

68. JoinData: foutcode 200 of 400

27jun23: Van: Renze van de Kuilen <renze@join-data.nl>

Verzonden: dinsdag 27 juni 2023 11:38

Aan: Floris Ruitkamp <Ruitkamp@qlip.nl>

CC: Gerhard van Steenberg <gerhard@join-data.nl>

Onderwerp: Response foutafhandeling

Hoi Floris, Wanneer een afnemer bijvoorbeeld een te groot tijdvak gebruikt bij het opvragen van data krijgt men nu de response zoals in de bijlage. Het valt me daarbij op dat jullie hierbij de statuscode '200' dus 'OK' teruggeven, dit lijkt me niet juist. Een code '400' – 'Bad request' is hierbij logischer.

Dan verdwijnt bij ons namelijk ook de melding die erboven wordt weergegeven omdat wij het nu interpreteren als een succesvolle response. Kunnen jullie dit aanpassen?

28jun23: Van: Floris Ruitkamp <Ruitkamp@qlip.nl>

Aan: Wesley van de Kamp <kamp@qlip.nl>; Ruit, J.B. de (Bob) <bob.deruit@frieslandcampina.com>;

'Renze van de Kuilen' <renze@join-data.nl>; Graumans Conny Agro-connect <c.graumans@chello.nl>;

'Erwin Speybroeck' <erwin.speybroeck@crv4all.com>

CC: 'Beuker, E. (Esther)' <Esther.Beuker@frieslandcampina.com>

Onderwerp: FW: Response foutafhandeling

Renze vraagt dus om een **400 terug te geven** als we een fout hebben als het **tijdvak te groot** is. (is nu een 200)

```
{
  "errors": [
    {
      "id": "9a7bfe84-cc55-40e1-bf97-e39d3e86e35f",
      "code": "03",
      "title": "Request geweigerd, gevraagd tijdvak te groot",
      "detail": "Request geweigerd, gevraagd tijdvak te groot",
      "status": 200
    }
  ]
}
```

Ook als een **datum ongeldig** is, zou een **400** moeten terug komen denk ik. Die geeft nu ook een 200

```
{
  "errors": [
    {
      "id": "a974fd07-4507-46df-b77f-987e3bc11c1d",
      "title": "Invalid date Time value \"2022-02-07T23:59:61\" (Second is out of range)",
      "detail": "Invalid date Time value \"2022-02-07T23:59:61\" (Second is out of range)",
      "status": 200
    }
  ]
}
```

Ook een **ongeldige locatie** geeft

```
{
  "errors": [
    {
      "id": "77356cc2-11e7-4e29-a59a-166dab69084d",
      "code": "0",
      "title": "For input string: \"a\"",
      "detail": "For input string: \"a\"",
      "status": 200
    }
  ]
}
```

Op internet lees ik het volgende: '...*The HyperText Transfer Protocol (HTTP) 400 Bad Request response status code indicates that the server cannot or will not process the request due to something that is perceived to be a client error (for example, malformed request syntax, invalid request message framing, or deceptive request routing)*....'

Het **voorstel van Renze** lijkt me daarom ook **correct**.

Vragen:

@ Renze : hoe pakken jullie de 200 en 400 melding verder op: nemen jullie dan "title" en "detail" over naar het antwoord richting de opvrager?

@ Bob: Ben jij het eens met de regel dat:

- Als het request geldig is, maar er geen data is, dat er dan een 200 error komt: met de opmerking "geen data gevonden in opgevraagde periode"

- Als het request ON-geldig is, dat er dan een 400 komt (ongeldige datum, ongeldige time-frame etc.)
- Als er “iets” anders fout gaar dat we dan een 500 geven (als een soort van Catch-the-rest)

@ Erwin – Is dit voor jou een acceptabele werkwijze (zoals beschreven bij Bob)

@ Allen – als hier wijzingen uit voor komen, dan vergt dit coördinatie over het moment van aanpassen (vandaar Conny ook in de cc). Ik hoor graag jullie reacties

28jun23: Van: Ruiter, J.B. de (Bob) <bob.deruiter@frieslandcampina.com>

Aan: Floris Ruiterkamp <Ruiterkamp@qlip.nl>; Wesley van de Kamp <kamp@qlip.nl>; 'Renze van de Kuilen' <renze@join-data.nl>; Graumans Conny Agro-connect <c.graumans@chello.nl>; 'Erwin Speybroeck' <erwin.speybroeck@crv4all.com>

CC: Beuker, E. (Esther) <Esther.Beuker@frieslandcampina.com>

Onderwerp: RE: Response foutafhandeling

@ Bob: Ben jij het eens met de regel dat:

- Als het request geldig is, maar er geen data is, dat er dan een 200 error komt: met de opmerking “geen data gevonden in opgevraagde periode”
- Als het request ON-geldig is, dat er dan een 400 komt (ongeldige datum, ongeldige time-frame etc.)
- Als er “iets” anders fout gaar dat we dan een 500 geven (als een soort van Catch-the-rest)

Ik ben het eens met alle 3 de punten.

