



DOCUMENTACIÓN GENERAL DE INTEGRACIÓN ESTÁNDAR PAGOEFFECTIVO- COMERCIO ELECTRONICO

CONTENIDO

1. OBJETIVO.....	4
2. ALCANCE	4
3. GENERALIDADES.....	4
3.1. ¿Qué es PagoEfectivo?.....	4
3.2. ¿Cómo funciona?.....	4
3.3. Estados de PagoEfectivo	4
4. AMBIENTES DE INTEGRACION	5
4.1. Ambiente de Integraciones.....	5
4.2. Ambiente de producción	5
5. Especificación Técnica de Integración.....	5
5.1. Servicios web	5
Headers	8
Body.....	8
5.2. Request URL Notificación JSON (notificación de estados del CIP).....	14

Documento Técnico - Integraciones

1. OBJETIVO

El objetivo de este documento es explicar cómo funciona la pasarela de pago PagoEfectivo y cómo este puede ser integrado en la plataforma de un comercio afiliado.

2. ALCANCE

El documento presenta la información del proceso de integración entre PagoEfectivo y la plataforma del comercio afiliado, así mismo, se incluye las especificaciones técnicas resaltando la descripción del contrato de servicios web y el sistema de notificación.

3. GENERALIDADES

3.1. ¿Qué es PagoEfectivo?

PagoEfectivo es el servicio de Pasarela de Pago para efectuar transacciones por Internet del Grupo El Comercio y busca brindar una alternativa de pago vía internet a toda persona que desee realizar pagos online sin la necesidad de contar con tarjetas de crédito o débito.

3.2. ¿Cómo funciona?

El proceso consiste en generar un Código de identificación de pago (CIP) que luego el usuario podrá pagarlo acercándose a uno de nuestros centros de pago afiliado; cuando se efectúe el pago, automáticamente el centro de pago notificará al sistema PagoEfectivo y a la vez PagoEfectivo notificará al comercio afiliado que el pago se ha realizado satisfactoriamente.

Ver <https://www.youtube.com/watch?v=N1NJabZgUKQ>

3.3. Estados de PagoEfectivo

Cuando el cliente decide pagar con PagoEfectivo, se crea un CIP desde la plataforma del comercio afiliado, este CIP nace con estado Pendiente de pago.

EL CIP creado cambia a estado Cancelado sólo cuando el cliente paga el CIP a través de uno de nuestros centros autorizados.

Cuando se crea un CIP, este tiene una fecha de expiración (varía dependiendo del rubro del negocio), si aún no se ha pagado el CIP y la fecha de expiración vence, el CIP pasa de estado Pendiente de pago ha estado Expirado.

4. AMBIENTES DE INTEGRACION

PagoEfectivo cuenta con dos ambientes para el proceso de integración, un ambiente de INTEGRACIONES y otro ambiente de PRODUCCIÓN.

Dichos ambientes son los siguientes:

4.1. Ambiente de Integraciones

Ambiente donde se desarrollará inicialmente el proyecto de integración.

- Autorización de generación de CIP
<http://pre1a.services.pagoefectivo.pe/v1/authorizations>
- Generar CIP
<http://pre1a.services.pagoefectivo.pe/v1/cips>

4.2. Ambiente de producción

Ambiente final donde el comercio afiliado podrá realizar los recaudos.

- Autorización de generación de CIP
<https://services.pagoefectivo.pe/v1/authorizations>
- Generar CIP
<https://services.pagoefectivo.pe/v1/cips>

5. Especificación Técnica de Integración

La integración entre el **[Comercio]** y **PagoEfectivo** se dará por medio de Servicios web (REST) para los casos de autorización, generación de CIP. Se requiere que el **[Comercio]** cuenta con certificado digital.

