

Estàndard de reutilització i obertura de dades

Estàndards de desenvolupament

Palma, setembre de 2025



**Vicepresidència Primera i Conselleria
d'Economia, Hisenda i Innovació**
Direcció General d'Estratègia Digital
i Desenvolupament Tecnològic



Índex

Historial de versions.....	3
1. Introducció.....	4
1.1. Objectius.....	4
2. API REST de reutilització i obertura de dades.....	6
2.1. Publicació i documentació de l'API REST.....	6
3. Serveis REST de reutilització i obertura de dades.....	8
3.1. Publicació i documentació dels serveis REST.....	8
3.2. Camps de sortida dels serveis REST.....	9
3.3. Paginació.....	12
3.4. Paràmetres d'entrada dels serveis REST.....	13
3.5. Validació de l'API i serveis REST.....	15
3.6. Guia i exemples d'implementació.....	15



Historial de versions

Data	Versió	Descripció	Autor
29/07/25	1.0	Redacció de l'estàndard	DGEDDT
16/09/25	1.1	Revisió de l'estàndard	DGEDDT
19/09/25	1.1	Publicació de l'estàndard	DGEDDT



1. Introducció

La reutilització i obertura de dades és un dels pilars fonamentals de la transformació digital de les administracions públiques. En aquest sentit, el Govern de les Illes Balears (GOIB) impulsa l'ús de serveis digitals que facilitin la consulta i reutilització de les seves dades, tant en àmbit intern com extern, mitjançant tecnologies basades en arquitectures obertes i formats estructurats. Aquest document defineix l'estàndard tècnic que han de complir les API REST i serveis REST per garantir una reutilització efectiva, segura i interoperable de les dades.

Aquest estàndard s'emmarca en el compliment de la **Directiva 2019/1024/UE** del Parlament Europeu i del Consell, de 20 de juny de 2019 relativa a les dades obertes i la reutilització de la informació del sector públic. També compleix amb la **Llei 37/2007**, de 16 de novembre, sobre reutilització de la informació del sector públic i en l'**Acord del Consell de Govern de 17 d'abril de 2023**, que estableix mesures per impulsar l'obertura i reutilització de les dades del Govern de les Illes Balears. L'objectiu d'aquesta normativa és fomentar la reutilització i obertura de les dades del Sector Públic per a finalitats socials, econòmiques i institucionals, per tal de promoure una administració més oberta, eficient i col·laborativa.

1.1. Objectius

L'estàndard de reutilització i obertura de dades definit en aquest document té com a finalitat establir les directrius tècniques comunes que han de seguir tots els serveis REST de reutilització i obertura de dades desenvolupats o gestionats per l'Administració de la Comunitat Autònoma de les Illes Balears (CAIB).

Els objectius principals d'aquest estàndard són:

- **Garantir l'accessibilitat i reutilització de les dades públiques** per part de ciutadania, empreses i altres administracions, mitjançant serveis digitals segurs, documentats i accessibles a través d'Internet.
- **Evitar sitges de dades i facilitar la reutilització interna de les dades** per part de la mateixa Administració per tal de millorar l'eficiència administrativa i optimitzar l'elaboració i l'avaluació de les polítiques públiques.
- **Estandarditzar la publicació i consum de dades** mitjançant API REST, amb estructures de dades coherents, metadades clares i formats oberts, principalment JSON.



- **Assegurar el compliment normatiu** en matèria de reutilització i obertura de dades en els àmbits europeu, estatal i autonòmic. També es vetllarà per garantir la protecció de dades personals i seguretat de la informació.
- **Impulsar la cultura de la dada oberta** com a recurs estratègic per al desenvolupament de serveis públics digitals, la innovació i la presa de decisions basada en dades.

2. API REST de reutilització i obertura de dades

Les aplicacions desenvolupades han de publicar serveis web que permetin la integració amb altres aplicacions i la reutilització de dades. Es restringeix l'ús de serveis SOAP en favor de serveis REST.

La publicació dels serveis REST es farà mitjançant una API REST. Es podran definir dos tipus d'API REST: externa i interna.

