

Guía práctica para la conexión del Usuario Final EST-BP

Índice

1.	OBJETIVO	4
2.	DEFINICIONES	4
3.	PLANILLA DE PRUEBAS	5
4.	LLAVES DE ACCESO AL API	7
5.	TOKEN	7
6.	MÉTODOS QUE CONFORMAN LA SOLUCIÓN	7
7.	API CARGA PLANILLA	8
➤	Parámetros de entrada	8
✓	Parámetros de salida	11
✓	Ejemplos	11
8.	API CONSULTA PREVALIDACIÓN DE PLANILLAS	14
✓	Parámetros de entrada	14
✓	Parámetro de salida	15
✓	Ejemplos	17
9.	API CONSULTA APLICACIÓN DE PAGOS DE PLANILLAS	20
✓	Parámetros de entrada	20
✓	Parámetro de salida	20
✓	Ejemplos	23
10.	API PREVALIDACIÓN PLANILLAS	25
✓	Parámetros de entrada	26
✓	Parámetros de salida	26

✓	Ejemplos	27
11.	API PAGO PLANILLAS	28
✓	Parámetros de entrada	28
✓	Parámetros de salida	28
✓	Ejemplos	29
12.	ATENCIÓN DE MENSAJES POR ERRORES DE VALIDACION	30

1. OBJETIVO

Desarrollar una guía detallada para el personal técnico de empresas que utilizan un ERP, proporcionando instrucciones claras sobre el uso de los API' s para el procesamiento de pagos de salarios mediante Transferencias Masivas con el Banco Popular, con el fin de facilitar una implementación eficiente y sin contratiempos.

Para realizar consultas sobre el desarrollo del API lo puede hacer directamente al correo electrónico TransferenciasMasivasatencionGestiones@bp.fi.cr.

2. DEFINICIONES

El API se conecta a través de internet y llaves que se entregarán al momento de la conexión, no se requiere un VPN ni router de conexión.

API (Application Programming Interface): Interfaz de programación que permite la comunicación y el intercambio de datos entre diferentes aplicaciones o sistemas.

ApiKey: Identificador único asignado a cada patrono, brindado por el BP. Esencial para acceder al API de TM, dicho identificador se encuentra codificado en base 64.

BP: Banco Popular.

Base 64: Sistema de codificación que convierte datos binarios en texto utilizando un conjunto de 64 caracteres específicos.

ClientId: Llave de seguridad proporcionada por el Banco, que se requiere para la generación del token, dicha llave se encuentra codificada en base 64.

ClientSecret: Llave de seguridad proporcionada por el Banco, necesaria para la conformación del token, dicha llave se encuentra codificada en base 64. Esta llave vence anualmente y se estará coordinando su actualización.

EST (Empresa de Soluciones Tecnológicas): Compañía especializada en ofrecer soluciones tecnológicas innovadoras para mejorar la eficiencia de los procesos empresariales.

ERP (Enterprise Resource Planning): Sistema integrado de gestión empresarial que automatiza y centraliza los procesos clave de una organización.

JSON(Java Script Object Notation): Formato de intercambio de datos ligero y fácil de leer.

Serializar: Es el proceso de convertir un objeto o una estructura de datos en un formato que pueda ser fácilmente almacenado o transmitido y luego reconstruido más tarde.

String: Cadena de datos.

Scope: El parámetro scope se usa para especificar qué recursos o acciones puede acceder o realizar una aplicación.

Grant_type: Define el tipo de flujo de autorización que se está utilizando para obtener un token de acceso.

GET: GET es uno de los métodos estándar que se utiliza para solicitar datos de un servidor. Este método es solo de lectura, lo que significa que no debe modificar ni afectar los datos del servidor. **TenantId:** Llave de seguridad proporcionada por el Banco, necesaria para crear el token, dicha llave se encuentra codificada en base 64.

TerminalId: Identificador único asignado a cada ERP, proporcionado por el Banco. Necesario para acceder al API de TM, dicho identificador se encuentra codificado en base 64.

Token: Cadena de caracteres utilizada para autenticar a un usuario, generada por el ERP con ClientSecret, ClientId, y TenantId. Necesario para acceder al API de TM.

TM (Transferencias Masivas): Proceso bancario para realizar transferencias de fondos de forma eficiente entre múltiples beneficiarios en una sola operación, Canal de pago.

URL: Para esta guía los URL que se muestran son ficticios y son únicamente para demostrar la existencia de un URL de conexión. Los mismos se brindarán posterior a la formalización de la conexión con el banco.

TLS (Transport Layer Security): La versión que se debe utilizar es la TLS 1.3.

3. PLANILLA DE PRUEBAS

Se proporciona a la empresa EST una planilla de pruebas que podrá ser utilizada para la simulación de pagos, bajo las siguientes condiciones:

- **Empresa y llaves de pruebas:**
El Banco proporcionará una empresa genérica, llaveTerminalId, llave APIKey, así como los datos para la generación del Token, todos los datos suministrados en módulo de pruebas serán de carácter genéricos y serán entregados una vez firmada el Acta indicada por el banco.
- **Monto de los pagos**
Los pagos realizados durante las pruebas no deberán superar los ₡10 por colaborador. Esto con el objetivo de no afectar las cuentas de prueba del sistema.
- **Datos incluidos:**
No se permitirá incluir nuevas cédulas, nombres ni cuentas bancarias. Solo se deben utilizar los datos previamente establecidos en la planilla de prueba.
- **Usuario autorizador:**
El Banco suministrará un usuario de pruebas, quien será el único autorizado para ingresar al sistema y ejecutar los pagos. Este usuario será invariable y exclusivo para el entorno de pruebas.
- **Formato de la planilla:**
No se deben realizar modificaciones al formato original de la planilla suministrada.
- **Estructura del archivo:**

La planilla contiene 1 línea correspondiente a los datos del patrono de pruebas, y 10 líneas correspondientes a colaboradores.

Consideraciones

- Los datos y URL utilizados en el entorno de pruebas son ficticios y no afectan en ningún momento las cuentas reales o sistemas del Banco. Por esta razón, las cuentas utilizadas en las pruebas no necesariamente corresponden a cuentas reales en producción.
- El banco se reserva el derecho de disponer en cualquier momento de los datos de prueba enviados a los EST, lo que incluye la posibilidad de modificarlos o eliminarlos sin previo aviso.
- **El logotipo del Banco Popular es de uso exclusivo y especial resguardo legal; su uso por terceros no autorizados está prohibido y puede conllevar acciones legales.**

3.1. FORMATO DE LA PLANILLA

3.1.1. PATRONO Y COLABORADORES

El patrono será representado en una fila dentro de la planilla de pruebas. Asimismo, cada colaborador se representará en una fila adicional de la misma planilla, la cual deberá contener la siguiente información en columnas:

Nombre completo	Cuenta Cliente	Cuenta IBAN	CRR.BILLING.CUSTOMER	Cédula	Monto	Descripción
-----------------	----------------	-------------	----------------------	--------	-------	-------------

- Nombre completo.
- Cuenta Cliente.
- Cuenta IBAN.
- CRR.BILLING.CUSTOMER.
- Cédula.
- Monto.
- Descripción.

