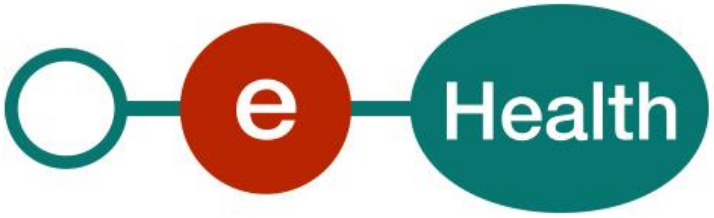


eHealth Consent WS - REST
Cookbook
Version 1.1

This document is provided to you, free of charge, by the

eHealth platform

Willebroekkaai 38

38, Quai de Willebroek

1000 BRUSSELS

Table of contents

To the attention of: “IT expert”, willing to integrate this web service.

1. Document management

1.1 Document history

Version	Date	Author	Description of changes / remarks
1.0	06/07/2022	eHealth platform	Initial version
1.1	03/08/2022	eHealth platform	§ 5.1.2 Tracing (added)

2. Introduction

2.1 Goal of the service

The existence of an active '*informed patient consent*' is one of the fundamental prerequisites for the healthcare providers to access patient's medical data. Therefore, the eHealth platform makes available to the concerned patients and the health care actors involved in the exchange, storage or referencing personal data a service to manage the '*informed patient consent*' as defined by the deliberation 12/047 of the CSSSS/SCSZG[1].

Technically, we identify the following attributes for an 'informed patient consent':

- The SSIN of the patient.
- The date of the consent registration (at the end-user side).
- The "type" of the consent.

If the consent is only valuable[2] for data posterior to the signing date, it is called '*prospective*' and otherwise '*retrospective*'[3]. According to the rules defined now, the only possible value for this attribute is '*retrospective'.* The attribute is present for backwards compatibility.

- The identity of the HCParty acting in the patient's name (if applicable).

The following operations will support the management of 'the informed patient consent':

POST /consents/{patientSsin}	Allows an end-user to declare an informed patient consent. Once the consent is successfully declared, it is considered an *'active'* consent.
DELETE /consents/{patientSsin}	Allows an end-user to declare the revocation of an informed patient consent. Once the consent is revoked, it is considered an *'inactive'* consent.
GET /consents/{patientSsin}	Allows an end-user to check the existence of an informed patient consent and to get the information about this consent (this consent can be '*inactive' 'active', 'revoked' or the patient is 'deceased'*).
GET /histories/{patientSsin}	Allows an end-user to consult the informed patient consent management history.

2.2 The consent is an "informed consent"

The WS only supports the technical registration of the consent. The information of the patient is the responsibility of the caller of the service.

2.3 Goal of the document

This document describes the management of the 'informed patient consent' service as provided by the eHealth platform. In this cookbook, we explain the structure and content aspects of the possible requests and the replies of eHealth Consent WS. An example illustrates each of those messages. In addition, a list of possible errors can be found in this document.

This information should allow (the IT department of) an organization to develop and use the WS call.

---

[1] ***https://www.ehealth.fgov.be/ehealthplatform***

[2] *At the level of the transaction, the date to be taken into account is the 'medical date' of the transaction.*

[3] *This does not mean that all documents with a medical date anterior to the signing date of the consent will automatically be made available.*

Some technical and legal requirements must be met in order to allow the integration of the eHealth WSs in client applications.

This document is neither a development nor a programming guide for internal applications; eHealth partners always keep total freedom within those fields. Nevertheless, in order to interact in a smooth, homogeneous and risk controlled way with a maximum of partners, eHealth partners must commit to comply with specifications, data format, and release processes described within this document. In addition, our partners in the health sector must also comply with the business rules of validation and integration of data within their applications as to minimize errors and incidents.

2.4 eHealth document references

All the document references can be found on the eHealth platform portal[4]. These versions or any following versions can be used for the eHealth platform service.