- **API externa.** S'hi publicaran els serveis REST accessibles des d'Internet que permetran la reutilització externa de les dades. Per defecte, els serveis REST de reutilització es publicaran en aquesta API externa.
- **API interna.** S'hi publicaran els serveis REST no accessibles des d'Internet que permetran la reutilització interna de les dades per part del GOIB. Per poder publicar un servei REST de reutilització en l'API interna, caldrà justificar-ne la necessitat i rebre l'autorització prèvia de la Direcció General d'Estratègia Digital i Desenvolupament Tecnològic (DGEDDT) del GOIB.

2.1. Publicació i documentació de l'API REST

- 1) Cada aplicació haurà de publicar la seva API REST (tant externa com interna) en el seu context, amb el format `/<codi-aplicació>api/` (per exemple, `/ibdonamapaapi/`).
- 2) L'API REST s'ha de documentar en català¹ mitjançant l'eina Swagger UI que ha de ser d'accés lliure sense cap mena de mecanisme d'autenticació o control d'accés.
- 3) El Swagger UI ha d'estar disponible a través d'una URL amb l'estructura següent: `https://<servidor>/<codi-aplicació>api/[interna|externa]`. Per

¹ Articles 21 i 22 del Decret 49/2018, de 21 de desembre, sobre l'ús de les llengües oficials en l'Administració de la Comunitat Autònoma de les Illes Balears.



exemple: «<https://intranet.caib.es/ibdonamapaapi/externa>» (no s'admetran altres estructures).

- 4) L'API REST ha de tenir un títol en majúscules que identifiqui el codi de l'aplicació, un guió (-) i el tipus d'API (API EXTERNA o API INTERNA). Un exemple seria, IBDONAMAPA – API EXTERNA.
- 5) També ha de mostrar una descripció de l'aplicació informàtica sobre la qual s'ha implementat l'API REST. Un exemple de descripció seria: Servei de digitalització del mapa de recursos i serveis per a l'atenció de dones víctimes de violència masclista de les Illes Balears.

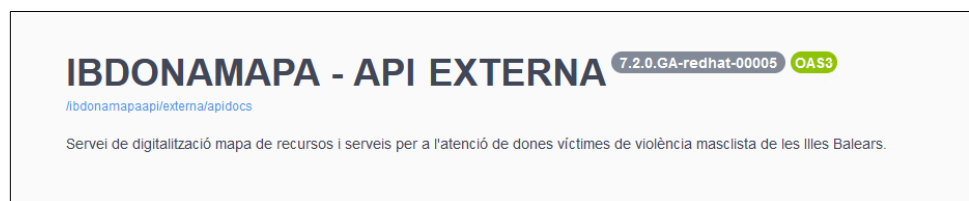


Figura 1: Exemple de descripció de l'API REST

- 6) El text de la pipella del navegador web ha de ser el mateix que el text del títol del Swagger UI. Exemple,

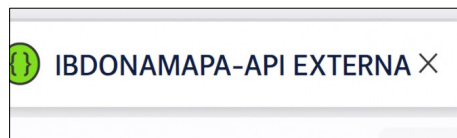


Figura 2: Exemple de nom de la pipella del navegador web.

- 7) Tots els serveis REST de reutilització publicats en el Swagger UI s'han de llistar agrupats en la mateixa secció (tag) anomenada **Serveis de reutilització**. Es mostra un exemple de com s'hauria de presentar la llista de serveis.



The screenshot shows the 'IBDONAMAPA - API REST EXTERNA' interface. At the top, there is a header with the title and version information (7.2.0.GA-redhat-00009) and an OAS3 logo. Below the header, there is a description of the service: 'Servel de digitalització mapa de recursos i serveis per a l'atenció de dones víctimes de violència masclista de les illes Balears.' A 'Servers' dropdown menu is visible, showing the URL 'https://Intranet.caib.es/lbdonamapaarpiexterna - Generated server ur!'. The main section is titled 'Serveis de reutilització' and contains three entries, each with a 'GET' method and a description:

Method	Endpoint	Description
GET	/reutilitzacio/recursos	Consulta paginada de recursos
GET	/reutilitzacio/edificis-tutelats	Consulta d'edificis tutelats per IB-Dona
GET	/reutilitzacio/ubicacions	Consulta paginada de ubicacions

Figura 3: Exemple de llista de serveis REST dins la secció Serveis de reutilització.