3.2. USUARIO AUTORIZADOR

El siguiente usuario autorizador será exclusivo para realizar las pruebas de conexión entre la empresa EST y el Banco. Este usuario cuenta con la información necesaria y válida para ejecutar los pagos de prueba, por lo tanto:

- No se permitirá realizar modificaciones a la información de este usuario.
- Únicamente el Banco estará autorizado para efectuar cualquier cambio relacionado, sin obligación de notificar previamente a la empresa EST.
- Todos los datos usados en pruebas son ficticios y no afectan cuenta en producción del Banco, por lo cual no necesariamente las cuentas son reales en producción.

Datos del Usuario Autorizador:

- ✓ **Los datos de acceso se proporcionarán una vez formalizado el servicio entre las partes.**

4. LLAVES DE ACCESO AL API

Las llaves de acceso serán proporcionadas por el banco y podrán ser modificadas sin previo aviso. Estas llaves serán utilizadas exclusivamente para fines de prueba y no serán válidas para el ingreso en ambiente de producción por parte de la EST

- ✓ **ApiKey:** 00
- ✓ **TerminalId:** 00

5. TOKEN

Los datos de acceso se proporcionarán una vez formalizado el servicio entre las partes.

6. MÉTODOS QUE CONFORMAN LA SOLUCIÓN

Todos los datos de seguridad que regulan el acceso a los API' s de TM (como ApiKey, TerminalId, ClientSecret, ClientId y TenantId) son enviados por correo electrónico y se encuentran codificados en Base64. Para su posterior uso en los métodos de la solución, estos datos deben ser decodificados de Base64 a formato string. Es importante destacar que no se deben enviar los datos hacia los API' s de TM en base 64.

Para la generación del token, es necesario utilizar el **ClientSecret**, **ClientId** y **TenantId** (todos ellos decodificados a formato string).

Para obtener el token, realiza lo siguiente:

- a) **URL de la solicitud del token:**
 - https://login.microsoftonline.com/tenant_id/oauth2/v2.0/token.
 - Donde debe reemplazar *tenant_id* por la llave **tenantId** suministrada.
- b) **Parámetro scope conformado en el siguiente formato:**
 - [api://client_id/default](#).
 - Donde debe reemplazar *client_id* por la llave **clientId** suministrada.
- c) **Parámetro grant_type:**
 - Agregar el parámetro grant_type con el valor "**client_credentials**" (sin comillas).
- d) **Parámetro client_id:**
 - Agregar el parámetro client_id con el valor de **clientId** suministrado.
- e) **Parámetro client_secret:**
 - Agregar el parámetro client_secret con el valor del **clientSecret** suministrado.
- f) **Realizar la solicitud:**
 - Realiza un **GET** a la URL indicada con los parámetros mencionados.

7. API CARGA PLANILLA

Este método se encarga de cargar los datos de las planillas a procesar en el sistema de Transferencias Masivas (Tanto para aplicar una prevalidación como para aplicar un pago de planilla), el JSON enviado no deben ser mayores a 100 KB, esto implica que la planilla se debe dividir en lotes de un tamaño no mayor a lo indicado.

- **Lotes (chunks):**
 - Cada lote debe tener un tamaño máximo de 100 KB.
 - Los lotes no pueden enviarse por separado; deben incluirse dentro de un único archivo de planilla.
 - Todos los lotes de una misma planilla deben compartir el mismo FileId, ya que forman parte del mismo archivo.
 - El parámetro Index indica el número de lote enviado.
 - El campo Data es el contenido del lote en formato base64.
- **Método de tipo:** Post.
 - ✓ **URL API pruebas:** <https://pagoplanillas.com/carga-planilla>
- **Parámetros de entrada**

Parámetros	Tipo	Query params /Request Body	Descripción
TerminalId	Guid	Query params	Identificador único asignado a cada sistema ERP, proporcionado por el BP
APIKey	Guid	Query params	Identificador único asignado a cada patrono, proporcionado por el BP
Token	Bearer token	Token	Token generado a partir del Client ID, Client Secret y Tenant ID proporcionados por el BP
Nombre	String	Query params	Nombre del usuario registrado en el sistema ERP que realiza la consulta.
Identificación	String	Query params	Identificación del usuario registrado en el sistema ERP que lleva a cabo la consulta.
FileId	Guid	Request body	Identificador de la planilla (asignado por el sistema que envía la información). Todos los lotes pertenecientes a una misma planilla deben poseer el mismo FileId, su uso es únicamente para identificar los lotes pertenecientes a una planilla en particular
Index	Int	Request body	Número de lote que se está enviando, es un consecutivo único para cada uno de los lotes pertenecientes a una planilla
Data	String	Request Body	Contiene el lote correspondiente al JSON de la planilla en un arreglo de bytes (al serializar se envía en un formato base 64)

El JSON que conforma la planilla y que debe ser dividido en lotes que se deben colocar en Data debe contener los siguientes campos

Parámetros	Tipo	Query params /Request Body	Descripción
EncabezadoTransaccion	Object	Request Body	Información del encabezado relacionada con la planilla.
Moneda	String	Request Body	Moneda en la que se realizarán los pagos de salarios
			Posibles valores: CRC

Parámetros	Tipo	Query params /Request Body	Descripción
			USD
PagoProgramado	Boolean	Request Body	Indica si el pago es programado. Posibles valores: True False
FechaAplicacionPago	String	Request Body	Fecha en la que se debe realizar el pago programado, válida solo si PagoProgramado es True.
PagoPatrono	Object	Request Body	Datos del patrono.
Nombre	String	Request Body	Nombre del patrono.
Identificación	String	Request Body	Número de cédula del patrono.
CuentaIBAN	String	Request Body	Número de cuenta IBAN del patrono, donde se realizará el débito de los pagos de salario.
MontoADebitar	Decimal	Request Body	Monto total que se debitará por el pago de salarios.
Descripcion	String	Request Body	Descripción del débito a aplicar.
DetalleColaboradores	Array	Request Body	Listado de datos de los colaboradores a los que se les aplicará el crédito del pago de salario.
Nombre	String	Request Body	Nombre del colaborador.
Identificación	String	Request Body	Número de identificación del colaborador.
CuentaIBAN	String	Request Body	Número de cuenta IBAN del colaborador, donde se realizará el crédito del pago de salario.
MontoAAcreditar	Decimal	Request Body	Monto que se acreditará al colaborador por el pago de salario.
Descripcion	String	Request Body	Descripción del crédito a aplicar.

