Listar Facturas

Permite obtener la lista paginada de todas las facturas registradas en el sistema, tanto facturas de venta como facturas de contrato. Cada elemento incluye el número de factura, resolución, fechas, valor total, saldo pendiente, tercero facturado, estado, información DIAN, enlace al PDF, detalle de ítems y datos de auditoría.

> **¿Para qué sirve este servicio?**
> Úsalo para sincronizar la información de facturación con tu sistema externo, generar reportes de cartera, alimentar dashboards con indicadores de facturación, validar el estado de cobro de los contratos o exportar la información a Excel y Power BI.

> **¿Solo necesitas facturas de venta?**
> Este endpoint devuelve facturas de venta y de contrato. Si tu integración solo debe operar sobre facturas de venta, usa `GET /invoices/sales/list`. Ese endpoint aplica el mismo contrato de filtros y paginación, pero excluye las facturas de contrato e informa cuántas descartó en `warnings[]`.

---

1. El Endpoint (La dirección web)

Apunta tu sistema a la siguiente dirección. Recuerda reemplazar **{{instancia}}** por la dirección web completa que utilizas para ingresar a tu plataforma.

```
GET https://{{instancia}}/service/v2/public/invoices/list
```

> **¿Qué debes colocar en {{instancia}}?**
> Es muy sencillo: corresponde a la **dirección web principal** que utilizas a diario para ingresar a tu plataforma (incluyendo la terminación `.nuby.app` o `.arrendasoft.co`).
> Por ejemplo, si para entrar a tu sistema escribes `inmobiliaria.nuby.app` o `inmobiliaria.arrendasoft.co` en tu navegador, esa será exactamente tu instancia. Solo asegúrate de no incluir el "https://" ni barras diagonales ("/") al final.

2. La Petición (¿Qué debes enviarnos?)

Este servicio no requiere cuerpo en la petición. Envía los encabezados requeridos y, opcionalmente, los parámetros de paginación y filtros en la URL:

Método	GET
Content-Type	application/json
Authorization	**Bearer token**, Token obtenido al consumir el servicio Login

> **¿Aún no tienes tu Token de acceso?**
> Para consumir este servicio necesitas un Token vigente. Consulta el servicio de Login para aprender cómo obtenerlo.

Parámetros de consulta (query string):

Parámetro	Tipo	Requerido	Por defecto	Descripción
`page`	integer	No	1	Número de página de resultados que se desea recuperar. Debe ser un número positivo.
`page_size`	integer	No	10	Número máximo de facturas por página. Debe ser un número positivo. Valor máximo permitido: **1000**. Si se envía un valor mayor, el sistema lo ajusta a 1000.
`estado`	integer	No	—	Filtra por el estado de la factura. Valores válidos: `1` (Facturada), `2` (Anulada), `3` (Pagada), `5` (Borrador), `6` (Anulada por NC).
`documento`	string	No	—	Filtra por el número de documento del tercero. La coincidencia es exacta.

Parámetro	Tipo	Requerido	Por defecto	Descripción
`pendientes`	boolean	No	false	Si es `true`, devuelve solo facturas pendientes de pago: estado **Facturada** y saldo mayor a cero.
`incluir_borrador`	boolean	No	false	Si es `true` y se usa junto con `pendientes=true`, incluye también las facturas en estado **Borrador**. Sin `pendientes=true` no tiene efecto.
`resolucion_id`	string	No	—	Filtra por una o varias resoluciones de facturación. Acepta un ID numérico o varios separados por coma (ej. `1,2,3`). Solo se aceptan enteros positivos. Si alguna resolución no existe, se ignora; si ninguna existe, el endpoint responde `400`.

> **Notas sobre los filtros**
> — No existe filtro por rango de fechas ni por fecha de modificación. La respuesta incluye `fecha_factura`, `fecha_vencimiento`, `fecha_envio_dian` y `fecha_creacion`, pero ninguno de ellos se puede usar como filtro.
> — No combines `estado` con `pendientes`: `pendientes` ya fija el estado en **Facturada**; si envías ambos, la consulta puede quedar sin resultados.
> — `incluir_borrador` solo aplica cuando `pendientes=true`.
> — Los filtros se combinan con **AND**.