ID	Title	Version	Date	Author
1	eHealth Services – Web Access	2.0	12/07/2018	eHealth platform
2	I.AM Connect - Mobile integration - Technical specifications	1.5	26/06/2021	eHealth platform
3	I.AM Connect – HealthCare Client Registration	1.2	28/06/2021	eHealth platform
4	I.AM Connect – M2M Client registration	1.1	07/07/2021	eHealth platform
5	SOA – Error guide	1.0	10/06/2021	eHealth platform
6	Request test case template	3.0	22/02/2018	eHealth platform
7	IAM Connect Claim mappers	1.0	28/05/2021	eHealth platform

[4] ***https://www.ehealth.fgov.be/ehealthplatform***

3. Support

3.1 Helpdesk eHealth platform

3.1.1 Certificates

In order to access the secured eHealth platform environment you have to obtain an eHealth platform certificate, used to identify the initiator of the request. In case you do not have one, please consult the chapter about the eHealth Certificates on the portal of the eHealth platform

- ***https://www.ehealth.fgov.be/ehealthplatform/nl/ehealth-certificaten***
- ***https://www.ehealth.fgov.be/ehealthplatform/fr/certificats-ehealth***

For technical issues regarding eHealth platform certificates

- Acceptance: ***acceptance-certificates@ehealth.fgov.be***
- Production: ***support@ehealth.fgov.be***

3.1.2 For issues in production

eHealth platform contact centre:

- Phone: 02 788 51 55 (on working days from 7 am till 8 pm)
- Mail: ***support@ehealth.fgov.be***
- *Contact Form :*
  - ***https://www.ehealth.fgov.be/ehealthplatform/nl/contact*** (Dutch)
  - ***https://www.ehealth.fgov.be/ehealthplatform/fr/contact*** (French)

3.1.3 For issues in acceptance

Integration-support@ehealth.fgov.be

3.1.4 For business issues

- regarding an existing project: the project manager in charge of the application or service
- regarding a new project or other business issues: ***info@ehealth.fgov.be***

3.2 Status

The website ***https://status.ehealth.fgov.be*** is the monitoring and information tool for the ICT functioning of the eHealth services that are partners of the Belgian eHealth system.