✓ **Parámetros de salida**

Parámetros	Tipo	Descripción
StatusCode	Int	Código que indica el estado de la consulta. Posibles valores 200: Exitoso 400: Solicitud incorrecta 500: Error interno
Traceld	Guid	Identificador interno único asociado a una trama o secuencia de eventos relacionada con la consulta en la API.
JSON	Object	Contiene la respuesta proporcionada por la API.
IssuedAt	Datetime	Fecha en que se obtuvo la información.

✓ **Ejemplos**

✓ **Ejemplo función generación de lotes a partir de un JSON**

```
public static List<byte[]> SplitJsonIntoChunks(string json, int chunkSize)
{
    // Convertir el JSON a un arreglo de bytes

    var jsonBytes = System.Text.Encoding.UTF8.GetBytes(json);

    var chunks = new List<byte[]>();

    // Dividir el arreglo de bytes en chunks

    for (int i = 0; i < jsonBytes.Length; i += chunkSize)
    {
        var size = Math.Min(chunkSize, jsonBytes.Length - i);
```

```
var chunk = new byte[size];

Array.Copy(jsonBytes, i, chunk, 0, size);

chunks.Add(chunk);

}

return chunks;

}
```

✓ Ejemplo request

https://pagoplanillas.com/carga-planilla?TerminalId=00000000-0000-0000-0000-000000000000&APIKey=00000000-000-0000-0000-000000000000&Nombre=Usuario Registrado&Identificacion=000000000

```
{  
  
    "FileId": "00000000-000-0000-0000-000000000000",  
  
    "Index": 0,  
  
    "Data": "eyJFbmNhYmV6YWVrVHJhbnNhY2Npb24iOnsiTW9uZWRhIjoIQ1JDliwiUGFnbiB1by2dyYW1hZG8  
iOmZhbnHNILCjGZWNoYUFWbGljYWNPb25QYWwvMjAyNS0wMS0yNCAnjowMCJ9LCJQYWdvUGF0cm9ubyl6e  
yJOE21icmUiOiJGVUeEQUNJT04gTVVKRdG8gcGFnbzBzYXhcmVln0seyJOE21icmUiOiJaT0JFSURBIE1PWUEg  
TEFDQVIPlwiSWRlbnRpZmljYWNPb24iOil2MDE4MDA0OTMiLCJEdWVudGaJQkFOljoIQ1lxODAxNjExMTA4NDZz  
NZUxMdUzNSIs"  
  
}
```

- ✓ **Ejemplo del JSON** del pago de la planilla

```
{
  "EncabezadoTransaccion": {
    "Moneda": "CRC",
    "PagoProgramado": true,
    "FechaAplicacionPago": "2022-12-15"
  },
  "PagoPatrono": {
    "Nombre": "Nombre patrono",
    "Identificacion":
      "0000000000",

```

<pre> "CuentaIBAN": "CR11116111111111111111", "MontoADebitar": 20.50, "Descripcion": "Pago de salarios" }, "DetalleColaboradores": [{ "Nombre": "Nombre Colaborador", "Identificacion": "0000000000", "CuentaIBAN": " CR11116111111111111111", "MontoAAcreditar": 10, "Descripcion": "Pago sal" },] </pre>
<pre> "Nombre": "Nombre Colaborador", "Identificacion": "0000000000", "CuentaIBAN": " CR11116111111111111111", "MontoAAcreditar": 10, "Descripcion": "Pago sal" }] </pre>

✓ **Ejemplo Response**

<pre> { "StatusCode": 200, "Traceld": "026d88f-f0e8-48c3-9b46-51203a184f85", "JSON": " e7a6136f4c774f2797dfe6ea1bff9bf8 cargado", "IssuedAt": "2024-03-06T13:27:55.5771975-06:00" } </pre>
--

✓ **Ejemplo Response**

<pre> { "StatusCode": 400, "Traceld": "e7a6136f4c774f2797dfe6ea1bff9bf8", "JSON": "El usuario autenticado no se encuentra autorizado", "IssuedAt": "2024-03-06T13:27:55.5771975-06:00" } </pre>

✓ **Ejemplo Response**

<pre> { "StatusCode": 400, "Traceld": "9ae2fc4de31847bc974b575411b6767e", "JSON": "El parametro ApiKey no tiene el formato esperado", "IssuedAt": "2024-03-06T13:30:38.61598-06:00" } </pre>
--

```
} }
```

8. API CONSULTA PREVALIDACIÓN DE PLANILLAS

Este método se encarga de realizar consultas sobre el estado actual de las planillas, así como de cada uno de los movimientos que han sido enviados para su prevalidación. La respuesta proporcionada por este API está en formato JSON, lo que permite una fácil integración y manejo de la información.

- ✓ **Método de tipo:** Get.
- ✓ **URL API pruebas:** <https://pagoplanillas.com/prevalidacion-planilla>
- ✓ **Parámetros de entrada**

Parámetros	Tipo	Query params /Request Body	Descripción
TerminalId	Guid	Query params	Identificador único asignado a cada sistema ERP, proporcionado por el BP
APIKey	Guid	Query params	Identificador único asignado a cada patrono, proporcionado por el BP
Token	Bearer token	Token	Token generado a partir del Client ID, Client Secret y Tenant ID proporcionados por el BP
Nombre	String	Query params	Nombre del usuario registrado en el sistema ERP que realiza la consulta.

Parámetros	Tipo	Query params /Request Body	Descripción
Identificación	String	Query params	Identificación del usuario registrado en el sistema ERP que lleva a cabo la consulta.
Id	Guid	Query params	Identificador de la planilla que ha sido enviada para prevalidación y que se desea consultar.
IndicePagina	Int	Query params	Número de página que se desea consultar en la respuesta.
CantidadRegistros	Int	Query params	Número de registros por página a obtener en la consulta (máximo 100).