Ejemplos de petición:

```
GET https://mi-inmobiliaria.nuby.app/service/v2/public/invoices/list
GET https://mi-inmobiliaria.nuby.app/service/v2/public/invoices/list?page=2
GET https://mi-inmobiliaria.nuby.app/service/v2/public/invoices/list?page_size=50
GET https://mi-inmobiliaria.nuby.app/service/v2/public/invoices/list?page=3&page_size=100
GET https://mi-inmobiliaria.nuby.app/service/v2/public/invoices/list?estado=1
GET https://mi-inmobiliaria.nuby.app/service/v2/public/invoices/list?documento=1234567890
```

```
GET https://mi-inmobiliaria.nuby.app/service/v2/public/invoices/list?pendientes=true
GET https://mi-inmobiliaria.nuby.app/service/v2/public/invoices/list?pendientes=true&incluir_borrador=true
GET https://mi-inmobiliaria.nuby.app/service/v2/public/invoices/list?resolucion_id=5
GET https://mi-inmobiliaria.nuby.app/service/v2/public/invoices/list?resolucion_id=1,2,3&pendientes=true&page_size=100
```

3. La Respuesta (¿Qué te entregaremos?)

Si tu Token es válido, el sistema te devolverá un objeto JSON con el estado de la respuesta, el listado de facturas y la información de paginación. Los resultados están ordenados por fecha de factura descendente (más recientes primero) y, en caso de empate, por `factura_id` descendente. La respuesta se verá similar a esta:


```
{
    "status": 200,
    "message": null,
    "body": [
        {
            "factura_id": 15230,
            "factura_numero": 8745,
            "resolucion": "FECO - 18764000002851",
            "fecha_factura": "2025-04-01",
            "fecha_vencimiento": "2025-04-06",
            "valor_total": "2850000.00",
            "saldo": "2850000.00",
            "tercero_id": 845,
            "documento_tercero": "80213654",
            "nombre_tercero": "CARLOS EDUARDO PINEDA VARGAS",
            "tipo_documento_tercero": "Cédula de ciudadanía",
            "estado_id": 1,
            "estado": "Facturada",
            "estado_dian": "Aceptada",
            "estado_cliente_dian": "Recibida",
            "observaciones": "CONTRATO 187 - CARLOS EDUARDO PINEDA VARGAS",
            "notas": 0,
            "creado_por": "Sandra",
            "fecha_envio_dian": "2025-04-01 08:30:15",
            "fecha_creacion": "2025-04-01 07:45:22",
```

```
            "pdf_link": "https://mi-
inmobiliaria.nuby.app/fpdf/contabilidad_documento.php?hash=abc123def456",
            "detalles": [
                {
                    "detalle_id": 1,
                    "producto": "Canon De Arrendamiento",
                    "producto_tipo": "canon",
                    "descripcion": "Canon De Arrendamiento De 01-Abr-2025 a 30-Abr-2025",
                    "contrato_id": 187,
                    "contrato_numero": "C-000187",
                    "fecha_inicio_periodo": "2025-04-01",
                    "fecha_fin_periodo": "2025-04-30",
                    "cantidad": 1,
                    "valor_unitario": 2394957.98,
                    "iva": 0,
                    "por_iva": 0,
                    "retencion": 0,
                    "por_retencion": 0,
                    "reteiva": 0,
                    "por_reteiva": 0,
                    "reteica": 0,
                    "por_reteica": 0,
                    "descuento": 0,
                    "por_descuento": 0
                },
                {
                    "detalle_id": 2,
                    "producto": "Administración",
                    "producto_tipo": "administracion",
                    "descripcion": "Administración De 01-Abr-2025 a 30-Abr-2025",
                    "contrato_id": 187,
                    "contrato_numero": "C-000187",
                    "fecha_inicio_periodo": "2025-04-01",
                    "fecha_fin_periodo": "2025-04-30",
                    "cantidad": 1,
                    "valor_unitario": 455042.02,
                    "iva": 0,
                    "por_iva": 0,
                    "retencion": 0,
                    "por_retencion": 0,
```

```
                    "reteiva": 0,
                    "por_reteiva": 0,
                    "reteica": 0,
                    "por_reteica": 0,
                    "descuento": 0,
                    "por_descuento": 0
                }
            ]
        },
        {
            "factura_id": 15229,
            "factura_numero": 8744,
            "resolucion": "FECO - 18764000002851",
            "fecha_factura": "2025-04-01",
            "fecha_vencimiento": "2025-04-06",
            "valor_total": "4500000.00",
            "saldo": "0.00",
            "tercero_id": 845,
            "documento_tercero": "80213654",
            "nombre_tercero": "CARLOS EDUARDO PINEDA VARGAS",
            "tipo_documento_tercero": "Cédula de ciudadanía",
            "estado_id": 3,
            "estado": "Pagada",
            "estado_dian": "Aceptada",
            "estado_cliente_dian": "Recibida",
            "observaciones": null,
            "notas": 0,
            "creado_por": "Sandra",
            "fecha_envio_dian": "2025-04-01 08:31:02",
            "fecha_creacion": "2025-04-01 07:46:10",
            "pdf_link": "https://mi-
inmobiliaria.nuby.app/fpdf/contabilidad_documento.php?hash=def456ghi789",
            "detalles": []
        }
    ],
    "pagination": {
        "total_records": 2,
        "total_pages": 1,
        "current_page": 1,
        "page_size": 10,
```

```
        "current_page_records": 2,
        "has_next_page": false,
        "has_previous_page": false
    },
    "filters_applied": {
        "documento": "80213654"
    }
}
```