5.1. Servicios web

Los servicios web que se consumen en el flujo de pago de **PagoEfectivo** son los siguientes:

- Autorización para Generar CIP
- Generar CIP

5.1.1 Autorización de generación de CIP.

Nombre:	v1/Authorizations
---------	-------------------

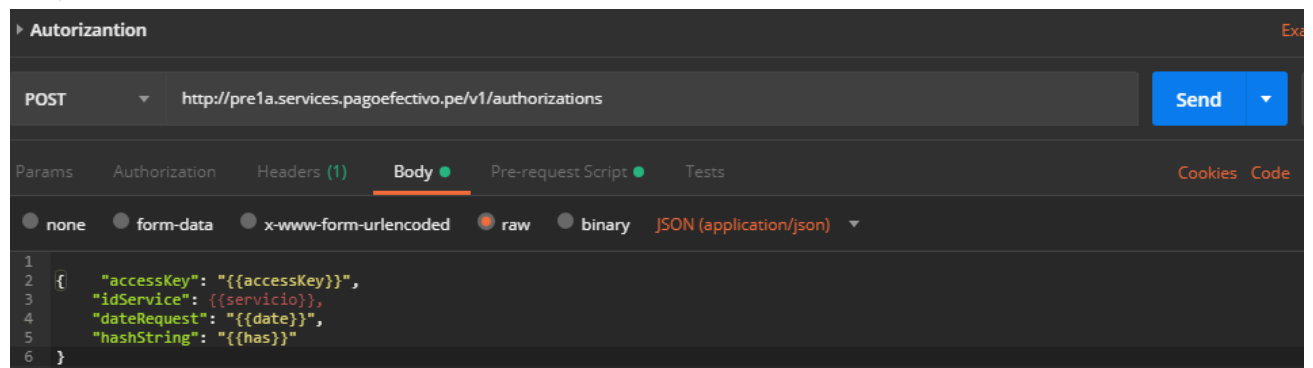
• Datos de entrada

Nro.	Campo	Atributo	Opcional/ Mandatorio	Longitud	Descripción
1	accessKey	AN	M	20	Llave de acceso otorgado por PagoEfectivo.
2	idService	N	M	2	Identificador del Comercio proporcionado por PagoEfectivo.
3	dateRequest	AN	M	25	Fecha que se solicita la autorización para generar CIP, formato UTC. yyyy-mm-ddThh:mm:ss(+/-)zzzz (GTM-5) Hora Lima / Quito / Bogotá Ejemplo: 2018-11-15T15:45:00-05:00

4	hashString	AN	M	64	<p>Valor SHA256 generado por la concatenación de los componentes:</p> <p><code>idService.accesKey.secretKey.dateRequest</code></p> <p>IdService, accesKey y secreKey son valores proporcionados por PagoEfectivo al Comercio, Ejemplo:</p> <p><code>8.AKIPJP77AHN2DKVIJURB.Nfybo8h0yN7C FN1ycX+XaG99pj/y3Vt25urt1LUR.2018-11-20T16:01:00-05:00</code></p> <p>SHA256:</p> <p><code>3faa0677fe78c30800d5c9ec71218ddaa81d15 ca08e39f50177bbe328906c744</code></p>
---	------------	----	---	----	--

- Datos de entrada – Ejemplo con POSTMAN**

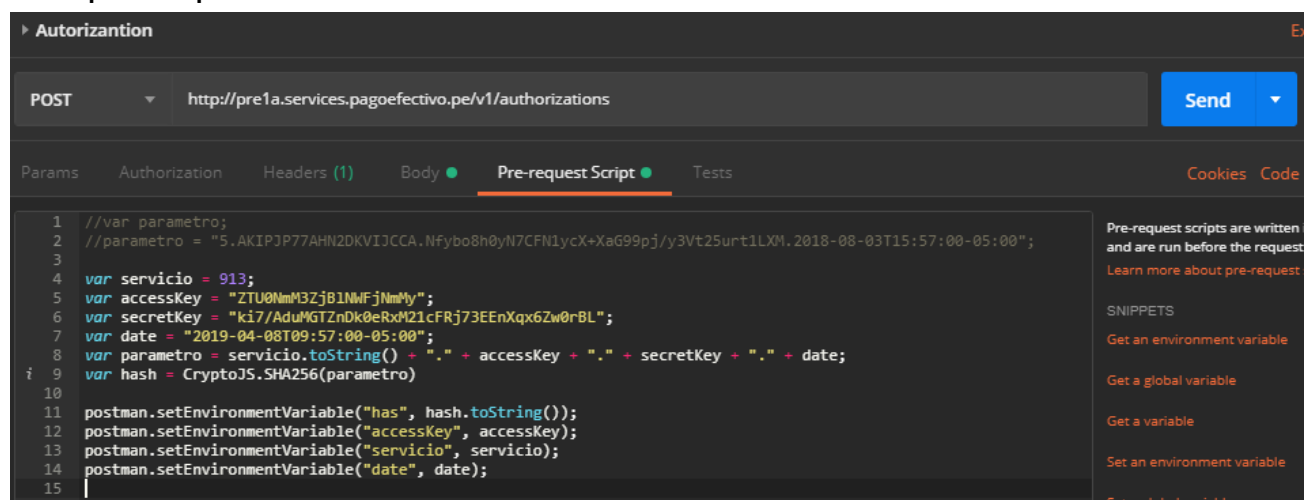
Body:



```

{
  "accessKey": "{{accessKey}}",
  "idService": "{{servicio}}",
  "dateRequest": "{{date}}",
  "hashString": "{{has}}"
}
  