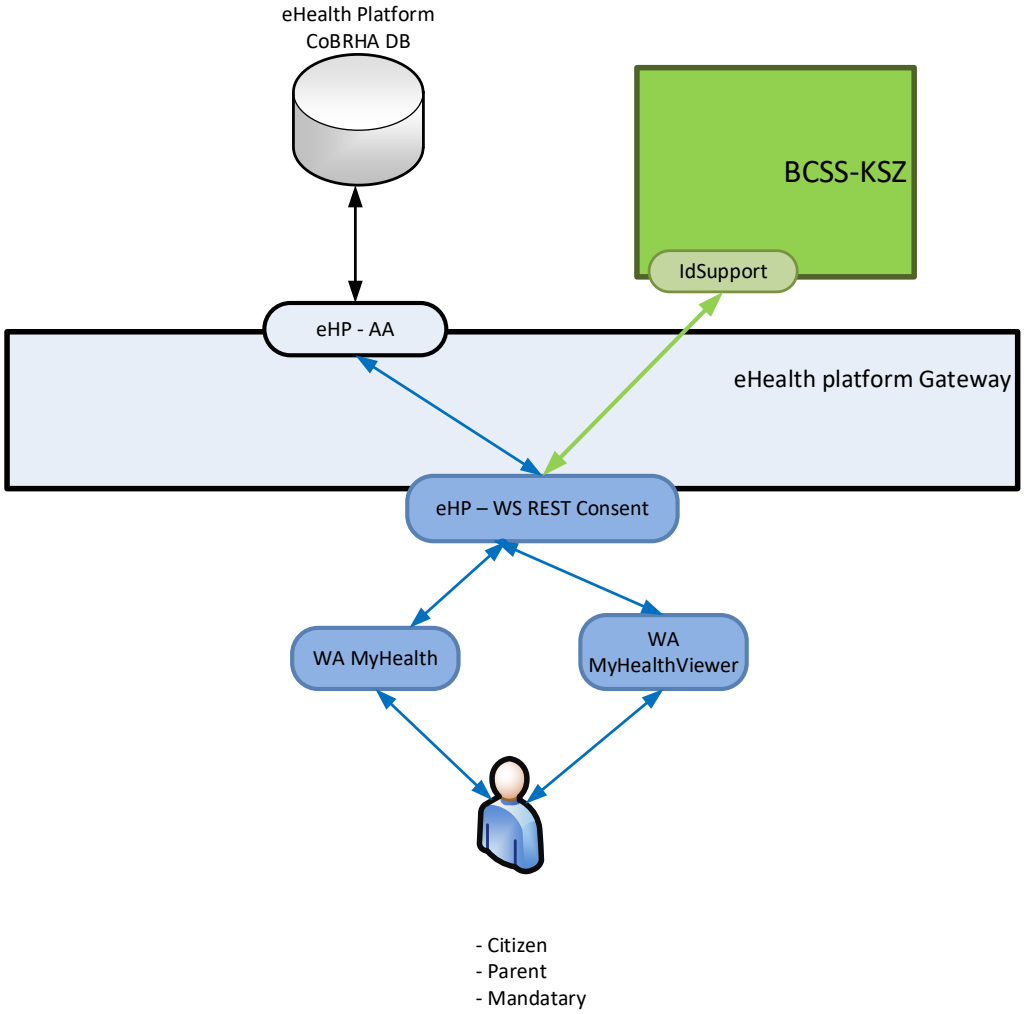
3.3 I.AM Connect

In order to use the Consent REST service you have to obtain an "Access token" which is delivered through I.AM Connect. You can find more information about I.AM Connect and how to register a client in I.AM Connect on the I.AM eHealth portal page:

Dutch version:
https://www.ehealth.fgov.be/ehealthplatform/nl/service-iam-identity-access-management

French version:
https://www.ehealth.fgov.be/ehealthplatform/fr/service-iam-identity-access-management

4. Global overview



This schema gives the users of this cookbook an overview of the management of the informed patient consent.

The MyHealth WA and MyHealthViewer WA will allow patients (incl. patient's parents or mandataries) to manage the informed consent (declaration, revocation, and consultation) by calling the eHealth Consent WS (REST).

5. Step-by-step

5.1 Technical requirements

5.1.1 eHealth platform Authentication

As explained previously, to use the Consent Rest service, you must have an access token delivered through I.AM Connect.

Several roles and profiles are defined for the using of the Consent Rest service.

Possible roles :

- REST-access : This role must be present in the access token for a end-user in order to use the POST, GET and DELETE methods of the service.
- monitoring : This role must be present in the access token in order to use the monitoring methods (/health) of the service.

Presentation of the roles and profiles in the access token:

```
"resource_access":
{
        "ehealth-consent-backend":
        {
                "roles":
                [
                "rest-access"
                ]
        }
}
```

The rest-access role can only be assigned to profiles authorized to use the WS Consent REST, namely:

- Citizen
- Parent
- Mandatary (only for mandate type “medicaldatamanagement”)

For more information on how to get an access token or information on access token structure (profile), please refer to section 2.4 of this cookbook.

5.1.2 Tracing

To use this service, the request SHOULD contain the following two http header values (see RFC ***https://datatracker.ietf.org/doc/html/rfc7231#section-5.5.3***):

1. User-Agent: information identifying the software product and underlying technical stack/platform. It MUST include the minimal identification information of the software such that the emergency contact (see below) can uniquely identify the component.
   a. Pattern: {minimal software information}/{version} {minimal connector information}/{connector-package-version}
   b. Regular expression for each subset (separated by a space) of the pattern: [[a-zA-Z0-9-\/]*\/[0-9azA-Z-_.]*
   c. Examples:
   User-Agent: myProduct/62.310.4 Technical/3.19.0
   User-Agent: Topaz-XXXX/123.23.X freeconnector/XXXXX.XXX
2. From: email-address that can be used for emergency contact in case of an operational problem.
   Examples:

From: ***info@mycompany.be***

5.2 The Consent REST Service

The REST interface is described with a JSON/ Swagger API available on the eHealthConsent eHealth portal page:

Dutch version:
https://www.ehealth.fgov.be/ehealthplatform/nl/service-ehealthconsent
French version:
https://www.ehealth.fgov.be/ehealthplatform/fr/service-ehealthconsent

The Consent WS has the following endpoints :

- Acceptance environment: ***https://api-acpt.ehealth.fgov.be/consent/v2***
- Production environment: ***https://api.ehealth.fgov.be/consent/v2***

5.3 POST /consents/{patientSsin}

This method allows to activate the consent of a patient.

Request

pathParameter	Description
patientSsin (mandatory)	Social Security Identification Number (SSIN) of the patient concerned by the consent.

queryParameter	Description
patientCardNumber (optional)	The support card number which can be : - The e-ID card number - The ISI+ card number The element *'patientCardNumber'* is optional and should not be used if the end user is the patient, the mandatary of a patient or the parent of a patient.

Examples:

```
POST /consent/v2/consents/12345678910
```

Response

Success:

HTTP code 201 (Created) returned in case of success for the activation of the consent.

Error:

HTTP code 409 (Conflict) returned in case of an consent already existing or the patient is deceased.

Failures:

For failure description please refer to section 8 of this cookbook.

5.4 DELETE /consents/{patientSsin}

This method allows to revoke the consent of a patient.