Estructura principal de la respuesta

Campo	Tipo	Descripción
`status`	integer	Código del estado de la respuesta. `200` para exitoso, `400` para error de validación, `500` para error interno.
`message`	string | null	Cuando la respuesta es exitosa vale `null`. Si no se encuentran facturas, contiene el mensaje informativo. Si hay error, describe el problema.
`body`	array | null	Listado de facturas devueltas. Arreglo vacío `[]` si no hay resultados. `null` en caso de error.
`pagination`	object | null	Información de paginación para recorrer los resultados. `null` en caso de error.
`filters_applied`	object | null	Filtros que realmente se aplicaron a la consulta. Solo incluye las claves de los filtros enviados. Es `null` cuando no se envió ningún filtro; en el error de resoluciones inexistentes conserva los filtros enviados.

Campos de cada factura (body)

Cada elemento dentro del arreglo `body` contiene las siguientes claves:

Campo	Tipo	Descripción
`factura_id`	integer	Identificador único de la factura en el sistema.
`factura_numero`	integer	Número consecutivo de la factura.
`resolucion`	string	Prefijo y número de la resolución de facturación en formato `PREFIJO - NÚMERO` (ej. `"FECO - 18764000002851"`).
`fecha_factura`	string	Fecha de emisión de la factura en formato `YYYY-MM-DD`.

Campo	Tipo	Descripción
`fecha_vencimiento`	string	Fecha de vencimiento de la factura en formato `YYYY-MM-DD`.
`valor_total`	string	Valor total de la factura con decimales (ej. `"2850000.00"`).
`saldo`	string	Saldo pendiente de pago de la factura con decimales (ej. `"0.00"` cuando está pagada).
`tercero_id`	integer	Identificador interno del tercero en el sistema.
`documento_tercero`	string	Número de documento de identidad del tercero/cliente facturado.
`nombre_tercero`	string	Nombre completo del tercero/cliente facturado.
`tipo_documento_tercero`	string | null	Tipo de documento del tercero (ej. `Cédula de ciudadanía`, `NIT`). `null` si el tercero no tiene tipo de documento configurado.
`estado_id`	integer	Identificador numérico del estado de la factura. Valores posibles: `1`, `2`, `3`, `5` y `6`.
`estado`	string	Estado textual de la factura. Valores posibles: `Facturada`, `Anulada`, `Pagada`, `Borrador`, `Anulada por NC`, `Contabilizada`.
`estado_dian`	string | null	Estado de la factura en la DIAN cuando es factura electrónica (ej. `"Aceptada"`). `null` si no aplica o no se ha enviado.
`estado_cliente_dian`	string | null	Estado de aceptación/rechazo del cliente ante la DIAN cuando es factura electrónica. `null` si no aplica.
`observaciones`	string | null	Observaciones registradas en la factura. `null` si no tiene.
`notas`	integer	Cantidad de notas débito o crédito asociadas a la factura.
`creado_por`	string | null	Nombre del usuario que creó la factura en el sistema. `null` si el usuario no existe o no está disponible.
`fecha_envio_dian`	string | null	Fecha y hora de envío de la factura a la DIAN en formato `YYYY-MM-DD HH:MM:SS`. `null` si no se ha enviado.
`fecha_creacion`	string	Fecha y hora de creación del registro en formato `YYYY-MM-DD HH:MM:SS`.

