

Documentación » Solución de problemas » Errores 400: estructura del request

Errores 400: estructura del request

Los siete errores de estructura que explican la mitad de los problemas reportados, con ejemplos correctos e incorrectos.

Anonymous · 11/09/2026

Errores 400: estructura del request

La mitad de los problemas que atendemos son errores **400** causados por la estructura del cuerpo de la petición. Esta es la lista de los patrones concretos que hemos visto, y sirve como verificación antes de reportar un caso.

1. Campos en el nivel equivocado

Los datos deben ir anidados donde el recurso los espera, no en la raíz del JSON.

Incorrecto — datos del DTE al nivel raíz:

```
{
  "emisor": "22222222-2",
  "receptor": "33333333-3",
  "dte": 33,
  "folio": 93,
  "auth": { "pass": { "rut": "11111111-1", "clave": "..." } }
}
```

Correcto — anidados dentro de **dte**:

```
{
  "auth": { "pass": { "rut": "11111111-1", "clave": "..." } },
  "dte": {
    "dte": 33,
    "emisor": "22222222-2",
    "receptor": "33333333-3",
    "folio": 93,
    "fecha": "2026-04-13",
    "total": 1407734
  }
}
```

Lo mismo ocurre con los **filtros** de los recursos del Portal MiPYME, que van dentro de un objeto

filtros:

Incorrecto:

```
{
  "auth": { "pass": { "rut": "11111111-1", "clave": "..." } },
  "FEC_DESDE": "2026-08-01",
  "FEC_HASTA": "2026-08-25",
  "TPO_DOC": "33"
}
```

Correcto:

```
{
  "auth": { "pass": { "rut": "11111111-1", "clave": "..." } },
  "filtros": {
    "FEC_DESDE": "2026-08-01",
    "FEC_HASTA": "2026-08-31",
    "TPO_DOC": 33,
    "ESTADO": "RRC"
  }
}
```

Este error en particular se manifiesta como un 409 de indisponibilidad del SII, no como un 400, lo que lo hace difícil de diagnosticar. Revisa [Intermitencias del SII](#).

2. Formato de fechas

Siempre **ISO**, **YYYY-MM-DD**:

Correcto	Incorrecto
2026-06-21	21-06-2026

3. Formato de períodos

Los períodos mensuales van como **YYYYMM**, **sin guión**:

Correcto	Incorrecto
202605	2026-05

Enviarlo con guión produce un 404 **Página no encontrada**, porque la ruta no calza. Algunos listados aceptan día en formato **YYYYMMDD**: confirma en la documentación del recurso.

4. Formato del RUT

Sin puntos, con guión y con dígito verificador, y la K en mayúscula:

Correcto	Incorrecto
11111111-1	76.192.083-9
22222222-2	22222222-2

Un RUT con puntos puede producir errores que parecen de otra naturaleza. Un caso real recibía un timeout de navegación al portal del SII cuya causa era "rut": "76.491.518-6".

5. Confundir el RUT de cada campo

Es frecuente confundir el RUT del **contribuyente que se consulta** con el de la **credencial que inicia sesión**. En muchos recursos son distintos:

- En la **ruta**: el RUT del contribuyente cuya información pides.
- En `auth.pass.rut`: el RUT de quien **inicia sesión**, que puede ser una persona natural autorizada.

En el recurso de emisión de eBoleta hay tres RUT y cada uno cumple un rol distinto:

Campo	Qué debe contener
<code>auth.pass.rut</code>	El usuario autorizado que inicia sesión
<code>dte.vendedor</code>	El mismo usuario que autentica
<code>dte.Encabezado.Emissor.RUTEmissor</code>	La empresa emisora

Enviar el RUT del usuario donde va la empresa produce el error "No tiene folios asignados", que parece un problema tributario y no lo es.

6. Parámetros que no existen

Agregar campos que el recurso no define —por ejemplo un `receptor_rut` inventado— puede hacer fallar la operación. Envía solo lo documentado.

7. Parámetros obligatorios ausentes o con valores inválidos

Algunos listados requieren parámetros fáciles de pasar por alto, como `pagina`. El mensaje los nombra: "El parámetro 'pagina' es requerido".

Y un valor fuera de la lista válida produce error aunque el JSON esté bien formado. Por ejemplo, en

documentos recibidos del Portal MiPYME los estados válidos son EMI, PRV, DCD, DEI, DFI, DIN, DPF, DRF, DRI, DRR, INI, RAC, RAD, RNR, RRC, RRH y RSR. Un valor como RCP no existe y la consulta falla.

Cómo resolverlo rápido

Ejecuta la operación en la [documentación interactiva](#) con **Try it out**: ahí ves el esquema exacto del body, los parámetros obligatorios y un ejemplo válido. Cuando funciona ahí, replícalo en tu código y compara ambos requests.

Un truco para acotar el problema

Envía el body **incompleto** a propósito. Si la API responde 400 con un JSON claro (“Debe especificar el tipo de DTE”, “Debe especificar el RUT del receptor”), las validaciones funcionan, y eso reduce el problema a un campo con **valor** inválido, no a un campo ausente.

Última actualización el 11/09/2026