```

PreRequest Script:



```
//var parametro;
//parametro = "5.AKIPJP77AHN2DKVIJCCA.Nfybo8h0yN7CFN1ycX+XaG99pj/y3Vt25urt1LXM.2018-08-03T15:57:00-05:00";
```

```
var servicio = 913;
var accessKey = "ZTU0NmM3ZjBINWFjNmMy";
var secretKey = "ki7/AduMGTZnDk0eRxM21cFRj73EEnXqx6Zw0rBL";
var date = "2019-04-08T09:57:00-05:00";
var parametro = servicio.toString() + "." + accessKey + "." + secretKey + "." + date;
var hash = CryptoJS.SHA256(parametro)
```

```
postman.setEnvironmentVariable("has", hash.toString());
postman.setEnvironmentVariable("accessKey", accessKey);
postman.setEnvironmentVariable("servicio", servicio);
postman.setEnvironmentVariable("date", date);
```

Nota: el campo Date tiene que ser un getdate (GTM-5 - Lima) la fecha y hora que se está solicitando la autorización.

Url: <https://...../v1/authorizations> (Post)

- Datos de salida – Autorización de generación de CIPs**

Nro.	Campo	Atributo	Opcional/ Mandatorio	Longitud	Descripción
1	code	N	M	3	Código del mensaje de respuesta
2	message	AN	M	200	Mensaje de respuesta
3	Data	AN	M		Detalle de la respuesta

- Datos de salida - Data**

Nro.	Campo	Atributo	Opcional/ Mandatorio	Longitud	Descripción
1	token	AN	M	25	Valor generado por PagoEfectivo que permite la generación de cip e invocación de las Apis PagoEfectivo.
2	codeService	AN	O	1-30	Código del comercio en PagoEfectivo.
3	tokenStart	AN	M	1-14	Fecha de Inicio de vigencia del token.
4	tokenExpires	AN	M	1-14	Fecha de expiración del token.

- Datos de salida - Ejemplo**

```
//Status: 200 OK
{
  "code": "100",
```

```

"message": "Solicitud exitosa.",
"data": {
  "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiJVUkIiLCJqdGkiOiJkZjI2MzFlNC1mNmVjLTQzMTEtYTVlNC02MzNiMmFhNzgwYTEiLCJuYW1laWQiOiI4IiwiaXhwIjoxNTcyNzUxMDM5fQ.L0ykLoaxgmI18RWgdCND-dkqn3TpKG9TZqPqe-qIMuY",
  "codeService": "SRV",
  "tokenStart": "2018-11-20T16:57:19-05:00",
  "tokenExpires": "2019-11-02T22:17:19-05:00",
}
}

```

Nota: con el token se puede generar varios CIP hasta que expire, la vigencia del token es según el acuerdo con el Comercio.