Aanvullend:

- In de response meenemen van welke **data provider** het bericht komt
- Foutmelding **indien er een datum mist** (dus alleen een start- of eind datum)
- Ruimte voor **interne versie nummers** in het bericht (nummer wordt uitgegeven door de data provider)

29jun23: Van: Erwin Speybroeck <erwin.speybroeck@crv4all.com>

Op zich geen probleem. Nu is het wel zo dat **JoinData altijd een 200** terug geeft als er bij JoinData zelf niets verkeerd gaat, met daarin dan de mogelijke error meldingen van de dataleverancier. Op zich zou het onze voorkeur hebben dat we die HTTP code van de data leverancier ook in de bovenliggende httpcode van JoinData terugkrijgen. Nu krijg je een http 200 met daarin een error, waar een andere http code staat.

29jun23, uit het werkoverleg van Esther en Conny:

Goed punt van Erwin; Esther neemt dit mee in het overleg met Bob.

30jun23, Bob de Ruiter: Nabrander, foutmeldingen in zowel JSON als XML? Afhankelijk of de UBL invoice wordt opgevraagd of niet. Wat is jullie idee hierbij?

30jun23: Van: Floris Ruiterkamp <Ruiterkamp@qlip.nl>

Wat mij betreft is dat **allemaal dezelfde manier van werken (JSON of UBL)**. Ik heb ook nog een nabrander, naar aanleiding van de aanvulling van Bob:

Aanvullend:

- *In de response meenemen van welke data provider het bericht komt*
- *Foutmelding indien er een datum mist (dus alleen een start- of eind datum)*
- *Ruimte voor interne versie nummers in het bericht (nummer wordt uitgegeven door de data provider)*

Bob.

1^e en 3^e bullit, zijn volgens veranderingen in de berichten structuur. De 2^e bullit en het voornemen om de inhoud van de foutmelding aan te passen (dus vaker een 400 ipv een 200), doet in principe niets met de berichten structuur. Daarom is mijn voorstel om dit als 2 aparte wensen te hanteren.

@Renze kan jij nog een voorstel doen op het fenomeen dat Erwin hieronder beschrijft?

“...Nu is het wel zo dat **JoinData altijd een 200** terug geeft als er bij JoinData zelf niets verkeerd gaat, met daarin dan de mogelijke error meldingen van de dataleverancier. Op zich zou het onze voorkeur hebben dat we die HTTP code van de data leverancier ook in de bovenliggende httpcode van JoinData terugkrijgen. Nu krijg je een http 200 met daarin een error, waar een andere http code staat...”

21aug23: Van: Erwin Speybroeck <erwin.speybroeck@crv4all.com>

Esther, Enige opmerking die ik heb: Ingeval van geen data zou ik persoonlijk geen error melding geven, maar gewoon de lege data array terug leveren. Dat is wat mij betreft duidelijk genoeg en het is op zich ook geen error. Mij is niet duidelijk waarom we naast de lege data array we ook nog een melding willen hebben.

29aug23, uit de OC:

Voor dit overleg:

- In issue 68 staat het verzoek van Floris om 2 opmerkingen van Bob apart te behandelen:
 1. In de response meenemen van welke data provider het bericht komt?
 2. Ruimte voor interne versie nummers in het bericht (nummer wordt uitgegeven door de data provider)?
- Besluit gevraagd: Beslisdocument ([bijlage 5b](#)) bespreken met voorgestelde aanscherpingen in foutmeldingen.

Uit dit overleg:

- **Aandachtspunt:** De error meldingen verdienen nadere analyse:
 - Error-meldingen kunnen afkomstig zijn van de DataProvider (RFC, Qlip) en/of van JoinData. Moet hierin naar de IT-service-provider c.q. de DataConsumer onderscheid gemaakt worden? En zo ja, hoe dan? En welke foutcodes mogen gebruikt worden door de DataProvider en welke door JoinData?
 - Voor één aanvraag voor een specifieke melkveehouder kan het zijn dat eDairy-data van twee verschillende zuivelaars door JoinData moet worden verzameld en moet worden aangeboden in de response, waarbij per zuivelaar andere foutmeldingen kunnen optreden. Hoe dit netje in de response te onderscheiden zonder dat het onnodig complex wordt?
- **Besluit:** Floris neemt het voortouw om dit samen met Renze en Bob verder uit te werken; indien nodig betreft eenieder de interne specialisten hierbij. De volgende aanpak wordt voorgesteld:
 1. in een flowchart het proces helder weergeven.
 2. uitschrijven welke foutmeldingen door wie genereerd worden.
 3. aangeven hoe deze foutmeldingen in de responses dienen (op verschillende niveaus?) dienen te worden meegegeven.
 4. impact-analyse voor het doorvoeren van eventuele changes (MAJOR, MINOR?).
 5. het resultaat van bovenstaande bespreken in de eDairy OC en besluiten over het vervolg.
- **Besluit:** Eenieder wordt gevraagd om nog eens goed naar bijlage 5a en 5b te kijken en eventuele inzichten – aandachtspunten – aanbevelingen uiterlijk vrijdag 8sep23 mee te geven aan Floris.

31aug23: Zie voor het vervolg ook issue 70.

Status 31aug23: In behandeling.