

Les erreurs avec problem+json

Le code de statut dit **qu'il** y a une erreur. Le body doit dire **laquelle**, et comment la corriger. Plutôt que d'inventer un format par API, il existe un standard :

`application/problem+json`.

Le format

```
HTTP/1.1 409 Conflict
Content-Type: application/problem+json

{
  "type": "https://api-ritif.example/problems/order-not-cancellable",
  "title": "La commande ne peut plus être annulée",
  "status": 409,
  "detail": "La commande 42 est en préparation. Le shaker tourne déjà.",
  "instance": "/v1/orders/42"
}
```

Champ	Rôle
<code>type</code>	Une URI qui identifie le type d'erreur. Idéalement, une page de documentation qui l'explique. Par défaut : <code>about:blank</code> .
<code>title</code>	Un résumé court, le même pour toutes les erreurs de ce type.
<code>status</code>	Le code de statut HTTP, répété pour faciliter la lecture.
<code>detail</code>	L'explication propre à cette occurrence, lisible par un humain.
<code>instance</code>	Une URI qui identifie cette occurrence précise (la ressource concernée, ou un identifiant de log).

Tous les champs sont facultatifs, mais `type`, `title` et `status` sont un minimum raisonnable.

Les extensions

On peut ajouter ses propres champs. Le cas le plus courant : les erreurs de validation, avec la liste des champs en faute.

```
HTTP/1.1 422 Unprocessable Content
Content-Type: application/problem+json

{
  "type": "https://api-ritif.example/problems/validation-error",
  "title": "Données invalides",
  "status": 422,
  "detail": "La requête contient 2 erreurs.",
  "errors": [
    {"field": "rating", "message": "doit être compris entre 1 et 5"},
    {"field": "comment", "message": "ne doit pas dépasser 500 caractères"}
  ]
}
```

Le client peut afficher chaque message sous le bon champ du formulaire, sans parser une phrase.

Les bonnes pratiques

- Le même format pour **toutes** les erreurs de l'API, y compris les `404` et les `500`.
- Un `detail` qui dit quoi corriger : « la note doit être comprise entre 1 et 5 », pas « erreur de validation ».
- Jamais d'informations internes dans une erreur.

Ceci n'est pas un message d'erreur, c'est une fuite d'informations :

```
{"error": "SQLSTATE[42S22]: Column not found: 1054 Unknown column 'prix' in  
/var/www/api/models/Cocktail.php line 42"}
```

Un attaquant y apprend votre base de données, votre langage, votre arborescence et le nom de vos colonnes. Loggez le détail côté serveur, renvoyez un message neutre au client.

Revision #3
Created 2 October 2026 20:00:49 by Thibaud FRICHET
Updated 13 October 2026 20:19:21 by Thibaud FRICHET