3. Serveis REST de reutilització i obertura de dades

En compliment dels principis recollits per la **Llei 37/2007**, de 16 de novembre, sobre reutilització de la informació del sector públic i l'**Acord de Govern, de 17 d'abril de 2023**, pel qual s'estableixen les mesures per impulsar l'obertura i reutilització de les dades del Govern de les Illes Balears és obligatori que els sistemes d'informació del GOIB estiguin dotats de serveis REST que permetin la reutilització, interna i/o externa, de les seves dades.

L'objectiu és evitar sistemes d'informació estancs que no puguin ser consultats i facilitar l'ús i reutilització de les dades que manegen. Aquests serveis REST han de permetre consultar les dades en un format obert, estructurat i de manera desagregada.

A continuació, s'exposen els requisits que han de complir els serveis REST de reutilització de dades. L'objectiu d'aquests requisits és estandarditzar i homogeneitzar la implementació d'aquests serveis REST.

3.1. Publicació i documentació dels serveis REST

- 1) Per defecte, els serveis REST de reutilització de dades s'han de publicar a l'API REST externa (accessible des d'Internet) i sense securitzar.



- 2) Si es volgués publicar el servei REST de reutilització en l'API interna (no accessible des d'Internet), caldrà justificar-ne la necessitat i rebre l'autorització de la DGEDDT.
- 3) En casos excepcionals, els serveis publicats en l'API interna podran estar securitzats mitjançant mecanismes d'autenticació i control d'accés. En aquest cas, se n'haurà de justificar la necessitat i rebre l'autorització de la DGEDDT.
- 4) Els serveis REST s'han de definir com a mètodes GET (NO com a mètodes POST).
- 5) Els serveis REST han de retornar les dades en format JSON.
- 6) La URL del servei REST de reutilització de dades ha d'incloure el nom del recurs que es consulta. Aquest nom ha d'estar escrit en català i en plural.
- 7) Aquesta URL ha de seguir l'estructura següent: `https://<servidor>/<codi-aplicació>api/[interna|externa]/reutilitzacio/<nom-recurs>`
«`https://www.caib.es/carpetaapi/externa/reutilitzacio/usuaris`»
- 8) Si el nom del recurs està compost per dues o més paraules s'han de separar per un guió (-). Per exemple,
«`https://www.caib.es/carpetaapi/externa/reutilitzacio/centres-concertats`»
- 9) Cada servei REST ha d'anar acompanyat d'una descripció completa i en català² de la seva funcionalitat.

3.2. Camps de sortida dels serveis REST

3.2.1. Camps de metadades

- 1) Les respostes dels serveis REST han de mostrar els següents 5 camps de metadades, en format *camelCase* i seguint el mateix ordre:
 1. **title:** nom del conjunt de dades (per exemple, Allotjaments turístics de les Illes Balears).
 2. **description:** descripció clara i concisa del conjunt de dades. S'explicarà què conté i per a què serveix el conjunt de dades.

² Articles 21 i 22 del Decret 49/2018, de 21 de desembre, sobre l'ús de les llengües oficials en l'Administració de la Comunitat Autònoma de les Illes Balears.



3. **spatial:** cobertura geogràfica de les dades. El contingut d'aquest camp contendrà un d'aquests valors: Illes Balears, Mallorca, Menorca, Eivissa o Formentera.
 4. **creator:** el [codi DIR3](#) de la unitat orgànica propietària de les dades.
 5. **dateDownload:** data de la descàrrega (en format ISO8601: 2022-07-26T12:58:55+02:00)
- 2) Tots els camps anteriors s'han d'agrupar dins el camp **metadata**, tal com es mostra en la figura següent:

```
{  
  "metadata": {  
    "title": "Allotjaments turístics Mallorca",  
    "description": "Dades dels allotjaments turístics registrats a l'illa de Mallorca.",  
    "spatial": "Mallorca",  
    "creator": "A04003894",  
    "dateDownload": "2025-07-02T16:00:00+02:00"  
  },  
}
```

Figura 4: Metadades del dataset.