✓ **Parámetro de salida**

Parámetros	Tipo	Descripción
StatusCode	Int	Código que indica el estado de la consulta. Posibles valores: 200: Exitoso 400: Solicitud incorrecta 404: Registro no encontrado 500: Error interno
TraceId	String	Identificador interno único asociado a la secuencia de eventos relacionada con la consulta en la API.
JSON	String	Contiene la respuesta en formato JSON proporcionada por el método de la API.
DatosPagoSalario	Object	Objeto retornado en el StatusCode 200, que contiene la información relacionada con la planilla enviada a prevalidar. Se encuentra dentro del parámetro JSON.
Id	Guid	Identificador de la planilla enviada a prevalidar, contenido en el parámetro DatosPagoSalario.
DatosPatrono	Object	Objeto que contiene información relacionada con el patrono, incluyendo identificación, nombre y número de cuenta. Se encuentra en el parámetro DatosPagoSalario.
Moneda	String	Moneda en la que se solicita realizar la transacción, contenida en el parámetro DatosPagoSalario. Posibles valores: CRC USD
Estado	String	Estado de la prevalidación de la planilla, contenido en el parámetro DatosPagoSalario. Posibles valores: En proceso de aplicación. Satisfactoria. Fallido.

Parámetros	Tipo	Descripción
		Satisfactoria con inconsistencia operativa. Satisfactoria con inconsistencia técnica. Inconsistencia técnica
ListaDatosTransaccion	Array	Listado de los movimientos (débitos y créditos) relacionados con la planilla, contenido en el parámetro JSON.
FTAplicacion	String	Identificador de la aplicación del débito o crédito; en la prevalidación, su valor siempre será null.
DescripcionPago	String	Descripción de la transacción
MontoMovimiento	Decimal	Monto de la transacción
TipoMovimiento	String	Indica el tipo de movimiento. Posibles valores: Débito Crédito
Estado	String	Estado en el que se encuentra el movimiento enviado a prevalidar. Sus posibles valores son: En proceso de aplicación. Aplicado. Inconsistente. Fallido.
FechaHora	DateTime	Fecha en que se aplicó el movimiento.
MensajeValidacion	List<String>	Lista de mensajes de validación relacionados con un movimiento.
DatosCliente	Object	Objeto que contiene información relacionada con los datos del cliente, como identificación, nombre y número de cuenta.
PaginasDisponibles	Int	Cantidad de páginas disponibles para consultar, dependiendo de la cantidad de registros solicitados y la cantidad total de registros disponibles.

Parámetros	Tipo	Descripción
TotalTransaccionesPagina	Int	Número de transacciones en la página mostrada para una planilla en particular.
TotalTransaccionesDisponibles	Int	Cantidad total de transacciones disponibles para consultar en una planilla en particular.
IssuedAt	DateTime	Fecha en que se obtuvo la información.

✓ **Ejemplos**

✓ **Ejemplo Request**

<https://pagoplanillas.com/prevalidacion-planilla?TerminalId=00000000-0000-0000-0000-000000000000&APIKey=00000000-000-0000-0000-000000000000&Nombre=Nombre usuario&Identificacion=000000000000&Id=0000000-0000-0000-0000-000000000000&IndicePagina=1&CantidadRegistros=100>

✓ **Ejemplo Response**

```
{
  "StatusCode": 200,
  "TraceId": "c676692555c74bb4be36b6e95158ff4d",
  "JSON": {
    "DatosPagoSalario": {
      "Id": "00000000-000-0000-0000-000000000000",
      "DatosPatrono": {
        "Identificacion": "000000000000",
        "Nombre": "Nombre patrono",
        "Cuenta": "CR11116111111111111111"
      },
    },
    "Moneda": "USD",
    "Estado": "Satisfactoria con inconsistencia operativa"
  },
  "ListaDatosTransaccion": [
    {
      "FTAplicacion": null,
      "DescripcionPago": "Pago sal",
      "MontoMovimiento": 1000.00,
      "TipoMovimiento": "Credito",
    }
  ]
}
```

```
"Estado": "Fallido",
"FechaHora": "2023-12-13T20:43:54.15",
"MensajeValidacion": [
    "Cod:VAL0022 - La moneda de la cuenta del patrono y la transacción no coinciden"
```

```
],
"DatosCliente": {
    "Identificacion": "0000000000",
    "Nombre": "Nombre del cliente",
```

```

    "Cuenta": "CR11116111111111111111"
},
{
    "FTAplicacion": null,
    "DescripcionPago": "Pago sal",
    "MontoMovimiento": 10.00,
    "TipoMovimiento": "Credito",
    "Estado": "Aplicado",
    "FechaHora": "2023-12-13T20:44:38.65",
    "MensajeValidacion": [],
    "DatosCliente": {
        "Identificacion": "0000000000",
        "Nombre": "Nombre del cliente",
        "Cuenta": "CR11116111111111111111"
    },
    "FTAplicacion": null,
    "DescripcionPago": "Pago de salarios",
    "MontoMovimiento": 20.00,
    "TipoMovimiento": "Debito",
    "Estado": "Aplicado",
    "FechaHora": "2023-12-13T20:43:14.027",
    "MensajeValidacion": [],
    "DatosCliente": {
        "Identificacion": "0000000000",
        "Nombre": "Nombre patrono",
        "Cuenta": "CR11116111111111111111"
```

```

    }
  }
],
  "PaginasDisponibles": 1,
  "TotalTransaccionesPagina": 3,
  "TotalTransaccionesDisponibles": 3
},
  "IssuedAt": "2023-12-14T09:03:04.7655187-06:00"
}

```

✓ **Ejemplo Response**

```

{
  "StatusCode": 400,
  "Traceld": "e7a6136f4c774f2797dfe6ea1bff9bf8",
  "JSON": "El usuario autenticado no se encuentra autorizado",
  "IssuedAt": "2024-03-06T13:27:55.5771975-06:00"
}

```

✓ **Ejemplo Response**

```

{
  "StatusCode": 404,
  "Traceld": "1c17183b301c40d4816063f156524506",
  "JSON": "No se encuentran planillas registradas con la identificación proporcionada.",
  "IssuedAt": "2024-03-06T13:29:55.1617968-06:00"
}

```

✓ **Ejemplo Response**

```

{
  "StatusCode": 400,
  "Traceld": "9ae2fc4de31847bc974b575411b6767e",
  "JSON": "El parametro ApiKey no tiene el formato esperado",
  "IssuedAt": "2024-03-06T13:30:38.61598-06:00"
}

```

9. API CONSULTA APLICACIÓN DE PAGOS DE PLANILLAS

Este método permite realizar consultas sobre el estado actual de las planillas, así como sobre cada uno de los movimientos que han sido enviados para su respectiva aplicación. La respuesta proporcionada por este API está en formato JSON, lo que facilita su interpretación y manejo.