5.1.2 Generar CIP

Nombre:	v1/cips
---------	---------

• Datos de entrada - Generar CIP

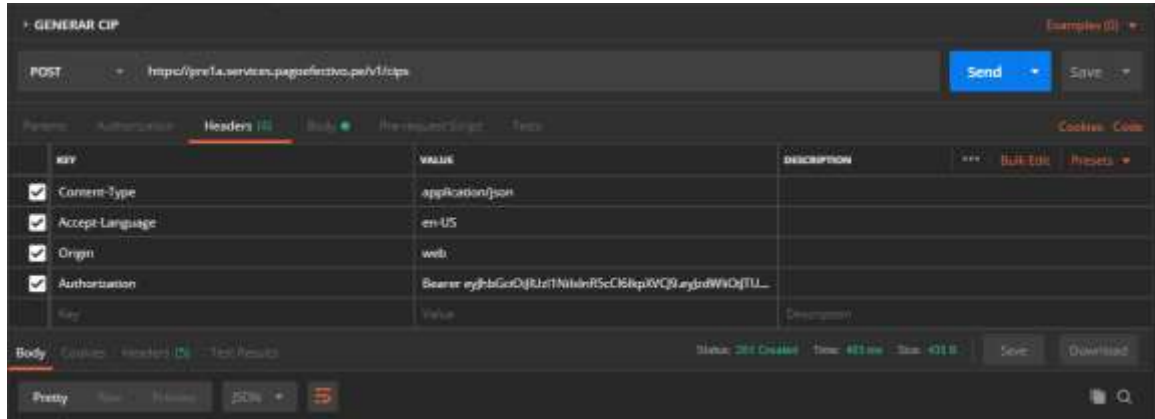
Headers					
Nro	Campo	Atributo	Opcional/ Mandatorio	Longitud	Descripción
1	Accept-Language	A	O	5	Idioma del invocador, valores permitidos es-PE (Español Perú) ó en-US (Ingles Americano) .
2	Origin	A	M	3-7	Identifica el origen de la invocación, debe ser uno de los siguientes: <ul style="list-style-type: none"> web android ios
3	Authorization	AN	M	206	Token de autenticación, ejemplo: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiJVUkIiLCJqdGkiOiJkZjI2MzFlNC1mNmVjLTQzMTEtYTVlNC02MzNiMmFhNzgwYTEiLCJuYW1laWQiOiI4IiwiaXhwIjoxNTcyNzUxMDM5fQ.L0ykLoaxgmI18RWgdCND-dkqn3TpKG9TZqPqe-qIMuY Entre Bearer y el token debe haber un carácter de espacio.

Body					
Nro	Campo	Atributo	Opcional/ Mandatorio	Longitud	Descripción
1	currency	A	M	3	Moneda de la operación, valores aceptados: <ul style="list-style-type: none"> PEN (soles) USD (Dólares)

- Datos de entrada - Ejemplo

Url: **Https://...../ v1/cips (Post)**

Header:



Content-Type: application/json

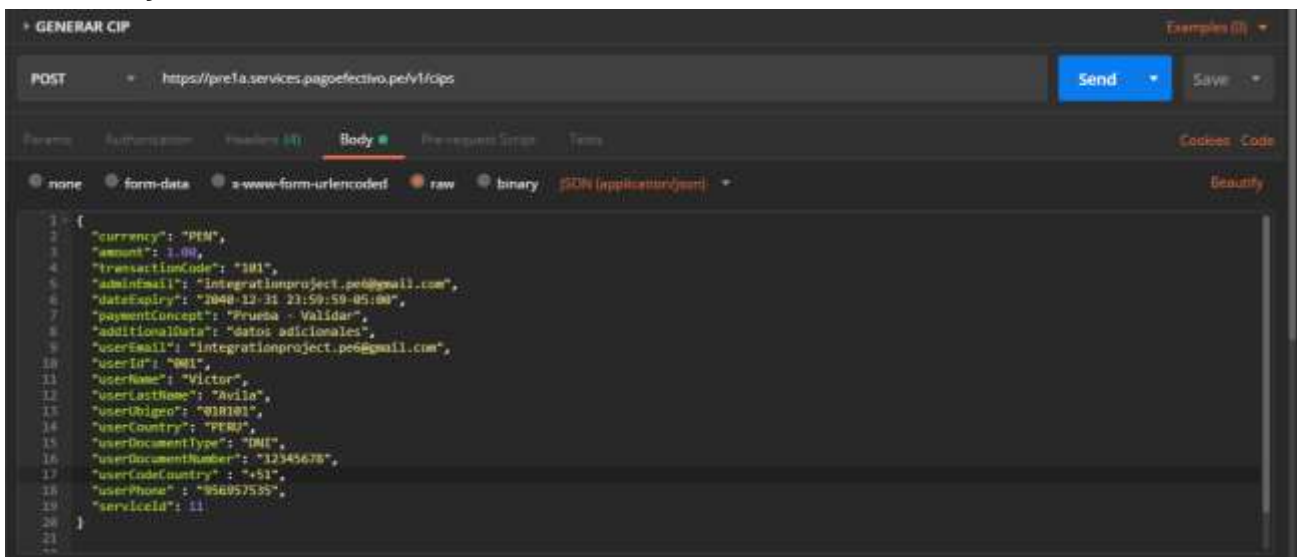
Accept-Language : " en-US"