Campo	Tipo	Descripción
`pdf_link`	string	Enlace seguro para visualizar o descargar el PDF de la factura. El enlace incluye un hash con vigencia temporal.
`detalles`	array	Lista de ítems de la factura. Cada elemento se describe en la sección **Campos de cada detalle**.

Campos de paginación (pagination)

El objeto `pagination` contiene las siguientes claves:

Campo	Tipo	Descripción
`total_records`	integer	Total de facturas que cumplen los filtros aplicados. No es necesariamente el total de facturas de la instancia.
`total_pages`	integer	Cantidad total de páginas disponibles (total de registros dividido por el tamaño de página).
`current_page`	integer	Número de la página actual consultada.
`page_size`	integer	Tamaño de la página (cantidad de registros solicitados por consulta).
`current_page_records`	integer	Cantidad de registros efectivamente devueltos en la página actual.
`has_next_page`	boolean	`true` si existen páginas posteriores que se pueden consultar.
`has_previous_page`	boolean	`true` si existen páginas anteriores que se pueden consultar.

Campos de filters_applied

El objeto `filters_applied` contiene solo las claves de los filtros que se enviaron. Si no se envió ningún filtro, el campo es `null`.

> **Paginación no incluida**
> Los parámetros `page` y `page_size` no se incluyen en `filters_applied`; su valor se refleja en el objeto `pagination`.

Campo	Tipo	Descripción

`estado`	integer	Solo si se envió `estado`. Estado numérico aplicado.
`documento`	string	Solo si se envió `documento`. Documento exacto aplicado.
`pendientes`	boolean	Solo si se envió `pendientes=true`. Indica que se aplicó el filtro de pendientes.
`incluir_borrador`	boolean	Solo si se envió `incluir_borrador` (true o false). Indica si se incluyeron borradores.
`resolucion_ids`	array<integer>	Solo si se envió `resolucion_id`. IDs de resolución que existen y se aplicaron.

> **Nombre del filtro en la respuesta**
> El query string se llama `resolucion_id` (singular), pero en `filters_applied` se publica como `resolucion_ids` (plural) y su valor es un arreglo de enteros.

Campos de cada detalle (detalles[])

Cada elemento del arreglo `detalles` contiene las siguientes claves:

Campo	Tipo	Descripción
`detalle_id`	integer	Identificador único del detalle de la factura.
`producto`	string	Nombre del producto o concepto facturado.
`producto_tipo`	string | null	Tipo de producto según su relación inmobiliaria (ej. `canon`, `administracion`, `interes`). `null` si el producto no tiene tipo configurado.
`descripcion`	string	Descripción del detalle o del periodo facturado.
`contrato_id`	integer | null	Identificador del contrato asociado al detalle. `null` en facturas de venta sin contrato.
`contrato_numero`	string | null	Consecutivo del contrato asociado. `null` si no aplica.
`fecha_inicio_periodo`	string | null	Fecha de inicio del periodo del movimiento en formato `YYYY-MM-DD`. `null` si el detalle no tiene movimiento asociado.

Campo	Tipo	Descripción
`fecha_fin_periodo`	string | null	Fecha de fin del periodo del movimiento en formato `YYYY-MM-DD`. `null` si el detalle no tiene movimiento asociado.
`cantidad`	number	Cantidad facturada.
`valor_unitario`	number	Valor unitario del ítem.
`iva`	number	Valor del IVA del ítem.
`por_iva`	number	Porcentaje de IVA aplicado.
`retencion`	number	Valor de la retención en la fuente.
`por_retencion`	number	Porcentaje de retención en la fuente.
`reteiva`	number	Valor de la retención de IVA.
`por_reteiva`	number	Porcentaje de retención de IVA.
`reteica`	number	Valor de la retención de ICA.
`por_reteica`	number	Porcentaje de retención de ICA.
`descuento`	number	Valor del descuento aplicado.
`por_descuento`	number	Porcentaje de descuento aplicado.