- ✓ **Método de tipo:** Get.
- ✓ **URL API pruebas:** <https://pago-planillas.com/pago-planilla> **Parámetros de entrada**

Parámetros	Tipo	Query params /Request Body	Descripción
TerminalId	Guid	Query params	Identificador único asignado a cada sistema ERP, proporcionado por el BP.
APIKey	Guid	Query params	Identificador único asignado a cada patrono, proporcionado por el BP.
Token	Bearer token	token	Token generado a partir del Client ID, Client Secret y Tenant ID provistos por el BP.
Nombre	String	Query params	Nombre del usuario registrado en el sistema ERP que realiza la consulta.
Identificación	String	Query params	Identificación del usuario registrado en el sistema ERP que lleva a cabo la consulta.
Id	Guid	Query params	Identificador de la planilla enviada a pagar previamente y que se desea consultar.
IndicePagina	Int	Query params	Número de página que se desea consultar.
CantidadRegistros	Int	Query params	Cantidad de registros por página que se desea obtener (máximo 100).

- ✓ **Parámetro de salida**

Parámetros	Tipo	Descripción
StatusCode	Int	Código que indica el estado de la consulta. Posibles valores: 200: Exitoso 400: Solicitud incorrecta

Parámetros	Tipo	Descripción
		404: Registro no encontrado 500: Error interno
TraceId	String	Identificador interno único asociado a la secuencia de eventos relacionada con la consulta en la API.
JSON	String	Contiene la respuesta en formato JSON proporcionada por el método de la API.
DatosPagoSalario	Object	Objeto retornado en el StatusCode 200, que contiene la información relacionada con la planilla enviada a pagar. Se encuentra dentro del parámetro JSON.
Id	Guid	Identificador de la planilla enviada a pagar, contenido en el parámetro DatosPagoSalario.
DatosPatrono	Object	Objeto que contiene información relacionada con el patrono, como identificación, nombre y número de cuenta. Se encuentra en el parámetro DatosPagoSalario.
Moneda	String	Moneda en la que se solicita realizar la transacción, contenida en el parámetro DatosPagoSalario. Posibles valores: CRC
		USD
Estado	String	Estado del pago de la planilla, contenido en el parámetro DatosPagoSalario. Posibles valores: En proceso de aplicación. Satisfactoria. Fallido. Satisfactoria con inconsistencia operativa.
		Satisfactoria con inconsistencia técnica.
		Inconsistencia técnica

Parámetros	Tipo	Descripción
ListaDatosTransaccion	Array	Listado de los movimientos (débitos y créditos) relacionados con la planilla, contenido en el parámetro JSON.
FTAplicacion	String	Identificador de la aplicación del débito o crédito.
DescripcionPago	String	Descripción de la transacción.
MontoMovimiento	Decimal	Monto de la transacción
TipoMovimiento	String	Indica el tipo de movimiento. Posibles valores: Débito Crédito
Estado	String	Estado del movimiento enviado a pagar. Posibles valores: En proceso de aplicación. Aplicado. Inconsistente. Fallido.
FechaHora	DateTime	Fecha que se aplicó el movimiento.
MensajeValidacion	List<String>	Lista de mensajes de validación relacionados con un movimiento.
DatosCliente	Object	Objeto que contiene información relacionada con los datos del cliente, como identificación, nombre y número de cuenta.
PaginasDisponibles	Int	Cantidad de páginas disponibles para consultar, dependiendo de la cantidad de registros solicitados y la cantidad total de registros disponibles.
TotalTransaccionesPagina	Int	Número de transacciones en la página mostrada para una planilla en particular.

Parámetros	Tipo	Descripción
TotalTransaccionesDisponibles	Int	Número de transacciones en la página mostrada para una planilla en particular.
IssuedAt	DateTime	Fecha en que se obtuvo la información.

✓ **Ejemplos**

✓ **Ejemplo Request**

<https://pago-planillas.com/pago-planilla?TerminalId=00000000-0000-0000-0000-000000000000&APIKey=00000000-000-0000-0000-000000000000&Nombre=Nombre usuario&Identificacion=0000000000&Id=00000000-0000-0000-0000-000000000000&IndicePagina=1&CantidadRegistros=100>

✓ **Ejemplo Response**

<pre>{ "StatusCode": 200, "TraceId": "c676692555c74bb4be36b6e95158ff4d", "JSON": { "DatosPagoSalario": { "Id": "00000000-000-0000-0000-000000000000", "DatosPatrono": { "Identificacion": "0000000000", "Nombre": "Nombre patrono", "Cuenta": "CR11116111111111111111" }, }, "Moneda": "USD", "Estado": "Satisfactoria con inconsistencia operativa" }, "ListaDatosTransaccion": [{ "FTAplicacion": null, "DescripcionPago": "Pago sal", "MontoMovimiento": 1000.00, "TipoMovimiento": "Credito", }] }</pre>
<pre>"Estado": "Fallido", "FechaHora": "2023-12-13T20:43:54.15", "MensajeValidacion": [</pre>

"Cod:VAL0022 - La moneda de la cuenta del patrono y la transacción no coinciden"], "DatosCliente": { "Identificacion": "0000000000", "Nombre": "Nombre del cliente", "Cuenta": "CR11116111111111111111"	
} }, { "FTAplicacion": FTHDj45, "DescripcionPago": "Pago sal", "MontoMovimiento": 10.00, "TipoMovimiento": "Credito", "Estado": "Aplicado", "FechaHora": "2023-12-13T20:44:38.65", "MensajeValidacion": [], "DatosCliente": { "Identificacion": "0000000000", "Nombre": " Nombre del cliente ", "Cuenta": " CR1111611111111111111111" }	
} }, { "FTAplicacion": FTHDj46, "DescripcionPago": "Pago de salarios", "MontoMovimiento": 20.00, "TipoMovimiento": "Debito", "Estado": "Aplicado", "FechaHora": "2023-12-13T20:43:14.027", "MensajeValidacion": [], "DatosCliente": { "Identificacion": "0000000000", "Nombre": "Nombre patrono", "Cuenta": " CR1111611111111111111111" }	
}], "PaginasDisponibles": 1,	

```
{
  "TotalTransaccionesPagina": 3,
  "TotalTransaccionesDisponibles": 3
},
"IssuedAt": "2023-12-14T09:03:04.7655187-06:00"
}
```

✓ **Ejemplo Response**

```
{
  "StatusCode": 400,
  "Traceld": "e7a6136f4c774f2797dfe6ea1bff9bf8",
  "JSON": "El usuario autenticado no se encuentra autorizado",
  "IssuedAt": "2024-03-06T13:27:55.5771975-06:00"
}
```

✓ **Ejemplo Response**

```
{
  "StatusCode": 404,
  "Traceld": "1c17183b301c40d4816063f156524506",
  "JSON": "No se encuentran planillas registradas con la identificación proporcionada.",
  "IssuedAt": "2024-03-06T13:29:55.1617968-06:00"
}
```

✓ **Ejemplo Response**

```
{
  "StatusCode": 400,
  "Traceld": "9ae2fc4de31847bc974b575411b6767e",
  "JSON": "El parametro ApiKey no tiene el formato esperado",
  "IssuedAt": "2024-03-06T13:30:38.61598-06:00"
}
```

10. API PREVALIDACIÓN PLANILLAS

Este método se encarga de llevar a cabo la prevalidación de los datos correspondientes a cada una de las transacciones de una planilla.