3.2.2. Camps dels registres de dades

Els registres de dades que retorni la consulta estaran compostos per un conjunt de camps que hauran de complir les especificacions següents:

- 1) Els noms dels camps s'han d'escriure en català³ i en format *camelCase*.
- 2) Tots els registres de dades han de tenir els mateixos camps, tant si aquests venen informats com si no. L'estructura de camps de tots els registres ha de ser la mateixa.
- 3) Els noms dels camps no podran incloure cap dels caràcters especials de les URL: ! * ' () ; : @ & = + \$, / ? % # []
- 4) Els camps data s'hauran de definir amb el format ISO 8601. Exemple: 2023-07-26T12:58:55+02:00
- 5) Els camps booleans seran true o false.
- 6) Els camps float/double no faran servir separador de milers, però faran servir el punt («.») com a separador de decimals: 1234.56

³ Articles 21 i 22 del Decret 49/2018, de 21 de desembre, sobre l'ús de les llengües oficials en l'Administració de la Comunitat Autònoma de les Illes Balears.



- 7) Sempre que sigui possible, les dades hauran d'incloure els camps següents:
 - a) Identificador (exemple, NIF de persona jurídica).
 - b) Coordenades geogràfiques (latitud/longitud, utm(x)/utm(y)). El format d'aquests camps ha de ser numèric.
 - c) Camp de gènere (home/dona/no binari).
 - d) Camp de sexe (masculí/femení).
 - e) Data d'alta del registre de dades.
 - f) Data darrera actualització de les dades.
- 8) Si les dades fan referència a localitat, municipi⁴, illa, província o comunitat autònoma s'hauran de retornar amb dos camps:
 - a) El nom oficial.
 - b) El codi segons l'INE (Institut Nacional d'Estadística).
- 9) Per a tot camp codi que tenguí associat un camp descripció, s'hauran de retornar els dos camps (codi i descripció). El nom d'aquests camps hauran de seguir aquest esquema:
 - a) **codi**[NomCamp]. Per exemple, codiTipusDocument.
 - b) **descripcio**[NomCamp]. Per exemple, descripcioTipusDocument.
- 10) Per defecte, les dades de caràcter personal s'ofuscaran llevat que la finalitat per la qual s'hagin obtingut permeti la reutilització interna d'aquestes. (Exemple, NIF de persona física). En tot cas, la no-ofuscació d'un camp amb dades de caràcter personal caldrà l'autorització prèvia amb la DGEDDT.
- 11) La codificació dels caràcters dels camps de sortida ha de ser UTF-8. S'haurà de tenir en compte el que explica a l'apartat 3.6 d'aquest document.
- 12) Es recomana que els registres de dades retornats tinguin una estructura plana de camps, sense jerarquies. En el cas que les jerarquies facilitin la comprensió i l'enteniment de les dades retornades, es podran fer servir.

⁴ Els noms oficials dels municipis de les Illes Balears es poden consultar en el Catàleg de Dades Obertes GOIB: <https://intranet.caib.es/opendatacataleg/dataset/municipis>



13) Tots els camps dels registres de dades s'han d'agrupar dins el camp **data** del tipus llista (array), tal com es mostra en la figura següent:

```
{
  "metadata": {
    "title": "Allotjaments turístics Mallorca",
    "description": "Dades dels allotjaments turístics registrats a l'illa de Mallorca.",
    "spatial": "Mallorca",
    "creator": "A04003894",
    "dateDownload": "2025-07-02T16:00:00+02:00"
  },
  "data": [
    {
      "nom": "Hotel Bellavista",
      "dataAlta": "2023-05-12",
      "adreca": "Carrer de la Mar, 15, Palma",
      "illa": "Mallorca"
    },
    {
      "nom": "Apartaments Sol i Platja",
      "dataAlta": "2022-11-03",
      "adreca": "Passeig Marítim, 45, Alcúdia",
      "illa": "Mallorca"
    }
  ]
}
```

Figura 5: Estructura de dades plana, sense jerarquies

3.3. Paginació

- 1) En el cas que el servei REST retorni una gran quantitat de dades i això pugui comprometre el bon funcionament de l'aplicació, s'haurà de paginar la sortida de dades. En aquest cas, la resposta haurà d'incloure aquests 7 camps que informen de la paginació:
 1. **totalCount:** nombre total de registres de dades existents.
 2. **itemsReturned:** nombre de registres de dades retornats en una pàgina.
 3. **pageSize:** nombre de registres de dades que es sol·liciten per pàgina (la mida per defecte d'una pàgina serà com a mínim de 100 registres de dades).
 4. **totalPages:** nombre total de pàgines.
 5. **page:** número de la pàgina actual (la primera pàgina ha de començar en 1).