Origin : web

Authorization : Bearer

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiJTU1YiLCJqdGkiOiIwYzUwMzZmNS0zZTk4LTQzZjU0OGM1Yi0yMGM2OTBiYTc0ZjciLCJyZW1laWQiOiIiXMSiImV4cCI6MTU5MjE4MzIxOX0.Lw2aRuQ3h16629m7KIEoonUvaAbVmS58yTiHhZmdvNY

Body:



```
{
  "currency": "PEN",
  "amount": 1.00,
  "transactionCode": "101",
  "adminEmail": "integrationproject.pe6@gmail.com",
  "dateExpiry": "2040-12-31 23:59:59-05:00",
  "paymentConcept": "Prueba - Validar",
  "additionalData": "datos adicionales",
  "userEmail": "integrationproject.pe6@gmail.com",
```

```
"userId": "001",
"userName": "Victor",
"userLastName": "Avila",
"userUbigeo": "010101",
"userCountry": "PERU",
"userDocumentType": "DNI",
"userDocumentNumber": "12345678",
"userCodeCountry" : "+51",
"userPhone" : "956957535",
"serviceId": 11
}
```

- **Datos de salida - Generar CIP**

Nro.	Campo	Atributo	Opcional/ Mandatorio	Longitud	Descripción
1	code	N	M	3	Código del mensaje de respuesta
2	message	AN	M	200	Mensaje de respuesta
3	Data	AN	M		Detalle de la respuesta

- **Datos de salida - Data**

Nro	Campo	Atributo	Opcional/ Mandatorio	Longitud	Descripción
1	cip	N	M	1-14	Código de CIP generado.
2	currency	A	M	3	Moneda de la operación, valores devueltos: <ul style="list-style-type: none"> • PEN (soles) • USD (Dólares)
3	amount	N	M	18	El Monto de la operación. El separador es un "." 15 dígitos numéricos 1 dígito carácter (.) 2 dígitos decimales
4	transactionCode	AN	M	1-36	Código que identifica la operación, valor proporcionado por el comercio, puede ser hasta 36 caracteres.
5	dateExpiry	DATETIME	M	25	Fecha de expiración el CIP. El formato es UTC yyyy-mm-ddThh:mm:ss(+/-)zzzz Ejemplo 2018-11-21T14:30:20-05:00. Si no se envía la fecha a expirar se tomará el tiempo acordado por defecto con el Comercio Electrónico.
6	cipUrl	AN	M	76	Url del documento html del CIP generado.

- **Datos de salida - Ejemplo**

```
//Status: 200 OK
{
  "code": 100,
  "message": "Successful request.",
  "data": {
    "cip": 2523136,
    "currency": "PEN",
    "amount": 1,
    "transactionCode": "101",
    "dateExpiry": "2040-12-31T23:59:59-05:00",
    "cipUrl": "http://pre1a.payment.pagoeffectivo.pe/AB803C1C-3266-4CFF-A236-4D9DD5AD260A.html"
  }
}
```

- **Datos de salida – Documento HTML del CIP generado.**



Respuesta de validaciones

- **Esquema Jon de respuesta.**

```
{
  "code": "111",
  "message": "Solicitud con datos inválidos.",
  "data": [
    {
      "code": "113",
      "message": "Valor inválido.",
      "field": "salePointCode"
    }
  ]
}
```

- **Mensajes de respuesta.**

200 - Ok (Solicitud exitosa)

CODE	MENSAJE (ES-PE)	DESCRIPCIÓN
100	Solicitud exitosa.	Solicitudes exitosas que no sean creaciones y eliminaciones.