✓ **Método de tipo:** Post.

✓ **URL API pruebas:** <https://pago-planillas.com/prevalidacion-planilla>

✓ **Parámetros de entrada**

Parámetros	Tipo	Query params /Request Body	Descripción
TerminalId	Guid	Query params	Identificador único del sistema ERP, proporcionado por el BP.
APIKey	Guid	Query params	Identificador único del patrono, también proporcionado por el BP.
Nombre	String	Query params	Nombre del usuario registrado en el sistema que realiza la prevalidación.
Identificación	String	Query params	Número de identificación del usuario registrado en el sistema que realiza la prevalidación.
Token	Bearer token		Token generado a partir del Client ID, Client Secret y TenantId proporcionados por el BP.
FileId	Guid	Request Body	Identificador de la planilla (asignado por el sistema que envía la información). Debe ser el mismo Id usado para identificar los lotes en la carga de planillas, su uso es únicamente para identificar los lotes pertenecientes a una planilla en particular

✓ **Parámetros de salida**

Parámetros	Tipo	Descripción
StatusCode	Int	Código que indica el estado de la consulta: 200: Exitoso 400: Solicitud incorrecta 500: Error interno
TraceId	String	Identificador interno único asociado a una trama o secuencia de eventos relacionada con la consulta en la API.

Parámetros	Tipo	Descripción
JSON	String	Contiene la respuesta proporcionada por la API. Cuando el StatusCode es 200, incluirá un identificador único de la planilla enviada para prevalidación.
IssuedAt	Datetime	Fecha en que se obtuvo la información.

✓ **Ejemplos**

✓ **Ejemplo Request**

<https://pago-planillas.com/prevalidacion-planilla?TerminalId=00000000-0000-0000-0000-000000000000&APIKey=00000000-000-0000-0000-000000000000&Nombre=UsuarioRegistrado&Identificacion=0000000000>

```
{
  "FileId": "00000000-000-0000-0000-000000000000",
}
```

✓ **Ejemplo Response**

```
{
  "StatusCode": 200,
  "Traceld": "6479cabcdfd34f7a994530d71e8193d4",
  "JSON": "61C803F6-D9F5-406D-A936-97F432BCF9F4",
  "IssuedAt": "2023-12-15T10:47:11.7205674-06:00"
}
```

✓ **Ejemplo Response**

```
{
  "StatusCode": 400,
  "Traceld": "e7a6136f4c774f2797dfe6ea1bff9bf8",
  "JSON": "El usuario autenticado no se encuentra autorizado",
  "IssuedAt": "2024-03-06T13:27:55.5771975-06:00"
}
```

✓ **Ejemplo Response**

```
{
  "StatusCode": 400,
  "Traceld": "9ae2fc4de31847bc974b575411b6767e",
}
```

```
"JSON": "El parametro ApiKey no tiene el formato esperado",
"IssuedAt": "2024-03-06T13:30:38.61598-06:00"
}
```

11. API PAGO PLANILLAS

Este método se encarga de aplicar los débitos y créditos asociados a las transacciones de una planilla.

- ✓ **Método de tipo:** Post.
- ✓ **URL API pruebas:** <https://pago-planillas.com/pago-planilla>
- ✓ **Parámetros de entrada**

Parámetros	Tipo	Query params /Request Body	Descripción
TerminalId	Guid	Query params	Identificador único del sistema ERP, proporcionado por el BP.
ApiKey	Guid	Query params	Identificador único del patrono, también proporcionado por el BP.
Token	Bearer token		Token generado a partir del Client ID, Client Secret y TenantId proporcionados por el BP.
FileId	Guid	Request Body	Identificador de la planilla (asignado por el sistema que envía la información). Debe ser el mismo Id usado para identificar los lotes en la carga de planillas, su uso es únicamente para identificar los lotes pertenecientes a una planilla en particular

- ✓ **Parámetros de salida**

Parámetros	Tipo	Descripción
StatusCode	Int	Código que indica el estado de la consulta.
		Posibles valores
		200: Exitoso
		400: Solicitud incorrecta
		500: Error interno

Parámetros	Tipo	Descripción
Traceld	Guid	Identificador interno único asociado a una trama o secuencia de eventos relacionada con la consulta en la API.
JSON	Guid	Contiene la respuesta proporcionada por la API. Cuando el StatusCode es 200, incluirá un identificador único de la planilla enviada para pagar.
IssuedAt	Datetime	Fecha en que se obtuvo la información.

✓ **Ejemplos**

✓ **Ejemplo Request**

<https://pago-planillas.com/pago-planilla?TerminalId=00000000-0000-0000-0000-000000000000&APIKey=00000000-000-0000-0000-000000000000&Nombre=UsuarioRegistrado&Identificacion=0000000000>

```
{
  "FileId": "0000000-000-0000-0000-000000000000",
}
```

✓ **Ejemplo Response**

```
{
  "StatusCode": 200,
  "Traceld": "6479cabcd9d34f7a994530d71e8193d4",
  "JSON": "61C803F6-D9F5-406D-A936-97F432BCF9F4",
  "IssuedAt": "2023-12-15T10:47:11.7205674-06:00"
}
```

✓ **Ejemplo Response**

```
{
  "StatusCode": 400,
  "Traceld": "e7a6136f4c774f2797dfe6ea1bff9bf8",
  "JSON": "El usuario autenticado no se encuentra autorizado",
  "IssuedAt": "2024-03-06T13:27:55.5771975-06:00"
}
```

✓ **Ejemplo Response**

```
{
```

```
"StatusCode": 400,
"TraceId": "9ae2fc4de31847bc974b575411b6767e",
"JSON": "El parametro ApiKey no tiene el formato esperado",
"IssuedAt": "2024-03-06T13:30:38.61598-06:00"
}
```

Horario de operación del canal de pago.