6. **nextUrl:** URL completa per accedir a la pàgina següent. Quan s'arribi a la darrera pàgina, el camp nextUrl haurà de ser null. La URL ha de ser navegable, exemple:
«<https://intranet.caib.es/cathosapi/externa/allotjaments-turistics?page=2&pageSize=50>».
7. **previousUrl:** URL completa per accedir a la pàgina anterior. Quan s'arribi a la primera pàgina, el camp previousUrl haurà de ser null.
- 2) Els camps **page** i **pageSize** hauran de ser paràmetres d'entrada opcionals de la consulta. El valor del paràmetre pageSize s'haurà de mantenir inalterat durant la navegació de la paginació. A continuació, mostrem un exemple d'URL amb aquests paràmetres:
«<https://intranet.caib.es/cathosapi/externa/allotjaments-turistics?page=2&pageSize=50>».
- 3) Tots els camps anteriors de paginació s'han d'agrupar dins el camp **metadata**, tal com es mostra en la figura següent:



```
{
  "metadata": {
    "title": "Allotjaments turístics Mallorca",
    "description": "Dades dels allotjaments turístics registrats a l'illa de Mallorca.",
    "spatial": "Mallorca",
    "creator": "A04003894",
    "dateDownload": "2025-07-02T16:00:00+02:00",
    "totalCount": 1250,
    "itemsReturned": 50,
    "pageSize": 50,
    "totalPages": 25,
    "page": 1,
    "nextUrl": "https://intranet.caib.es/cathosapi/externa/allotjaments-turistics?page=2&pageSi",
    "previousUrl": null
  },
  "data": [
    {
      "nom": "Hotel Bellavista",
      "dataAlta": "2023-05-12",
      "adreca": "Carrer de la Mar, 15, Palma",
      "illa": "Mallorca"
    },
    {
      "nom": "Apartaments Sol i Platja",
      "dataAlta": "2022-11-03",
      "adreca": "Passeig Marítim, 45, Alcúdia",
      "illa": "Mallorca"
    }
  ]
}
```

Figura 6: Metadades de paginació.

3.4. Paràmetres d'entrada dels serveis REST

Els serveis REST han de tenir definits paràmetres d'entrada per filtrar les dades que es consulten. Aquests paràmetres han de complir les característiques següents:

- 1) Els paràmetres d'entrada no han de tenir valors per defecte. Han de ser buits.
- 2) Tots els paràmetres d'entrada han de ser opcionals, no poden ser obligatoris. En el cas de voler definir paràmetres d'entrada obligatoris, caldrà justificar-ne l'obligatorietat i rebre l'autorització de la DGEDDT.
- 3) Els paràmetres d'entrada s'han de passar per URL, per exemple: ? clau1=valor1&clau2=valor2.



- 4) Els camps dels registres de dades de la resposta del servei REST (apartat 3.2.2) han de formar part del conjunt dels paràmetres d'entrada (exemples, id, nom, tipus, etc.). Els noms dels camps i dels paràmetres han de coincidir.
 - a) Els camps de tipus data han de tenir associats dos paràmetres d'entrada per tal de poder filtrar mitjançant un rang de dates (data inici i data fi). Exemple: dataPublicacioInici i dataPublicacioFi. Hauran d'estar en format ISO8601: 2022-07-26.
 - b) Els camps de tipus numèric han de tenir associats dos paràmetres d'entrada per tal de poder filtrar mitjançant un rang de valors (mínim i màxim). Exemple: importTotalMin i importTotalMax.
 - c) Els paràmetres float/double no faran servir separador de milers, però faran servir el punt («.») com a separador de decimals: 1234.56
 - d) No es poden definir paràmetres d'entrada a partir de camps amb dades personals.
- 5) Si les dades es retornen en més d'una llengua, s'ha d'afegir un paràmetre d'entrada que indiqui la llengua en què es retornen les dades. El valor d'aquest paràmetre haurà de seguir el format ISO 639-1, per exemple idioma=ca. Per defecte, si no s'indica cap valor per a aquest paràmetre les dades s'han de retornar en català.
- 6) Si la resposta està paginada, els camps pageSize i page s'hauran de definir com a paràmetres d'entrada. En el servidor se n'haurà de controlar els valors i, si aquests són null o no vàlids, s'hauran d'aplicar valors per defecte.
- 7) En el Swagger UI, els paràmetres d'entrada s'han de descriure en català⁵ de manera completa i precisa. Si tenen valors per defecte o un rang de valors concrets s'han d'indicar. Per exemple:

Indica l'idioma de la resposta: 'ca' (català) per defecte, o 'es' (espanyol).