> **Nota sobre facturas sin resultados**
> Si ninguna factura cumple los filtros, la respuesta tendrá HTTP `200` con `body` vacío (`[]`), el mensaje `"No se encontraron facturas con los criterios especificados."` y los valores de paginación en cero. Si se enviaron filtros, `filters_applied` conserva los filtros aplicados.

---

4. Seguridad y Posibles Errores

> **¡Tu pase tiene fecha de caducidad!**

Por medidas de seguridad, el `token` que te entregamos **solo dura 1 hora**. Una vez transcurrido ese tiempo, el pase expirará y el sistema te bloqueará el acceso devolviéndote un error `401`. Cuando esto ocurra, tu sistema simplemente debe volver a consumir el servicio de Login para pedir un pase nuevo y continuar trabajando.

Así se ve el error que te devolverá el sistema cuando tu token se haya vencido o intentes usar un pase inválido:

```
{
    "statusCode": 401,
    "error": {
        "type": "SERVER_ERROR",
        "description": "JWT Token expired."
    }
}
```


Si alguno de los parámetros de paginación o filtros no cumple con las validaciones, el sistema devolverá un error `400`. Por ejemplo:


```
{
    "status": 400,
    "message": "El parámetro \"page_size\" debe ser un número positivo.",
    "body": null,
    "pagination": null,
    "filters_applied": null
}
```


Posibles errores:

Código HTTP	Descripción
400	**Formato de autenticación inválido.** El encabezado `Authorization` no tiene el formato `Bearer {token}`. Mensaje: `"JWT Token not send."` **Parámetros inválidos.** Posibles causas: — `page` no es numérico o no es positivo. Mensaje: `"El parámetro \"page\" debe ser un número positivo."` — `page_size` no es numérico o no es positivo. Mensaje: `"El parámetro \"page_size\" debe ser un número positivo."` — `estado` no es uno de los valores permitidos. Mensaje: `"El parámetro \"estado\" debe ser uno de los siguientes valores: 1 (Facturada), 2 (Anulada), 3 (Pagada), 5 (Borrador), 6 (Anulada por NC)."` — `resolucion_id` contiene valores no numéricos o no positivos. Mensaje: `"El parámetro \"resolucion_id\" debe contener solo IDs numéricos enteros positivos separados por coma (ej: 1,2,3)."` — `resolucion_id` no contiene ningún ID válido. Mensaje: `"El parámetro \"resolucion_id\" no contiene IDs válidos."` — Ninguna de las resoluciones enviadas existe. Mensaje: `"Ninguna de las resoluciones enviadas existe en el sistema."`

Código HTTP	Descripción
401	**Token ausente, no encontrado o expirado.** Posibles causas: — No se envió el encabezado `Authorization`. Mensaje: `"JWT Token required."` — El token no existe en el sistema. Mensaje: `"JWT Token not found."` — El token expiró. Mensaje: `"JWT Token expired."` En todos los casos debes generar un token nuevo consumiendo el servicio de **Login**.
500	Error interno del servidor al procesar la consulta de facturas. Mensaje: `"Error interno del servidor al obtener las facturas."`

> **Sobre los permisos**
> Este endpoint valida el token, pero no exige un scope específico. El acceso depende de que el cliente OAuth esté registrado en la instancia. Si tu integración requiere control de permisos por scope, usa los endpoints modernos del ciclo administrativo, como `GET /invoices/sales/list`, que exige `catalogs:read`.

---

5. Ejemplos de integración

Aquí tienes ejemplos de código listos para que tus desarrolladores los adapten a tu plataforma:

cURL

```
# Listar facturas con paginación por defecto
curl -X GET "https://{{instancia}}/service/v2/public/invoices/list" \
-H "Content-Type: application/json" \
-H "Authorization: Bearer TU_TOKEN_AQUI"

# Listar facturas página 3 con 100 registros por página
curl -X GET "https://{{instancia}}/service/v2/public/invoices/list?page=3&page_size=100" \
-H "Content-Type: application/json" \
-H "Authorization: Bearer TU_TOKEN_AQUI"

# Filtrar por estado (1 = Facturada)
curl -X GET "https://{{instancia}}/service/v2/public/invoices/list?estado=1" \
-H "Authorization: Bearer TU_TOKEN_AQUI"
```

```
# Filtrar por documento del tercero y solo pendientes
curl -X GET
"https://{{instancia}}/service/v2/public/invoices/list?documento=1234567890&pendientes=tru
e" \
-H "Authorization: Bearer TU_TOKEN_AQUI"

# Incluir facturas en borrador junto con las pendientes
curl -X GET
"https://{{instancia}}/service/v2/public/invoices/list?pendientes=true&incluir_borrador=tr
ue" \
-H "Authorization: Bearer TU_TOKEN_AQUI"

# Filtrar por una o varias resoluciones
curl -X GET "https://{{instancia}}/service/v2/public/invoices/list?resolucion_id=1,2,3" \
-H "Authorization: Bearer TU_TOKEN_AQUI"
```

PHP

```
<?php

$instance = 'tu_instancia'; // Reemplaza con tu instancia real
$token = 'TU_TOKEN_AQUI'; // Token obtenido del servicio Login

// Parámetros de paginación y filtros
$page = 1;
$page_size = 50;

$queryParams = http_build_query([
    'page' => $page,
    'page_size' => $page_size,
    // Filtros opcionales. Elimina los que no necesites.
    'estado' => 1,
    'documento' => '1234567890',
    'pendientes' => 'true',
    'incluir_borrador' => 'true',
    'resolucion_id' => '1,2,3'
```

```
]);

$url = "https://{$instance}/service/v2/public/invoices/list?{$queryParams}";

$ch = curl_init($url);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
curl_setopt($ch, CURLOPT_HTTPHEADER, [
    'Content-Type: application/json',
    "Authorization: Bearer {$token}"
]);

$response = curl_exec($ch);
$http_code = curl_getinfo($ch, CURLINFO_HTTP_CODE);

if (curl_errno($ch)) {
    echo 'Error: ' . curl_error($ch);
} else {
    echo "Código de estado HTTP: " . $http_code . "\n";

    $info = json_decode($response, true);
    $body = $info['body'] ?? [];

    if ($http_code === 200 && !empty($body)) {
        $pagination = $info['pagination'];
        echo "Facturas encontradas: {$pagination['total_records']} (página
{$pagination['current_page']} de {$pagination['total_pages']})\n";

        foreach ($body as $factura) {
            echo "  Factura #{$factura['factura_numero']} - {$factura['nombre_tercero']} -
{$factura['estado']} - Total: \${$factura['valor_total']} - Saldo:
\${$factura['saldo']}\n";
        }
    } else {
        echo "Mensaje: " . ($info['message'] ?? 'Sin resultados') . "\n";
    }
}
curl_close($ch);

?>
```

Python

```
import requests

# 1. Configura tus datos de acceso
instancia = 'mi-inmobiliaria.nuby.app'  # Reemplaza con tu dirección web completa
token = 'TU_TOKEN_AQUI'  # Token obtenido del servicio Login

# 2. Prepara la dirección y los parámetros
url = f"https://{instancia}/service/v2/public/invoices/list"

params = {
    "page": 1,
    "page_size": 50,
    # Filtros opcionales. Elimina los que no necesites.
    "estado": 1,
    "documento": "1234567890",
    "pendientes": "true",
    "incluir_borrador": "true",
    "resolucion_id": "1,2,3"
}

headers = {
    "Content-Type": "application/json",
    "Authorization": f"Bearer {token}"
}

# 3. Envía la petición GET y procesa la respuesta
try:
    response = requests.get(url, params=params, headers=headers)

    print(f"Código de estado HTTP: {response.status_code}")

    if response.status_code == 200:
        info = response.json()
        body = info.get('body', [])
        pagination = info.get('pagination', {})
```

```
        if body:
            print(f"Facturas: {pagination['total_records']} total (página
{pagination['current_page']} de {pagination['total_pages']})")
            for factura in body:
                print(f"  #{factura['factura_numero']} - {factura['nombre_tercero']} -
{factura['estado']} - Total: ${factura['valor_total']} - Saldo: ${factura['saldo']}")
        else:
            print(f"Mensaje: {info.get('message', 'Sin resultados')}")
    else:
        print("Error al consultar las facturas.")
        print(f"Detalle del error: {response.text}")

except Exception as e:
    print(f"Ocurrió un error de conexión: {e}")
```

