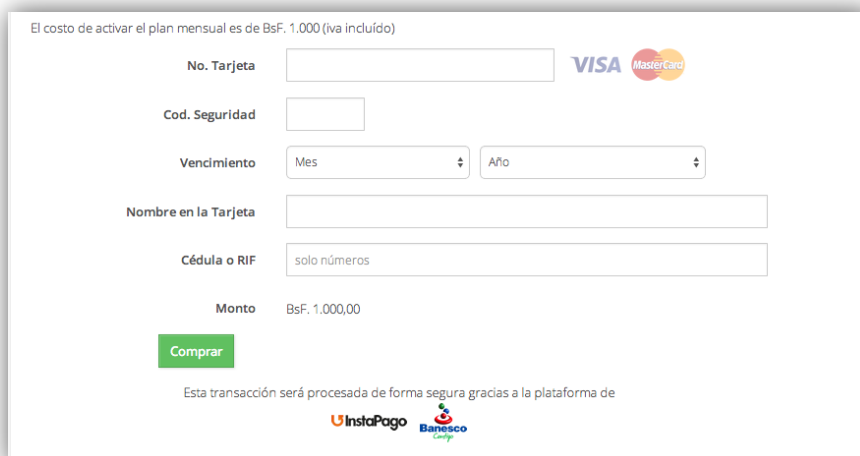
Los parámetros de certificación presentados a continuación definen los puntos que serán evaluados por parte del equipo técnico de InstaPago, a fin de garantizar que el comercio y su portal web (o aplicación móvil) cuenta con los requerimientos mínimos para ser certificados. **Esta certificación es un requerimiento indispensable para permitir que el comercio pase a producción con la plataforma de pagos, y el banco adquirente genere el afiliado comercial.** Los parámetros a verificar son los siguientes:

- El portal debe requerir el registro del usuario para acceder al proceso de compra.
- El certificado de seguridad (SSL) debe estar instalado tanto para el proceso de compras, como para el proceso de registro de usuario. Los parámetros y pruebas que serán objeto a evaluar en el certificado del SSL para ser aprobado es el siguiente:



<b>Configuración SSL</b>	<ul style="list-style-type: none"> <li>▪ SSL 2.0 debe estar Deshabilitado.</li> <li>▪ SSL 3.0 debe estar Deshabilitado.</li> <li>▪ TLS 1.0 debe estar Habilitado.</li> <li>▪ TLS 1.1 debe estar Habilitado.</li> <li>▪ TLS 1.2 debe estar Habilitado.</li> <li>▪ Conjuntos cifrados débiles deben estar deshabilitados.</li> <li>▪ Debe ser compatible con Forward Secrecy</li> </ul>
<b>Pruebas a aplicar en los Certificados</b>	<ul style="list-style-type: none"> <li>▪ Sin cadena de confianza</li> <li>▪ El certificado aún no es válido</li> <li>▪ Certificado caducado</li> <li>▪ Falta de correspondencia con el nombre de host</li> <li>▪ Certificado revocado</li> <li>▪ Certificado autofirmado</li> <li>▪ Firma débil</li> <li>▪ Vulnerabilidad ante claves débiles de Debian</li> <li>▪ Los identificadores uniformes de recursos (URIs) resuelven a direcciones IPv4</li> <li>▪ Los identificadores uniformes de recursos (URIs) resuelven a direcciones IPv6</li> </ul>
<b>Pruebas HTTP</b>	<ul style="list-style-type: none"> <li>▪ Sin contenido mixto (HTTP y HTTPS)</li> <li>▪ El nombre de dominio resuelve a una dirección IPv4</li> <li>▪ El nombre de dominio resuelve a una dirección IPv6</li> </ul>

- Formulario de pago:
  - No se deben permitir caracteres especiales en ninguno de los campos del formulario.
  - Limitar la opción del navegador para almacenar información de campos sensibles (número de tarjeta, fecha de vencimiento, CVV2/CVC).
  - Campos:
    - Nombre: debe permitir sólo caracteres alfabéticos incluyendo la letra ñ y el espacio en blanco para que el usuario pueda introducir su apellido.
    - Cédula: debe permitir sólo caracteres numéricos, limitar la cantidad a 6 dígitos como mínimo y 8 dígitos como máximo.
    - Tipo de Tarjeta (este campo no es necesario): mostrar opciones únicamente VISA y MASTERCARD.
    - Número de tarjeta: sólo caracteres numéricos, limitar a 15 dígitos como mínimo y 16 dígitos como máximo.
    - Fecha de Vencimiento: no debe permitir fechas en pasado sólo fechas válidas que sean mayor a la fecha en curso.
    - CVC: sólo caracteres numéricos, limitar a 3 dígitos y enmascarar los dígitos.
  - Al final del formulario se debe incluir el siguiente texto: “Esta transacción será procesada de forma segura gracias a la plataforma de:” debajo de éste texto debe incluir los logotipos de InstaPago (lado izquierdo) y Banesco (lado derecho), ambos a la misma altura.

### Ejemplo:



El costo de activar el plan mensual es de BsF. 1,000 (iva incluido)

No. Tarjeta   

Cod. Seguridad

Vencimiento Mes  Año



Nombre en la Tarjeta

Cédula o RIF

Monto BsF. 1,000.00

[Comprar](#)

Esta transacción será procesada de forma segura gracias a la plataforma de

- Una vez el usuario presione el botón de aceptar o procesar, para activar el proceso de cobro, éste deberá ser bloqueado u ocultado mientras espera respuesta del sistema, de forma que el usuario no lo pueda presionar más de una vez.
  - o Mensaje de respuesta: Una vez sea recibida la respuesta de InstaPago, se debe mostrar la información correspondiente al cliente, incluyendo el voucher generado por el sistema.

Ejemplo:



Si ya se encuentra listo para certificar su sitio web, envíenos un correo a [soporte@instapago.com](mailto:soporte@instapago.com) con los siguientes datos:

- Nombre del comercio
- RIF
- URL donde se realizará la certificación
- URL donde se encuentra instalado el certificado SSL

# Recomendaciones

Los siguientes comentarios no son requisitos para la certificación del proceso de compra; sin embargo, representan alternativas de prácticas utilizadas en experiencias previas y que han logrado dar buenos resultados en el flujo de las operaciones y experiencia de los usuarios.

- **Flujo de inicio de sesión:** A pesar de que el inicio de sesión es obligatorio para poder generar la compra, la experiencia del usuario es mejor si ese proceso se realiza únicamente en el momento que el cliente desea proceder finalmente con la compra; evitando así interrumpir la experiencia en el portal en el llenado del carrito de compras.
- **Botón de aceptar o procesar:** Cuando el botón de aceptar o procesar es bloqueado u oculto, mientras se espera la respuesta del sistema, se recomienda colocar una animación en la cual se indique que existe un proceso en curso.
- **Validar cada campo tanto del lado del cliente como del servidor.**
- **Navegadores:** Revisar que el funcionamiento sea óptimo en los navegadores disponibles, incluyendo versiones anteriores de los mismos. Es común encontrar usuarios con versiones de navegadores anteriores, por limitaciones en la versión del sistema operativo.
- **Enviar al correo del usuario una confirmación de la compra:** incluyendo los datos de la respuesta de InstaPago, mostrando el voucher de compra.