Figura 7: Exemple de descripció completa i precisa.

⁵ Articles 21 i 22 del Decret 49/2018, de 21 de desembre, sobre l'ús de les llengües oficials en l'Administració de la Comunitat Autònoma de les Illes Balears.



3.5. *Validació de l'API i serveis REST*

- 1) Tant l'API com els serveis REST han de complir totes les especificacions definides en aquest document per ser aprovats pel Departament de Desenvolupament Tecnològic del Govern de les Illes Balears.
- 2) Si la sortida del servei REST està paginada s'ha de poder validar la paginació i la navegació entre pàgines. El joc de proves ha de ser suficient com per paginar com a mínim 3 pàgines. Es recorda que el nombre mínim de registres de dades per pàgina és de 100.
- 3) S'han de poder validar totes les metadades de la sortida del servei REST. Totes les metadades s'han d'informar amb dades reals.
- 4) S'han de poder validar tots els camps dels registres de dades. Tots els camps dels registres de dades han de venir informats de manera adient.

3.6. *Guia i exemples d'implementació*

- 1) Es recomana que cada aplicació tingui un client java que pugui consumir els serveis REST.
- 2) S'ha d'emprar la codificació UTF-8 per als caràcters de sortida dels serveis REST. Per implementar això, s'haurà de tenir en compte el següent:
 - a) Els fitxers .java hauran d'utilitzar el joc de caràcters UTF-8.
 - b) S'haurà d'afegir la següent etiqueta en tots els fitxers XML:

```
<?xml version="1.0" encoding="UTF-8"?>
```
 - c) S'haurà d'afegir la propietat següent en el fitxer pom.xml del projecte Maven pare:

```
<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
```
 - d) En els cas de les API REST, la codificació s'haurà d'afegir a la classe que genera la sortida del servei REST seguint el següent exemple:

```
String value = new String(myString.getBytes("UTF-8"), "UTF-8");
```
- 3) A banda, és recomanable revisar que:
 - a) La JVM que fa servir el JBoss tingui configurat el paràmetre següent:

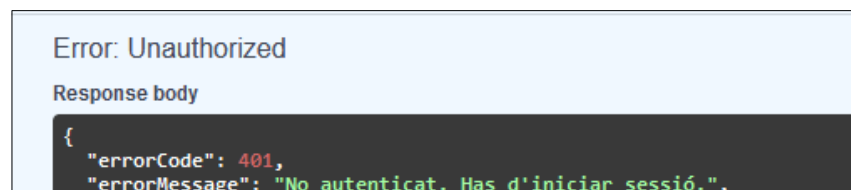
```
-Dfile.encoding=UTF-8
```



- b) També es recomana que les variables d'entorn del servidor que controlen la configuració regional i d'idioma tenguin configurat el charset com a UTF-8.
- 4) Si els serveis REST de reutilització i obertura de dades definits a l'API interna estan securitzats mitjançant mecanismes d'autenticació i control d'accés, s'hauran de tenir en compte els següents punts:
- a) Crear un usuari d'aplicació per poder-los executar. El nom i els rols assignats a aquest usuari dependrà de l'entorn en què es creï. A continuació, es mostra una taula amb els noms i rols que haurien de tenir segons l'entorn:

Entorn	Nom usuari	Rols
Preproducció	\$opendata_<codi-aplicació>_pre	<PREFIX-APLICACIÓ>_API_EXTERNA i/o <PREFIX-APLICACIÓ>_API_INTERNA
Producció	\$opendata_<codi-aplicació>	<PREFIX-APLICACIÓ>_API_EXTERNA i/o <PREFIX-APLICACIÓ>_API_INTERNA

- b) Si un usuari sense permisos intenta executar un servei REST autenticat, s'haurà de mostrar un avís d'aquest tipus:



```
Error: Unauthorized
Response body
{
  "errorCode": 401,
  "errorMessage": "No autenticat. Has d'iniciar sessió.",
}
```

Figura 8: Exemple d'error en l'accés i execució d'un servei REST.