400 - Bad request (Solicitud incorrecta)

CODE	MENSAJE (ES-PE)	DESCRIPCIÓN
111	Solicitud con datos inválidos.	Los datos de la solicitud no son validos.
112	Requerido.	Un atributo o elemento requerido no ha sido especificado.
113	Valor invalido.	Un atributo o elemento especificado contiene datos no válidos.
114	Formato inválido.	Un valor de atributo o elemento especificado no cumple con el formato esperado.
115	Longitud inválida.	Un valor de cadena no cumple la restricción de longitud de un atributo.
116	Rango inválido.	El intervalo de valores no es válido
117	Rango de fecha inválido.	El intervalo de fechas no es válido
508	Servicio asociado al token es inválido.	
509	La fecha de expiración supera el límite establecido.	
510	La fecha de expiración es menor a la fecha actual.	
511	Usted ha generado un Código CIP que excede el valor máximo permitido en este comercio. Por favor intente con un importe de menor cuantía.	
591	Expirado.	
592	Generado.	
593	Pagado.	
594	Eliminado.	
595	No encontrado.	

500 - Internal server error (Error interno del servidor)

CODE	MENSAJE (ES-PE)	DESCRIPCIÓN
166	Error interno del servidor.	La solicitud ha fallado debido a un error interno en el servidor.

5.2. Request URL Notificación JSON (notificación de estados del CIP)

La notificación de PagoEfectivo hacia el comercio afiliado es por medio de una petición web (WebRequest), hacia una ruta (UrlNotificación) que haya establecido el comercio afiliado. El método de la petición es de tipo POST.

Esta notificación se da en los siguientes casos:

- Pago de CIP.

La notificación está en formato JSON, donde:

Request Header:

Parámetro	Tipo	Descripción
PE-Signature	Cadena	Firma en formato HMAC SHA256 hexadecimal. El comercio deberá validar el contenido de la solicitud usando el valor de este parámetro.

Ejemplo Request Header:

PE-Signature: 20e525c086cf0ee3e28dc2e1b7412073b0ad747395841ea7d6324f2d7ee40a05

Request Body:

Nombre	Tipo de dato	Descripción y Estructura (Tipo de Dato o Estructura)
eventType	Cadena	Tipo de evento de notificación. Valores posibles: "cip.paid"
operationNumber	Cadena (0..14)	Número de Operación que genero el evento.
(*)data	<u>Objeto</u>	Objeto que contiene datos de la notificación

(*)data

Nombre	Tipo de dato	Descripción y Estructura (Tipo de Dato o Estructura)
cip	Cadena (1..14)	Identificador único del CIP
currency	Cadena (3)	Moneda. Codigo de moneda ISO 4217
amount	Numérico	Monto. Formato: N(18,2)

Ejemplo Request Body:

```
{
  "eventType" : "cip.paid",
  "operationNumber" : "1253",
  "data" : {
    "cip" : "11546901",
    "currency" : "PEN",
    "amount" : 22.05
  }
}
```

Para validar la integridad de los datos enviados al comercio se compara la firma enviada por PagoEfectivo y la generada por el comercio

```
PE-Signature = HMAC_SHA256("secretKey", "requestBody")
```

- **PE-Signature:** Firma enviada por PagoEfectivo en el header de la solicitud
- **secretKey:** Clave otorgada al comercio.
- **requestBody:** Contenido del body de la solicitud, es una cadena JSON sin indentación:

```
{"eventType":"cip.paid","operationNumber":"1253","data":{"cip":"11546901",  
,"currency":"PEN","amount":22.05}}.
```

La plataforma de PagoEfectivo espera que el portal responda estado 200 en la recepción de la notificación. De acontecer un error en la recepción, la plataforma intentará realizar un segundo intento.