Request

pathParameter	Description
patientSsin (mandatory)	Social Security Identification Number (SSIN) of the patient concerned by the consent.
queryParameter	Description
patientCardNumber (optional)	The support card number which can be : - The e-ID card number - The ISI+ card number The element *'patientCardNumber'* is optional and should not be used if the end user is the patient, the mandatary of a patient or the parent of a patient.

Example:

```
DELETE /consent/v2/consents/12345678910
```

Response

Success:

HTTP code 204 (No Content) returned in case of success for the revocation of the consent.

Error:

HTTP code 404 (Not Found) returned in case of no existing consent matching the given arguments.

HTTP code 409 (Conflict) returned if the patient is deceased.

Failures:

For failure description please refer to section 8 of this cookbook.

5.5 GET /consents/{patientSsin}

This method allows to consult the consent of a patient.

Request

pathParameter	Description
patientSsin (mandatory)	Social Security Identification Number (SSIN) of the patient concerned by the consent.

Example:

```
GET /consent/v2/consents/12345678910
```

Response

Success:

HTTP code 200 (OK) returned in case of success for the consultation of the consent.

In this case, the consent is found with a status revoked, active or deceased and the body contains the following consent information:

- Identification of the concerned patient.
- The date of the declaration of patient consent.
- The date of the revocation of patient consent.
- The status of the consent (GIVEN / REVOKED / DECEASED)

Examples :

- Patient with active consent


```
{
  "patient": {
    "identifier": [
      {
        "type": "ssin",
        "value": "12345678910"
      }
    ]
  },
  "signDate": "2022-05-30",
  "revokeDate": null,
  "status": "GIVEN"
}
```


- Patient with inactive (revoked) consent


```
{
  "patient": {
    "identifier": [
      {
        "type": "ssin",
        "value": "12345678910"
      }
    ]
  },
  "signDate": "2022-05-30",
  "revokeDate": "2022-05-30",
  "status": "REVOKED"
}
```


- Patient with inactive (deceased) consent


```
{
  "patient": {
    "identifier": [
      {
        "type": "ssin",
        "value": "12345678910"
      }
    ]
  },
  "signDate": "2022-05-30",
  "revokeDate": null,
  "status": "DECEASED"
}
```


Error:

HTTP code 404 (Not Found) returned in case of no consent information matching the given arguments.

Failures:

For failure description please refer to section 8 of this cookbook.

5.6 GET /histories/{patientSsin}

The requests of type “GET” on path /histories allow the user to consult the informed patient consent management history ie the list of information relating to changes in informed consent status.

For each status change, information about the author, about the patient, about the date and time as well as about the operation are returned.

Request

pathParameter	Description
patientSsin (mandatory)	Social Security Identification Number (SSIN) of the patient concerned by the consent.
queryParameter	Description
pageSize (optional)	The parameter allowing to limit the number of entries in the history. The entries are organized in descending chronological order and if the 'pageSize' parameter is not specified then all consent status changes are returned (limited to a maximum of 1500 entries).

Example:

```
GET consent/v2/histories/12345678910?pageSize=2
```

Response

Success:

HTTP code 200 (OK) returned in case of success for the consultation of the consent management history.

In this case, at least one entry is found and the body contains the list of all entries found (limited by the *'pageSize'* parameter if specified).

Each entry contains the following consent information:

- The author responsible for the status change represented by a sequence of ‘identifier’ elements.
- The date and time of the change.
- The operation (REVOKE_CONSENT / DECLARE_CONSENT).

Examples :

- Consult the history with pageSize limited to 2 entries


```
[
 {
   "author": [
     {
       "identifier": [
         {
           "type": "local",
           "value": "1990000332"
         }
       ],
       "name": "eHealth Consent",
       "firstName": null,
       "qualificationCode": "application"
     },
     {
       "identifier": [
         {
```

```
            "type": "ssin",
            "value": "12345678910"
          }
        ],
        "name": null,
        "firstName": null,
        "qualificationCode": "patient"
      }
    ],
    "timestamp": "2022-05-30T09:23:43+02:00",
    "operation": "REVOKE_CONSENT"
  },
  {
    "author": [
      {
        "identifier": [
          {
            "type": "local",
            "value": "1990000332"
          }
        ],
        "name": "eHealth Consent",
        "firstName": null,
        "qualificationCode": "application"
      },
      {
        "identifier": [
          {
            "type": "ssin",
            "value": "12345678910"
          }
        ],
        "name": null,
        "firstName": null,
        "qualificationCode": "patient"
      }
    ],
    "timestamp": "2022-05-30T09:14:04+02:00",
    "operation": "DECLARE_CONSENT"
  }
]
```


Error:

HTTP code 404 (Not Found) returned in case of no consent history information matching the given arguments.

