

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	6
5. TOKEN.....	7
6. MÉTODOS QUE CONFORMAN LA SOLUCIÓN	7
7. API CARGA PLANILLA	7
➤ Parámetros de entrada.....	8
✓ Parámetros de salida	10
✓ Ejemplos	10
8. API CONSULTA PREVALIDACIÓN DE PLANILLAS.....	13
✓ Parámetros de entrada.....	13
✓ Parámetro de salida.....	14
✓ Ejemplos	16
9. API CONSULTA APLICACIÓN DE PAGOS DE PLANILLAS	19
✓ Parámetros de entrada.....	19
✓ Parámetro de salida.....	19
✓ Ejemplos	22
10. API PREVALIDACIÓN PLANILLAS	24
✓ Parámetros de entrada.....	25
✓ Parámetros de salida	25

✓	Ejemplos	26
11.	API PAGO PLANILLAS	26
✓	Parámetros de entrada	27
✓	Parámetros de salida	27
✓	Ejemplos	28

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.

2. DEFINICIONES

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.

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.

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:

- **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.
- **Empresa de prueba:**
El Banco proporcionará una empresa configurada especialmente para este entorno de pruebas.

Consideraciones sobre los Datos de Prueba y URL: Todos los datos como 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.

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 descodificados 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.

➤ **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:

Parámetros	Tipo	Query params /Request Body	Descripción
			CRC 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.
CuentalBAN	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.
CuentalBAN	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.

Parámetros	Tipo	Query params /Request Body	Descripción
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[]>();
}
```



```
"PagoProgramado": true,
"FechaAplicacionPago": "2022-12-15"
},
"PagoPatrono": {
  "Nombre": "Nombre patrono",
  "Identificacion":
"0000000000",
  "CuentaBAN": "CR11116111111111111111",
  "MontoADebitar": 20.50,
  "Descripcion": "Pago de salarios"
},
"DetalleColaboradores": [
  {
    "Nombre": "Nombre Colaborador",
    "Identificacion": "0000000000",
    "CuentaBAN": " CR11116111111111111111",
    "MontoAAcreditar": 10,
    "Descripcion": "Pago sal"
  },
  {
    "Nombre": "Nombre Colaborador",
    "Identificacion": "0000000000",
    "CuentaBAN": " CR11116111111111111111",
    "MontoAAcreditar": 10,
    "Descripcion": "Pago sal"
  }
]
}
```

✓ **Ejemplo Response**

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

✓ **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"
}
```

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.
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
Traceld	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=0000000000&Id=0000000-0000-0000-0000-000000000000&IndicePagina=1&CantidadRegistros=100>

✓ **Ejemplo Response**

```
{
  "StatusCode": 200,
  "Traceld": "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",
    }
  ]
}
```

```
"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**

```
{
  "StatusCode": 200,
  "Traceld": "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",
      "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": 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": "CR11116111111111111111"
  }
},
{
  "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": "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"  
}
```

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
Traceld	String	Identificador interno único asociado a una trama o secuencia de eventos relacionada con la consulta en la API.
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": "6479cabcd34f7a994530d71e8193d4",  
  "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
TraceId	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": "0000000-000-0000-0000-0000000000",  
}
```

✓ Ejemplo Response

```
{  
  "StatusCode": 200,  
  "Traceld": "6479cabcd34f7a994530d71e8193d4",  
  "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"  
}
```