Horarios de operación

Día	Hora inicio	Hora fin	Sistema disponible ⓘ
Lunes	6:00 a.m. 	10:00 p.m. 	☒
Martes	6:00 a.m. 	10:00 p.m. 	☒
Miércoles	6:00 a.m. 	10:00 p.m. 	☒
Jueves	6:00 a.m. 	10:00 p.m. 	☒
Viernes	6:00 a.m. 	10:00 p.m. 	☒
Sábado	6:00 a.m. 	10:00 p.m. 	☒
Domingo	6:00 a.m. 	10:00 p.m. 	☒

12. ATENCIÓN DE MENSAJES POR ERRORES DE VALIDACION

Los mensajes de error son los mismos para las prevalidaciones como para los pagos finales de planilla. El cuadro está dividido en los mensajes que describen el error en el API y los mensajes que retornan al usuario final, éstos últimos pueden variar.

Código	Descripción del error en el API	Mensaje que retorna, el cual puede variar
VAL0000	Mensaje retornado cuando se ha presentado un error inesperado.	Ocurrió un error inesperado
VAL0001	Mensaje retornado cuando la fecha de aplicación sobrepasa el máximo de días de programación.	El rango máximo de días para programar fue excedido.
VAL0002	Mensaje retornado cuando el identificador de la empresa proveedora de software no se encuentra registrado.	El identificador de la empresa proveedora de software no está registrado.
VAL0003	Mensaje retornado cuando se quiere hacer un pago en una moneda no permitida.	La transacción únicamente debe realizarse en colones (CRC) o dólares (USD).
VAL0004	Mensaje retornado cuando la transacción se encuentra en proceso.	La transacción está actualmente en proceso.
VAL0005	Mensaje retornado cuando el monto total de la planilla es diferente a la sumatoria de todos los pagos a los colaboradores.	El monto total de la planilla no coincide con la sumatoria de todos los pagos a los colaboradores.
VAL0006	Mensaje retornado cuando la identificación del patrono no coincide con los formatos correctos.	La identificación del patrono no se ajusta a los formatos establecidos: 09 dígitos para personas físicas, 12 dígitos para personas físicas residentes y 10 dígitos para personas jurídicas.
VAL0007	Mensaje retornado cuando el IBAN del patrono no coincide con el formato correcto.	La cuenta IBAN del patrono no presenta el formato correcto.
VAL0008	Mensaje retornado cuando el IBAN no contiene en el código del país las letras CR.	La cuenta IBAN no presenta el formato correcto.
VAL0009	Mensaje retornado cuando el IBAN no es del Banco Popular.	La cuenta IBAN no pertenece al Banco Popular.
VAL0010	Mensaje retornado cuando la identificación de la empresa proveedora de software no coincide con los formatos correctos.	La identificación de la empresa proveedora de software no se ajusta a los formatos establecidos: 09 dígitos para personas físicas, 12 dígitos para personas físicas residentes y 10 dígitos para personas jurídicas.
VAL0011	Mensaje retornado cuando el monto a debitar es menor a cero.	El monto a debitar es menor o igual a cero.
VAL0012	Mensaje retornado cuando la identificación del colaborador no coincide con los formatos correctos.	La identificación del colaborador no se ajusta a los formatos establecidos: 09 dígitos para personas físicas, 12 dígitos para personas físicas residentes y 10 dígitos para personas jurídicas.
VAL0013	Mensaje retornado cuando el IBAN del colaborador no coincide con el formato correcto.	La cuenta IBAN del colaborador no presenta el formato correcto.
VAL0014	Mensaje retornado cuando la cuenta iban no contiene en el código del país las letras CR.	La cuenta IBAN del colaborador no presenta el formato correcto.
VAL0015	Mensaje retornado cuando el IBAN del colaborador no es del Banco Popular.	La cuenta IBAN del colaborador no es del Banco Popular.
VAL0016	Mensaje retornado cuando el monto a acreditar a un colaborador es menor o igual a cero.	El monto a acreditar al colaborador es menor o igual a cero.
VAL0017	Mensaje retornado cuando las transacciones mensuales por colaborador sobrepasan el máximo de transacciones mensuales.	La cantidad de transacciones mensuales por colaborador supera el límite de transacciones máximas permitido.

VAL0018	Mensaje retornado cuando el IBAN no coincide con el número de identificación del patrono.	El número de cuenta IBAN del patrono no coincide con la identificación de éste.
VAL0019	Mensaje retornado cuando el IBAN no se encuentra en estado activo.	La cuenta IBAN del patrono no está activa. Se recomienda ponerse en contacto con el Banco.
VAL0020	Mensaje retornado cuando el IBAN del patrono tiene restricciones.	La cuenta IBAN del patrono presenta restricciones. Se recomienda ponerse en contacto con el Banco.
VAL0021	Mensaje retornado cuando la cuenta del patrono no tiene fondos suficientes para realizar el débito.	La cuenta del patrono no cuenta con fondos suficientes para realizar el débito.
VAL0022	Mensaje retornado cuando la cuenta del patrono no tiene la misma moneda que se indica en la transacción.	La moneda de la cuenta del patrono y la de la transacción no coinciden.
VAL0023	Mensaje retornado cuando la Política Conozca a su Cliente del patrono tiene un estado diferente a activo y/o posee restricciones.	La Política Conozca a su Cliente tiene una condición especial. Se recomienda ponerse en contacto con el Banco.
VAL0024	Mensaje retornado cuando la empresa proveedora de software posee alguna restricción.	la empresa proveedora de software presenta restricciones relacionadas con su identificación.
VAL0025	Mensaje retornado cuando el IBAN no coincide con el número de identificación del colaborador.	El número de cuenta IBAN del colaborador no coincide con la identificación de éste.
VAL0026	Mensaje retornado cuando el IBAN no se encuentra activo.	La cuenta IBAN del colaborador no está activa. Se recomienda ponerse en contacto con el Banco.
VAL0027	Mensaje retornado cuando el IBAN del colaborador tiene restricciones.	La cuenta IBAN del colaborador tiene una condición especial. Se recomienda ponerse en contacto con el Banco.
VAL0028	Mensaje retornado cuando la Política Conozca a su Cliente del colaborador tiene un estado diferente a activo y/o posee restricciones.	La Política Conozca a su cliente del colaborador tiene una condición especial. Se recomienda ponerse en contacto con el Banco.
VAL0029	Mensaje retornado cuando la cuenta del colaborador no tiene la misma moneda que se indica en la transacción.	La moneda de la cuenta del colaborador y de la transacción no coinciden.
VAL0030	Mensaje retornado cuando no se ha logrado comprobar la cédula de la empresa proveedora de software en el Core del Banco.	La identificación de la empresa proveedora de software no pudo ser verificada. Se recomienda ponerse en contacto con el Banco.
VAL0031	Mensaje retornado cuando no se ha logrado comprobar la cédula del patrono en el Core del Banco.	La identificación del patrono no pudo ser verificada. Se recomienda ponerse en contacto con el Banco.
VAL0032	Mensaje retornado cuando no se ha logrado comprobar la cédula del colaborador en el Core del Banco.	La identificación del colaborador no pudo ser verificada. Se recomienda ponerse en contacto con el Banco.
VAL0033	Mensaje retornado cuando el CRR Billing Customer del colaborador no existe o se encuentra inactivo.	No se ha creado la relación entre el colaborador y el patrono. Se recomienda ponerse en contacto con el banco.
VAL0034	Mensaje retornado cuando la identificación del patrono no se encuentra contenida en el CRR Billing Customer del colaborador.	No se ha creado la relación entre el colaborador y el patrono. Se recomienda ponerse en contacto con el Banco.
VAL0035	Mensaje retornado cuando no se ha logrado comprobar la cédula del patrono en el sistema del Banco (SISBI).	La identificación del patrono no pudo ser verificada. Se recomienda ponerse en contacto con el Banco.