JavaScript

```
// 1. Configura tus datos de acceso
const instancia = 'mi-inmobiliaria.nuby.app'; // Reemplaza con tu dirección web completa
const token = 'TU_TOKEN_AQUI'; // Token obtenido del servicio Login

// 2. Prepara la dirección con parámetros
const params = new URLSearchParams({
    page: 1,
    page_size: 50,
    // Filtros opcionales. Elimina los que no necesites.
    estado: 1,
    documento: '1234567890',
    pendientes: 'true',
    incluir_borrador: 'true',
    resolucion_id: '1,2,3'
});

const url = `https://${instancia}/service/v2/public/invoices/list?${params}`;

// 3. Función para consultar las facturas
async function consultarFacturas() {
```

```
    try {
        const response = await fetch(url, {
            method: 'GET',
            headers: {
                'Content-Type': 'application/json',
                'Authorization': `Bearer ${token}`
            }
        });

        console.log(`Código de estado HTTP: ${response.status}`);

        if (response.ok) {
            const info = await response.json();
            const body = info.body || [];
            const pagination = info.pagination || {};

            if (body.length > 0) {
                console.log(`Facturas: ${pagination.total_records} total (página
${pagination.current_page} de ${pagination.total_pages})`);
                body.forEach(factura => {
                    console.log(`  #${factura.factura_numero} - ${factura.nombre_tercero}
- ${factura.estado} - Total: $${factura.valor_total} - Saldo: $${factura.saldo}`);
                });
            } else {
                console.log(`Mensaje: ${info.message || 'Sin resultados'}`);
            }
        } else {
            console.log('Error al consultar las facturas.');
            const errorData = await response.text();
            console.log(`Detalle del error: ${errorData}`);
        }
    } catch (error) {
        console.error(`Ocurrió un error de conexión: ${error}`);
    }
}

// 4. Ejecutamos la función
consultarFacturas();
```

Power Query M (Excel / Power BI)

```
let
    // 1. Configura tus datos de acceso
    instancia = "mi-inmobiliaria.nuby.app",  // Reemplaza con tu dirección web completa
    token = "TU_TOKEN_AQUI",  // Token obtenido del servicio Login

    // 2. Prepara la dirección de la petición con parámetros y filtros
    url = "https://" & instancia &
"/service/v2/public/invoices/list?page=1&page_size=1000&estado=1&pendientes=true",

    // 3. Envía la petición GET
    response = Web.Contents(url, [
        Headers = [
            #"Content-Type" = "application/json",
            #"Authorization" = "Bearer " & token
        ]
    ]),

    // 4. Decodifica la respuesta JSON
    jsonResponse = Json.Document(response),
    body = jsonResponse[body],

    // 5. Convierte el listado en tabla
    // La columna "detalles" es una lista anidada; se puede expandir después si se
necesita.
    tabla = Table.FromList(body, Splitter.SplitByNothing(), null, null,
ExtraValues.Error),
    expandido = Table.ExpandRecordColumn(tabla, "Column1",
        {"factura_id", "factura_numero", "resolucion", "fecha_factura",
"fecha_vencimiento", "valor_total", "saldo", "tercero_id", "documento_tercero",
"nombre_tercero", "tipo_documento_tercero", "estado_id", "estado", "estado_dian",
"estado_cliente_dian", "observaciones", "notas", "creado_por", "fecha_envio_dian",
"fecha_creacion", "pdf_link"},
        {"FacturaID", "Numero", "Resolucion", "FechaFactura", "FechaVencimiento",
"ValorTotal", "Saldo", "TerceroID", "DocTercero", "NombreTercero", "TipoDocTercero",
"EstadoID", "Estado", "EstadoDian", "EstadoClienteDian", "Observaciones", "Notas",
"CreadoPor", "FechaEnvioDian", "FechaCreacion", "PdfLink"})
in
```

```
expandido
```

Revisión #18
Creado el 1 agosto 2025 19:06:57 por Jorge Iván González
Actualizado el 20 noviembre 2026 16:00:17 por Isabel Higuita