Failures:

For failure description please refer to section 8 of this cookbook.

6. Risks and security

6.1 Risks & safety

6.2 Security

6.2.1 Business security

In case the development adds an additional use case based on an existing integration, the eHealth platform must be informed at least one month in advance with a detailed estimate of the expected load. This will ensure an effective capacity management.

When technical issues occur on the WS, the partner can obtain support from the contact center (see Chap 3).

If the eHealth platform should find a bug or vulnerability in its software, the partner must update his application with the latest version of the software, within ten (10) business days.

If the partner finds a bug or vulnerability in the software or web service made available by the eHealth platform, he is obliged to contact and inform us immediately. He is not allowed, under any circumstances, to publish this bug or vulnerability.

6.2.2 The use of username, password and token

The username, password, and token are strictly personal.
Every user takes care of his username, password and token, and he is forced to confidentiality of it. It is prohibited to transfer them to partners and clients. Until inactivation, every user is responsible for every use, including the use by a third party.

7. Implementation aspects

7.1 Procedure

This chapter explains the procedures for testing and releasing an application in acceptation or production.

7.1.1 Initiation

If you intend to use the eHealth platform service, please contact ***info@ehealth.fgov.be***. The project department will provide you with the necessary information and mandatory documents.

7.1.2 Development and test procedure

You have to develop a client in order to connect to our WS. Most of the information needed to integrate is published on the portal of the eHealth platform.

Upon request, the eHealth platform provides you with test cases (see ***Request Test Case template***) in order for you to test your client before releasing it in the acceptance environment.

7.1.3 Release procedure

When development tests are successful, you can request to access the acceptance environment of the eHealth platform. From this moment, you start the integration and acceptance tests. The eHealth platform suggests testing during minimum one month.

After successful acceptance tests, the partner sends his test and performance results with a sample of “eHealth request” and “eHealth answer” by email to his point of contact at the eHealth platform.

Once a release date has been agreed on, the eHealth platform prepares the connection to the production environment and provides the partner with the necessary information. During the release day, the partner provides the eHealth platform with feedback on the test and performance tests.

For further information and instructions, please contact: ***integration-support@ehealth.fgov.be***

7.1.4 Operational follow-up

Once in production, the partner using the eHealth platform service for one of his applications will always test in the acceptance environment first, before releasing any adaptations of his application in production. In addition, he will inform the eHealth platform on the progress and test period.

7.2 Test cases

eHealth recommends performing tests for all of the following cases:

- POST /consents : successful declaration of an informed patient consent
- DELETE /consents : successful revocation of an informed patient consent
- GET /consents : successful consultation status of an informed patient consent
- GET /histories : successful consultation history
- In addition, you should also run negative test cases.

8. Error and failure messages

8.1 HTTP codes

8.1.1 HTTP 2xx

The meaning of 2xx HTTP codes is described in each Operation in section 5 of this document.

8.1.2 HTTP 500

This code means that an internal technical error occurred during the processing of the request.
This does not necessarily mean that the error is not due to a wrong input in the request.
Please double-check that your request is correct before reaching to the helpdesk.

8.1.3 HTTP 400

This code means that the request could not be performed due to a validation error.
The body returned explains what went wrong in your request.
Example:

```
[
  {
    "code": "VAL002",
    "message": "The provided inss has a wrong checksum."
  }
]
```

Here are the possible values that you can get in the body for the “400” errors:

BIZ001	"Consent already exists."
BIZ002	"No Consent found."
BIZ003	"The provided patient ssin: {$[patient.ssin]} is different than patient ssin in token: {$[JWT.patient.ssin]}"
BIZ004	" The consent of a deceased patient cannot be modified."
VAL002	" The provided patient ssin: ${patient_id_invalid_checksum} has an incorrect checksum."
VAL002	"The provided patient ssin: {$[patient.ssin]} must only contain digits."
VAL002	" The provided patient ssin: {$[patient.ssin]} has an incorrect length. Length should be 11. Got {$[patient.ssin.length()]}."
VAL002	" The provided patient ssin: {$[patient.ssin]} is malformed."
VAL003	" The cardNumber must be specified for professionals."
VAL004	" The provided cardNumber: {$[Patient.cardNumber]} is not valid: ${IDSupport.error}"
VAL011	" The provided page size: {$[pageSize]} is incorrect. It should be strictly positive."