VAL0036	Mensaje retornado cuando la cuenta del patrono no se encuentra habilitada para realizar el pago de salario (ligada a CRR billing).	La cuenta del patrono no está habilitada para realizar pagos.
VAL0037	Mensaje retornado cuando la identificación del patrono no se encuentra contenida en el CRR Billing de él mismo.	No se ha creado el patrono. Se recomienda ponerse en contacto con el Banco.
VAL0038	Mensaje retornado cuando el CRR Billing del patrono no existe o se encuentra inactivo.	No se ha creado el patrono o el mismo se encuentra en estado inactivo. Se recomienda ponerse en contacto con el banco.
VAL0039	Mensaje retornado cuando la identificación del patrono no se encuentra contenida en el CRR Billing Customer de él mismo.	No se ha creado el patrono. Se recomienda ponerse en contacto con el Banco.
VAL0040	Mensaje retornado cuando el CRR Billing Customer del patrono no existe o se encuentra inactivo.	No se ha creado la relación patrono-cliente. Se recomienda ponerse en contacto con el banco.
VAL0041	Mensaje retornado cuando la fecha de programación es menor a la actual.	La fecha de programación no puede ser anterior a la fecha actual.
VAL0042	Mensaje retornado cuando se supera la cantidad de transacciones semanales permitidas para un colaborador.	La cantidad de transacciones semanales por colaborador superan el límite de transacciones máximas permitido.
VAL0043	Mensaje retornado cuando el débito a la cuenta del patrono no pudo ser aplicado por el Core del Banco.	No fue posible aplicar el débito a la cuenta del patrono. Se recomienda ponerse en contacto con el Banco.
VAL0044	Mensaje retornado cuando el crédito a la cuenta del colaborador no pudo ser aplicado por el Core del Banco.	No fue posible aplicar el crédito a la cuenta del colaborador. Se recomienda ponerse en contacto con el Banco.
VAL0045	Mensaje retornado cuando el Core del Banco no responde un FT al aplicar el débito al patrono.	No se recibió un comprobante al realizar el débito en la cuenta del patrono.
VAL0046	Mensaje retornado cuando el Core del Banco no responde un FT al aplicar el crédito al patrono.	No se recibió un comprobante al realizar el crédito en la cuenta del colaborador.
VAL0047	Mensaje retornado cuando el Core del Banco no logra aplicar el débito al patrono, pero no responde errores con la razón.	No se logra aplicar el débito al patrono, sin embargo, no se dispone de detalles sobre la causa del inconveniente. Se recomienda ponerse en contacto con el Banco.
VAL0048	Mensaje retornado cuando el Core del Banco no logra aplicar el crédito al colaborador, pero no responde errores con la razón.	No se logra aplicar el crédito al colaborador, sin embargo, no se dispone de detalles sobre la causa del inconveniente. Se recomienda ponerse en contacto con el Banco.
VAL0049	Mensaje retornado cuando el pago es programado pero no se envió una fecha de aplicación.	El pago no se programó porque no se ha recibido la fecha de aplicación.
VAL0050	Mensaje retornado cuando se ha presentado un error inesperado en la reversión a la cuenta del patrono.	Ocurrió un error inesperado durante el proceso de reversión a la cuenta del patrono.
VAL0051	Mensaje retornado cuando no se encuentra una planilla relacionada con el id proporcionado.	No se encuentran planillas registradas con la identificación proporcionada.
VAL0052	Mensaje retornado cuando la categoría de la cuenta del patrono no es permitida.	La categoría de la cuenta del patrono no es permitida para realizar el pago de salario.

VAL0053	Mensaje retornado cuando la categoría de la cuenta del colaborador no es permitida.	La categoría de la cuenta del colaborador no es permitida para recibir el pago del salario.
VAL0054	Mensaje retornado cuando la agencia del patrono registrada en el sistema del Banco (SISBI) es incorrecta.	La agencia del patrono no es válida.
VAL0055	Mensaje retornado cuando la cédula del patrono no coincide con la registrada en el canal de Transferencias Masivas.	La cédula del patrono no está relacionada con el apiKey.
VAL0056	Mensaje retornado cuando el sistema no se encuentra disponible.	El sistema no se encuentra disponible.
VAL0057	Mensaje retornado cuando la cuenta del Colaborador no tiene fondos suficientes para realizar el débito.	La cuenta del colaborador no cuenta con fondos suficientes para realizar el débito.
VAL0058	Mensaje retornado cuando el débito a la cuenta del colaborado no pudo ser aplicado por el Core del Banco.	No fue posible aplicar el débito a la cuenta del colaborador. Se recomienda ponerse en contacto con el Banco.
VAL0059	Mensaje retornado cuando el Core del Banco no logra aplicar el débito al colaborador, pero no responde errores con la razón.	No se logra aplicar el débito al colaborador, sin embargo, no se dispone de detalles sobre la causa del inconveniente. Se recomienda ponerse en contacto con el Banco.
VAL0060	Mensaje retornado cuando el crédito a la cuenta del patrono no pudo ser aplicado por el Core del Banco.	No fue posible aplicar el crédito a la cuenta del patrono. Se recomienda ponerse en contacto con el Banco.
VAL0061	Mensaje retornado cuando el Core del Banco no logra aplicar el crédito al patrono pero no responde errores con la razón.	No se logra aplicar el crédito al patrono, sin embargo, no se dispone de detalles sobre la causa del inconveniente. Se recomienda ponerse en contacto con el Banco.